Université de Toulouse Paul-Sabatier
IMT (Institut de Mathématiques de Toulouse)

Habilitation à diriger des recherche

présentée en première version en vu d'obtenir le grade de Docteur habilité, spécialité « Mathématiques appliquées »

par

Raphaël Loubère

Contribution au domaine des méthodes numériques Lagrangiennes et Arbitrary-Lagrangian-Eulerian

Habilitation soutenue le 28 Juin 2013 devant le jury composé de :

M.	Andy Barlow	Atomic Weapon Establishment, UK	(Rapporteur)
M.	Daniel Bouche	CEA-DIF, France	(Examinateur)
M.	Stéphane Clain	Universitade Do Minho, Portugal	(Examinateur)
M.	Frédéric Coquel	Université Paris VI, France	(Rapporteur)
M.	Pierre Degond	Université Paul-Sabatier Toulouse 3, France	(Directeur)
M.	Patrick Hild	Université Paul-Sabatier Toulouse 3, France	(Examinateur)
M.	Richard Liska	Czech Technical University in Prague, République Tchèque	(Examinateur)
M.	Pierre-Henri Maire	CEA-CESTA, France	(Examinateur)
M.	Philip Roe	University Michigan, U.S.A	(Rapporteur)
M.	Richard Saurel	IUF Université Aix Marseille 1, France	(Examinateur)

A Margaux, Calixte et Clotilde.

Socrates once said, "I swear it upon Zeus an outstanding runner cannot be the equal of an average wrestler."

From the last day of my student time up to now, at the mathematical laboratory of Bordeaux university, the CEA, the Los Alamos National Laboratory in the New-Mexican desert, the Mathematical institute in Toulouse, I came across so many outstanding persons. Some of them ultimately became friends of mine and I am particularly proud of it. Let me use this acknowledgment section to praise them the best way I can.

First of all I would like to warmly thank Philip Roe, Frédéric. Coquel and Andrew J. Barlow for carefully reviewing this habilitation. This is a thankless job and the fact that they spent some of their free time on this thesis is genuinely appreciated — especially because they are important scientists with busy life.
Also I would like to thank Daniel Bouche, Stéphane Clain, Patrick Hild, Richard Liska, Pierre-Henri Maire, and Richard Saurel for being member of this jury : Their presence is greatly appreciated. I am also indebted to Pierre Degond who accepted to supervise this work and be part of the jury. During my stay in the U.S.A I met many important persons who helped me and inspired me. Amongst many of them stand Mikhail Shashkov who mentored me for three years back then. He introduced me to this new world of ALE and I am very grateful for his support and the possibility he offered me to parasite his aura.
I am also very grateful to Burton Wendroff with whom I had the chance to share an office at Los Alamos. Let me put that straight : Burt is a legend. Sharing time, stories, discussions and working with him were some of great moments of my scientific life mostly because behind the legend there is a adorable man, even greater than the legend.
The last very important scientific person I had to meet in the U.S.A was Ed J. Caramana. During a difficult time for Los Alamos National Laboratory I enjoyed his vehement diatribe against "the system" in general but also his passionate faith for the compatible staggered Lagrangian scheme. He was kind enough to spend hours trying to teach me each and every subtle details of this scheme, which sometimes, I can confess now, were not entirely clear to me. Nevertheless he was truly patient and associated me with his investigations so that, once again, I was shining through him.
Of course I can not forget the sympathetic atmosphere of the ex-T-7 team hold by very friendly colleagues plus some others whom I met during intense working session or simply to share an espresso with or without sugar.
My life could not be as pleasant as it is now without the unfailing support from the Czech Republic. Well, maybe not the entire country, but at least three good Czech fellows, Richard Liska, Pavel Váchal and Milan Kuchařík. Scientifically and personally it is always a pleasure to meet them.
Back to France I wish to acknowledge the scientific support of Remi Abgrall since the time of my PhD up to now.
Presumably the person I am the most indebted to is Pierre-Henri Maire. Mainly because he has accepted to pair up with me for many different research topics. In some sense we are a kind of improbable scientific couple ; he is methodical and rigorous while I do everything with too much artistic freedom, he has bright ideas while I have stupid bugs, he is all dressed up to the nines while I still dress up like a teenager, etc. But despite that, I am proud of the work we have done.
While Pierre-Henri was a team leader at CELIA (University of Bordeaux) he made me meet Jérôme Breil and Stéphane Galera with whom we actively collaborated. I had the chance to meet Vladimir Tikhonchuk (CELIA) during my PhD, this was during previous century. He was literally building a team mixing physics and numerics and did a great job. Very supportive from the beginning he is surely at the origin of my work keeping contact with real applications. Jean Ovadia did supervise my PhD from the engineering point of view and, I always look back at this time with good and refreshing souvenirs, I would like to use this opportunity to acknowledge his still-on-going-spiritual

contribution. It is often scary to realize by how much one mimics some of our fathers !
In Toulouse, since 2006 I had the chance to collaborate and interact on personal basis with many colleagues at IMT ; some of them became co-workers, some others partners for coffee, which is equally important for me. More specifically I would like to praise Fabrice Deluzet, Marie-Hélène Vignal, Alexei Lozinski and Giacomo Dimarco for their contributions as collaborators but also as friends. I am also indebted to Stéphane Clain who used to be the 'colleague next door' and is now my best address in Portugal. We used to collaborate in Toulouse while he was at IMT, and we still do since he arrived in Braga, Portugal. Not so long ago we were both proud when Steven Diot (our shared PhD student) has defended not because of our contribution, most of the job has been done by Steven, but for the probable good (enough) supervision we were able to provide. I wish we will pursue on this path. By the way I wish to thank Steven for the good MOOD he breathed into our common scientific lives.
Then there is a whole crowd of collaborators and friends from Paris which I would like to thank for their warm welcome each time I have to spend time there. Especially Philippe Hoch whom I often bother evenings and nights in Paris, also Jean-Philippe Braeunig for good time out and also letting me participating to the development of his code.
I would like to take the opportunity to warmly thank Renaud Motte from CEA and his whole team (more specifically the ones with whom I interact : Alexandra, Christophe, Corinne, Jean-Philippe, Laurent, Mathieu, Pascal and Raphaël), I wish we will pursue this collaboration for many more years. Also I would like to thank the CEA/DAM-DIF in general for a true support during the last six years which made my scientific life fairly easy : this is truly appreciated.
Should it be in Paris or Cargèse (Corsica) it is always a pleasure to work, interact, or simply exchange inappropriate jokes with Jean-Michel Ghidaglia. Thanks to him I have been introduced to the CMLA crowd where I met some of his co-workers. Also I was welcomed into some of his projects. Especially the connection with Florian de Vuyst is particularly fruitful and I genuinely hope that this is only the beginning of some good science with holograms and other 'geeky gadgets'.
Of course none of the above great moments would be possible without the stable basis brought by my in-laws and my family. I would like to deeply thank them for begin always supportive and helpful. Without any doubt they are more important than the number of written sentences of this paragraph dedicated to them.
Finally I am mostly grateful to Clotilde for her sure and constant support for these years. Not only she has accepted to marry (and have home 24/7) a pale reproduction of a character from "The big Bang Theory" series but she has also agreed to have two wonderful kids with me. Sometimes I wonder if she is not as much disturbed as I am.

Enfin à Margaux ma Princesse, et Calixte mon Champion merci de remplir mes jours de bonheur et d'insouciance.

Toulouse, le 23 juin 2013.

Contents

Prologue and general introduction

This prologue is intended to provide some guide lines in order for the reader to follow my peregrinations, my collaborations and recast the publication list within this historical context.

Institut de physique fondamentale de Bordeaux, France
Centre d'Etude Lasers Intenses et Applications (CELIA)
November 2002 - January 2003

After the defense of a PhD from the University in Bordeaux in October 2002 [1] with R. Abgrall (INRIA Bordeaux) and J. Ovadia (retired fellow from CEA-CESTA) as supervisors, I have spent three months at the Fundamental Physics department under the supervision of V. Tikhonchuk (University of Bordeaux, CELIA laboratory). During this time I mostly interacted with S. Weber. The development of the cell-centered Lagrangian numerical scheme from [1, 2] has been pursued in a laser-plasma interaction context. This scheme has been coupled during this time to physics models in order to build a parallel transport simulation code for Inertial Confinement Fusion (ICF) [3, 4].

Los Alamos National Laboratory (LANL), New Mexico, U.S.A
Theoretical division - Mathematical, Modelisation and Analysis Group
February 2003 - December 2005

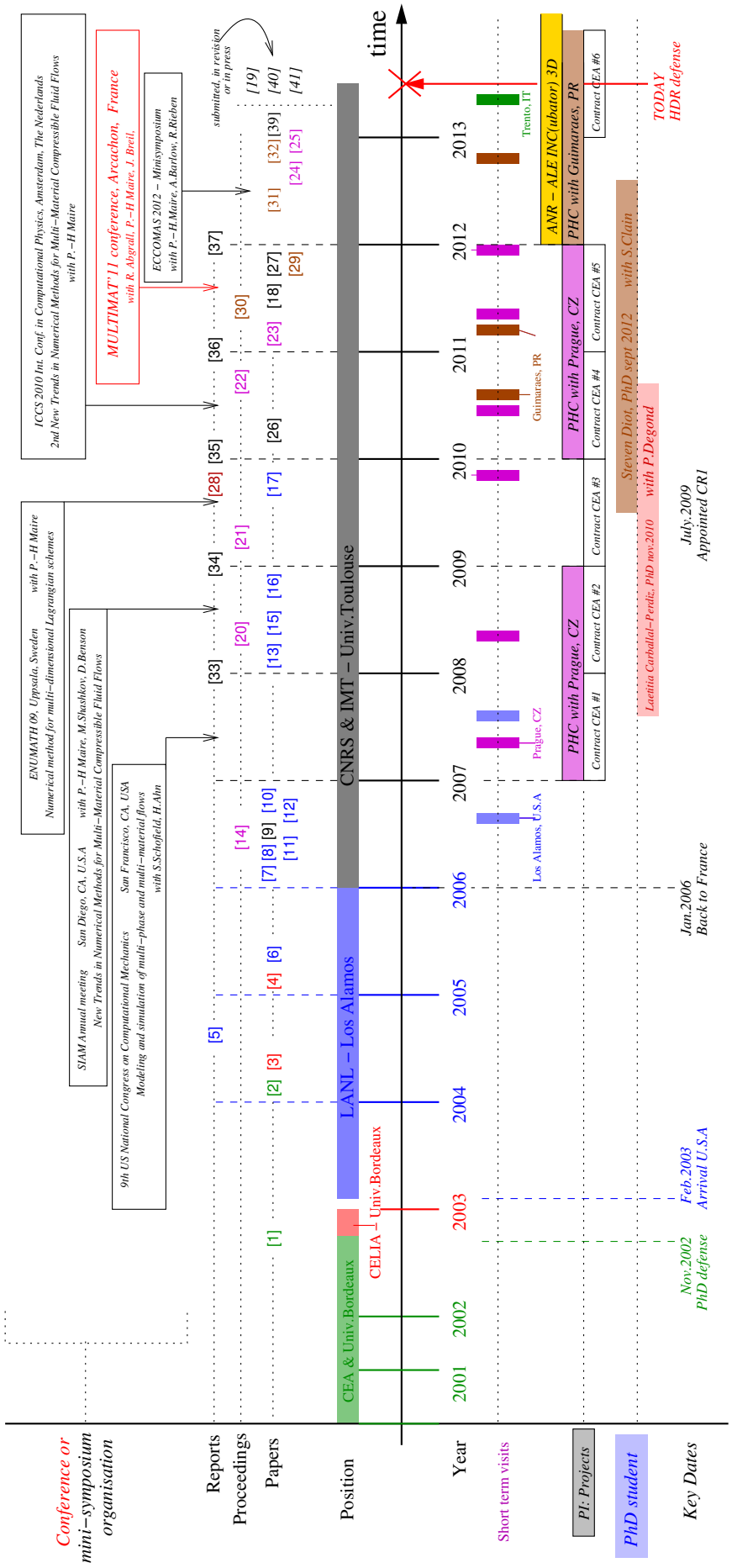
In February 2002 I have started a postdoctoral position at the Los Alamos National Laboratory (LANL) in New Mexico, U.S.A, under the supervision of M. Shashkov (a.k.a Misha). Most of my coauthors from this period are/were staff members at LANL or long/short term visitors (summer students, PhD students, postdocs) or invited professors.
Prior to my arrival at LANL a former postdoc of Misha, J. Campbell had developed a compatible staggered Lagrangian scheme in 2D on unstructured meshes. I have used this code to build an Arbitrary-Lagrangian-Eulerian simulation code for hydrodynamics equations called ALE INC(ubator) [5, 6]. More precisely I have implemented the untangling and rezone capabilities and developed with Misha a conservative remap module [7] and associated repair methods [8] with M. Staley and B. Wendroff (a.k.a. Burt). With B. Despres, summer visitor at LANL we have also written a 1D analysis of repair methods in [9].
In parallel E. J. Caramana associated me to his research on the compatible staggered Lagrangian scheme implemented in ALE INC(ubator). We have investigated the treatment of exceptional points (known also as T junction, dendritic zones) [10] and a vorticity damping artificial viscosity [11]. Later a joint effort with A. L. Bauer, D. E. Burton, M. J. Shashkov and P. P. Whalen gave birth to an

article dealing mostly with the analysis of this Lagrangian scheme in term of consistency, stability and accuracy in [12].
While developing multi-material capability into ALE INC(ubator) I have faced the situation of inadequate interface reconstruction method when three or more materials are present within the same mixed cell. Consequently with some of my colleagues of the ex-T-7 group at LANL, namely S.P. Schofield, R.V. Garimella and M.M. Francois, we have developed a material order independent interface reconstruction method using power diagrams in [13] which has however been published only in 2008.
Using the Czech connection of M. Shashkov I was lucky enough to meet M. Kuchařík, R. Liska and P. Váchal during their summer visits and we have started a never-ending (up to now) collaboration on ALE and Lagrangian numerical methods [14].
In December 2005, at the end of this postdoctoral position at LANL, I was involved in three main subjects of research : unstructured ALE simulation code development [5,6] (meshing, rezoning, remapping, repair), staggered Lagrangian scheme analysis and advanced interface reconstruction methods.

CNRS and Institut de Mathématiques de Toulouse (IMT), France
Mathématiques pour l'Industrie et la Physique (MIP) group
January 2006 - now

In January 2006 I was appointed as researcher at the Centre National de Recherche Scientifique (CNRS) at the Mathematics Institut in Toulouse (IMT).
For the first years I have mainly pursued research with colleagues from LANL. I also reconnected with former colleagues and advisors from Bordeaux and CEA researchers or postdocs appointed to CELIA laboratory, namely P.-H. Maire, J. Breil and S. Galera. The connection with the Czech team has been also strengthened with two EGIDE grants called Partenariats-Hubert-Curien (P.H.C) "Barrande" for 2007-2008 and 2010-2011. These grants consist of enhancing already existing collaborations by funding two short term visits of French researchers in Czech Republic and Czech researchers in France.
Concerning the analysis of Lagrangian scheme, B. Wendroff, A.L. Bauer and I have written the article [15] dealing with the proof of a conjectured stability result that was only numerically observed in [12]. Further with M. Shashkov and B. Wendroff we have analysed the problem of volume consistency of the staggered grid Lagrangian hydrodynamics scheme in [16]. The interface reconstruction method *via* power diagrams has been extended by the same team from Los Alamos (as in [13]) to a second-order accurate material-order-independent method in [17]. Recently in a collaboration with J.M. Ghidaglia (CMLA, ENS-Cachan) and J.P. Braeunig (CEA-DIF), we have adapted some techniques used for Lagrangian schemes and classical interface reconstruction to improve their Eulerian scheme in [18]. Moreover dealing with filament and structures smaller than the cell size is difficult with classical interface reconstruction techniques. As a consequence, following an idea from J. Ovadia (retired fellow from CEA-CESTA), C. Fochesato, R. Motte from the CEA-DIF and I have built an interface reconstruction method devoted to filament in [19].
The exchanges between IMT, CELIA and CTU gave also rise to several publications. We have investigated the comparison between staggered and cell-centered Lagrangian and ALE hydrodynamical methods in [20]. Following an idea of P.-H. Maire of recasting some technics used for cell-centered Lagrangian schemes into staggered Lagrangian schemes, P.-H. Maire, P. Váchal and I have developed a set of articles dealing with this general formalism to derive artificial viscosity and its second-order accurate version in 2D in [21, 22, 23]. Recently we also have extended this approach

to 3D with the same co-authors in [24]. Still surfing on the Czech connection I paired up with L.Bednarik, M. Kucharik and R. Liska to study to the concept of slide-line for the 2D compatible staggered Lagrangian scheme in [25].
Back in 2008 the ALE formalism in ALE INC(ubator) and in the CELIA ALE code CHIC did not allow any change of mesh connectivity while rezoning. Consequently with M. Shashkov and the CELIA team (P.-H. Maire, J. Breil and S. Galera) we have extended the ALE formalism to allow topology modifications of the mesh during the computation [26, 27]. This approach is called "ReALE" standing for Reconnection-based ALE.
In Toulouse at IMT with P. Degond we shared a PhD student (L. Carballal-Perdiz) from September 2007 up to November 2010 the subject of the PhD was the development of a multi-scale finite element method dedicated to the prediction of air contaminant transport on multiple scales [28]. The team involved in this research was also constituted of F. Deluzet, A. Lozinski and J.-M. Rovarch through a collaboration with DGA ("Direction Générale de L'Armement").
More recently a collaboration with S. Clain (a former colleague at IMT now appointed associate Professor at the Universidade do Minho, Guimaraẽs in Portugal) and our shared PhD student S. Diot brought by the opportunity to explore the world of very-high order Eulerian finite volume schemes and develop the MOOD method (Multi-dimensional Optimal Order Detection) for unstructured meshes in 2D in [29, 30, 31, 32] and in 3D in article [33]. This method is based on an unlimited high-degree polynomial reconstruction leading to a high-order accurate scheme complemented with an *a posteriori* polynomial order reduction on problematic detected cells. This method has shown very promissing behaviors both on advection and Euler equations on unstructured, non-regular 2D and 3D meshes. The PhD has been defended in August 2012. Meanwhile we won a P.H.C grant (program "Pessoa") for 2012-2013 to exchange researchers and students between the Portuguese institution and IMT which has already led to a common proceedings [31] and fruitful and promissing discussions.
Since 2006 I also have a very fruitful collaboration with researchers from CEA-DIF that led to many studies the topic of which covers staggered Lagrangian schemes and ALE methodology in [34, 35, 36, 37, 38].
A brand new collaboration with G. Dimarco (IMT) at the end of year 2011 led to the development of a fast discrete velocity method for kinetic equations in [39]. This method has been implemented on a mono-processor machine and we have shown that this method is efficient even in full dimensions : 3D in space and 3D in velocity, leading to the effective discretization of six dimensions. A second-order accurate extension of this method is under review in [40].



The time table on this page displays most of my research activities *versus* the time in abscissa : The positions I was appointed to, the short term visits in foreign institutions, the PhD students I mentored and the projects I run as Principal Investigator (PI).
Articles and reports are located according to the year of publication using a color to emphasize the collaboration it refers to, as instance navy blue color is associated to colleagues from Los Alamos, red color refers to articles written while being at CELIA, purple color corresponds to projects conducted with Czech colleagues, brown color to Portuguese colleagues and black to French collaborations. The very top lines provides the list of mini-symposia and conferences I organized or co-organized.

$\sim$

This thesis mainly focuses on works related to the domain of Lagrangian numerical schemes and Arbitrary-Lagrangian-Eulerian methods. Most of them are already published in international journals. Para-phrasing these publications would be of little interest for the readers. Instead I have tried to state the main contribution brought by some publications and to articulate them together in order to clarify the unity behind the scene.
Nevertheless I have borrowed some sentences and rephrased paragraphs from some of my papers which have been written with co-authors. Unavoidably some of the phrases the reader will find in this habilitation have been produced by some of my co-authors to whom I am very grateful. Finaly some descriptions in this thesis are freely inspired by seminal papers and books which are cited at the begining of each associated paragraph.

Undoubtedly my work is led by the constant desire to improve the code ALE INC(ubator) developed at LANL and still maintained at IMT. Also, due to my natural tendency to interact with people, I have had a lot of golden opportunities for collaboration... which I took without any hesitation.

$\sim$

At the time of the publication of this habilitation the number of articles published in international peer review journals is 25, the number of proceedings in international conference with review is 4, the number of unpublished work made for National laboratories (Los Alamos National Laboratory or CEA-DAM) is 12. All are cited at the begining of the bibliography.
Moreover the following articles are discussed in this thesis :

1. **Lagrangian chapter**
   - [12] entitled *The internal consistency, accuracy and stability of the discrete compatible Formulation of Lagrangian Hydrodynamics*
   - [15] entitled *On stabiliy analysis of staggered schemes*
   - [16] entitled *Volume consistency in a staggered grid Lagrangian hydrodynamics scheme*
   - [11] entitled *"Curl-q" : A vorticity damping artificial viscosity for essentially irrotational Lagrangian hydrodynamics calculations*.
   - [10] entitled *The Force/Work Differencing of Exceptional Points in the Discrete, Compatible Formulation of Lagrangian Hydrodynamics*
   - [25] entitled *Enhancement of Lagrangian slide lines as a combined force and velocity boundary condition*.
   - [23] entitled *Staggered Lagrangian discretization based on cell-centered Riemann solver and associated hydro-dynamics scheme*
   - [24] *3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver based artificial viscosity*
2. **ALE chapter**
   - [7] entitled *A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods*
   - [8] entitled *The Repair Paradigm : New Algorithms and Applications to Compressible Flow* and paper [9] entitled *Convergence and Sensitivity Analysis of Repair Algorithms in 1D*

  - [26] entitled *ReALE : a reconnection-based arbitrary-Lagrangian-Eulerian method* and paper [27] entitled *ReALE : a Reconnection Arbitrary-Lagrangian-Eulerian method in cylindrical geometry*
  - [13, 17] entitled *Material order independent interface reconstruction using power diagrams* and *A second-order accurate material-order-independent interface reconstruction technique for multi-material flow simulations*

3. **Miscellaneous chapter**
  - [29, 32, 33] entitled *A high-order finite volume method for hyperbolic systems : Multi-dimensional Optimal Order Detection (MOOD) Improved Detection Criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials The MOOD method in the three-dimensional case : Very-High-Order Finite Volume Method for Hyperbolic Systems*
  - [39] entitled *Towards an ultra efficient kinetic scheme. Part I : basics on the BGK equation*
  - [18] *A totally Eulerian Finite Volume solver for multi-material fluid flows : Enhanced Natural Interface Positioning (ENIP) Dealing with more than two materials in FVCF-ENIP method*

Finally some presentations of the conferences and minisymposia organized by the "MULTIMAT community" can be downloaded under the conferences' links at `http://www.math.univ-toulouse.fr/HYDRO`. This web site maintained by myself in Toulouse is intended to create links for this specific community. This community revolves around developers of ALE type of numerical methods and meets every other year during an international week of conference appropriately called "MULTIMAT conference". Successful conferences held in Paris in 2002, in Oxford in 2005, in Prague in 2007, in Pavia in 2009 and Arcachon 2011 aim at bringing together researchers from universities and research labs to discuss the state of the art for multi-material hydrodynamics simulations. As far as I can tell the custom of these meetings started thanks to a minisymposium organized by Mikhail Shashkov during a SIAM Annual Meeting in San Diego in 2001. Nowadays the community is also trying to meet during 'less crowded' workshops organized by some of the main characters, the last one was held during ECCOMAS conference in Vienna in September 2012.

These meetings are of great importance to maintain some alive, dynamical and friendly competitive atmosphere between the members of the community.

Compatible staggered Lagrangian schemes

1

Contents

While new cell-centered Lagrangian schemes devoted to ALE simulations have been developed in the first 10 years of this centrury, see [42] for an exhaustive historical background presentation and most of all for the detailed description of the schemes developed in [43, 44, 45, 46, 47, 48, 49, 50, 51, 42], the literature on staggered Lagrangian numerical methods dates back to the origin of computers. Nonetheless it is still very much alive these days in a finite difference context [52, 53, 54, 12, 11, 10, 55, 56, 57, 46] or in a finite element context [58, 59, 60, 61, 62].

This chapter more specifically deals with the so-called compatible staggered Lagrangian numerical scheme dedicated to solve hydrodynamics equations on general polygonal/polyhedral grid. This method has been popularized by E. J. Caramana *et al.* in a series of articles in the late 90's [55, 63, 64, 12].

We first present the historical background of this venerable numerical method and then describe the version one considers for solving the compressible hydrodynamics equations. Apart from different notation and alternative ways of presenting the scheme, the results presented in these first two sections have not been obtained by myself. Contrarily in sections 1.3 and 1.4 are presented some of

the contributions obtained by my colleagues and myself concerning the developement, understanding and analysis of this numerical method. However an exhaustive presentation of the scheme is mandatory to genuinely enlight the difficulties and features of this numerical scheme.

1.1 History and presentation

The origins of this compatible staggered Lagrangian numerical scheme are probably to be found in classified document from the Los Alamos National Laboratory, NM, U.S.A, during World War II and the Manhattan project where the "calculation of certain time-dependent fluid flows played an important part in the wartime work of the laboratory" (preface to the first edition of [65]).

Indeed the Lagrangian formulation of the equations of hydrodynamics has a very old and venerable history. The very first numerical calculations that resemble modern computer simulations in the numerical issues considered utilized fluid equations in the Lagrangian frame of reference in 1D [66].

Newton's second law of motion, which is central to any Lagrangian frame of reference relates the force $\boldsymbol{F}_p$ acting on a point of mass M_p and its acceleration $\boldsymbol{A}_p$ computed as the second derivative in time of its position $\boldsymbol{X}_p$: $\boldsymbol{F}_p = m_p \boldsymbol{A}_p$. The discretization by respect to time quite naturaly involves three time levels denoted $n-1, n$ and $n+1$ and a three-level leap-frog scheme with the force centered at time level n. All early Lagrangian schemes in 1D [66] or 2D [65] utilized such a staggered discretization in time. Although this forms a simple and intuitive numerical integration scheme, it leaves the velocity of a mass point defined as the difference between its displacement vector at two different time levels, and therefore the velocity is trully defined only at the $n+1/2$ time levels. When one then considers the total energy of a fluid as a sum of kinetic energy and internal energy that can be exchanged between each other by the action of forces, this sum is difficult to conserve exactly in discrete form owing to the fact that the two components that comprise it are defined at different time levels. Quoting Caramana in [12] *When velocity dependent forces are explicitly added to this model, as with the artificial viscosity [67], this type of time integration becomes somewhat clumsy and looks even contrived [68], since the artificial viscosity terms must be lagged in time to preserve numerical stability.* The spatial discretization of the force in all early versions of Lagrangian hydrodynamics [65, 69] is some form of what is presently known as finite-volume differencing. That is, these various forms calculate the force as a stress (scalar pressure plus deviators) times a normal surface-area vector. The most modern of these older force calculations is the diamond differencing scheme due to Wilkins [69], which uses closed surface area contours to calculate the force acting on a point, and thus properly conserves linear momentum. Other authors arrange the force contributions together in various ways to form the total force acting on each fluid element such that strict conservation of linear momentum may, or may not, be obtained. Most Lagrangian hydrodynamics codes employ a spatially staggered placement of dependent variables with stress, density, and specific internal energy given in cells surrounded by points that have associated position and velocity vectors. This enables the calculation of forces by means of various kinds of finite-volume differencing, with masses and volumes ascribed to both cells and points in an interleaved manner. A difficulty with the older work is that there was no agreement amongst the various authors of these different algorithms as to how these schemes, aside from the noted common features, should be constructed. The choices made were largely arbitrary and not derived from solid mathematical concepts.

An early attempt to remedy this lack of a sound theoretical basis is the work of Goad [70], who used the method of virtual work to derive a form of finite-volume force differencing of the stress in 2D cylindrical geometry. This work was little noted, partly because this type of scheme does not yield the limit of 1D spherical geometry from 2D cylindrical geometry.

Up to our knowledge the work that first places this type of finite-volume algorithm on a firm theoretical basis is due to Favorskii [71], and independently, Margolin and Adams [72]. The first paper shows that the discrete equations in Lagrangian form can be generally derived from a variational principle. It also justifies the use of the surface area vectors of closed volumes as appropriate discretization objects, a practice which was previously employed, but not always correctly, because the surface areas about a point did not in all cases sum to zero. The second paper parallels this work. Its central thrust is to use the continuity equation in discrete form to derive finite-volume differencing given a discrete expression for the volume of a cell. This results in, and also justifies the use of surface areas to calculate the force. It emphasizes that the difference formulas that are derived are "operator" expressions that can be used to calculate discrete derivatives of any function, and not just of the velocity field. It is the discrete form of the continuity equation as emphasized in [72] that is central to the internal consistency of the scheme. A further extension of the work of Favorskii is nowadays known as the "method of support operators" [73]. Although this work is more general than just its application to the equations of Lagrangian hydrodynamics, it is this system of equations that is used in its original exposition. This method also utilizes the continuity equation in discrete form to derive the divergence operator and then uses the vector identities in summation form to derive discrete versions of all other operators. It emphasizes the relation in discrete form of the divergence and gradient operators as negative adjoints of each other as in the continuum case.
These publications all revolve around the central idea that the discrete equations must obey the global properties of the continuum ones in order to be considered as valid discretizations that will then mirror continuum conservation properties in their discrete analogs. As such they remove the arbitrary and heuristic formulations of the previous codes based on the older work [69]. Somewhat after the previously cited developments is the seminal work of Burton [74, 75], which discretizes the fluid equations in Lagrangian form on a staggered spatial grid utilizing subgrid quantities termed subcell masses and subcell forces, from which the cell and point masses, and the total force acting on a point, are constructed. A two-level time integration scheme is also utilized so that both kinetic energy and internal energy are defined at the same time level. The basic reasoning used by Burton to demonstrate conservation of total energy is the same as that employed in the method of support operators [73], and thus incorporates the important features of the previous works [73, 72, 74]. However, Burton's formulation is more general in that he does not consider forces, or differential operators, of any specific origin. Instead, he utilizes an arbitrary subcell force that allows the specification of forces of any forces from functional form. The associated work is completely defined, and, total energy is also exactly conserved. The only restriction on the discrete form of the subcell force is constraint of momentum conservation. He also notes [75] that this formulation of the Lagrangian hydrodynamics equations contains two distinct definitions of cell volume, and considers this difference to be a form of entropy error. It is this latter work of Burton that we refer as the "discrete, compatible formulation of Lagrangian hydrodynamics", and which was initially constructed on arbitrary polyhedral grids [75]. The word "discrete" has been inserted in [12] to emphasize that these equations are essentially created in discrete form, as opposed to being the discretization of a system of PDE's. As such, one may or may not be able to rigorously take the continuum limit to obtain the latter ; this depends on the kinds of forces that are employed, as instance artificial viscosity and anti-hourglass forces.
Finally, the discrete, compatible formulation of Lagrangian hydrodynamics was developed to be an algebraic identity : this identity consists of two arbitrary scalars, the cell and point masses, and one arbitrary vector, the subcell force, such that given the usual definition of total energy conservation is always fulfilled. As such, it describes a priori truth that cannot be confuted, since in primitive form

it makes no assertion about any physical system. The quality with which the discrete, compatible formulation of Lagrangian hydrodynamics may describe certain physical situations is mostly, if not entirely, dependent on the quality of the specification of the three abstract quantities that compose it.

In next section we state the governing equation, notation and derive the compatible staggered Lagrangian scheme and some of its associated properties.

1.2 Presentation of the compatbile staggered Lagrangian scheme

1.2.1 Governing equations and notation

In this chapter we mainly focus on two dimension space $\mathbb{R}^2$ paved with polygonal cells. The model equations under consideration are the hydrodynamics equations for which we neglect viscous stress and heat conduction. In other words we mainly focus on the gas dynamics equations expressed as conservation laws of mass, momentum and total energy.

In Lagrangian framework, the two-dimensional gas dynamics equations write

$$\rho \frac{d}{dt}\left(\frac{1}{\rho}\right) - \nabla \cdot \boldsymbol{U} = 0, \tag{1.1}$$

$$\rho \frac{d}{dt}\boldsymbol{U} + \boldsymbol{\nabla} P = \boldsymbol{0}, \tag{1.2}$$

$$\rho \frac{d}{dt} E + \nabla \cdot (P\boldsymbol{U}) = 0, \tag{1.3}$$

where ρ is the density, $\boldsymbol{U}$ the velocity, E the specific total energy and $\frac{d}{dt}$ denotes the material derivative. The first equation expresses the volume conservation equation, whereas the second and third ones are the momentum and total energy conservation equations. Volume conservation equation is often referred to as the Geometric Conservation Law (GCL). The previous system is equipped with a thermodynamics closure (equation of state EOS) $P = P(\rho, \varepsilon)$, where the specific internal energy is given by $\varepsilon = E - \frac{\boldsymbol{U}^2}{2}$. Note that for smooth solutions energy equation can be rewritten as

$$\rho \frac{d}{dt}\varepsilon + P\nabla \cdot \boldsymbol{U} = 0, \tag{1.4}$$

and, substituting volume equation yields

$$\rho \frac{d}{dt}\varepsilon + P\rho \frac{d}{dt}\left(\frac{1}{\rho}\right) = 0. \tag{1.5}$$

Recalling Gibbs relation for temperature T and specific entropy S : $TdS = d\varepsilon + Pd\left(\frac{1}{\rho}\right)$, and the second law of thermodynamics, namely $T\frac{dS}{dt} \geq 0$, implies that for non-smooth flows the following relation holds :

$$\rho \frac{d}{dt}\varepsilon + P\nabla \cdot \boldsymbol{U} \geq 0. \tag{1.6}$$

As a consequence, internal energy equation can be viewed as an entropy evolution equation since

$$\rho \frac{d}{dt}\varepsilon + P\rho \frac{d}{dt}\left(\frac{1}{\rho}\right) \geq 0. \tag{1.7}$$

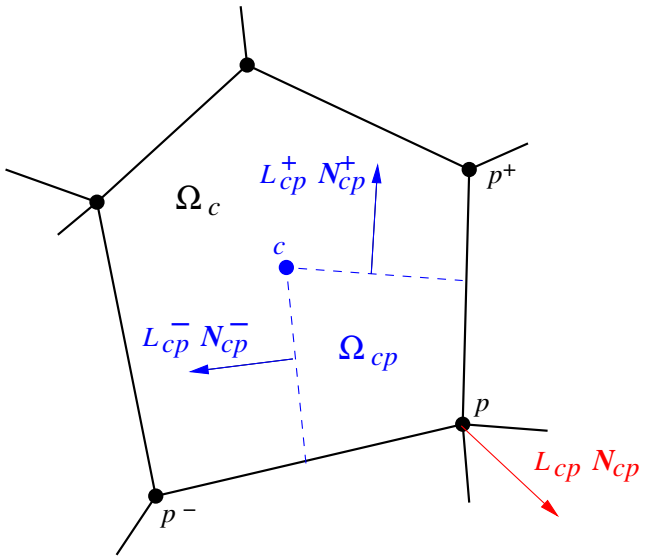


FIGURE 1.1 – *Fragment of a polygonal grid. Position and velocity are defined at grid points while thermodynamic variables are located at cell centers. A polygonal cell, Ω_c, is subdivided into subcells Ω_{cp}. Points are denoted by subscript p and counterclockwise ordered p^-, p, p^+.*

The previous system (1.1-1.3) can therefore be rewritten as a non-conservative system by replacing the energy equation by (1.6). The last equations are the trajectory equations

$$\frac{d\boldsymbol{X}}{dt} = \boldsymbol{U}(\boldsymbol{X}(t), t), \qquad \boldsymbol{X}(0) = \boldsymbol{x}, \tag{1.8}$$

expressing the Lagrangian motion of any point initially located at position $\boldsymbol{x}$.

We use a staggered placement of variables in which position and velocity are defined at grid points while thermodynamic variables are located at cell centers, refer to Fig. 1.1. An unstructured grid consisting of a collection of non-overlapping polygons is considered. Each polygonal cell is assigned a unique index c and is denoted Ω_c. Each vertex/point of the mesh is assigned a unique index p and we denote $\mathcal{C}(p)$ the set of cells sharing a particular vertex p. Each polygonal cell is subdivided into a set of subcells ; each being uniquely defined by a pair of indices c and p and denoted Ω_{cp}. This subcell is constructed by connecting the cell center of Ω_c to the mid-points of cell edges impinging at point p. The union of subcells Ω_{cp} that share a particular vertex p allows to define the dual vertex-centered cell Ω_p related to point p with $\Omega_p = \bigcup_{c\in\mathcal{C}(p)} \Omega_{cp}$. Using the previous notation, we can define the primary grid $\bigcup_c \Omega_c$ and the dual grid $\bigcup_p \Omega_p$. The volumes of the primary and dual cells are functions of time t. For a vertex p of cell Ω_c we denote its previous and next vertices by p^- and p^+. Here, following [55], we make the fundamental but questionable assumption that the subcells are Lagrangian volumes. This means that the subcell mass m_{cp} is constant in time. Therefore, being given the initial density field $\rho^0(\boldsymbol{x})$ one deduces the initial mean density in cell c

$$\rho_c^0 = \int_{\Omega_c(0)} \rho^0(\boldsymbol{x}) d\boldsymbol{x} / V_c^0, \tag{1.9}$$

where V_c^0 is the volume of cell Ω_c at time $t = 0$. Subcell mass is defined as $m_{cp} = \rho_c^0\ V_{cp}^0$, where V_{cp}^0 is the initial volume of subcell Ω_{cp}. By summation of Lagrangian subcell masses one defines

Lagrangian cell/point masses as

$$m_c = \sum_{p \in \mathcal{P}(c)} m_{cp}, \qquad m_p = \sum_{c \in \mathcal{C}(p)} m_{cp}, \tag{1.10}$$

where $\mathcal{P}(c)$ is the set of counterclockwise ordered vertices of cell c.

1.2.2 Compatible discretization

We construct staggered Lagrangian schemes using the well known methodology of compatible discretization which has been presented in [74, 55, 12]. The cornerstone of this type of discretization is the subcell force that acts from subcell cp onto point p, see Fig. 1.1. In this approach, the discretization of the internal energy equation in terms of subcell forces is deduced from total energy conservation. Here, we reproduce the derivation of Maire [42] starting from a generic abstract form of the subcell force so that an entropy inequality is satisfied, which ensures that kinetic energy is dissipated into internal energy through shock waves. The subcell force writes as a pressure contribution plus a viscous contribution also known as artificial viscosity or pseudo-viscosity.

Geometric Conservation Law (GCL). Here, we use a discretization of the volume equation (1.1) that is compatible with the GCL. By GCL compatibility we mean that we are deriving a discrete divergence operator for the volume equation by requiring consistency of the divergence of the velocity field with the time rate of change of volume of the cell, refer to [76]. By noticing that $m_c = \rho_c V_c$, where $\rho_c = \rho_c(t)$ and $V_c = V_c(t)$ are the cell density and volume, we can write

$$m_c \frac{d}{dt}\left(\frac{1}{\rho_c}\right) = \frac{d}{dt} V_c,$$

using the fact that the cell mass is constant in time. Moreover, remarking that the cell volume can be expressed as a function of the position vectors of its vertices as follows

$$V_c(t) = \sum_{p \in \mathcal{P}(c)} \frac{1}{2} \left(\boldsymbol{X}_p \times \boldsymbol{X}_{p^+} \right) \cdot \boldsymbol{e}_z,$$

where $\boldsymbol{e}_z$ is the unit vector of the canonical basis in z direction. We deduce that the time rate of change of the cell volume writes

$$\frac{d}{dt} V_c = \sum_{p \in \mathcal{P}(c)} \boldsymbol{\nabla}_{\boldsymbol{X}_p} V_c \cdot \frac{d}{dt} \boldsymbol{X}_p.$$

Here, we have simply applied the chain rule differentiation. Setting $\frac{d}{dt}\boldsymbol{X}_p = \boldsymbol{U}_p$ where $\boldsymbol{U}_p$ is the vertex velocity, we rewrite this last equation as

$$\boxed{\frac{d}{dt} V_c - \sum_{p \in \mathcal{P}(c)} L_{cp} \boldsymbol{N}_{cp} \cdot \boldsymbol{U}_p = 0,} \tag{1.11}$$

where $L_{cp}\boldsymbol{N}_{cp}$, with $\boldsymbol{N}_{cp}^2 = 1$, stands for the corner vector defined by

$$L_{cp}\boldsymbol{N}_{cp} = \boldsymbol{\nabla}_{\boldsymbol{X}_p} V_c. \tag{1.12}$$

This corner vector is a fundamental geometric object which is nothing but the gradient of the cell volume at point p. Its explicit expression in terms of points coordinates writes

$$L_{cp}\boldsymbol{N}_{cp} = \frac{1}{2}\begin{pmatrix} Y_{p^+} - Y_{p^-} \\ -(X_{p^+} - X_{p^-}) \end{pmatrix},$$

where (X_p, Y_p) denote the coordinate of the position vector $\boldsymbol{X}_p$. This kind of formalism is well known and has been used in staggered and cell-centered (free Lagrange) discretizations long time ago [76, 77]. We note that (1.11) is compatible with the discrete version of the trajectory equation (1.8)

$$\frac{d}{dt}\boldsymbol{X}_p = \boldsymbol{U}_p, \quad \boldsymbol{X}_p(0) = \boldsymbol{x}_p.$$

This leads to a compatible definition of the discrete divergence operator over cell c as

$$(\nabla \cdot \boldsymbol{U})_c = \frac{1}{V_c}\sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp}\cdot\boldsymbol{U}_p. \tag{1.13}$$

We also emphasize that the corner vector $L_{cp}\boldsymbol{N}_{cp}$ satisfies the fundamental geometric identity

$$\sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp} = \boldsymbol{0}, \tag{1.14}$$

which is equivalent to the result that the summation of the outward normals to a closed polygonal contour is equal to zero.

Finally, we have obtained a compatible discretization of the volume equation (1.1) which writes

$$m_c\frac{d}{dt}\left(\frac{1}{\rho_c}\right) - \sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp}\cdot\boldsymbol{U}_p = 0. \tag{1.15}$$

Momentum equation. The semi-discrete momentum equation over the dual cell Ω_p writes

$$\boxed{m_p\frac{d}{dt}\boldsymbol{U}_p + \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp} = \boldsymbol{0}.} \tag{1.16}$$

Here, $\boldsymbol{F}_{cp}$ is the subcell force from cell c that acts on point p, which is defined by

$$\boldsymbol{F}_{cp} = \int_{\partial\Omega_p(t)\cap\Omega_c(t)} P\boldsymbol{N}\mathrm{d}l, \tag{1.17}$$

where $\mathrm{d}l$ is an infinitesimal length. Momentum equation (1.16) is nothing but the Newton law applied to particle of mass m_p moving with velocity $\boldsymbol{U}_p$.

Specific internal energy equation. Here we derive a semi-discrete internal energy equation that ensures total energy conservation using the concept of subcell force, following the approach initially described in [55]. Let us introduce total kinetic energy and total internal energy

$$\mathcal{K}(t) = \sum_p \frac{1}{2}m_p\boldsymbol{U}_p^2(t), \tag{1.18}$$

$$\mathcal{E}(t) = \sum_c m_c\varepsilon_c(t), \tag{1.19}$$

where ε_c is the cell averaged specific internal energy. Total energy is then defined as

$$E(t) = \mathcal{K}(t) + \mathcal{E}(t). \tag{1.20}$$

The conservation of total energy without taking into account boundary conditions simply writes

$$\frac{d}{dt}E = \frac{d}{dt}\mathcal{K} + \frac{d}{dt}\mathcal{E} = 0. \tag{1.21}$$

The substitution of kinetic and internal energies recalling that cell/point masses are Lagrangian objects, i.e. they not depend on time, yields

$$\frac{d}{dt}\mathcal{K} + \frac{d}{dt}\mathcal{E} = \sum_c m_c \frac{d}{dt}\varepsilon_c + \sum_p m_p \frac{d}{dt}\boldsymbol{U}_p \cdot \boldsymbol{U}_p,$$

then using (1.21) one deduces

$$\sum_c m_c \frac{d}{dt}\varepsilon_c + \sum_p m_p \frac{d}{dt}\boldsymbol{U}_p \cdot \boldsymbol{U}_p = 0.$$

Using the semi-discrete momentum equation (1.16) yields

$$\sum_c m_c \frac{d}{dt}\varepsilon_c - \sum_p \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp} \cdot \boldsymbol{U}_p = 0,$$

and interchanging the order in the double sum one finally gets

$$\sum_c \left(m_c \frac{d}{dt}\varepsilon_c - \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp} \cdot \boldsymbol{U}_p \right) = 0. \tag{1.22}$$

A sufficient condition for total energy conservation is obtained by requiring the previous equation to hold in each cell c

$$\boxed{m_c \frac{d}{dt}\varepsilon_c - \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp} \cdot \boldsymbol{U}_p = 0.} \tag{1.23}$$

Notice that this choice is not unique and other discretizations would provide the total energy conservation given a definition of total enery such as (1.20). Any of such discretization is refered to as a "compatible discretization" under Caramana's appelation. The word "discrete" also used in Caramana's phrasing refers to the fact that the discrete equations are rather deduced than derived from the constinuous equations.
Once the subcell force is known, then momentum and internal energy can be updated using equations (1.16) and (1.23).

Summary of the compatible discretization. We summarize the semi-discrete equations that govern the time rate of change of the primary variables $(\frac{1}{\rho}_c, \boldsymbol{U}_p, \varepsilon_c)$

$$m_c \frac{d}{dt}(\frac{1}{\rho}_c) - \sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp} \cdot \boldsymbol{U}_p = 0, \tag{1.24}$$

$$m_p \frac{d}{dt}\boldsymbol{U}_p + \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp} = \boldsymbol{0}, \tag{1.25}$$

$$m_c \frac{d}{dt}\varepsilon_c - \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp} \cdot \boldsymbol{U}_p = 0. \tag{1.26}$$

We point out that the mesh motion is given by the trajectory equations

$$\frac{d}{dt}\boldsymbol{X}_p = \boldsymbol{U}_p(\boldsymbol{X}_p(t), t), \qquad \boldsymbol{X}_p(0) \quad = \boldsymbol{x}_p, \tag{1.27}$$

which is compatible with the GCL. The thermodynamic closure is given by the equation of state which writes $P_c = P(\rho_c, \varepsilon_c)$. We emphasize that this subcell-based compatible discretization ensures total energy conservation regardless of the subcell force form.
Although our description of the staggered compatible Lagrangian scheme is different from the descriptions of Burton [74, 78] or Caramana [55, 12] it shares with them the same fundamental objects : Cell/point masses and subcell force such that the compatible discretization intrinsically leads to the conservation of total energy by construction. What must be the components of a sucell force is almost left to the developer (or user). Under this subcell force concept many different physical or numerical effects are in fact gathered. First the pressure force takes into account the $\boldsymbol{\nabla} P$ term in (1.2). Then the artificial viscosity force is designed to handle shock wave and steep fronts and as such stabilizes the scheme. It also assures (1.6) to hold. The anti-hourglass force is a pure numerical concept which is meant to fight back parasitical grid motion known as "hourglass modes" [56]. Elasto-plasticity terms can be expressed into this force formalism [79], slide-line [80, 25] or internal boundary conditions also. In fact many physical models can be recast into this fruitful compatible discretization [1], and, no matter what is put under this definition, conservation is preserved.

1.2.3 Subcell forces

Let us provide in this section a definition of the subcell force invoking Galilean invariance and thermodynamic consistency. Subcell pressure force is then deduced and several artificial viscous forces and anti-hourglass subpressure force are further described.

Galilean invariance is a principle of relativity which states that the fundamental laws of physics are the same in all inertial frames. It is one of the key requirements of many physical models adopted in theoretical and computational mechanics. To fulfill Galilean invariance, the previously derived specific internal energy equation (1.23) must remain unchanged under a uniform translation of frame. Let $\boldsymbol{A}$ denote the uniform translation velocity. Then equation (1.23) transforms into

$$m_c \frac{d}{dt}\varepsilon_c - \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp} \cdot (\boldsymbol{U}_p + \boldsymbol{A}) = 0.$$

By substituting (1.23) into this last equation leads to

$$\sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp} \cdot \boldsymbol{A} = 0,$$

which must hold for all vectors $\boldsymbol{A}$. Therefore, specific internal energy equation remains invariant under uniform translation if and only if

$$\boxed{\sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp} = \boldsymbol{0}.} \tag{1.28}$$

1. This is probably one reason why this discretization has been successful amongst physicists from national laboratories along with the fact that using artificial viscosity methods are inherently simpler than operator splitting methods (such as Godunov methods) in that the level of numerical complexity does not increase as the number of dimensions and/or the amount of physics included increases.

We note that this result has been already quoted in [12] page 576 and probably elsewhere before. This condition also implies total momentum conservation without taking into account boundary conditions. To demonstrate this, it suffices to time-differentiate the global momentum defined as

$$\boldsymbol{\mathcal{Q}} = \sum_p m_p \boldsymbol{U}_p, \tag{1.29}$$

to obtain

$$\begin{aligned} \frac{d}{dt}\boldsymbol{\mathcal{Q}} &= \sum_p m_p \frac{d}{dt}\boldsymbol{U}_p \\ &= -\sum_p \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp}, \quad \text{thanks to momentum equation,} \\ &= -\sum_c \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}, \quad \text{by interchanging the double sums.} \end{aligned} \tag{1.30}$$

Thus, $\frac{d}{dt}\boldsymbol{\mathcal{Q}} = \boldsymbol{0}$ due to condition (1.28), which completes the proof.

A corollary of the Galilean invariance condition is that specific internal energy equation (1.23) can also be rewritten into

$$m_c \frac{d}{dt}\varepsilon_c - \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}\cdot(\boldsymbol{U}_p - \boldsymbol{U}_c) = 0, \tag{1.31}$$

where $\boldsymbol{U}_c$ is any arbitrary piecewise constant cell based velocity. This equation will be used in the next section.

Subcell pressure force

Let us investigate the thermodynamic consistency of the semi-discrete scheme by computing the time rate of change of entropy in a cell c. Using Gibbs formula, one gets

$$m_c T_c \frac{d}{dt} S_c = m_c \left[\frac{d}{dt}\varepsilon_c + P_c \frac{d}{dt}\left(\frac{1}{\rho_c}\right)\right], \tag{1.32}$$

where S_c and T_c are the specific entropy and temperature of cell c. Substituting into (1.32) the specific internal energy equation (1.23) and the volume equation (1.15) yields

$$m_c T_c \frac{d}{dt} S_c = \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}\cdot\boldsymbol{U}_p + P_c \left(\sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp}\cdot\boldsymbol{U}_p \right) \tag{1.33}$$

$$= \sum_{p\in\mathcal{P}(c)} (\boldsymbol{F}_{cp} + L_{cp}P_c\boldsymbol{N}_{cp})\cdot\boldsymbol{U}_p. \tag{1.34}$$

For smooth flow the right hand side of the last equation must be zero leading to the form of the *subcell pressure force* as

$$\boxed{\boldsymbol{F}_{cp}^{\text{press}} = -L_{cp}\ P_c\ \boldsymbol{N}_{cp},} \tag{1.35}$$

which corresponds to the discretization of (1.17). One trivially verifies that

$$\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{\text{press}} = -P_c \sum_{p\in\mathcal{P}(c)} L_{cp}\boldsymbol{N}_{cp} = \boldsymbol{0}, \tag{1.36}$$

thanks to identity (1.14), which, as a side effect implies that momentum conservation is ensured.

Artificial viscous force

While cell-centered Lagrangian schemes rely on some sort of Riemann solvers to add numerical viscosity [43, 50, 51], staggered Lagrangian schemes historicaly rely on artificial viscosity [67]. The artificial viscosity, otherwize called pseudo-viscosity, is historically refered to as the 'q' term. The illuminating idea of von Neumann and Richtmyer was to introduce a purely artificial dissipative mechanism of such a form and strength that the shock transition would be a smooth one extending over a small number of cell length, and then to include this dissipation into the finite difference equations, [67] page 312 CHAP. 12 SEC. 12.10. Only a linear term introduced by Landshoff in [81] was present in the form of the original artificial viscosity. As a consequence the thickness of the 'transition layer' (i.e. the shock spreading) was varying with the shock strength approaching zero for a very strong shock and tending to infinity for a very weak one. However their wish was to have a constant thickness of the shock spreading, so von Neumann and Richtmyer added a quadratic term which they interpret to be as "using a small viscosity coefficient for weak shocks" ([67] page 312). In an unpublished work from Los Alamos from the 50's Rosenbluth suggested that the artificial viscosity should be zero when the fluid is undergoing an expansion, this 'trick' is nowadays known as the 'artificial viscosity switch'.
Since the seminal work of von Neumann and Richtmyer there is still no universally satisfactory form of the artificial viscosity suitable for all problems although many authors contributed to the subject. As instance Schultz [82, 83] introduced the nowadays known "edge-based" artificial viscosity further popularized by Caramana in [57], Richards [84], Wilkins analysed the viscosity coefficients [85] in the U.S.A, this work was also previously achieved by Kurapatenko in Soviet Union [86], Christensen interpreted the artificial viscosity as a Riemann solver [87] related to the works of Dukowicz [88, 89], Noh studied the errors that arise when using artificial viscosity in [90], Benson revised most of these works in the review paper [52] and also contributed to flux-limited shock viscosity in [91], Shashkov proposed a tensor extension of the artificial viscosity using mimetic finite difference method in [53, 92] even if several attempts of 'tensorization' have been tried before [93, 83] and, more recently, by Rieben and Kolev [62] and Owen [94] contributed to the subject in slighly different contexts [2]
The fact that different forms for this force are often utilized depending on the type of problem being studied is the major remaining deficiency of this class of hydrodynamics methods. In the following we briefly present three models to compute *artificial viscous subcell forces*.

Bulk viscosity based on original works [67, 81] considers a cell centered "pseudo-pressure"

$$q_c = c_1 \rho_c a_c^* |\Delta \boldsymbol{U}| + c_2 \rho_c (\Delta \boldsymbol{U})^2, \tag{1.37}$$

where $c_1 \leq c_2$ are two constants of the order of unity and $\Delta \boldsymbol{U}$ is a measure of the velocity difference over the cell and a_c^* and ρ_c are respectively the sound-speed and density in cell c. However the Kurapatenko combinaison [86] of linear and non-linear terms for material with ratio of specific heats γ is often used instead of (1.37)

$$q_c^{\text{Kur}} = \rho_c \left(c_2 \frac{\gamma - 1}{4} |\Delta \boldsymbol{U}| + \sqrt{c_2^2 \left(\frac{\gamma - 1}{4} \right)^2 (\Delta \boldsymbol{U})^2 + c_1^2 (a_c^*)^2} \right) |\Delta \boldsymbol{U}|. \tag{1.38}$$

This expression has been derived for an ideal equation of state to determine the form of the term that, quoting Caramana in [57] "must be added to the pressure in front of a steady-state

2. This list of published works is not entended to be exhautive, rather the works have been chosen to spread along the years from the 50's up to now and the authors have been cited as to give to the reader key names in the field. Following the cited works from these researchers surely provides an almost exhaustive view of the artificial viscosity quest, the rest being unpublished and classified works kept in library of national laboratories.

