



Swansea University
Prifysgol Abertawe



A Novel Arbitrary Lagrangian Eulerian Framework for Large Strain Solid Dynamics

Thomas B. J. Di Giusto

Submitted to the Faculty of Science and Engineering of Swansea University and to the Barcelona School of Civil Engineering of Universitat Politècnica de Catalunya.
in partial fulfilment of the requirements for the Degree of

Doctor of Philosophy

2024

Declarations

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed...

Date.....

30 September 2023

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed....

Date.....

30 September 2023

I hereby give consent for my thesis, if accepted, to be available for electronic sharing after expiry of a bar on access approved by the Swansea University.

Signed....

Date.....

30 September 2023

The University's ethical procedures have been followed and, where appropriate, that ethical approval has been granted.

Signed.

Date.....

30 September 2023

ACKNOWLEDGEMENTS

I would like to express my profound gratitude to Professor Antonio J. Gil from the Zienkiewicz Institute, and Doctor Chun Hean Lee from the Glasgow Computational Engineering Centre for their outstanding supervision. By the science they taught me, by the never-ending support and abnegation they demonstrated to my behalf, and by the opportunity they gave me to live this adventure, I can say without a doubt that I am truly honoured to have worked with these two. My deepest thanks then go to Professor Javier Bonet, who timely contributed to the development of the ALE formulation. Furthermore, I would like to express my most sincere recognition to Professor Antonio Huerta and Doctor Matteo Giacomini for their support during my secondment to UPC BarcelonaTech, and for the guidance they provided throughout the past years. My thorough and full dedication to this work was made possible thanks to the financial support from the project Marie Skłodowska-Curie ITN-EJD ProTechTion, funded by the European Union Horizon 2020 research and innovation program with grant number 764636.

That period of time has also been for me a point of concurrency with other researchers, comrades, and friends that I will not forget. It is the right moment to show my appreciation to Paulo for being an awesome colleague, as well as Loeiz, Stefano, Bernard, Osama and Jibran for being great people in my entourage. In addition to this, I think it is more than fair to acknowledge the help I received from the different online communities specialised in OpenFOAM, LaTeX, Physics, Mathematics and Engineering. More generally, I thank the scientific world for the numerous resources that are shared. Besides, I praise the friendship with my friend Mickaël as he gave me the most peculiar support I could think of.

Last, but definitely not least, I dedicate this work to the women of my life. My grandmother Jeanine, who has gone to rest within that time, has always blessed me with her unconditional love and support. My mother Nathalie, who lavished upon me precious opportunities and an excellent education that makes me who I am. My aunt Silvia, whose presence and words are an important moral backing. My girlfriend Lucía, who has helped me see the light in many ways.

Thank you all for your support.

In the realm of Computer-Aided Engineering applied to fast solid dynamics, the intricate mechanical behaviours exhibited by materials when subjected to strong dynamic forces, high speed impacts and complex interactions are modelled efficiently and with high fidelity. Employed in diverse fields such as aerospace, automotive, defence and more, the principal interest is to simulate and comprehend the responses of solids, providing insights into stress propagation and deformation patterns. However, the pursuit of such ambitious goals faces inherent limitations: the accurate representation of material behaviours is an ongoing challenge, and the intricate interplay between simulation accuracy and computational efficiency demands thoughtful insights. More specifically, the chosen kinematics paradigm and the discretisation of the continuum often restrict numerical frameworks in the array of problems they can simulate. Simulations in fast solid dynamics may feature locking, numerical instabilities, checker-boarding, or other difficulties related to the nonlinear nature of the equations of state.

In the objective to address the aforementioned shortcomings, this thesis will build on the set of equations introduced in [1, 2] by developing a new mixed formulation based on first-order hyperbolic equations and written with the Arbitrary Lagrangian-Eulerian viewpoint. That approach, used here to describe solid bodies and studied by [3–6], aims at circumventing bottlenecks of Lagrangian and Eulerian methods by distinguishing the behaviour of the mesh from the evolution of the continuum. The ALE formulation introduces a referential (fixed) domain separate from the spatial and material domains and used for motion description. The computational mesh partially follows the material points to reduce element distortion. A key aspect of this work is to adapt the mesh via solving dedicated conservation laws incorporated in a general mixed formulation, removing the need of an *ad hoc* procedure. The ALE methodology shows promise in addressing challenges in large strain solid dynamics, including hyper-velocity dynamic impact/contact and crack propagation. An acoustic Riemann solver based on upwinding stabilisation, as well as a linear gradient reconstruction, will be used to counteract instabilities brought by the Vertex-Centred Finite Volume Method employed in the framework, and to enhance the overall accuracy. The nonlinear hardening laws will be solved using a Newton-Raphson algorithm. The new framework introduced in this work will be implemented from scratch on the open-source platform OpenFOAM, a tool of choice in industrial and academic environments. The time integration will be tackled by the multi-stage Total Variation Diminishing Runge-Kutta method. Eventually, the robustness and accuracy of the novel computational framework will be examined through a series of challenging numerical examples involving complex body deformations, as well as plastic and thermal considerations.

- **Journals**

T. B. J. Di Giusto, C. H. Lee, A. J. Gil, J. Bonet, M. Giacomini. *A first-order hyperbolic Arbitrary Lagrangian Eulerian conservation formulation for non-linear solid dynamics*. In Press: *International Journal for Numerical Methods in Engineering (IJNME)* <https://doi.org/10.1002/nme.7467>.

T. B. J. Di Giusto, C. H. Lee, A. J. Gil, J. Bonet, C. Wood, M. Giacomini. *A first-order hyperbolic Arbitrary Lagrangian Eulerian conservation formulation for nonlinear solid dynamics in irreversible processes*. Under preparation: *Journal of Computational Physics (JCP)*.

- **Conferences and Congresses**

T. B. J. Di Giusto, A. J. Gil, C. H. Lee, A. Huerta, M. Giacomini, J. Bonet *A novel Arbitrary Lagrangian formulation for a system of first order conservation laws in fast solid dynamics*. UK Association for Computational Mechanics (UKACM) Research Highlight: United Kingdom, 2020.

T. B. J. Di Giusto, A. J. Gil, C. H. Lee, A. Huerta, M. Giacomini *Coupling materials for fluid-structure interactions in OpenFOAM*. International Conference on Adaptive Modelling and Simulation (ADMOS) 2021: Gothenburg, Sweden, 2021.

T. B. J. Di Giusto, A. J. Gil, C. H. Lee, A. Huerta, M. Giacomini *A novel Arbitrary Lagrangian Eulerian formulation for a system of first order conservation laws in fast solid dynamics*. UK Association for Computational Mechanics (UKACM) 2021: Loughborough, United Kingdom, 2021.

- **Workshops**

T. B. J. Di Giusto, A. J. Gil, C. H. Lee, A. Huerta, M. Giacomini, J. Bonet *A new computational tool for non-linear multi-material Continuum Mechanics*. Zienkiewicz Centre for Computational Engineering 2020: Swansea, United Kingdom, 2020. **Best poster award**.

T. B. J. Di Giusto, A. J. Gil, C. H. Lee, A. Huerta, M. Giacomini, J. Bonet *A novel Arbitrary Lagrangian formulation for a system of first order conservation laws in fast solid dynamics*. Zienkiewicz Centre for Computational Engineering 2021: Swansea, United Kingdom, 2021.

Acknowledgements	iv
Abstract	v
Research Output	vii
List of Figures	xxiii
List of Tables	xxvii
List of Algorithms	xxix
Listings	xxxix
Abbreviations	xxxiv
Nomenclature	xxxv
List of Symbols	xxxv
1 Introduction	1
1.1 Motivation	2
1.2 State of the art in solid dynamics	2
1.2.1 Tetrahedral meshes	3
1.2.2 Finite Volume Methods	4
1.2.3 Mixed methodologies	5
1.2.4 Arbitrary Lagrangian Eulerian formulations	5
1.3 Scope and outline of the thesis	8
2 Arbitrary Lagrangian Eulerian Formulation for Isothermal Hyperelasticity	10
2.1 Introduction	11
2.1.1 Notation and useful properties	11
2.2 Continuum description and kinematics	11
2.3 ALE transformations	15
2.3.1 Total Lagrangian Scalar Conservation law	15
2.3.2 Total Lagrangian Tensorial Conservation law	16
2.3.3 Arbitrary Lagrangian Eulerian Scalar Conservation law	16
2.3.4 Arbitrary Lagrangian Eulerian Tensorial Conservation law	17
2.3.5 Non-conservative ALE transformations	18

2.4	ALE conservation laws	18
2.4.1	Material Geometric Conservation Law	19
2.4.2	Conservation of Linear Momentum	20
2.4.3	Geometric conservation laws	20
2.4.4	Conservation of other geometric mappings	21
2.5	ALE Mixed Formulation	22
2.6	Conclusion	23
3	Isothermal Inelasticity	24
3.1	Introduction	25
3.2	Polyconvex elastic models	25
3.3	Nearly incompressible constitutive models	26
3.3.1	Isochoric Mooney Rivlin model	27
3.3.2	Nearly incompressible Hencky model	28
3.4	Hyperbolicity	28
3.4.1	Hyperbolicity of the ALE system	29
3.5	Multiplicative plasticity	31
3.5.1	Work conjugacy in traditional work-hardening approach	32
3.5.2	ALE framework for plasticity	34
3.5.3	Newton-Raphson iterative procedure	35
3.6	Conclusion	36
4	Extension to thermal inelasticity	37
4.1	Introduction	38
4.2	First Law of thermodynamics	38
4.3	Thermo-elasticity	39
4.3.1	The Ballistic energy	40
4.3.2	Second Law of thermodynamics	41
4.3.3	Conservation of internal energy and entropy	42
4.4	Nearly-incompressible thermo-mechanical constitutive models	42
4.4.1	Mie-Grüneisen Equation of state	43
4.5	Thermal visco-plasticity	44
4.5.1	Extension to thermo-plasticity	44
4.6	Conclusion	46
5	Material motion	47
5.1	Introduction	48
5.1.1	Geometrical requirements	48
5.2	Material Balance law	48
5.2.1	Neo-Hookean mesh model	50
5.3	Conclusion	50
5.3.1	Different ALE frameworks	51
6	Numerical discretisation	54
6.1	Introduction	55
6.2	Dual mesh	55
6.3	Finite Volume Method	56
6.3.1	Godunov-type approach	57
6.4	Godunov upwinding stabilisation	58
6.4.1	Contact scenario	58
6.5	Linear reconstruction	60
6.6	Discretisation of the mixed formulation	61
6.6.1	Boundary conditions	63