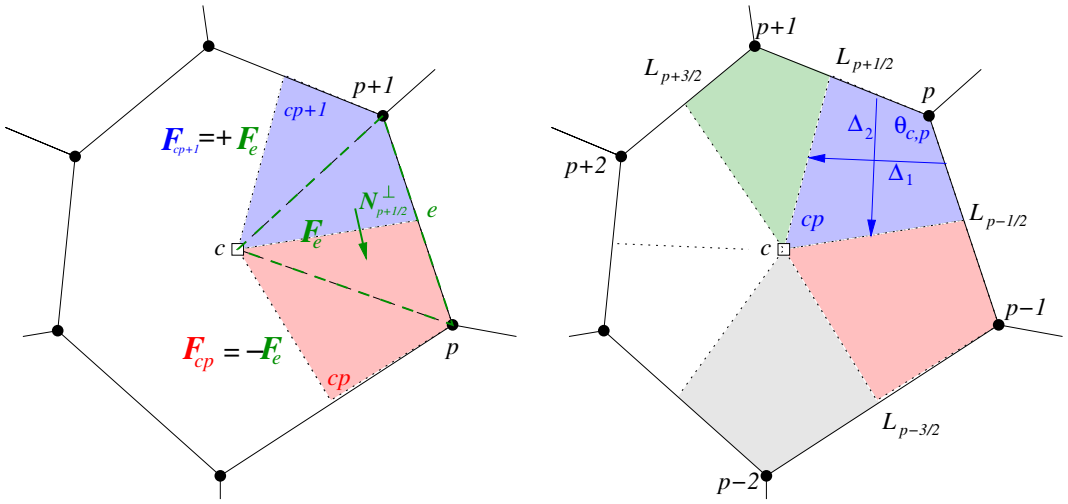


Figure 1.2 – *Notation involved for the artificial viscosity models — Left : the edge viscous force is first computed related to the triangular zone associated to an edge e (green triangle). Then this edge-based viscous force is associated to subcell cp with a + sign and cp′ with a minus sign — Right : three tensor viscous force*

shock in order to achieve the pressure behind the shock, using the jump conditions. In this instance ρ_c and a_c^* are the density and sound speed ahead of the shock, and $\Delta \boldsymbol{U}$ is the velocity jump across it". The use of (1.37) or (1.38) consists of adding the 'q' force to the pressure force (1.35) to get the following viscous subcell force

$$\boxed{\boldsymbol{F}_{cp}^{\mathrm{q}} = \begin{cases} -L_{cp}\, q_c^{\mathrm{Kur}}\, \boldsymbol{N}_{cp} & \text{if} \quad (\nabla \cdot \boldsymbol{U})_c \leq 0 \\ 0 & \text{else} \end{cases}} \tag{1.39}$$

Note that either q or q^{Kur} are positive constants over the cell. This form of artificial viscosity is dissipative because recalling (1.34) with the artificial viscous force (1.39) we have that for a non-smooth flow the right hand side of this equation must be positive (that is to say the scheme must be dissipative), and we convince ourselves that

$$\begin{aligned} m_c T_c \frac{d}{dt} S_c &= \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp}^{\mathrm{q}} \cdot \boldsymbol{U}_p \\ &= \sum_{p \in \mathcal{P}(c)} -L_{cp}\, q_c^{\mathrm{Kur}}\, \boldsymbol{N}_{cp} \cdot \boldsymbol{U}_p \\ &= -q_c^{\mathrm{Kur}} \sum_{p \in \mathcal{P}(c)} L_{cp} \boldsymbol{N}_{cp} \cdot \boldsymbol{U}_p \\ &= -q_c^{\mathrm{Kur}}\, V_c (\nabla \cdot \boldsymbol{U})_c \geq 0, \end{aligned}$$

due to the fact that $(\nabla \cdot \boldsymbol{U})_c \leq 0$.
This formulation has been widely used but its main drawbacks lay in its inability to vanish for rigid rotation or uniform compression or (sometimes depending on the implementation) along a front of constant phase, see discussion in [57] page 85 for more details.

Edge viscosity based on [82, 83] and popularized in [57] is based on the computation of a 'q' term for each edge of a cell. A clear description of the edge viscosity forces is provided in Appendix A of [57]. Here we only para-phrase this Appendix. Let us consider one edge e defined by two

succesive points $p, p+1$ of cell c, see Fig.1.2, such that the velocity difference over this edge is $\Delta \boldsymbol{U}_e = \boldsymbol{U}_{p+1} - \boldsymbol{U}_p$ and the associated unit vector is $\widehat{\Delta \boldsymbol{U}}_e$. Let us first define the edge-based viscous force as

$$\boxed{\boldsymbol{F}_e = \begin{cases} (1-\psi_e) q_e^{\text{Kur}} (\Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp}) \widehat{\Delta \boldsymbol{U}}_e & \text{if} \quad \Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp} \leq 0, \\ 0 & \text{if} \quad \Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp} > 0 \end{cases}} \tag{1.40}$$

where

$$q_e^{\text{Kur}} = \rho_e \left(c_2 \frac{\gamma - 1}{4} |\Delta \boldsymbol{U}_e| + \sqrt{c_2^2 \left(\frac{\gamma - 1}{4} \right)^2 (\Delta \boldsymbol{U}_e)^2 + c_1^2 (a_e^*)^2} \right), \tag{1.41}$$

and $\boldsymbol{N}_{p+1/2} = \frac{\boldsymbol{X}_{p+1} - \boldsymbol{X}_p}{|\boldsymbol{X}_{p+1} - \boldsymbol{X}_p|}$ is the unit normal along the edge direction and $\boldsymbol{N}_{p+1/2}^{\perp}$ is the perpendicular unit vector to $\boldsymbol{N}_{p+1/2}$. ρ_e and a_e^* are edge-based density and sound-speed respecively which can be computed as

$$\rho_e = \frac{2 \rho_p \rho_{p+1}}{\rho_p + \rho_{p+1}}, \qquad a_e^* = \min(a_p^*, a_{p+1}^*). \tag{1.42}$$

Moreover $0 \leq \psi_e \leq 1$ is the edge limiter dedicated to make the artificial viscosity to vanish for uniform compression, rigid rotation, and along a front of constant phase. We refer the reader to [57] for the exact definition and calculation of ψ_e. Finally the sign of $(\Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp})$ represents the "switch" to turn off the artificial viscosity for expansion seen from the edge e. For a zone under compression, $(\nabla \cdot \boldsymbol{U})_c \leq 0$, for the triangular subzonal edge of cell c to be under compression we postulate the condition $\Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp} \leq 0$. To build a subcell viscous force it remains to distribute $\boldsymbol{F}_e$ between the two subcells cp and $cp+1$, this is brought about by setting $\boldsymbol{F}_{cp+1}^q = -\boldsymbol{F}_e$ and $\boldsymbol{F}_{cp}^q = +\boldsymbol{F}_e$ noticing that a contribution with a minus sign from left neighbor edge is also associated to subcell force $\boldsymbol{F}_{cp}^q$, see Fig.1.2-left.
Dissipativity in this semi-discrete form is ensured because

$$\begin{aligned}
m_c T_c \frac{d}{dt} S_c &= \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp}^{\text{q}} \cdot \boldsymbol{U}_p \\
&= \sum_{p \in \mathcal{P}(c)} (\boldsymbol{F}_e^{\text{right}} - \boldsymbol{F}_e^{\text{left}}) \cdot \boldsymbol{U}_p \qquad \longleftarrow \text{two edges imping. on } p \\
&= \sum_{e \in \mathcal{E}(c)} \boldsymbol{F}_e \cdot (\boldsymbol{U}_p - \boldsymbol{U}_{p+1}) \qquad \longleftarrow \text{switch to sum over edges} \\
&= \sum_{e \in \mathcal{E}(c)} (1-\psi_e) q_e^{\text{Kur}} (\Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp}) \widehat{\Delta \boldsymbol{U}}_e \cdot (-\Delta \boldsymbol{U}_e) \\
&= -\sum_{e \in \mathcal{E}(c)} \underbrace{\frac{(1-\psi_e) q_e^{\text{Kur}}}{|\Delta \boldsymbol{U}_e|}}_{\geq 0} \underbrace{(\Delta \boldsymbol{U}_e \cdot \boldsymbol{N}_{p+1/2}^{\perp})}_{\leq 0} \quad \longleftarrow \text{because } \widehat{\Delta \boldsymbol{U}}_e = \frac{\Delta \boldsymbol{U}_e}{|\Delta \boldsymbol{U}_e|} \\
&\geq 0.
\end{aligned}$$

The main drawback of the artificial viscosity model is the occurence of "spurious jets" along axes as instance in the Noh problem on Cartesian grid, see Fig.1.8 left panel in section 1.4.1.

Tensor viscosity based on [53] where the entire theory is described. Instead of reproducing the derivation of such tensorial subcell based artificial viscosity force we give its final form

$$\boxed{\boldsymbol{F}_{cp}^{q} = V_c \left[\frac{1}{L_{p+1/2}} \left\{ \boldsymbol{R}_{p+1} + \boldsymbol{R}_p \right\} - \frac{1}{L_{p-1/2}} \left\{ \boldsymbol{Q}_p + \boldsymbol{Q}_{p-1} \right\} \right],} \tag{1.43}$$

where

$$\begin{aligned}
\boldsymbol{R}_{p+1} &= \frac{W_{c,p+1}}{\sin^2 \theta_{c,p+1}} \left(\mu_{c,p+1} \boldsymbol{G}_{p+1/2} + \cos\theta_{c,p+1}\ \mu_{c,p+1} \boldsymbol{G}_{p+3/2} \right), \\
\boldsymbol{R}_{p} &= \frac{W_{c,p}}{\sin^2 \theta_{c,p}} \left(\mu_{c,p} \boldsymbol{G}_{p+1/2} + \cos\theta_{c,p}\ \mu_{c,p} \boldsymbol{G}_{p-1/2} \right), \\
\boldsymbol{Q}_{p} &= \frac{W_{c,p}}{\sin^2 \theta_{c,p}} \left(\mu_{c,p} \boldsymbol{G}_{p-1/2} + \cos\theta_{c,p}\ \mu_{c,p} \boldsymbol{G}_{p+1/2} \right), \\
\boldsymbol{Q}_{p-1} &= \frac{W_{c,p-1}}{\sin^2 \theta_{c,p-1}} \left(\mu_{c,p-1} \boldsymbol{G}_{p-1/2} + \cos\theta_{c,p-1}\ \mu_{c,p-1} \boldsymbol{G}_{p-3/2} \right).
\end{aligned}$$

Here the notation is relative to point p which is a vertex of cell c, see Fig. 1.2-right, the previous points are indexed $p-2, p-1$, the next ones $p+1, p+2$. The edge connecting p and $p+1$ is indexed $p+1/2$ and the unit vector along this edge is refered to as $\boldsymbol{T}_{p+1/2}$ and its length is $L_{p+1/2}$. $\theta_{c,p}$ is the angle between the two edges of cell c meeting at point p. Moreover for all edge we define

$$\boldsymbol{G}_{p+1/2} = \frac{\boldsymbol{U}_{p+1} - \boldsymbol{U}_p}{L_{p+1/2}}, \tag{1.44}$$

and W's are some weights satisfying $W_{c,p} \geq 0$ and $\sum_{p \in \mathcal{P}(c)} W_{c,p} = 1$. Usually, for a quadrilateral cell, $W_{c,p}$ is defined as one half the area of the triangle in cell c which contains the angle at point p divided by the cell volume. For non-quadrilateral cells normalization is needed. It remains to define the μs which are some viscosity coefficients, a kind of Kurapatenko q term,

$$\mu_{cp} = (1 - \psi_{cp}) \rho_{cp} \left(c_2 \frac{\gamma+1}{4} |\Delta \boldsymbol{U}_{cp}| + \sqrt{c_2^2 \left(\frac{\gamma+1}{4} \right)^2 (\Delta \boldsymbol{U}_{cp})^2 + c_1^2 (a_c^*)^2} \right) l_{cp}. \tag{1.45}$$

This expression requires the definition of a velocity jump $\Delta \boldsymbol{U}_{cp}$ and a characteristic length l_{cp} in subcell cp. The definition of these values is a major source of difficulties for multi-dimensional artificial viscosity. Naive definitions result in instabilities for large aspect ratio cells. In [53] the authors have found a length definition that does not cause any problem for large aspect ratios, nor when a small change in velocity or geometry can result in large change in the length of velocity terms,

$$l_{cp} = \begin{cases} 2\sqrt{V_{cp}} \sqrt{\frac{|\Delta_1|}{|\Delta_2|}} & \text{if} \quad \widehat{\Delta_1} \cdot \widehat{\boldsymbol{U}_{av}} > \widehat{\Delta_2} \cdot \widehat{\boldsymbol{U}_{av}} \\ 2\sqrt{V_{cp}} \sqrt{\frac{|\Delta_2|}{|\Delta_1|}} & \text{if} \quad \widehat{\Delta_1} \cdot \widehat{\boldsymbol{U}_{av}} \leq \widehat{\Delta_2} \cdot \widehat{\boldsymbol{U}_{av}} \end{cases} \tag{1.46}$$

where Δ_1, Δ_2 are the lengths across the subcell and the hat symbol refers to the associated unit vectors, see Fig. 1.2-right. Moreover we define $\boldsymbol{U}_{av} = \frac{1}{4}(\boldsymbol{U}_{p-1/2} + \boldsymbol{U}_p + \boldsymbol{U}_{p+1/2} + \boldsymbol{U}_c)$ where

$\boldsymbol{U}_{p+1/2} = \frac{1}{2}(\boldsymbol{U}_p + \boldsymbol{U}_{p+1})$ and the cell-centered velocity is given by $\boldsymbol{U}_c = \frac{1}{|\mathcal{P}(c)|} \sum_{p\in\mathcal{P}(c)} \boldsymbol{U}_p$. Finally the velocity jump is taken as the maximum velocity jump across the subcell

$$\Delta \boldsymbol{U}_{cp} = 2\max(|\Delta \boldsymbol{U}_1|, |\Delta \boldsymbol{U}_2|), \tag{1.47}$$

where $\Delta \boldsymbol{U}_{1,2}$ are the velocity jumps along $\Delta_{1,2}$ respectively. Finally a switch is added the following way

$$\Delta \boldsymbol{U}_{cp} = \begin{cases} 2\max(|\Delta \boldsymbol{U}_1|, |\Delta \boldsymbol{U}_2|) & \text{if} \quad (\nabla \cdot \boldsymbol{U}_{cp}) \leq 0 \\ 0 & \text{else} \end{cases}, \tag{1.48}$$

where $\nabla \cdot \boldsymbol{U}_{cp}$ is a measure of the velocity divergence within subcell cp. The brute force proof of dissipativity of tensor viscosity is demanding and we end up with conditions which are not explicitely set in the original work [53]

$$\begin{aligned} m_c T_c \frac{d}{dt} S_c &= \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{\mathrm{q}} \cdot \boldsymbol{U}_p \\ &= -\sum_{e\in\mathcal{E}(c), e=[\boldsymbol{X}_p, \boldsymbol{X}_{p+1}]} (\boldsymbol{U}_{p+1} - \boldsymbol{U}_p) \cdot \Big(\\ &\quad (\boldsymbol{U}_p - \boldsymbol{U}_{p-1}) \frac{W_{cp}\mu_{cp}\cos\theta_{cp}}{L_{p-1/2}\sin^2\theta_{cp}} \\ &\quad + (\boldsymbol{U}_{p+1} - \boldsymbol{U}_p) \left[\frac{W_{cp}\mu_{cp}\cos\theta_{cp}}{L_{p+1/2}\sin^2\theta_{cp}} \frac{W_{cp+1}\mu_{cp+1}\cos\theta_{cp+1}}{L_{p-1/2}\sin^2\theta_{cp+1}} \right] \\ &\quad + (\boldsymbol{U}_{p+2} - \boldsymbol{U}_{p+1}) \frac{W_{cp+1}\mu_{cp+1}\cos\theta_{cp+1}}{L_{p+3/2}\sin^2\theta_{cp+1}} \Big), \end{aligned}$$

which is positive for each edge only if simultaneously

$$(\boldsymbol{U}_{p+1} - \boldsymbol{U}_p) \cdot (\boldsymbol{U}_p - \boldsymbol{U}_{p-1}) \leq 0, \quad \text{and} \quad (\boldsymbol{U}_{p+1} - \boldsymbol{U}_p) \cdot (\boldsymbol{U}_{p+2} - \boldsymbol{U}_{p+1}) \leq 0, \tag{1.49}$$

because all terms $\frac{W\mu\cos\theta}{L\sin^2\theta}$ are positive. Reasonnably we adopt (1.49) as definition of the switch $(\nabla \cdot \boldsymbol{U}_{cp}) \leq 0$ which appears in (1.48).

In the last two models of artificial viscosity, edge based and tensorial, a limiter ψ has been introduced to fulfill the wave-front invariance property. However, it has been shown in [95] that such a limiter, even very well adapted when the grid is aligned with the flow, produces some numerical artifacts for grids non aligned with the flow. Such artifacts generaly leads to numerical instabilities and evident loss of symmetry. The design of a valid limiter for non-aligned grid is still an open problem for such artificial viscosity models.

Anti-hourglass subcell force

As already quoted, following [55], we make the fundamental assumption that the subcells are Lagrangian volumes. Consequently following the compression or expansion of the cell, the subcell volume $V_{cp}(t)$ may change. Being Lagrangian, the subcell preserves its mass m_{cp}, hence its density varies as : $\rho_{cp}(t) = m_{cp}/V_{cp}(t)$. As the cell-centered specific internal energy ε_c is constant inside the cell, we use the equation of a state to define the subcell pressure as

$$P_{cp} = P(\rho_{cp}, \varepsilon_c), \qquad \text{with} \quad \rho_{cp} = m_{cp}/V_{cp}. \tag{1.50}$$

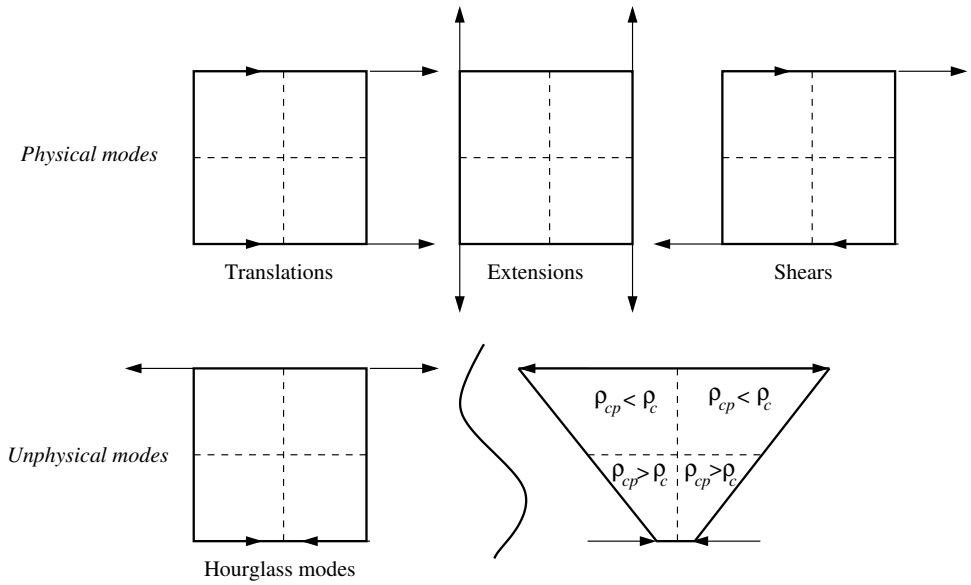


Figure 1.3 – *Physical and unphysical modes of a quadrilateral cell. Top : translations, extensions and shears (considering symmetry) — Bottom : unphysical hourglass modes. All but the two hourglass modes are physical but only for the hourglass modes do the subcell densities differ from the cell density.*

Subcell pressure force has been initially introduced by Caramana and Shashkov [56] to control artificial grid distortions, such as the hourglass modes[3]. In order to illustrate this effect let us recall that a quadrilateral cell has eight degrees of freedom : two translations, two extensions, two shears and two hourglass modes, see figure 1.3. All but the two hourglass modes are physical but only for the hourglass modes does the subcell density differ from the cell density to which it belongs. Within a cell we observe $\rho_{cp} > \rho_c$ for several subcells and $\rho_{cp} < \rho_c$ for others. The subcell pressures P_{cp} also differ from the cell pressure P_c. The subcell pressure method uses this effect to calculate subcell forces that are proportional to the difference between the subcell and the cell pressures, and oppose the hourglass motion. In this approach, the *subcell anti-houglass* force is defined as

$$\boxed{\boldsymbol{F}_{cp}^{\Delta P} = L_{cp}(P_{cp} - P_c)\boldsymbol{N}_{cp} + \frac{1}{2}\left[\left(P_{cp} - P_{cp^-}\right) L_{cp}^{-}\boldsymbol{N}_{cp}^{-} + \left(P_{cp} - P_{cp^+}\right) L_{cp}^{+}\boldsymbol{N}_{cp}^{+}\right]} \tag{1.51}$$

where P_{cp^-} and P_{cp^+} are the previous and next neighbor subcell pressures with respect to subcell cp and $L_{cp}^{\pm}\boldsymbol{N}_{cp}^{\pm}$ are the internal geometrical vector to subcell cp see Fig. 1.1. The subcell force $\boldsymbol{F}_{cp}^{\Delta P}$ is usually multiplied by a merit factor z_{merit} which ranges from 0 to 1.
Recall that conservation of global momentum, $\mathcal{Q}$ see (1.29), is implied by the relation on subcell forces (1.28), that is to say $\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp} = \boldsymbol{0}$. Let us prove that anti-hourglass subcell forces also verify

$$\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{\Delta P} = \boldsymbol{0}. \tag{1.52}$$

3. Other attempts to damp such artificial grid distortions can be found with artificial viscosity-like terms in Wilkins or using SUPG stabilized formulation [96].

Let us start by summing (1.51) for all points p of cell c

$$\begin{aligned}\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{\Delta P} &= \sum_{p\in\mathcal{P}(c)} L_{cp}(P_{cp}-P_c)\boldsymbol{N}_{cp} + \frac{1}{2}\left[\left(P_{cp}-P_{cp^-}\right)L_{cp}^-\boldsymbol{N}_{cp}^- + \left(P_{cp}-P_{cp^+}\right)L_{cp}^+\boldsymbol{N}_{cp}^+\right] \\ &= \sum_{p\in\mathcal{P}(c)} L_{cp}P_{cp}\boldsymbol{N}_{cp} + \frac{1}{2}\left[\sum_{p\in\mathcal{P}(c)}\left(P_{cp}-P_{cp^-}\right)L_{cp}^-\boldsymbol{N}_{cp}^- + \sum_{p\in\mathcal{P}(c)}\left(P_{cp}-P_{cp^+}\right)L_{cp}^+\boldsymbol{N}_{cp}^+\right],\end{aligned} \tag{1.53}$$

where we have used the geometrical identity (1.14) to deduce $\sum_{p\in\mathcal{P}(c)} L_{cp}P_c\boldsymbol{N}_{cp} = \boldsymbol{0}$. Let us focus on the square bracket terms

$$\mathcal{SB} = \frac{1}{2}\left[\sum_{p\in\mathcal{P}(c)} P_{cp}\left(L_{cp}^-\boldsymbol{N}_{cp}^- + L_{cp}^+\boldsymbol{N}_{cp}^+\right) - \sum_{p\in\mathcal{P}(c)} P_{cp^-}L_{cp}^-\boldsymbol{N}_{cp}^- - \sum_{p\in\mathcal{P}(c)} P_{cp^+}L_{cp}^+\boldsymbol{N}_{cp}^+\right],$$

Assuming periodic boundary conditions we can use index shifts in the second and third sums (p^- becomes p and p^+ becomes p) to obtain

$$\begin{aligned}\mathcal{SB} &= \frac{1}{2}\left[\sum_{p\in\mathcal{P}(c)} P_{cp}\left(L_{cp}^-\boldsymbol{N}_{cp}^- + L_{cp}^+\boldsymbol{N}_{cp}^+\right) - \sum_{p\in\mathcal{P}(c)} -P_{cp}L_{cp}^+\boldsymbol{N}_{cp}^+ - \sum_{p\in\mathcal{P}(c)} -P_{cp}L_{cp}^-\boldsymbol{N}_{cp}^-\right], \\ &= \sum_{p\in\mathcal{P}(c)} P_{cp}\left(L_{cp}^-\boldsymbol{N}_{cp}^- + L_{cp}^+\boldsymbol{N}_{cp}^+\right) \\ &= -\sum_{p\in\mathcal{P}(c)} L_{cp}P_{cp}\boldsymbol{N}_{cp}.\end{aligned} \tag{1.54}$$

Back substituting this last equation into (1.53) yields the expected result

$$\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{\Delta P} = \boldsymbol{0}.$$

Total subcell/nodal force

The total subcell force that applies onto point p from cell c is constituted of the pressure, artificial viscosity and anti-hourglass forces

$$\boxed{\boldsymbol{F}_{cp} = \boldsymbol{F}_{cp}^{\text{press}} + \boldsymbol{F}_{cp}^{q} + \boldsymbol{F}_{cp}^{\Delta P}.} \tag{1.55}$$

Recall that many other physical models could be added to the system of equations and, thanks to the compatible discretization, this would induce other types of subcell forces to be added to (1.55). By summation of subcells cp around a point p we construct the total nodal force that applies onto p as

$$\boldsymbol{F}_p = \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp}, \tag{1.56}$$

which is further used in (1.25) to compute the evolution of the momentum equation.

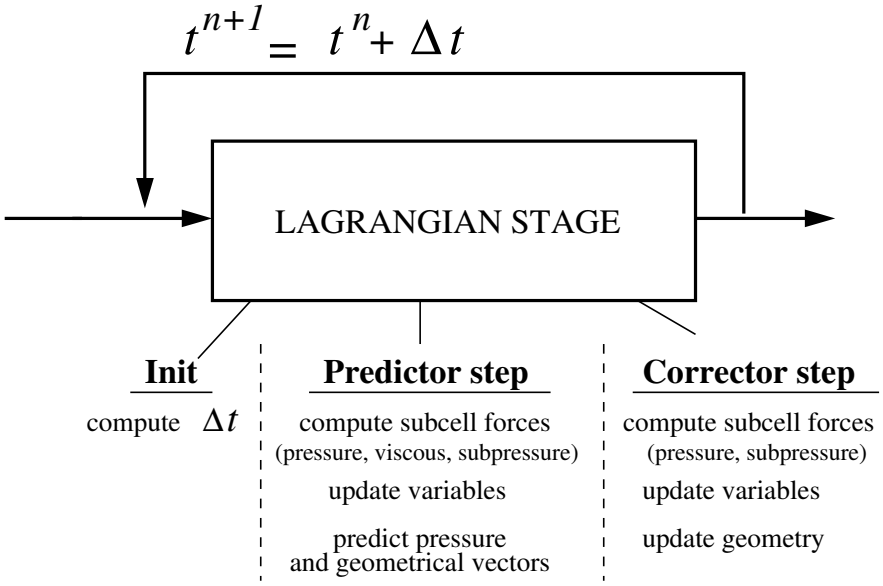


Figure 1.4 – *Simplistic view of the predictor-corrector temporal scheme used in the compatible staggered Lagrangian stage. See the algorithm in the text for an exhaustive description.*

1.2.4 Time discretization

The time discretization is obtained by means of a classical two-step predictor-corrector scheme to gain second-order accuracy as presented in Fig. 1.4 and in the following algorithm.
Being given geometric quantities and physical variables at time t^n, we first predict the time centered geometrical quantities and pressures that are later used in the corrector step to update physical and geometric variables. There exist several other ways to exhibit a predictor-corrector scheme for this system, each having some interesting properties and drawbacks (as instance one can avoid to call the equation of state routine in the predictor phase or one can avoid to update the velocity in the predictor phase). Nevertheless this algorithm is symmetric (apart from steps 0. and 9. which are specific to the predictor step) which simplifies its implementation and the associated code maintenance.

Predictor step.

0. Compute subcell artificial viscous force $\boldsymbol{F}_{cp}^{q,n}$, deduce the time step Δt
1. Compute subcell pressure force $\boldsymbol{F}_{cp}^{\text{press},n} = -L_{cp}^{n} P_c^n \boldsymbol{N}_{cp}^{n}$
2. Compute subcell anti-hourglass force $\boldsymbol{F}_{cp}^{\Delta P,n}$
3. Compute total subcell forces

$$\boldsymbol{F}_{cp}^{n} = \boldsymbol{F}_{cp}^{\text{press},n} + \boldsymbol{F}_{cp}^{q,n} + \boldsymbol{F}_{cp}^{\Delta P,n}$$

4. Update momentum equation

$$\begin{aligned} m_p \left(\boldsymbol{U}_p^{n+1} - \boldsymbol{U}_p^{n} \right) &= -\Delta t \sum_{c \in \mathcal{C}(p)} \boldsymbol{F}_{cp}^{n} \\ \boldsymbol{U}_p^{n+1/2} &= \frac{1}{2} \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^{n} \right) \end{aligned}$$

5. Update internal energy equation

$$m_c \left(\varepsilon_c^{n+1} - \varepsilon_c^{n} \right) = \Delta t \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp}^{n} \cdot \boldsymbol{U}_p^{n+1/2}$$

6. Update vertex position

$$\boldsymbol{X}_p^{n+1} = \boldsymbol{X}_p^{n} + \Delta t \, \boldsymbol{U}_p^{n+1/2}$$

7. Recompute cell/subcell volumes, geometrical entities and densities at t^{n+1}

$$\rho_c^{n+1} = \frac{m_c}{V_c^{n+1}}, \qquad \rho_{cp}^{n+1} = \frac{m_{cp}}{V_{cp}^{n+1}}$$

8. Compute updated pressures

$$\begin{aligned} P_c^{n+1} &= P\left(\rho_c^{n+1}, \varepsilon_c^{n+1} \right) \\ P_{cp}^{n+1} &= P\left(\rho_{cp}^{n+1}, \varepsilon_{cp}^{n+1} \right) \end{aligned}$$

9. Compute time centered geometrical entities and predicted pressures

$$\begin{aligned} P_c^{n+\frac{1}{2}} &= \frac{1}{2} \left(P_c^{n+1} + P_c^{n} \right) \\ P_{cp}^{n+\frac{1}{2}} &= \frac{1}{2} \left(P_{cp}^{n+1} + P_{cp}^{n} \right) \end{aligned}$$

Corrector step.

0. —
1. Compute subcell pressure force $\boldsymbol{F}_{cp}^{\text{press},n+1/2} = -L_{cp}^{n+1/2} P_c^{n+1/2} \boldsymbol{N}_{cp}^{n+1/2}$
2. Compute subcell anti-hourglass force $\boldsymbol{F}_{cp}^{\Delta P,n+1/2}$
3. Compute total subcell forces

$$\boldsymbol{F}_{cp}^{n+1/2} = \boldsymbol{F}_{cp}^{\text{press},n+1/2} + \boldsymbol{F}_{cp}^{q,n} + \boldsymbol{F}_{cp}^{\Delta P,n+1/2}$$

4. Update momentum equation

$$\begin{aligned} m_p \left(\boldsymbol{U}_p^{n+1} - \boldsymbol{U}_p^{n} \right) &= -\Delta t \sum_{c \in \mathcal{C}(p)} \boldsymbol{F}_{cp}^{n+1/2} \\ \boldsymbol{U}_p^{n+1/2} &= \frac{1}{2} \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^{n} \right) \end{aligned}$$

5. Update internal energy equation

$$m_c \left(\varepsilon_c^{n+1} - \varepsilon_c^{n} \right) = \Delta t \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp}^{n+1/2} \cdot \boldsymbol{U}_p^{n+1/2}$$

6. Update vertex position

$$\boldsymbol{X}_p^{n+1} = \boldsymbol{X}_p^{n} + \Delta t \, \boldsymbol{U}_p^{n+1/2}$$

7. Recompute cell/subcell volumes, geometrical entities and densities at t^{n+1}

$$\rho_c^{n+1} = \frac{m_c}{V_c^{n+1}}, \qquad \rho_{cp}^{n+1} = \frac{m_{cp}}{V_{cp}^{n+1}}$$

8. Compute updated pressures

$$\begin{aligned} P_c^{n+1} &= P\left(\rho_c^{n+1}, \varepsilon_c^{n+1} \right) \\ P_{cp}^{n+1} &= P\left(\rho_{cp}^{n+1}, \varepsilon_{cp}^{n+1} \right) \end{aligned}$$

End of time step.
Final data $\rho_c^{n+1}, \rho_{cp}^{n+1}$, ε_c^{n+1} and P_c^{n+1}, then $\boldsymbol{U}_p^{n+1}$, $\boldsymbol{X}_p^{n+1}$ and mesh related entities (volumes, lengths, corner vectors, etc.).

In the following we focus on important details of several steps of the algorithm.

0. Compute subcell artificial viscous force $\boldsymbol{F}_{cp}^{q,n}$ and deduce the time step Δt.
This step is only performed for the prediction. The time step is constrained by the classical CFL condition which roughly states that any information can not travel across more than one cell during the time step. In other words

$$\Delta t \leq \min_c \left(\frac{L_c^{\text{characteristic}}}{S_c} \right), \tag{1.57}$$

where $L_c^{\text{characteristic}}$ is a cell-based characteristic length. We take $L_c^{\text{characteristic}} = \min_{e \in \mathcal{E}(c)} L_e$ where $\mathcal{E}(c)$ is the set of edges of cell c. Moreover S_c is the characteristic sound-speed of cell c which is constituted of a_c the actual sound-speed in cell c and a "viscous" sound speed obtained from the artificial viscous model a_c^{viscous}, therefore $S_c = \sqrt{a_c^2 + (a_c^{\text{viscous}})^2}$. The occurence of a "viscous" sound speed is mandatory to allow a valid definition of a time step when the initial specific internal energy is zero or close to zero, which in the case of a perfect gas leads to a sound-speed a_c close to zero, hence a time step tending to infinity. In addition to this we classicaly multiply the time step with a security coefficient of $1/4$ hence the time step is given by

$$\Delta t \leq \frac{1}{4} \min_c \left(\frac{\min_{e \in \mathcal{E}(c)} L_e}{\sqrt{a_c^2 + (a_c^{\text{viscous}})^2}} \right). \tag{1.58}$$

We notice that the viscous subcell forces are only computed at time t^n and not updated for the corrector step. Up to our knowledge there is no rigorous reason to justify this point. The fact that updating the viscous force does not drastically modify the results and the relative cost of such step seem two acceptable points in favour of such legacy.
Finally in order to avoid too violent cell volume change during one single time step, Δt is not allowed to increase by more than 20% percents compared to its previous value.

1- Compute subcell pressure force $\boldsymbol{F}_{cp}^{\text{press},*} = -L_{cp}^* P_c^* \boldsymbol{N}_{cp}^*$.
Note that the time centering of the geometrical entities $-L_{cp}^* \boldsymbol{N}_{cp}^*$ and the pressure P_c^* are the same. This is the reason why at the end of the predictor step all geometrical entities must be recomputed to match the time centering of the predicted pressure (at time $t^{n+1/2}$ in our algorithm).

5- Update internal energy equation

$$m_c \left(\varepsilon_c^{n+1} - \varepsilon_c^n \right) = \Delta t \sum_{p \in \mathcal{P}(c)} \boldsymbol{F}_{cp}^* \cdot \boldsymbol{U}_p^{n+1/2}. \tag{1.59}$$

Using $\boldsymbol{U}_p^{n+1/2} = \frac{1}{2} \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^n \right)$ in the equation above is mandatory to fulfill the total energy conservation. To prove it let us start from the momentum equation

$$m_p \left(\boldsymbol{U}_p^{n+1} - \boldsymbol{U}_p^n \right) = -\Delta t \sum_{c \in \mathcal{C}(p)} \boldsymbol{F}_{cp}^*. \tag{1.60}$$

Multiplying (1.60) by $\frac{1}{2} \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^n \right)$ yields the evolution of the kinetic energy equation for point p

$$\begin{aligned} \frac{1}{2} m_p \left(\boldsymbol{U}_p^{n+1} - \boldsymbol{U}_p^n \right) \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^n \right) &= -\Delta t \sum_{c \in \mathcal{C}(p)} \boldsymbol{F}_{cp}^* \cdot \frac{1}{2} \left(\boldsymbol{U}_p^{n+1} + \boldsymbol{U}_p^n \right), \\ \frac{1}{2} m_p \left(\boldsymbol{U}_p^{n+1} \right)^2 &= \frac{1}{2} m_p \left(\boldsymbol{U}_p^n \right)^2 - \Delta t \sum_{c \in \mathcal{C}(p)} \boldsymbol{F}_{cp}^* \cdot \boldsymbol{U}_p^{n+1/2}. \end{aligned}$$

By suming over all points p (and neglecting boundary conditions or assuming periodic boundary conditions) one gets

$$\frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n+1}\right)^2 = \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n}\right)^2 - \Delta t \sum_p \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp}^{*}\cdot\boldsymbol{U}_p^{n+1/2}. \tag{1.61}$$

On the other hand rewritting (1.59) under the form

$$m_c\varepsilon_c^{n+1} = m_c\varepsilon_c^{n} + \Delta t \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{*}\cdot\boldsymbol{U}_p^{n+1/2},$$

and suming over all cells c yields

$$\sum_c m_c\varepsilon_c^{n+1} = \sum_c m_c\varepsilon_c^{n} + \Delta t \sum_c \sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{*}\cdot\boldsymbol{U}_p^{n+1/2}. \tag{1.62}$$

At last adding (1.61) and (1.62) gives

$$\begin{aligned}\sum_c m_c\varepsilon_c^{n+1} + \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n+1}\right)^2 &= \sum_c m_c\varepsilon_c^{n} + \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n}\right)^2 \\ &\quad -\Delta t\left(\sum_p\sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp}^{*}\cdot\boldsymbol{U}_p^{n+1/2} - \sum_c\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp}^{*}\cdot\boldsymbol{U}_p^{n+1/2}\right)\end{aligned}$$

switching the sum signs in one of the two terms in parenthesis (i.e. $\sum_p\sum_{c\in\mathcal{C}(p)} \equiv \sum_c\sum_{p\in\mathcal{P}(c)}$) must convince the reader that the term in parenthesis is indeed equal to zero. Therefore the previous equation reduces to

$$\sum_c m_c\varepsilon_c^{n+1} + \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n+1}\right)^2 = \sum_c m_c\varepsilon_c^{n} + \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p^{n}\right)^2,$$

which clearly states that if total energy is defined as $E = \sum_c m_c\varepsilon_c + \frac{1}{2}\sum_p m_p \left(\boldsymbol{U}_p\right)^2$, then this quantity is conserved only if the subcell force is dot-producted with $\boldsymbol{U}_p^{n+1/2}$ in equation (1.59).

8- Compute updated pressures

$$P_c^{n+1} = P\left(\rho_c^{n+1}, \varepsilon_c^{n+1}\right), \qquad P_{cp}^{n+1} = P\left(\rho_{cp}^{n+1}, \varepsilon_{cp}^{n+1}\right).$$

The subcell pressures could instead be computed in steps 2. if the anti-hourglass forces $\boldsymbol{F}_{cp}^{\Delta P,n}$ are to be evaluated. (The subcell densities can be also computed during this step as both subcell densities and subcell pressures are only utilized to compute anti-hourglass forces). Consequently, depending on implementation or efficiency reasons the computation of subcell entities can be removed from steps 7. and 8. and moved to steps 2. of the algorithm.

9- Compute time centered geometrical entities and predicted pressures

$$P_c^{n+\frac{1}{2}} = \frac{1}{2}\left(P_c^{n+1} + P_c^{n}\right), P_{cp}^{n+\frac{1}{2}} = \frac{1}{2}\left(P_{cp}^{n+1} + P_{cp}^{n}\right).$$

The goal of the prediction step is to center all entities at time $t^{n+1/2}$ which are later used to compute forces and advance point position (and by association geometrical entities like cell volume and cell density), point velocity and cell centered specific internal energy. These pressures contribute to the subcell pressure forces computed in step 1. of the corrector stage.

1.2.5 Boundary conditions

Boundary conditions in a Lagrangian formulation are of two different types : prescribed normal velocity or pressure.
If a prescribed normal velocity is enforced along two edges impinging at a boundary point p, say $v_p^* = \boldsymbol{U}_p \cdot \boldsymbol{N}_p$ where $\boldsymbol{N}_p$ is a unit outward normal at point p, then it is usually considered as sufficient to add a velocity correction between the two equations of step 4. in the algorithm of section 1.2.4 both for predictor and corrector steps. This sub-step consists of modifying the point velocity $\boldsymbol{U}_p^{n+1}$ after its evaluation in such a way that its normal component be equal to v_p^* and its tangential component remain unchanged. Indeed if we call $\boldsymbol{T}_p$ the unit tangential vector at point p, this amounts to find the components of velocity $\widehat{\boldsymbol{U}}_p$ such that :

$$\widehat{\boldsymbol{U}}_p \cdot \boldsymbol{N}_p = v_p^*, \tag{1.63}$$

$$\widehat{\boldsymbol{U}}_p \cdot \boldsymbol{T}_p = \boldsymbol{U}_p^{n+1} \cdot \boldsymbol{T}_p, \tag{1.64}$$

knowing v_p^*, $\boldsymbol{U}_p^{n+1}$, $\boldsymbol{N}_p$ and $\boldsymbol{T}_p$. The boundary condition friendly velocity $\widehat{\boldsymbol{U}}_p$ is given by

$$\widehat{\boldsymbol{U}}_p = v_p^* \boldsymbol{N}_p - (\boldsymbol{U}_p^{n+1} \cdot \boldsymbol{T}_p)\boldsymbol{T}_p. \tag{1.65}$$

For pressure boundary condition P^* we usually define ghost subcells around boundary points for which we set the pressure to be P^*. This modification is to be operated before the update of momentum equation, say between step 3. and 4. This amounts to modify (1.35) for any subcell having an edge on the boundary line as if a subcell from a ghost cell c' is present

$$\boldsymbol{F}_{cp}^{\text{press}} = -L_{cp}\, P_c\, \boldsymbol{N}_{cp} - L_{c'p}\, P^*\, \boldsymbol{N}_{c'p}. \tag{1.66}$$

By construction of the ghost subcell we have $L_{c'p}\boldsymbol{N}_{c'p} = -L_{cp}\boldsymbol{N}_{cp}$, hence

$$\boldsymbol{F}_{cp}^{\text{press}} = -L_{cp}\, (P_c - P^*)\, \boldsymbol{N}_{cp}. \tag{1.67}$$

Obviously v^* and P^* may be space/time dependent boundary conditions so that accelerated piston, spacial varying pressure boundary conditions as well as no-slip boundary conditions and constant pressure can be easily applied.

Note that most of the previous proofs (conservation, Galilean invariance, etc) where periodic boundary conditions were assumed can be revamped using more complex boundary conditions, the principles of the proof being the same, only the equations are more involved, see [42] as instance.

1.2.6 Cylindrical $r-z$ geometry

This compatible staggered Lagrangian scheme has also been extended to 2D $r-z$ cylindrical geometry either using a so-called *control volume* (CV) discretization, which does not maintain cylindrical symetry, or a so-called *area-weighted* (AW) discretization which can [55]. Note that our section is freely inspired from Section 3 of paper [55], and, undoubtedly, the reader must compulse the aforementioned paper.

Control volume formulation

The description of control volume (CV) formulation can be found in [55] but its formulation surely dates back to the 50's. Let us provide a brief description of this formulation. First we call r_p, z_p the coordinates of a point $\boldsymbol{X}_p$. The formulation starts with a true cylindrical cell volume definition

$$V_c = \int_{\Omega_c} r \, dr \, dz. \tag{1.68}$$

This volume integral is indeed a function of point positions. Any geometrical entities such as length, normals, surfaces can be also derived in cylindrical geometry. (A more advanced description of all algebraic manipulations involved in this formulation can also be found in [97].) Then using (1.13), a discrete divergence operator can be defined. Accordingly the associated discrete gradient can be derived. This essentially determines the form of the geometrical corner vector $L_{cp}\boldsymbol{N}_{cp}$. Subcell mass is also computed taking into account the cylindrical volumes which following (1.10) determines the cell and point masses. Nodal force can then be constructed and finally the energy equation is updated in a compatible way. As we can observe the same 'volume' definition is used both to compute density and cell volume but also the work within the energy equation. Consequently total energy and momentum are conserved.
The problem of CV formulation is that it will not preserve two-dimensional cylindrical $r-z$ symetry of a spherical flow. As quoted by Caramana in [55] and further observed in simple numerical examples, *one can specify one-dimensional, spherically symmetric initial and boundary conditions and the numerical solution, computed with the control volume scheme described above, will not remain spherical in time*. Spherical symmetry CV formulation is violated because the areas along the angular direction are not equal even when the angles between the radial lines are equal. Therefore for symmetric pressure distribution along radius the calculated force can not be radial, leading to symmetry violation. However for an equal angular mesh, 2D cylindrical symmetry is preserved as in Cartesian geometry because the lengths along the angular direction are equal. Consequently the computed nodal force perpendicular to the radial direction vanishes for a spherically symmetric distribution of pressures [63].

Area-weighted formulation

The description of area-weighted (AW) formulation can be found in [55] but it dates back to the time of the Green Book [69, 98, 82] and revamped in [99, 100]. This method is an example of discretization of axisymetrical equations which preserves spherical symmetry of the numerical solution on equal angular mesh. As a consequence AW is often the prefered Lagrangian discretization for problem with cylindrical geometry. For this type of scheme one begins by postulating the form of the gradient operator, based on physical reasoning of what is necessary for symmetry preservation for an equal angular mesh. This implicitly determines the zone volume definition. Strictly speaking such schemes violate momentum conservation ; in compatible form they may give rise to entropy errors. As seen previously for an equal angular mesh, 2D cylindrical symmetry is preserved in Cartesian geometry. This property is used to construct the area-weighted schemes in cylindrical geometry focusing on preserving this spherical symmetry. To obtain the area-weighted schemes one simply multiplies the vector lengths, as deïňĄned in Cartesian geometry, of the entire force contour defined with respect to a given point, p, by the value of the radius r_p. Then the Lagrangian nodal mass m_p is also defined at point p as an effective local inertia $(\rho A)_p$ times r_p

$$m_p = r_p(\rho A)_p, \tag{1.69}$$

so that the momentum equation (1.16) in cylindrical geometry becomes

$$r_p(\rho A)_p \frac{d}{dt}\boldsymbol{U}_p = -\sum_{c\in\mathcal{C}(p)} r_p \boldsymbol{F}_{cp},$$

where $\boldsymbol{F}_{cp}$ is the subcell Cartesian force. Then r_p cancels in the previous equation yielding

$$(\rho A)_p \frac{d}{dt}\boldsymbol{U}_p + \sum_{c\in\mathcal{C}(p)} \boldsymbol{F}_{cp} = \boldsymbol{0}, \tag{1.70}$$

which is essentially the same as (1.16) for Cartesian geometry. The area inertia is further defined using the true cylindrical initial mass m_p using definition (1.69). Caramana provides in [55] a simple way to compute nodal, cell and subcell masses and volumes in cylindrical geometry : in short the cell volume is split into $\mathcal{P}(c)$ Cartesian triangular volumes [4] denoted $A_{pp'}$ multiplied by the "radius" $\frac{1}{3}(r_p + r_{p'} + r_c)$, so that the AW cylindrical cell volume is

$$V_c = \sum_{p\in\mathcal{P}(c)} A_{pp'} \frac{1}{3}(r_p + r_{p'} + r_c). \tag{1.71}$$

Finally the specific internal energy evolution is computed by the work obtained by multiplying each subcell force by the nodal radius times the nodal velocity

$$m_c \frac{d}{dt}\varepsilon_c - \sum_{p\in\mathcal{P}(c)} r_p \boldsymbol{F}_{cp}\cdot\boldsymbol{U}_p = 0, \tag{1.72}$$

where once again $\boldsymbol{F}_{cp}$ is the subcell Cartesian force. Due to the presence of r_p in front of $\boldsymbol{F}_{cp}$ in (1.72) one deduces that momentum can not be conserved [5]
Unfortunately keeping in mind the future use of this Lagrangian scheme within a conservative ALE code the AW formulation can not be considered. Because the conservative remapping step relies on integral definition of volume and mass, but, as already seen, the AW cell volume (1.71) can not be reinterpreted into a proper integral formula contrarily to CV formulation. This enforces us to abandon the AW formulation in favour of the CV formulation for a conservative ALE code.

1.2.7 Discussion

The central feature of this more modern form of Lagrangian hydrodynamics is its ability to exactly conserve mass, momentum, and total energy without the need to use these quantities directly as variables. It instead retains density, velocity, and specific internal energy as dependent variables as did the earlier version of this algorithm [65]. Total energy conservation is ensured by the use of a "compatible" discretization while the conservation of momentum is obtained by assuring that subcell forces, no matter which physical process they may represent, do sum to zero within a cell. Finally mass conservation is trivially fulfilled due to the Lagrangian formalism.
Even if this is not presented in this short description, many different physical models can and have been coupled to this staggered Lagrangian scheme : elasto-plasticity, radiative transport, diffusion equation, multi-material treatment, etc. Consequently this Lagrangian scheme has been used and still is in many Lagrangian or ALE simulation codes. Nevertheless analysis and understanding of its intricate nature is still an on-going work even if the scheme is ancient.
Some of such investigations are presented in the next section.

4. Such triangle is the zone defined with two adjacent points p and p' and the cell center c.
5. Indeed because of factor r_p, (1.28) turns into $\sum_{p\in\mathcal{P}(c)} r_p \boldsymbol{F}_{cp} \neq \boldsymbol{0}$, in general.

1.3 Numerical analysis

This section presents some results on the numerical analysis of the compatible staggered Lagrangian scheme. More specifically the internal consistency, accuracy and stability issues are addressed in the first subsection as paper [12] is presented. Then the proof of a result from [12] which has been published in [15] is summarized in the second subsection. Finally in the third subsection the volume consistency of the scheme is investigated by briefly reviewing the results of paper [16].

1.3.1 Internal consistency, accuracy and stability

Led by E.J. Caramana, a joined effort with several colleagues from LANL gave rise in 2006 to paper [12] the title of which is *"The internal consistency, accuracy and stability of the discrete compatible Formulation of Lagrangian Hydrodynamics"*. The goals of this work were

To study the internal consistency of the scheme by analyzing the difference between the two definitions of cell volume the scheme utilizes : A compatible cell volume V_c^{comp} deduced from the discrete version of the divergence of the velocity (1.11) and a cell volume obtained from point coordinates $V_c^{coord} = f\left(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_{|\mathcal{P}(c)|}\right)$ [6] The derivatives of the cell volume with respect to its coordinate dependence is used to define the geometrical vectors (1.12) associated with point p, as was done in the work of Favorskii [71], and Margolin and Adams [72]. Thus the geometrical vectors used to construct the subcell forces are not arbitrarily specified, as with the older versions of this type of hydrodynamics [69], but are a consequence of the chosen volume definitions. The analysis in the article shows that if the geometrical vectors are time-centered and for zero force then the two volumes in 2D Cartesian geometry are equal, *a posteriori* justifying the $t^{n+1/2}$ time centering at the end of the predictor phase (step 9. of the predictor stage of the algorithm in section 1.2.4). When the forces are not zero then it is shown that the difference in coordinate and compatible volumes is of order Δt^3 on a single timestep and the time accuracy (globally integrated up to a final time t^n) is of the order Δt^2. It was shown that this difference can be used to ascertain many properties of a simulation, and as such has direct and practical significance.

To construct non-dimensional internal consistency norms based on the difference in these two aforementioned volumes. These can be used to operationally measure the non-dimensional magnitude of the truncation error of a calculation by placing the geometrical vectors from which the subcell force is calculated at the fully advanced time level on the corrector step, that is to say t^{n+1} instead of $t^{n+1/2}$.

To validate the error indicators on a set of numerical tests (Guderley, Noh, Sedov in 2D cylindrical geometry and also in 3D). They also serve to illustrate how they can be utilized to assess the quality of numerical simulations. We have demonstrated that the size of the error associated with the coordinate and compatible volumes is significant only when severe numerical difficulties, such as numerical instability, arise. The accuracy in both space and time were also measured, and results were found to correspond to the first or second order accuracy that one expects in space or time with the type of finite-volume differencing employed.

To analyse the numerical stability of the two level predictor-corrector time integration scheme employed : First with a simplified set of model equations using standard Fourier stability analysis, and then using a properly constructed test problem that verifies this analysis for the

6. In the article a cell is refered to as a "zone" z, a subcell to a "corner", and these two volumes are denoted V_z^{crd} and V_z^{cmp}.

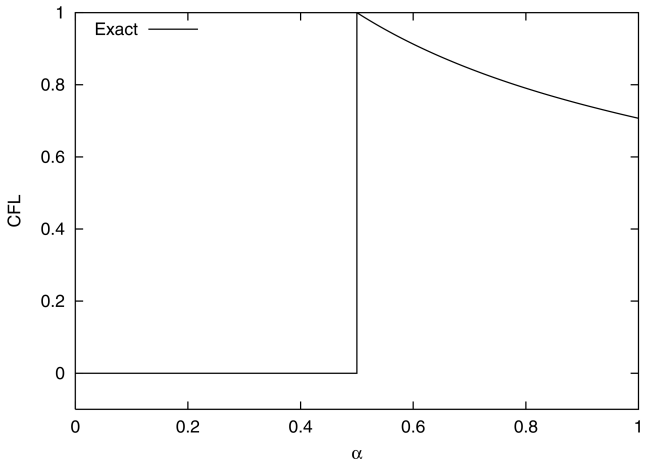


Figure 1.5 – *Numerical results from paper [12]. Maximum CFL number as a function of parameter α.*

actual system of equations in all three spatial dimensions. In this stability analysis we studied the effect of the time centering parameter α of the pressure on the corrector step (step 9. of the predictor step of the algorithm in section 1.2.4) when written as :

$$P_c^{n+\frac{1}{2}} = \alpha P_c^{n+1} + (1-\alpha) P_c^n .$$

We numerically showed that our compatible system of equations with predictor-corrector time integration is stable for CFL number $CFL \leq 1/\sqrt{2\alpha}$ with $\alpha \geq 1/2$; otherwise, it is unconditionally unstable (see the Fig. 1.5). This somewhat justifies the optimal choice of $\alpha = 1/2$ which maximizes the usable CFL coefficient.

1.3.2 Stability (again)

A non-classical stability bound has been uprised in [12]. However we were not able to rigorously prove it and we had to resort to numerical sampling only at the very end of the proof to validate this stability bound cleverly conjectured by B. Wendroff. This was unfortunate. Consequently with the help of B. Wendroff we attacked this proof again in paper [15] entitled *On stabiliy analysis of staggered schemes*.
Some years ago, M. Shashkov, proposed a simple problem for the purpose of testing the stability of Lagrangian hydrodynamics codes. The initial data are given as zero velocity, constant pressure and constant density. One then computes the total kinetic energy, which should be zero for all time. We have discovered in [12] that it was unstable for certain mesh ratios that "folklore" indicated should be otherwise — the instability manifested itself as an explosive growth of kinetic energy. The surprise here is not that a 1D symmetry preserving 2D code could be unstable for 1D data; the surprise lies in the contradiction of the "folklore". Briefly, the compatible staggered Lagrangian scheme has two parameters, α and β[7] such that if $\alpha = \beta = 1/2$, the scheme resembles a time centered (Crank-Nicolson) scheme, while if $\alpha = \beta = 1$, it looks like backward Euler. For the Courant-Friedrichs-Lewy (CFL) condition CFL $= 1$, the scheme is stable for $\alpha = \beta = 1/2$. However, for $\alpha = 1$, $\beta = 1/2$ the scheme is unstable for CFL > 0.71, indicating that as α increases, the CFL limit decreases.

7. One parameter for each 'entity location', that is to say α for cell-centered variables, β for node-centered ones.

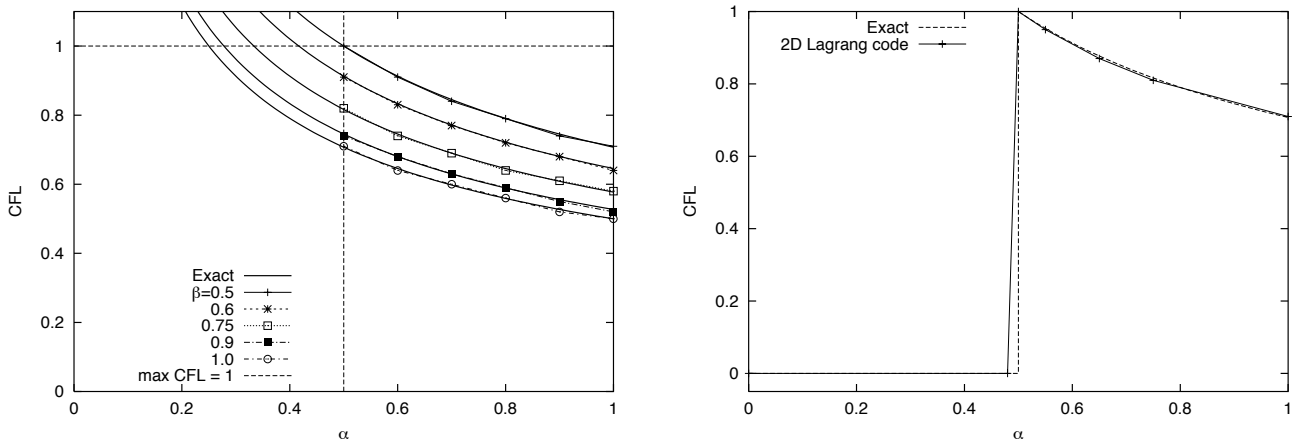


FIGURE 1.6 – *Numerical results from paper [15]. Maximum CFL number λ as a function of α (β fixed) for the kinetic energy to remain on the order of machine precision for the stability test case of M.Shashkov — **(a)** Wave equations. The vertical dashed line is the $\alpha = 1/2$ limit, the horizontal one is the CFL $= 1$ limit. The Theorem predicts the continuous thick lines, the code produces the data for different β. Any scheme defined by a value $\alpha \geq 1/2$ and $\beta \geq 1/2$ is stable with the following CFL number $\lambda \leq 1/(2\sqrt{\alpha\beta})$ — **(b)** Hydrodynamics equations for $\beta = 1/2$. The therorem predicts the dashed line ; the 2D code produces the continuous line. Any scheme defined by a value $\alpha \geq 1/2$ is stable with the CFL number $\lambda \leq 1/\sqrt{2\alpha}$ and is unstable otherwise.*

In [12] we have created a 1D problem using the same data ; we seed a 1D staggered-grid predictor-corrector compatible Hydrodynamics Lagrangian code with a small random perturbation of the pressure. Moreover, we have created the multi-dimension version of the problem and have run our 2D and Caramana's 3D code as well. We observed the same phenomenon in 1D, 2D, and 3D [12]. The stability limit was conjectured by B. Wendroff to be CFL $= 1/(2\sqrt{\alpha\beta})$, and numerical tests in [12] using the full nonlinear equations show that this is true provided $\alpha \geq 1/2$ and $\beta \geq 1/2$. In this paper we prove the above conjecture in 2D, which contains 1D as a special case. Of course, there is no possibility of doing this for the full nonlinear problem - the Euler equations. Ultimately, one applies a von Neumann analysis to the linearized system. In [12] we almost succeeded in doing this theoretical analysis in 1D. In this paper we have succeeded in 2D, by using the numerical radius of the amplification matrix as a tool, an idea apparently first applied in [101]. This proof is the goal of this paper.

We showed on wave equations as a model equation that the staggered implicit scheme is unconditionally stable for $\alpha \geq 1/2$, $\beta \geq 1/2$; moreover we showed that the predictor-corrector staggered scheme is stable for $\alpha \geq 1/2$, $\beta \geq 1/2$ and CFL $\leq 1/(2\sqrt{\alpha\beta})$. In a specific paragraph we demonstrated on a specific 1D example that the schemes for $\alpha < 1/2$ or $\beta < 1/2$ can not be stable as some Fourier components are amplified. Finally, we have showed 2D numerical results for the wave and Euler equations using a compatible Lagrangian Hydrodynamics code. In Fig. 1.6 we reproduce the graphics from [15] showing the numerical experiments made with the wave equations (panel **(a)**) and the hydrodynamics equation (panel **(b)**). The exact and experimental maximal CFL number as a function of α, β and only α are in perfect agreement.

1.3.3 Volume consistency

One annoying feature of the compatible staggered Lagrangian scheme is the existence of two different cell volumes as pointed out in [12]. Although this difference was involved in a measure of consistency for numerical simulations, the conclusion of this work left a sour taste and a feeling of incompleteness. With M. Shashkov and B. Wendroff we further investigated this point and this work has led to paper [16] entitled *Volume consistency in a staggered grid Lagrangian hydrodynamics scheme*.
Let us remind that the classic compatible staggered Lagrangian compressible hydrodynamics scheme involves a choice of how internal energy is advanced in time. The options depend on two ways of defining cell volumes : an indirect one, that guarantees total energy conservation, and a direct one that computes the volume from its definition as a function of the cell vertices. It is shown that the motion of the vertices can be defined so that the two volume definitions are identical. In this note we construct a modification of the scheme such that we remove the ambiguity in the definition of cell volume that results from requiring both total energy conservation and the modeling of the internal energy advance from the differential equation $\frac{d\varepsilon}{dt} + p\frac{d(1/\rho)}{dt} = 0$. This is brought about by appropriately relating the motion of cell vertices to the cell volume change. This approach is purely algebraic.

More precisely we showed that the two volume definitions are equivalent if and only if certain matrices are equal. We explicitly gave the form of these matrices. The classical explicit discretizations of the scheme are such that these matrices are not equal. We have therefore developed a modification of the scheme involving an "inner consistency iterative procedure" [8] for the matrices to match at convergence. This procedure uniquely implies the discretizations of momentum equation and internal energy equation in order to get volume consistency and total energy conservation. Then we have tested this modification in 2D axisymetric geometry $(r-z)$ on the Coggeshall adiabatic compression problem [57]. These results compare the inconsistent Control Volume (iCV) scheme, say the classical scheme and the proposed consistent Control Volume (CV) scheme. They are reproduced in Fig 1.7. We observed that in addition to energy conservation the cell entropies are almost exactly conserved.

1.4 Special additions

In this section I present some other topics related to the compatible staggered Lagrangian scheme to which my colleagues and myself have studied. More precisely in this section we treat the following subjects :

In the endless story on human being struggling with artificial viscosity, E.J. Caramana and I made a contribution with article [11] entitled *"Curl-q" : A vorticity damping artificial viscosity for essentially irrotational Lagrangian hydrodynamics calculations*. The goal of this work is to supplement the so-called edge artificial viscosity, see section 1.2.3, with a viscous term that is designed to eliminate spurious vorticity.

The purpose of the work made by E.J. Caramana and I, published in [10] under the title *The Force/Work Differencing of Exceptional Points in the Discrete, Compatible Formulation of Lagrangian Hydrodynamics* is to complete the compatible formulation of Lagrangian hydrodynamics by addressing the remaining finite-volume discretization questions that arise when treating grid

8. From an implementation point of view this consists in an inner loop within the usual predictor-corrector loop.

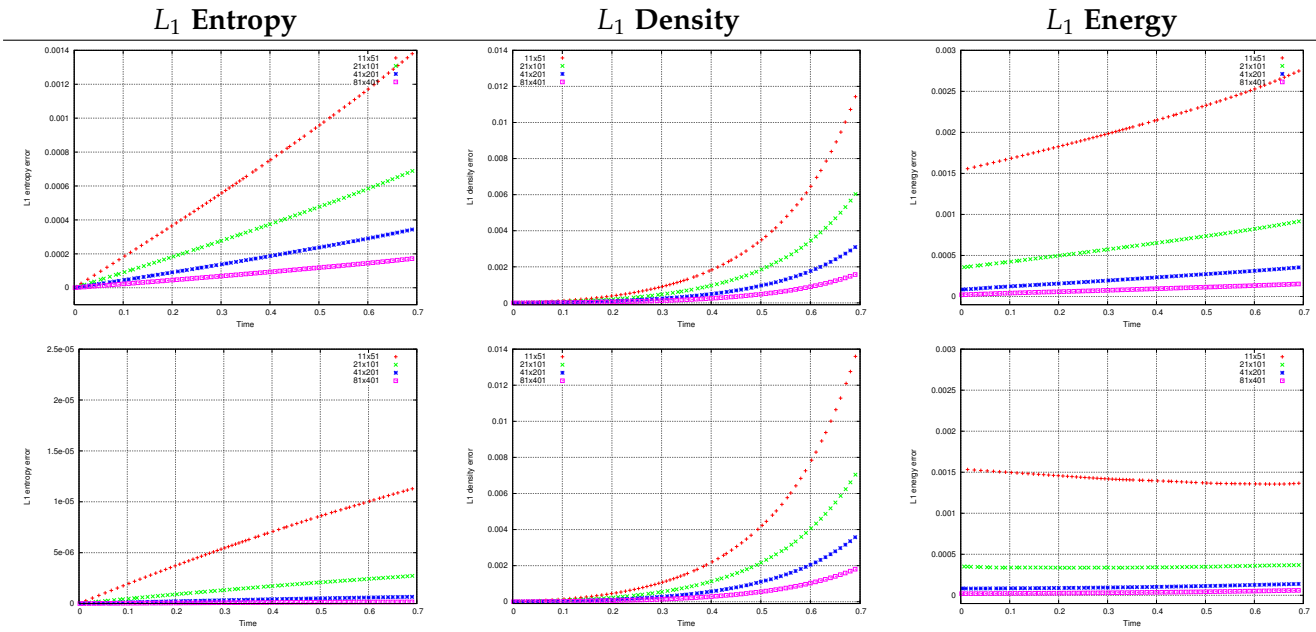


FIGURE 1.7 – *Numerical results from paper [16]. Coggeshall problem on a quarter of a disk in $r-z$ geometry — Entropy (left panels), Density (middle panels), Energy (right panels) — L_1 errors as functions of time for successively refined meshes 11×51 up to 81×401 for a CFL condition $1/4$. — Top line : Inconsistent control Volume (iCV) scheme — Bottom line : Consistent control Volume (CV) scheme. The scales for the entropy error plots are different as the consistent control volume scheme exhibits a quasi-exact entropy conservation.*

points that must be internally enslaved within the grid to prevent timestep collapse. In other words what we have called "exceptional points". The work in article [10] focuses on developing a treatment of exceptional points such that collateral damages brought by the existence of exceptional points are literaly (or at least virtualy) absent.

With M. Kucharik, R. Liska and L. Bednarik at CTU in Prague (Czech Republic) we have faced a situation where a slide-line treatment for the compatible staggered Lagrangian scheme was needed. Starting with the paper of E.J. Caramana [80] we have published in [25] a work based on two enhancements — interpolated interaction instead of a simple one-to-one point interaction described in the previous article, and a numerical surface tension model improving the stability of the interface. Both improvements stabilize the slide line and lead to more realistic results, as shown on selected numerical examples such as pure sliding, some sanity checks such as the Saltzman piston, two sliding rings, and some more realistic simulations like the explosion with sliding and the bullet in a channel.

Trying to unite cell-centered and staggered Lagrangian schemes into a common framework in order to extrude the similarity of these supposedly different approaches is the next considered topic. This subject has been brought to light by P.-H. Maire all alone. At the very end of his thoughte he has been further joined by P. Váchal and me for several publications in 2D [23] *Staggered Lagrangian Discretization Based on Cell-Centered Riemann Solver and Associated Hydrodynamics Scheme*, and its counterpart in 3D [24] : *3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver based artificial viscosity*. The bridge between the cell-centered and staggered approaches has led us to re-derive the staggered compatible scheme, and most of all re-define the concept of subcell force invoking Galilean invariance and thermodynamical consistency. More precisely we have drawn the basics to design a new form of artificial

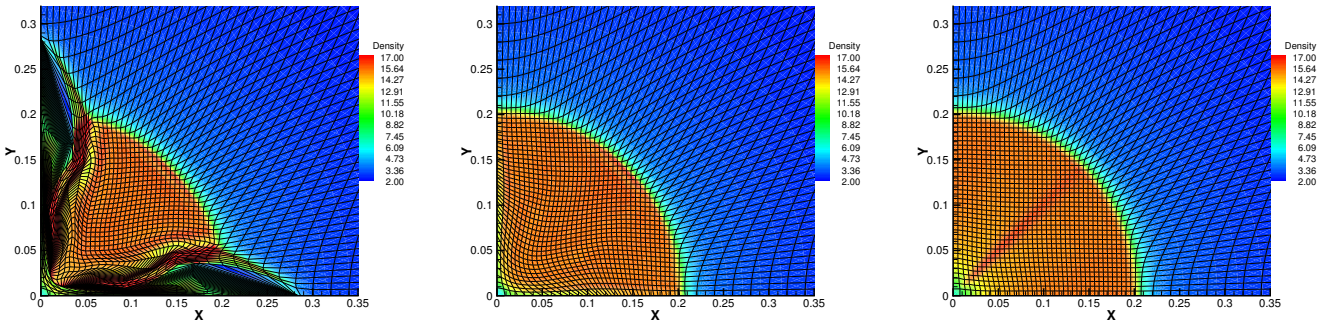


Figure 1.8 – *Numerical results from paper [11]. Noh problem, 2D Cartesian geometry — Mesh and density at time* $t = 0.6$*, the exact radius of the shock wave is* $r = 0.2$ *— Left : Edge viscosity without curl-q,* 1422 *time steps, — Middle : Edge viscosity with curl damping,* 606 *time steps, — Right : Tensor viscosity,* 423 *time steps.*

viscosity subcell force driven by a subcell-based positive definite tensor, M_{cp}, which is the true essence of the numerical scheme.

1.4.1 Vorticity damping artificial viscosity

The bane of Lagrangian hydrodynamics calculations in multi-dimensions is the appearance of vorticity that causes tangling of the mesh and consequent run termination. This vorticity may be numerical or physical in origin, and is in addition to the spurious "hourglass" modes associated with quadrilateral or hexahedral zones that in pure form have both zero curl and divergence associated with their velocity field. The purpose of this paper is to introduce a form of vorticity damping, based on a edge-centered artificial viscosity [57], see also section 1.2.3, that extends the runtime and range of calculations over which a pure Lagrangian code can compute.

The origin of this work was the superiority of the tensor artificial viscosity [53] to damp spurious vorticity especially for the Noh problem on quadrangular grid for which the edge-centered artificial viscosity [57] is creating jets along axes leading to a serious lack of robustness and symmetry. The analysis of the tensor artificial viscosity was not trivial. The hope of recasting it into an edge-centered artificial viscosity that would ease the comparison and as such enlight the extra-terms was not a success. Consequently E.J. Caramana and I adopt a different strategy and developed an extra-term to the edge-centered artificial viscosity denoted as the "curl-q", because it is a function of the curl of the velocity field in a zone. Notice that this new "curl-q" does not resolve shock waves and is always to be utilized with an artificial viscosity that performs this task. This curl-q force is formulated as an analogy to the edge-centered artificial viscosity. The development and justification of this curl-q force is provided in this work. Moreover numerical results are given both in 2D and 3D showing the effectiveness of the approach. In particular, results are contrasted between this new term and the tensor artificial viscosity [53]. It is shown that these two forms give quite similar results in 2D. As instance we reproduce in Fig. 1.8 the results for the Noh problem on 2D Cartesian geometry with initially square zones (50×50 cells). Moreover in Fig. 1.9 we reproduce the 3D Noh results obtained with and without the curl-q vorticity damping term. The main advantage of this "curl-q" is its ability to damp curl like motion and this is also its main drawback when fluid instabilities, that is to say rotational flows, are expected to occur.

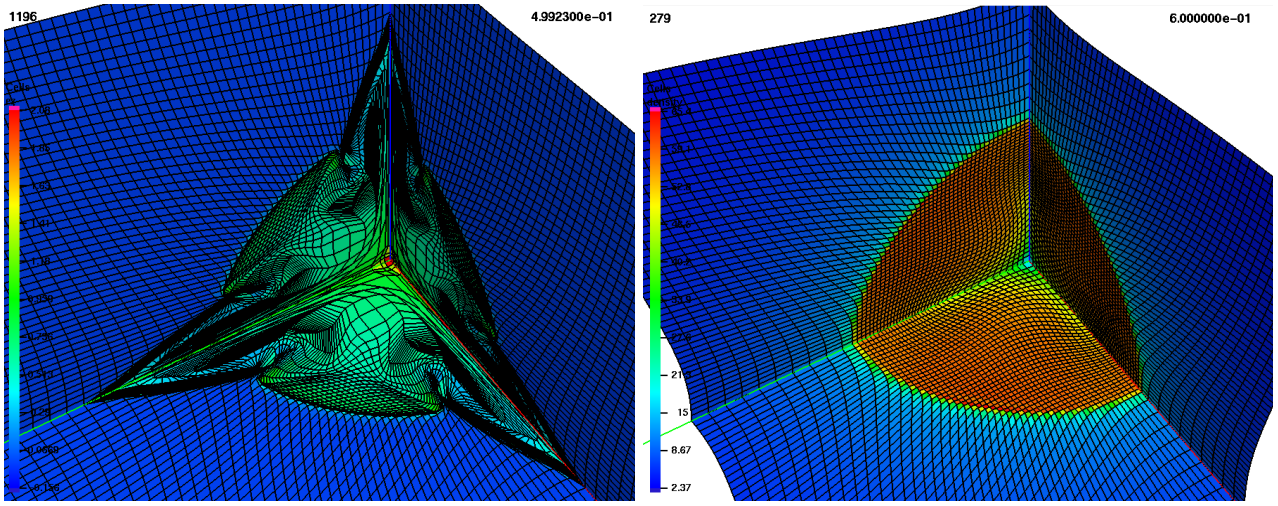
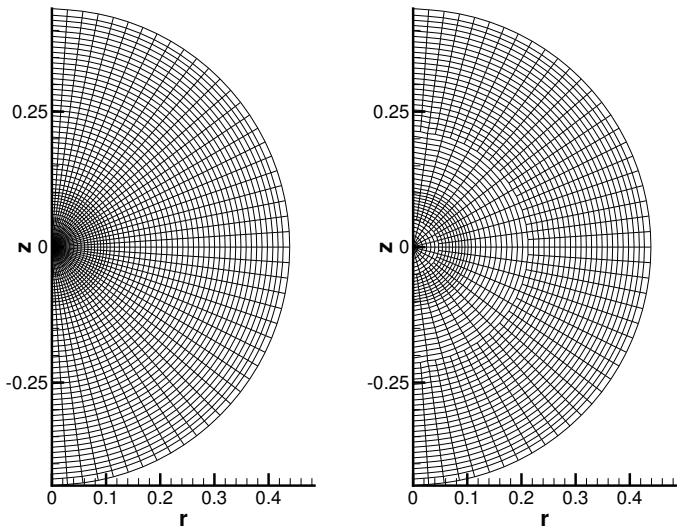


FIGURE 1.9 – *Numerical results from paper [11]. Spherical Noh problem, 3D Cartesian geometry, one octant,* 50^3 *cubic initial grid — Edge viscosity — Corner is the origin. Left : without curl-q damping, fails at time* $t = 0.5$ *due to mesh tangling at* 1196 *time steps. Specific internal energy is shown for contrast, density is unphysical due to the severe mesh distortion. — Right : with curl-q damping at time* $t = 0.6$, 279 *time steps. Variable plotted is density, which matches the exact solution (shock wave at radius* $r = 0.2$, *density of* 64, *red color).*

1.4.2 Dealing with exceptional points

The purpose of the work made by E.J. Caramana and I and published in [10] under the title *The Force/Work Differencing of Exceptional Points in the Discrete, Compatible Formulation of Lagrangian Hydrodynamics* is to complete the compatible formulation of Lagrangian hydrodynamics by addressing the questions that arise when treating grid points that must be internally enslaved within the grid to prevent timestep collapse, in other words "exceptional points".



Left : Example of a polar mesh with 51×49 *cells. Right : Radial mesh with three layers of exceptional points.*

The figure to the left presents an example of a three layer of exceptional point polar mesh, one also calls termination lines the mesh lines stopping at an exceptional point. A zone with an exceptional point is also sometimes called a dendritic zone. The reason to use such a mesh is the potential gain in CPU time compared to a full radial mesh : The CFL number is often ruled by the smallest cell length which is located on the triangles at the origin. Roughly speaking with three layers of exceptional points one expects three times bigger CFL number, therefore a three times less expensive computation. It is this decrease in spatial grid stiffness without degradation in solution quality that the procedure proposed in this work is intended to produce.

Indeed dendritic zones could be treated as pentagons by the compatible staggered Lagrangian scheme but the fact that any exceptional point is dynamical often leads to lack of robustness and tendency to unpleasant fatal mesh tangling situations. The work in article [10] focuses on developing a treatment of exceptional points such that collateral damages brought by the existence of exceptional points are literaly (or at least virtualy) absent.

First we have shown that the types of grids that must be used to reduce numerical perturbations about exceptional points involve a restriction of the type of zone to be uniform across the grid. A pentagon must become a quadrilateral. It was also shown that the basic discretization about exceptional, or nondynamical, points could be largely handled by three basic rules that involve already computed subcell masses and forces so that the number of additional operations is small. These involved impedance matching of subcell volumes and masses by donation to nearby dynamical points to which the velocity of the nondynamical exceptional points is enslaved. An appropriate addition of the remaining subcell forces, after median mesh grid adjustment, and their subsequent division and donation to neighboring dynamical points is performed such that force equilibrium is achieved for uniform stress (a necessary sanity check). It was shown that for the internal energy equation to obey momentum conservation about nondynamical points, it is required that the definition of what region in space constitutes a "zone" be generalized. A zone becomes the smallest region in space for which its associated subcell forces sum to zero. Thus, primitive zones that contain common nondynamical points must be glued together in calculating the work from the internal energy equation. In this work the edge-centered artificial viscosity [57] or the tensor artificial viscosity [53] are considered and adapted to the presence of exceptional points if needed. Numerical results were shown to validate the procedures given, and to quantify the magnitude of the errors that necessarily occur with the introduction of terminated lines (and their associated nondynamical points). In our work a straight piston is used as a sanity check then the Saltzman problem is simulated with or without exceptional points. Then the spherical Noh problem in axisymmetric geometry is treated as to measure the symmetry error introduced by the presence of exceptional points. Finally the Guderley problem in axisymmetric geometry is run to assess the efficiency of the treatment. Moreover the Guderley problem with false center of convergence is further simulated to show the robustness of our approach when the grid is no more aligned with the flow.

As an illustration we reproduce in Fig. 1.10 the results on a straight piston for the original scheme and the scheme coupled with our donation technique. This sanity check shows how the technique is able to treat termination lines without spurious side effects whereas the original scheme presents an oscillation. In Fig. 1.11 the axisymmetric Guderley problem is simulated (compression of an homogeneous gas initialy at rest due to external velocity boundary condition) on a polar $\Delta r \times \Delta\theta = 51 \times 49$ mesh and on the same effective resolution mesh but with three termination lines. This problem is run to a time $t = 0.8$ at which point the shock wave has reached the true center of convergence, is propagating outward, and has recrossed the inner layer of terminated points. The results are very comparable in quality but the number of timesteps needed to run without and with terminations is about 3000 and 1000.

1.4.3 Slide-lines

When developing a simulation code based on a Lagrangian scheme at one point we can have to face situations for which slide-lines are required. Many hydrodynamical problems involve shear flows along material interfaces. If the materials move along each other but are tied to a single Lagrangian computational mesh without any sliding treatment, severe mesh distortions appear which can eventually cause the failure of the simulation.

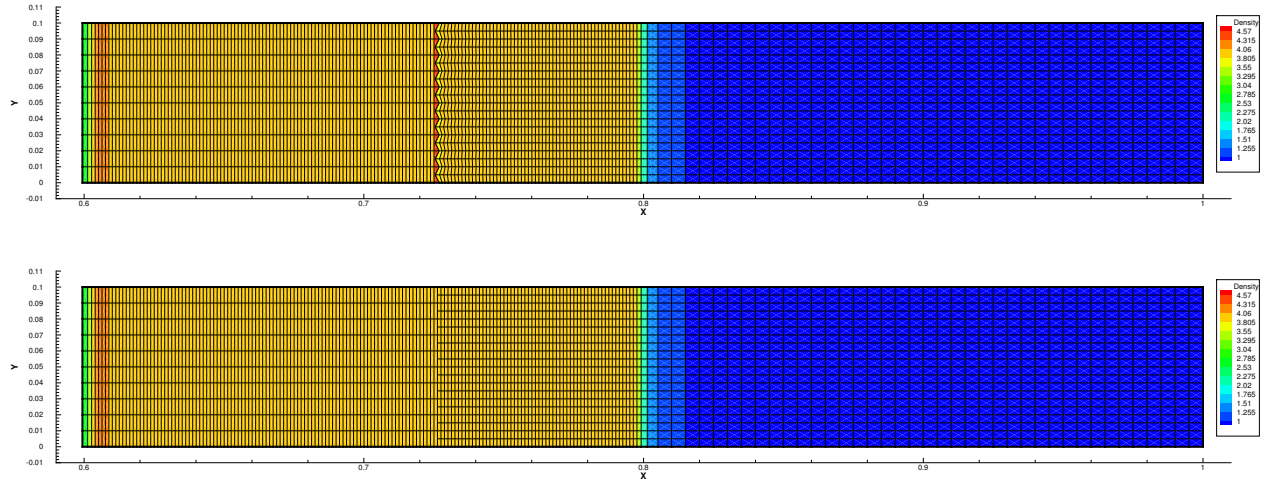


FIGURE 1.10 – *Numerical results from paper [10]. Straight piston with termination lines — (a) Mesh and density at* $t = 0.6$ *original scheme — (b) Mesh and density at* $t = 0.6$ *with the technique proposed.*

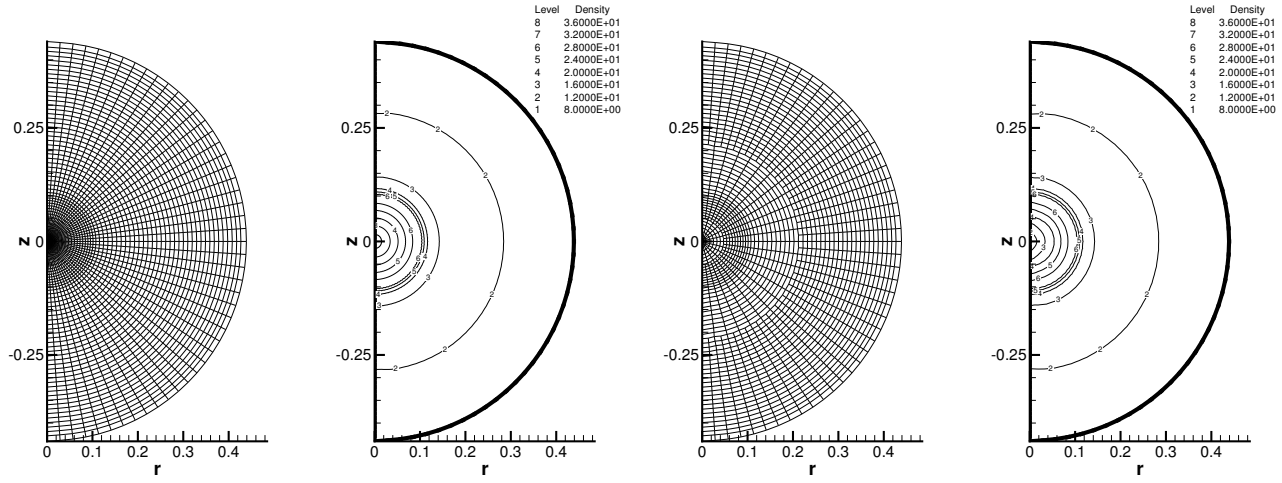


FIGURE 1.11 – *Numerical results from paper [10]. Guderley problem at* $t = 0.8$ *— Left : Final mesh and density isolines with no termination layers (*$\sim$ 3000 *time steps). — Right : Two termination layer final mesh and density isolines (*$\sim$ 1000 *time steps).*

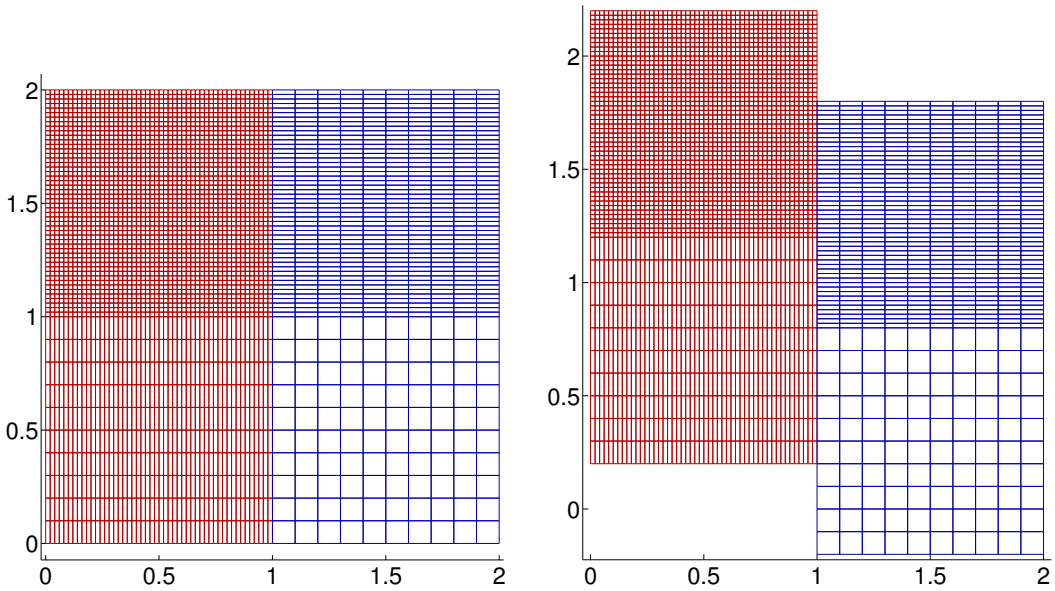


Figure 1.12 – *Numerical results from paper [80]. Initial and final* 40×50 *and* 10×50 *meshes of the pure sliding sanity check. Two vertical blocks of fluid (Left and Right) meshed with non-uniform grids are sliding with velocity* $\boldsymbol{U}_L = (0, +v)$ *and* $\boldsymbol{U}_R = (0, -v)$, $v = 1$ *in our test. The vertical slide line is initiated at* $x = 1$ *and must remain vertical during the sliding.*

With M. Kucharik, R. Liska and L. Bednarik at CTU in Prague (Czech Republic) we have implemented a slide-line treatment into the PALE code (Prague ALE) based on the compatible staggered Lagrangian scheme on quadrilateral meshes. Starting with the paper of E.J. Caramana [80] we have published paper [25] entitled *Enhancement of Lagrangian slide lines as a combined force and velocity boundary condition*.

In this work we first review the 2D approach described by E.J. Caramana in [80] and suggest two enhancements - interpolated interaction instead of a simple one-to-one point interaction described in the previous article, and a numerical surface tension model improving the stability of the interface. Both improvements stabilize the slide line and lead to more realistic results, as shown on selected numerical examples such as pure sliding, Saltzman piston, sliding rings, explosion with sliding and a bullet in a channel. In Fig. 1.12 we reproduce the sanity check of two columns of gas sliding on each other. The meshes are of different size. Nonetheless the sliding is perfectly reproduced. The same pure sliding test is further run with two annulii. In Fig. 1.13 one reproduces the results obtained by the original approach of E.J. Caramana *vs* our improved technique.

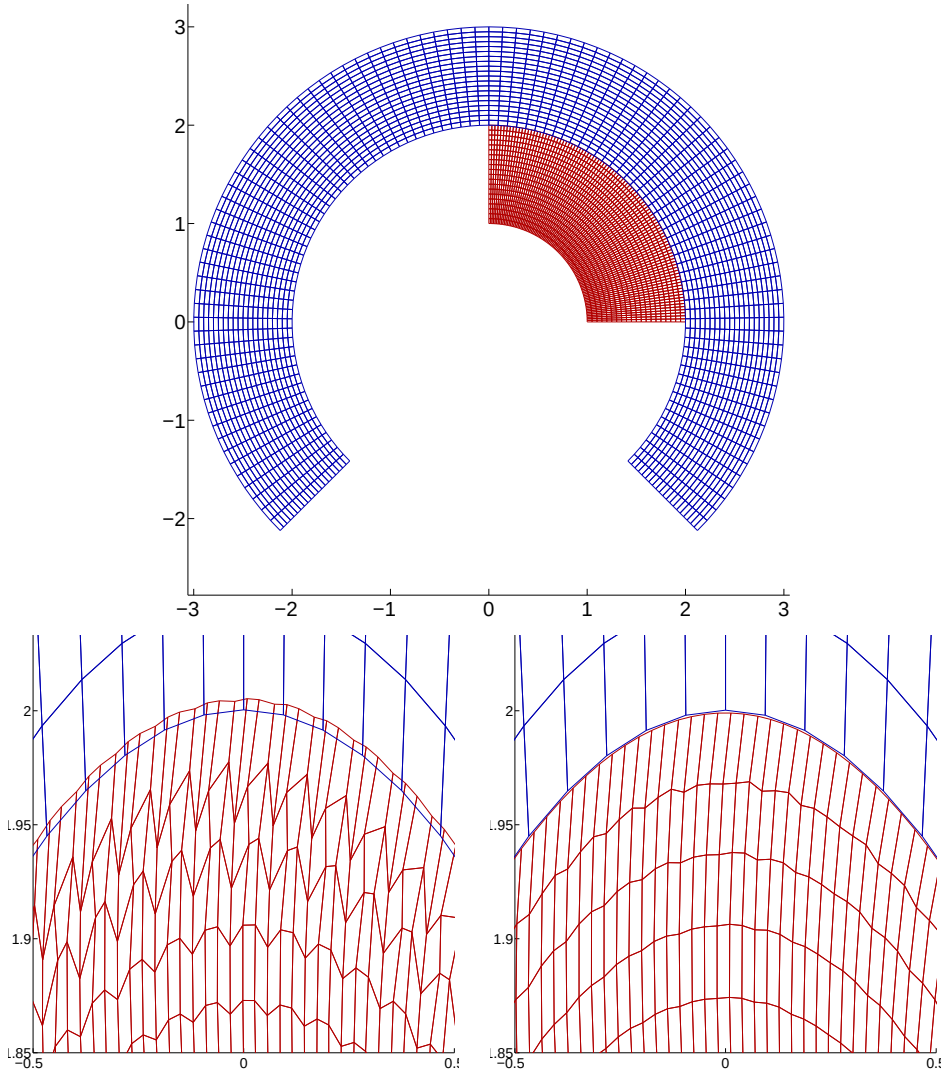


FIGURE 1.13 – *Numerical results from paper [80]. Top : Initial meshes for sliding rings problem separated by a circular slide line of radius $R = 2$. Bottom : Zoom to the interesting part along the slide line (aspect ratio not preserved). Comparison of computational meshes for the original approach (left) and our improved method (right) proposed in [80].*

1.5 Uniting cell-centered and staggered Lagrangian schemes

In this section one presents the work done mainly by P.-H. Maire, P. Váchal and I on uniting cell-centered and staggered Lagrangian schemes into a common framework. The goal is to extrude the similarity of these supposedly different approaches. I need however to give credit to P.-H. Maire for his breakthrough idea on bridging the two approaches and his ability to divert fruitful tools from his work on cell-centered schemes [48, 49, 102, 50, 103, 104, 105, 51, 42]. This has led to still on-going research and several publications in 2D [23], and 3D [24], proceedings [21, 22] from international conferences. It is worth noticing that a very similar approach has been developed independently by A. Burbeau-Augoula in [106] almost at the same time. In this section I review the 2D and 3D publications on this subject.

2D cell-centered Riemann solver based artificial viscosity. In article [23] *Staggered Lagrangian Discretization Based on Cell-Centered Riemann Solver and Associated Hydrodynamics Scheme*, P.-H. Maire, P. Váchal and I have drawn the basics to design a new form of artificial viscosity for the compatible staggered Lagrangian scheme.
More precisely this work suggests a general formalism to derive staggered discretizations for Lagrangian hydrodynamics on general unstructured meshes in two dimensions. This unified formalism uses the concept of subcell mass and force from the compatible staggered Lagrangian scheme community and a Riemann solver based artificial viscosity from cell-centered Lagrangian scheme community, see [104, 105] for details.
This artificial viscosity form is formulated invoking Galilean invariance and thermodynamic consistency. Moreover the satisfaction of entropy inequality is ensured by using a subcell-based positive definite tensor, M_{cp} for cell c and point p. This tensor is the core of the scheme as it uniquely defines the artificial viscosity and the nature of the scheme *per se*. Let us remind the final form of the subcell force (1.55) for the compatible staggered Lagrangian scheme

$$\boldsymbol{F}_{cp} = \boldsymbol{F}_{cp}^{\text{press}} + \boldsymbol{F}_{cp}^{q} + \boldsymbol{F}_{cp}^{\Delta P},$$

which is constituted of the pressure force $\boldsymbol{F}_{cp}^{\text{press}}$ (1.35), the artificial viscous force $\boldsymbol{F}_{cp}^{q}$ (1.39, 1.40) and the anti-hourglass force $\boldsymbol{F}_{cp}^{\Delta P}$ (1.51). In our work a sufficient condition to obtain the satisfaction of the second law of thermodynamics is to set the subcell force as

$$\boldsymbol{F}_{cp} = \boldsymbol{F}_{cp}^{\text{press}} + \mathsf{M}_{cp}(\boldsymbol{U}_p - \boldsymbol{U}_c), \tag{1.73}$$

where M_{cp} is a 2×2 subcell-based matrix. A seemingly new degree of freedom, the cell-centered velocity $\boldsymbol{U}_c$, is in reality deduced from the point velocities and the subcell-based tensor. Indeed substituting the subcell force expression, $\boldsymbol{F}_{cp} = -L_{cp}P_c\boldsymbol{N}_{cp} + \mathsf{M}_{cp}(\boldsymbol{U}_p - \boldsymbol{U}_c)$, into the Galilean invariance condition, $\sum_{p\in\mathcal{P}(c)} \boldsymbol{F}_{cp} = \boldsymbol{0}$, leads to the following system satisfied by the cell-centered velocity $\boldsymbol{U}_c$

$$\mathsf{M}_c\boldsymbol{U}_c = \sum_{p\in\mathcal{P}(c)} \mathsf{M}_{cp}\boldsymbol{U}_p, \tag{1.74}$$

where $\mathsf{M}_c = \sum_{p\in\mathcal{P}(c)} \mathsf{M}_{cp}$ is a symmetric positive definite matrix. Once the definition of the subcell matrix M_{cp} is known, one can solve the previous system to get a unique expression of the cell-centered velocity. By analogy with the node-centered approximate Riemann solver introduced in the context of cell-centered Lagrangian discretization [50], we present one cell-centered approximate Riemann solver. This solver allows to determine one particular form of the subcell matrix M_{cp}. To

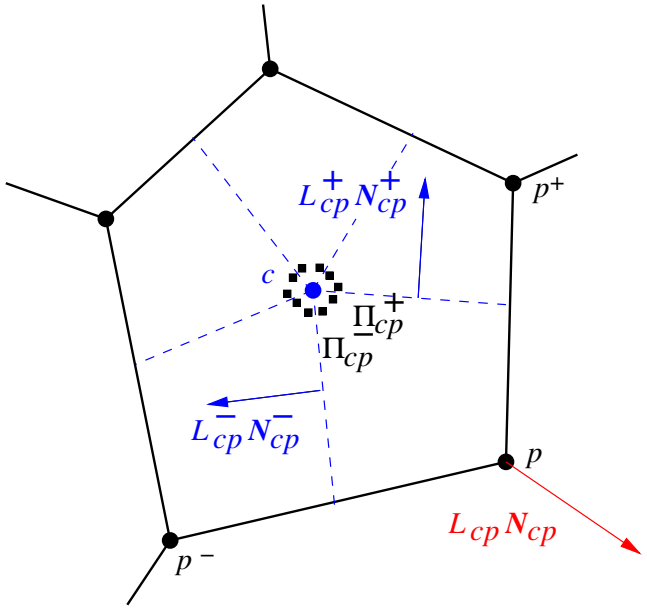


FIGURE 1.14 – *Notation used in the cell-centered Riemann solver. Two pressures per subcell (■) are introduced at the cell center : Π_{cp}^{+}, Π_{cp}^{-}. They are related to the outward normal vectors $L_{cp}^{+}\boldsymbol{N}_{cp}^{+}$, $L_{cp}^{-}\boldsymbol{N}_{cp}^{-}$. In total, $2 \mid \mathcal{P}(c) \mid$ pressures are introduced within cell Ω_c.*

this end, let us introduce two pressures at the cell center per subcell denoted by Π_{cp}^{-}, Π_{cp}^{+}. These pressures are related to the normals $\boldsymbol{N}_{cp}^{+}$, $\boldsymbol{N}_{cp}^{-}$ which are the unit outward normals to the subcell boundaries inside the cell, refer to Fig. 1.14. The subcell force is then defined as

$$\boldsymbol{F}_{cp} = L_{cp}^{-}\Pi_{cp}^{-}\boldsymbol{N}_{cp}^{-} + L_{cp}^{+}\Pi_{cp}^{+}\boldsymbol{N}_{cp}^{+}. \tag{1.75}$$

The cell-centered pressures are obtained by means of the half-Riemann problems

$$P_c - \Pi_{cp}^{-} = Z_{cp}^{-}\left(\boldsymbol{U}_c - \boldsymbol{U}_p\right)\cdot\boldsymbol{N}_{cp}^{-}, \tag{1.76}$$
$$P_c - \Pi_{cp}^{+} = Z_{cp}^{+}\left(\boldsymbol{U}_c - \boldsymbol{U}_p\right)\cdot\boldsymbol{N}_{cp}^{+}, \tag{1.77}$$

where Z_{cp}^{-}, Z_{cp}^{+} denote the swept mass fluxes, and $\boldsymbol{U}_c$ is the cell-centered velocity which remains to be computed. The swept mass fluxes, Z_{cp}^{-}, Z_{cp}^{+}, are defined following Dukowicz [88] as

$$Z_{cp}^{-} = \rho_c\left[\sigma_c + c_Q\Gamma_c \mid (\boldsymbol{U}_c - \boldsymbol{U}_p)\cdot\boldsymbol{N}_{cp}^{-} \mid\right], \qquad Z_{cp}^{+} = \rho_c\left[\sigma_c + c_Q\Gamma_c \mid (\boldsymbol{U}_c - \boldsymbol{U}_p)\cdot\boldsymbol{N}_{cp}^{+} \mid\right]. \tag{1.78}$$

Here, σ_c is the isentropic sound speed, c_Q a user-defined parameter (usually set to 1 in our simulations) and Γ_c a material dependent coefficient, which for a γ gas law is defined by

$$\Gamma_c = \begin{cases} \frac{\gamma+1}{2} & \text{if } (\nabla\cdot\boldsymbol{U})_{cp} < 0, \\ 0 & \text{if } (\nabla\cdot\boldsymbol{U})_{cp} \geq 0, \end{cases} \tag{1.79}$$

where $(\nabla\cdot\boldsymbol{U})_{cp} = -\frac{1}{V_{cp}}L_{cp}\boldsymbol{N}_{cp}\cdot(\boldsymbol{U}_c - \boldsymbol{U}_p)$ is the subcell contribution to the velocity divergence. In case of rarefaction wave, we recover the acoustic approximation whereas in case of shock wave we get the well known two-shock approximation.

Using (1.76)-(1.77) the subcell force is rewritten

$$\boldsymbol{F}_{cp} = \left(L_{cp}^{-}\boldsymbol{N}_{cp}^{-} + L_{cp}^{+}\boldsymbol{N}_{cp}^{+}\right) P_c + \mathsf{M}_{cp}\left(\boldsymbol{U}_p - \boldsymbol{U}_c\right), \tag{1.80}$$

where

$$\mathsf{M}_{cp} = Z_{cp}^{-}L_{cp}^{-}(\boldsymbol{N}_{cp}^{-} \otimes \boldsymbol{N}_{cp}^{-}) + Z_{cp}^{+}L_{cp}^{+}(\boldsymbol{N}_{cp}^{+} \otimes \boldsymbol{N}_{cp}^{+}) \tag{1.81}$$

is a 2×2 symmetric positive definite matrix. Because $L_{cp}^{-}\boldsymbol{N}_{cp}^{-} + L_{cp}^{+}\boldsymbol{N}_{cp}^{+} = -L_{cp}\boldsymbol{N}_{cp}$ then the subcell force writes like (1.73) where the subcell matrix is given by (1.81). The generic form of the subcell force has been retrieved. Our expression of the subcell matrix is directly linked to the half-Riemann invariants (1.76)-(1.77). The cell-centered velocity $\boldsymbol{U}_c$ is obtained by solving the system $\mathsf{M}_c\boldsymbol{U}_c = \sum_{p\in\mathcal{P}(c)} \mathsf{M}_{cp}\boldsymbol{U}_p$, recalling that $\mathsf{M}_c = \sum_{p\in\mathcal{P}(c)} \mathsf{M}_{cp}$ and that M_{cp} is given by (1.81). M_c is symmetric positive definite which ensures its invertibility. Remark that this system is non-linear due to the dependency of the swept mass flux on the cell-centered velocity. This non-linear system can be solved using an iterative procedure such as fixed point or Newton algorithms. In practice, few iterations are needed to get convergence, in fact we only use two iterations. Once the cell-centered velocity is known, the subcell force is deduced from equation (1.73). The present cell-centered approximate Riemann solver can be viewed as a two-dimensional extension of the work initiated by Christensen in one-dimensional framework [87]. The viscous part of subcell force is an important potential link between staggered and cell-centered Lagrangian schemes. While some of the existing artificial viscosity implementations can be reformulated by means of the proposed symmetric positive definite tensor, others still seem to resist this simple interpretation. From this viewpoint there remains enough space for deeper investigation with the prospect of finding similarities and differences between the Godunov and Lagrange like methods.

An elegant way to incorporate the anti-hourglass-like forces (1.51) within the framework consists of incorporating subcell pressure effects by the substitution of P_{cp} into the half-Riemann problems. In other words, one replaces P_c in (1.76-1.77) by P_{cp} as follows

$$P_{cp} - \Pi_{cp}^{-} = Z_{cp}^{-}\left(\boldsymbol{U}_c - \boldsymbol{U}_p\right)\cdot\boldsymbol{N}_{cp}^{-}, \tag{1.82}$$

$$P_{cp} - \Pi_{cp}^{+} = Z_{cp}^{+}\left(\boldsymbol{U}_c - \boldsymbol{U}_p\right)\cdot\boldsymbol{N}_{cp}^{+}. \tag{1.83}$$

The swept mass fluxes are also modified using the subcell density ρ_{cp} and sound speed σ_{cp} as

$$Z_{cp}^{\pm} = \rho_{cp}\left[\sigma_{cp} + c_Q\Gamma_c \mid \left(\boldsymbol{U}_c - \boldsymbol{U}_p\right)\cdot\boldsymbol{N}_{cp}^{\pm} \mid\right].$$

The corresponding subcell force is modified accordingly

$$\boldsymbol{F}_{cp} = -L_{cp}P_{cp}\boldsymbol{N}_{cp} + \mathsf{M}_{cp}\left(\boldsymbol{U}_p - \boldsymbol{U}_c\right). \tag{1.84}$$

Then, the system solving the cell-centered velocity rewrites as

$$\boldsymbol{U}_c = \mathsf{M}_c^{-1}\sum_{p\in\mathcal{P}(c)}\left(\mathsf{M}_{cp}\boldsymbol{U}_p - L_{cp}P_{cp}\boldsymbol{N}_{cp}\right). \tag{1.85}$$

Let us provide an interpretation of the two terms that determine the cell-centered velocity. The first term at the right-hand side is a weighted interpolation of nodal velocities at cell center, whereas the second corresponds to a discretization of the pressure gradient at cell center. This interpretation is obtained by computing the pressure gradient integral over the cell as

$$(\boldsymbol{\nabla}P)_c = \frac{1}{V_c}\int_{\partial\Omega_c} P\boldsymbol{N}\,\mathrm{d}S = \frac{1}{V_c}\sum_{p\in\mathcal{P}(c)}\int_{\partial\Omega_{cp}\cap\partial\Omega_c} P\boldsymbol{N}\,\mathrm{d}S = \frac{1}{V_c}\sum_{p\in\mathcal{P}(c)} L_{cp}P_{cp}\boldsymbol{N}_{cp}. \tag{1.86}$$

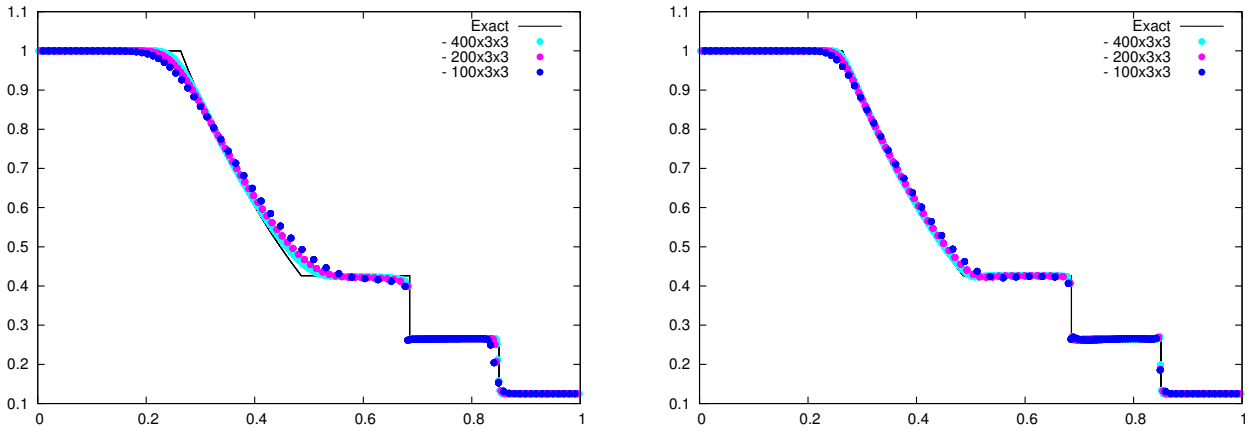


FIGURE 1.15 – *Numerical results from paper [23]. Sod problem at* $t_{final} = 0.2$ *for* $100, 200, 400$ *cells in x direction — Cell-centered density for the generic scheme and the scheme using piecewise linear velocity.*

Then the cell-centered velocity reads

$$\boldsymbol{U}_c = \sum_{p \in \mathcal{P}(c)} \mathsf{M}_c^{-1} \mathsf{M}_{cp} \boldsymbol{U}_p - V_c \mathsf{M}_c^{-1} \left(\boldsymbol{\nabla} P\right)_c . \tag{1.87}$$

This formula degenerates to the previous formula (1.74) in case of uniform subcell pressure over the cell. The extra pressure gradient term induced by the subcell pressures acts as a supplementary viscous term that is usually present in approximate Riemann solver.

An extension to higher order of accuracy in space using piecewise linear reconstruction of velocity field is also developed. The extension in time is obtained with the classical predictor-corrector scheme. One noticable new feature is the vector limitation procedure which is frame independent and thus preserves desirable properties like rotational symmetry. Performance of the new method is demonstrated on a set of classical and demanding numerical tests, using various structured and unstructured computational meshes, Sod, Sedov, Noh, and Saltzman problems and the linear phase of a Rychtmyer-Meshkov instability.

In Fig. 1.15 one reproduces the results obtained by the generic scheme and its extension using piecewise linear velocity on the Sod problem run with the 2D code. These clearly show the improvement gained by the later.

3D cell-centered Riemann solver based artificial viscosity. The promissing approach developed in [23], that is to say the cell-centered Riemann solver based artificial viscosity in 2D, has been further extended by P. Váchal, P.-H. Maire and I in three dimensions. This article appropriately entitled *3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver based artificial viscosity* [24] is depicted in this subsection.

The framework designed in 2D in [23] is extended to 3D. As the original framework was designed for unstructured 2D mesh, its extension to 3D is almost trivial. Besides the natural complication to face when developing a 3D code, we had to extend the notion of reconstruction and limitation of vector field to 3D. In fact the frame invariant limitation demands the definition of two directions in the plane perpendicular to the point velocity which is considered. Several tests have shown that this choice has a significant effect on robustness and/or accuracy. The main focus of this paper was to obtain the more complete and comprehensive picture of the efficiency of the 3D method

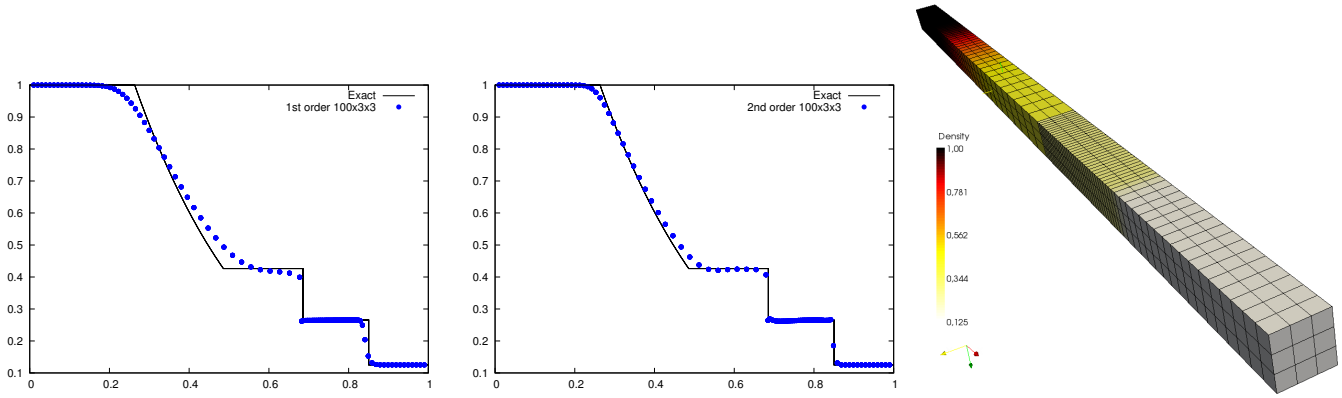


Figure 1.16 – *Numerical results from paper [24]. Sod problem at $t_{final} = 0.2$ for 100 cells in x direction and 3 in y and z directions — Left : generic scheme — Middle : generic scheme using piecewise linear velocity — Right : 3D view. Cell-centered density as a function of x for all cells and 3D view.*

implemented into a simulation code *via* test cases. As for any 3D extension of an existing method great care has to be paid to the implementation details. We then ran sanity checks with the 3D code to retrieve 1D Sod results as in Fig. 1.16. In Fig.1.17 we also show that 2D Sedov results are also retrieved by the 3D code. One has run several 3D problems : Sedov, Noh, Saltzman and Rayleigh-Taylor on hexaedric and polar meshes. The generic compatible staggered Lagrangian scheme and its extension using piecewise linear velocity using a cell-centered Riemann solver based artificial viscosity have been compared on the 3D Sedov problem with a 3D implementation of the compatible staggered Lagrangian scheme using the edge based artificial viscosity popularized in [57]. We have shown that the new approach is able to reproduce almost perfectly spherical symmetry whereas the edge based artificial viscosity presents some spurious mesh instability, see Fig. 1.18

$\sim$

This chapter was devoted to the description of a family of discrete compatible Lagrangian schemes and some of related investigations. As previously mentioned our primary goal was to build an ALE simulation code for compressible hydrodynamical flow. As such this family of staggered Lagrangian scheme is only one of the building brick of such a code. Presumably this is the most important brick as the Lagrangian scheme may be considered as the "engine" of any ALE code. As a consequence some of our investigations were specifically dedicated to better understand such family of numerical schemes.

The next chapter presents some of our investigations related to the two other parts of an ALE code, namely the rezone and remap steps.