6.7	Time-marching scheme	65
6.7.1	Implicit source terms	66
6.8	Conclusion	66
7	OpenFOAM implementation	67
7.1	Introduction	68
7.1.1	OpenFOAM versions	69
7.2	Simulation workflow	69
7.2.1	Code structure	69
7.2.2	Setup of the numerical simulation	72
7.2.3	Algorithm workflow	74
7.3	Conclusion	75
8	Numerical simulations: Isothermal Cases	76
8.1	Introduction	77
8.1.1	Analytical prescription of ALE motion	77
8.2	Satisfaction of geometric conservation laws: Patch test	78
8.3	Mesh convergence: Low dispersion swinging cube	83
8.4	L-shaped block	87
8.5	Nearly incompressible bending column	93
8.6	Frictionless impact of a copper bar on a rigid wall.	97
8.7	Necking of a circular copper bar	108
9	Numerical simulations: Thermoelasticity and thermoplasticity	115
9.1	Introduction	116
9.2	Translation column	116
9.3	Swinging cube with heat	122
9.4	L-Shaped block (heat)	127
9.4.1	Discontinuous temperature distribution	129
9.4.2	Linear temperature distribution	134
9.5	Taylor impact with thermal coupling	139
9.5.1	Case 1	139
9.5.2	Case 2	148
9.6	Necking with thermal effects	154
10	Concluding remarks	162
10.1	General conclusions	163
10.2	Future works	165
A	Mathematical foundations	167
A.1	Introduction	168
A.2	Useful Linear Algebra	168
A.2.1	Useful Algebraic properties	168
A.3	Differential calculus	169
A.3.1	Differential operators	169
A.4	Transformation of equations in the continuum	170
A.4.1	On Green-Ostrogradsky theorem	170
A.4.2	On Reynold's Transport theorem	171
A.4.3	On Euler's theorem	172
B	Fundamentals of solid mechanics	173
B.1	Introduction	174
B.2	Prerequisites on classical nonlinear solid mechanics	174

B.2.1	Kinematics	174
B.2.2	Stress and equilibrium	175
B.3	Large Strain thermo-elasticity	176
B.3.1	Calorimetry relationships	176
B.3.2	Volumetric energies	177
B.3.3	On the Lie time derivative	178
B.4	Tools for the Arbitrary Lagrangian Eulerian approach	178
B.4.1	Wave speeds connections	178
B.5	On the degeneration to Lagrangian and Eulerian formulations	179
C	Second law of Thermodynamics	180
D	OpenFOAM components	185
	References	195

LIST OF FIGURES

1.1	Applications of high-strain CSM: (a) Aerospace, (b) Defence, (c) Automotive industry, and (d) Metallurgy.	3
2.1	ALE kinematics description.	12
6.1	Control volumes for (a) an interior node and (b) a boundary node for a median dual tessellation visualised in 2D.	55
6.2	Area vector and its local contribution, for an edge ab_2 of a three dimensional tetrahedral mesh.	56
7.1	Solver structure: files related to the main solver; files related to the compilation are omitted.	70
7.2	Solver structure: library sources; files related to the compilation are omitted.	72
7.3	Workflow of an OpenFOAM simulation. The pre-processing steps are represented in blue, the post-processing steps are represented in red and the solving step is represented in grey.	74
7.4	Case directory structure.	74
8.1	Visualisation of the analytical material motion (8.1) on a unit cube with a magnitude $\beta = 10^{-2}$ m. A range of 8 nodes is highlighted in red so their displacements can be followed.	78
8.2	Translation: Problem setup.	78
8.3	Translation:(a) Structured (S1) and (b) unstructured (U1) meshes. An overview of the mesh is presented on the left and a crinkle clip of half of the column is presented on the right.	79
8.4	Translation: Time evolution of the L^2 error in Jacobian and velocity components when using a Total Lagrangian Formulation and solving for $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ on a structured (blue) and on an unstructured (green) mesh. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using a discretisation of $4 \times 24 \times 4$ elements across on a tetrahedral mesh. A neo-Hookean model is used with parameters listed in Table 8.2.	80
8.5	Translation: Time evolution of the L^2 -error in Jacobian $\ J - 1\ _{L^2}$ when solving for $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ (blue), $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi\}$ (green) and $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Phi\}$ (red). Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	81

8.6	Translation: Time evolution of the L^2 -error in velocity components $\ v_i - v_{0,i}\ _{L^2}$ when solving for $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ (blue), $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi\}$ (green) and $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Phi\}$ (red). Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	81
8.7	Translation: Snapshots of the deformation at time $t = 0, 0.8, 1.6, 2.4, 3.2, 4.0, 4.8$ s, along with distribution $\ J - 1\ _{L^2}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	82
8.8	Swinging cube: Geometry.	83
8.9	Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the velocity, based on the closed-form (8.2) and the ALE mixed formulation (2.76). A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	86
8.10	Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the first Piola Kirchhoff stress tensor, based on the closed-form (8.2) and the ALE mixed formulation (2.76). A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	86
8.11	L-shaped block: Problem setup.	87
8.12	L-shaped block: Presentation of the S1, S2 and S3 meshes.	88
8.13	L-shaped block: Pressure distribution at time $t = 24$ s using, from the left to the right, the S1, S2 and S3 meshes together with the proposed ALE method and the S3 mesh together with the TLF method. Results are obtained with the impulse boundary condition (8.5). A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	88
8.14	L-shaped block: Time evolution of the mesh deformation at time $t = 0.1, 0.2, 0.3, \dots, 1.0, 1.1$ s (left to right, top to bottom), along with contours of $\ \mathbf{u}_\Psi\ $, when solving for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S2. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	89
8.15	L-shaped block: Time evolution of the deformation at time $t = 0, 1, 2, 3, \dots, 19$ s (left to right, top to bottom), along with pressure contours, when solving for the $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	90
8.16	L-shaped block: Time evolution of the global linear momentum for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	91

8.17 L-shaped block: Time evolution of the global angular momentum for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	91
8.18 L-shaped block: Time evolution of a) the global energy components for mesh S3, and b) the dissipation for meshes S1, S2 and S3, and the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition 8.5. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	92
8.19 L-shaped block: Time evolution of the velocity components at a) point $A = (0, 10, 0)^T$, b) point $B = (6, 0, 0)^T$, and c) point $C = (3, 3, 3)^T$ for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition 8.5, and using meshes S1, S2 and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	92
8.20 Bending column: Problem setup.	93
8.21 Bending column: Snapshots of the mesh deformation at time $t = 0.25s$ for different values of the parameter $\beta = 0, 1, 1.5, 2$ (left to right, top to bottom) along with contours of the mesh motion magnitude when solving for the $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	94
8.22 Bending column: Time evolution of the deformation at time $t = 0, 0.1, 0.2, \dots, 1.9s$ (left to right, top to bottom) along with the pressure distribution when solving for the $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	95
8.23 Bending column: Time evolution of a) the global energy components on mesh S3, and b) the total energy on meshes S1, S2 and S3, for the Total Lagrangian formulation and the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. A neo-Hookean model and the volumetric potential in [239] is used with parameters in Table 8.15. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.16.	96
8.24 Bending column: Time evolution of the spatial a) velocity x component and b) the displacement x component at point $A = (0.5, 6, 0.5)^T$ for the Total Lagrangian formulation and for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained on meshes S2, S2 and S3. A neo-Hookean model and the volumetric potential in [239] is used with parameters in Table 8.15. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.16.	96
8.25 Taylor impact bar: Sketch of the problem.	97
8.26 Taylor impact bar: Meshes U1, U2, and U3.	99
8.27 Taylor bar impact: Snapshots of the material deformation (top row) and spatial deformation (bottom row, zoomed view of the lower section) using the ALE formulation for $\alpha_{ALE} = 0, 0.5, 0.75, 1$ on mesh U1. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	101

8.28 Taylor bar impact: Refinement analysis at time $t = 80 \mu s$ using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	102
8.29 Taylor bar impact: Snapshots of the equivalent plastic strain at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	103
8.30 Taylor bar impact: Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	104
8.31 Taylor bar impact: Snapshots of the deformation with pressure contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	105
8.32 Taylor bar impact: (a) Total energy $E = E_{\chi}/J_{\Psi}$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	106
8.33 Taylor bar impact: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	106
8.34 Taylor bar impact: Comparison of spatial deformation together with von-Mises contour plot at time $t = 0, 10, 20, 30, 40 \mu s$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu s$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	107
8.35 Necking bar: Sketch of the problem.	108
8.36 Dynamic axisymmetric necking: Spatial (right side) and material (left) deformation at final time $t = 25 \mu s$, using the ALE formulation with an ALE parameter $\alpha_{ALE} = 0, 0.5, 0.75, 1$ and $\mu_{ALE} = 1$, on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	109
8.37 Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times $15, 20, 25 \mu s$, using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	110

8.38	Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times 15, 20, 25 μs , using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U2. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	111
8.39	Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times 15, 20, 25 μs , using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	112
8.40	Dynamic axisymmetric necking: Spatial (right side) and material (left) deformation at final time $t = 25 \mu s$, using the ALE formulation with an ALE parameter $\alpha_{ALE} = 0, 0.5, 0.75, 1$ and $\mu_{ALE} = 0.02$, on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	113
8.41	Dynamic axisymmetric necking: (a) energy components, for the Total Lagrangian and the ALE formulation on mesh U3, and (b) Total energy $E = E_{\chi}/J_{\Psi}$ on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	113
8.42	Dynamic axisymmetric necking: Time evolution of a) bottom radius, and b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.	114
9.1	Translation with heat: Snapshots of the deformation at time $t = 0, 0.8, \dots, 4.8s$, along with distribution $\ J - 1\ _{L^2(\Omega_{\chi})}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	117
9.2	Translation with heat: Snapshots of the deformation at time $t = 0, 0.8, \dots, 4.8s$, along with distribution $\ \theta - \theta_R\ _{L^2(\Omega_{\chi})}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	118
9.3	Translation with heat: Time evolution of the L^2 -error in Jacobian and temperature when using (red) the complete ALE system, (green) a reduced system without solving variable J_{Ψ} and (blue) a further reduced system without solving variables $\mathbf{F}_{\Psi}, J_{\Psi}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	119
9.4	Translation with heat: Time evolution of the L^2 error in velocity components when using (red) the complete ALE system, (green) a reduced system without solving variable J_{Ψ} and (blue) a further reduced system without solving variables $\mathbf{F}_{\Psi}, J_{\Psi}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.	119