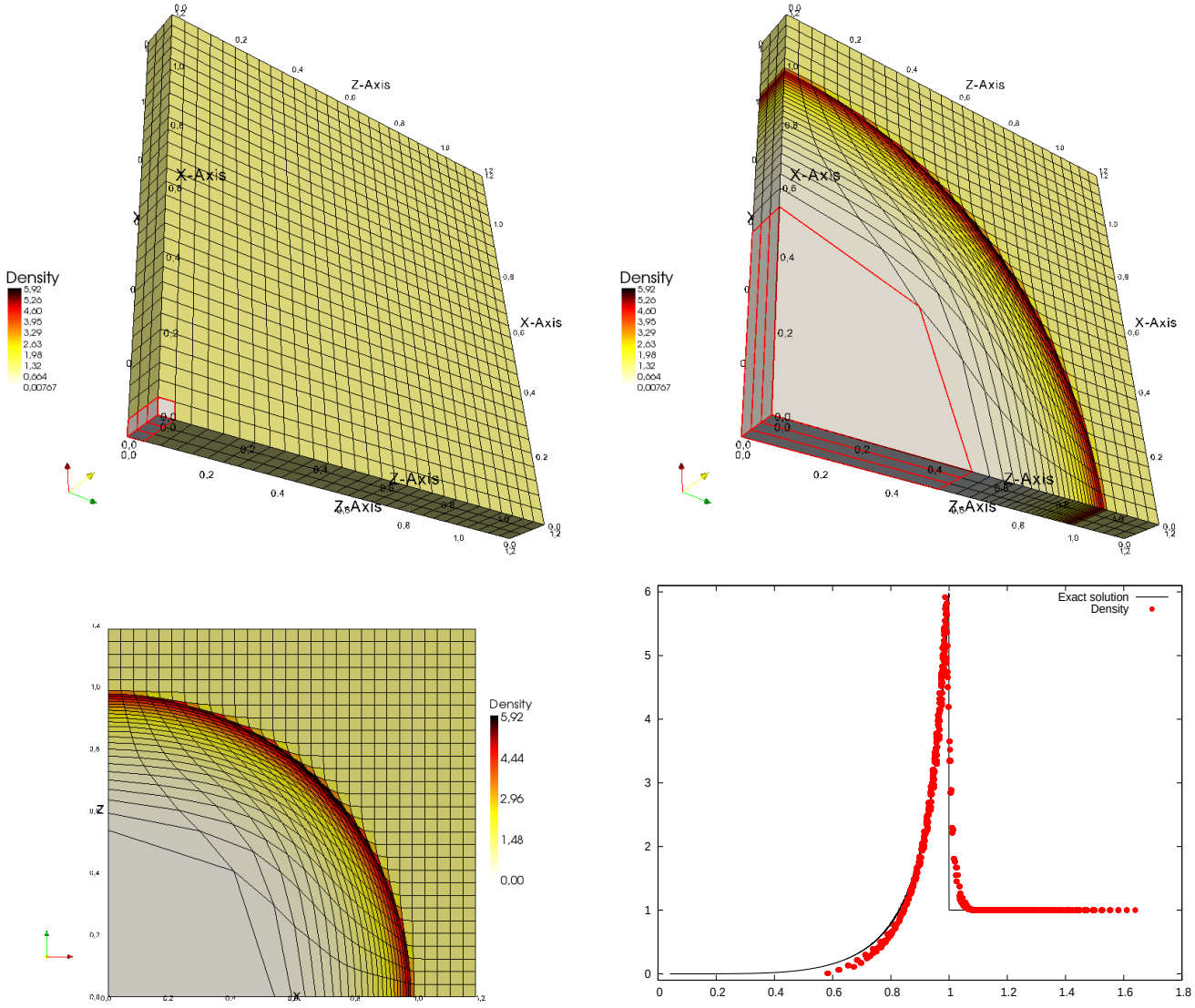


FIGURE 1.17 – *Numerical results from paper [24]. Sedov problem at* $t_{final} = 1$ *— Top : Density map and mesh on a* $30 \times 3 \times 30$ *Cartesian grid at initial and final time (the red highlighted cells are the ones with initial high energy) — Bottom-left : 2D x-z plane at* $y = 0$*, density map and mesh — Bottom-right : Density as a function of cell radius (all cells are displayed).*

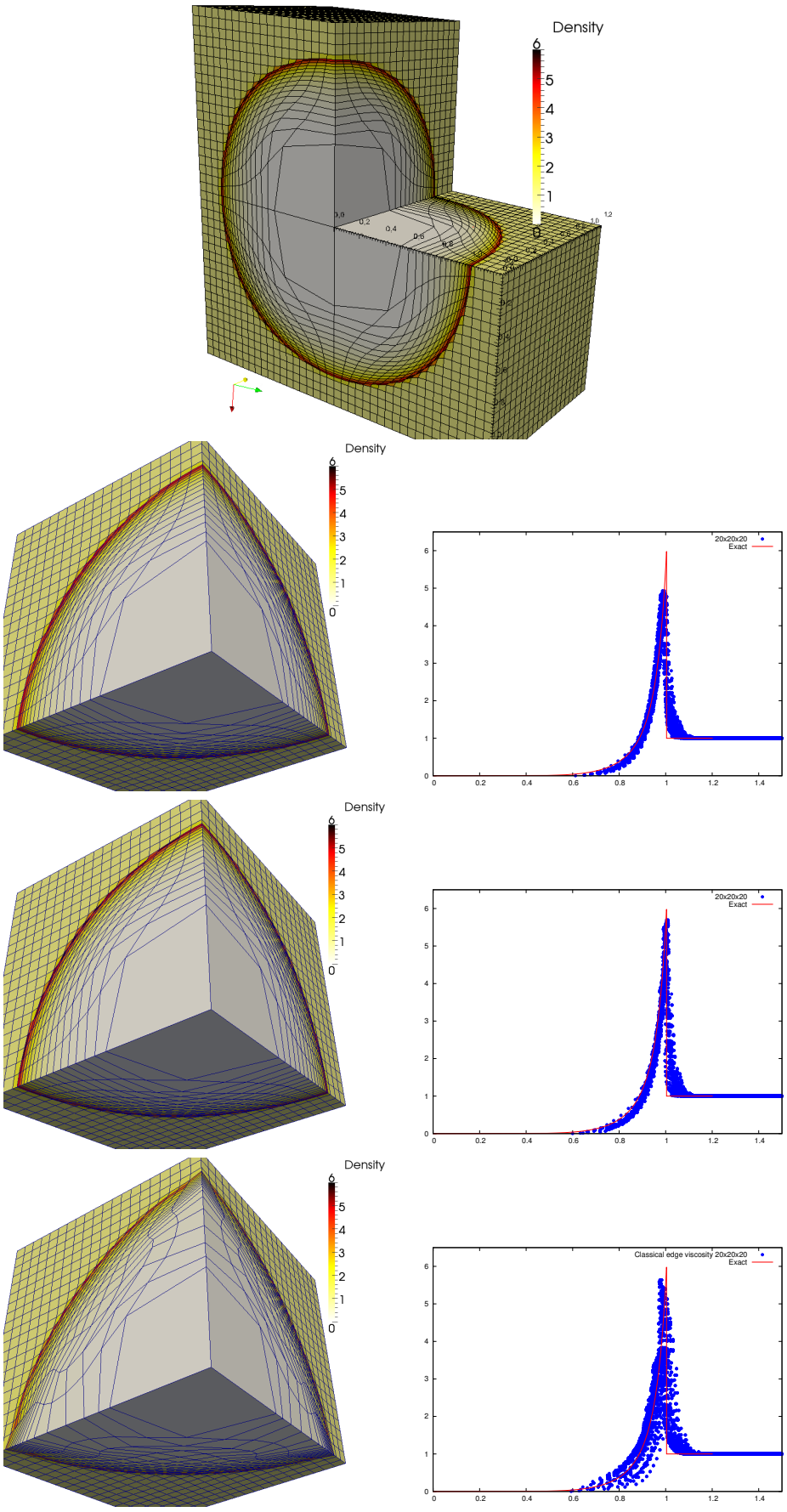


Figure 1.18 – *Numerical results from paper [24]. 3D Sedov problem on a* $20 \times 20 \times 20$ *hexahedral mesh at final time* $t = 1.0$ *— Density and mesh for the generic scheme at the very top line — Density as a function of radius for all cells in the domain for the remaining lines — Top-line : generic scheme — Middle-line : scheme with piecewise linear reconstruction of velocity — Bottom-line : Classical compatible staggered scheme with edge viscosity.*

Arbitrary-Lagrangian-Eulerian schemes

2

Contents

In numerical simulations of multidimensional fluid flow, the relationship between the motion of the computational grid and the motion of the fluid is an important issue. Two choices that are typically made representing either a Lagrangian framework, in which the mesh moves with the local fluid velocity, or an Eulerian framework, in which the fluid flows through a grid fixed in space. More generally, however, the motion of the grid can be chosen arbitrarily. The philosophy of the Arbitrary Lagrangian-Eulerian methodology (ALE ; cf. [107, 108, 52, 109, 110, 111, 112, 113]) is to exploit this degree of freedom to improve both the accuracy and the efficiency of the simulation. The main elements of many ALE algorithms are an explicit Lagrangian phase, a rezone phase in which a new grid is defined, and a remap phase which transfers the Lagrange solution onto the new grid [109]. Most ALE codes use a grid of fixed connectivity that, in two spatial dimensions, is formed by quadrilaterals or by a mix of quadrilaterals and triangles, the latter being considered as degenerate quadrilaterals. Ultimately, we are interested in the development of ALE methods for meshes whose connectivity may change during the calculation. In such methods, the total number of cells may change with time, as well as the number of edges bounding each cell, leading to the appearance of general polygonal cells.

As a first step toward this goal, in section 2.2, we present some of our contributions to the context of ALE methods on a mesh with fixed connectivity, but we allow the mesh to contain general polygonal cells. Extending the ALE methodology to this more general mesh is valuable in itself as it simplifies the setup process for computational domains with complex geometrical shapes and helps to avoid artificial mesh imprinting due to the restrictions of a purely quadrilateral mesh, [75, 114]. In this section we mainly focus on the remapping stage of the ALE methodology using the compatible staggered Lagrangian scheme presented in the previous chapter. This also includes the repair technique to conservatively corrected unphysical remapped variables. Then in section 2.3 we extend the fixed ALE methodology to a reconnection ALE, further called ReALE, to allow meshes

whose connectivity may change during the calculation. In this case only the rezone part of the ALE algorithm is revamped using the Voronoi tesselation machinery.
In section 2.4 we present some investigations we have performed to compare three ALE codes on representative and demanding test cases. Namely we have compared the following codes : CHIC at CELIA, University of Bordeaux, PALE code standing for 'Prague ALE', from Prague's team in Czech Republik, and ALE INC(ubator) at IMT, University of Toulouse. In section 2.5 we present some specifics techniques to deal with multi-material fluid flows. More specifically we focus on order-independent interface reconstruction technique to deal with more than three materials in mixed cells. Some numerical results are provided throughout the chapter to show the behaviors of the proposed techniques.

Let us begin by resetting the context and point why the research pursued was, to some degree, justified.

2.1 History and presentation

As previously stated the ALE methodology chosen for the code ALE INC(ubator) is built on three successive phases : Lagrangian, rezone and remap.

Lagrangian phase. We consider the discrete compatible staggered Lagrangian scheme presented and studied in the previous chapter, see chapter 1. Notice that this scheme employs staggered variables (cell-centered density and energy, vertex-based velocity) on general polygonal mesh. Moreover this scheme does need the subcell-based density and pressure which are further used to compute subpressure forces that fight back hourglass parasital grid motion, see the previous chapter for more details. These remarks are of great importance when the remap phase will be defined.

Rezone phase. The rezone phase consists in defining the new grid onto which the conservative variables are to be remaped. This new grid must be "better" than the previous Lagrangian grid. However providing a unambiguous definition of "better" in the previous sentence is a genuine problematic point.
It is more or less agreed [1] that a better mesh must be smoother than the Lagrangian mesh. In fact in the literature the rezone phase is often called the smoothing phase. One of the most cited name in this field is probably A. Winslow [115, 116] who has used the property of elliptic regularity to smooth grids. An elliptic operator is applied to the node positions leading to a better node equidistribution and to a relaxed Lagrangian mesh. The main drawback of this technique lays in its underlying metrics. Usually metrics are based on Euclidean distance and equidistribution of length, surface and angle between edges. Consequently this implies that according to this measure the "best" quadrangle is a square, the "best" triangle is an equilateral triangle, and, generaly the "best" polygon is the associated regular polygon. If the initial mesh is not an optimal mesh for this underlying metrics then the rezone strategy instantaneously starts to improve the mesh, even if no physical motion has occured yet. In other words in some part of the domain the initial and supposedly valid cells (because the user has provided a good enough mesh) are reshaped by the rezoning strategy before any physical process has started. This is not acceptable. Some cures have been developed such as zone/vertex triggers. Given constraints these triggers keep a cell/vertex Lagrangian, that is to say unmodified, by the rezoner. Quoting A.Barlow [117] "Constraints [...]

1. but it is not an issue without contention !

keep a node Lagrangian until some condition is reached e.g. element quality criterian or physical condition is reached in surrounding elements", see also [118] for an example of triggers. Such triggers are activated according to constraints and condition which are not obvious to derive. In other words it seems that there is nowadays no universal rezoner which satisfies the whole ALE community.

For another important contributor to the field of mesh generation, mesh metrics consult P. Knupp's articles, as instance [119, 120, 121, 122, 123, 124].

It is agreed that the new mesh must be at minima valid for the numerical Lagrangian method. In our case one requires that each cell remains a convex polygon. In the unlikely situation where the Lagrangian grid is tangled, before applying any rezoning technique, one applies the untangling procedure described in [125] and depicted in Fig. 2.1. This procedure computes the feasible set for a problematic node. More precisely for an invalid polygon and a bad node one determines the feasible set which is the valid space for the node to move in, that further leads to surrounding valid polygons. Although this feasible set may not always exist the authors of [125] have provided an extension of their method to deal with such a situation. Amazingly this feasible set approach can also be used as a mesh smoother [118]. In ALE INC(ubator) several smoothing techniques have been

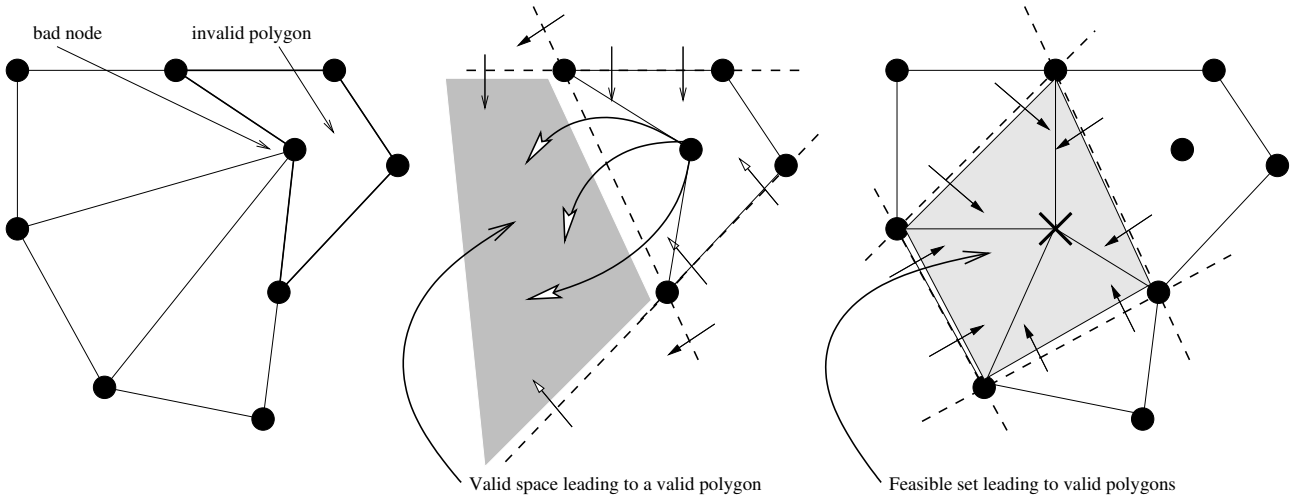


FIGURE 2.1 – *Description of the untangling technique from [125]. For an invalid polygon and a bad node one determines the feasible set which is the valid space for the node to move in that further leads to surrounding valid polygons. The middle panel shows the feasible set if one only considers to fix the invalid polygon only. The right panel shows the final feasible set leading to valid neighbor polygons.*

implemented. Winslow smoothing and the Reference Jacobian rezoning technique from [124], see the details in this paper. Our main contribution to this rezone phase is the mesh reconnection that is described in section 2.3.

One related issue with rezone and remap is the strategy to decide when a rezone and remap step is needed. This point has been little noticed and very few investigations have been carried out mainly because the rezoning step already demands several parameters to be fixed, adding two or three more parameters for the strategy does not weaken the entire process. Nevertheless we have tried to develop an automatic choice of rezone and remap strategy for our ALE code in [126]. The goal of this report is to gather several "measures" of solution quality to help the ALE code to develop its own capability to detect when and how often to rezone and remap. We skip this description but refer the reader to [126] for details.

Remap phase. The remapping phase is considered as conservative transfer (or advection) of the physical quantities from the Lagrangian mesh onto the rezoned and smoother mesh. To ease the phrasing we call the Lagrangian mesh "old" and the rezoned mesh "new", likewise for any entity defined on the Lagrangian or rezoned mesh.
An exhaustive list of contributors to this field is almost impossible as this should embrace the key words interpolation techniques, advection methods, (flux-corrected) transport methods and remapping *per se*. However some very much related works are to be found in [52, 127, 128, 129, 7, 130, 131, 132, 133, 134, 135]. In this introduction we only consider the so-called unsplit methods (as noticed by Benson [52] "despite the fact that they were never split in the first place"). By the way one urges the reader to refer to the review made by Benson [52] to have an overview of legacy remapping methods.
At first glance defining a remapping technique between an old and new mesh for a single conserved variable, say the mass, being the density multiplied by a surface, is fairly easy. First one defines a conservative representation of the old density on the old mesh ; a piecewise constant representation in a finite volume sense. Second the exact geometrical intersection between a new cell and the old mesh is computed, it consists of a set of polygons that pave the new cell without overlapping and gap. Third the new mass in the new cell is computed as the sum of all old masses present in the intersection polygons (the mass is computed as the density integrated over the intersection polygons), see Fig. 2.2 left. This straighforward method has some drawbacks :

1. it is an exact method only for a constant density function leading, in some sense, to a first-order accurate method,
2. the exact intersection of two polygonal meshes is a demanding algorithm to implement and, most of all, a relative expensive method.

This has led several authors to reconstruct the underlying function as a piecewise linear function [127, 128]. If so the remapping method is exact for linear function and, as such, is considered as a second-order accurate method. However generation of non-physical remapped quantities enforces the utilization of slope or flux limiters in the reconstruction. Such limiters may not always be trivial to properly define as instance when a vector field is to be reconstructed. Usually the limitation is independently applied to each component leading to a frame dependent limitation [2].
On the other hand to overcome the cost of the exact intersection, in the case of a fixed connectivity rezoning, some authors have proposed to use a kind of donor cell method [3]. This method is often refered to as the *swept region remapping* [127, 128] because in the 2D context this method considers the motion of each edge from its old position to its new position, see Fig. 2.2 right. This method is tremendously less expensive than the exact intersection remapping as only the region swept by the edges of a generic cell are to be computed. Between two cells sharing an edge the swept region implicitely determines a "donor" cell. This donor cell donates the integrated quantity of conservative variable over the swept region to the second cell. The swept region remapping has some drawbacks also. The mass flux can only occur between cells sharing one edge, corner cells as a consequence never interact (as instance cells 0 and 2 in Fig. 2.2-right). In other word the flux may be evaluated from an inappropriate cell (edge sharing cell instead of corner cell) [137]. Moreover the swept region may auto-intersect or may be inaccurately computed, respectively observe the red swept region and green/magenta regions on Fig. 2.2 on the right panel. These geometrical errors can however be overcome at little cost as instance with clever techniques such as the ones developed in [138].

2. This is one of the reasons why frame invariant limiters [23, 136] have been recently designed.
3. This corresponds to the forward in time upwind scheme for a transport equation at constant speed.

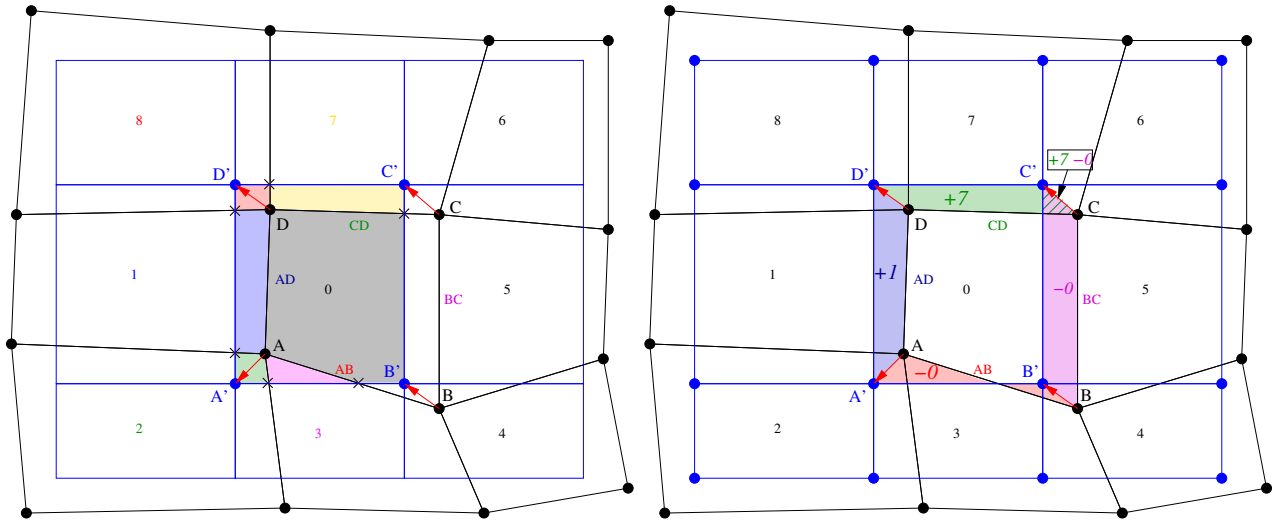


FIGURE 2.2 – *Remapping methods from an old black mesh onto a new blue mesh keeping the same connectivity. Cell* 0 *is surrounded by cells* $1, \ldots, 8$. *— Left : exact intersection based remapping. New blue cell* 0 *is paved with small pieces of old neighbor cells. Red polygon corresponds to the intersection with corner black cell* 8 *(red number), blue polygon to the intersection with left black cell* 1 *(blue number), grey polygon to the surface of old cell* 0 *remaining in new cell* 0. *— Right : swept region method. This method only considers flux between cells across a common edge. Edge* AD *moves from its old position to its new position* $A'D'$*, doing so it sweeps an area* $ADD'A'$*. The new mass is updated taking some mass from the donor cell* 1 *(hence the* $+1$ *in the picture). Sometimes swept areas overlap (dashed green and magenta triangle), or some swept area auto-intersects (swept region of edge* AB *in red), this leads to second-order errors.*

Nonetheless the remapping of a single cell-centered variable apart from second-order errors is exactly or approximately achieved. One important difficulty rises because of the staggered placement of variables to be remapped. While density and mass are located at cells (or subcells if anti-hourglass force is used), velocity is defined at points. Momentum is therefore the dual cell centered value (we will also say point centered value which is an abuse of notation)

$$\boldsymbol{Q}_p = m_p \boldsymbol{U}_p, \tag{2.1}$$

with the mass point $m_p = \sum_{c \in \mathcal{C}(p)} m_{cp}$. More precisely the momentum is defined on the dual mesh (a dual cell is defined by the subcells around a point). Consequently the momentum should be conservatively remapped from the Lagrangian dual mesh onto the rezoned dual mesh. However the mass in the dual cell has presumably changed impacted by the cell-centered mass remap phase. Properly taking into account this interleaved cell-centered mass remap and point-centered momentum remap is not obvious and usually demands some additional approximations which may or may not be properly justified. The reader is refered to [52] section 3.5 for an overview of the complication brought by staggered placement of velocity and density variables when conservative momentum remapping is desired.

Furthermore energy remapping phase still needs to be discussed. At first glance either internal to total energy can be remapped. Most of staggered hydrocodes remap internal energy like any other conserved quantity. Therefore total internal energy is conserved. Unfortunately momentum conservation does not imply kinetic energy conservation. Consequently the total energy as the sum of kinetic and internal energies decreases in time because of the numerical diffusion generated during the momentum remap. Unavoidably total energy conservation is lost with the unpleasant possibility that shock waves may weaken. Some tricks are then triggered to reduce the loss of total energy at shock fronts, see Benson [52] (section “advecting energy”) like the drastic conversion

of kinetic energy losses into internal energy. In some sense these tricks are intended to "repair" the damages brought by the remapping phases. Alternative strategies have also been considered like the remap of the total energy then the deduction of the internal energy from substraction to the kinetic energy deduced from remapped momentum. However monotonicty of the resulting internal energy is not ensured which, more or less, may lead to unrealistic heating of materials [139] [4] If so retrieving physical relevant quantities may demand some 'boarderline' repair actions [5]. As a matter of fact the problem of interleaved remap phases of conserved quantities defined on different geometrical entities is not a trivial task when mass, momentum and total energy conservation is required in addition to the fact that physical relevant variables must ultimately be provided [6].

Some of our investigations on ALE falls within this framework and we present in the next section a selection of published works.

4. Alternative ways exist. In [140] the authors propose a potentially kinetic-energy-conservative algorithm for remapping nodal velocity in a staggered Lagrangian scheme, the improved algorithm is based on the minimization of a functional which may introduce oscillations in the velocity remapped field. Consequently the authors suggest to combine this approach with the low-order donor method by flux-corrected remap (FCR).

5. By 'boarderline' we explicitely emphasize the fact that sometimes for a simulation to run to completion, some shameful 'arrangements' with physcics have to be taken.

6. This somehow explains why effective cell-centered Lagrangian schemes are of particular importance in the context of ALE.

2.2 Arbitrary-Lagrangian-Eulerian (ALE)

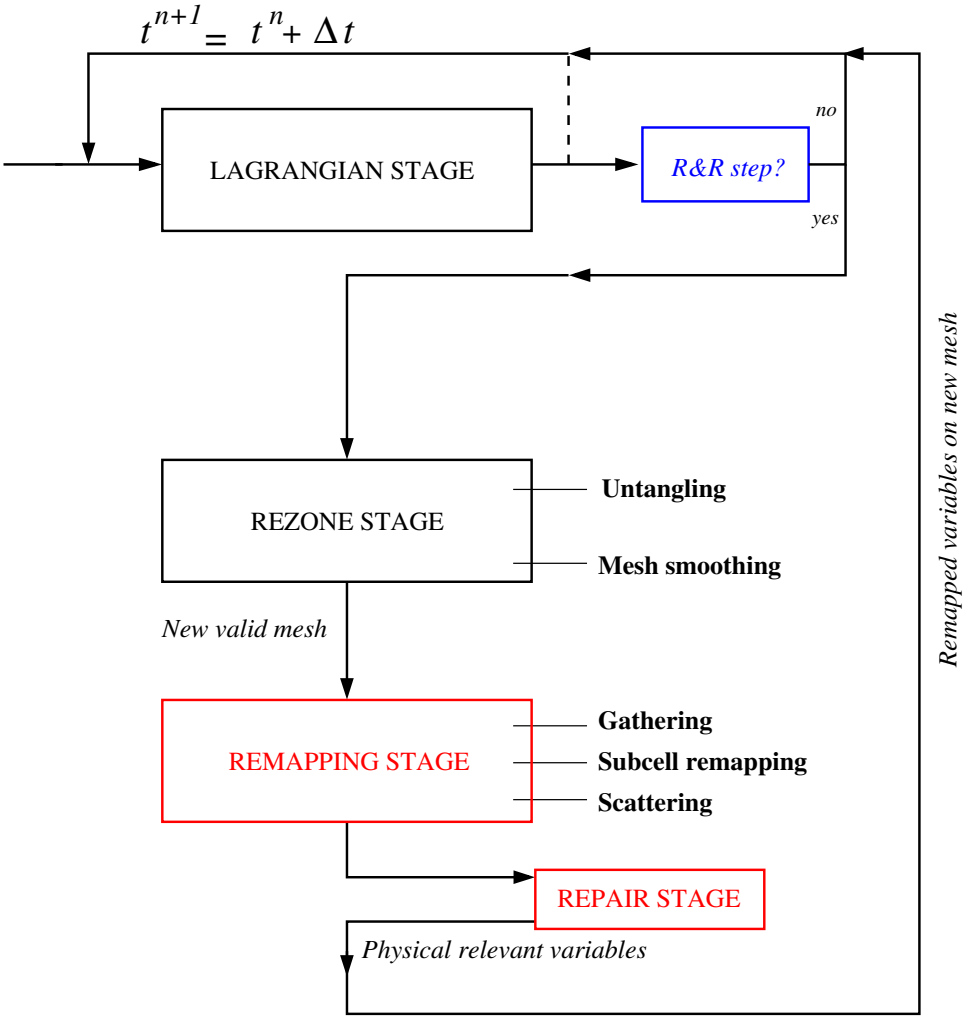


Figure 2.3 – *Description of the ALE scheme within ALE INC(ubator). A Lagrangian scheme is followed by a rezone phase which determines a smoothed mesh onto which a remapping phase conservatively projects the physical variables and ultimately provides physical relevant variables (thanks to a so-called repair technique). Our contributions mainly focus on the red boxes.*

A simplified algorithm of the ALE method implemented into ALE INC(ubator) is depicted in Fig. 2.3. In the following sections we mainly focus on our papers dealing with the "remapping stage" and the "repair stage" of this algorithm (the red bloxes in the figure).

2.2.1 Remapping

While developing the 2D ALE code ALE INC(ubator) with M.J Shashkov at the Los Alamos National Laboratory [5] we had to face the situation of remapping cell-centered density and specific internal energy and nodal velocity under the constraints of mass, momentum and total energy conservation for general polygonal mesh. These variables are the ones provided by the compatible staggered Lagrangian scheme [55, 63, 64, 12] described in the previous chapter. The staggered placement of variables complexifies the notion of conservation as the remapped cell-centered entities

must somehow be consistent with the remapped vertex-based entities.

In article [7] entitled *"A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods"*, M.J. Shashkov and I proposed a solution to this problem.
Before describing our solution to this interleaved remapping phases let us make a general comment related to the compatible staggered Lagrangian discretization that was eluded during the derivation of the scheme but is of some importance for the remapping phase of the ALE method. As already seen the primary energy variable is the cell centered specific internal energy ε_c. Moreover the conserved total energy used to derive the compatible staggered Lagrangian formulation, in other words the fact that total energy is conserved to round-off error in the whole domain, is implicitely obtained considering the subcell-centered total energy

$$E_{cp} = m_{cp}\varepsilon_c + \frac{1}{2}m_{cp}\|\boldsymbol{U}_p\|^2. \tag{2.2}$$

This derives from the total kinetic energy equation (1.18) and total internal energy equation (1.19) the sum of which defines the total energy (1.20) as a global entity over the domain (see the derivation of the compatible scheme in section 1.2.2)

$$E = \sum_c m_c\varepsilon_c + \sum_p \frac{1}{2}m_p\|\boldsymbol{U}_p\|^2.$$

The subcell-centered total energy (2.2) is revealed using the previous equation and the definition of cell mass m_c (see (1.10)) because

$$\begin{aligned} E = \sum_c \left(m_c\varepsilon_c + \sum_{p\in\mathcal{P}(c)} \frac{1}{2}m_{cp}\|\boldsymbol{U}_p\|^2 \right) &= \sum_c \left(\left(\sum_{p\in\mathcal{P}(c)} m_{cp} \right) \varepsilon_c + \sum_{p\in\mathcal{P}(c)} \frac{1}{2}m_{cp}\|\boldsymbol{U}_p\|^2 \right) \\ &= \sum_c \sum_{p\in\mathcal{P}(c)} m_{cp} \left(\varepsilon_c + \frac{1}{2}\|\boldsymbol{U}_p\|^2 \right) = \sum_c \sum_{p\in\mathcal{P}(c)} E_{cp}, \end{aligned}$$

leading to the definition of the subcell-centered total energy (2.2) given above. Remark that working on dual cells produces the same subcell-centered total energy definition because

$$\begin{aligned} E = \sum_p \left(\sum_{c\in\mathcal{C}(p)} m_{cp}\varepsilon_c + \frac{1}{2}m_p\|\boldsymbol{U}_p\|^2 \right) &= \sum_p \left(\sum_{c\in\mathcal{C}(p)} m_{cp}\varepsilon_c + \frac{1}{2} \left(\sum_{c\in\mathcal{C}(p)} m_{cp} \right) \|\boldsymbol{U}_p\|^2 \right) \\ &= \sum_p \sum_{c\in\mathcal{C}(p)} m_{cp} \left(\varepsilon_c + \frac{1}{2}\|\boldsymbol{U}_p\|^2 \right) = \sum_p \sum_{c\in\mathcal{C}(p)} E_{cp}. \end{aligned}$$

Therefore a compatible definition of total energy with the discrete staggered Lagrangian scheme reveals that total energy must be a subcell-centered entity.
In addition our discretization employs subcell masses that serve to introduce anti-hourglass force [56], see section 1.2.3. This adds an additional requirement to the remap phase — that the subcell densities (corresponding to subcell masses) have to be conservatively interpolated.

As a consequence the main goal of the work in [7] is to build subcell-centered conservative mass, momentum and total energy entities, remap based on subcells and finally recover the primary variables by association of new subcell remapped entites.

In this work, we assume that the rezone algorithm produces mesh that is "close" to the Lagrangian mesh so that a local remapping algorithm (i.e, where mass and other conserved quantities are only exchanged between neighboring cells) can be used. The swept remapping is used in practice although an exact intersection algorithm has been also implemented and tested.
Our new remapping algorithm consists of three stages.

First : *gathering stage*. We define mass, momentum, internal energy, and kinetic energy in the subcells. They are defined in such a way that the corresponding total quantities (defined as the sums over subcells) are the same as those at the end of the Lagrangian phase, ensuring that the gathering stage is conservative.

Second : *subcell remapping stage*. We use the algorithm described in [128] to remap mass, momentum, internal, and kinetic energy from the subcells of the Lagrangian mesh to the subcells of the new rezoned mesh. This algorithm is linearity-preserving and computationally efficient. It consists of a piecewise linear reconstruction and an approximate integration based on the notion of swept regions. The algorithm does not require finding the intersections of the Lagrangian mesh with the rezoned mesh, which contributes to its efficiency. The algorithm is conservative : total mass, momentum, internal and kinetic energy over subcells of the rezoned mesh are the same as mass, momentum, internal and kinetic energy over subcells of Lagrangian mesh. The total energy is also conserved, being the sum of (individually conserved) internal and kinetic energies.

Third : *scattering stage*. We recover the primary variables — subcell density, nodal velocity, and cell-centered specific internal energy — on the new rezoned mesh.
- Subcell density is recovered by using the remapped mass and volume of the subcell of the rezoned mesh. The subcell masses and the corresponding densities are then adjusted using a conservative repair procedure [128], [141], [142], [8], [9] to enforce local bounds (see also the next section for details), which may be violated during the remapping stage. This produces the final subcell density and the corresponding subcell mass that will be used in next time step.
- Next, we define the remapped nodal momenta using the remapped subcell momenta, in such a way that total momenta is conserved. New velocity components are defined by dividing by nodal mass. Then nodal velocity is repaired, resulting in the final velocity that will be used to move the point during the Lagrangian phase in the next computational cycle.
- To enforce the conservation of total energy, the discrepancy between the remapped kinetic energy in the cell and the kinetic energy that is computed from the remapped subcell masses and the final nodal velocities is contributed to the remapped internal energy in the cell. Finally, the internal energy and the corresponding specific internal energy are conservatively repaired.

Our remapping algorithm satisfies the following requirements :
- *Conservation*. The total mass, momenta and energy of the new mesh must be the same as that of the old mesh. This property, combined with the same conservation properties of the Lagrangian phase, guarantees the conservation of the overall ALE method.
- *Bound-preservation*. The remapped density, velocity components and internal energy have to be contained within physically justified bounds, which are determined from the corresponding fields in the Lagrangian solution. For example, density and internal energy have to be positive. Moreover, because we assume that the new mesh is obtained from a small displacement of the old mesh, one can require that the new values lie between bounds determined by the values of its neighbors on the old mesh, [128, 141, 8, 9].

– *Accuracy*. It is straightforward to define accuracy in the remap of density ; we will require that the remap of density is linearity-preserving. That is, if the values on the old mesh are obtained from a global linear function, then the values on the new mesh have to coincide with the values of the same linear function on the new mesh. For the remap of velocity, there are several different notions related to accuracy. For example, one widely used test of consistency is the so-called DeBar condition (see for example [52]) which can be stated as follows : if a body has a uniform velocity and spatially varying density, then the remapping process should exactly reproduce a uniform velocity. For internal energy unfortunately the situation is more complicated. We have demonstrated the accuracy of our new algorithm through the practical expedient of well-chosen test problems.
– *Reversibility*. If the new and old meshes are identical, then the remapped primary variables should endure no change. This property is closely related to the notion of being free of inversion error, see [52], where it is stated that if the new and old grids coincide, then the remapped velocity on new mesh should coincide with the velocity on the old mesh.

We have also demonstrated computationally that our new remapping method is robust and accurate for a series of test problems in 1D (Sod shock tube, Collela-Woodward blastwave, Le Blanc shock tube) and 2D (Sedov problem on quadrangular and polygonal meshes).
As instance in Fig. 2.4 one reproduces the results of the ALE scheme in its Eulerian, Lagrangian and ALE regimes for the Sedov problem on polygonal grid.

Using the remapping method developed in this paper we have constructed a full staggered ALE code working for polygonal meshes. The method combines and generalizes previous work on the Lagrangian and rezoning phases [55, 124, 125], and includes this new remapping algorithm [7]. This work has been implemented into the code ALE INC(ubator) [5].

2.2.2 Repair

With B. Wendroff and M. Staley we have investigated new repair methods in a paper entitled *The Repair Paradigm : New Algorithms and Applications to Compressible Flow* in [8]. The goal in this paper is to improve upon and apply the repair idea introduced in [142, 141, 128].

A repair method can be viewed as a way to correct values on a discrete mesh by redistributing the conserved quantity so that conservation and a maximum principle are preserved. The maximum principle states that new values should obey certain upper and lower bounds obtained from old values. In this way, not only are non-physical quantities eliminated, but oscillations are reduced (albeit not necessarily eliminated). We therefore seek repair algorithms that can be applied to CFD problems, advection problems, or other situations where values of a discrete variable must be placed in bounds without violating a conservation law and without introducing significant errors in the dynamics. As stated in [141] (Section 8, page 275), repair is a mass redistribution nonlinear filter. Notice that this technique is vaguely apparented to other methods for the correction of nonphysical data, such as Flux Corrected Transport, which are discussed in [143].

As we have seen a critical part of Lagrangian-based methods and ALE methods for Computational Fluid Dynamics (CFD) is the ability to remap or interpolate data from one computational mesh to another. Remapping is also essential for pure Lagrangian methods, because they can lead to tangled grids that must then be untangled with a concomitant remap step. Even if the basic scheme produces only physically meaningful quantities, a remapping method can create out-of-bounds quantities such as negative densities or pressures. In some CFD codes, the offending values are simply set to a small positive number when this occurs, at which point mass or total energy is no

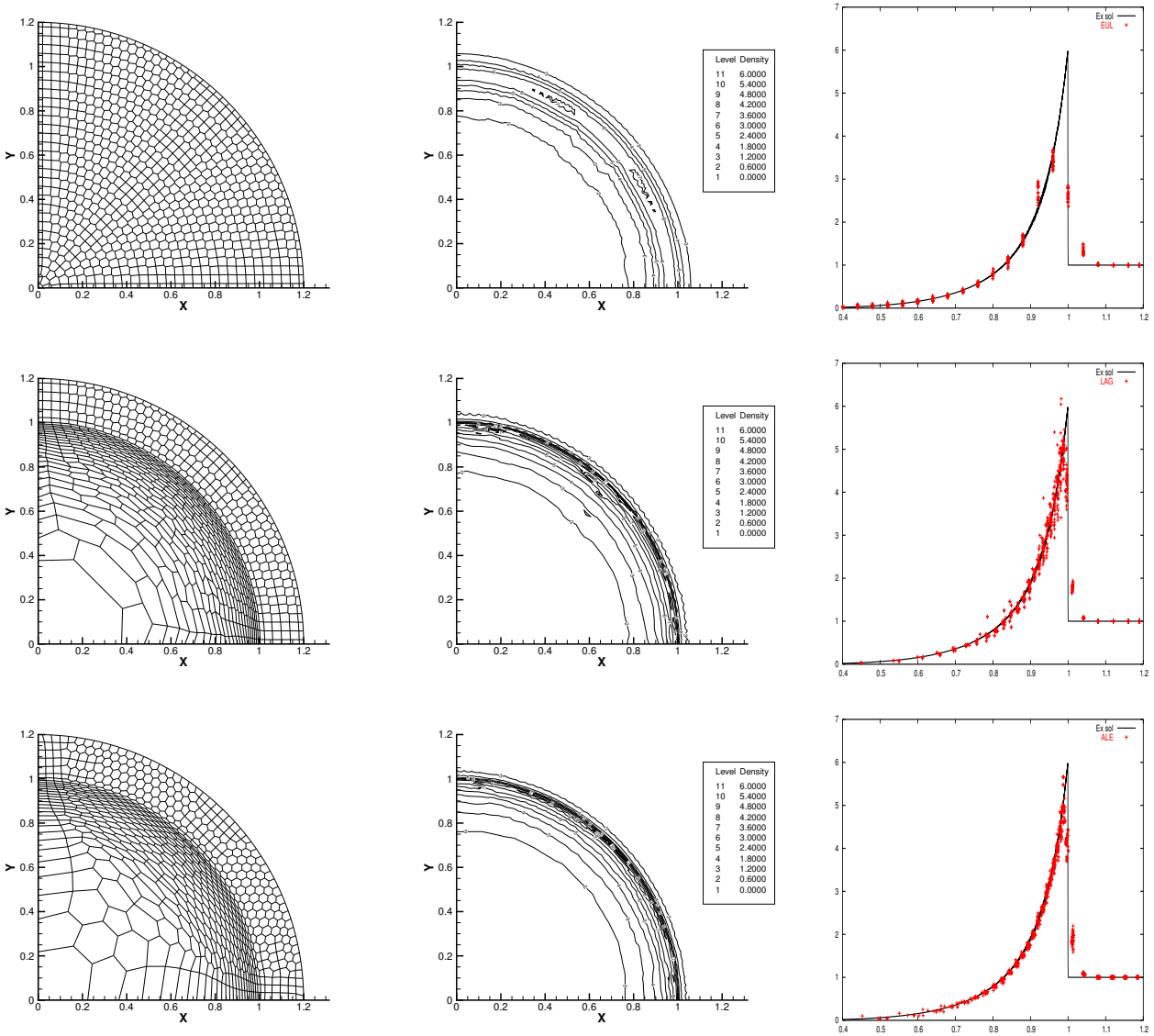


FIGURE 2.4 – *Numerical results from paper [7]. Sedov problem on polygonal Mesh — Mesh (left column), density isolines (middle column) and density as function of radius for all cells (right column) at $t = 1.0$ — Eulerian regime (top line), Lagrangian regime (middle line), ALE regime (bottom line).*

longer conserved. In most instances the error thereby created is negligible, but we have shown that in at least one example the error is significant. It is possible, by taking great care with the remapping in the CFD context, to maintain positive mass density. This is done by first extending the given mean densities in each original cell to the whole domain so that the new distribution is everywhere positive, and then computing new mean values by exact integration over the cells of the new grid. Total energy can be remapped in this way, but then there is no guarantee that internal energy will be positive. Furthermore, in three dimensions, exact integration is computationally intensive. Another context in which non-physical data can occur is in divergence-free advection of a concentration that must retain values between zero and one. High quality advection schemes, some of which are based on remapping ideas ([144], [134]), unavoidably have this fault ([143]). In the case of advection of

a concentration, repair keeps the newly computed concentration in a cell between the maximum and minimum concentrations in neighboring old cells, thus guaranteeing at least that the new concentration is between zero and one.

In this paper we have reviewed and applied several conservative repair methods that can be used in situations where variables must stay between predefined bounds while respecting conservation. Such situations occur often, in hydrodynamics for example, when the density or the specific internal energy becomes negative due to remapping. Such unphysical situations must be cured, but replacing negative values by small positive numbers is not acceptable from the point of view of conservation. The methods developed in this paper are

Local order-dependent repair. This is perhaps the most obvious local repair algorithm. The underlying idea is to expand the neighborhood of a cell i which needs repair, until enough room is found in this neighborhood. Suppose cell i has a negative density, but the minimum bound is 0. This repair algorithm expands the neighborhood of cell i until enough mass can be found and removed from the neighborhood to fill cell i and produce a repaired density equal to the minimum bound, that is, to 0. Then, the next cell is checked and repaired if necessary. A similar concept is applied to repair an over-bound value. This repair algorithm is order-dependent, meaning that the final result depends on the order in which cells are visited. This unphysical order-dependence is unacceptable in many practical situations.

Global order-independent repair. A simple, global order-independent repair algorithm clips out-of-bounds values to their bounds, counts the total discrepancy this produces in the quantity to be conserved, and spreads the discrepancy over the entire mesh. This method is order-independent because any cell that has to be repaired is immediately brought to its nearest bound and contributes to a total discrepancy which is not accounted for until all individual cells have been repaired. Clearly, this algorithm is conservative and order-independent. It is also symmetry-preserving, in the sense that equivalent cells (cells which have the same mass and bounds) are treated in the same manner. The drawback is that such a repair process can violate the physics in computational fluid dynamics. In other words, such a repair process on hydrodynamics problems can severely perturb the physics of the phenomena one is trying to study ; it may violate the causality. However for a pure advection problem, the global repair algorithm can be appreciated for its simplicity and its ability to parallelize.

Local order-independent repair. We have also developed an iterative, order-independent repair algorithm that addresses the disadvantages of the previous algorithm. This is a two-stage algorithm : one stage repairs all values that are above their upper bound, and the other stage repairs all values that are below their lower bound. Upper bounds can be fixed before lower bounds, or vice versa. The order affects the result, but given a choice of order, the algorithm produces the same result regardless of the order in which cells are examined. This algorithm converges, is conservative and order-independent, and can preserve a 1D symmetry with the modifications outlines above. However, on parallel machines this algorithm (as well as the local order-dependent method) is slow. This is largely due to the neighborhood expansion needed by both algorithms.

Mixed local/global order-independent repair. The global repair algorithm needs very little communication and can be used very efficiently in a parallel framework. We consider an amalgamation of the local algorithm, which gives more physically meaningful results, and the global algorithm, which is more parallelizable. The mixed local/global order-independent repair algorithm is based on the assumption that most of the out-of-bounds cells can be fixed locally (using only the immediate neighborhood) because they are due to very small disturbances, and that, as

a corollary, only a few cells need to find room/mass far away from their location. The idea of this algorithm is to repair as many cells as possible with the local order-independent algorithm, and then if some of the cells are still out-of-bounds, to repair them with the global repair algorithm.

1. **Local** treatment : for all out-of-bounds cells, try to repair with the Local order-independent symmetry-preserving algorithm, but without expanding any immediate neighborhoods. If the current cell still has excess mass, then leave it out of bounds. Iterate this process in order to converge to a situation where either every cell is repaired, or the remaining unrepaired cells cannot be repaired using their closest neighborhood. Our experimentation indicated that few iterations are needed.
2. **Global** treatment : for any remaining out-of-bounds cells, perform the global repair. This step finally fixes the remaining out-of-bounds cells, the number of which is presumably small, and which should be out of bounds only by small amounts

The mixed local/global repair algorithm is conservative, because each of its steps is conservative. Moreover, both the local and global treatments are order-independent and symmetry preserving, and there is no particular difficulty with parallelization because there is no indefinite neighborhood expansion. If any cells are still out-of-bounds after the first step, the global repair fixes them. The earlier argument stating that this method can violate causality still holds, but the effect is far less pronounced because very few cells will remain to be fixed after the initial local treatment, and the amounts by which they need to be fixed will be less. Therefore, the causality violation is negligible.

Of the methods presented in this paper, we believe that the mixed local/global repair scheme best meets the requirements of locality and efficiency. This method was applied to an advection example and to test cases in an ALE hydrodynamics framework, where the use of a conservative repair algorithm allowed us to :

– preserve the accuracy of the underlying method, as in the Sod Riemann problem ;
– stabilize and improve bad profiles, as in the Le Blanc Riemann problem ;
– maintain physical and reasonable results, as in the blast wave interaction problem and the Sedov problem.

In Fig. 2.5 one represents the results obtained on the Sedov problem in ALE regime (rezone and remap every ten cycles) at times $t = 0.1$ and $t = 1.0$, the density variable and the mesh are displayed. At time $t = 1.0$ one provides the cell-based density value as a function of the radius (right panel). When no repair is performed in this test case, the code stops due to the creation of negative internal energy after the fourth remapping. On the other hand, the use of a repair method fixes the parasitic negative values and allows us to observe good results. The maximum density with the Mixed Local/Global repair method is 5.62.

In paper [9] entitled *Convergence and Sensitivity Analysis of Repair Algorithms in 1D* B. Després made most of the the convergence analysis of some simple Repair algorithms and I joint his effort to numerically retrieve his convergence and sensitivity results. In this paper we limit the theoretical study to transport equation in 1D and to some simple remapping strategy in 1D. In order to prove the main convergence result we introduced a box of size p in which the distribution of the excess of mass is performed. Numerical results are proposed for the Lax-Wendroff (LW) scheme plus Repair and the DownWind (DW) scheme plus Repair. The LW intends to be representative of high-order prediction schemes. The DW scheme intends to be representative of highly anti-dissipative prediction schemes. In a specific section we study the gas dynamics equations in 1D with a Lagrange+Remap code. This code is built on two components : a staggered Lagrangian scheme and

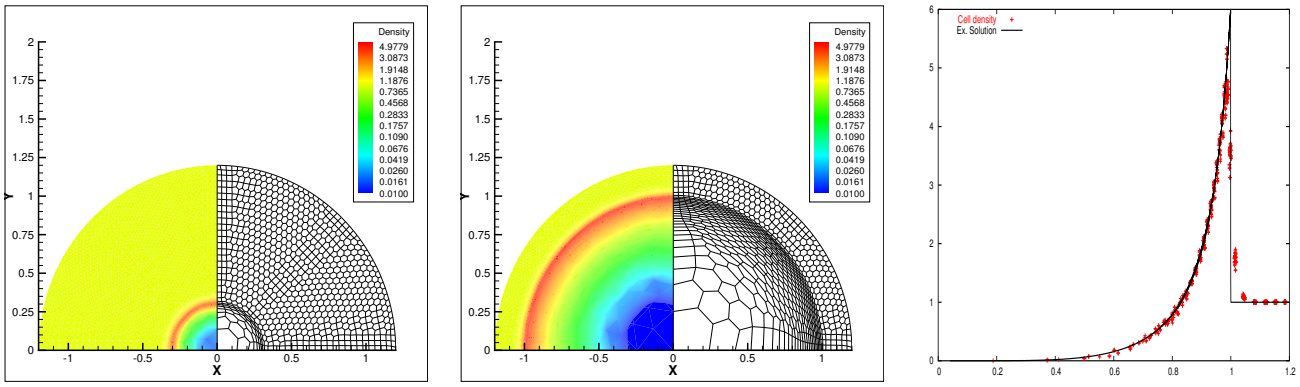


Figure 2.5 – *Numerical results from paper [8]. Sedov problem on polygonal Mesh —Sedov blast wave on a polygonal mesh (*1325 *nodes and* 775 *cells). ALE–1o regime — Left-Middle : mesh and density contours (exponential scale) at* $t = 0.1$ *and* $t = 1.0$*. Right : density at* $t = 1.0$ *(cell-based value as a function of the radius)*

a Remap strategy which may need repair. The Sod and Le Blanc shock tubes and the blastwave of Colella-Woodward are tested. It seems that the size of the box (parameter p) is not necessarily an important parameter if one uses a high-order prediction scheme as the Lax-Wendroff scheme for non oscillating computations. But with a more anti- dissipative prediction scheme as the Downwind scheme, the results can vary with p. When p is too large the numerical solution may not be correct. It gives some indication that it is much preferable to restrict ourselves to local Repair (i.e with a local redistribution of the mass).

An important feature of the Repair paradigm is the simplicity and versatility in any dimension. Moreover the repair process is independent of the kind of mesh used ; cell-centered values or nodal values can be repaired the same way. We only need the notion of neighborhood to define the bounds and to redistribute the amount of conservative variable. Therefore any repair algorithm is suitable for staggered formulation where physical variables are not defined at the same location.

2.3 ReALE : Reconnection ALE

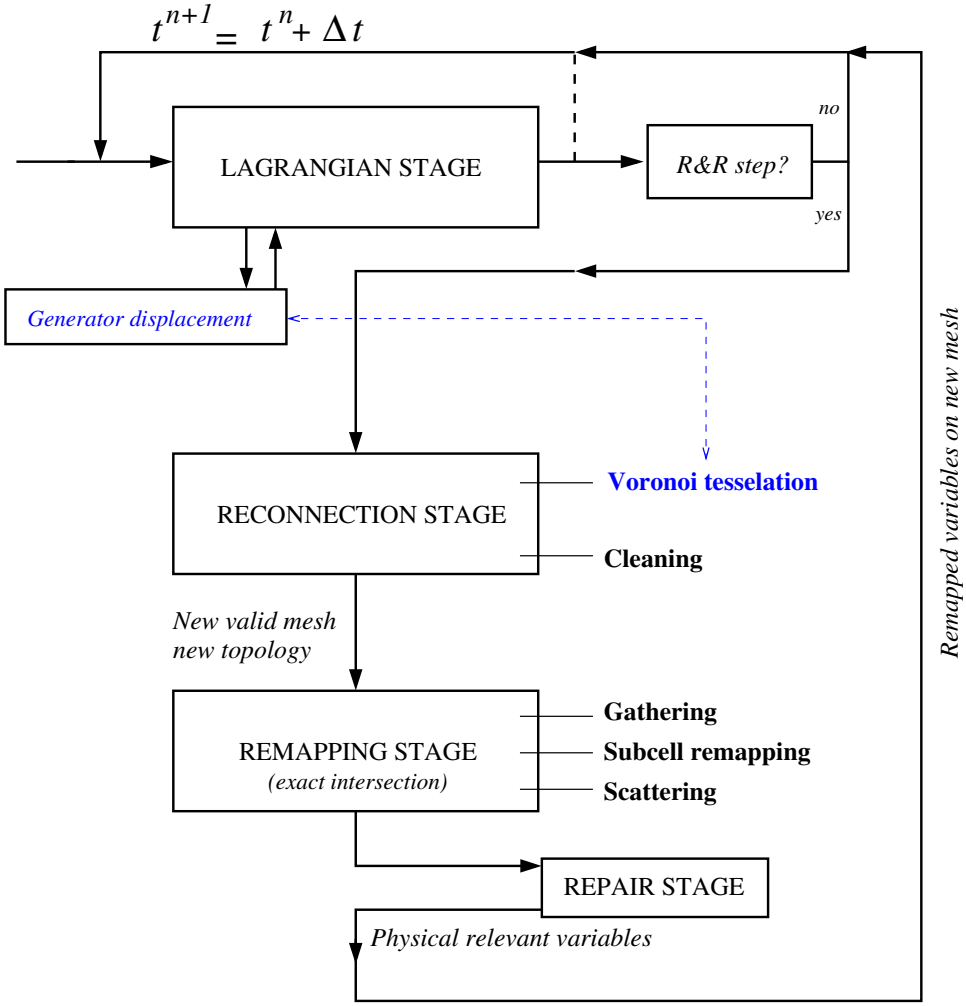


FIGURE 2.6 – *Description of a ReALE (Reconnection ALE) scheme within ALE INC(ubator). A Lagrangian scheme is followed by a reconnection stage which determines the Voronoi tesselation from moving generators (computed within the Lagrangian stage) onto which a remapping phase conservatively projects the physical variables and ultimately provides physically relevant variables (thanks to a so-called repair technique). The remapping must be done by exact intersection between the old Lagrangian mesh and the new mesh from the reconnection stage, the later may have a different connectivity.*

In this section we present the joint effort with colleagues from LANL (M. Shashkov) and CEA (J. Breil, S. Galera, P.-H. Maire) to extend the ALE technique to allow mesh reconnection during the simulation. In Fig.2.6 one proposes a sketch of our Reconnection ALE (ReALE) algorithm to provide the big picture to be compared to a fixed connectivity ALE in Fig. 2.3. Some of the work made within this context is described in details in this chapter. This work has led us to write paper [26] entitled *"ReALE : a reconnection-based arbitrary-Lagrangian-Eulerian method"* which sets the fundation of two reconnection-based ALE codes : a cell-centered ReALE code named CHIC developed and maintained at CELIA in Bordeaux , and ALE INC(ubator) in its ReALE version maintained at IMT in Toulouse. In paper [27] an extension to cylindrical geometry is also proposed.

In this section we recall the genesis of ReALE and the reasons why this work has been undertaken. Then a quick refresher on Voronoi machinery is proposed followed by a some details and specific treatment that a ReALE code needs. Finally ReALE results are provided.

"Why fooling around with a Reconnection ALE method ?"

The genesis of this work starts with the following statement : The most difficult and least developed phase of ALE is the rezoning phase. A review of existing rezone strategies for ALE methods is presented in [124] including analysis of alternative approaches [145, 146, 147, 108, 148, 112, 149, 116, 150, 151, 115]. A review of a more general class of methods, namely moving mesh methods, is presented in [152]. Ideally the mesh has to adapt to the solution. Any adaptive scheme is composed of three main ingredients : an optimal-mesh criterion, an error estimator or error indicator, and an algorithm of the strategy for the mesh improvement. These ingredients answer to the following questions : How should the optimal mesh be defined ? Where are mesh changes required ? And how should the improved mesh be constructed ? For standard ALE methods a strategy for mesh improvement is based on moving the spatial grid.

Generally speaking the goal of rezoning is to improve the efficiency of the ALE method However, to design an adaptive method one needs a quantitative assessment of optimality. The problem is that, for non-linear equations of gas dynamics in 2D and 3D, at the moment, it is not feasible to obtain such quantitative assessment. For this reason practitioners are usually using some qualitative approaches. In real complex ALE simulation the most basic goal of rezoning is simply to run calculation to completion without user intervention and still achieve reasonable accuracy (recall that we always can run ALE in Eulerian=Lagrange-Plus-Remap mode, which will be robust but less accurate). Even this goal is usually not achieved in most production ALE codes. For example, even for very popular methods based on Winslow smoothing, [115, 116], practical simulations require the introduction of numerous geometrical and physics based triggers and lockers, that is, mesh constraints that typically keep a node Lagrangian until some condition is reached e.g. element quality criterion (to detect cell distortion or collapse) or physical condition is reached in surrounding elements (for example, did the cell fully detonate ?) [112, 153, 154, 138].