- 9.5 Translation with heat: Snapshots of the deformation at time $t = 0, 0.02, 0.04, 0.06, 0.08, 0.0885s$, along with distribution $\|J - 1\|_{L^2(\Omega_\chi)}$ when solving for $\{\mathbf{p}_\chi \mathbf{F}_\Phi J_\Phi\}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3. 120
- 9.6 Translation with heat: Snapshots of the deformation at time $t = 0, 0.02, 0.04, 0.06, 0.08, 0.0885s$, along with distribution $\|\theta - \theta_0\|_{L^2(\Omega_\chi)}$ when solving for $\{\mathbf{p}_\chi \mathbf{F}_\Phi J_\Phi\}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3. 120
- 9.7 Swinging cube with heat: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the velocity, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3. 123
- 9.8 Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the first Piola Kirchhoff stress tensor, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3. 125
- 9.9 Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3. . 126
- 9.10 L-shaped block (heat): Snapshots of mesh deformation at time $t = 0, 0.1, \dots, 1, 1.1s$ (left to right, top to bottom) the ALE formulation with a prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9 and using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. 128
- 9.11 L-shaped block (heat): Refinement analysis at time $t = 23s$ for the ALE formulation using meshes S1, S2 and S3, and for the Total Lagrangian formulation using mesh S3 (from left to right). Representation of the temperature contours (top half) and pressure contours (bottom half). Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 129

- 9.12 L-shaped block (heat): Sequence of deformation with pressure contours at times $t = 0, 20, 20, 25, 30$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 130
- 9.13 L-shaped block (heat): Sequence of deformation with temperature contours at times $t = 0, 20, 20, 25, 30$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 131
- 9.14 L-shaped block (heat): Time evolution of the (a) energy components on mesh S3 using the Total Lagrangian formulation and the ALE formulation, and (b) the total energy on meshes S1, S2 and S3 using the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 131
- 9.15 L-shaped block (heat): Time evolution of the components of the global linear momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 132
- 9.16 L-shaped block (heat): Time evolution of the components of the global angular momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 132
- 9.17 L-shaped block: Time evolution of the velocity components at point (a) $A = (0, 10, 0)^T$, (b) $B = (6, 0, 0)^T$, and (c) $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, and the discontinuous temperature profile (9.4), and using meshes S1, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 132

- 9.18 L-shaped block: Time evolution of the temperature at point (a) point $A = (0, 10, 0)^T$, (b) point $B = (6, 0, 0)^T$, and (c) point $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, and the discontinuous temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 133
- 9.19 L-shaped block (heat): Refinement analysis at time $t = 23$ s for the ALE formulation using meshes S1, S2 and S3, and for the Total Lagrangian formulation using mesh S3 (from left to right). Representation of the temperature contours (top half) and pressure contours (bottom half). Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 134
- 9.20 L-shaped block (heat): Time evolution of the deformation with pressure contours at time $t = 23$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 135
- 9.21 L-shaped block (heat): Time evolution of the deformation with temperature contours at time $t = 23$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9. 136
- 9.22 L-shaped block (heat): Time evolution of the (a) energy components on mesh S3 using the Total Lagrangian formulation and the ALE formulation, and (b) the total energy on meshes S1, S2 and S3 using the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.4), and using mesh S3. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 136
- 9.23 L-shaped block (heat): Time evolution of the components of the global linear momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13. 137

9.24 L-shaped block (heat): Time evolution of the components of the global angular momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	137
9.25 L-shaped block: Time evolution of the velocity components at point (a) $A = (0, 10, 0)^T$, (b) $B = (6, 0, 0)^T$, and (c) $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, the linear temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	137
9.26 L-shaped block: Time evolution of the temperature at point (a) point $A = (0, 10, 0)^T$, (b) point $B = (6, 0, 0)^T$, and (c) point $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, the linear temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.	138
9.27 Taylor bar impact with thermal coupling: Refinement analysis at time $t = 80 \mu s$ with contour plots of von-Mises stress magnitude (left) and equivalent plastic strain (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	140
9.28 Taylor bar impact with thermal coupling: Comparison of material deformation together at time $t = 0, 20, 40, 80 \mu s$ using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	141
9.29 Taylor bar impact with thermal coupling: Snapshots of the equivalent plastic strain at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	142
9.30 Taylor bar impact with thermal coupling: Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	143
9.31 Taylor bar impact with thermal coupling: Snapshots of the deformation with pressure contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	144

9.32 Taylor bar impact with thermal coupling: Snapshots of the deformation with temperature contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	145
9.33 Taylor bar impact with thermal coupling: (a) Total energy $E = E_{\chi}/J_{\Psi}$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	146
9.34 Taylor bar impact with thermal coupling: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	146
9.35 Taylor bar impact with thermal coupling: Comparison of spatial deformation together with temperature contour plot at time $t = 0, 10, 20, 30, 40 \mu s$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu s$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	147
9.36 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Refinement analysis at time $t = 80 \mu s$ with contour plots of von-Mises stress magnitude (left) and temperature (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	148
9.37 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	150
9.38 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Snapshots of the deformation with temperature contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	151
9.39 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): (a) Total energy $E = E_{\chi}/J_{\Psi}$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	152
9.40 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	152

9.41 Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Comparison of spatial deformation together with temperature contour plot at time $t = 0, 10, 20, 30, 40 \mu s$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu s$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	153
9.42 Bar necking with thermal coupling: Refinement analysis at time $t = 30$ ms with contour plots of von-Mises stress magnitude (left) and temperature (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	155
9.43 Bar necking impact with thermal coupling: Snapshots of the von-Mises contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	156
9.44 Bar necking impact with thermal coupling: Snapshots of the pressure contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	157
9.45 Bar necking impact with thermal coupling: Snapshots of the equivalent plastic strain at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	158
9.46 Bar necking impact with thermal coupling: Snapshots of the temperature contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	159
9.47 Bar necking with thermal coupling: (a) Total energy, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	160
9.48 Bar necking with thermal coupling: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	160
9.49 Bar necking with thermal coupling: Snapshots spatial deformation together with temperature and plastic strain contours at time $t = 0, 5, 10, 15$ ms (first column, top to bottom) and $t = 20, 25, 28, 30$ ms (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	161

B.1	Local traction and normal vector.	175
B.2	Comparison of volumetric potentials using models presented in Table B.1. .	177

LIST OF TABLES

5.1	Examples of ALE mixed formulations with a prescribed material motion. . .	52
5.2	Examples of ALE mixed formulations with solved material mesh motion. . .	53
6.1	Facial contribution to a boundary node a for a boundary condition on spatial and material quantities.	64
6.2	Nodal contribution to a boundary node a for a boundary condition on spatial and material velocities.	64
8.1	Summary of boundary conditions.	78
8.2	Translation: Table of physical parameters.	79
8.3	Translation: Table of parameters for the prescribed mesh motion in Equation (8.1).	79
8.4	Summary of boundary conditions.	83
8.5	Swinging cube: Table of parameters for the physical model and for the closed-form solution.	84
8.6	Swinging cube: Table of parameters for the prescribed mesh motion in Equation (8.1).	84
8.7	Swinging cube: numerical values for the L^1 -error of the velocity components, based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	84
8.8	Swinging cube: numerical values for the L^1 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	85
8.9	Swinging cube: numerical values for the L^2 -error of the velocity components, based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	85

8.10	Swinging cube: numerical values for the L^2 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.	85
8.11	Summary of boundary conditions.	87
8.12	L-shaped block: Table of parameters.	87
8.13	L-shaped block: Table of parameters for the mesh motion.	88
8.14	Summary of boundary conditions.	93
8.15	Bending column: Table of parameters.	93
8.16	Translation: Table of parameters for the prescribed mesh motion in Equation (8.1).	93
8.17	Summary of boundary conditions.	97
8.18	Taylor impact bar: Table of parameters.	98
8.19	Taylor bar impact: Table of parameters for the mesh motion.	98
8.21	Taylor bar impact: speed-up chart comparing the values of time increments for the Total Lagrangian and the ALE formulation. Figures are rounded to the third decimal. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	98
8.20	Taylor bar impact: final bottom radii chart for the Total Lagrangian and the ALE formulation using different meshes. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.	100
8.22	Summary of boundary conditions.	108
8.23	Necking bar: Table of parameters for the mesh motion in Equation (8.1).	108
9.1	Translation with heat: Table of physical parameters.	116
9.2	Swinging cube with heat: Table of parameters for the physical model and for the closed-form solution.	122
9.3	Swinging cube: Table of parameters for the prescribed mesh motion in Equation (8.1).	123
9.4	Swinging cube with heat: numerical values for the dimensionless L^1 -error of the velocity components and the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.	124
9.5	Swinging cube with heat: numerical values for the dimensionless L^1 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.	124

9.6	Swinging cube with heat: numerical values for the dimensionless L^2 -error of the velocity components and the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.	124
9.7	Swinging cube with heat: numerical values for the dimensionless L^2 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.	125
9.8	L-shaped block (heat): Table of parameters.	127
9.9	L-shaped block (heat): Table of parameters for the prescribed mesh motion in Equation (8.1).	127
9.10	Taylor impact with thermal coupling: Table of parameters.	139
9.11	Taylor bar impact with thermal coupling: speed-up chart comparing the values of time increments for the Total Lagrangian and the ALE formulation. Figures are rounded to the third decimal. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.	141
10.1	Novelties of this thesis	165
B.1	Volumetric energy potentials.	177

LIST OF ALGORITHMS

1	Hencky-based von Mises plasticity with isothermal hardening.	34
2	Newton-Raphson procedure.	36
3	Hencky elasticity model with von-Mises plasticity and Johnson-Cook hardening.	45
4	Update of \mathbf{P}_W	50
5	Workflow of the visco-plasticity algorithm, featuring the multi-stage Runge-Kutta time integrator.	75

7.1	<code>vcALEFoam.C</code>	69
7.2	<code>solidModel.C</code>	71
D.1	<code>saveOldTime.H</code>	186
D.2	<code>gradients.H</code>	186
D.3	<code>rhs.H</code>	187
D.4	<code>weakBCs.H</code>	188
D.5	<code>integrator.H</code>	190
D.6	<code>updateVariables.H</code>	190
D.7	<code>strongBCs.H</code>	191
D.8	<code>updateStress.H</code>	192
D.9	<code>backwardEuler.H</code>	193