As it is mentioned in [153, 154], the mesh movement philosophy applied to most applications, related to high-speed multimaterial flows, is to develop algorithms that will move the mesh in such a way as to maintain robustness while staying as close as possible to the Lagrangian mesh motion. The Lagrangian mesh motion naturally follows most flow features of interest such as shocks, material interfaces and steep gradients and allows users to focus zoning in materials of interest. Mesh relaxation is then used in regions of high material deformation to improve mesh quality. In standard ALE methods, which use fixed mesh topology, nodes are moved to refine some areas of the problem at the expense of derefining other parts of the problem. Generally, the increase of mesh resolution is limited, and, most importantly it can degrade the mesh quality leading to robustness problems.

One if not the main reason for this is that standard ALE codes utilize a fixed topology mesh, defined at the outset, which in general will not be able to adapt to the dynamically evolving interface shape

(or contact discontinuity) in spite of efforts at regularization, see the next figure for an illustration.

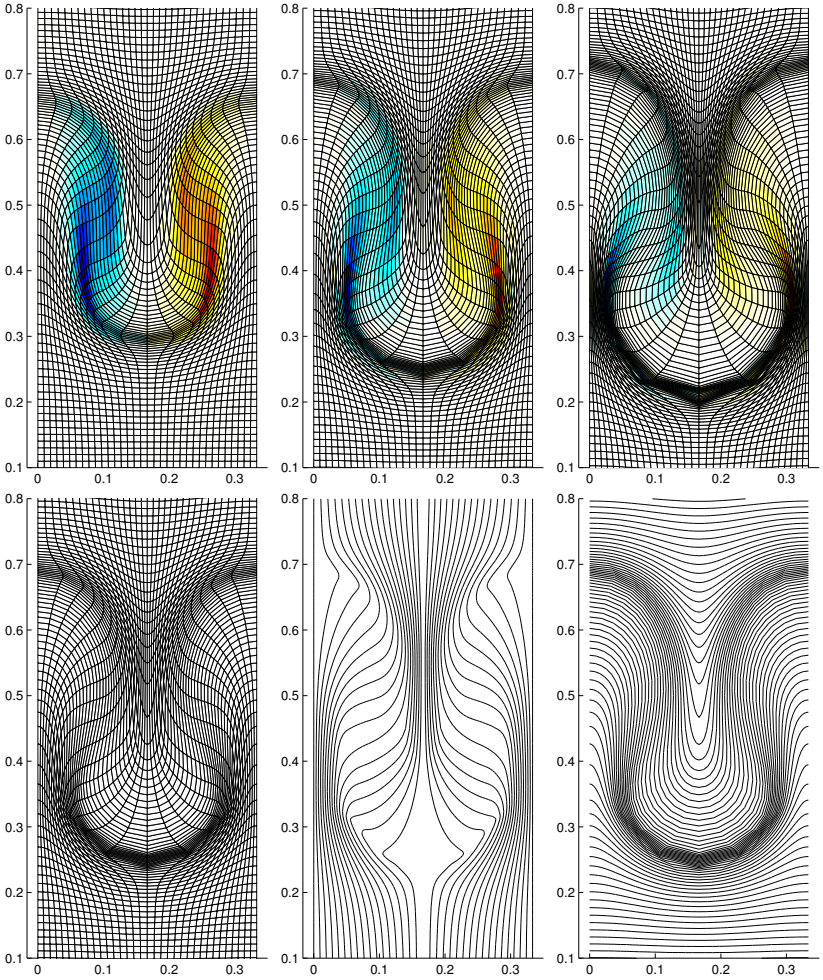


Illustration of mesh stagnation in a fixed ALE simulation. Rayleigh-Taylor problem on logicaly rectangular mesh.
Top panels : mesh fragments and vorticity color map at time moments $t = 7, 8, 9$. Color scale is from blue (minimal negative vorticity) to red (maximal positive vorticity). White color corresponds to zero vorticity.
Bottom panels : fragment of mesh at $t = 8$: mesh fragment (left), "Vertical" logical lines (middle), "Horizontal" logical lines (right).
Because one can not expect the mesh to follow anymore the vortex like motion without tangling, the rezone strategy locks the mesh : the code can not help but continue in an almost Lagrange+Remap strategy.

The most general solution to this difficulty, while preserving a Lagrangian nature of mesh motion is to relax the constraints on mesh topology and allow reconnection. The idea of using mesh reconnection to solve partial differential equations is not new. To the best of our knowledge, in context of computational gas dynamics the ideas related to mesh reconnection were first used in [155] In this seminal paper the authors suggest to use a set of point Lagrangian particles and surround them with domains (parcels) to describe the media. The shape and size of these parcels are determined by the positions of the particles. The connectivity of the set of particles is not fixed but can vary with time depending on relative positions of the particles. After connectivity is established the set of neighboring particles defines the stencil on which Lagrangian equations are discretized.
Paper [155] has all basic ideas that lead to development of so-called free-Lagrange (or free-Lagrangian) methods, [156, 157, 158], which were very popular in 80's and early 90's The name free-Lagrange was introduced in [159] and the corresponding code was called FLAG at LANL.
More recently in the context of cell centered Lagrangian scheme S. Del Pino in [160] has developed a metric-based adaptation technique which also allows automatic triangular mesh reconnection. Allowing mesh reconnection is not a new idea *per se* and several related subjects can be found in the recent literature such as [161], [160] and [162, 163].
In paper [26] a detailed discussion on so called free-Lagrangian methods enlights that these are not genuinely "Lagrangian" methods. Because free-Lagrangian methods are not really Lagrangian then,

Figure 2.7 – *Example of Voronoi meshes from mother nature. Left : dragonfly wings. Middle : girafes. Right : turtle.*

explicitly or implicitly, they incorporate a rezone phase, and consequently, a corresponding remap phase. Some of them explicitely states this remap phase. Nevertheless if the free-Lagrange method does not have a remap phase, errors related to it will manifest itself one way or another. It leads us to the conclusion that methods where connectivity of the mesh can change have to be developed in reconnection-based ALE (ReALE) framework, where rezone stage includes reconnection. Let us note that similar philosophy was used in [147], even so authors of [147] do not call their method ALE or free-Lagrange.

As standard for ALE method, the main elements in ReALE simulation are an explicit Lagrangian phase in which the solution and grid are updated (without changing connectivity), a rezoning phase in which a new grid is defined (which includes changing connectivity and also adding or deleting cells or vertices of the parcels), and a remapping phase in which the Lagrangian solution is transferred (conservatively interpolated) onto the new grid. Our rezoning phase allowing mesh reconnection is based on the Voronoi machinery, however for the sake of clarity we remind some aspects of it in the next paragraph.

"Voronoi machinery"

Voronoi diagrams were first investigated by René Descartes (French philosopher 1596-1650) and applied by Lejeune Dirichlet (Belgium/German mathematician 1805-1859) when exploring quadratic forms, however the diagrams were named after Georgy E. Voronoi (Ukrainian mathematician 1868-1908). To be fair the real inventor of these diagrams is Mother Nature as illustrated in Fig.2.7.

Given generators $\boldsymbol{G}_i$, $i = 1, 2, \ldots, G$, a distance function $d(\boldsymbol{G}_i, \boldsymbol{G}_j)$ the Voronoi cell Ω_j is the set of all points closer to $\boldsymbol{G}_j$ than to any of the other $\boldsymbol{G}_i$

$$\Omega_j = \left\{ \boldsymbol{X} \;\; s.t \;\; d(\boldsymbol{G}, \boldsymbol{G}_j) < d(\boldsymbol{X}, \boldsymbol{G}_i), \; \forall \, i = 1, \ldots, G, \;\; i \neq j \right\}. \tag{2.3}$$

A collection of Voronoi cells $\{\Omega_1, \Omega_2, \ldots, \Omega_G\}$ defines the Voronoi tesselation of $\mathbb{R}^2$ associated to the set of generators. For now we implicitely clip the Voronoi tesselation to the computational polygonal domain.

The "Voronoi machinery" denotes the ability of Voronoi tesselation to assimilate new generators and to perform the necessary modification of connectivity that the presence of these new generators implies. For example in Fig. 2.8 the Voronoi machinery is illustrated when generators (red dots) are successively added (two sequences are shown : 5, 6, 7 and 11, 12, 13 generators). On purpose the

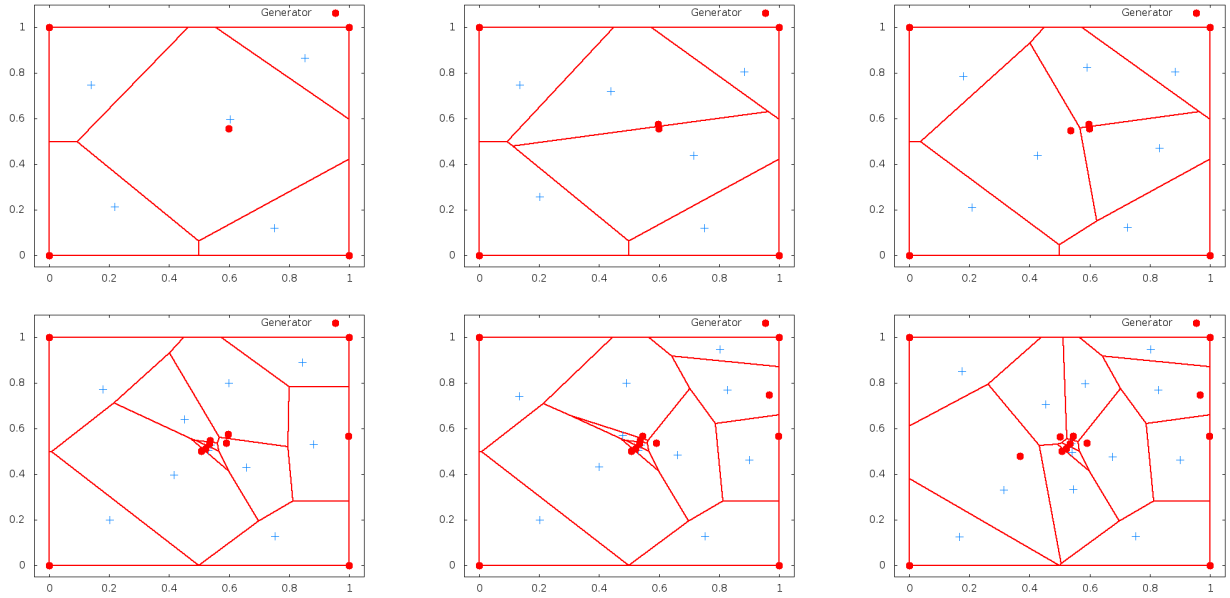


FIGURE 2.8 – *Example of Voronoi meshes obtained from a set of generators (red dots). Cell centroids (blue crosses) generally do not coincide with generators. This illustrate the reconnection ability of the Voronoi machinery when generators are added. Top panels : sequence with* 5, 6, 7 *generators. Bottom panels : sequence with* 11, 12, 13 *generators.*

generators are located close to the domain center, this illustrates the possible roughness of a Voronoi mesh : close cells can have very different surfaces, number and size of edges, shape, etc. Moreover if $\boldsymbol{X}_c$ denotes the centroid of the Lagrangian cell Ω_c, according to

$$\boldsymbol{X}_c = \frac{1}{\mid \Omega_c \mid} \int_{\Omega_c} \boldsymbol{X} \mathrm{d}V,$$

where $\mid \Omega_c \mid$ denotes the volume of the cell Ω_c then we observe in Fig. 2.8 that the centroids (blue crosses) almost never coincide with the generators (red bullets). In fact when these two coincide then the Voronoi tesselation is called Centroidal Voronoi Tesselation (CVT), see Fig.2.9 last panel. In $\mathbb{R}^2$ a CVT is a mesh made of perfect hexagonal cells. One simple constructive iterative algorithm to obtain a CVT (and smooth a Voronoi tesselation when convergence is not reached) is due to Lloyd [164] :

0. Iteration k. Start with generators $\left\{\boldsymbol{G}_i^k\right\}$, $i = 1, \ldots, G$.
1. Build Voronoi cells Ω_i^k associated to $\boldsymbol{G}_i^k$ for all $i = 1, \ldots, G$.
2. Compute $\boldsymbol{X}_i^k$ centroid of Voronoi cell Ω_i^k for all $i = 1, \ldots, G$.
3. Set $\boldsymbol{G}_i^{k+1} = \boldsymbol{X}_i$ for all $i = 1, \ldots, G$.
4. If satisfied with obtained mesh quit, else $k \leftarrow k + 1$ and go back to 1.

In Fig.2.9 one presents an example of the result of the iterative Llyod's algorithm [164] for iterations 1, 2, 3, 10, 20 and 100. This shows that initialy non uniformly distributed generators produce a non-smooth Voronoi tesselation whereas the successive meshes obtained with Llyod's algorithm are more and more regular. At convergence (last panel) a centroidal Voronoi tesselation for which centroid and generator do coincide is produced. A last drawback of Voronoi tesselation is its unpleasant ability to create arbitrary small edges. For a hydrodynamical Lagrangian scheme small edge may

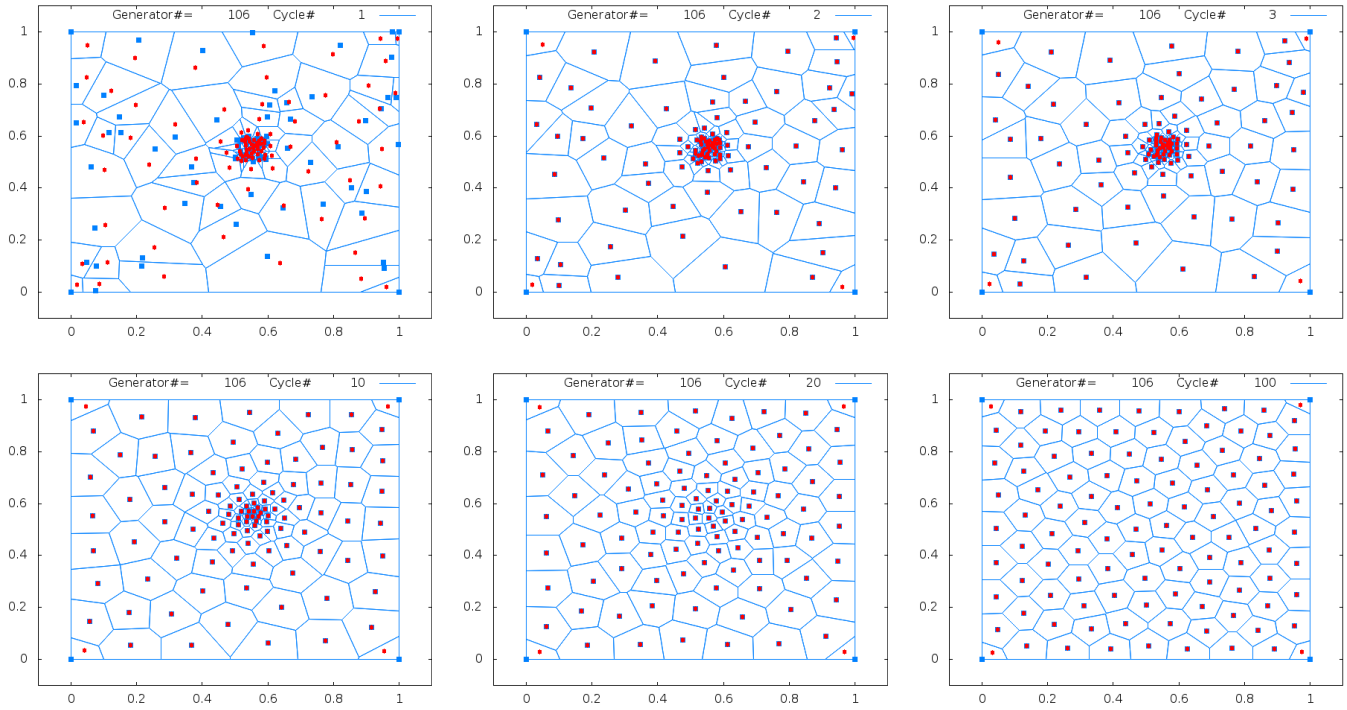


Figure 2.9 – *First panel : example of Voronoi mesh made of* 106 *generators (blue squares). Notice that cell centroid (red circles) usually does not coincide with cell generator. Second to sixth panels : iterations* 2, 3, 10, 20 *and* 100 *of Llyod's algorithm which ultimately produces a centroidal Voronoi tesselation for which centroid and generator do coincide.*

drive the time step to zero, consequently we have added a specific "small edge cleaning" procedure. Given a user-specified threshold edges which length is smaller are discarded and the connectivity is modified accordingly. As instance on the last panel of Fig.2.9 some small edges have been kept, as instance around point $\boldsymbol{X} = (0.2, 0.72)$ or $(0.38, 0.28)$, and some edges have been discarded as instance around $\boldsymbol{X} = (0.78, 0.56)$. As we will see in this the following sections incomplete CVT and Voronoi machinery are heavily used in ReALE.

ReALE specifics : initialisation and Lagrangian phases

For ReALE all three phases (Lagrangian, Rezone and Remap) are supposed to satisfy specific requirements which are different from standard ALE methods. We assume, that at the beginning of the calculation ($t = 0$) as well at the beginning of each time step (after rezone phase) the computational mesh consists of Voronoi cells corresponding to some set of generators. In other words the distribution of generators entirely defines the mesh, see Fig. 2.8 for examples of Voronoi mesh. Because of reconnection in rezone phase, the Lagrangian phase of the ReALE method has to deal with discretization of the Lagrangian equations on general polygonal meshes and corresponding update of this polygonal mesh is supposed to be Lagrangian. There are several papers dealing with discretization of Lagrangian equation on general polygonal meshes [74, 78, 114, 147, 165, 166]. We used the compatible mimetic finite discretizations [53, 54] on staggered mesh, which is historically close to [74, 78, 114] for ALE INC(ubator) (see previous chapter) and the CELIA team used the newly developed cell-centered discretizations based on Godunov approach [45, 49, 50].

ReALE specifics : rezone phase

The rezone phase of ReALE has to include both mesh movement and reconnection procedure. In paper [26] we used a set of generators and the machinery of Voronoi diagrams to do both mesh movement and mesh reconnection, see Fig.2.8 for an illustration of mesh reconnection *via* the Voronoi machinery. More precisely our rezone strategy consists of a special movement of generators. It is close to Lagrangian in some sense, but also include some smoothing procedure based on notion of centroidal Voronoi tesselation (CVT see the previous paragraph) [167]. The final position of a generator at time $t^{n+1} = t^n + \Delta t$ where Δt is the current time step is

$$\boldsymbol{G}_c^{n+1} = \boldsymbol{G}_c^{n+1,\text{lag}} + \omega_c \left(\boldsymbol{X}_c^{n+1} - \boldsymbol{G}_c^{n+1,\text{lag}} \right), \tag{2.4}$$

where the position vector of the generator of the Lagrangian cell Ω_c^n is denoted $\boldsymbol{G}_c^n$, and $\boldsymbol{G}_c^{n+1,\text{lag}}$ is a Lagrangian-like displacement of the generator obtained by :

$$\boldsymbol{G}_c^{n+1,\text{lag}} = \boldsymbol{G}_c^n + \Delta t \boldsymbol{U}_c, \tag{2.5}$$

where $\boldsymbol{U}_c$ is the "Lagrangian" velocity of the generator within the cell. This velocity is computed so that the generator remains located in the new Lagrangian cell. To this end we define this velocity to be the average of the velocities of the points of the cell, namely

$$\boldsymbol{U}_c = \frac{1}{\mid \mathcal{P}(c) \mid} \sum_{p \in \mathcal{P}(c)} \boldsymbol{U}_p^{n+\frac{1}{2}}.$$

Remind that $\mathcal{P}(c)$ denotes the set of points of the Lagrangian cell Ω_c and $\boldsymbol{U}_p^{n+\frac{1}{2}}$ is the time-centered velocity of point p between times t^n and t^{n+1} used to displace mesh point $\boldsymbol{X}_p^n$ to its new position

$\boldsymbol{X}_p^{n+1}$, that is to say following equation

$$\boldsymbol{X}_p^{n+1} = \boldsymbol{X}_p^n + \Delta t\, \boldsymbol{U}_p^{n+1/2}.$$

Finally $\omega_c \in [0;1]$ in (2.4) is a parameter that remains to be determined. The updated position of the generator is therefore defined by mean of a convex combination between the new Lagrangian-like position, $\boldsymbol{G}_c^{n+1,\text{lag}}$ and the centroid $\boldsymbol{X}_c^{n+1}$ of the Lagrangian cell. With this convex combination, the updated generator lies in between its Lagrangian position at time t^{n+1} and the centroid of the Lagrangian cell Ω_c^{n+1}. We note that for $\omega_c = 0$ we get a Lagrangian-like motion of the generator whereas for $\omega_c = 1$ we obtain a centroidal-like motion, which tends to produce a smoothed mesh. This latter case is equivalent to perform one Lloyd iteration [167, 168], see section 2.3 and Fig.2.9. We compute ω_c requiring that the generator displacement satisfies the principle of material frame indifference, that is for pure uniform translation or rotation we want ω_c to be zero. To this end, we construct ω_c using invariants of the right Cauchy-Green strain tensor associated to the Lagrangian cell Ω_c between times t^n and t^{n+1}. First, we define the deformation gradient tensor F

$$\mathsf{F} = \frac{\partial \boldsymbol{X}^{n+1}}{\partial \boldsymbol{X}^n},$$

where $\boldsymbol{X}^{n+1} = (X^{n+1}, Y^{n+1})^t$ denotes the vector position of a point at time t^{n+1} that was located at position $\boldsymbol{X}^n = (X^n, Y^n)^t$ at time t^n. The deformation gradient tensor is nothing but the Jacobian matrix of the map that connects the Lagrangian configurations of the flow at time t^n and t^{n+1}, in the two-dimensional case its components write

$$\mathsf{F} = \begin{pmatrix} \frac{\partial X^{n+1}}{\partial X^n} & \frac{\partial X^{n+1}}{\partial Y^n} \\ \frac{\partial Y^{n+1}}{\partial X^n} & \frac{\partial Y^{n+1}}{\partial Y^n} \end{pmatrix}.$$

The right Cauchy-Green strain tensor, C, is obtained by right-multiplying F by its transpose, i.e.

$$\mathsf{C} = \mathsf{F}^t\mathsf{F}.$$

In our case, C is a 2×2 symmetric positive definite tensor. This tensor reduces to the unitary tensor in case of uniform translation or rotation. It admits two positive eigenvalues, which are denoted λ_1 and λ_2 with the convention $\lambda_1 \leq \lambda_2$. These eigenvalues can be viewed as the rates of expansion in directions given by the eigenvectors during the transformation. To determine ω_c, we first construct the cell-averaged value of the deformation gradient tensor, F_c, and then the cell-averaged value of the Cauchy-Green tensor by setting $\mathsf{C}_c = \mathsf{F}_c^t\mathsf{F}_c$. Noticing that the two rows of the F matrix correspond to the gradient vectors of the X and Y coordinates, we can set $\mathsf{F}^t = \left[\boldsymbol{\nabla}_n X^{n+1}, \boldsymbol{\nabla}_n Y^{n+1}\right]$, where for any functions $\psi = \psi(\boldsymbol{X}^n)$, $\boldsymbol{\nabla}_n\psi = \left(\frac{\partial\psi}{\partial X^n}, \frac{\partial\psi}{\partial Y^n}\right)^t$. With these notations, let us define the cell-averaged value of the gradient of the ψ function over the Lagrangian cell Ω_c^n

$$(\boldsymbol{\nabla}_n\psi)_c = \frac{1}{|\,\Omega_c^n\,|}\int_{\Omega_c^n} \boldsymbol{\nabla}_n\psi \mathrm{d}V = \frac{1}{|\,\Omega_c^n\,|}\int_{\partial\Omega_c^n} \psi\boldsymbol{N}\mathrm{d}S.$$

Here, we have used the Green formula and $\boldsymbol{N}$ is the unit outward normal to the boundary of the cell Ω_c^n refered as to $\partial\Omega_c^n$. Assuming that this cell is a polygon and using the trapezoidal rule we obtain the following approximation for the previous integral

$$(\boldsymbol{\nabla}_n\psi)_c = \frac{1}{|\,\Omega_c^n\,|}\sum_{p=1}^{|\mathcal{P}(c)|}\frac{1}{2}\left(\psi_p^n + \psi_{p+1}^n\right)L_{p,p+1}^n\boldsymbol{N}_{p,p+1}^n, \tag{2.6}$$

where ψ_p^n is the value of ψ evaluated at point $\boldsymbol{X}_p^n$ and $L_{p,p+1}^n \boldsymbol{N}_{p,p+1}^n$ is the unit outward normal to the edge $[\boldsymbol{X}_p^n, \boldsymbol{X}_p^{n+1}]$. Applying (2.6) to $\psi = X^{n+1}$ and $\psi = Y^{n+1}$ we get a cell-averaged expression of the gradient tensor F and then deduce from it the cell-averaged value of the right Cauchy-Green tensor C_c. Knowing this symmetric positive definite tensor in each cell, we compute its real positive eigenvalues $\lambda_{1,c}$, $\lambda_{2,c}$. We finally define the parameter ω_c as follows

$$\omega_c = f(\lambda_1, \lambda_2) = \frac{1 - \alpha_c}{1 - \alpha_{\min}}, \tag{2.7}$$

where $\alpha_c = \frac{\lambda_{1,c}}{\lambda_{2,c}}$ and $\alpha_{\min} = \min_c \alpha_c$. We emphasize the fact that for uniform translation or rotation $\lambda_{1,c} = \lambda_{2,c} = 1$ and $\omega_c = 0$, therefore the motion of the generator is quasi Lagrangian and we fulfill the material frame indifference requirement. For other cases, ω_c smoothly varies between 0 and 1. The behavior of ω_c parameter has been tested with ReALE on Sedov problem [169] in Fig. 2.10 for three different generator displacement strategies : $\omega_c = 0$ to get a Lagrangian-like motion, $\omega_c = 1$ to obtain a centroidal-like motion, and ω_c defined by (2.7) using the strategy previously described. While the quasi-Lagrangian motion produces irregular mesh and results the centroidal-like motion generates an ultra regular and smoothed solution. However the mesh is so smoothed that the flow features (compressed cells after the shock wave as instance) have been litteraly washed out. Also notice that the part of the mesh at radius greater than 1 has not been attained by any wave at time $t = 1$. Consequently one expects the rezone/reconnection strategy to spare this region from smoothing. This is clearly not the case for the centroidal motion[7]. Conversely the deformation-tensor based generator motion furnishes a regular mesh and a more accurate solution than the quasi-centroidal motion. Above and over the generators have followed the fluid motion in an almost Lagrangian fashion without washing out the flow feature. Finally for the region beyond radius one, the generator velocity being zero, the original mesh is maintained untouched. This is due to the fact that $\lambda_{1,c} = \lambda_{2,c} = 1$ then $\omega_c = 0$ leading to a Lagrangian motion of generators the velocity of which is **0** (no wave has attained this region yet).

ReALE specifics : remap phase

In the remapping phase, the Lagrangian solution is transferred (conservatively interpolated) onto the rezoned mesh. Lagrangian mesh is the result of one time step Lagrangian movement of the Voronoi mesh corresponding to the distribution of the generators at the previous time step. The new rezoned mesh is the Voronoi mesh corresponding to the positions of generators created by the rezone phase. During the rezone phase generators are moved in an "almost" Lagrangian way and because Voronoi cells are changing their shape continuously with respect to positions of the generators ; rezoned and Lagrangian meshes are "close". However, in general, the connectivity of the Lagrangian and rezoned mesh are different. Consequently remapping methods have to be able to conservatively transfer flow parameters from one polygonal mesh to another. We use an exact intersection (overlay) based remap [129, 147, 127, 128], see the **Remap phase** paragraph of the section HISTORY AND PRESENTATION in this chapter. However, one could take advantage of how Lagrangian and rezoned meshes are constructed and design more efficient methods [170, 171, 172, 173] which has been done for CHIC code as instance.

7. Modifying untouched regions is one unacceptable feature of many rezoning strategies. Intrinsically, if the initial mesh is not optimal with respect to the underlying metric of the rezoner, sooner or later the algorithm will try to "optimize" the mesh in places where the user does not expect it to happen. Some solutions using triggers and filters can delay this feature but once nodes are marked for rezoning then the flaw will emerge.

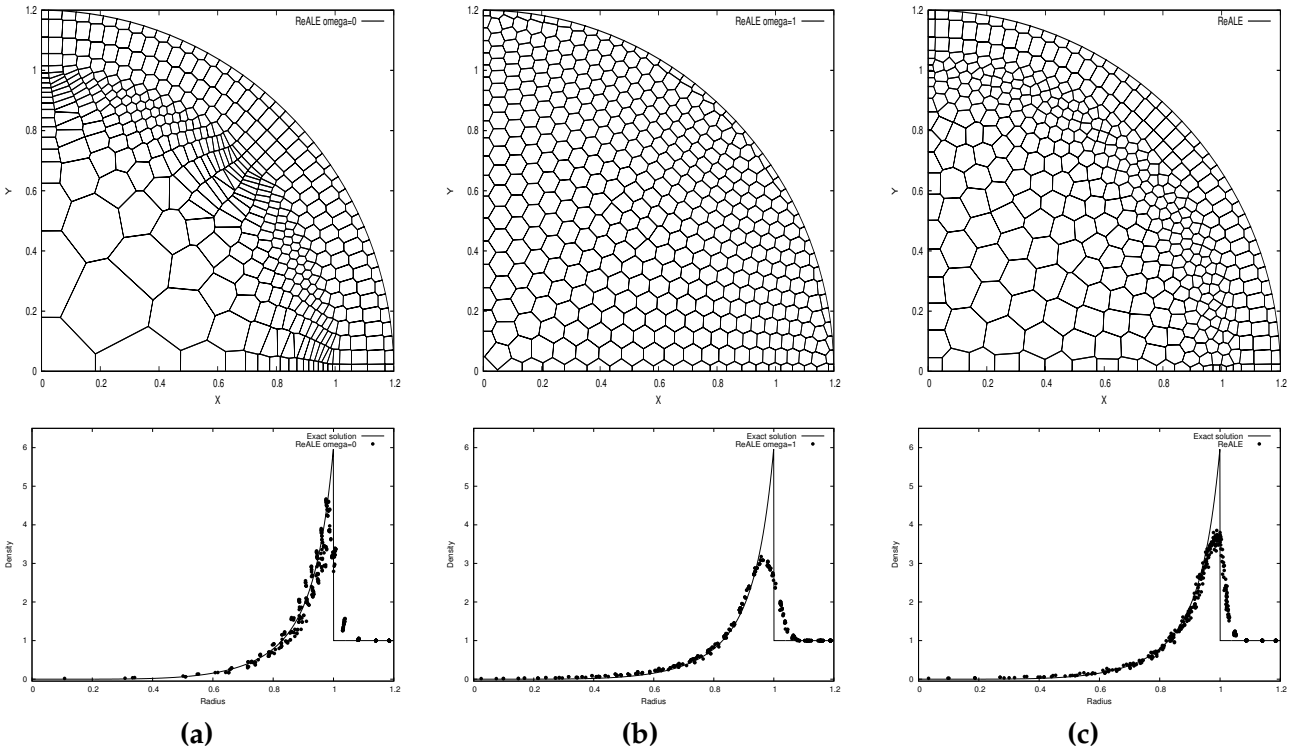


Figure 2.10 – *Results from [26]. Sedov problem at time $t = 1.0$ for different generator displacement strategies— Staggered ALE INC(ubator) code — Top panels : mesh. Bottom panels : density as a function of radius for all cells vs the exact solution (line) — **(a)** Quasi-Lagrangian generator motion $\omega_c = 0$ — **(b)** Quasi-centroidal generator motion $\omega_c = 1$ — **(c)** Deformation-tensor based generator motion $\omega_c = f(\lambda_1, \lambda_2)$.*

ReALE results

In paper [26] we have performed a set of test cases to show the behaviors of the ReALE technique with ALE INC(ubator) and CHIC codes. The Sedov test case has been used as a sanity check as almost no reconnection occurs and it has been shown that the reconnection treatment does not pollute the computation.
Next we have considered a two-material Riemann problem in 2D, the so-called triple point problem depicted in Fig. 2.11.

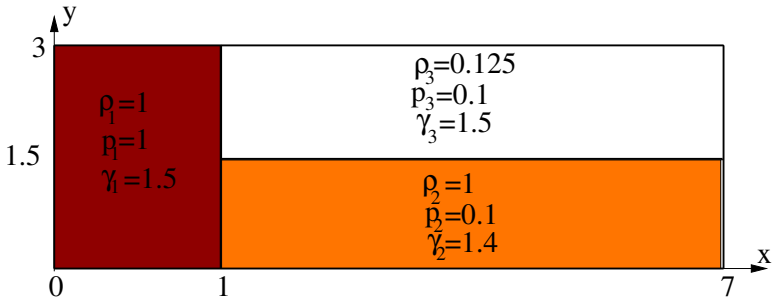


FIGURE 2.11 – *Figure from paper [26]. Triple point problem initialization.*

Due to the difference of density and gamma, two shocks in top and bottom domains propagate with different speeds. This creates a shear along initial horizontal contact discontinuity and a vorticity formation. Capturing the vorticity is the difficult part of such simulation when standard Lagrangian or ALE methods are used. In the following figure one reproduces the triple point problem results for Lagrangian, ALE and ReALE. These figures are devoted to visually measure "how much Lagrangian a method is". In such figures one displays the mesh where the cells have been colored according to in which domain they were initially located (white, red or orange). This way of presenting the results allows to observe if each cell move in an almost Lagrangian fashion. (In fact in this test case the orange cells must roll over with the white ones due to the vortex motion.)
In Fig.2.12 the first panel (time $t = 1.67$) corresponds, more or less, to the time after which any Lagrangian scheme inexorably fails due to mesh tangling. In the second and third panels of Fig.2.12 are displayed the ALE/ReALE results at final time $t = 5$. The second panel clearly shows the stagnation of the ALE mesh. During the Lagrangian stage of ALE the mesh is trying to follow the flow, but the development vorticity eventually leads to a tangled mesh. Next the rezoning step slightly relaxes the mesh and as such acts against the mesh motion. On the next time step of the Lagrangian phase, the mesh is trying to follow vorticity development and, again, is approaching a tangling situation. The fixed connectivity ALE mesh is the result of these competing processes : ultimately the mesh stagnates. ALE regime freezes the mesh to a position and the computation continues in an almost Eulerian fashion (as Lagrange+Remap because the rezone phase systematically backs up the Lagrangian t^{n+1} mesh onto the previous t^n Lagrangian mesh).
ReALE regime, third panel of Fig.2.12 nicely follows the vortex as cells are carried within the vortex in an almost Lagrangian fashion. As a consequence ReALE has a better accuracy as Voronoi cells are able to roll up and reconnect to new neighbors when necessary. This attests the ability of ReALE to properly follow such type of fluid motion.

In Fig. 2.13 one reproduces on the left the mesh and specific internal energy (left panels) and, the mesh where the cells have been colored according to the domain they were initially located (middle panels). On right panels we display the value of ω_c which shows that the regions where $\omega_c > 0.7$ are

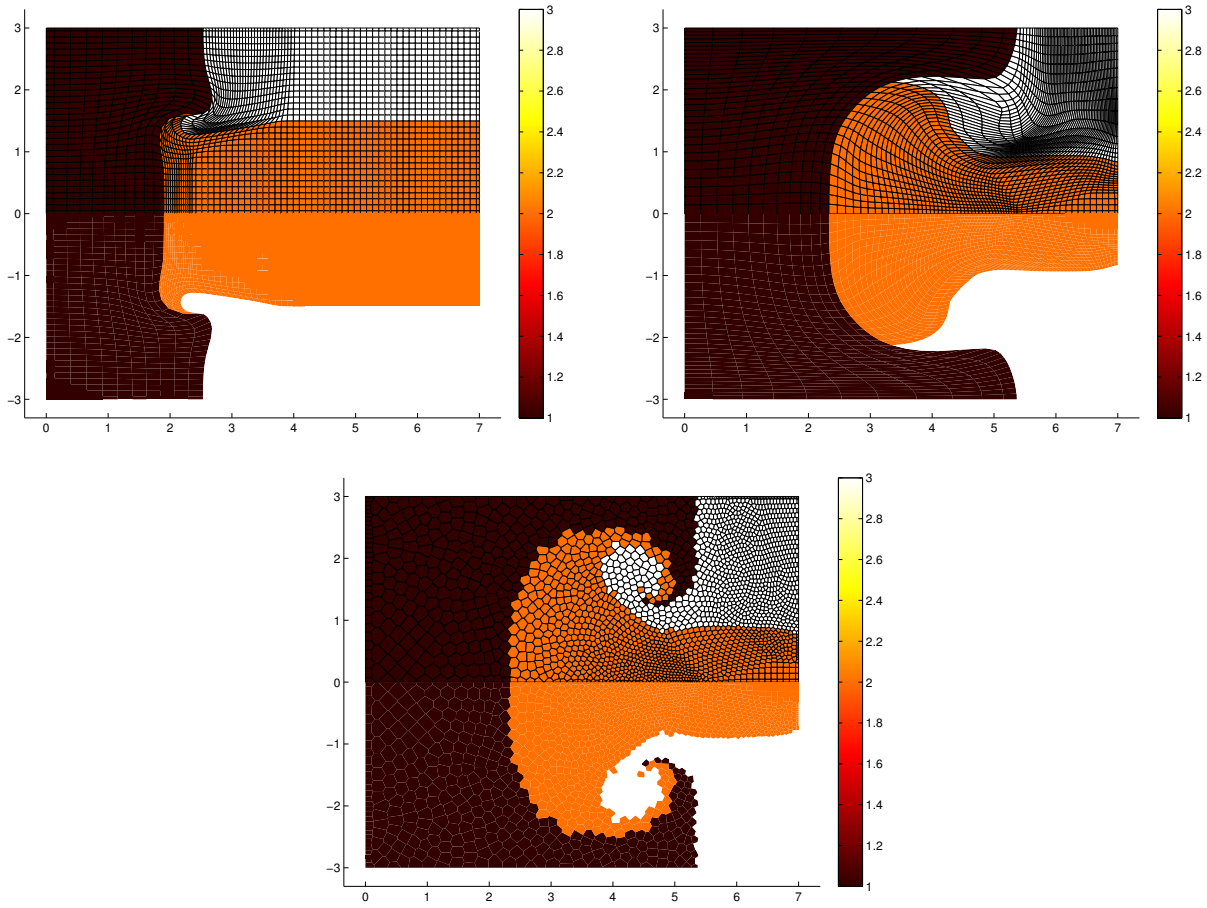


Figure 2.12 – *Results from paper [26]. Triple point problem results for Lagrangian, ALE and ReALE. The colors corresponds to the initial position of the cell within the three materials in Fig. 2.11 (Results from CHIC code.)*

clearly the regions where directional deformation occurs and where the mesh must be smoothed by the CVT technique.

The last test case proposed in [26] is the Rayleigh-Taylor instability. While fixed ALE results present the mesh stagnation as for the triple point problem (ultimately fixed ALE runs in a Lagrangian+Remap regime), the ReALE approach is able to follow the complex motion of the fluids see Fig. 2.14. In this figure one only reproduces the results at later time $t = 15$ when the top heavy fluid has reached the bottom and moves upward. Due to the vorticity, mixing between the two fluids does occur. (Usually with fixed ALE one shows the final time $t = 7$ or 8 before stagnation occurs. In paper [26] we have also presented intermediate times from $t = 8$ to $t = 15$ showing the efficiency of ReALE). The density, mesh, vorticity and cells colored by initial location are shown.

The paper [27] entitled *"ReALE : a Reconnection Arbitrary-Lagrangian-Eulerian method in cylindrical geometry"* has extended the ReALE concept to cylindrical geometry. In this paper we have shown that the whole ReALE concept does adapt to an already existing ALE code in cylindrical geometry (CHIC code in this case). Several test cases in cylindrical geometry have been run to assess this point, Sedov blastwave, an Helium/bubble shock interaction test (compared with experimental

results), and the rise of a light bubble under gravity. In Fig. 2.15 one reproduces the results on the Helium/bubble shock interaction test case for different times (the test case is depicted on the very next figure). In Fig. 2.16 one reproduces the rise of a light bubble under gravity (density, vorticity, velocity vector and mesh are displayed). The velocity vectors clearly show that due to high vorticity mesh reconnection occurs. Details and comments are to be found in paper [27].

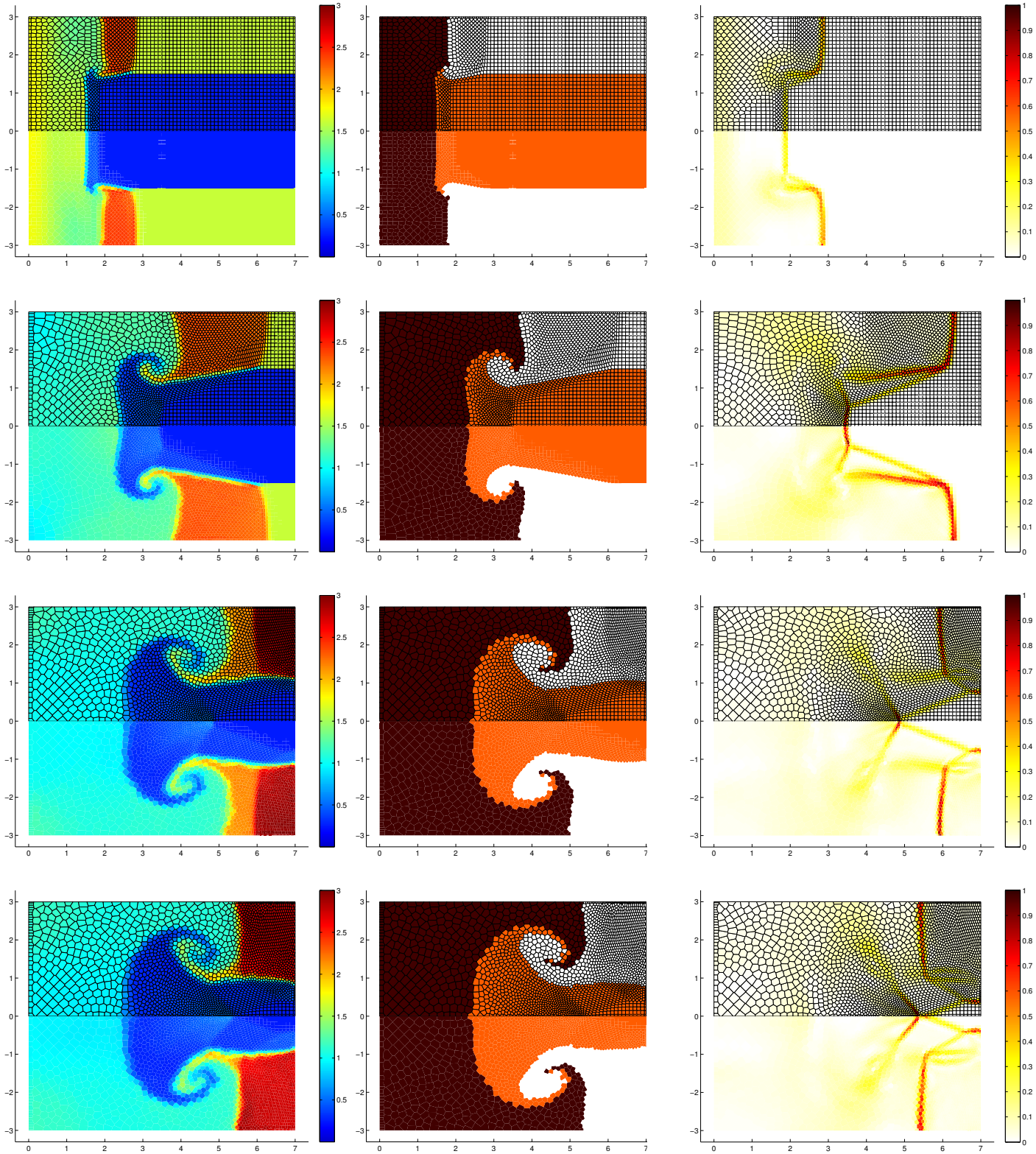


Figure 2.13 – *Numerical results from paper [26]. Triple point problem at several times for ReALE for internal energy and mesh (left) and cell color corresponds to the initial position in the domain (middle) and factor ω_c (right). From top to bottom : times $t = 1, 3, 4.5, 5$. $t = 1$ corresponds to the failure time for the Lagrangian version of the code, $t = 4.5$ roughly corresponds to the mesh stagnation time for the fixed ALE version although the ALE code runs until completion.*

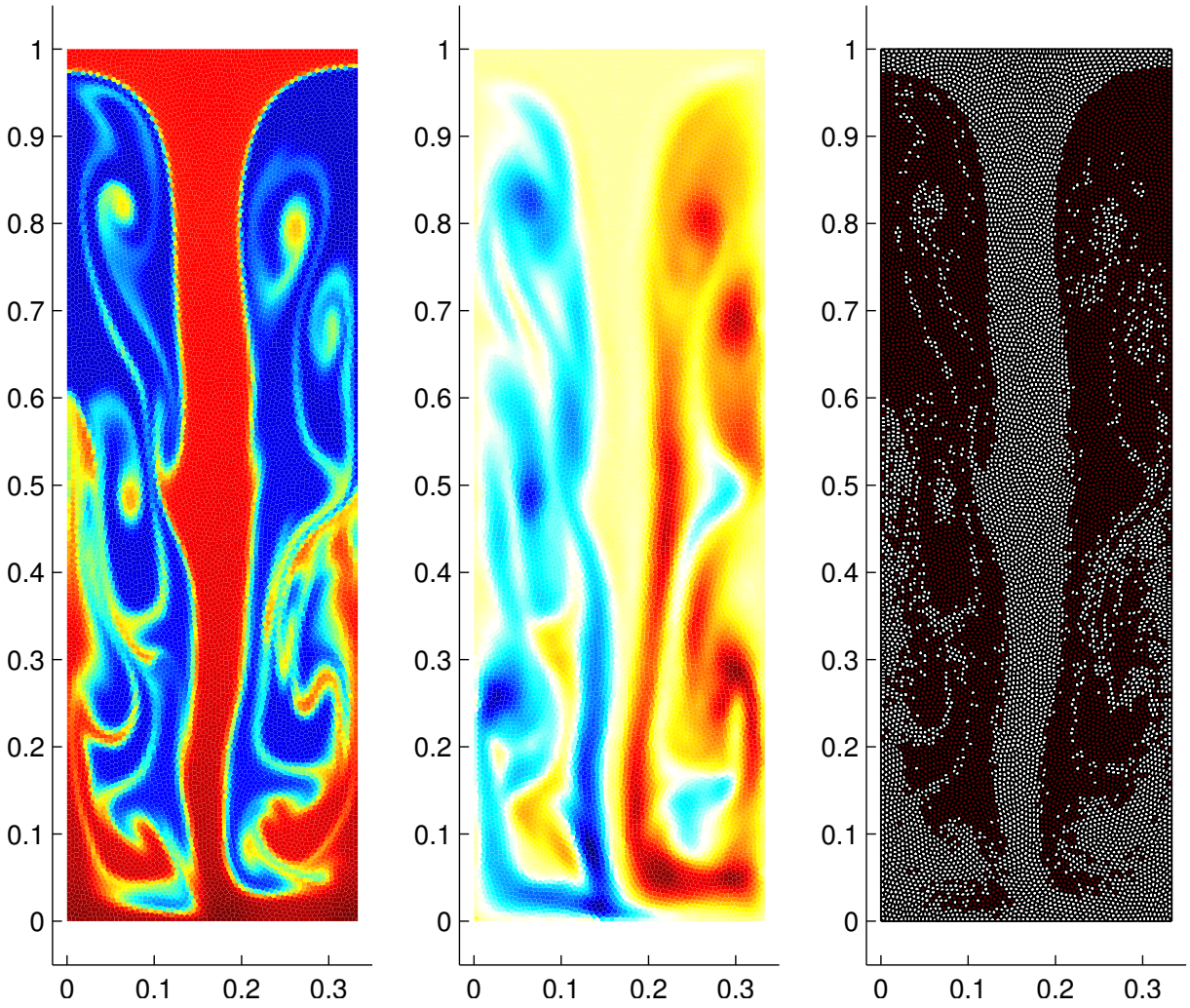


FIGURE 2.14 – *Numerical results from paper [26]. Rayleigh-Taylor instability with CHIC code — ReALE with* 66×200 *generators at* $t = 15$ *— Density, vorticity and, mesh and cells colored from their initial domain (red : from top-heavy fluid, white : from bottom-light fluid). Vorticity scale is from* -11.78 *to* 9.63 *(blue for the minimal value to red for the maximal one).*

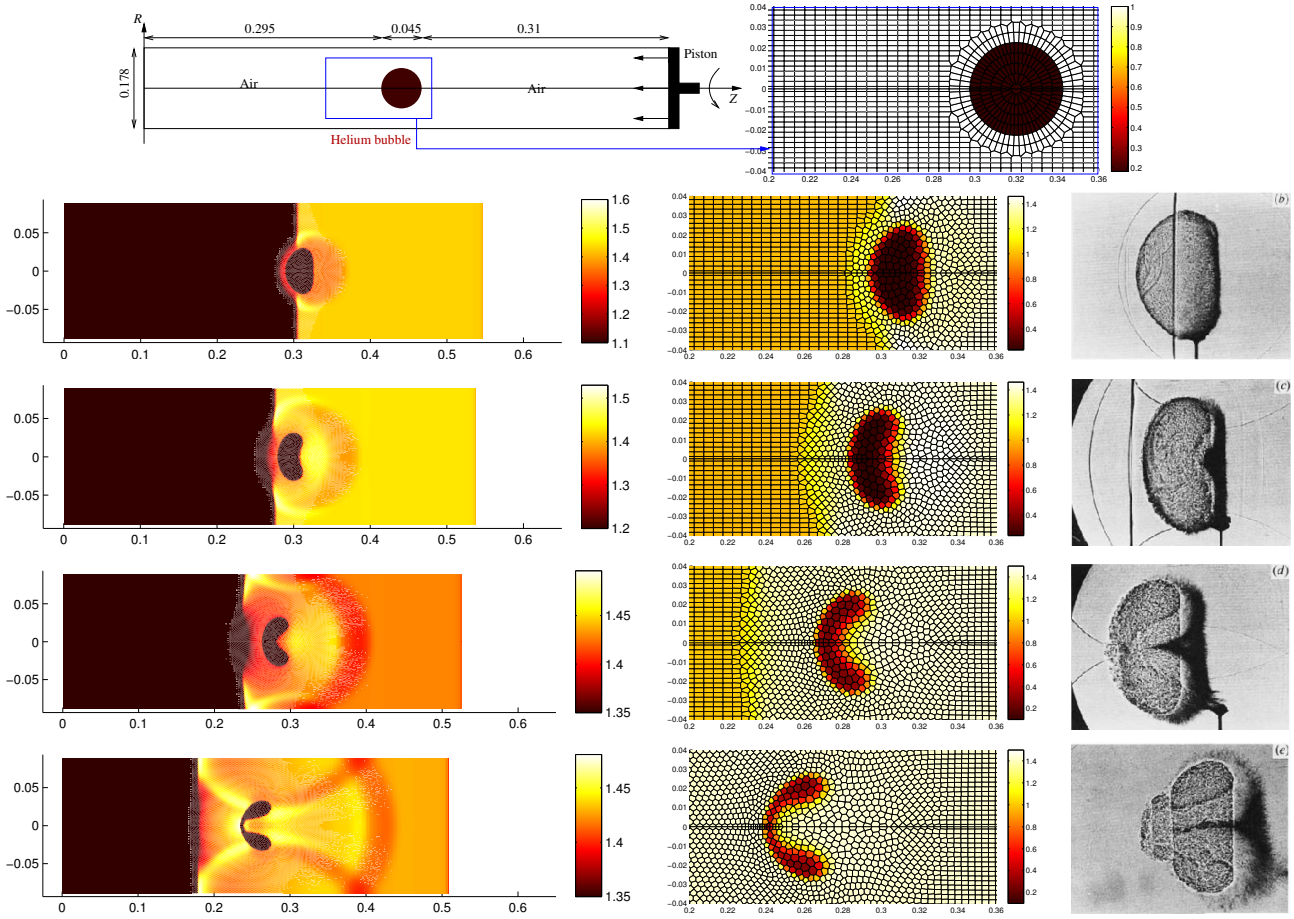


Figure 2.15 – *Numerical results from paper [27]. ReALE in cylindrical geometry results on the $M = 1.25$ shock interaction with a spherical helium bubble — Left : density waves in the domain. Middle : zoom on mesh and density (color) around the bubble. Right : Schlieren graphics (experimental results from [174]) — From top to bottom : $t_b = t_i + 82\ 10^{-6}$, $t_c = t_i + 145\ 10^{-6}$, $t_d = t_i + 223\ 10^{-6}$, $t_e = t_i + 1007\ 10^{-6}$ where $t_i = 657.463\ 10^{-6}$ is the time of the shock/bubble interaction.*

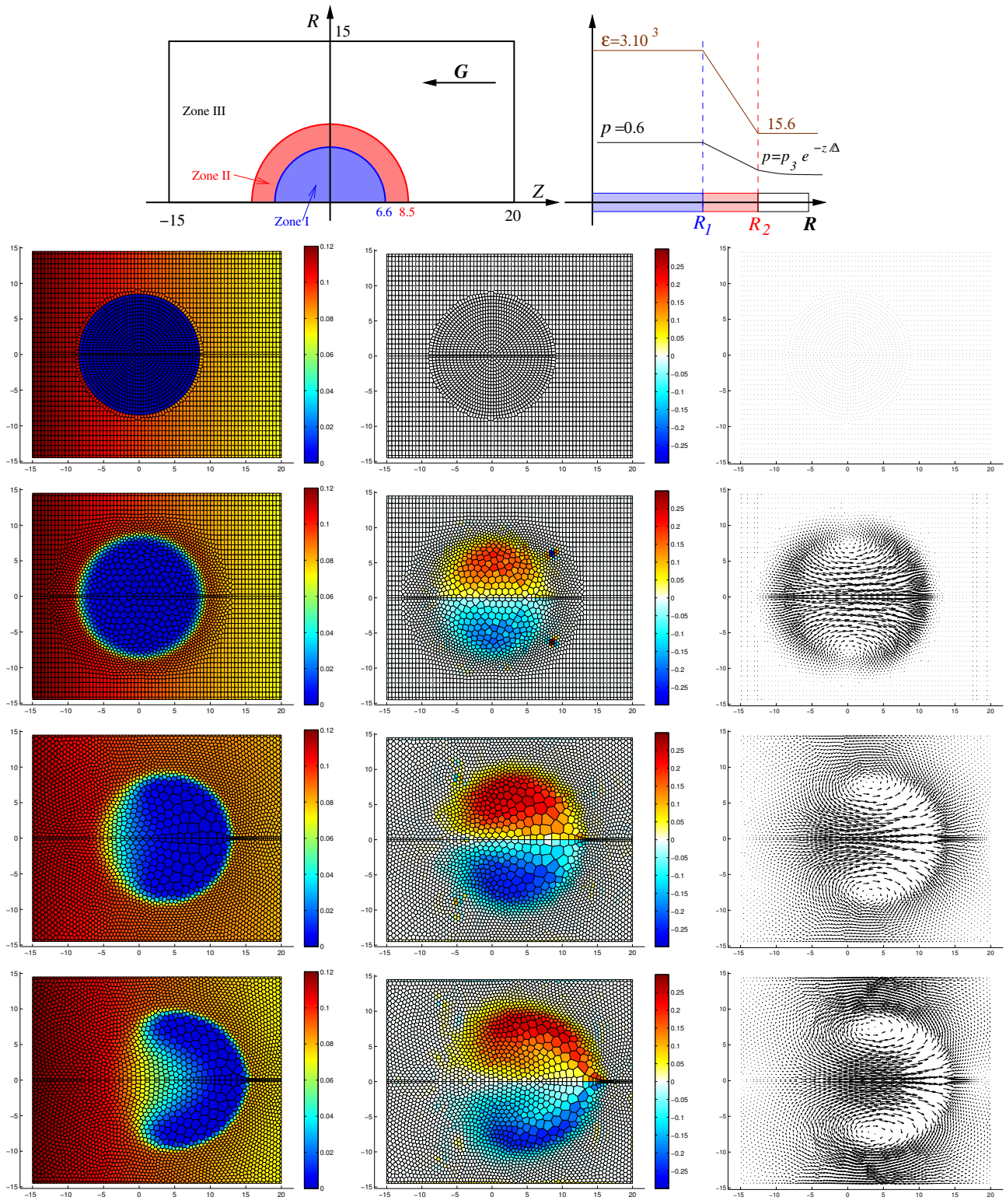


FIGURE 2.16 – *Numerical results from paper [27]. Top sketch : rise of a light bubble under gravity, the light bubble (Zone I) has a radius of R_1 and a transition layer is initialized between R_1 and R_2 (Zone II). The rest of the domain $\boldsymbol{R} > R_2$ is some air at rest (Zone III) where $\boldsymbol{R} = \sqrt{R^2 + Z^2}$. Gravity is oriented in the Z direction. The pressure and internal energy profiles are sketched in the right panel — 3 × 3 panels : numerical results from ReALE in cylindrical geometry — Left column : density and mesh — Middle column : vorticity and mesh — Right column : velocity vectors — Top-bottom : time $t_0 = 0$, $t_1 = 1$, $t_2 = 8$, $t_4 = 14$.*

2.4 ALE codes comparison

In 2007-2008 with Richard Liska and Pavel Váchal (from the Czech connection in Prague) and J. Breil, S. Galera and P.-H. Maire (CELIA, university of Bordeaux) we have investigated the behaviors of our three ALE codes. At this time the codes' description was

CHIC code uses the cell-centered Lagrangian method, condition number rezoning and cell-centered swept region remapping.