ABBREVIATIONS

2TG	Two-step Taylor-Galerkin
ALE	Arbitrary Lagrangian Eulerian
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CATIA	Computer-Aided Three-dimensional Interactive Application
CCFVM	Cell-Centred Finite Volume Method
CEL	Coupled Eulerian-Lagrangian
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrichs-Lewy
CSD	Computational Structure Dynamics
CSM	Computational Solid Mechanics
DGCL	Discrete Geometric Conservation Law
EALE	Efficient ALE
FE	Finite Element
FEM	Finite Element Method
FSI	Fluid-Structure Interaction(s)
FVM	Finite Volume Method
GCL	Geometric Conservation Law
HDG	Hybridizable Discontinuous Galerkin
IVP	Initial Value Problem
JST	Jameson-Schmidt-Turkel
LBB	Ladyzhenskaya-Babuška-Brezzi
MPI	Message Passing Interface
MUSCL	Monotonic Upstream-centered Scheme for Conservation Laws
OF	OpenFOAM
OpenFOAM	Open-source Field Operation And Manipulation
PDE	Partial Differential Equation(s)
PG	Petrov-Galerkin
PIMPLE	PISO-SIMPLE
PISO	Pressure-Implicit with Splitting of Operators
QUICK	Quadratic Upstream Interpolation for Convective Kinematics
RHS	Right Hand Side
RITSS	Remeshing and Interpolation Technique by Small Strain
RK	Runge-Kutta

RTT	Reynold's Transport Theorem
SIMPLE	Semi-Implicit Method for Pressure Linked Equations
SPH	Smoothed Particle Hydrodynamics
SUPG	Streamline Upwind Petrov-Galerkin
TLF	Total Lagrangian Formulation
TVD	Total Variation Diminishing
TVD-RK	Total Variation Diminishing Runge-Kutta
URL	Updated Reference Lagrangian
VCFV	Vertex-Centred Finite Volume
VCFVM	Vertex-Centred Finite Volume Method
VMS	Variational Multi Scale
WENO	Weighted Essentially Non-Oscillatory

LIST OF SYMBOLS

Greek symbols

$\bar{\tau}$	von-Mises equivalent stress
$\bar{\varepsilon}_p$	Equivalent plastic strain
\bar{E}	Young modulus
χ	Referential coordinates
δ	Hardening exponent parameter
$\dot{\bar{\varepsilon}}_p$	Equivalent plastic strain rate
$\dot{\bar{\varepsilon}}_p^0$	Initial equivalent plastic strain rate
η	Entropy density
η_χ	Scaled entropy density
γ	Plastic multiplier
Ψ	Helmholtz free energy
κ	Bulk modulus
λ	First Lamé parameter
α	State variable
σ_F	Conjugate of F
σ_H	Conjugate of H
τ	Kirchhoff stress
μ	Second Lamé parameters (shear modulus)
ν	Poisson's ratio
Ω	Material configuration
Ω_χ	Referential configuration
Ω_x	Spatial configuration
$\partial\Omega$	Boundary of the material configuration
$\partial\Omega_\chi$	Boundary of the referential configuration
$\partial\Omega_x$	Boundary of the spatial configuration
ρ_R	Material density
σ_J	Conjugate of J
θ	Temperature magnitude
θ_{melting}	Melting temperature
$\theta_{\text{transition}}$	Transition temperature
θ_R	Reference temperature
$\tilde{\eta}_R$	Entropy density caused by deformation \mathcal{X} after temperature returned to reference value θ_R
τ_∞	Limit yield stress
τ_y	Yield stress
τ_y^0	Initial yield stress
Θ	Temperature magnitude in terms of $\mathcal{X}_{\eta\alpha}$

ϑ	Delta of temperature
ϑ_χ	Scaled delta of temperature
Φ	Material mapping
Ψ	Spatial mapping
φ	Natural mapping
ϕ_f	von-Mises plasticity yield criterion

Latin symbols

\dot{D}_f	Rate of dissipation for a state variable
α	
\dot{w}_p	Rate of plastic dissipation
E	Cartesian vector
F_Φ	Spatial deformation gradient
F_Ψ	Material deformation gradient
Γ_0	Mie-Grüneisen thermal coefficient
\dot{v}	Spatial velocity
H_Φ	Spatial cofactor
H_Ψ	Material cofactor
J_Φ	Spatial Jacobian
J_Ψ	Material Jacobian
$C_{N_x N_x}$	Acoustic tensor
\mathcal{S}	Tensorial source term expressed in the referential configuration
\mathcal{S}_R	Tensorial source term expressed in the material configuration
\mathcal{U}	Tensorial quantity expressed in the referential configuration
\mathcal{U}_R	Tensorial quantity expressed in the material configuration
\mathcal{B}	Ballistic energy
\mathcal{B}_χ	Scaled ballistic energy
\mathcal{E}	Internal energy density in terms of $\mathcal{X}_{\eta\alpha}$
\mathcal{E}'	Deviatoric component of the internal energy
\mathcal{F}	Flux vector
\mathcal{F}_I	Flux vector in the I -th direction
\mathcal{F}_{N_x}	Normal flux vector
\mathcal{H}	Hamiltonian functional
\mathcal{L}	Lagrangian functional
\mathcal{S}	Scalar source term expressed in the referential configuration
\mathcal{S}_R	Scalar source term expressed in the material configuration
\mathcal{U}	Scalar quantity expressed in the referential configuration

\mathcal{U}_R	Scalar quantity expressed in the material configuration	$\mathcal{X}_{\theta\alpha}$	Triplet of the geometric measures, temperature and variable: $\{\mathbf{F}, \mathbf{H}, J, \theta, \alpha\}$
\mathcal{E}	Internal energy density in terms of $\nabla_{\mathbf{x}}\varphi, \eta, \alpha$	\mathcal{X}_{θ}	Triplet of the geometric measures and temperature: $\{\mathbf{F}, \mathbf{H}, J, \theta\}$
\mathbf{b}	Left Cauchy-Green/Finger stress tensor	$V_{\mathbf{X}}$	Material velocity pre-multiplied by the transpose of the material cofactor
\mathbf{b}_e	Left Cauchy-Green/Finger elastic stress tensor	\mathbf{f}	Scaled body forces
\mathbf{b}_p	Left Cauchy-Green/Finger plastic stress tensor	\mathbf{f}_R	Body forces
\mathbf{C}	Right Cauchy-Green stress tensor	\mathbf{p}	Spatial linear momentum
\mathbf{C}_e	Right Cauchy-Green elastic stress tensor	$\mathbf{p}_{\mathbf{X}}$	Scaled linear momentum
\mathbf{C}_p	Right Cauchy-Green plastic stress tensor	$\mathbf{p}\mathbf{W}$	Material linear momentum
\mathbf{F}	Natural deformation gradient	\mathbf{Q}	Heat flux vector
\mathbf{H}	Natural cofactor	$\mathbf{Q}_{\mathbf{X}}$	Scaled heat flux vector
\mathbf{l}	Velocity gradient	\mathbf{t}	Traction vector
\mathbf{P}	First Piola Kirchhoff stress tensor	$\mathbf{t}_{\mathbf{X}}$	Referential traction vector
\mathbf{P}	equivalent First Piola Kirchhoff stress tensor for the material linear momentum	\mathbf{v}	Natural velocity
\mathbf{S}	Second Piola-Kirchhoff stress tensor	\mathbf{W}	Material velocity
$\mathbf{N}_{\mathbf{X}}$	Referential outward pointing normal vector	\mathbf{X}	Material coordinates
$\mathbf{N}_{\mathbf{X}}$	Material outward pointing normal vector	\mathbf{x}	Spatial coordinates
$\mathbf{N}_{\mathbf{x}}$	Spatial outward pointing normal vector	C	Johnson-Cook parameter
\mathbf{P}_{Φ}	Spatial stress conjugate	c	Wave speed
\mathbf{P}_{Ψ}	Material stress conjugate	c_p	Pressure wave speed
q'	Johnson-Cook exponent parameter	c_s	Shear wave speed
\mathcal{F}	Scalar source term expressed in the referential configuration	c_v	Specific heat at constant volume
\mathcal{F}_R	Scalar source term expressed in the material configuration	$c_{\mathbf{X}}$	Scaled wave speed
\mathfrak{F}	Tensorial source term expressed in the referential configuration	dV	Volume measure on the material configuration
\mathfrak{F}_R	Tensorial source term expressed in the material configuration	$dV_{\mathbf{X}}$	Volume measure on the referential configuration
$\tilde{\mathcal{E}}$	Internal energy density in terms of $\mathcal{X}_{\theta\alpha}$	$dV_{\mathbf{x}}$	Volume measure on the spatial configuration
$\tilde{\mathcal{E}}_R$	Internal energy density per unit undeformed volume caused by deformation \mathcal{X} after temperature returned to the reference value θ_R	E	Total energy
$\tilde{\mathcal{E}}'_R$	Non-thermally coupled distortional component of internal energy	$E_{\mathbf{X}}$	Scaled total energy
\tilde{U}_R	Isothermal component of the volumetric potential	H	Hardening/softening modulus
\mathcal{X}	Triplet of the geometric measures: $\{\mathbf{F}, \mathbf{H}, J\}$	J	Natural Jacobian
$\mathcal{X}_{\eta\alpha}$	Triplet of the geometric measures, entropy and variable: $\{\mathbf{F}, \mathbf{H}, J, \eta, \alpha\}$	K	Kinetic energy
\mathcal{X}_{η}	Triplet of the geometric measures and entropy: $\{\mathbf{F}, \mathbf{H}, J, \eta\}$	p	Pressure
		q	Mie-Grüneisen material coefficient
		s	Heat source term
		$s_{\mathbf{X}}$	Scaled heat source term
		t	Time
		U	Volumetric component of the internal energy
		$\nabla_{\mathbf{x}} \times$	Curl operator with respect to \mathbf{x}
		$\nabla_{\mathbf{X}} \times$	Curl operator with respect to \mathbf{X}
		$\nabla_{\mathbf{x}} \times$	Curl operator with respect to \mathbf{x}
		$\nabla_{\mathbf{x}} \cdot$	Divergence operator with respect to \mathbf{x}
		$\nabla_{\mathbf{X}} \cdot$	Divergence operator with respect to \mathbf{X}
		$\nabla_{\mathbf{x}} \cdot$	Divergence operator with respect to \mathbf{x}
		$\nabla_{\mathbf{x}}$	Gradient operator with respect to \mathbf{x}
		$\nabla_{\mathbf{X}}$	Gradient operator with respect to \mathbf{X}
		$\nabla_{\mathbf{x}}$	Gradient operator with respect to \mathbf{x}
		$\llbracket \cdot \rrbracket$	Jump operator
		$\mathcal{L}_v(\mathbf{b}_e)$	Lie derivative of \mathbf{b}_e with respect to \mathbf{v}

CHAPTER 1

INTRODUCTION

“Tant que vous ne savez pas décrire correctement un problème, ce que vous croyez être une solution, en fait sera tiré au hasard; et quand vous tirez au hasard, la probabilité que vous atteignez votre cible est extrêmement faible. ”

— JEAN-MARC JANCOVICI, 2019

1.1 Motivation

High-energy impact or stretching problems are dynamic events involving large deformations where a high amount of energy is imparted onto a structure in a very short period of time [7–11]. They are encountered in a broad array of areas including aerospace, defence, metal forming and transportation. These events can occur in various situations, such as explosions, collisions, or the penetration of a projectile. In the context of large deformations, the material response can exhibit a variety of physical phenomena such as a large accumulation of plastic deformation [12, 13], shear banding [14–16], fragmentation [17, 18], and failure [18–20]. Additionally, important fluctuations of the temperature magnitude can also affect the material response [21–23], leading to thermal softening [15, 24–26] or even melting [27, 28]. A careful study of these events is crucial to better understand the behaviour of materials under extreme loading conditions, and for the development of improved structures.

Another challenge in simulating high-energy impact problems is to obtain the solution of the nonlinear equations of primary variables. These problems require advanced numerical methods that are able to accurately capture complex dynamics at stake such as the propagation of shock waves [29–32]. Shock waves are intense compression waves that can cause significant deformation and damage in a material. In commercial software, the simulation of problems involving shocks is traditionally based on the finite element method and the governing equations are expressed in Lagrangian form. Extensive research [33, 34] is dedicated to better comprehend the underlying mechanisms behind wave propagations in solid dynamics, to improve the computational technologies. Our research group contributed to this endeavour with, for instance, the works of Lee et al. [2], Aguirre et al. [35], Haider et al. [36], Campos et al. [37], by notably using Riemann-based stabilisation technology borrowed from CFD.

However, commercial actors propose turnkey software solutions whose technology can be limited in complex situations such as large deformations and thermal phenomena. Since industries rely on this practice, denoted as Computer-Aided Engineering (CAE), to analyse and optimise their activity, it is crucial to propose enhanced techniques. Software in CAE is typically composed of a wide range of tools (e.g. 3D geometry tool sets, mesh generators [38, 39]) and numerical frameworks to solve complex engineering problems [40]. Concretely, these programs can be utilised to identify potential design flaws early in the development process, to check that products meet specific reliability and performance requirements, or as a common working platform for which no laboratory testing is conceivable. They can also be used as a mean of comparison in lieu of analytical solutions [20, 36, 41, 42]. Commercial solutions (ABAQUS [43], ANSYS [44], ALTAIR RADIOSS [45], COMSOL [46], PAM-CRASH [47]) are versatile development platforms widely used in industries, but can be rather limited in the development of new features because of their inflexible implementation. On the other hand, open-source software (OpenFOAM [48–50], SALOME [51], FREEFEM [52]) offers great extensibility capacities when used and improved by a large community. OpenFOAM has originally been developed as an explicit CFD package based on the cell-centred finite volume method. However, it offers limited solutions for solid mechanics and for using the Vertex-Centred Finite Volume Method, which motivated its use in this work.

1.2 State of the art in solid dynamics

A myriad of real-life applications involving structural bodies have been computationally simulated in the past years by academic and industrial actors. Novel numerical strategies play a crucial role in the industrial and academic progress. In the field of biomedical engineering, the study of blood vessels [53], shock-wave therapy [20], heart valves [54] and

stent-like structure [55, 56] are of growing interest. Additionally, the numerical simulations of implosions [23, 57], blast propagation [29] collapse of a beryllium shell [58], and fast projectile impacts [59] are of concern in defence. In the context of FSI, it is common to use visual frameworks on notorious problems such as water slamming in a hull [60] and tank sloshing [61], free oscillating cantilevers [62, 63], spacecraft parachutes [64, 65], or turbomachinery [66]. Visual examples of various applications can be viewed on Figure 1.1.



(a) Spacecraft atmospheric entry.



(b) Penetration of a projectile.



(c) Car crash testing.



(d) Friction Stir Welding.

FIGURE 1.1: Applications of high-strain CSM: (a) Aerospace¹, (b) Defence², (c) Automotive industry³, and (d) Metallurgy⁴.

1.2.1 Tetrahedral meshes

Tetrahedral meshes are made of polyhedral elements enclosed by four triangular faces. In Computational Solid Mechanics (CSM), and more especially in commercial software, the use of linear tetrahedral elements has grown in popularity in the past years due to their ability to accurately represent complex geometries with mature widespread technologies such as automated Delaunay and advancing front algorithms [39, 67].

Tetrahedral meshes have the ability to provide high-quality elements, which leads to more accurate solutions and reduced computational time (see for instance the work of Schneider et al. [68]). They are robust and typically allow for more accurate representation of curved surfaces and better modelling of complex geometries [69]. However, the evaluation and storage of unknown variables is substantially costly in the case of elaborate constitution laws such as inelastic materials due to their high order. This motivates the use of simpler linear tetrahedra. Nevertheless, they present a number of other shortcomings. First, tetrahedral-based finite element methods are typically not easy to implement, and may suffer from a lack of robustness and accuracy [69]. Second, tetrahedra can perform poorly in some situations (such as bending) due to spurious pressure oscillations [70], and volumetric

¹Credits: https://www.esa.int/Science_Exploration/Human_and_Robotic_Exploration/Exploration/Aerothermodynamics_of_aerocapture_and_high_speed_Earth_re-entry.

²Credits: https://commons.wikimedia.org/wiki/File:Challenger_2_Tank_Firing_a_Shell_at_Night_MOD_45157416.jpg.

³Credits: <https://www.globalncap.org/news/car-to-car-test-demonstrates-zero-star-double-standard>.

⁴Credits: <https://www.globalncap.org/news/car-to-car-test-demonstrates-zero-star-double-standard>.

[71] and shear locking [72]. This is especially witnessed in near incompressibility scenarios (where the Poisson's ratio is close to 0.5) [73]. Third, the order or convergence for stresses is usually reduced in comparison with the displacements [74], (this is not only the case for tetrahedra) . Eventually, high frequency noise can be witnessed in the vicinity of shocks and sharp gradients [75] when using Newmark-type schemes [76].

Recent works have addressed some of the aforementioned issues. Gil et al. [76–78] and Lee et al. [2, 79, 80] paved the way for a new mixed formulation that was used in conjunction with a tetrahedral mesh to recover equal order of convergence for stresses and displacements. That investigation was further continued in the research group by Aguirre [81], Hassan et al. [82], and in this work. Alternatively, Abboud et al. [83] was able to recover the convergence of stresses by means of a variational multiscale method. Moreover, the use of tetrahedral elements [24, 83, 84] can be compared to using hexahedra [85–88] in the context of plasticity. Rectangular elements usually allow for deformations of better quality, and tetrahedra may struggle in near-incompressibility scenarios but they usually require a significantly smaller amount of computational resource and storage [89, 90].

The progress in mesh generation techniques has led to the development of new algorithms for higher-quality tetrahedral meshes [91]. The use of parallel computing has made it possible to solve large-scale problems using tetrahedral meshes, which was previously hindered by the high computational cost. The reader is invited to refer to [68, 83, 92, 93] for a more exhaustive insight on tetrahedral meshes.

1.2.2 Finite Volume Methods

The Finite Volume Method (FVM) consists in the partition of a continuous domain into a set of discrete volumes, or cells, and uses the conservation laws to relate the flux of a quantity across the boundaries of each cell. There are numerous extensive studies on this class of methods (the reader could refer to Eymard et al. [94] for a recent study of the convergence of Finite Volume Methods). Its main advantages reside in the conservation of primary variables over distorted geometries at the discrete level, and in its easy application on unstructured grids.

The Cell-Centred Finite Volume Method (CCFVM) is a type of Finite Volume Method traditionally used in Computational Solid Mechanics. It is based on the idea of dividing the continuous domain into a collection of small, discrete control volumes or cells [36, 93, 95]. In the CCFVM, the primary variables are defined at the centroid of cells, and constitute the approximation of the solution throughout the entire cell. One of the main advantages of the CCFVM is its ease of implementation. Material properties are assumed to be constant within each control volume and the integration of the PDE is performed over the control volume.

The Vertex-Centred Finite Volume Method (VCFVM) is a numerical method used to solve a set of PDE that constitute the behaviour of materials, and where unknown variables are located at mesh vertices, or nodes [56, 58, 81, 82, 96]. One of the main advantages of the VCFVM is its flexibility in mesh use. It can be applied to unstructured meshes, which allows for more freedom in the geometry of the domain being modelled. This can make it easier to handle complex boundaries and shapes, such as those found in engineering and geophysical applications. Additionally, the VCFVM is robust with respect to the choice of mesh, meaning that the solution will not change significantly if the mesh is refined or coarsened. Moreover the use of high order interpolation and reconstruction techniques allows for more accurate solutions and faster convergence. Furthermore, the VCFVM is relatively easy to implement compared to other methods, as it does not require the use of a matrix solver or the solution of a system of equations, with regards to the node per element ratio [69]. Nonetheless, it can be difficult to parallelize the VCFV method [97] in comparison to the other methods because of the difficulty to reconstruct fluxes at a given

interface [98].