PALE standing for Prague ALE. A staggered Lagrangian compatible Lagrangian scheme with Winslow rezoner for this study [8] and partial subcell swept region remapper.

ALE INC(ubator) code uses the staggered Lagrangian method, the Reference Jacobian rezoning and a full subcell swept region remapping.

The goal was to run these codes in their nominal configuration [9] and compare their general behaviors.

As an illustration we reproduce here the Sedov blast wave problem [169] in Cartesian 2D geometry. In the paper we have also run sanity checks (1D Sod shock tube) and shock/bubble interaction test cases. In Fig.2.17 are presented the results of the three ALE codes and an Eulerian scheme based on Lax and Liu scheme [175],[176] for the density variable for the final time $t_{final} = 1$. The top/middle rows correspond to a 30×30 initial Cartesian mesh, the bottom row corresponds to a 60×60 mesh. The top row presents the meshes colored by density. We plot in middle and bottom rows the cell density (for all cells) as a function of the cell radius *versus* the exact solution which consists of a shock wave located at $r = 1$ at $t_{final} = 1$ and an exponential type of density function before the diverging shock (travelling then towards to the right of the pictures) the peak of which is at $\rho_{max} = 6$. The three Lagrangian codes produce decent results, the cell-centered being the most accurate one (see the maximum peak as instance). The two staggered Lagrangian codes perform alike and the Eulerian scheme is the least performing of the four methods. Notice, that all Lagrangian methods are performing for this test much better than the Eulerian method as the moving Lagrangian mesh concentrates more cells close to the circular shock.

8. PALE also enjoys Reference Jacobian rezoning, condition number smoothing.
9. In other words not trying to tweak all fixes and parameters that may help.

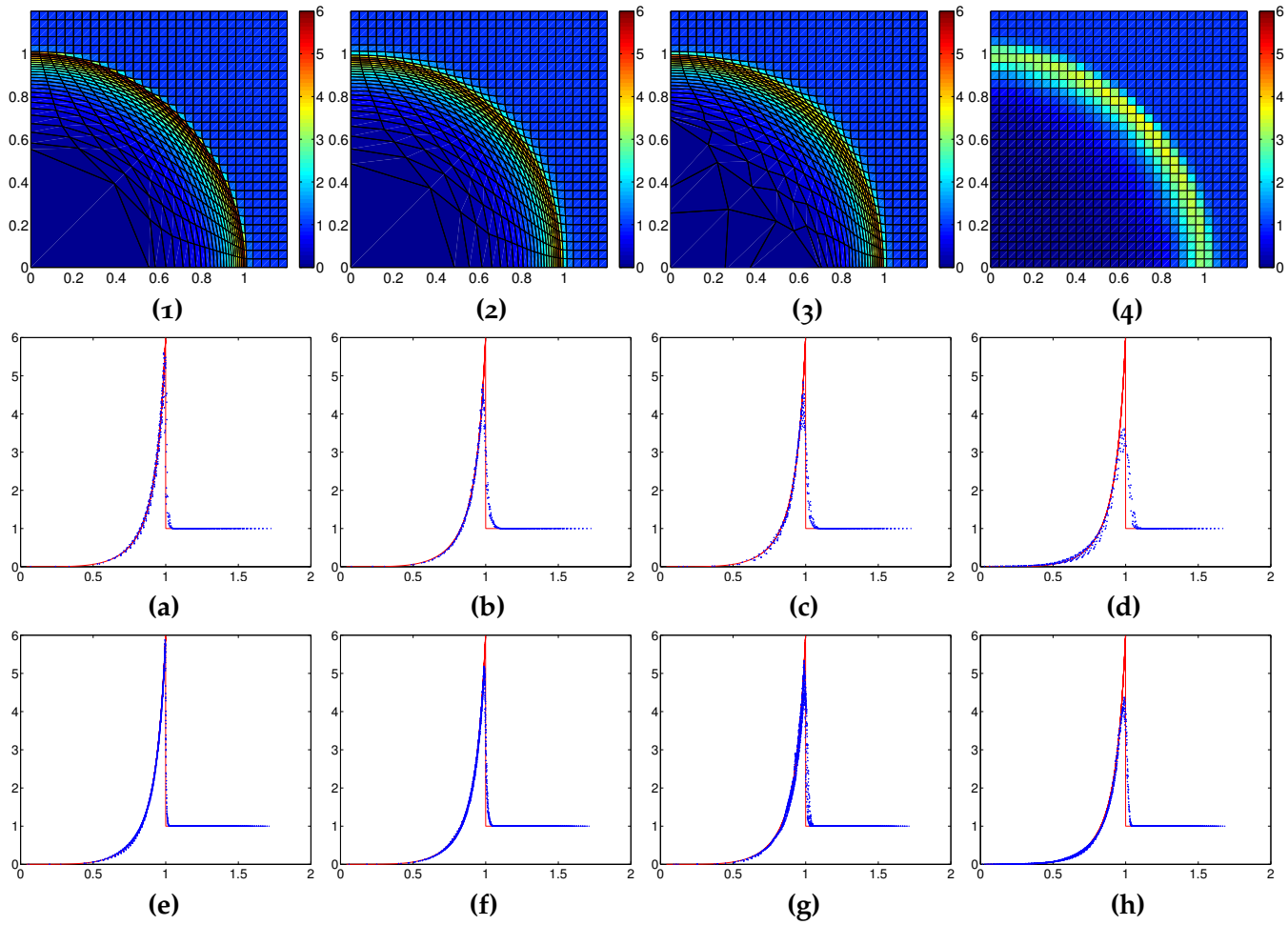


FIGURE 2.17 – *Sedov problem — Density at* $t = 1$ — **(a)** *: CHIC* 30×30 *cells* — **(b)** *: ALE INC* 30×30 *cells* — **(c)** *: PALE* 30×30 *cells* — **(d)** *: Eulerian scheme* 30×30 *cells* — **(e)** *: CHIC* 60×60 *cells* — **(f)** *: ALE INC* 60×60 *cells* — **(g)** *: PALE* 60×60 *cells* — **(h)** *: Eulerian scheme* 60×60 *cells.*

2.5 Multi-material treatment

Dealing with more than one material requires several new developements in a staggered ALE code.

The staggered Lagrangian scheme should be extended to treat mixed cells. Several techniques are well documented, see as instance [177]. One simple technique is based on concentration equations. The multi-material flow is considered as a multi-component mixture of miscible fluids. Each fluid is characterized by its concentration being a passive scalar used to define the location of an interface within mixed cells. The equation of state of the mixture of materials must be determined usually by an *ad hoc* assumption like pressure/temperature equilibrium. Another approach based on volume of fluid (VOF) [178] method which introduces a Lagrangian tracking of material interfaces is often prefered when no mixing between materials is expected. Consequently mixed cells are present in the computational domain. These consists of the collocation of two or more materials within the same cell. The time evolution of a mixed cell is obtained using a closure model that computes a mixed thermodynamic state function of the thermodynamic states of each material taking into account the volume fractions (i.e. the rates of presence). Several closure models have been developed as instance pressure/temperature equilibrium or relaxation [179, 180, 181, 182, 183], subcell dynamics formulations [184] and other comparable techniques [185, 186, 187, 188]. As already seen each mixed cell must be split into pure materials knowing the volume fractions fields. Generally a reconstruction of the interface between two materials in mixed cell is performed using a straight segment following the well-known Youngs' method [189, 190] (or also known after [191] as PLIC *Piecewise Linear Interface Calculation* the extension of SLIC method *Simple Line Interface Calculation* [192]).

Some requirements could also be added to the rezone method, as instance some interfaces between materials can be preserved during the rezone phase. However most of the time the rezoning phase is kept as it is.

The remapping phase should adapt to the mixing model employed in the Lagrangian phase. In the case of concentration equations each concentration must be consistently remapped as instance by using some scaling as in [177]. On the other hand in the case of a VOF method remapping is made material by material and usually demands the use of an exact intersection between the rezoned mesh and the Lagrangian mesh with reconstructed interfaces in mixed cells.

Due to their strict conservation of materials, volume-of-fluid (VOF) methods using interface reconstruction are widely used. However the effective management and capture of interfaces is essential for accurate and reliable simulation of multi-material and multi-phase flows. As already said VOF methods do not explicitly track the interface between materials, but rather advect volume fractions which prescribe the material composition of each cell of the mesh. When the interface between materials is needed, the interface is recreated based on the material volume fraction in the cell and its surrounding cells [52, 191, 193, 194]. A common problem impacting these reconstruction methods is their dependence on a specified material ordering, i.e. if more than two materials are present in a cell, the reconstruction may depend on the sequence in which the materials are processed. As an illustration we present in Fig. 2.18 the nested dissections obtained by Youngs' method in one mixed cell with three materials : two different orders of treatment produce different final interfaces (see panels (d) and (h)). This is undesirable as it may improperly locate materials within the cell. Moreover this may result in material being incorrectly fluxed into neighbouring cells.
In 2005 with some colleagues from the Los Alamos National Laboratory we have put some effort to develop an order-independent interface reconstruction method for general grids.

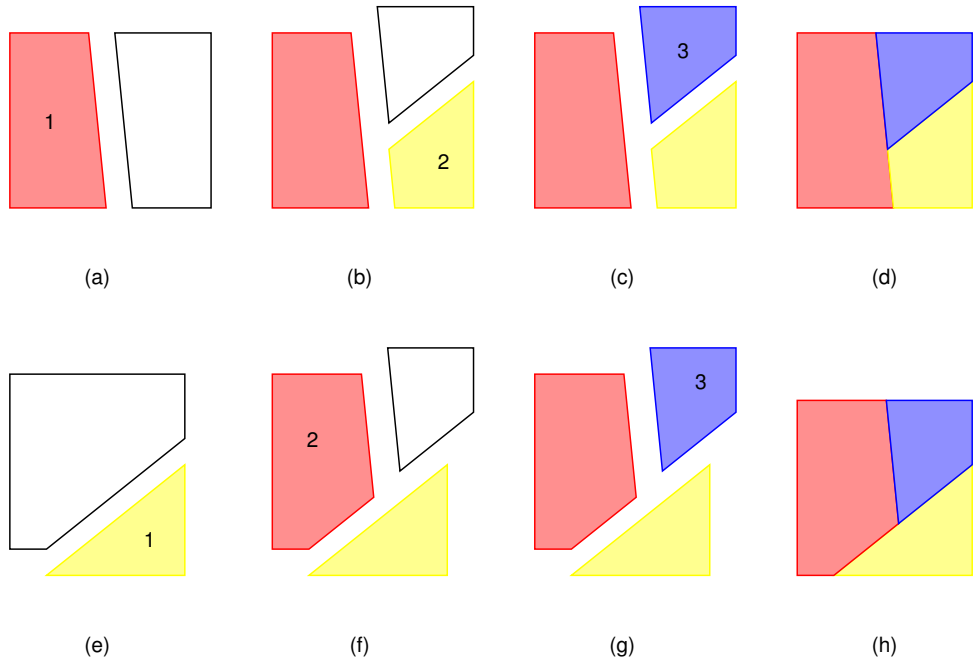


FIGURE 2.18 – *Nested dissection interface reconstruction for three materials (a) the first material is removed leaving a smaller available polygon, (b) the second material is removed from the available polygon, (c) the remaining available polygon is assigned to material 3, (d) the resulting partitioning of the computational cell. (e)-(g) show the same procedure but the materials are processed in a different order leading to a different reconstruction (h).*

2.5.1 Interface reconstruction techniques using Power Diagram

In a set of two articles [13, 17] respectively entitled *Material order independent interface reconstruction using power diagrams* and *A second-order accurate material-order-independent interface reconstruction technique for multi-material flow simulations* with my colleagues from Los Alamos S.P. Schofield, R.V. Garimella and M.M. Francois we have developed a method that can reconstruct a multi-material interface with no dependence on material ordering. The method is very general : it works on unstructured grids, accommodates an arbitrary number of materials and extends naturally to three dimensions. The method does not assume a topology for the material regions, i.e. a layer structure or triple point configuration. Furthermore, all of the material regions created are convex. Notice that at the same time V. Dyadechko with M. Shashkov from Los Alamos have developed an alternative and concurrent technique called Moment of Fluid method (MOF) [195]. This technique has been tested in 3D [196] and in [197] a comparative study between different methods including the power diagram method from [13, 17] and MOF has been carried out.

The method proposed in [13] consists of two steps :

First, the relative location of the materials within the mixed cell is approximated ;

Second, a power diagram is used to split the mixed cell into pure polygons the surfaces of which fulfill the volume fractions of the materials.

Relative location of the materials within the mixed cell. The method utilizes a particle attraction model or approximate centroid calculation to infer the relative location of the materials in the cell. In the first step of the method, a number of particles representing the materials are placed in multi-

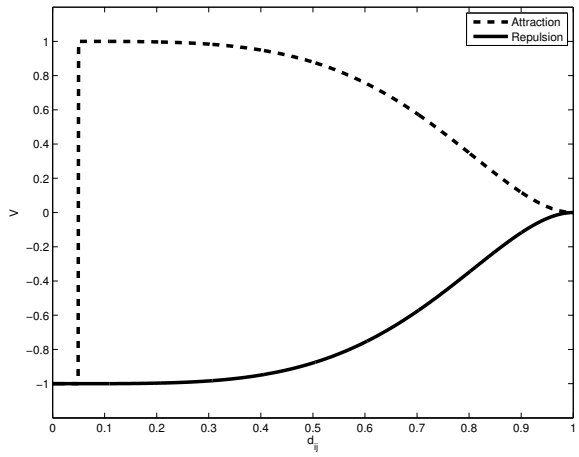


Figure 2.19 – *Particle attraction and repulsion "forces" used in the model.*

material cells and any pure or mixed neighboring cells. A particle, P_i, has a position, $\mathbf{x}_i$, velocity $\mathbf{V}_i = \frac{d\mathbf{x}_i}{dt}$ and material $m(i)$, and is constrained to stay within the cell in which it is initially placed. Taking inspiration from molecular dynamics [198, 199] and smoothed particle hydrodynamics [200, 201], we evolve the particle positions according to "forces" based on the particles' relative locations and materials. The positions of the particles are updated through time integration of a set of ordinary differential equations,

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{V}_i \tag{2.8}$$

$$\mathbf{V}_i = \sum_{j:\, m(j)=m(i)} \mathbf{V}_{\mathbf{att}}(\mathbf{x}_i, \mathbf{x}_j) + \sum_{j:\, m(j)\neq m(i)} \mathbf{V}_{\mathbf{rep}}(\mathbf{x}_i, \mathbf{x}_j)$$

where $\mathbf{V}_{\mathbf{att}}$ and $\mathbf{V}_{\mathbf{rep}}$ are the prescribed attractive and repulsive "forces" in the direction $\mathbf{x}_j - \mathbf{x}_i$. Particles of the same material attract each other until they are very close, at which point they start to repel each other. Particles of different materials repel each other. In our tests, the particles start at random locations within their cell, but they can be initialized using other means such as their relative locations in a cell at a previous time step.
The particle-particle "forces" (plotted in Fig. 2.19) are prescribed as

$$\mathbf{V}_{\mathbf{att}}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} -1, & d_{ij} < \delta \\ 1 - 2d_{ij}^4 + d_{ij}^8, & \delta \leq d_{ij} \leq 1 \\ 0, & d_{ij} > 1 \end{cases} \tag{2.9}$$

$$\mathbf{V}_{\mathbf{rep}}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} -(1 - 2d_{ij}^4 + d_{ij}^8), & d_{ij} \leq 1 \\ 0, & d_{ij} > 1 \end{cases} \tag{2.10}$$

$$\tag{2.11}$$

where $d_{ij} = \frac{||\mathbf{x}_i - \mathbf{x}_j||}{2.5h}$ is the distance between points scaled by an interaction distance, taken to be 2.5 times the characteristic mesh size h, and $\delta = 0.05$. Unlike a traditional mechanical model, the "forces" here actually prescribe the instantaneous velocities of the particles. In a cell, $\mathcal{C}_i$, the number of particles, $N(\mathcal{C}_i)$ is

$$N(\mathcal{C}_i) = \left\lfloor N_p \times \frac{\|\mathcal{C}_i\|}{A_0} \right\rfloor \tag{2.12}$$

where N_p is a prescribed constant (usually around 30), $\|\mathcal{C}_i\|$ is the area of the cell, A_0 is a reference cell area for the grid (for example on a uniform Cartesian grid, $A_0 = h^2$ where h is the grid spacing) and $\lfloor a \rfloor$ is the floor function giving the greatest integer less than or equal to a. Each particle has a designated material type, corresponding to a material present in the cell. Each material that is present in the cell is represented by the same number of particles, $N(\mathcal{C}_i)/N_m^i$, where N_m^i is the number of materials present in the cell. We found that making the number of particles representing each material proportional to the volume fraction of the material often leads to unsatisfactory results. If the volume fraction is small, the material will be represented by only a few particles, which are not sufficient to provide a reliable estimate of the location of the material within the cell. In addition, we found that for unstructured, general polygonal grids, making the number of particles proportional to the area of the cell was important. Otherwise, the particles tend to cluster in regions of the mesh with a concentration of smaller cells.

Once the particles are distributed, the particle model is run. Since the model prescribes instantaneous velocities and not true forces, the particles may remain in perpetual motion unless the system is forced to cool. The velocities determine the kinetic energy of the system which in turn defines the temperature. The velocity of each particle is rescaled at each time step to decrease the kinetic energy of the system and force the particles to settle into a final configuration. At time step n in the time integration of Equation 2.8, the kinetic energy of all the particles is

$$KE^n = \sum_i \frac{1}{2} ||\mathbf{V}_i||^2 \tag{2.13}$$

After the system is sufficiently agitated, typically after 5 to 10 time steps, we force the kinetic energy to decrease as,

$$KE^{n+1} \leq \alpha KE^n \tag{2.14}$$

where $0 < \alpha < 1$. In practice, α is set to be $0.7 - 0.9$. If $KE^{n+1} \geq KE^n$, all the particle velocities are scaled as

$$\mathbf{V}_i' = \sqrt{\alpha \frac{KE^n}{KE^{n+1}}} \mathbf{V}_i. \tag{2.15}$$

To speed up the calculation, we use a variable time step with a new Δt calculated after each time step as

$$\Delta t = \frac{0.1}{2||\mathbf{V}_{max}||}, \tag{2.16}$$

where $||\mathbf{V}_{max}|| = \max_i ||\mathbf{V}_i||$ where $\mathbf{V}_i$ is as defined in Equation 2.8. The positions are then updated as

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \Delta t \ \mathbf{V}_i'. \tag{2.17}$$

If a particle goes outside the cell, it is placed back in the cell by repositioning it to the center of the triangle formed by the old position, the new position, and the center of the cell. If that fails, the particle is kept in its old position.

The particles are allowed to evolve for a number of time steps until the average kinetic energy of each particle has dropped below a specified stopping criteria. The particle model exhibits rapid convergence to the particle clusters, usually requiring under 20 time steps to converge to approximately the final positions. We have conducted statistical tests that show the model displays little sensitivity to the random initial particle positions, with standard deviations in the final material locations typically less than 5 percent of the mesh spacing [202].

This particle technique is able to produce the results in Fig. 2.20 where in a mixed cell, for each material randomly initialised particles (the number of which depend on the volume fraction) approximately gather around the material centroid. This particle method can produce approximate centroids in 2D and 3D.

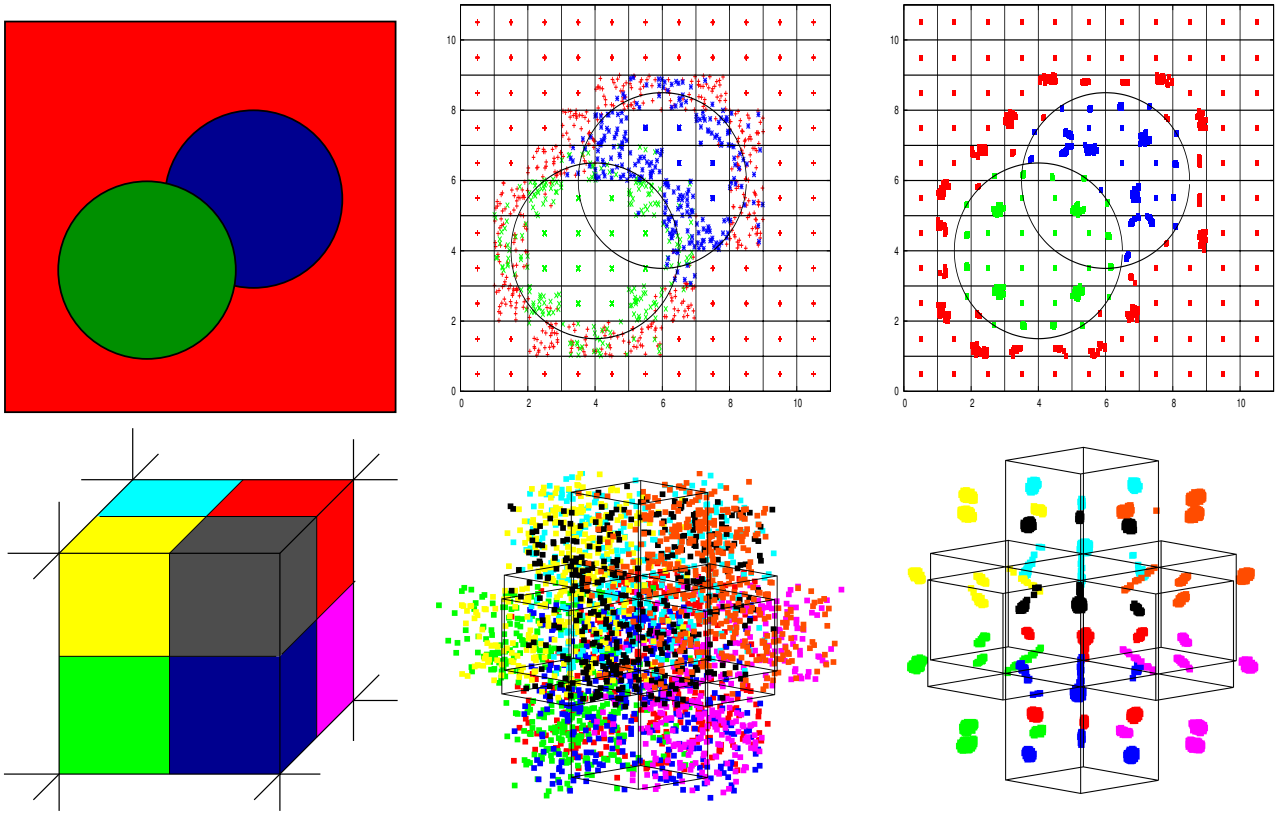


Figure 2.20 – *Numerical results obtained with the particle attraction model to infer the relative location of the materials in mixed cells. Top panels : two enlaced disks (left panel) are described by particles which are randomly distributed with mixed cells (middle). At convergence of the partcle model (right panel) particles do represent an accurate approximation of the material centroids — Bottom panels : approximation for 3D eight materials case (zoom on the central 8 materials mixed cell on the left panel), initial distribution of particles in the neighborhood of the central mixed cell (middle panel) and final location of particles in the enighborhood (right panel).*

An alternative technique in [13] is the direct calculation of an approximate center of mass of each material in a subset of the mesh around the cell being reconstructed.

Given the volume fractions of materials in cells in a mesh, our task is to determine the relative locations of materials in a multi-material cell. To do this, we must ideally recover the characteristic function for each material in the domain. While it is possible to reconstruct the characteristic function in 1D [203], no method (other than interface reconstruction itself) exists to do this in higher dimensions. Therefore, we make a simplifying assumption that a smooth function, called the volume fraction function, exists for each material and that its pointwise cell-centered values are given by cell-wise volume fraction data. This smooth function represents the distribution of material in the mesh cells and in that sense, it can be considered analogous to a density distribution function for the material. However, we should note that the volume fraction function is not a clearly defined mesh-independent continuous function like the density function. Swartz [204] describes it as the

function that quantifies the relative amount of a material present in a small window that moves around in a domain with a sharp interface. Defined this way, it is clear that the volume fraction function steepens as the size of the window (or in other words, the mesh size) gets smaller and the gradient of the function blows up as the window size goes to zero. Nevertheless, for a given mesh, we will treat the volume fraction function like a smooth, density distribution function.

We then compute a piecewise linear approximation for this smooth volume fraction function using standard methods used in higher-order finite-volume methods [205]. Finally, continuing the analogy with the density function, we compute the center of mass of the materials in cells from the linear reconstruction as described below.

Consider a mesh on which we have cell-centered values f_i of a function $f(\mathbf{x})$. In each cell $\mathcal{C}_i$, we reconstruct a linear approximation, $\tilde{f}_i(\mathbf{x})$, of the function such that

$$\tilde{f}_i(\mathbf{x}) = f_i + \nabla f \cdot (\mathbf{x} - \mathbf{x}_c(\mathcal{C}_i)), \tag{2.18}$$

where $\mathbf{x}_c(\mathcal{C}_i)$ is the centroid of the cell. ∇f is the gradient of the function that we wish to approximate and it is considered to be constant within the cell. The gradient may be computed either by a Green-Gauss [205] or a least-squares technique [191]. On structured and unstructured grids, we use all vertex and edge connected neighbors in the gradient computation. For a least-squares technique, the same mesh cells are used in the computation with each entry weighted by the inverse of the squared distance between the centroid of the cell being reconstructed and the centroid of the neighboring cell as described in [191]. The computed gradient is limited using Barth-Jesperson-type limiter [206] to preserve local bounds on the volume fraction function. The limiter is calculated using all vertex connected neighbors. The limited gradient is indicated by $\boldsymbol{\delta} = \phi \nabla f$ with $\phi \in (0, 1]$. Then, the approximate center of mass of the function $f(\mathbf{x})$ over the domain Ω_i as approximated by the function $\tilde{f}_i(\mathbf{x})$ is given by

$$\bar{\mathbf{x}} = \frac{\int_{\Omega_i} \mathbf{x} \tilde{f}_i(\mathbf{x}) d\Omega}{\int_{\Omega_i} \tilde{f}_i(\mathbf{x}) d\Omega} = \frac{1}{\|\Omega_i\| f_i} \int_{\Omega_i} \mathbf{x}(f_i + \boldsymbol{\delta} \cdot (\mathbf{x} - \mathbf{x}_c(\mathcal{C}_i))) d\Omega \tag{2.19}$$

where $\|\Omega_i\|$ is the area of the domain Ω_i.

The obvious choice of domain Ω_i for integrating this equation is the cell, $\mathcal{C}_i$ and this works well for structured meshes. The calculation of equation 2.19 for a polygon may be done with the application of Stokes' theorem in the plane, for details see [207]. However, for unstructured meshes, we have found that integrating over the cell domain induces a strong bias in the orientation of the reconstructed interface based on the cell geometry. In order to eliminate this effect, we integrate instead over the smallest square, $S(\mathcal{C}_i) \supseteq \mathcal{C}_i$, whose center coincides with the centroid of the cell, $\mathbf{x}_c(\mathcal{C}_i)$ and encloses the computational cell.

For two materials, this choice of integration domain is equivalent to a gradient-based method when using a power diagram interface reconstruction. In a power diagram based reconstruction of a two material cell, the interface normal depends only on the direction of the vector pointing from one material locator to the other. For two materials, m and n, with material locators $\mathbf{x}_m$ and $\mathbf{x}_n$ and volume fractions f_m and $1 - f_m$ respectively, the normal to the interface between them given by the power diagram reconstruction will be

$$\begin{aligned} \mathbf{x}_m - \mathbf{x}_n &= \frac{1}{\|S(\mathcal{C}_i)\|} \left(\frac{1}{\bar{f}_m} + \frac{1}{1 - \bar{f}_m} \right) \int_{y_0}^{y_1} \int_{x_0}^{x_1} \mathbf{x} \left(\boldsymbol{\delta} \cdot (\mathbf{x} - \mathbf{x_c}) \right) \, dx \, dy \\ &= \frac{\Delta^2}{12} \left(\frac{1}{\bar{f}_m (1 - \bar{f}_m)} \right) \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} \end{aligned}$$

where $S(\Omega_i) = [x_0, x_1] \times [y_0, y_1]$ and $\Delta = x_1 - x_0 = y_1 - y_0 = \sqrt{||S(\mathcal{C}_i)||}$. That is, the normal is a positive constant times the gradient. Hence, the interface normal will be the computed gradient. In addition, this choice of integration domain makes the calculation of equation 2.19 trivial and provides a better initial reconstruction for a starting point to the interface smoothing procedures.

Power diagram Using the relative location of materials obtained from the particle method or from a piecewise linear reconstruction of the volume fraction function, the interface is reconstructed using a weighted Voronoi diagram, known as a power diagram, such that the required volume fractions are exactly matched [208, 209].
In [13] we propose several static test cases to show the behavior of this new method. We reproduce in the following Fig. 2.21 the static four material interface reconstruction on structured and unstructured meshes using (a) particles and power diagrams, (b) approximate centroids and power diagrams, and in (c), (d) Youngs' method [190, 189] with two different material orderings. Notice that Youngs' method can not represent quadruple point, therefore no material ordering will provide an acceptable and accurate result.

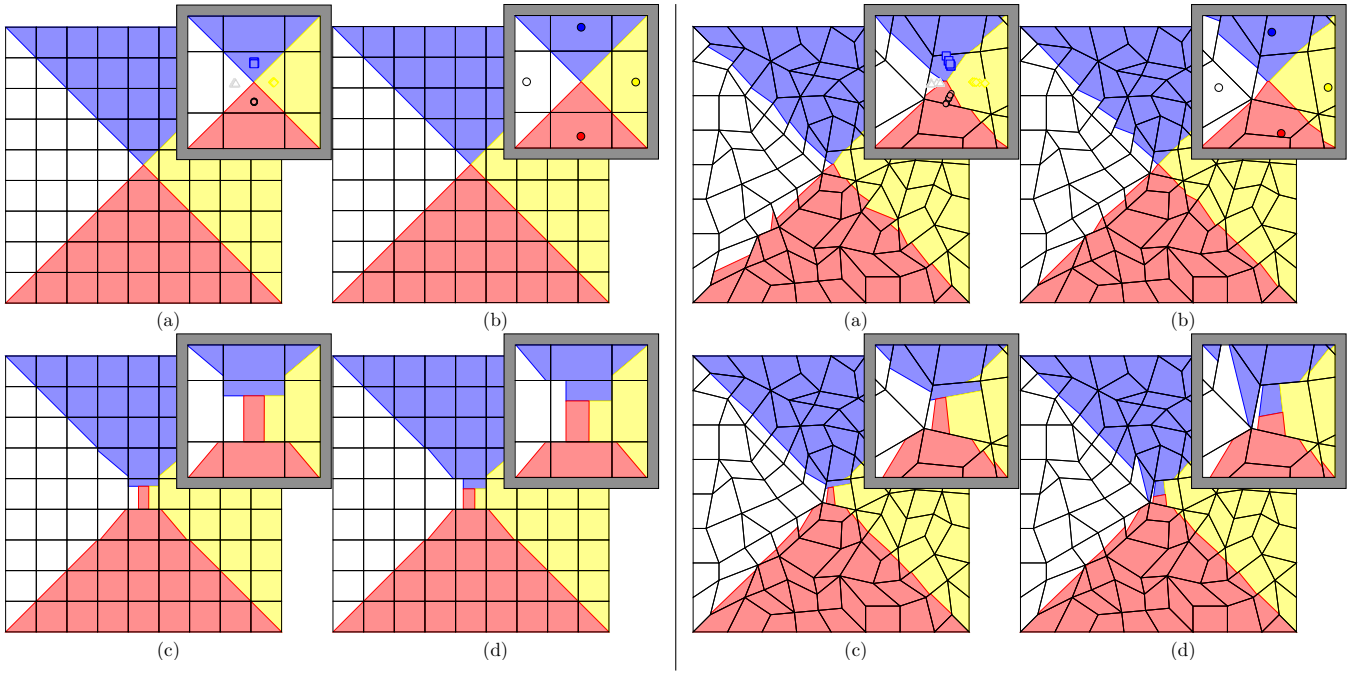
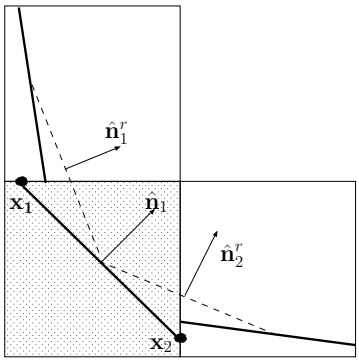
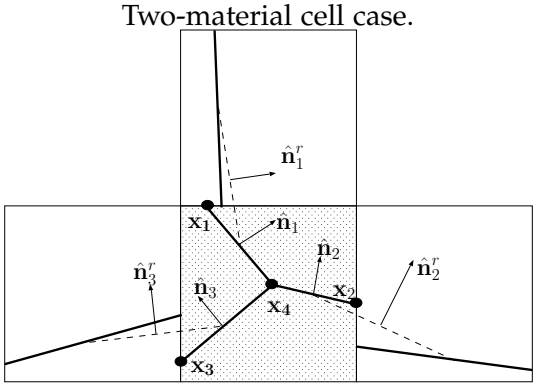


Figure 2.21 – *Numerical results from paper [13]. Four material interface reconstruction on structured and unstructured meshes using (a) particles and power diagrams (b) approximate centroids and power diagrams (c), (d) Youngs' method with two different material orderings. The insets show the four material cell at the center of the mesh. The converged particles locations for the center cell are also shown in the inset in (a). The approximate centers of mass for the center cell are shown in (b).*

Neither the particle model nor the approximate center of mass method when combined with a power diagram-based reconstruction exactly reproduces a straight line, indicating that both methods are only first-order accurate in this sense. Consequently in [17] the interfaces are improved by minimizing an objective function that smoothes interface normals while enforcing convexity and volume constraints for the pure material subcells.



Two-material cell case.



Three material cell case.

Consider a 2D cell with N_m materials, N_s interface segments and N_p interface points. The smoothing procedure repositions the cell's N_p interface points so that it minimizes the discrepancy between the normal of each of its interface segments and normals of reference interface segments in neighboring cells (separating the same materials). The constraints imposed on this process are that the volume fractions of the materials in the cells must be matched exactly and that all the pure material subcells remain convex. Naturally, interface points on the boundary of the cell must remain on the boundary and interior points must remain strictly inside the cell. The local objective function for smoothing in a particular cell is written as :

$$F_i(\mathbf{s}) = \sum_{j=1}^{N_s} \sum_{k=1}^{(N_r)_j} \|\hat{\mathbf{n}}_j(\mathbf{s}) - \hat{\mathbf{n}}_k^r\|^2.$$

As an illustration the figure to the left presents the definition of reference normals.

In order to numerically validate this approach we have proposed in [17] a set of test cases : the diagonal translation of a four material disk and a four material vortex test in an incompressible reversible velocity field. Convergence tests from the article show the second-order accuracy of the proposed method.
As instance one presents in Fig. 2.22 the translation of a four material disk In this figures one compares our second-order approach with Youngs' method using two different prescribed material orders. The four material vortex test case's results are presented in Fig. 2.23 where our first-order method is also presented.

$\sim$

This chapter was devoted to the description of some of our investigations of the rezone and remap phases of and ALE code. Rezone has been extended to allow mesh reconnection and gave rise to the so-called ReALE method. Finally we have presented our contribution to interface reconstruction methods which are able to deal with more than two materials in multi-material fluid flows.

The next chapter presents some of our investigations which are not genuinely related to ALE *per se*. This chapter deals with very high-order MOOD schemes, kinetic schemes and interface reconstruction technique in an Eulerian context.

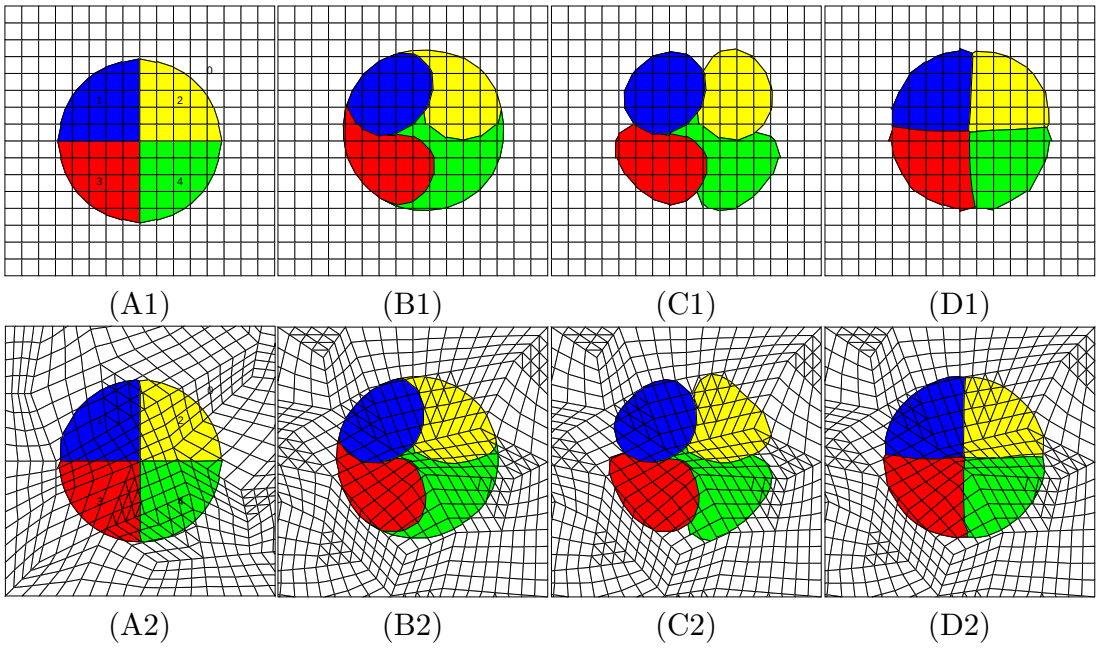


Figure 2.22 – *Numerical results from paper [17]. Final configuration of the four material circle shown in (A1) with material numbers after diagonal translation with a velocity of* $(1,1)$ *at time* $t = 0.5$ *using the interface reconstruction methods : (B1) Youngs' with material order* $(0,1,2,3,4)$ *(C1) Youngs' with material order* $(1,2,3,4,0)$ *(D1) our second order method. (A2)-(D2) show the same results on an unstructured mixed triangle and quadrilateral grid.*

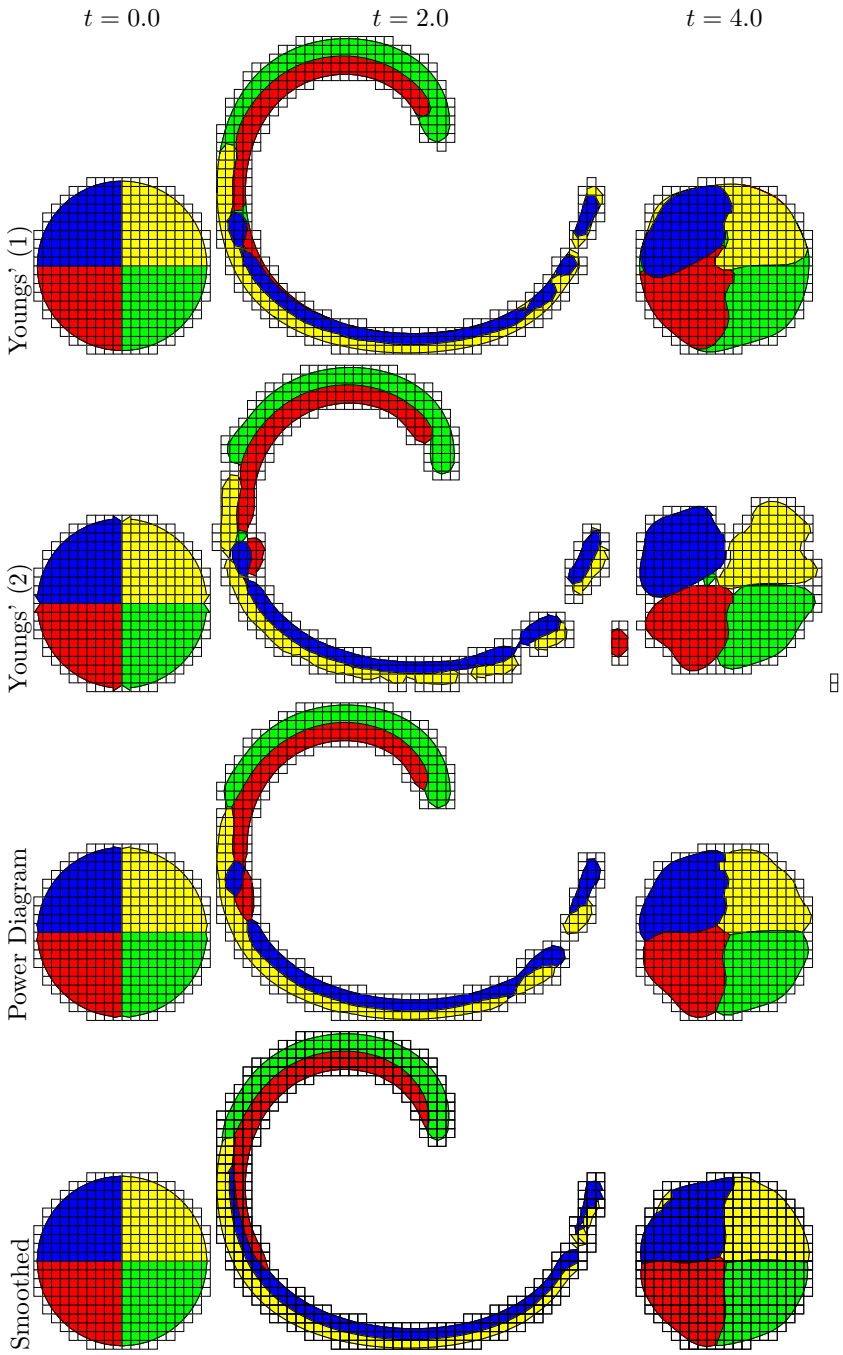


FIGURE 2.23 – *Numerical results from paper [17]. Material interface configuration for the four material vortex test initialy $t = 0$ (left column) at maximum stretch $t = 2$ (middle column) and at complete reversal time $t = 4$ (right column) run on a 64×64 grid. The material numbers are 0 for the white material, 1 for blue, 2 for yellow, 3 for red and 4 for green. For the method Youngs' (1), the material ordering was $(0, 1, 2, 3, 4)$. For method Youngs' (2), the order was $(1, 2, 3, 4, 0)$. Power diagram corresponds to the firts-order method from [13] whereas smoothed corresponds to the second-order method from [17].*

Other more or less related investigations

3

Contents

In this chapter we present some topics that we have treated which are more or less related to Lagrangian and ALE numerical schemes. I came across these projects thanks to colleagues and collaborators, and I would like to take the opportunity to thank them for feeding me with their idea. An exhaustive description of the context, the existing methods and the details of our approaches is available in the published papers and are not rephrased here. Instead we justify why these topics of research have been initiated and emphasize some difficulties that have been resolved and also some others which are still to be solved. Moreover several numerical results from our papers will be reproduced in order to show how and why these investigations led to numerical methods and simulation codes that are of interest for the community.

More precisely this chapter is organized in the following sections :

A section devoted to the Multi-dimensional Optimal Order Detection (MOOD) method. This method has been developed during the three years of a PhD (2009-2012) S. Diot shared with S. Clain (university Do Minho, Guimaraes Portugal). The Multidimensional Optimal Order Detection (MOOD) method which has been developed and improved in set of three successive papers [29, 32, 33] and also a set of proceedings [30, 31]. This method is a very high order finite volume method based on polynomial reconstruction based on *a posteriori* polynomial degree decrementing which plays the role of a classical limitation. This provides a different manner of considering how, where and when limitation is needed.

A section dedicated to the presentation of an ultra Fast Kinetic Scheme (FKS) [39]. This work has been done with G. Dimarco (IMT) following one of his idea. This consists of a new ultra efficient numerical method for solving kinetic equations in the case of the BGK relaxation operator. The scheme is based on a splitting technique between transport and collision. The key

idea is to solve the collision part on a grid and then to solve exactly the transport linear part by following the characteristics backward in time. The main difference between the method proposed and semi-Lagrangian methods is that here we do not need to reconstruct the distribution function at each time step. This allows to tremendously reduce the computational cost of the method and it permits to compute solutions of full six dimensional kinetic equations on a single processor laptop machine.

A section devoted to the interface reconstruction technique within the Finite Volume with Characteristic Flux scheme [210]. This scheme is an Eulerian finite volume scheme based on characteristic decomposition of the flux between neighbor cells. A two material extension has been proposed by J.-P Braeunig *et al* in [211, 212], this extension employs a VOF approach with a SLIC interface reconstruction method. However we have shown that some inconsistency in the interface reconstruction method led to poor advection results. Our contribution proposed an improvement of this so-called Natural Interface Positioning (NIP) method [211, 212]. Our modification, called Enhanced Natural Interface Positioning has been published in [18] and its extension to deal with more than two materials is to be published in a forthcoming paper [41].

3.1 Very high order finite volume scheme : The Multidimensional Optimal Order Detection method (MOOD)

In this work we solve the advection equation and the Euler system of hydrodynamics with a method built on the basics of the Finite Volume (FV) numerical scheme. FV method considers piecewise constant values of the variables per cell (*i.e* mean values) and computes their evolution in time. As already known FV method using constant states is only first-order accurate in space. Higher order accurate methods can be obtained as instance using polynomial reconstruction using mean values to evaluate more accurately the flux. Unfortunately some sort of limitation is needed to avoid spurious oscillations near discontinuous profiles (shock wave or contact discontinuity as instance). Close to discontinuities any stable scheme must degenerate to an at most first-order accurate one.

3.1.1 MOOD key idea : "*a posteriori*" limitation

Classical high order polynomial reconstruction schemes such as the Monotonic Upstream-centered Schemes for Conservation Laws (MUSCL) based on Godunov approach and introduced by van Leer [213], or various Essentially Non-Oscillatory schemes (ENO) proposed by Harten, Osher and Shu [214, 215, 216], are based on an *a priori* limiting procedure to achieve some stability property. In MUSCL like methods unlimited slopes are reduced through the use of a slope limiter whereas the least oscillating polynomial is chosen for ENO/WENO like methods. There is a vast literature about slope/flux limiters, some of them are now known after their discoverers (van Leer [217, 218, 213], van Albada [219], Sweby [220], Barth-Jespersen [221], Venkatakrishnan [222], Koren [223], etc.) or their particularity (minmod [224], superbee [224], monotonized central [218], etc.). Finding the Essentially Non-Oscillating polynomial (hence the name of the method (W)ENO), improving the choice of reconstruction stencils and reducing the possible huge number of stencils have also led to a considerable sum of articles see as instance [214, 215, 216], more specifically [225] and the bibliography herein.
In any case these types of limitation are performed *a priori* by a clever analyze of the available data. This implies that the "worst case scenario" must always be considered as plausible, and, as a consequence, the "precautionary principle" applies. In other words because scientific investigation has found a plausible risk of instability development, *a priori* limitations strike more often and harder than necessary. We believe that these limitations can be relaxed only if further information emerges that provides evidence that no harm will result when using unlimited reconstruction. One way to attain this goal is to check *a posteriori* if a solution has failed to fulfill some stability criteria [1]
The principles of the MOOD method are as simple as : first compute a candidate solution without any limitation, then detect if this solution locally fails to fulfill some stability criteria (problematic regions) and further uses limiter only on problematic regions to recompute the new candidate solution. The new candidate solution is then checked again for eligibility. The MOOD method follows this fundamentally different way. A maximal polynomial degree is set. Then a polynomial degree reduction plays the role of *a posteriori* limitation. An iterative procedure which decrements polynomial degree in problematic regions provides the optimal local polynomial reconstruction which satisfies given stability criteria.
Doing so we can ensure the positivity of the scheme by construction if the lowest order scheme is. Moreover we ensure that the numerical solution obtained is one of the most accurate solution achievable because every higher order polynomial reconstructions have been tested but the first

1. Treating *a posteriori* if a solution is valid is not new and can be found in the context of remapping methods (decreasing of polynomial order in [226], repair methods [141, 142, 8, 9]) and presumably in many other areas.

reconstruction leading to an acceptable solution [2].
In [29] we introduced this MOOD concept which provides up to third-order approximations to hyperbolic scalar or vectorial solutions for two-dimensional geometry. Then in [32] we have extended to general unstructured 2D meshes and to sixth-order convergence in space. Finally in [33] the 3D version has been deployed. We refer the reader to these papers to an exhaustive description of the MOOD method.

3.1.2 MOOD performances in 1D, 2D and 3D

The MOOD method has been entirely developed and extensively tested by S. Diot with his 2D and 3D codes. It has led to three publications [29, 32] and [33]. For the first paper [29] we have presented the method and the associated concepts of cell and edge polynomial degree. An effective third order of accuracy of the MOOD method on advection equation on irregular structured grid has been achieved. Then on Euler equations we have shown that the MOOD method with piecewise parabolic or linear reconstructions is nicely performing on classical test cases (Sod shock tube, four state Riemann problem, Mach 3 step problem, double Mach reflection). The MOOD method is also favorably compared to classical Finite Volume and MUSCL like methods.
The second publication [32] introduces the extension of MOOD to 2D unstructured meshes with higher order polynomials (up to six). In this paper we have shown that the expected high order of convergence is reached both for advection and on smooth solutions of the Euler equations. To reach the sixth-order of accuracy for a $\mathbb{P}_5$ polynomial reconstruction we have relaxed the strict discrete maximum principle which is a cause of limitation to second-order of accuracy along with the use of non-conservative variable reconstructions, see also [227] on this point. A detector of smooth solution has been designed, it avoids polynomial degree decrementing and ensures a high order of accuracy on smooth profiles. The method has been tested on unstructured and non-conformal meshes. For the advection section we have tested the MOOD method on a smooth solution (double sine translation) and on discontinuous profiles (solid body rotation). For the Euler equations we have considered and isentropic vortex problem which admits a smooth exact solution. The MOOD method can effectively produce the optimal order of accuracy, up to sixth-order for a $\mathbb{P}_5$ polynomial reconstruction. The 1D Lax shock tube has further been run to compare MOOD with classical WENO method. The double Mach problem has been used to assess the low storage and speed-up of the MOOD method on three different single core machines. We finally have tested the ability of MOOD method to capture physics in realistic conditions by simulating the experiment proposed in where a planar shock impacts a cylindrical cavity, see Figs. 3.1.
The last paper deals with the 3D version of the method on unstructured grid. This paper is meant to prove that the MOOD method can be developed in 3D on single core machines. The method has been extensively tested on different machines to estimate its actual cost. On advection equation we have shown that the method can reach effective high order of accuracy. For Euler equations we have designed a 3D extension of the impact of a shock wave on a cylindrical cavity. A spherical version of the Sod shock tube has also been simulated. Last the interaction of a shock wave with a quarter of cone has been run with the fourth order accurate MOOD method. This is a 3D extension of the so-called "interaction of a shock wave with a wedge" problem. We reproduce in Fig. 3.2 the figure from the paper where the mesh (colored by the cell volume) and the principal waves are shown.

2. defined as a solution fulfilling user-given stability criteria such as physical admissibility of the solution, positivity of some variables, non-oscillatory solution, etc.

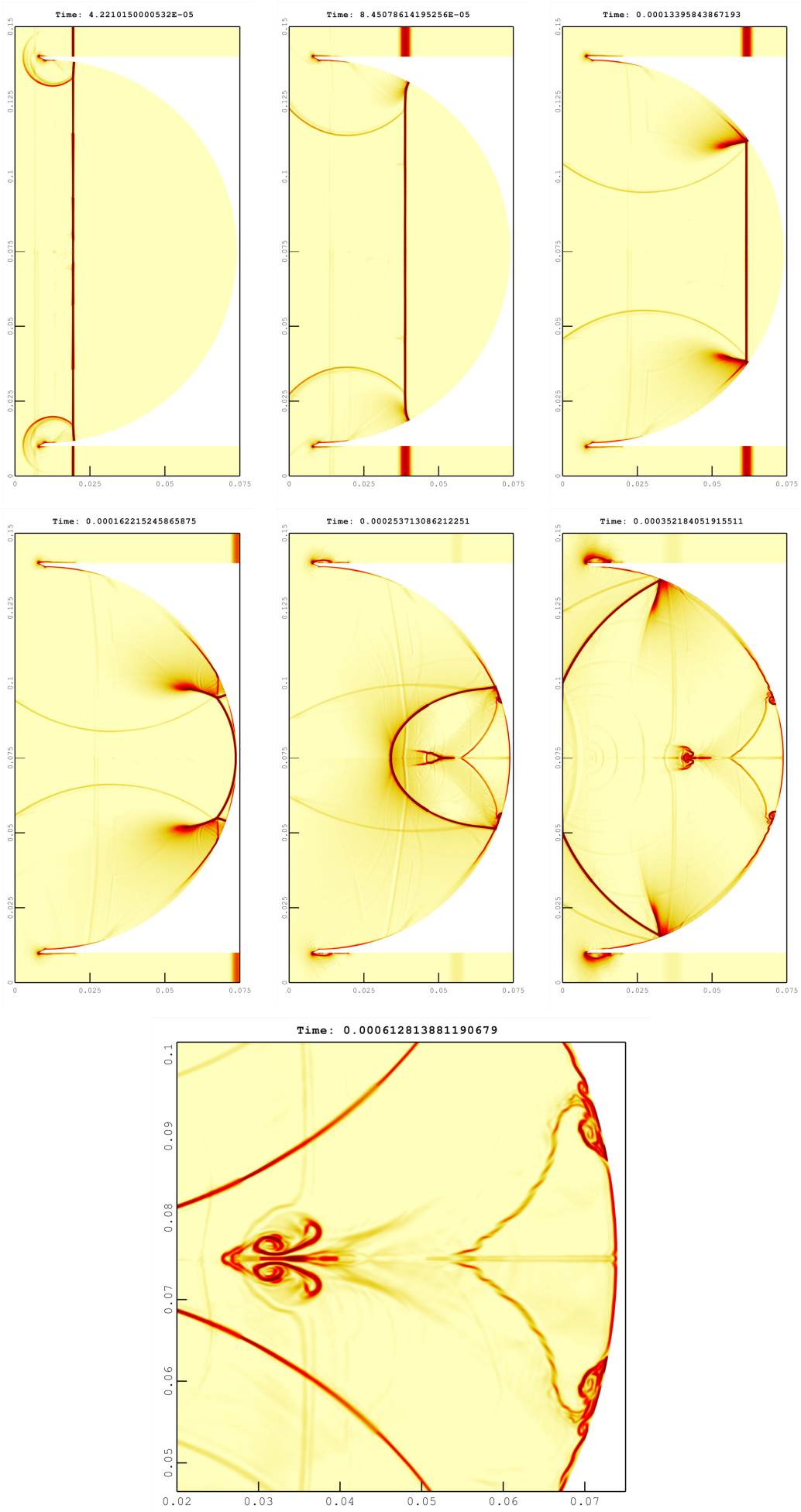


FIGURE 3.1 – *Results of the 2D MOOD-$\mathbb{P}_2$ method on the impact of a shock wave on a cylindrical cavity. Gradient*

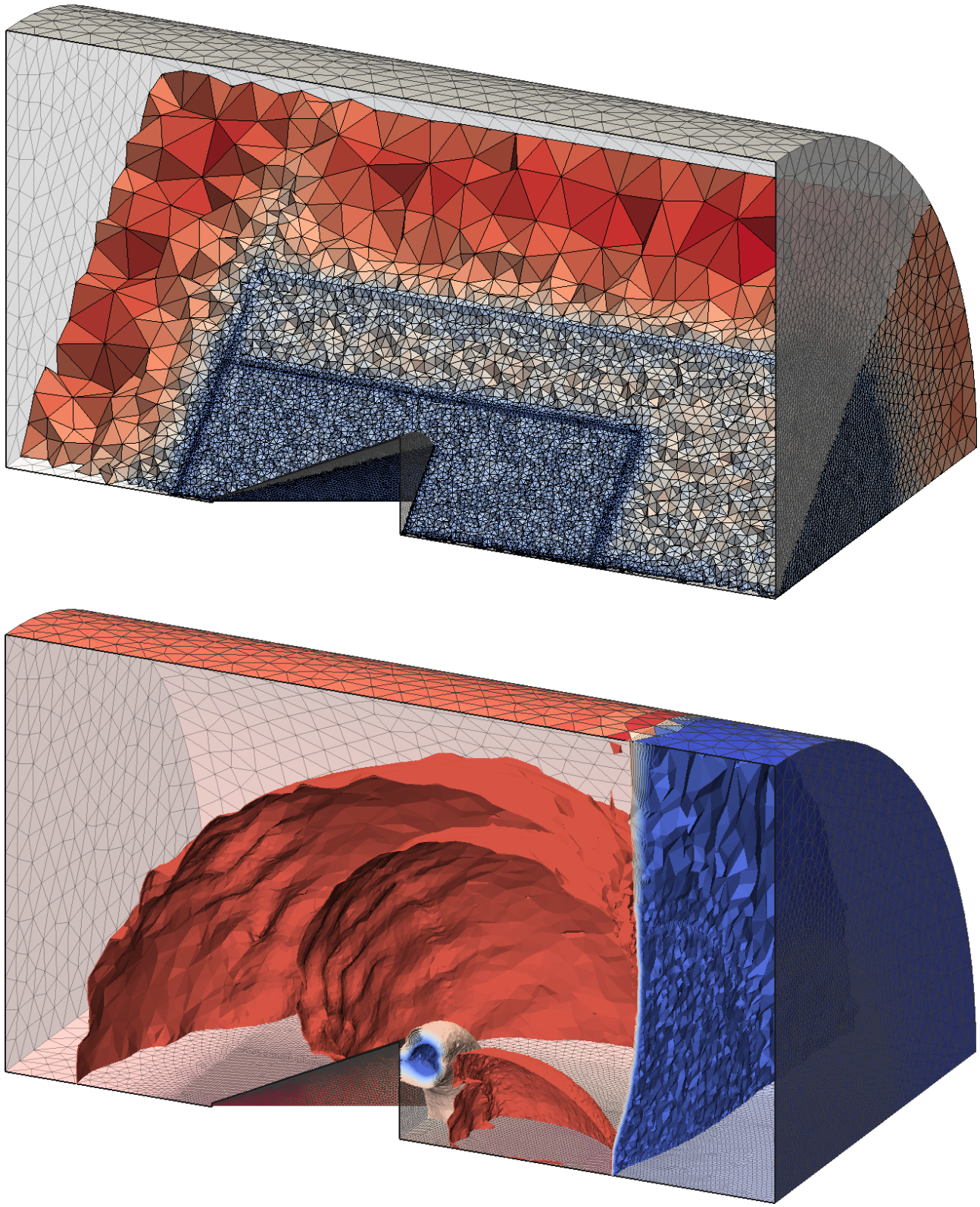


Figure 3.2 – *Results of the 3D MOOD-$\mathbb{P}_3$ method on interaction of a shock wave with a half cone. Top panel : view of the interior of the tetrahedral mesh with the different zones of refinement. Bottom panel : isosurfaces corresponding to the principal waves.*

3.2 Ultra efficient 3D kinetic scheme

Recently with my colleague Giacomo Dimarco (IMT) we have designed an ultra efficient 3D kinetic scheme [39]. The main idea has been developed by Giacomo and my role has been reduced to implement the 3D version of this idea in an efficient numerical simulation code based on the straighforward collision operator, i.e. the BGK (Bhatnagar-Gross-Krook) relaxation operator. The purpose of this work is to show that simulating kinetic equations in seven dimensions ($\mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}$ respectively for 3D in space, 3D in velocity and 1D in time) is feasible with nowadays laptop with our new approach.