1.2.3 Mixed methodologies

Mixed formulations originate from the Hu-Washizu variational principle [99] and initially relied on the simultaneous use of displacement-based and stress-based variables to describe deformation of a continuum. Mixed formulations address numerical issues traditionally encountered in problems featuring nearly incompressible materials, where the volumetric strains accumulate numerical errors and result in an unstable representation of stresses, by introducing dependent/coupled variables solved in a system of equations. They are particularly useful when dealing with non-linear materials and with hyperelastic and viscoelastic materials, and more generally in problems involving large deformations and nonlinearities. The mixed formulations in solid mechanics require the solution of both kinematic and equilibrium equations, which can be challenging to implement. However, they can handle large deformations and nonlinearity more effectively than other traditional methods, which can lead to more accurate and reliable solutions. An early mixed formulation was used on a system of two dimensional first-order conservation laws in Wilkins [100] with a staggered approach. Later, a $\{\mathbf{p}, \mathbf{F}\}$ mixed system [101–104] was introduced for two dimensional CCFVM small strain problems in the linear elastic regime. In the past few years, extended research has been produced on a new type of mixed methodology for solid dynamics:

- | | |
|---------------------------------------|---------------------------------|
| 1. 2D HDG [105], | 8. 3D JST SPH [80], |
| 2. 2D upwind CCFVM [2], | 9. 3D SUPG SPH [108], |
| 3. 2D 2TG FEM [106], | 10. 3D JST VCFVM [56, 82, 108], |
| 4. 3D JST VCFVM [81], | 11. 3D upwind-SPH [109], |
| 5. 3D stabilised PG FEM [79], | 12. 3D VMS PG [23], |
| 6. 3D upwind VCFVM [58] | 13. 3D URL SPH [20, 37], |
| 7. 3D upwind CCFVM [36, 75, 97, 107], | 14. 3D explicit VCFVM [110]. |

This methodology consists in a system of first-order nonlinear conservation laws, including balance laws and geometric conservation equations. Initially, the linear momentum-deformation gradient $\{\mathbf{p}, \mathbf{F}\}$ mixed formulation [2] showcased the same order of accuracy for velocities and stresses, which permitted the use of low order elements without exhibiting volumetric locking in nearly incompressible scenarios. It was supplemented by the conservation of the Jacobian J [111], the cofactor \mathbf{H} [78], uncoupled energy E [81] and coupled entropy η [23]. In the work of [112], a mixed $\{\mathbf{p}, \mathbf{F}, E\}$ formulation of hyperbolic conservation was presented for thermo-visco-plastic applications.

1.2.4 Arbitrary Lagrangian Eulerian formulations

Arbitrary Lagrangian-Eulerian (ALE) methods are a class of numerical methods that are widely used in solid mechanics. These methods provide a way to model the motion of a solid body by describing it in an Eulerian frame of reference while using Lagrangian coordinates to describe the internal deformation of the body [113, 114]. The ALE method is particularly useful in the simulation of problems that involve large deformations such as those encountered in elastodynamics [4], when plasticity is involved [6, 83]. The principle of the ALE method is to introduce a referential domain that is fixed, used for motion description, and separate from the material and spatial domains. The computational mesh neither attaches to the material (Lagrangian description) nor remains fixed in space (Eulerian description), but it partially follows the material points to reduce element distortion

One of the main advantages of the ALE method is its ability to handle large deformations and topological changes. Because the method uses a fixed background grid, it is suitable for problems where the geometry of the domain changes dramatically over time, such as transient multiphysics or Fluid-Structure Interactions [115]. Additionally, the ALE method can be used in problems with strong shocks and high-speed impacts, where large deformations and topological changes may occur. ALE algorithms are also known to be compatible with non-conforming meshes, which are seen in problems involving contact, friction, free surfaces or moving boundaries. The mesh regularization is a key element of the ALE method, and the prohibitive cost of some methods (e.g. Laplacian Smoothing) or the introduction of new restrictions (e.g. Transfinite Mapping Methods [113]) may become an important setback in the setup of the numerical problem. Moreover, the ALE method is notoriously sensitive to the choice of the background grid and the mesh quality, and may require additional techniques (e.g. topological changes, mesh adaptation).

A relatively recent and thorough meta-analysis of the ALE method in Computational Solid Dynamics (CSD), Computational Fluid Dynamics (CFD) [116] and Fluid-Structure Interactions (FSI) [115], is done in Donea et al. [113]. ALE methods most prominently consist in giving an expression to a material velocity \mathbf{W} so that it can be combined with a physical velocity \mathbf{v} or a spatial velocity $\hat{\mathbf{v}}$. The most cost-effective method is to prescribe the mesh motion by means of an analytical function, or to compute it with user-defined strategies [117]. However, complex and more versatile approaches rely on mesh regularisation, which is done in two phases. First, the position of nodes belonging to the material boundaries has to be determined. The material boundary surfaces can be known *a priori* [118, 119] via the velocity by using a Lagrangian approach. This methodology was used by Kohl et al. [18, 120] and applied to crack propagation problems where the crack path is known in advance. Otherwise, boundaries with unknown motion need to be tracked. In the works of Haber et al. [121–123], the initial configuration was treated as an unknown in the same way as the current deformed state, and mixed-like formulation was implemented on a Finite Element solver. In the work of Hughes et al. [117]), the same idea that initial and current configurations are solved separately is taken to a transient Finite Element solver. Moreover, important numerical considerations must be enforced for stability, such as the symmetry condition $\mathbf{W} \cdot \mathbf{N}_{\mathbf{X}} = 0$ along the material surface brought to light by Huerta et al. [124]. After boundaries are characterised, the second phase is dedicated to the interior mesh motion and is denoted as interior mesh rezoning. Transfinite mapping methods [125–127], originally used for mesh generation, can describe also simple surfaces at a low computational cost, but are highly restricted on mesh topologies [128–132]. Throughout the development of ALE frameworks [133], several mesh motion techniques with roots in physical considerations emerged. Laplacian Smoothing⁶, used for the first time by Winslow et al. [134, 135] and Hansbo [136], consists in solving a Poisson equation for the nodal mesh velocity field, resulting in lines of equal potential. That can lead to poor performance in nonconvex domains and a high computational cost. Hermansson et al. [137] proposed a similar approach focused on avoiding element folding. On the other hand, the Laplacian smoothing technique with variable diffusivity was applied to mesh velocity on unstructured grids in Löhner et al. [138] in the context of Fluid-Structure interactions. In Masud et al. [139] and Kanchi et al. [140], constraints were added to improve element quality in boundary layers. Further works such as Brackbill et al. [141] were dedicated to improve Winslow’s stencil but little benefits were obtained, and problems such as mesh overlapping were not completely addressed. A mesh deformation technique based on the biharmonic operator was presented in Helenbrook [142], and showed great interest as the motion was not relying on a second-order operator and had a better tolerance to large

⁶Also referred to as elliptic mesh generation, or equipotential relaxation, the technique of Winslow is among the oldest mesh motion technique and originally used for mesh generation.

deformations despite a higher computational cost. The use of explicit-implicit mesh update technique is reportedly used by Hughes et al. [143] and Belytschko et al. [144] in an attempt to reduce the computational cost, but it has received little success to the difference of behaviour concerning the advection of material points across the domain regarding time integration. Alternatively, in the works of Ghosh et al. [145], Hughes et al. [146], Argyris et al. [147], and Ortiz et al. [148], the use of unconditionally stable implicit methods yielded second-order accuracy and uniformity of the mesh motion over the entire domain but was extremely costly. Mesh motion based on a fractional-step method were used in explicit quadrilateral Finite Element formulations in Benson et al. [149–151] and Liu et al. [152, 153] for elasto-plastic applications. An interior mesh smoothing approach relies on geometric quality measures, such as iterative averaging procedures [154–158] or on the local minimisation of element squeezing and distortions. Ahn et al. [159] uses weighted residuals between element shapes and desired size. The advantage of this rezoning lies in its simplicity and its general character. Mesh adaptation is another type of strategy used in ALE frameworks during the mesh correction step [31, 160]. Notably, Askes et al. [161] used r-adaptive spatial mesh relocation where the mesh motion is controlled by local plasticity indicators. that technique was later coupled with h-adaptive mesh refinement in [162, 163]. In the work of Cescutti et al. [164], an adaptive mesh refinement based on the minimisation of a functional of the mesh distortion is used. Knupp et al. [165] proposes the Reference Jacobian method [166] as a rezoning strategy to improve the mesh quality, resulting in an accurate, robust and flexible ALE framework. More recently, the local mesh rezoning was employed by Shashkov [167] with Voronoi tessellation. The methodology of Barlow et al. [116] is to further extend the inverse harmonic operator of Winslow [168].

Due to the presence of permanent and complex deformations, the ALE has seen an interest in plasticity applications. Among the numerous examples, one can cite the simulation of forming processes [119, 128, 131, 158, 169–172] and impacts [6, 173]. In the relatively recent works of [6, 174], a conservative approach is employed on a Finite Element framework to solve the convection of mesh particles where the conservation law of the material velocities depend on an associated material stress field. As it is the case for physical balance laws in Lagrangian approaches, the new material equations of motion can be numerically treated in the following distinct ways [151]:

- unsplit method (also denoted as direct ALE, particle tracking or particle-based method), where the evolution equation for stresses is integrated in time and implemented in the weak form of the balance of momentum [175, 176]. This approach is utilised by Liu et al. [118] in an explicit time-stepping algorithm, and by Ghosh et al. [145] who used an implicit unsplit formulation.
- split method (fractional-step/mesh-based method/indirect ALE [116]), which consists in resolving the ALE equations in two different phases: a material (Lagrangian) phase followed by a convection (Eulerian/transport) phase. This method is suitable for upgrading Lagrangian formulation to ALE as only the addition of the convection step is required. A non-exhaustive list of examples includes the works of Huétink et al. [119] (implicit split formulation for metal-forming processes), Huerta et al. [177] (fast-transient explicit split formulation), Rodríguez-Ferran et al. [178] (hyperelastic split), and [174] (ALE operator split applied to thermomechanical inelastic materials).
- Hybrid methods. For instance in [6], an advection problem is solved exactly (unsplit) whereas plastic internal variables are split. Alternatively, In [179], the remap stage consists in a hybrid approach that combines the approximate (swept) flux approach in single-material regions and the intersection approach in multi-material regions.

One can also mention [180], where three FE-based methods were tested in geotechnical engineering: Remeshing and Interpolation Technique by Small Strain (RITSS) [181, 182],

Efficient ALE (termed EALE) [149, 183–185] and Coupled Eulerian–Lagrangian (CEL) [186–189]. This mixed methodology motivates the use of mixed formulations. The CFD community mainly uses Eulerian frameworks, but ALE methods have also been used in complex problems [190] and in the study of FSI [115], and since it is beyond the scope of this work, the reader is directed to [113] as well as [191, 192].

1.3 Scope and outline of the thesis

This thesis delves into the fundamental principles of ALE conservation laws in Computational Solid Mechanics (CSM). Different inelastic models will be examined, and the hyperbolicity of the system will be studied. A mesh motion solver, based on the conservation of the material linear momentum, will be presented and different models for solving mesh equilibrium will be proposed. Then, numerical methods used to solve the ALE equations will be discussed. They have two facets: the VCFVM will be applied to ALE equations in the context of solid mechanics, together with the use of an acoustic Riemann solver; the multi-stage Total Variation Diminishing Runge-Kutta (TVD-RK) method will be used as time integrator of the scheme. The open-source platform OpenFOAM was chosen for the implementation of the novel set of ALE conservation laws. The challenges that arise in their implementation will be addressed. By the end of this thesis, the reader will have an understanding of the fundamental principles of the ALE method, their application in various problems, and the numerical methods used to solve the governing equations. This knowledge will provide a strong foundation for further research and practical applications in the field of solid mechanics.

Outline of the thesis

- Chapter 2: *Arbitrary Lagrangian Eulerian Formulation for Isothermal Hyperelasticity*
This chapter introduces ALE conservation laws and how they can be derived from a Total Lagrangian formulation, together with their jump conditions.
- Chapter 3: *Isothermal Inelasticity*
For closure of the ALE problem, polyconvex hyperelastic and elasto-plastic constitutive models are presented. Hyperbolicity is also discussed via the ALE eigenproblem.
- Chapter 4: *Extension to thermal inelasticity*
This chapter presents how the framework can be adapted to include the laws of thermodynamics. This includes the conservation of energy in the ALE framework, constitutive thermo-mechanical volumetric coupling, and isotropic strain-rate hardening with thermal softening.
- Chapter 5: *Material motion*
This chapter presents the conservation law governing the mesh motion.
- Chapter 6: *Numerical discretisation*
In this chapter, the vertex-centred finite volume method is applied to discretise the ALE conservation law. The time integration by means of a multi-stage Runge-Kutta method is also presented.
- Chapter 7: *OpenFOAM implementation*
This chapter presents the implementation of the ALE mixed formulation on the open-source platform OpenFOAM.

- Chapter 8: *Numerical simulations: Isothermal Cases*
The ALE formulation is used in this chapter to solve numerical cases in an isothermal elasticity and elasto-plasticity context.
 - Chapter 9: *Numerical simulations: Thermoelasticity and thermoplasticity*
In this chapter, the ALE formulation is used to solve numerical cases where thermal effects are considered.
 - Chapter 10: *Concluding remarks*
The novelties and the future research opened-up by this work are presented in this chapter.
-
- Appendix A: *Mathematical foundations*
This appendix contains a recollection of the main definition and properties useful for the understanding of the ALE formulation.
 - Appendix B: *Fundamentals of solid mechanics*
A brief summary of the basic concepts of solid mechanics, and some additional results are presented in this appendix.
 - Appendix C: *Second law of Thermodynamics*
This appendix is dedicated to the study of the Ballistic energy to prove the second law of thermodynamics for an ALE system.
 - Appendix D: *OpenFOAM components*
This appendix contains additional information about the numerical framework.

CHAPTER 2

ARBITRARY LAGRANGIAN EULERIAN FORMULATION FOR ISOTHERMAL HYPERELASTICITY

*“We have to remember that what we observe is not nature herself,
but nature exposed to our method of questioning.”*

— WERNER HEISENBERG, *Physics and Philosophy*

2.1 Introduction

In the field of solid mechanics, the study of conservation laws is an essential aspect for the understanding of the behaviour of structures under various loads and conditions. Specifically, the study of conservation laws in the context of the Arbitrary Lagrangian-Eulerian framework is crucial for modelling problems involving large deformations and/or moving boundaries [113]. This chapter delves into the fundamental ALE conservation laws, including the balance of mass and the geometric mappings, and the balance of linear momentum.

The ALE method is a powerful tool for modelling problems involving large deformations and/or moving boundaries, and consists in the following: a referential (fixed) domain is introduced and used for the description of the motion, separate from the material domain (Lagrangian description) and the spatial domain (Eulerian description), and so that the computational mesh is neither attached to the material nor kept fixed in space. This approach aims at combining the advantages of both Lagrangian and Eulerian approaches. The ALE method for computational dynamics was extensively studied for CFD and CSM problems [113], and its theoretical description is elegantly and thoroughly presented in [4, 5]. The Lagrangian description is used to track the motion of individual particles within the solid, while the Eulerian description is used to describe the deformation of the solid as a whole.

This chapter first provides an overview of the tools needed to understand the ALE methodology, including the quantities used to describe the continuum and the kinematics of a deforming solid. The balances quantities and the geometric mappings will be presented and provided with a conservation equation. The main objective of this chapter is to provide a comprehensive understanding of the inner mechanism of the ALE framework, including the continuum description and kinematics, as well as giving the set of basic conservation laws to be used in the context of solid mechanics.

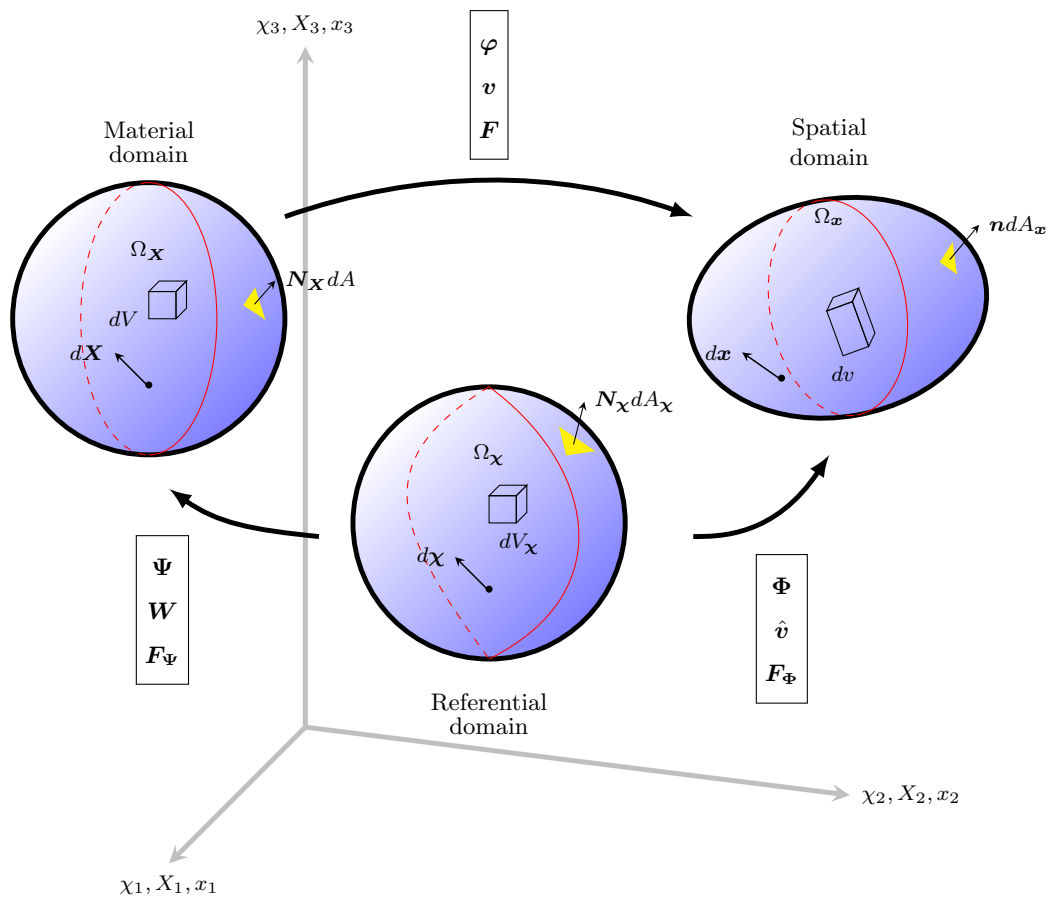
2.1.1 Notation and useful properties

The notation used in this thesis plays an important role in the attempt to convey the key concept of the ALE formulation in a clear manner. This work stems from the works of Armero [6, 193], Steinmann [3–5], Huerta and Donea [113, 177], Rodriguez and Ferran [178], Lee [21, 79, 80, 108, 109], Gil [76–78, 194] and Bonet [10, 195–198], and the concepts borrowed from these previous works will be combined under the notation of this work.

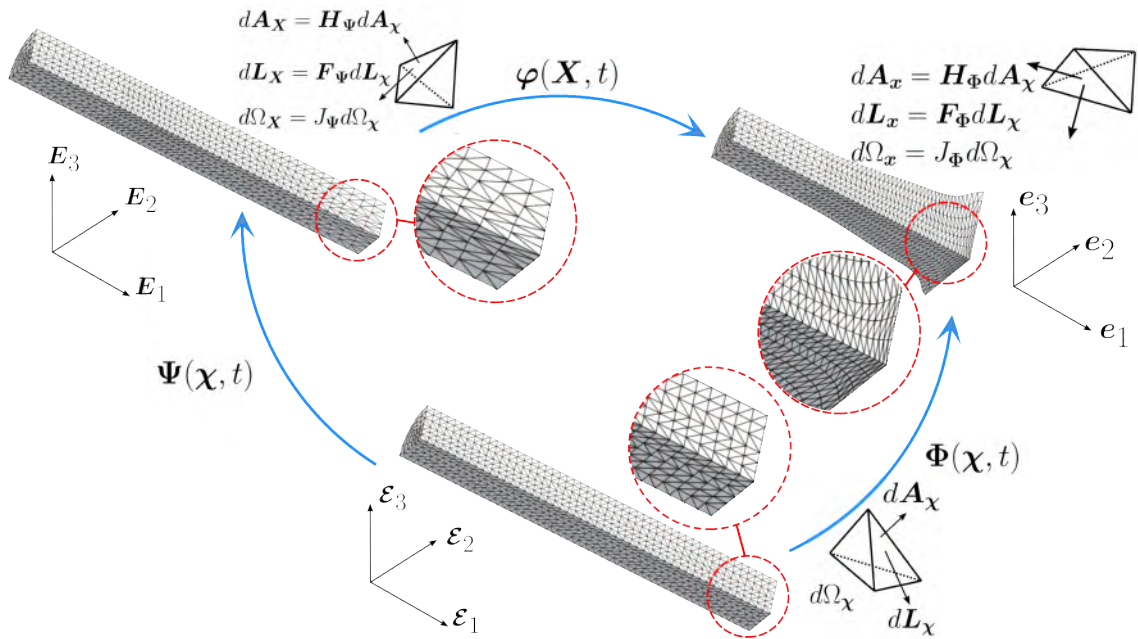
First and second order tensors will be respectively written in lower case and upper case bold font (i.e. \mathbf{v} , \mathbf{P}). Scalar quantities will be written in normal weight font (e.g. k). The mathematical operators and properties used in this thesis are shown in the Appendix A. The reader is invited to refer to the literature (e.g. [199]) for a more exhaustive list of properties, as well as the previous publications of the group [2, 20, 75, 81, 82]. In particular, the tensorial cross product [196] and the thermodynamics in [22, 110] are utilised in this work.