3.2.1 Quick refresher on the context

The kinetic equations provide a mesoscopic description of gases and more generally of particle systems. In many applications, the correct physical solution for a system far from thermodynamical equilibrium, such as rarefied gases or plasmas, requires the resolution of a kinetic equation [228]. However, the numerical simulation of these equations with deterministic techniques presents several drawbacks due to the large dimension of the problem. The distribution function depends on seven independent variables : three coordinates in physical space, three coordinates in velocity space and the time leading to the seven dimensions already mentioned. This "curse of dimentionality" is often used as a blanket excuse for not dealing with high-dimensions. This has led the researchers to find solution to avoid the use of the seven dimensions or, at least, to reduce the burden of dealing with them. Probabilistic techniques such as Direct Simulation Monte Carlo (DSMC) methods [229, 230, 231, 232] are extensively used in real situations due to their flexibility and low computational cost compared to finite volume, finite difference or spectral methods for kinetic equations [233, 234, 235, 236, 237]. On the other hand, DSMC solutions are affected by large fluctuations. Moreover, in non stationary situations it is impossible to use time averages to reduce these fluctuations and this leads to, either poorly accurate solutions, or again to computationally expensive simulations. For this reason, many different works have been dedicated to reduce some of the disadvantages of Monte Carlo methods, see as instance [230] for an overview on efficient and low variance Monte Carlo methods.
In this work, we consider the development of a new deterministic method to solve kinetic equations. The key point is an efficient discretization of the linear transport part of these equations. The proposed method is based on the so-called discrete velocity models (DVM) [235] and on the semi Lagrangian approach [238, 239]. The DVM models are obtained by discretizing the velocity space into a set of fixed discrete velocities [240, 235, 236, 241]. As a result of this discretization, the original kinetic equation is then represented as a set of linear transport equations plus an interaction term which couples all the equations. In order to solve the resulting set of equations, the most common strategy consists in an operator splitting strategy [242] : the solution in one time step is obtained by the sequence of two stages. First one integrates the space homogeneous equations and then, in the second stage, the transport equation using the output of the previous step as initial condition. More sophisticated splitting techniques can be employed, which permits to obtain high order in time discretizations of the kinetic equations as for instance the Strang splitting method [243]. In any case, the resulting method is very simple and robust but the main drawback is again the excessive computational cost. It is a matter of fact that the numerical solution through such microscopic models and deterministic schemes remains nowadays too expensive especially in multi-dimensions even with the use of super-computers.

3.2.2 Ultra Fast-Kinetic-Scheme (FKS)

To overcome this problem, we propose to use a Lagrangian technique which exactly solves the transport stage on the entire domain and then to project the solution on a grid to compute the contribution of the collision operator. The resulting scheme shares many analogies with semi-Lagrangian methods [238, 239, 234] and with Monte Carlo schemes [244], but on the contrary to them, the method is as fast as a particle method while the numerical solution remains fully deterministic, which means that there is no source of statistical error.
The main features of the method proposed in this work can be summarized as follows :

- The BGK equation is discretized in velocity space by using the discrete velocity models (DVM) method. The principle of Discrete Velocity Model (DVM) [235] is to set a grid in the velocity space and to transform the kinetic equation in as set of N linear hyperbolic evolution equations with source terms.
- A time splitting procedure is employed between the transport and the relaxation operators for each of the N evolution equations. First- and second-order Strang time splittings [243] are considered.
- The transport part is solved exactly, which means without using a spatial mesh. The initial data of this step is given by the solution of the relaxation operator.
- The relaxation part is solved on the grid. The initial data for this step is given by the value of the distribution function in the center of the cells after the transport step.

We refer the reader to [39] for the details and we only describe the key points of our fast algorithm. The algorithm relies on a very efficient transport step performed on a logical rectangular grid. First the 1D velocity bounded space $\mathcal{U} = [u_{min}, u_{max}]$ is represented by N_v particles uniformly distributed

$$u_p = u_{min} + (p-1)\Delta u + \Delta u/2, \tag{3.1}$$

with $\Delta u = (u_{max} - u_{min})/N_v$. The same process is made for v and w components of the velocity space leading to $N_p \times N_p \times N_p$ particles which pave $\mathcal{U} \times \mathcal{V} \times \mathcal{W}$. Because a generic particle moves with constant velocity $\boldsymbol{U}_p = (u_p, v_p, w_p)$ the transport step consists of solving the N_v^3 equations indexed by p with

$$\widetilde{\boldsymbol{X}}_p^{n+1} = \boldsymbol{X}_p^n + \Delta t\, \boldsymbol{U}_p \qquad \forall p = 1, \cdots, N_v^3. \tag{3.2}$$

The first key point in our approach is to work with a regular spacial mesh made of $N_i \times N_j \times N_k$ cells, all cells being the same. Each cell is indexed with three indexes i, j, k for each spacial direction. N_v^3 particles are localized and further will evolve within each spacial cell. Each particle carries its own moments, mass m_p, momentum $m_p \boldsymbol{U}_p$ and energy $\frac{1}{2} m_p \|\boldsymbol{U}_p\|^2$. The mass is computed *via* the distribution fonction feeded with the macroscopic state of the current cell.
The second key point is to set the initial position of all N_v^3 particles at the cell center of their associated cell $\boldsymbol{X}_{i,j,k}$, that is to say

$$\boldsymbol{X}_p^0 = \boldsymbol{X}_{i,j,k}, \qquad \forall p = 1, \cdots, N_v^3. \tag{3.3}$$

After the transport step (3.2) of particle p in cell $\Omega_{i,j,k}$ either $\widetilde{\boldsymbol{X}}_p^{n+1}$ remains in cell $\Omega_{i,j,k}$ or it lands in a neighbor cell Ω_{i_l,j_l,k_l} where $i_l = i + a$ with $a = -1, 0$, or $+1$ (idem for $j_l = j + b$ and $k_l = k + c$). If particle p remains within its cell then the cell moments are not modified. Contrarily if particle p leaves its cell then the cell $\Omega_{i,j,k}$ moments are decreased by particle p's moments whereas the cell Ω_{i_l,j_l,k_l} moments are increased. The new moments in each cell are therefore decreased due to the leaving particles and increased by the incoming particles, see Figure 3.3-left. By construction

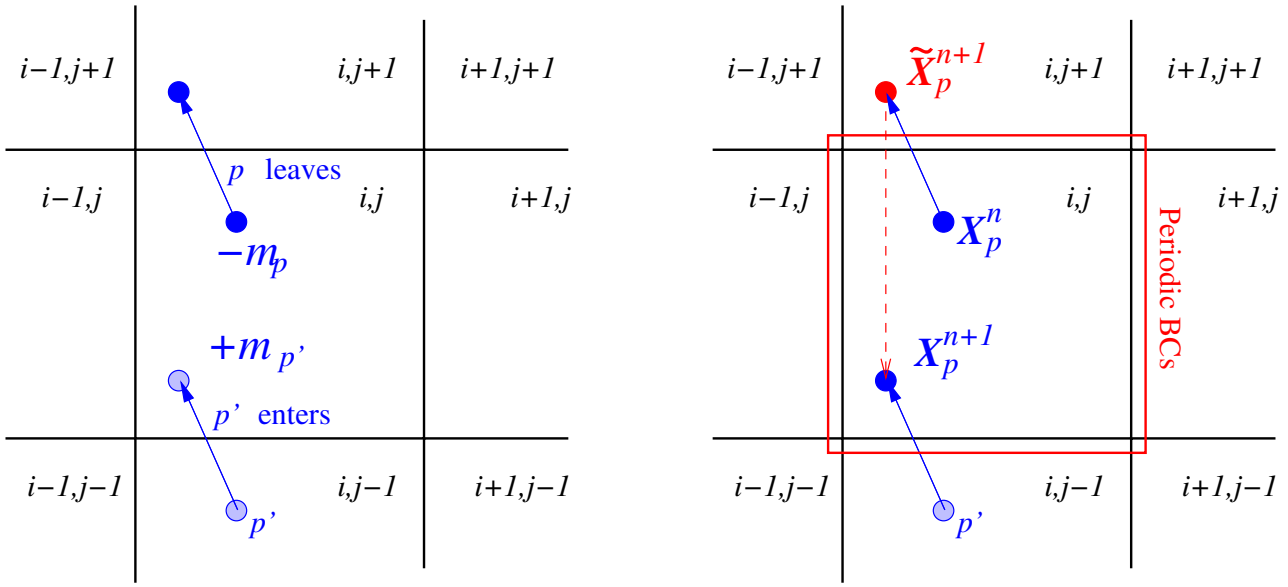


FIGURE 3.3 – *Sketch of the 2D transport step in the fast kinetic scheme — Left : a particle p leaves the current cell i, j with its mass m_p (the cell moments are therefore decreased) and it lands in cell $i, j+1$ (it contributes to the cell moments). If p leaves the cell then a sister particle p' enters and contributes to the moments in cell i, j — Right : the same situation for the particle positions. The true new position $\boldsymbol{X}_p^{n+1}$ is computed as if periodic boundary conditions are applied to cell i, j.*

conservation of moments is ensured.

Due to the initialization of particle positions the fact that p leaves its cell implies that a "sister" particle p' on the other side of the cell is entering, see Figure 3.3-right. This "sister" particle has a position $\widetilde{\boldsymbol{X}}_{p'}^{n+1}$ in $\Omega_{i,j,k}$ which is the same as $\widetilde{\boldsymbol{X}}_{p'}^{n+1}$ in its new cell Ω_{i_l,j_l,k_l}. It is then easy to see that this situation is equivalent to assume periodic boundary conditions on each cell. Therefore the new particle p position in cell $\Omega_{i,j,k}$ is

$$\boldsymbol{X}_p^{n+1} = \boldsymbol{X}_p^n + \Delta t\, \boldsymbol{U}_p \qquad \text{subject to periodic BCs on} \quad \Omega_{i,j,k}. \tag{3.4}$$

Doing so only the positions of particles need to be updated for only one spacial cell. In addition only one set of N_v^3 particle positions need to be stored, which drastically reduces the memory consumption of the method. The information which must be kept in the case particle p leaves the cell, is the integer vector (a, b, c) which determines in which cell particle p lands. Finally the algorithm simply consists of transporting the particles and marking the particles leaving their cell, computing the moments of the leaving particles, update the cell moments, compute the moments of the incoming particles and re-update the cell moments.

Thanks to this approach we are able to compute the solution of the full six dimensional kinetic equation on a laptop. This is, up to our knowledge, the first time that the full kinetic equation has been solved with a deterministic scheme on a single processor machine for acceptable mesh sizes and in a reasonable amount of time (around ten hours for 100^3 space $\times$ 12^3 velocity space mesh points).

3.2.3 Numerical experiments in 3D/3D

Here we report some simulations of the full 3D/3D problem [3]. As already mentioned the goal is to numerically show that such a kinetic scheme can reasonably perform on six dimensions on a mono-processor laptop. All simulations have been carried out on a HP EliteBook 8740W Intel(R) Core(TM) i7 Q840@1.87GHz running under a Ubuntu (oneiric) version 11.10. The code has been compiled with gfortran 4.6 compiler with -O3 optimization flags.
The 3D Sod shock tube has been run with the 3D/3D FKS method. The left state of the 1D Sod problem is set for any cell c with cell center radius $r_c \leq 1/2$, conversely the right state is set for cell radius $r_c > 1/2$. The final time is $t_{\text{final}} = 0.1$. The domain is the unit cube and the mesh is composed of $N_x \times N_x \times N_x$ cells with $\Delta x = 1/N_x$ and $\Delta x = \Delta y = \Delta z$. The problem is run with $N_x = 50$ (125000 cells), $N_x = 100$ (1 million cells) and $N_x = 200$ (8 millions cells). The velocity space is either $[-10;10]$ discretized with 12^3 points, or $[-15;15]$ discretized with 13^3 points. This leads to consider up to $200^3 \times 13^3 \simeq 17.7$ milliards of particles. The time step is fixed to 95% of the maximum time step allowed, as prescribed by the CFL condition, apart from the last time step. Symmetric boundary conditions are considered. In Figure 3.4 the density is plotted as a function of the radius (left panel) and the colored density on a 3D view (right panel) for $N_x = 50$ (middle panels) and $N_x = 200$ (bottom panels). The two different choices for the bounds and the mesh points in velocity space do not significantly change the results hence only the solution with bounds $[-10;10]$ and with 12^3 mesh points is reported. The reference solution is obtained with ALE INC(ubator) code [5] with 1000 cells in radial and 20 cells in angular directions. Moreover in Figure 3.4 (top panel) we present the convergence of the density as a function of cell center radius for all cells for the $50 \times 50 \times 50$, $100 \times 100 \times 100$ and $200 \times 200 \times 200$ cells meshes. These curves are compared to the reference solution in straight thick line and they show that the results are converging towards the reference solution. In table 3.1 we gather the number of time steps and the total CPU time T for 50^3 and 100^3 cell meshes for the two different configurations : one with $N_v = 13$ and the velocity space $[-15,15]$ and the second one with $N_v = 12$ and the velocity space $[-10,10]$. For the 50^3 mesh the simulation takes 45 minutes or 1.36 hour depending on the configuration. For the finer 100^3 mesh the simulation takes either 11 hours or 24 hours The memory consumption ranges from 124Mb to 924Mb depending on the configurations and it scales with the number of cells N_c.

Then, we compute the cost per cycle T_{cycle} and per cycle per cell T_{cell}. One observe that the cost per cycle per cell is an almost constant equal to 4×10^{-4}s or 5.5×10^{-4}s. The extrapolation of the CPU time T for a 200^3 mesh at T_{cell} fixed leads to one or two weeks computation for the two configurations and a memory storage of about 900MB. In Figure 3.5 we plot the CPU time (red or blue symbols for each configuration and mesh points of the velocity space) and the extrapolation curves $CPU(N_x, N_c, T_{\text{cell}}) = \frac{N_{\text{cycle}}}{N_x} N_c T_{\text{cell}}$ for the 3D Sod problem up to time $t_{\text{final}} = 0.1$ for single processor laptop computation on a fixed mesh in velocity space of $N_v = 12^3$ points. We deduced that the FKS method can be used at most on a single processor machine up to a $200 \times 200 \times 200$ cells for roughly one week of computation. One also notices that the CPU time linearly scales on a log/log graph as expected (right panel of Figure 3.5)

In the future we would like to extend the method to non uniform meshes, more advanced boundary conditions and different discretization of the velocity space. One expects with this last point to increase the accuracy of the schemes without losing its attractive efficiency. To avoid the loss of accuracy close to the fluid limit, we want to couple the FKS method to an high order solver for the

3. We consider the case in which the projection is made towards the equilibrium at each time step. We recall that, in this regime, the numerical method gives the worst results in terms of precision, on the other hand, exact solution are known and this permits to make fair comparisons.

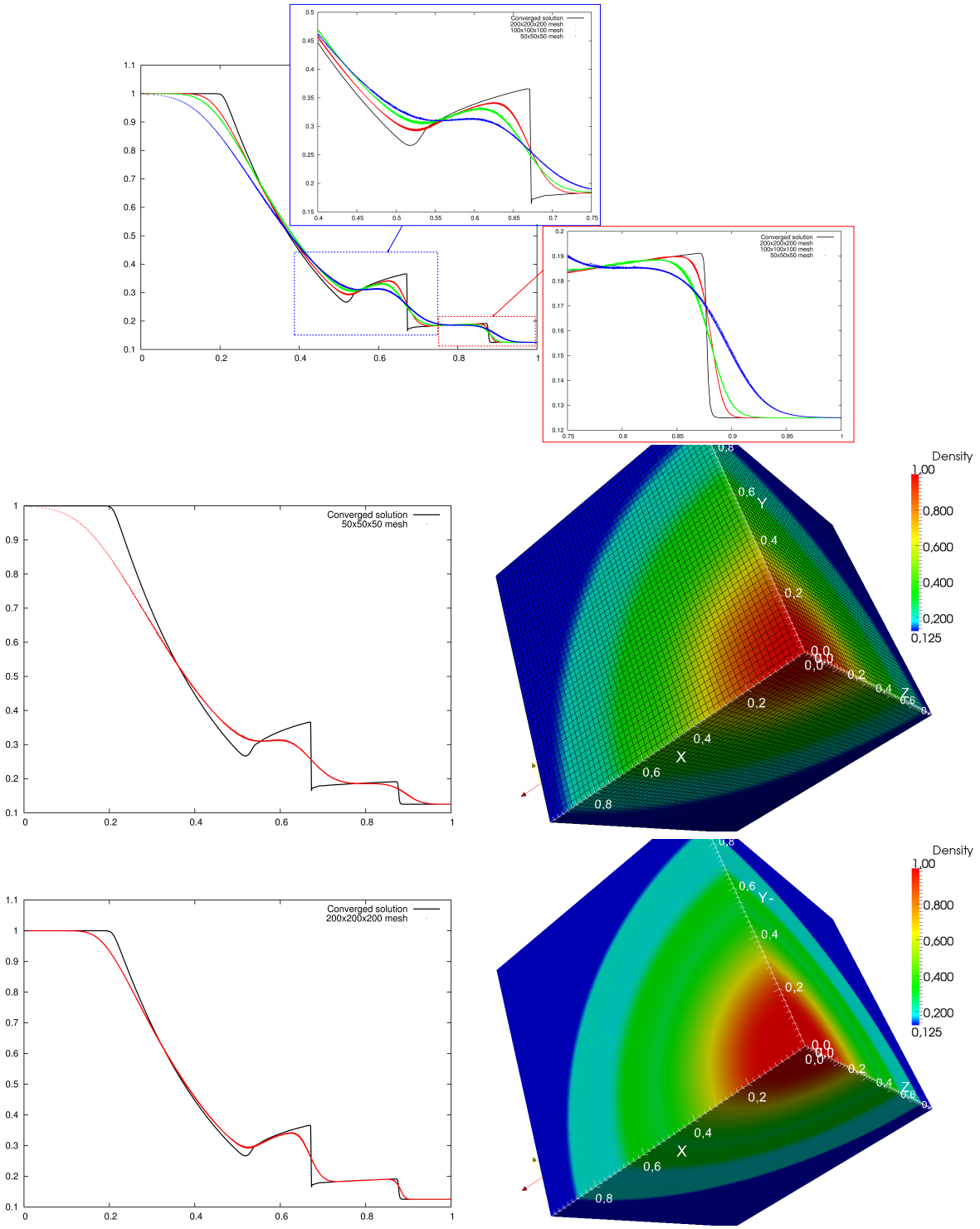


FIGURE 3.4 – *Sod problem at* $t_{final} = 0.1$ *for* $N_x \times N_x \times N_x$ *cells (for* $N_x = 50, 100, 200$*) for the velocity space* $[-10; 10]$ *discretized with* 12^3 *mesh points. — Top : Convergence of density as a function of cell center radius for all cells vs converged solution (straight thick line) for the three meshes with zooms on contact and shock waves. Left : Density as a function of cell center radius (middle :* $N_x = 50$*, bottom :* $N_x = 200$*) Right : 3D view of density on the unit cube* $N_x = 50$ *(middle) and* $N_x = 200$ *(bottom) (the mesh is only shown for* $N_x = 50$*).*

N_v^3 Bnds	**Cell #** $N_c \times N_v^3$ $N_x \times N_y \times N_z \times N_v^3$	**Cycle** N_{cycle}	**Time** T (s)	**Time/cycle** T_{cycle} (s)	**Time/cell** T_{cell} (s)	**Mem** (MB)
13^3 ± 15	$25^3 \times 13^3$ $= 3.4328125 \times 10^6$	32	346s (5.76mn)	10.81	6.92×10^{-4}	2.4
	$50^3 \times 13^3$ $= 274.625000 \times 10^6$	81	4900s (1.36h)	60.50	4.84×10^{-4}	15.5
	100×13^3 $= 2.1970 \times 10^9$	160	85720s (23.8h)	535.75	5.36×10^{-4}	115.5
extrapol.	200×13^3 $= 1.7576 \times 10^{10}$	320	$\sim 1.4 \times 10^6$s (16d)	~ 4400	5.5×10^{-4}	~ 900
12^3 ± 10	$25^3 \times 12^3$ $= 27 \times 10^6$	27	218s (3.63mn)	8.07	5.17×10^{-4}	2.3
	$50^3 \times 12^3$ $= 125 \times 10^3$	54	2702s (45mn)	50.03	4.00×10^{-4}	15.4
	$100^3 \times 12^3$ $= 1.728 \times 10^9$	107	38069s (10.57h)	355.79	3.56×10^{-4}	115.4
extrapol.	$200^3 \times 12^3$ $= 1.3284 \times 10^{10}$	214	~ 633440s (7d)	~ 2960	3.7×10^{-4}	~ 900

Table 3.1 – *3D Sod shock tube. The time per cycle is obtained by $T_{cycle} = T/N_{cycle}$ and the time per cycle per cell by $T_{cell} = T/N_{cycle}/N_c$. The lines marked with extrapol. have been extrapolated by fixing N_c, N_{cycle} and T_{cell}.*

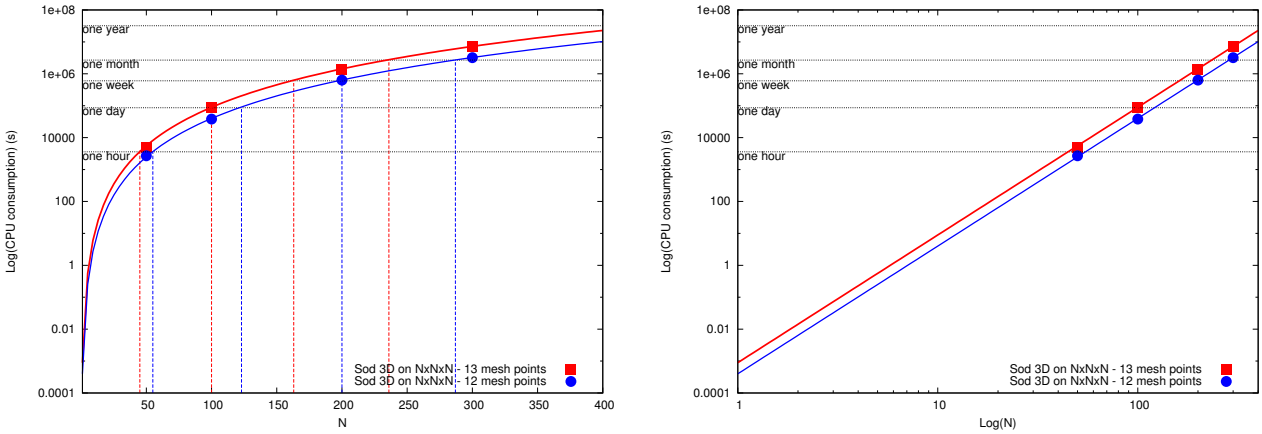


Figure 3.5 – *Left : Log of the CPU time consumption for the 3D Sod problem at $t_{final} = 0.1$ as a function of N (for $N \times N \times N$ cell meshes) on a single processor laptop The red/blue squares are taken from Table 3.1, the thick red/blue curves are the extrapolation curve from T_{cell}. The horizontal lines corresponding to one hour, one day, week, month and year are also plotted. $N = 100$ corresponds to the 'one million cells' in space — Right : Log/Log scale.*

system of equations which describes the fluid limit. Finally, we want to extend the method to other kinetic equations as the Boltzmann or the Vlasov equation and uses GPU infrastructure to speed-up such computation even more.

3.3 Interfaces in a finite volume scheme : Enhanced Natural Interface Positioning (ENIP)

Note that our goal in this section is not the description of the Finite Volume with Characteristic Flux scheme and its ultimate details and we refer the reader to the bibliography [210, 211, 212] for this matter. Instead we focus on the interface reconstruction technique that is a part of this scheme and for which we have proposed several improvements described in one article for the two-material case [18] *A totally Eulerian Finite Volume solver for multi-material fluid flows : Enhanced Natural Interface Positioning (ENIP)*, and one preprint for multi-material case [41] *Dealing with more than two materials in FVCF-ENIP method*.

The so-called ENIP technique is nowaday part of the FVCF scheme (Finite Volume with Characteristic Flux) introduced in [210] for simulating single phase compressible flows or multi-phase models without sharp interface capturing. This scheme has been supplemented with the so called NIP method (Natural Interface Positioning), see [212], to deal with multi-material fluid flows with sharp interface capturing. It is a cell centered totally Eulerian scheme, in which material interfaces are represented by a discontinuous piecewise linear curve. A treatment for interface evolution is proposed on Cartesian structured meshes which is locally conservative in mass, momentum and total energy and allow the materials to slide on each others. Discrete conservation laws are written on partial volumes as well as on pure cells, considering the interface in the cell as a moving boundary without any diffusion between materials. A specific data structure called *condensate* is introduced in order to write a finite volume scheme even when the considered volume is made of moving boundaries, i.e. interfaces. This treatment includes an explicit computation of pressure and velocity at interfaces. In [212], 2D results are shown illustrating the capability of the method to deal with perfect sliding, high pressure ratios and high density ratios. This former method however produces non satisfactory results in the context of advection of geometrical shapes especially when dealing with low Mach numbers. Generally speaking most of the advection and reconstruction methods have a tendency to destroy the shape of advected objects due to numerical approximations. This former method behaves similarly, but gives very poor results when advecting geometrical shapes especially when dealing with low Mach number flows. In this work, we have proposed a new method called ENIP (Enhanced NIP) that is an improvement of the NIP method by a more accurate treatment of condensates. In fact both NIP and ENIP are depicted in parallel in Fig.3.6. These pictures present the ideal example of a left-to-right advection of a cubic block of gray material. The condensate in x-direction made of the top part of the block is considered. On the left part of the figure is schematically presented the original method from J.-P. Braeunig *et al* [212] to deal with a condensate. On the right part of the figure we present our improvement of the method. The first four steps are identical to both approach.

(A) : the situation at t^n presents a block of material (gray) which uniquely defines the volume fractions of material in each cell. From the volume fractions an interface is reconstructed using Youngs' method [189], this is the blue segment associated with a unique blue normal.

(B) : a SLIC representation is built. In other words the material is represented by a vertical interface and located on either side of the cell according to the angle of the normal against a vertical line. There is an ambiguity when the interface is aligned with the condensate direction (cell 3 and 4 in Fig. 3.6). In such a case the location of the material on either side is based on roundoff.

(C) : the pieces of material in contact are glued together, as instance on both sides of cell interface j. The condensate is therefore a succession of material layers and interfaces labled with roman

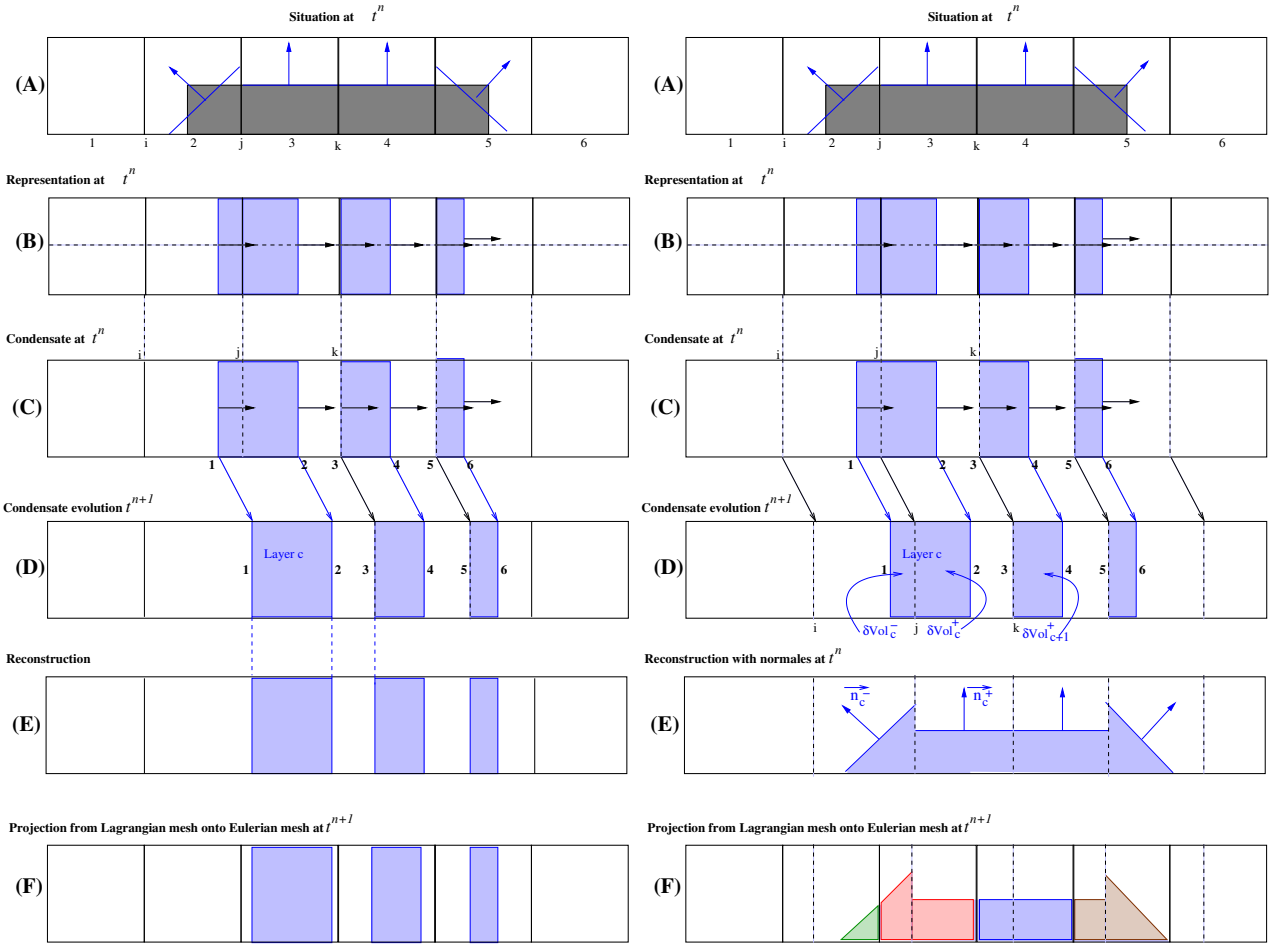


FIGURE 3.6 – *NIP vs ENIP method — **(A)** Situation at t^n with real materials geometry, interfaces and normals to them. **(B)** Representation of partial volumes at t^n. **(C)** Construction of a condensate at t^n by merging layers of contiguous partial volumes of the same material. **(D)** Evolution of condensate in a Lagrangian fashion during Δt. **(E)** Condensate reconstruction at t^{n+1}. **(F)** Condensate projection/remapping from Lagrangian mesh onto original mesh.*

numbers. Next the numerical scheme computes the material interface normal velocity (black arrows).

(D) : the interfaces/layers move with the previously computed normal velocities during the time step to reach t^{n+1}. For this step ENIP method also displaces the cell interface assuming a piece-wise linear velocity field along the condensate. Consequently the cells are pseudo-Lagrangian ones ; they may compress or expand.

(E) : this reconstruction step serves as retrieving a fair representation of the underlying material locations in the pseudo Lagrangian cells. This step did not exist in NIP. For ENIP we assume the normals to be unmodified during the time step, hence the t^n normals are considered.

(F) : materials are finally remapped back onto the Eulerian cells. While this step is obvious for NIP it is more demanding for ENIP because several polygonal shapes must be computed, see the same colored polygons which intersect to the same Eulerian cell.

Using a very simple example, the advection of a square, an inconsistency in the NIP interface

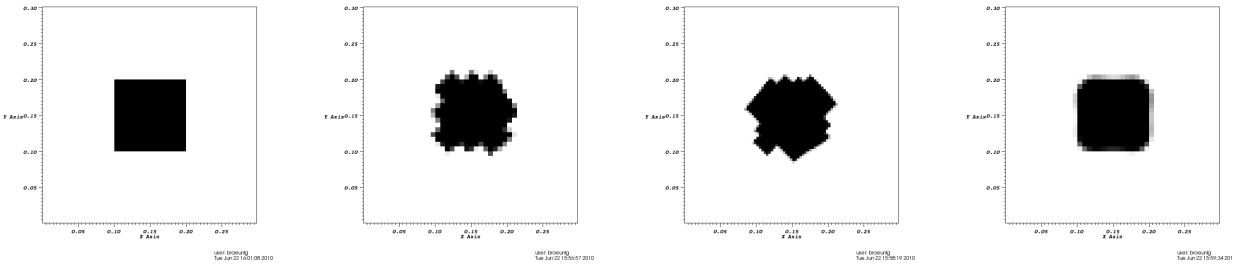


Figure 3.7 – *Inconsistency of NIP method and improvement gained by ENIP method — Advection of a square (zoom around the exact position of the initial and final square) — From left to right : exact solution, classical NIP with a* 60×60 *mesh, classical NIP with a* 120×120 *mesh and ENIP with a* 60×60 *mesh.*

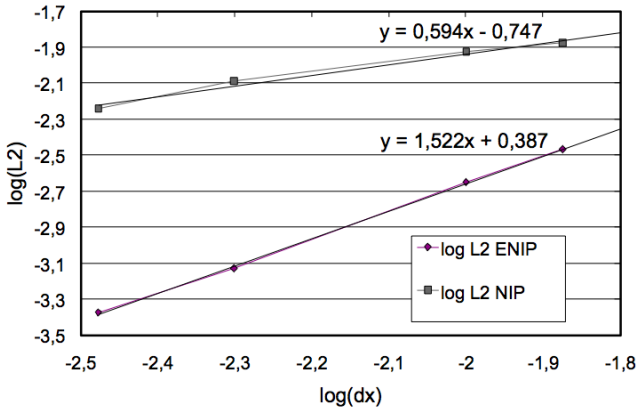


Figure 3.8 – *Convergence of ENIP vs NIP for a pure advection problem. The log of the* L_2 *error is displayed as a function of the log of* Δx*.*

reconstruction method is exhibited in Fig.3.7. An initial square $[0.1; 0.1] \times [0.2; 0.2]$ is located into the domain $\Omega = [0 : 0.4] \times [0; 0.6]$. The density into the square is set to $\rho_0(x) = 1$ whereas it is set to $\rho_0(x) = 0$ outside. In the pure advection context with a constant velocity (u, v) this square shape should be perfectly conserved . The exact solution at any point x and any time t is $\rho^{ex}(x, y, t) = \rho_0(x - u\ t, y - v\ t)$. The test consists in advecting the square with the constant velocity field $u = 1$, $v = 3$ up to the time $t = 0.1$ then reversing the advection field by setting $u = -1$, $v = -3$ up to final time $t = 0.2$ so that the final configuration exactly fits the initial one. In Figure 3.7 are shown the exact solution (top-left) and the results obtained with a 60×60 mesh for NIP (top-right) and ENIP (bottom-right). ENIP is visibly able to preserve the shape of the square whereas NIP is not. A mesh refinement of NIP computation (120×120 mesh for the bottom-left panel) does not improve the situation. If the numerical method provides an approximated solution called ρ_i^n in cell i at time t^n then the error in L_α norm is evaluated by ($\alpha = 1, 2$)

$$\varepsilon_\alpha = \frac{\sum_i |\rho_i^n - \rho^{ex}(x_i, t^n)|^\alpha}{\sum_i |\rho^{ex}(x_i, t^n)|^\alpha}. \tag{3.5}$$

The errors for the L_2 norm for successively refined meshes have been computed for both methods and, systematically ENIP over-tops NIP. Moreover in Figure 3.8 we display the log-log scale results for the error showing the improvement gained by ENIP ; indeed the slope which represents a measure of the numerical order of convergence is improved by a factor 2.5 (0.6 for NIP and 1.5 for ENIP).

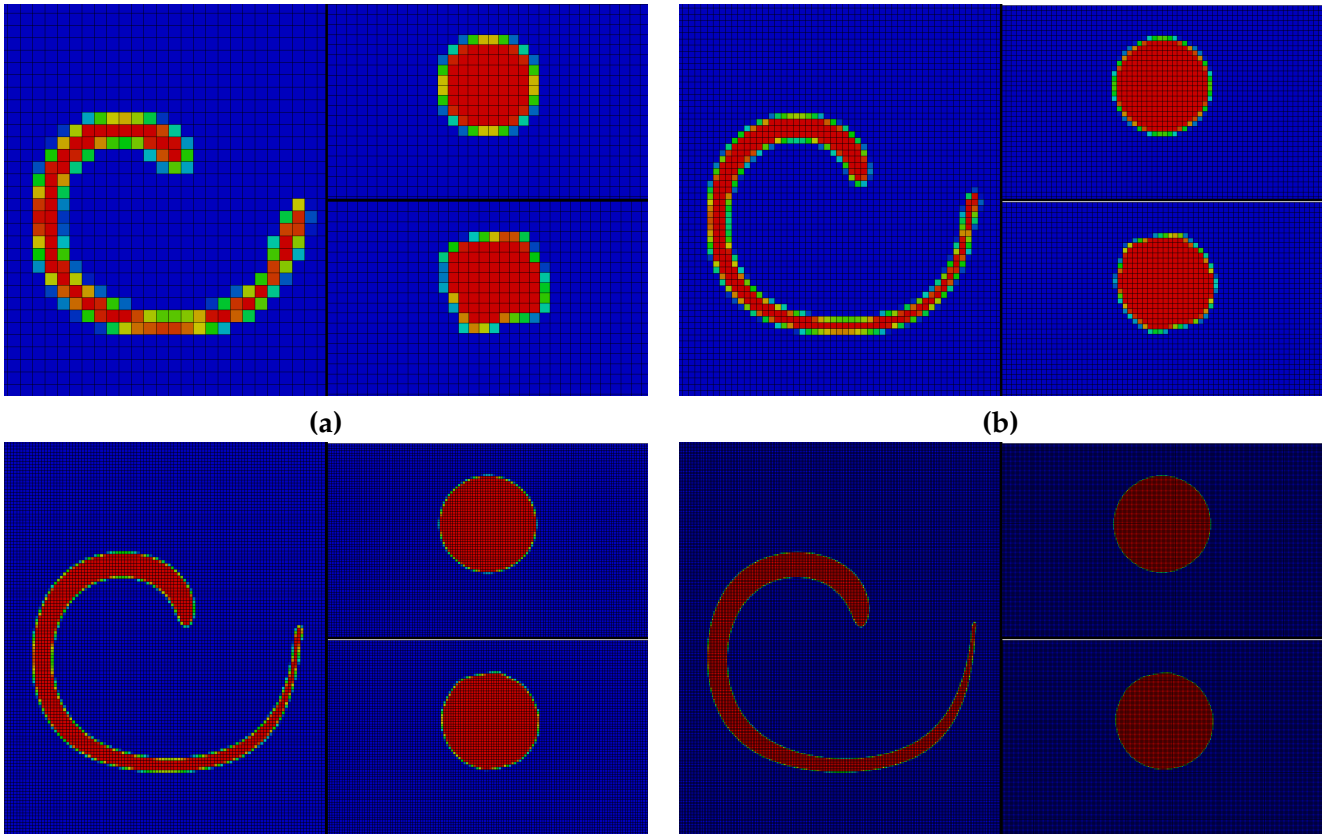


FIGURE 3.9 – *Disk embedded into a vortex problem — Results of ENIP method for the volume fractions (blue : material 1, red : material 2, any other color refers to a mixed cell).* ***(a)*** *:* 32×32 *mesh,* ***(b)*** *:* 64×64 *mesh,* ***(c)*** *:* 128×128 *mesh,* ***(d)*** *:* 256×256 *mesh. For comparison purposes for each mesh resolution we plot the initial* $t = 0$ *and final* $t = 4$ *times on the top of each other and on their left the time of maximal stretch* $t = 2$*.*

A standard volume tracking test case is the disk embedded into a vortex. It consists of a circle of radius 0.15 centered at $(0.5, 0.75)$. The computational domain is $\Omega = [0,1] \times [0,1]$. The mesh is a regular structured grid made of squares of size $\Delta x \times \Delta y$ with $\Delta x = \Delta y = 1/N$ with N the number of cells both in x and y directions. The incompressible velocity field is given by the streamfunction

$$\Psi = \cos\left(\frac{\pi}{4}t\right)\frac{1}{\pi}\sin^2(\pi x)\sin^2(\pi y) \tag{3.6}$$

with the velocity field defined to be $\boldsymbol{U} = (u, v) = (-\frac{\partial \Psi}{\partial y}, \frac{\partial \Psi}{\partial x})$. Due to the periodicity of Ψ, at time $t = 4$, the material configuration should be identical to the condition at time $t = 0$. The simulations were run to a final time of $t = 4.0$ with intermediate results at $t = 2.0$. Several succesively refined meshes are used : $N = 32, 64, 128$ and 256 and the results are plotted in Fig.3.9 for the initial $t = 0$, maximal stretch $t = 2$ and final $t = 4$ times. A two material disk is considered, one material is indexed by 2 in the disk (red color) and the second one is indexed by 1 in the surrounding (blue color). The results of colored volume fractions are shown, any color different from red and blue indicates a mixed cell. Numerical investigations on this test case have shown that the method is only first order accurate. However its extension to deal with more than two materials is almost trivial. In a forthcoming preprint [41] we have tested this approach with the simple onion-skin

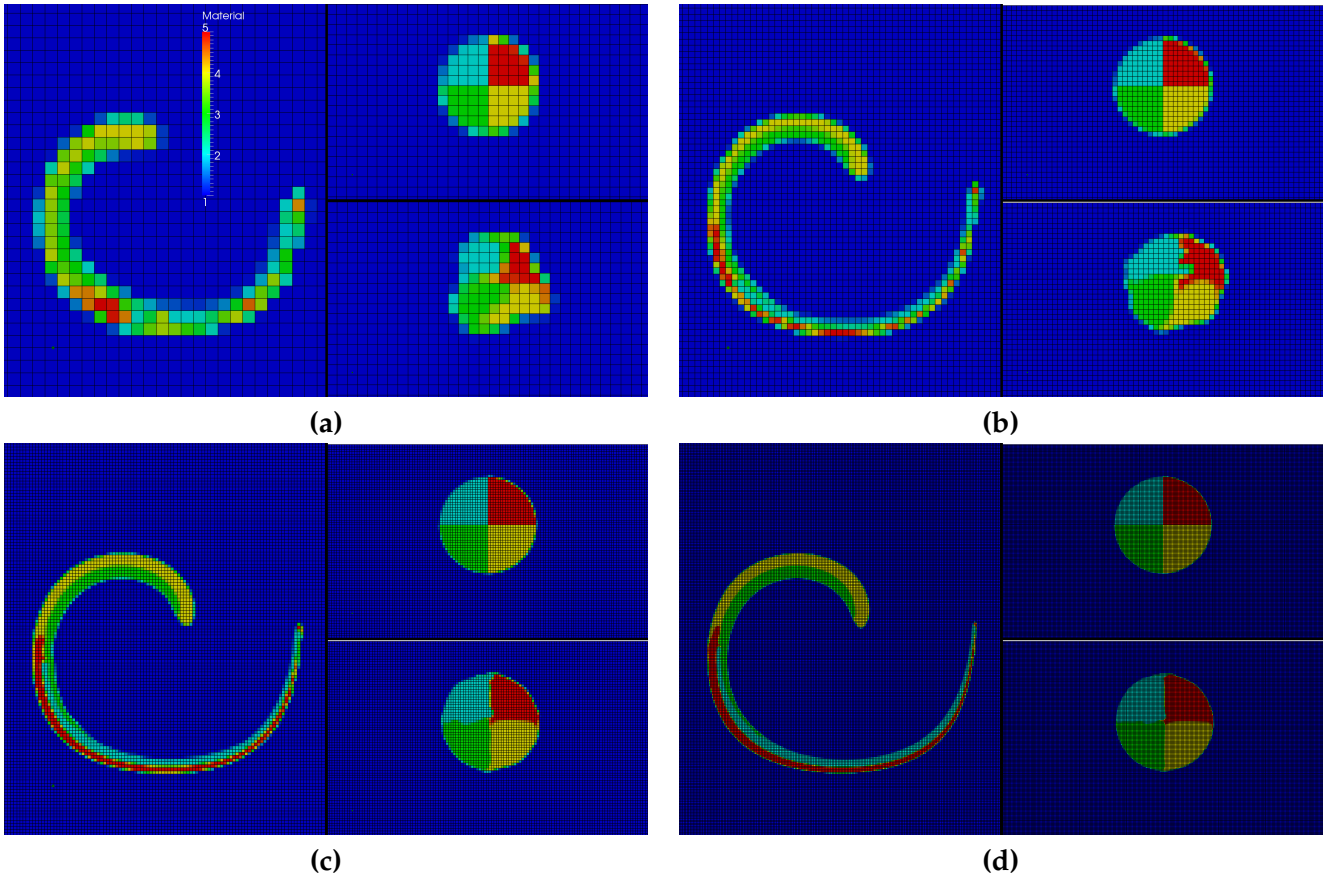


Figure 3.10 – *Four material disk embedded into a vortex problem — Results of ENIP method for the volume fractions.* ***(a)*** *:* 32×32 *mesh,* ***(b)*** *:* 64×64 *mesh,* ***(c)*** *:* 128×128 *mesh,* ***(d)*** *:* 256×256 *mesh. For comparison purposes for each mesh resolution we plot the initial* $t = 0$ *and final* $t = 4$ *times on the top of each other and on their left the time of maximal stretch* $t = 2$*.*

interface reconstruction method however more accurate method (MOF [195], Power diagram [17]) can also be considered within this framework. As an illustration we present the volume fractions in Fig. 3.10 for the vortex test case but with a four material disk (see the initial disk for each resolution). The disk material indexes are 2 (light blue), 3 (green), 4 (yellow) and 5 (red) and the surrounding material is labeled by 1 (navy blue). Each cell with a mixed color is indeed a mixed cell. The general shape of the materials is rather well preserved.

Conclusions and perspectives

This thesis has presented some of the works my collaborators and I were able to pursue. Most of these investigations revolve around a joined effort to improve ALE simulation codes; Lagrangian scheme analysis and test, rezone and reconnect strategies, remapping algorithms, multi-material treatment and interface reconstruction techniques. My goal in this thesis was not only to provide some details of these works but also to provide 'the big picture behind the scene' and to try when possible to give some historical context. Moreover I have introduced several other topics which are not genuinely related to ALE such as the so-called very high-order MOOD method, some ultra fast kinetic scheme and multi-material finite volume VFFC scheme.

This is some of the current research pursued and all of these subjects are still under investigation. This helps a lot when perspectives are to be drawn ! Everything in what has been presented in this thesis is far from begin satisfactory, and improvements will be searched in the near future.
One of my goals is to construct a 3D version of a reconnection ALE code that could run on a single workstation with "light" parallelization technique. By using the most appropriate and efficient numerical methods we should be able to build such a tool for the community. As a consequence the Lagrangian schemes, the rezoner and the remapper must be somewhat revamped to adapt to the inherent difficulties of a 3D simulation code. In short, more efficient remappers (less memory-consuming), more stable Lagrangian schemes and trully poyhedral rezoner must be designed.

I pay a lot of attention to the work on bridging cell-centered and staggered Lagrangian schemes, in fact I do believe that this is a good way to improve the schemes by feeding them with success from the others. After all these formulations are probably not so far away from each others. Moreover comparing numerical methods is a thankless activity but absolutely needed to deeply understand their intrinsic behaviors which, at the end of the day, only reveal when difficult tests cases are simulated.

Unavoidable the ALE formulation is fruitful and applying this machinery to other physical contexts is kicking : elasto-plasticity, interaction of fluid/structure as instance. Before being able to properly treat these subjects we surely will have to define a good remapper for tensors. Being able to reconstruct a high-order accurate representation of a tensor, being able to limit this and begin able to remap onto a new grid are some tools that we need to construct. All of these require some careful design and study which are on my "to do" list.

Concerning the context of very high-order finite volume schemes, we are currently extruding the

essence of the original MOOD method ; it seems to be a more general design principle to build numerical schemes than a numerical method devoted to hydrodynamics. The MOOD paradigm can virtually be extended to treat steady-state solutions, parabolic/elliptic system of equations, source terms. Moreover developing a MOOD-like method mixing different schemes such as unlimited Discontinuous Galerkin schemes may be interesting. We plan to pair up our MOOD scheme with ADER like techniques. An implicit version of the MOOD scheme is also currently being worked out. Finally the validation of the MOOD concept on genuine physical contexts is under investigations. Concentrating our experiences on MOOD into a dedicated library for the community is also a wishful thinking of mine. Undoubtedly this topic will occupy some of my research time.

Because for the next years I do not plan to change my natural tendency to take advantages of good research topics, it is irrefutable that my work will also cover some other eclectic subjects.

Appendix

Abbreviations and acronyms

The abbreviations and acronyms used in this document are listed in the follwoing table.

AWE	:	Atomic Weapon Establisment, Aldermaston, U.K
CEA	:	"Commisariat à l'énergie atomique" in France
CEA-DIF	:	CEA center of Bruyères-le-Châtel, France
CEA-CESTA	:	CEA center of Le Barp, France
CELIA	:	"Centre d'Etude Laser et Interaction et Applications", University of Bordeaux, France
CMLA	:	"Centre de Mathématiques et de Leurs Applications", ENS-Cachan, Paris, France
CNRS	:	"Centre National de Recherche Scientifique" in France
CVUT	:	Czech Technical University in Prague, Czech Republic
DGA	:	"Direction Générale de L'Armement" in France
IMB	:	"Institut de Mathématiques de Bordeaux" in France
IMT	:	"Institut de Mathématiques de Toulouse" in France
INRIA	:	"Institut National de Recherche en Informatique et Automatique" in France
LANL	:	Los Alamos National Laboratory, Los Alamos, New Mexico, U.S.A
LLNL	:	Lawrence Livermore National Laboratory, Livermore, California, U.S.A
LMJ	:	Mega-joule laser located at the CEA-CESTA in France
MIP	:	"Mathématiques pour l'Industrie et la Physique" group within IMT
NIF	:	National Ignition Facility laser located at the LLNL

Bibliography

[1] R. Loubère. *Une méthode particulaire Lagrangienne de type Galerkin discontinu. Application à la mécanique des fluides et à l'interaction laser plasma*. PhD thesis, Université Bordeaux I, 2002. (Cited page 1.)

[2] R. Abgrall, R. Loubère, and J. Ovadia. A Lagrangian Discontinuous Galerkin-type method on unstructured meshes to solve hydrodynamics problems. *Int. J. Numer. Meth. Fluids*, 44 :645–663, 2004. (Cited page 1.)

[3] Weber S., G. Riazuelo, P. Michel, R. Loubère, F. Walraet, V. T. Tikhonchuk, V. Malka, J. Ovadia, and G. Bonnaud. Modeling of laser-plasma interaction on hydrodynamics scales : Physics developments and comparison with experiments. *Laser and particle beams*, 22 :189–195, 2004. (Cited page 1.)

[4] S. Weber, P.-H. Maire, R. Loubère, G. Riazuello, P. Michel, J. Ovadia, and V. Tikhonchuk. A transport simulation code for inertial confinement fusion (ICF) relevant laser/plasma interaction. *Comp. Phys. Commun.*, 168 :141–158, 2005. (Cited page 1.)

[5] R. Loubère. First steps into ALE INC(ubator) - version 2.0.0. Technical Report LAUR-04-8840, Los Alamos National Laboratory Report, 2005. see also R. Loubère, M. Shashkov, 2D Arbitrary-Lagrangian-EUlerian (ALE) code on polygonal grids for shock wave simulation, Khimicheskaya Fizika, 2005, 24 (2), pp 19-31. (Cited page 1, 55, 58 et 102.)

[6] R. Loubère and M. Shashkov. 2D arbitrary-Lagrangian-Eulerian (ALE) code on polygonal grids for shock wave simulation. *Khimicheskaya Fizika*, 24(2) :19–31, 2005. (Cited page 1.)

[7] R. Loubère and M. Shashkov. A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods. *J. Comput. Phys.*, 209(1) :105–138, 2006. (Cited page 1, 5, 52, 56, 58 et 59.)

[8] R. Loubère, M. Staley, and B.Wendroff. The repair paradigm : New algorithms and applications to compressible flow. *J. Comput. Phys.*, 1(2) :385–404, 2006. (Cited page 1, 5, 57, 58, 62 et 95.)

[9] B. Després and R. Loubère. Convergence and sensitivity analysis of repair algorithms in 1D. *Int. J. Finit. Vol.*, 3(1), 19 January 2006. (Cited page 1, 5, 57, 61 et 95.)

[10] R. Loubère and E. J. Caramana. The force/work differencing of exceptional points in the discrete, compatible formulation of Lagrangian hydrodynamics. *J. Comput. Phys.*, 216(1) :1–18, 2006. (Cited page 1, 5, 7, 34, 35, 37, 38 et 39.)

[11] E. J. Caramana and R. Loubère. "Curl-q" : A vorticity damping artificial viscosity for Lagrangian hydrodynamics calculations. *J. Comput. Phys.*, 215(2) :385–391, 2006. (Cited page 1, 5, 7, 34, 36 et 37.)

[12] A. L. Bauer, D. E. Burton, E. J. Caramana, R. Loubère, M. J. Shashkov, and P. P. Whalen. The internal consistency, stability, and accuracy of the discrete, compatible formulation of Lagrangian hydrodynamics. *J. Comput. Phys.*, 218(2) :572–593, 2006. (Cited page 2, 5, 7, 8, 9, 12, 15, 16, 31, 32, 33, 34 et 55.)

[13] S.P. Schofield, R.V. Garimella, M.M. Francois, and R. Loubère. Material order independent interface reconstruction using power diagrams. *International Journal for Numerical Methods in Fluids*, 56 :643–659, 2008. (Cited page 2, 6, 83, 86, 88 et 91.)

[14] M. Kuchařík, R. Liska, R. Loubère, and M. Shashkov. Arbitrary-Lagrangian-Eulerian (ALE) method in cylindrical coordinates for laser/plasma simulation. In Sylvie Benzoni-Gavage and Denis Serre, editors, *Hyperbolic problems : Theory, Numerics, Application, Proceedings of the 11th Int. Conf on Hyperbolic problems, ENS Lyon July 17-21*. Springer-Verlag Berlin Heidelberg, 2006. (Cited page 2.)

[15] A.L. Bauer, R. Loubère, and B.B. Wendroff. On stabiliy analysis of staggered schemes. *SIAM Journal on Numerical Analysis*, 46 :996–1011, 2008. (Cited page 2, 5, 31, 32 et 33.)

[16] R. Loubère, M. Shashkov, and B. Wendroff. Volume consistency in a staggered grid Lagrangian hydrodynamics scheme. *J. Comput. Phys.*, 227 :3731–3737, 2008. (Cited page 2, 5, 31, 34 et 35.)

[17] Samuel P. Schofield, Rao V. Garimella, Marianne M. Francois, and Raphaël Loubère. A second-order accurate material-order-independent interface reconstruction technique for multi-material flow simulations. *J. Comput. Phys.*, 228 :731–745, 2009. (Cited page 2, 6, 83, 88, 89, 90, 91 et 110.)

[18] Jean-Philippe Braeunig, Jean-Michel Ghidaglia, and Raphaël Loubère. A totally Eulerian finite volume solver for multi-material fluid flows : Enhanced natural interface positioning (ENIP). *European Journal of Mechanics - B/Fluids*, 31 :1–11, 2012. (Cited page 2, 6, 94 et 106.)

[19] C. Fochesatto, R. Loubère, R. Motte, and J. Ovadia. An interface reconstruction method devoted to filament. *Computer and Fluids*, 2012. to be submitted. (Cited page 2.)

[20] Richard Liska, Raphaël Loubère, Pierre-Henri Maire, Jerome Breil, Stephane Galera, and Pavel Váchal. Comparison of staggered and cell-centered Lagrangian and ALE hydrodynamical methods. In Tadmor, E and Liu, J and Tzavaras, A, editor, *HYPERBOLIC PROBLEMS : THEORY, NUMERICS AND APPLICATIONS, PART 2*, volume 67, Part 2 of *Proceedings of Symposia in Applied Mathematics*, pages 755–764, 2009. 12th International Conference on Hyperbolic Problems, College Park, MD, JUN 09-13, 2008. (Cited page 2.)

[21] R. Loubère, P.-H. Maire, and P. Váchal. Formulation of a staggered two-dimensional Lagrangian scheme by means of cell-centered approximate riemann solver. In Maya Neytcheva Editors Gunilla Kreiss Per Lötstedt, Axel Malqvist, editor, *Proceedings of ENUMATH 2009, the 8th European Conference on Numerical Mathematics and Advanced Applications, Uppsala, Sweden*. Springer, 2009. (Cited page 2 et 42.)

[22] Raphaël Loubère, Pierre-Henri Maire, and Pavel Váchal. A second-order compatible staggered Lagrangian hydrodynamics scheme using a cell-centered multidimensional approximate Riemann solver. In *ICCS 2010 - INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE, PROCEEDINGS*, volume 1 of *Procedia Computer Science*, pages 1925–1933, 2010. International Conference on Computational Science(ICCS), Amsterdam, NETHERLANDS, MAY 31-JUN 02, 2010. (Cited page 2 et 42.)

[23] R. Loubère, P.-H. Maire, and P. Váchal. Staggered Lagrangian discretization based on cell-centered riemann solver and associated hydro-dynamics scheme. *Communication in Computational Physics*, 10(4) :940–978, 2011. doi :10.4208/cicp.170310.251110a. (Cited page 2, 5, 35, 42, 45 et 52.)

[24] P. Váchal R. Loubère, P.-H. Maire. 3D staggered Lagrangian hydrodynamics with cell-centered riemann solver based artificial viscosity. *Int. J. Numer. Meth. Fluid*, 72(13) :22–42, 2013. Article first published online : 27 SEP 2012, DOI : 10.1002/fld.3730. (Cited page 3, 5, 35, 42, 45, 46, 47 et 48.)

[25] M. Kuchařík, R. Loubère, L. Bednárik, and R. Liska. Enhancement of Lagrangian slide lines as a combined force and velocity boundary condition. *Computer and Fluids*, in press :–, 2012. available on line 15 juin 2012. (Cited page 3, 5, 15, 35 et 40.)

[26] R. Loubère, P.-H Maire, M. Shashkov, J. Breil, and S. Galera. ReALE : A Reconnection-based Arbitrary-Lagrangian-Eulerian method. *J. Comput. Phys.*, 229 :4724–4761, 2010. (Cited page 3, 6, 63, 65, 69, 72, 73, 74, 76 et 77.)

[27] R. Loubère, P.-H. Maire, and M. Shashkov. ReALE : A reconnection arbitrary-Lagrangian-Eulerian method in cylindrical geometry. *Computer and Fluids*, 46 :59–69, 2011. doi :10.1016/j.compfluid.2010.08.024. (Cited page 3, 6, 63, 74, 75, 78 et 79.)

[28] L. Carballal-Perdiz, P. Degond, F. Deluzet, R. Loubère, and J.-M. Rovarch. PRACT-X : PRediction of Air Contaminant Transport on multiple (X) scales. Technical report, Rapport DGA - Convention 2007 - 25 - 011, 2009. in preparation with A. Lozinski "Multiscale finite element method for perforated domains". (Cited page 3.)

[29] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for hyperbolic systems : Multi-dimensional Optimal Order Detection (MOOD). *J. Comput. Phys.*, 230(10) :4028–4050, 10 May 2011. (Cited page 3, 6, 93 et 96.)

[30] S. Clain, S. Diot, and R. Loubère. Multi-dimensional Optimal Order Detection (MOOD) - A very high-order finite volume scheme for conservation laws on unstructured meshes. In J. Fort, J. Fürst, J. Halama, R. Herbin, and Hubert, editors, *FVCA 6, International Symposium, Prague, June 6-10, 2011*, volume 4 of *Proceedings in Mathematics, 1st Edition*. Springer, 2011. 1065 p. 106 illus. in color. (Cited page 3 et 93.)

[31] S. Clain, S. Diot, R. Loubère, G. Machado, R. Ralha, and R.M.S. Pereira. Very high-order finite volume method for one-dimensional convection diffusion problems. In *MATHEMATICAL MODELS FOR ENGINEERING SCIENCE (MMES'11), International conference of the Institute for Environment, Engineering, Economics and Applied Mathematics (IEEEAM)*, 2012. (Cited page 3 et 93.)

[32] S. Diot, S. Clain, and R. Loubère. Improved detection criteria for the multi-dimensional optimal order detection (MOOD) on unstructured meshes with very high-order polynomials. *Computer and Fluids*, 64 :43–63, 2012. http ://dx.doi.org/10.1016/j.compfluid.2012.05.004. (Cited page 3, 6, 93 et 96.)

[33] R. Loubère S. Clain, S. Diot. The MOOD method in the three-dimensional case : Very- high-order finite volume method for hyperbolic systems. *Int. J. Numer. Meth. Fluid*, 2013. accepted in January 2013. (Cited page 3, 6, 93 et 96.)

[34] R. Loubère. Etude des schémas numériques ALE multi matériaux 1. Technical report, Rapport final du contrat d'étude liant de CEA-DAM et l'IMT, 2007. 101 pages. (Cited page 3.)

[35] R. Loubère. Etude des schémas numériques ALE multi matériaux 2. Technical report, Rapport final du contrat d'étude liant de CEA-DAM et l'IMT, 2008. 78 pages. (Cited page 3.)

[36] R. Loubère. Etude des schémas numériques ALE multi matériaux 3. Technical report, Rapport final du contrat d'étude liant de CEA-DAM et l'IMT, 2009. 144 pages. (Cited page 3.)

[37] R. Loubère. Etude des schémas numériques ALE multi matériaux 4. Technical report, Rapport final du contrat d'étude liant de CEA-DAM et l'IMT, 2010. 97 pages. (Cited page 3.)

[38] R. Loubère. Etude des schémas numériques ALE multi matériaux 5. Technical report, Rapport final du contrat d'étude liant de CEA-DAM et l'IMT, 2011. 67 pages. (Cited page 3.)

[39] G. Dimarco and R. Loubère. Towards an ultra efficient kinetic scheme part I : basics on the bgk equation. *J. Comput. Phys.*, 2012. accepted. (Cited page 3, 6, 93, 99 et 100.)

[40] G. Dimarco and R. Loubère. Towards an ultra efficient kinetic scheme part I : the high-order case. *J. Comput. Phys.*, 2013. submitted. (Cited page 3.)

[41] Bruno Blais, Jean-Philippe Braeunig, Jean-Michel Ghidaglia, and Raphaël Loubère. Dealing with more than two materials in FVCF-ENIP method. *submitted to European Journal of Mechanics - B/Fluids*, 2012. (Cited page 94, 106 et 109.)

[42] P.-H. Maire. Contribution à la modélisation numérique de la fusion par confinement inertiel. *Habilitation à Diriger des Recherches, Université de Bordeaux*, 2011. (Cited page 7, 12, 28 et 42.)

[43] B. Després and C. Mazeran. Lagrangian gas dynamics in two dimensions and Lagrangian systems. *Arch. Rational Mech. Anal.*, 178 :327–372, 2005. (Cited page 7 et 17.)

[44] G. Carré, S. Delpino, B. Després, and E. Labourasse. A cell-centered Lagrangian hydrodynamics scheme in arbitrary dimension. *J. Comput. Phys.*, 228(14) :5160–5183, 2009. (Cited page 7.)

[45] P.-H. Maire and B. Nkonga. Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. *J. Comput. Phys.*, 228(3) :799–821, 2009. (Cited page 7 et 69.)

[46] A.J. Barlow. A compatible finite element multi-material ALE hydrodynamics algorithm. *International Journal for Numerical Methods in Fluids*, 56 :953–964, 2008. (Cited page 7.)

[47] B. Rebourcet. Comments on the filtering of numerical instabilities in Lagrangian hydrocodes. Conference on Numerical methods for multi-material fluid flows ; Czech Technical University in Prague on September 10 - 14, 2007 ; http://www-troja.fjfi.cvut.cz/ multimat07/presentations/tuesday/Rebourcet_filtering.pdf. (Cited page 7.)

[48] P.-H. Maire, R. Abgrall, J. Breil, and J. Ovadia. A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *SIAM J. Sci. Comput.*, 29(4) :1781–1824, 2007. (Cited page 7 et 42.)

[49] P.-H. Maire and J. Breil. A second-order cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *Int. J. Numer. Meth. Fluids*, 56 :1417–1423, 2008. (Cited page 7, 42 et 69.)

[50] P.-H. Maire. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *J. Comput. Phys.*, 228 :2391–2425, 2009. (Cited page 7, 17, 42 et 69.)

[51] A. Barlow and P. Roe. A cell centred lagrangian godunov scheme for shock hydrodynamics. *Comput. Fluids*, 46(1) :133–136, 2011. (Cited page 7, 17 et 42.)

[52] D. J. Benson. Computational methods in Lagrangian and Eulerian hydrocodes. *Comp. Meth. Appl. Mech. Engrg.*, 99 :235–394, 1992. (Cited page 7, 17, 49, 52, 53, 58 et 82.)

[53] J. C. Campbell and M. J. Shaskhov. A tensor artificial viscosity using a mimetic finite difference algorithm. *J. Comput. Phys.*, 172(4) :739–765, 2001. (Cited page 7, 17, 20, 21, 36, 38 et 69.)

[54] J. C. Campbell and M. J. Shashkov. A compatible Lagrangian hydrodynamics algorithm for unstructured grids. *SelÃğuk J. Appl. Math.*, 4(2) :53–70, 2003. (Cited page 7 et 69.)

[55] E. J. Caramana, D. E. Burton, M. J. Shashkov, and P. P. Whalen. The construction of compatible hydrodynamics algorithms utilizing conservation of total energy. *J. Comput. Phys.*, 146(1) :227–262, 1998. (Cited page 7, 11, 12, 13, 15, 21, 28, 29, 30, 55 et 58.)

[56] E. J. Caramana and M. J. Shashkov. Elimination of artificial grid distorsion and hourglass–type motions by means of Lagrangian subzonal masses and pressures. *J. Comput. Phys.*, 142 :521–561, 1998. (Cited page 7, 15, 22 et 56.)

[57] E. J. Caramana, M. J. Shashkov, and P. P. Whalen. Formulations of artificial viscosity for multidimensional shock wave computations. *J. Comput. Phys.*, 144 :70–97, 1998. (Cited page 7, 17, 18, 19, 34, 36, 38 et 46.)

[58] G. Scovazzi, M. A. Christon, T. J.R. Hughes, and J. N. Shadid. Stabilized shock hydrodynamics : I. A Lagrangian method. *Comp. Meth. Appl. Mech. Engrg.*, 196 :923–966, 2007. (Cited page 7.)

[59] G. Scovazzi. Stabilized shock hydrodynamics : II. Design and physical interpretation of the SUPG operator for Lagrangian computations. *Comp. Meth. Appl. Mech. Engrg.*, 196 :966–978, 2007. (Cited page 7.)

[60] G. Scovazzi, E. Love, and M. J. Shashkov. Multi-scale Lagrangian shock hydrodynamics on Q1/Po finite elements : Theoretical framework and two-dimensional computations. *Comp. Meth. Appl. Mech. Engrg.*, 197 :1056–1079, 2008. (Cited page 7.)

[61] V.A. Dobrev, T.E. Ellis, Tz. V. Kolev, and R. N. Rieben. Curvilinear finite elements for Lagrangian hydrodynamics . *Int. J. Numer. Meth. Fluid*, 2010. doi : 10.1002/fld.2366. (Cited page 7.)

[62] Tz.V. Kolev and R.N. Rieben. A tensor artificial viscosity using a finite element approach. *J. Comput. Phys.*, 228(22) :8336–8366, 2010. (Cited page 7 et 17.)

[63] E. J. Caramana and P. P.Whalen. Numerical preservation of symmetry properties of continuum problems. *J. Comput. Phys.*, 141 :174–198, 1998. (Cited page 7, 29 et 55.)

[64] E. J. Caramana, C. L. Rousculp, and D. E. Burton. A compatible, energy and symmetry preserving Lagrangian hydrodynamics algorithm in three-dimensional Cartesian geometry. *J. Comput. Phys.*, 157 :89–119, 2000. (Cited page 7 et 55.)

[65] R. D. Richtmyer and K. W Morton. *Difference Methods for Initial Value Problems 2nd ed.* Interscience Publishers, 1967. (Cited page 8 et 30.)

[66] H.A. Bethe, R. Christy, R.R. Davis, R.P. Feynman, S. Frankel, E.J. Konopinski, N. Livesay, N. Metropolis, and E. Nelson. IBM calculations of implosion hydrodynamics. Technical report, Los Alamos National Laboratory report, LA-94, June 20 1944. unclassified, cover only. (Cited page 8.)

[67] J. von Neumann and R. D. Richtmyer. A method for the numerical calculations of hydrodynamical shocks. *J. Appl. Phys.*, 21 :232–238, 1950. (Cited page 8 et 17.)

[68] R. D. Richtmyer and K. W. Morton. *Difference methods for initial-value problems*. John Wiley, Interscience Publishers, 1967. (Cited page 8.)

[69] M. Rotenberg (Eds.) B. Alder, S. Fernbach, editor. *Methods in Computational Physics, Advances in Research and Applications,Fundamental Method in Hydrodynamics*, volume 3. Academic Press, 1964. (Cited page 8, 9, 29 et 31.)

[70] W.B. Goad. WAT : a numerical method for two-dimensional unsteady fluid flow. Technical Report LAMS-2365, Los Alamos National Laboratory Report, 1960. (Cited page 8.)

[71] A.P. Favorskii. Variational-discrete models of hydrodynamics equations. *Diff. Equations*, 16 :1308–1321, 1980. (Cited page 9 et 31.)

[72] T.F. Adams L.G. Margolin. Spatial differencing for finite difference codes. Technical Report LA-10249, Los Alamos National Laboratory Repor, 1985. (Cited page 9 et 31.)

[73] A.A. Samarskii, V.F. Tishkin, A.P. Favorskii, and M.J. Shashkov. Operational finite difference schemes. *Diff. Equations*, 17 :854–862, 1981. (Cited page 9.)

[74] D. E. Burton. Exact conservation of energy and momentum in staggered-grid hydrodynamics with arbitrary connectivity. *Advances in the Free Lagrange Method*, Springer-Verlag, New-York, 1990. (Cited page 9, 12, 15 et 69.)

[75] D. E. Burton. Multidimensional Discretization of Conservation Laws for Unstructured Polyhedral Grids. Technical Report UCRL-JC-118306, Lawrence Livermore National Laboratory, 1994. (Cited page 9 et 49.)

[76] L.G. Margolin, M.J. Shashkov, and M.A. Taylor. Symmetry-preserving discretizations for Lagrangian gas dynamics. In P. Neittaanmäaki, T. Tiihonen, and P. Tarvainen, editors, *Proceedings of the 3rd European Conference, Numerical Mathematics and Advanced Applications*, pages 725–732. World Scientific, 2000. (Cited page 12 et 13.)

[77] M. Shashkov. *Conservative finite difference methods on general grids*. CRC Press, 1996. (Cited page 13.)

[78] D. E. Burton. Multidimensional discretization of conservation laws for unstructured polyhedral grids. *Report UCRL-JC-118306, Lawrence Livermore National Laboratory*, 1994. Proceedings of Second International Workshop on Analytical Methods and Process Optimization in Fluid and Gas Mechanics (SAMGOP), Arzamas-16, Russia. (Cited page 15 et 69.)

[79] P.-H. Maire, R. Abgrall, J. Breil, R. Loubère, and B. Rebourcet. A nominally second-order cell-centered lagrangian scheme for simulating elastic-plastic flows on two-dimensional unstructured grids. *J. Comput. Phys.*, 235 :626–665, 2013. also available at http://hal.inria.fr/hal-00701802. (Cited page 15.)

[80] E. J. Caramana. The implementation of slide lines as a combined force and velocity boundary condition. *J. Comput. Physics*, 228(11) :3911–3916, 2009. (Cited page 15, 35, 40 et 41.)

[81] R. Landshoff. A numerical method for treating fluid flow in the presence of shocks. Technical Report LA-1930, Los Alamos National Laboratory, 1955. (Cited page 17.)

[82] W.D. Schulz. *Methods in Computational Physics*, volume 3, chapter Two-dimensional Lagrangian hydrodynamic difference equations, pages 1–45. Academic Press, 1964. (Cited page 17, 18 et 29.)

[83] W.D. Schulz. Tensor artificial viscosity for numerical hydrodynamics. *J. Math. Phys.*, 1964. (Cited page 17 et 18.)

[84] G.T. Richards. Derivation of a generalized von neumann pseudo-viscosity with directional properties. Technical Report UCRL-14244, Lawrence Livermore Radiation Laboratory, 1965. (Cited page 17.)

[85] M. L. Wilkins. Use of artificial viscosity in multidimensionnal fluid dynamic calculations. *J. Comput. Phys.*, 36 :381–403, 1980. (Cited page 17.)

[86] V.F. Kurapatenko. *Difference Methods for Solutions of Problems of Mathematical Physics*, chapter -. Janenko, N. N., American Mathematical Society, Providence, RI,, 1967. (Cited page 17.)

[87] R. B. Christensen. Godunov methods on a staggered mesh - an improved artificial viscosity. Technical Report UCRL-JC-105269, Lawrence Livermore National Laboratory, 1991. (Cited page 17 et 44.)

[88] J. K. Dukowicz. A general, non-iterative Riemann solver for Godunov's method. *J. Comput. Phys.*, 61 :119–137, 1984. (Cited page 17 et 43.)

[89] J. K. Dukowicz and B. Meltz. Vorticity errors in multidimensional Lagrangian codes. *J. Comput. Phys.*, 99 :115–134, 1992. (Cited page 17.)

[90] W. F. Noh. Errors for calculations of strong shocks using artificial viscosity and an artificial heat flux. *J. Comput. Phys.*, 72 :78–120, 1987. (Cited page 17.)

[91] D. J. Benson. A new two-dimensional flux-limited shock viscosity for impact calculations. *Computer Methods in Applied Mechanics and Engineering*, 93(1) :39 – 95, 1991. (Cited page 17.)

[92] K.Lipnikov. A framework for developing a mimetic tensor artificial viscosity for lagrangian hydrocodes on arbitrary polygonal meshes. *J. Comput. Phys.*, 229 :7911–7941, 2010. (Cited page 17.)

[93] G.Maenchen and S. Sack. *Methods in Computational Physics*, volume 3, chapter The Tensor code, pages 181–210. Academic Press, 1964. (Cited page 17.)

[94] M.Owen. A tensor artificial viscosity for SPH. *J. Comput. Phys.*, 201(2) :601 – 629, 2004. (Cited page 17.)

[95] R. Loubère. On the effect of the different limiters for the tensor artificial visosity for the compatible langrangian. Technical report, Los Alamos report, LAUR-05-9301, 2005. (Cited page 21.)

[96] G. Scovazzi and E. Love. A generalized view on galilean invariance in stabilized compressible flow computations. *Int. J. Numer. Meth. Fluid*, 64(10-12) :1065–1083, 2010. (Cited page 22.)

[97] M. Kuchařík. *Arbitrary-Lagrangian-Eulerian (ALE) methods in plasma physics*. PhD thesis, Czech technical University in Prague, 2006. (Cited page 29.)

[98] M.L. Wilkins. *Methods in Computational Physics*, volume 3, chapter Calculation of elastic-plastic flows, pages 211–263. Academic Press, 1964. (Cited page 29.)

[99] V. F. Tishkin, N. N. Tiurina, and A. P. Favorskii. Finite-difference schemes for calculating hydrodynamic flows in cylindrical coordinates. *preprint, No. 23, Keldysh Institute of Applied Mathematics, Moscow, Russia*, 1978. (Cited page 29.)

[100] N. V. Mikhailova, V. F. Tishkin, N. N. Tyurina, A. P. Favorskii, and M. Y. Shashkov. Numerical modelling of two-dimensional gas-dynamic flows on a variable-structure mesh. *USSR Comput. Math. Math. Phys.*, 26(5) :74–84, 1988. (Cited page 29.)

[101] P. Lax and B. Wendroff. Difference schemes for hyperbolic equations with high order of accuracy. *Comm. Pure Appl. Math.*, XVII :381–398, 1964. (Cited page 33.)

[102] P.-H. Maire, J. Breil, and S. Galera. A cell-centered arbitrary Lagrangian Eulerian (ALE) method. *Int. J. Numer. Meth. Fluid*, 56 :1161–1166, 2008. (Cited page 42.)

[103] P.-H. Maire. A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry. *J. Comput. Phys.*, 228 :6882–6915, 2009. (Cited page 42.)

[104] P.-H. Maire. A unified sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids. *Int. J. Numer. Meth. Fluid*, 65(11) :1281–1294, 2010. (Cited page 42.)

[105] P.-H. Maire. A high-order one-step sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids. *Computer and Fluids*, 46(1), 2011. (Cited page 42.)

[106] A. Burbeau-Augoula. A Node-Centered Artificial Viscosity Method for Two-Dimensional Lagrangian Hydrodynamics Calculations on a Staggered Grid. *Commun. Comput. Phys.*, 8 :877–900, 2009. (Cited page 42.)

[107] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *J. Comput. Phys.*, 14 :227–253, 1974. Reprinted at *J. Comp. Phys.*, 135 :203–216, 1997. (Cited page 49.)

[108] D. J. Benson. An efficient, accurate, simple ALE method for nonlinear finite element programs. *Comp. Meth. Appl. Mech. Engrg.*, 72 :305–350, 1989. (Cited page 49 et 64.)

[109] L.G. Margolin. Introduction to "An Arbitrary Lagrangian-Eulerian Computing Method for All Flow Speeds". *J. Comput. Phys.*, 135 :198–202, 1997. (Cited page 49.)

[110] D.S. Kershaw, M.K. Prasad, M.J. Shaw, and J.L. Milovich. 3D unstructured mesh ALE hydrodynamics with the upwind discontinuous finite element method. *Comp. Meth. Appl. Mech. Engrg.*, 158 :81–116, 1998. (Cited page 49.)

[111] P. Kjellgren and J.Hyvarien. An arbitrary Lagrangian-Eulerian finite element method. *Computational Mechanics*, 21 :81–90, 1998. (Cited page 49.)

[112] J. S. Peery and D. E. Carroll. Multi-material ALE methods in unstructured grids. *Comp. Meth. Appl. Mech. Engrg.*, 187 :591–619, 2000. (Cited page 49 et 64.)

[113] B. R. Pember R. W. Anderson, N. S. Elliott. An arbitrary lagrangian-eulerian method with adaptive mesh refinement for the solution of the euler equations. *J. Comput. Phys.*, 199(2) :598–617, 2004. (Cited page 49.)

[114] D. E. Burton. Consistent finite-volume discretization of hydrodynamics conservation laws for unstructured grids. *Report UCRL-JC-118788, Lawrence Livermore National Laboratory*, 1994. (Cited page 49 et 69.)

[115] A. M. Winslow. Equipotential zoning of two-dimensional meshes. Technical Report UCRL-7312, Lawrence Livermore National Laboratory, 1963. (Cited page 50 et 64.)

[116] A. Winslow. Numerical solution of the quasilinear Poisson equations in a nonuniform triangle mesh. *J. Comput. Phys.*, 1 :149–172, 1966. (Cited page 50 et 64.)

[117] A. Barlow. ALE and AMR mesh refinement techniques for multi-material hydrodynamics problems. ICFD Workshop on Mesh Refinement Techniques, Oxford, U.K, 2005. (Cited page 50.)

[118] Markus Berndt, Milan Kuchařík, and Mikhail J. Shashkov. Using the feasible set method for rezoning in ALE. *Procedia CS*, 1(1) :1885–1892, 2010. (Cited page 51.)

[119] P. Knupp. Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part I – a framework for surface mesh optimization. *Int. J. Numer. Meth. Eng.*, 48 :401–420, 2000. (Cited page 51.)

[120] P. Knupp and N. Robidoux. A framework for variational grid generation : Conditioning the jacobian matrix with matrix norms. *SIAM J. on Scientific Computing*, 21(6) :2029–2047, 2000. (Cited page 51.)

[121] P. Knupp. Algebraic mesh quality metrics. *SIAM J. Sci. Comput.*, 23(1) :193–218, 2001. (Cited page 51.)

[122] P. Knupp, L. G. Margolin, and M. Shashkov. Reference Jacobian optimization-based rezone strategies for arbitrary Lagrangian Eulerian methods. *J. Comput. Phys.*, 176 :93–128, 2002. (Cited page 51.)

[123] P. Knupp. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elements in Analysis and Design*, 39(3) :217 – 241, 2003. (Cited page 51.)

[124] P. Knupp, L. G. Margolin, and M. Shashkov. Reference Jacobian optimization-based rezone strategies for arbitrary Lagrangian Eulerian methods. *J. Comput. Phys.*, 176 :93–128, 2002. (Cited page 51, 58 et 64.)

[125] Pavel Vachal, Rao V. Garimella, and Mikhail J. Shashkov. Untangling of 2D meshes in ALE simulations. *Journal of Computational Physics*, 196(2) :627 – 644, 2004. (Cited page 51 et 58.)

[126] R. Loubère. On the automatic choice of the rezone strategy in arbitrary-Lagrangian-Eulerian hydrocodes. Technical report, Personal communication, 2006. (Cited page 51.)

[127] L.G. Margolin and M. Shashkov. Second-order sign-preserving conservative interpolation (remapping) on general grids. *J. Comput. Phys.*, 184(1) :266–298, 2003. (Cited page 52 et 71.)

[128] M. Kuchařík, M. Shashkov, and B. Wendroff. An efficient linearity-and-bound-preserving remapping methods. *J. Comput. Phys.*, 188 :462–471, 2003. (Cited page 52, 57, 58 et 71.)

[129] J. Grandy. Conservative remapping and regions overlays by intersecting arbitrary polyhedra. *J. Comput. Phys.*, 148 :433–466, 1999. (Cited page 52 et 71.)

[130] R. Pember and R. Anderson. Comparison of direct Eulerian Godunov and Lagrange plus remap artificial viscosity schemes for compressible flow. Technical Report 2001-2644, AIAA, 2001. (Cited page 52.)

[131] W.B. VanderHeyden and B.A. Kashiwa. Compatible fluxes from van Leer advection. *J. Comput. Phys.*, 146 :1–28, 1998. (Cited page 52.)

[132] J.K. Dukowicz and J.W. Kodis. Accurate conservative remapping (rezoning) for Arbitrary-Lagrangian-Eulerian computations. *SIAM J. Stat. Comput.*, 8 :305–321, 1987. (Cited page 52.)

[133] D. J. Benson. Momentum advection on a staggered mesh. *J. Comput. Phys.*, 100 :143–162, 1992. (Cited page 52.)

[134] J.K. Dukowicz and J. Baumgardner. Incremental remapping as a transport/advection algorithm. *J. Comput. Phys.*, 160 :318–335, 2000. (Cited page 52 et 59.)

[135] P.J. O'Rourke and M.S. Sahota. A variable explicit/implicit numerical method for calculating advection on unstructured meshes. *J. Comput. Phys.*, 143 :312–345, 1998. (Cited page 52.)

[136] G. Luttwak and J. Falcovitz. Slope limiting for vectors : A novel vector limiting algorithm. *Int. J. Numer. Meth. Fluid*, 2010. doi : 10.1002/fld.2367. (Cited page 52.)

[137] Peter H. Lauritzen, Christoph Erath, and Rashmi Mittal. On simplifying incremental remap-based transport schemes. *Journal of Computational Physics*, 230(22) :7957 – 7963, 2011. (Cited page 52.)

[138] P. Hoch. Mesh quality and conservative projection in Lagrangian compressible hydrodynamic. Conference on Numerical methods for multi-material fluid flows ; Czech Technical University in Prague on September 10 - 14, 2007. (Cited page 52 et 64.)

[139] F. L. Adessio, D. E. Carroll, K. K. Dukowicz, J. N. Johnson, B. A. Kashiwa, M. E. Maltrud, and H. M. Ruppel. CAVEAT : a computer code for fluid dynamics problems with large distortion and internal slip. Technical Report LA-10613-MS, Los Alamos National Laboratory, 1986. (Cited page 54.)

[140] David Bailey, Markus Berndt, Milan Kuchařík, and Mikhail Shashkov. Reduced-dissipation remapping of velocity in staggered arbitrary lagrangian-eulerian methods. *Journal of Computational and Applied Mathematics*, 233(12) :3148 – 3156, 2010. Finite Element Methods in Engineering and Science (FEMTEC 2009). (Cited page 54.)

[141] M. Shashkov and B. Wendroff. The repair paradigm and application to conservation laws. *J. Comput. Phys*, 198 :265–277, 2004. (Cited page 57, 58 et 95.)

[142] L.G Margolin and M. Shashkov. Remapping, recovery and repair on staggered grid. *Computer Methods in Applied Mechanics and Engineering*, 193 :4139–4155, 2004. (Cited page 57, 58 et 95.)

[143] Anabela Oliveira and André B. Fortunato. Toward an oscillation-free, mass conservative, Eulerian-Lagrangian transport model. *Journal of Computational Physics*, 183(1) :142 – 164, 2002. (Cited page 58 et 59.)

[144] P. Colella. Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, 87(1) :171 – 200, 1990. (Cited page 59.)

[145] R. Lohner and C. Yang. Improved ALE mesh velocities for moving bodies. *Communications in Numerical Methods in Engineering*, 12 :599–608, 1996. (Cited page 64.)

[146] C. Stoker, C. Gay, F. Bay, and J.-L. Chenot. A velocity approach for the ALE-method applied for 2D and 3D problems. In *Simulation of Material Processing : Theory, Methods and Applications*. Balkema, Rotterdam, 1998. (Cited page 64.)

[147] J. K. Dukowicz, M. C. Cline, and F. S. Addessio. A general topology Godunov method. *J. Comput. Phys.*, 82 :29–63, 1989. (Cited page 64, 66, 69 et 71.)

[148] S. Giuliani. An algorithm for continuous rezoning of the hydrodynamic grid in arbitrary Lagrangian-Eulerian computer codes. *Nuclear Engineering and Design*, 72 :205–212, 1982. (Cited page 64.)

[149] J.U. Brackbill and J.S. Saltzman. Adaptive zoning for singular problems in two dimensions. *J. Comput. Phys.*, 46 :342–368, 1982. (Cited page 64.)

[150] M.J. Baines. *Moving Finite Elements*. Oxford Science Publications, Clarendon Press, Oxford, 1994. (Cited page 64.)

[151] J.K. Dukowicz. A simplified adaptive mesh technique derived from the moving finite element method. *J. Comput. Phys.*, 56 :324–342, 1984. (Cited page 64.)

[152] K. Lipnikov and M. Shashkov. The error-minnimization-based strategy for moving mesh methods. *Commun. in Comput. Phys.*, 1(1) :53–81, 2006. (Cited page 64.)

[153] A.J. Barlow. ALE and AMR mesh refinement techniques for multi-material hydrodynamics problems. ICFD Workshop on Mesh Refinement Techniques 7th December 2005, `http://www.icfd.rdg.ac.uk/Workshops/AMR/meshreftechf.pdf`. (Cited page 64.)

[154] A.J. Barlow. Challenges and recent progress in developing numerical methods for multi-material ALE hydrocodes. ICFD 25 year Anniversary Conference 15-16th September 2008 Oxford University, `http://www.icfd.rdg.ac.uk/ICFD25/Talks/ABarlow.pdf`. (Cited page 64.)

[155] J.R. Pasta and S. Ulam. Heuristic numerical work in some problems of hydrodynamics. *Math. Tables and other Aids to Computation*, XIII(65) :1–12, 1959. (Cited page 65.)

[156] M.J. Fritts, W.P. Crowley, and H. Trease, editors. *The Free-Lagrange Method : Proceedings of the First International Conference of Free-Lagrange Methods*. Springer-Verlag, New York, 1985. Lecture Notes in Physics, Vol. 238. (Cited page 65.)

[157] J.U. Brackbill and J.J. Monaghan, editors. *Proceedings of the Workshop on Particle methods in fluid dynamics and plasma physics*. Elsevier, 1987. *Comp. Phys. Commun.*, 48(1), 1987. (Cited page 65.)

[158] H. Trease, M.J. Fritts, and W.P. Crowley, editors. *Advances in the Free-Lagrange Method : including contributions on adaptive gridding and the smooth particle hydrodynamics method : Proceedings of the Next Free-Lagrange Conference*. Springer-Verlag, New York, 1990. Lecture Notes in Physics, Vol. 395. (Cited page 65.)

[159] W.P. Crowley. FLAG : A Free-Lagrange method for numerically simulating hydrodynamic flows in two dimensions. In M. Holt, editor, *Proceddings of the Second International Conference on Numerical Methods in Fluid Dynamics, September 15-19, 1970, University of California, Berkeley*, volume 8, pages 37–43, Berlin, 1971. Lecture Notes in Physics, Springer. (Cited page 65.)

[160] Stéphane Del Pino. Metric-based mesh adaptation for 2D Lagrangian compressible flows. *J. Comput. Phys.*, 230(5) :1793–1821, March 2011. (Cited page 65.)

[161] Morrell J. M., Sweby P. K., and A. Barlow. A cell by cell anisotropic adaptive mesh ALE scheme for the numerical solution of the euler equations. *J. Comput. Phys.*, 226 :1152–1180, 2007. (Cited page 65.)

[162] P. Hoch. Semi-conformal polygonal mesh adaptation seen as grid velocity formulation for ALE simulations. Conference on Numerical methods for multi-material fluid flows MULTIMAT'09 ; Pavia, Italy 2009, available at `www.math.univ-toulouse.fr/HYDRO`. (Cited page 65.)

[163] P. Hoch. A two-dimensional discontinuous ALE(DISCALE) finite volume framework on arbitrary unstructured conical meshes. Conference on Numerical methods for multi-material fluid flows MULTIMAT'11 ; Arcachon, France 2011, available at `www.math.univ-toulouse.fr/HYDRO` or `http://multimat2011.celia.u-bordeaux1.fr/Multimat2011/Tuesday_AM/Hoch.pdf`. (Cited page 65.)

[164] S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2) :129–137, 1982. (Cited page 67.)

[165] S.S. Sokolov, A.I. Panov, A.A. Voropinov, I.G. Novikov, I.V. Sobolev, and A.V. Yalozo. Method TIM for computation of three-dimensional problems of mechanics of continuum media using unstructured polyhedral Lagrangian meshes (In Russian). *Questions of Atomic Science and Techniques, Series : Mathematical modeling of physical processes*, 3 :37–52, 2005. (Cited page 69.)

[166] S.S. Sokolov, A.A. Voropinov, I.G. Novikov, A.I. Panov, I.V. Sobolev, and A.A. Pushkarev. Method TIM-2D for computation of problems of mechanics of continuum media using unstructured polygonal with arbitrary connections at the nodes(In Russian). *Questions of Atomic Science and Techniques, Series : Mathematical modeling of physical processes*, 4 :29–44, 2006. (Cited page 69.)

[167] Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi tesselations : applications and algorithms. *SIAM Review*, 41 :637–676, 1999. (Cited page 69 et 70.)

[168] Q. Du, M. Emelianenko, and M. Gunzburger. Convergence of the Lloyd algorithm for computing centroidal Voronoi tesselations. *SAIM J. Numer. Anal.*, 44(1) :102–119, 2006. (Cited page 70.)

[169] L. I. Sedov. *Similarity and Dimensional Methods in Mechanics*. Academic Press, New York, 1959. (Cited page 71 et 80.)

[170] M. Kuchařík and M. Shashkov. Extension of efficient, swept-integration-based conservative remapping method for meshes with changing connectivity. *International Journal for Numerical Methods in Fluids*, 56(8) :1359–1365, 2007. (Cited page 71.)

[171] M. Kuchařík, J. Breil, S. Galera, P.-H. Maire, M. Berndt, and M. Shashkov. Hybrid remap for multi-material ALE. *Computers and Fluids*, 46(1) :293 – 297, 2011. 10th ICFD Conference Series on Numerical Methods for Fluid Dynamics (ICFD 2010). (Cited page 71.)

[172] Milan Kuchařík and Mikhail Shashkov. One-step hybrid remapping algorithm for multi-material arbitrary Lagrangian-Eulerian methods. *Journal of Computational Physics*, 231(7) :2851 – 2864, 2012. (Cited page 71.)

[173] Markus Berndt, Jérôme Breil, Stéphane Galera, Milan Kuchařík, Pierre-Henri Maire, and Mikhail Shashkov. Two-step hybrid conservative remapping for multimaterial arbitrary Lagrangian-Eulerian methods. *Journal of Computational Physics*, 230(17) :6664 – 6687, 2011. (Cited page 71.)

[174] J.-F. Haas and B. Sturtevant. Interaction of weak-shock waves. *J. Fluid Mech.*, 181 :41–76, 1987. (Cited page 78.)

[175] P. D. Lax and X. D. Liu. Solution of two dimensional Riemann problem of gas dynamics by positive schemes. *SIAM J. Sci. Comput.*, 19(2) :319–340, 1998. (Cited page 80.)

[176] X. D. Liu and P. D. Lax. Positive schemes for solving multi-dimensional hyperbolic systems of conservation laws. *CFD Journal*, 5(1) :1–24, 1996. (Cited page 80.)

[177] S. Galera, P.-H. Maire, and J. Breil. A two-dimensional unstructured cell-centered multi-material ALE scheme using VOF interface reconstruction. *J. Comput. Phys.*, 229 :5755–5787, 2010. (Cited page 82.)

[178] C.W. Hirt and B.D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.*, 39 :201–225, 1981. (Cited page 82.)

[179] H.Stewart and B. Wendroff. Two-phase flow : models and methods. *J. Comput. Phys.*, 56 :363–409, 1984. (Cited page 82.)

[180] R.E. Tipton. CALE mixed zone pressure relaxation model. Private communication, 1989. (Cited page 82.)

[181] A. Murrone and H. Guillard. A five equation reduced model for compressible two phase flow problems. *Journal of Computational Physics*, 202 :664–698, 2005. (Cited page 82.)

[182] Yu. Yanilkin Yu. Bondarenko. Computation of the thermodynamic parameters in the mixed cells in gas dynamics. *Mathematical Modeling*, 14 :63–81, 2002. (Cited page 82.)

[183] B. Despres and F. Lagoutière. Numerical resolution of a two-component compressible fluid model with interfaces. *Progress in Computational Fluid Dynamics*, 7(6) :295–310, 2007. (Cited page 82.)

[184] A. Barlow. A new Lagrangian scheme for multimaterial cells. In *Proceedings of European Congress on Computational Methods in Applied Sciences and Engineering. ECCOMAS Computational Fluid Dynamics Conference*, 4-7 September 2001. Swansea, Wales, U.K. (Cited page 82.)

[185] Sadchikov VV. Delov VI. Comparison of several models for computation of thermodynamical parameters for heterogeneous Lagrangian cells. *VANT, Mathematical Modeling of Physical Processes*, 1 :57–70, 2005. (Cited page 82.)

[186] Yanilkin YV. Goncharov EA. New method for computations of thermodynamical states of the materials in the mixed cells. *VANT, Mathematical Modeling of Physical Processes*, 3 :16–30, 2004. (Cited page 82.)

[187] M. Shashkov. Closure models for multimaterial cells in arbitrary LagrangianâĂŞEulerian hydrocodes. *International Journal for Numerical Methods in Fluids*, 56(8) :1497–1504, 2008. Special Issue : Institute for Computational Fluid Dynamics. (Cited page 82.)

[188] James R. Kamm and Mikhail J. Shashkov. A pressure relaxation closure model for one-dimensional, two-material Lagrangian hydrodynamics based on the riemann problem. *Communications in Computational Physics*, 7 :927–976, 2010. (Cited page 82.)

[189] D.L. Youngs. Time dependent multi-material flow with large fluid distortion. In M.J. Baines (Eds.) K.W. Morton, editor, *Numerical Methods for Fluid Dynamics*, pages 274–285. Academic Press, New York,, 1982. (Cited page 82, 88 et 106.)

[190] D.L. Youngs. An interface tracking method for a 3d Eulerian hydrodynamics code. Technical Report Technical Report AWE/44/92/35, Atomic Weapon Research Establishment, Aldermaston, Berkshire, UK, 1984. (Cited page 82 et 88.)

[191] W.J. Rider and D.B. Kothe. Reconstructing volume tracking. *J. Comput. Phys.*, 121 :112–152, 1998. (Cited page 82 et 87.)

[192] Woodward P Noh WF. SLIC (simple line interface calculation). In *5th International Conference on Numerical Methods in Fluid Dynamics*, 1976. In van der Vooren AI and Zandbergen PJ. (Cited page 82.)

[193] Benson DJ. Volume of fluid interface reconstruction methods for multi-material problems. *Applied Mechanics Review*, 55(2) :151–165, 2002. (Cited page 82.)

[194] Scardovelli R. and Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31 :567–603, 1999. (Cited page 82.)

[195] Vadim Dyadechko and Mikhail Shashkov. Reconstruction of multi-material interfaces from moment data. *Journal of Computational Physics*, 227(11) :5361 – 5384, 2008. (Cited page 83 et 110.)

[196] H.T. Ahn and M. Shashkov. Multi-material interface reconstruction on generalized polyhedral meshes. *J. Comput. Phys.*, 226 :2096–2132, 2007. (Cited page 83.)

[197] M. Kuchařík, R. Garimella, S. Schofield, and M. Shashkov. A comparative study of interface reconstruction methods for multi-material ALE simulations. *J. Comput. Phys.*, 2009. (Cited page 83.)

[198] M.P. Allen and D.J. Tildesley. *Computer simulation of liquids*. Oxford University Press, New York,, 1987. (Cited page 84.)

[199] J.M. Haile. *Molecular dynamics simulation : elementary methods*. Wiley-Interscience, New-York, 1992. (Cited page 84.)

[200] W. Benz. *Smooth particle hydrodynamics : a review*, volume In *The Numerical modelling of nonlinear stellar pulsations : problems and prospect*, NATO ASI series. Series C, Mathematical and physical sciences of *Series C, Mathematical and physical sciences*. Kluwer Academic Publishers, 1989. (Cited page 84.)

[201] R.A. Gingold and J.J Monaghan. Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Monthly Notices Royal Astronomical Society*, 181 :375–389, 1977. (Cited page 84.)

[202] S.P. Schofield, R.V. Garimella, M.M. Francois, and R. Loubère. Multi-material interface reconstruction using particles and power diagrams. Technical Report LAUR-06-8740, Los Alamos National Laboratory Report, 2006. (Cited page 85.)

[203] A. Harten. ENO scheme with subcell resolution. *J. Comput. Phys.*, 83 :148, 1989. (Cited page 86.)

[204] B.K. Swartz. The second-order sharpening of blurred smooth borders. *athematics of Computation*, 52(186) :675–, 1989. (Cited page 86.)

[205] T.J. Barth. Aspects of unstructured grids and finite-volume solvers for Euler and Navier-Stokes equations. *VKI/NASA/AGARD Special Course on Unstructured Grid Methods for Advection Dominated Flows AGARD Publication R-787*, 1995. (Cited page 87.)

[206] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In D. Kroner, M. Ohlberger, and C. Rohde, editors, *An introduction to Recent Developments in Theory and Numerics for Conservation Laws, Proceedings of the International School on Theory and Numerics for Conservation Laws*, pages 195–284, Berlin, 1997. Lecture Notes in Computational Science and Engineering, Springer. (Cited page 87.)

[207] S. F. Bockman. Generalizing the formula for areas of polygons to moments. *The American Mathematical Monthly*, 92 :131–, 1989. (Cited page 87.)

[208] F. Aurenhammer. Power diagrams : properties, algorithms and applications. *SIAM Journal of Computing*, 16 :78–96, 1987. (Cited page 88.)

[209] F. Aurenhammer. Voronoi diagrams - a survey of fundamental geometric data structures. *ACM Computing Surveys*, 23(3) :345–405, 1991. (Cited page 88.)

[210] Jean-Michel Ghidaglia, Anela Kumbaro, and Gérard Le Coq. On the numerical solution to two fluid models via a cell centered finite volume method. *European Journal of Mechanics - B/Fluids*, 20(6) :841 – 867, 2001. (Cited page 94 et 106.)

[211] J.-P. Braeunig. *Sur la simulation d'Ecoulements multimatériaux par une méthode Eulérienne directe avec capture d'interface en dimensions 1, 2 et 3*. PhD thesis, Ecole Normale Supérieure de Cachan, 2007. (Cited page 94 et 106.)

[212] J.-P. Braeunig, B. Desjardins, and J.-M. Ghidaglia. A totally Eulerian finite volume solver for multi-material fluid flows. *European Journal of Mechanics - B/Fluids*, 28(4) :475 – 485, 2009. (Cited page 94 et 106.)

[213] B. van Leer. Towards the ultimate conservative difference scheme. V - A second-order sequel to Godunov's method. *J. Comput. Phys.*, 32 :101–136, July 1979. (Cited page 95.)

[214] A. Harten and S. Osher. Uniformly High-Order Accurate Nonoscillatory Schemes. I. *SIAM Journal on Numerical Analysis*, 24 :279–309, April 1987. (Cited page 95.)

[215] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes,. *J. Comput. Phys.*, 71(2) :231–303, August 1987. (Cited page 95.)

[216] C.W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *J. Comput. Phys.*, 83(1) :32–78, July 1989. (Cited page 95.)

[217] B. van Leer. Towards the Ultimate Conservation Difference Scheme. II. Monotonicity and Conservation Combined in a Second-Order Scheme. *J. Comput. Phys.*, 14 :361, March 1974. (Cited page 95.)

[218] B. van Leer. Towards the ultimate conservative difference scheme. III - Upstream-centered finite-difference schemes for ideal compressible flow. IV - A new approach to numerical convection. *J. Comput. Phys.*, 23 :263–299, March 1977. (Cited page 95.)

[219] G. D. van Albada, B. van Leer, and W. W. Roberts, Jr. A comparative study of computational methods in cosmic gas dynamics. *Astron. Astrophysics*, 108 :76–84, April 1982. (Cited page 95.)

[220] P. K. Sweby. High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. *SIAM Journal on Numerical Analysis*, 21 :995–1011, October 1984. (Cited page 95.)

[221] Timothy J Barth and Dennis C Jespersen. The design and application of upwind schemes on unstructured meshes. *AIAA Paper*, 89(89-0366) :1–12, 1989. (Cited page 95.)

[222] V. Venkatakrishnan. Convergence to steady state solutions of the euler equations on unstructured grids with limiters. *Journal of Computational Physics*, 118(1) :120 – 130, 1995. (Cited page 95.)

[223] D. Kuzmin. On the design of general-purpose flux limiters for finite element schemes. I. Scalar convection. *J. Comput. Phys.*, 219 :513–531, December 2006. (Cited page 95.)

[224] P. L. Roe. Characteristic-based schemes for the Euler equations. *Annual Review of Fluid Mechanics*, 18 :337–365, 1986. (Cited page 95.)

[225] G.S. Jiang and C.W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys*, 126 :202–228, 1995. (Cited page 95.)

[226] P. Hoch. An arbitrary lagrangian-eulerian strategy to solve compressible fluid flows. HAL, `http://hal.archives-ouvertes.fr/hal-00366858`, 2009. private communication. (Cited page 95.)

[227] G. Carre, S. Del Pino, K. Pichon Gostaf, E. Labourasse, and A. Shapeev. Polynomial Least-Squares reconstruction for semi-Lagrangian cell-centered hydrodynamic schemes. *ESAIM Proceedings*, 28 :100–116, 2009. (Cited page 96.)

[228] Carlo Cercignani. *The Boltzmann equation and its applications.* Applied Mathematical Sciences, Vol. 67. New York etc. : Springer-Verlag. XII, 455 p. ; DM 98.00 , 1988. (Cited page 99.)

[229] G. A. Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Clarendon, Oxford, 1994. (Cited page 99.)

[230] Russel E. Caflisch. Monte Carlo and quasi-Monte Carlo methods. *Acta Numerica*, 7(-1) :1–49, 1998. (Cited page 99.)

[231] L. Pareschi R. E. Caflisch. Towards an hybrid method for rarefied gas dynamics. *IMA Vol. App. Math.*, 135 :57–73, 2004. (Cited page 99.)

[232] Kenichi Nanbu. Direct simulation scheme derived from the Boltzmann equation. I. monocomponent gases. *Journal of the Physical Society of Japan*, 49(5) :2042–2049, 1980. (Cited page 99.)

[233] G. Russo F. Filbet. High order numerical methods for the space non-homogeneous Boltzmann equation. *J. Comput. Phys.*, 186 :457–480, 2003. (Cited page 99.)

[234] P. Bertrand F. Filbet, E. Sonnendrücker. Conservative numerical schemes for the vlasov equation. *J. Comput. Phys.*, 172 :166–187, 2001. (Cited page 99 et 100.)

[235] L. Mieussens. Discrete velocity model and implicit scheme for the BGK equation of rarefied gas dynamic. *Math. Models Meth. App. Sci.*, 10 :1121–1149, 2000. (Cited page 99 et 100.)

[236] Andrzej Palczewski, Jacques Schneider, and Alexandre V. Bobylev. A consistency result for a discrete-velocity model of the Boltzmann equation. *SIAM J. Numer. Anal.*, 34(5) :1865–1883, October 1997. (Cited page 99.)

[237] G. Puppo S. Pieraccini. Implicit-explicit schemes for BGK kinetic equations. *SIAM J. Sci. Comp.*, 32 :1–28, 2007. (Cited page 99.)

[238] Nicolas Crouseilles, Thomas Respaud, and Eric Sonnendrücker. A Forward semi-Lagrangian Method for the Numerical Solution of the Vlasov Equation. *Computer Physics Communications*, 180(10), 2009. (Cited page 99 et 100.)

[239] Nicolas Crouseilles, Michel Mehrenberger, and Eric Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *J. Comput. Phys.*, pages 1927–1953, 2010. (Cited page 99 et 100.)

[240] J. Schneider A.V. Bobylev, A. Palczewski. On approximation of the Boltzmann equation by discrete velocity models. *C. R. Acad. Sci. Paris Ser. I. Math.*, 320 :639–644, 1995. (Cited page 99.)

[241] J. Schneider A. Palczewski. Existence, stability, and convergence of solutions of discrete velocity models to the Boltzmann equation. *J. Statist. Phys.*, 91 :307–326, 1998. (Cited page 99.)

[242] S. Mischler L. Desvillettes. About the splitting algorithm for Boltzmann and BGK equations. *Math. Mod. & Meth. in App. Sci.*, 6 :1079–1101, 1996. (Cited page 99.)

[243] G. Strang. On the construction and the comparison of difference schemes. *SIAM J. Numer. Anal.*, pages 506–517, 1968. (Cited page 99 et 100.)

[244] Jun S. Liu. *Monte Carlo strategies in scientific computing*. Springer, 2001. (Cited page 100.)

Ce document a été préparé à l'aide de l'éditeur de texte GNU Emacs et du logiciel de composition typographique LaTeX 2_ε. Les figures ont été réalisées avec le logiciel Xfig, les résultats numériques ont été majoritairement réalisés à l'aide de Gnuplot et Paraview.

Titre Contribution au domaine des méthodes numériques Lagrangiennes et Arbitrary-Lagrangian-Eulerian

Résumé Ce mémoire présente des travaux portant (i) sur les méthodes numériques lagrangiennes et (ii) sur le développement des méthodes dites arbitrairement lagrangienne-eulérienne (ALE). Ces deux thématiques ont en commun de tenter de résoudre les équations de la mécanique des fluides compressibles en multi-dimensions sur des maillages mobiles se déplaçant soit à la vitesse du fluide (lagrangienne), soit à une vitesse arbitraire (ALE). En particulier nous abordons les problèmes de viscosité artificielle, de consistance et précision, de stabilité, de consistance en volume, le traitement des points exceptionnels ou encore les lignes de glissement. Dans le chapitre ALE nous proposons des études sur les phase de projection conservative, correction a posteriori, reconnexion topologique de maillage ou de reconstruction d'interface dans des mailles mixtes. (iii) Une troisième partie propose un ensemble de sujets plus hétéroclites : reconstruction d'interface dans des schémas multi-matériaux sur maillage fixe, schémas cinétiques ultra rapides, et des schémas de type volumes finis d'ordre très élevé.

Mots-clés Schéma lagrangien, ALE, projection, remaillage, reconnection de maillages, tesselation de Voronoi, méthode MOOD, schéma cinétique, 3D, reconstruction d'interfaces

Title Contribution to Lagrangian and Arbitrary-Lagrangian-Eulerian numerical schemes

Abstract This thesis presents our work related to (i) Lagrangian schemes and (ii) Arbitrary-Lagrangian-Eulerian numerical methods (ALE). Both types of methods have in commun to solve the multidimension compressible Euler equations on a moving grid. The grid moves with either the fluid velocity (Lagrangian) or an arbitrary velocity (ALE). More specifically we deal with some problems related to artifical viscosity, internal consistency, stability, accuracy, exceptional points and slide line treatments. In the ALE chapter we study the remap and rezone phases but also the mesh reconnection to build a ReconnectionALE scheme and further some interface reconstruction techniques. In a third chapter (iii) other resarch topics are presented like interface reconstruction techniques on a fixed grid finite volume scheme, ultra fast kinetic scheme, and very high-order finite volume schemes.

Keywords Lagrangian scheme, Arbitrary-Lagrangian-Eulerian scheme, remap, rezone, reconnect, repair, Voronoi tesselation, MOOD method, kinetic scheme, 3D, interface reconstruction.