2.2 Continuum description and kinematics

The description of particle motion and kinematics in solid mechanics typically resorts to using a material and a spatial domain that interact with each other using deformation mappings. The Figure 2.1 depicts a) a sketch of the geometric mappings that will be introduced in this chapter as well as b) a visual description based on a numerical example. The two classical approaches [3, 6] to describe a particle's motion are: a) the Lagrangian framework, where fixed material positions \mathbf{X} are followed through the ambient space as



(a) Sketch of the three geometric mappings.



(b) Example of a numerical example with complementary information.

FIGURE 2.1: ALE kinematics description.

the solid deforms in the spatial domain and, b) the Eulerian framework, where material positions \mathbf{X} are followed through the ambient material at fixed spatial positions \mathbf{x} . Particles in the material domain are mapped to the spatial domain by the mapping φ , denoted as natural, and defined as

$$\begin{aligned}\varphi: \Omega_{\mathbf{X}} &\rightarrow \Omega_{\mathbf{x}} \\ \mathbf{X} &\mapsto \mathbf{x} = \varphi(\mathbf{X}, t).\end{aligned}\tag{2.1}$$

Its corresponding velocity \mathbf{v} ⁷, and deformation gradient \mathbf{F} are defined as

$$\mathbf{v} = \left. \frac{\partial \varphi}{\partial t} \right|_{\mathbf{X}}; \quad \mathbf{F} = \frac{\partial \varphi}{\partial \mathbf{X}}.\tag{2.2a,b}$$

The deformation gradient \mathbf{F} , or fibre map, is a two-point tensor mapping line quantities and the Jacobian is used to map volumes. Areas can be mapped with the cofactor tensor \mathbf{H} , defined using the tensorial cross product (see the works of Boer [200] and Bonet et al. [196]) as

$$\mathbf{H} = \frac{1}{2} \mathbf{F} \times \mathbf{F},\tag{2.3}$$

Alternatively, the cofactor can be defined as

$$\mathbf{H} = J \mathbf{F}^{-T}.\tag{2.4}$$

As for volumes, the geometric map that relates a material volume with a spatial volume is the Jacobian J , expressed as

$$J = \det \mathbf{F}.\tag{2.5}$$

The ALE formulation is characterised by the consideration of an additional independent and fixed reference domain [3], where objects can be bijectively mapped to the material and spatial domains. Particles are mapped from the referential domain to the material domain by means of the mapping Ψ defined as

$$\begin{aligned}\Psi: \Omega_{\chi} &\rightarrow \Omega_{\mathbf{X}} \\ \chi &\mapsto \mathbf{X} = \Psi(\chi, t).\end{aligned}\tag{2.6}$$

Its corresponding material velocity \mathbf{W} , and material deformation gradient \mathbf{F}_{Ψ} are defined as

$$\mathbf{W} = \left. \frac{\partial \Psi}{\partial t} \right|_{\chi}; \quad \mathbf{F}_{\Psi} = \frac{\partial \Psi}{\partial \chi}.\tag{2.7a,b}$$

Particles can also be mapped onto the spatial domain with the mapping Φ defined as

$$\begin{aligned}\Phi: \Omega_{\chi} &\rightarrow \Omega_{\mathbf{x}} \\ \chi &\mapsto \mathbf{x} = \Phi(\chi, t).\end{aligned}\tag{2.8}$$

Its corresponding spatial velocity $\hat{\mathbf{v}}$, and spatial deformation gradient \mathbf{F}_{Φ} are defined as

$$\hat{\mathbf{v}} = \left. \frac{\partial \Phi}{\partial t} \right|_{\mathbf{x}}; \quad \mathbf{F}_{\Phi} = \frac{\partial \Phi}{\partial \chi}.\tag{2.9a,b,c}$$

⁷The material time derivative is a differential notation widely used in solid mechanics and corresponds to the time derivative with fixed material coordinates. It is denoted as [113]

$$\dot{\mathbf{v}} = \frac{d\mathbf{v}}{dt} = \left. \frac{\partial \mathbf{v}}{\partial t} \right|_{\mathbf{x}}.$$

Using the above notation, the time rate of the natural deformation gradient can be written as $\dot{\mathbf{F}} = \nabla_{\mathbf{x}} \mathbf{v}$.

The material and spatial cofactors are defined as

$$\mathbf{H}_\Psi = \frac{1}{2} \mathbf{F}_\Psi \times \mathbf{F}_\Psi, \quad \mathbf{H}_\Phi = \frac{1}{2} \mathbf{F}_\Phi \times \mathbf{F}_\Phi. \quad (2.10a,b)$$

They can alternatively be defined as

$$\mathbf{H}_\Psi = J_\Psi \mathbf{F}_\Psi^{-T}, \quad \mathbf{H}_\Phi = J_\Phi \mathbf{F}_\Phi^{-T}. \quad (2.11a,b)$$

The material J_Ψ and spatial J_Φ Jacobians are respectively defined as [196]

$$J_\Psi = \frac{1}{3} \mathbf{H}_\Psi : \mathbf{F}_\Psi; \quad J_\Phi = \frac{1}{3} \mathbf{H}_\Phi : \mathbf{F}_\Phi, \quad (2.12a,b)$$

and can be alternatively expressed as

$$J_\Psi = \det \mathbf{F}_\Psi; \quad J_\Phi = \det \mathbf{F}_\Phi. \quad (2.13a,b)$$

Both Jacobians ensure $J_\Psi, J_\Phi > 0$, hence the existence of an inverse for their respective deformation gradient. It is important to emphasize that the natural mapping φ can be defined as a composition of the material and spatial mappings as

$$\varphi = \Phi \circ \Psi^{-1}. \quad (2.14)$$

Following Equation (2.14), similar relations can be outlined for the natural velocities, gradients and Jacobians as

$$\hat{\mathbf{v}} = \mathbf{v} + \mathbf{F}\mathbf{W}; \quad \mathbf{F} = \mathbf{F}_\Phi \mathbf{F}_\Psi^{-1}; \quad \mathbf{H} = \mathbf{H}_\Psi \mathbf{H}_\Phi^{-1}; \quad J = J_\Phi J_\Psi^{-1}. \quad (2.15a,b,c,d)$$

Tensorial objects are mapped from one configuration to the other due to the geometric mappings (Refer to Figure 2.1 for a visual representation). The deformation gradients (or fibre maps) are two-point geometric tensors that map edges/vector elements. Consider $d\boldsymbol{\chi}$, $d\mathbf{X}$ and $d\mathbf{x}$, respectively elemental vectors in the referential, material and spatial domains, are therefore linked as

$$d\mathbf{X} = \mathbf{F}_\Psi d\boldsymbol{\chi}; \quad d\mathbf{x} = \mathbf{F}_\Phi d\boldsymbol{\chi}; \quad d\mathbf{x} = \mathbf{F} d\mathbf{X}. \quad (2.16a,b,c)$$

Similarly, the Jacobians, or volume maps, are linked as

$$dV = J J_\Psi dV_\chi; \quad dv = J_\Phi dV_\chi; \quad dv = J dV. \quad (2.17a,b,c)$$

Area maps are the cofactor of their respective deformation gradients, and are responsible for mapping geometric surfaces from one domain to another. For the three mappings, they state

$$\mathbf{N}_\mathbf{X} dA = \mathbf{H}_\Psi \mathbf{N}_\chi dA_\chi, \quad (2.18a)$$

$$\mathbf{n} dA_\mathbf{x} = \mathbf{H}_\Phi \mathbf{N}_\chi dA_\chi, \quad (2.18b)$$

$$\mathbf{n} dA_\mathbf{x} = \mathbf{H} \mathbf{N}_\mathbf{X} dA. \quad (2.18c)$$

Additional quantities such as the right Cauchy-Green tensor $\mathbf{C} = \mathbf{F}^T \mathbf{F}$, as well as their useful properties, are discussed in Appendix A.

Remark 2.2.1 *The mixed formulation that is presented in this work includes geometric conservation laws that are represented by the deformation gradient and its minors (the area map \mathbf{H} and the volume map J), for the three mappings. As a consequence, these geometric mappings are considered independent from each other. Nonetheless, they are related through geometric compatibility conditions [196] that are strongly satisfied in the continuum and that can be expressed for the three mappings as*

$$\mathbf{F} = \nabla_{\mathbf{x}}\varphi; \quad \mathbf{H} = [\det(\nabla_{\mathbf{x}}\varphi)](\nabla_{\mathbf{x}}\varphi)^{-T}; \quad J = \det(\nabla_{\mathbf{x}}\varphi), \quad (2.19a,b,c)$$

$$\mathbf{F}_{\Psi} = \nabla_{\mathbf{x}}\Psi; \quad \mathbf{H}_{\Psi} = [\det(\nabla_{\mathbf{x}}\Psi)](\nabla_{\mathbf{x}}\Psi)^{-T}; \quad J_{\Psi} = \det(\nabla_{\mathbf{x}}\Psi), \quad (2.20a,b,c)$$

$$\mathbf{F}_{\Phi} = \nabla_{\mathbf{x}}\Phi; \quad \mathbf{H}_{\Phi} = [\det(\nabla_{\mathbf{x}}\Phi)](\nabla_{\mathbf{x}}\Phi)^{-T}; \quad J_{\Phi} = \det(\nabla_{\mathbf{x}}\Phi). \quad (2.21a,b,c)$$

2.3 ALE transformations

This section presents the steps to transform a conservation law from a Total Lagrangian viewpoint to an Arbitrary Lagrangian Eulerian viewpoint, in global and local form. The transformation will be shown for a scalar and a tensorial generic quantity. It consists of the following steps: a) the re-parametrisation of the right-hand side, composed of time-independent quantities, from the material to the referential domain, b) the use of Reynold's transport theorem on the left hand side to convert the time derivative with respect to material coordinates to a time derivative with respect to referential coordinates, and c) the presentation of the corresponding local forms.

In the following section, it is practical to define the change in a differential operator between a material and a referential configuration. For a vector \mathbf{v} , and using the Nanson's rule and the divergence theorem⁸, it gives

$$\int_{\Omega_{\mathbf{X}}} \nabla_{\mathbf{x}} \cdot \mathbf{v} \, dV = \int_{\partial\Omega_{\mathbf{X}}} \mathbf{v} \cdot \mathbf{N}_{\mathbf{X}} \, dA = \int_{\partial\Omega_{\mathbf{X}}} \mathbf{v} \cdot \mathbf{H}_{\Psi} \mathbf{N}_{\mathbf{X}} \, dA_{\mathbf{X}} = \int_{\Omega_{\mathbf{X}}} \nabla_{\mathbf{x}} \cdot (\mathbf{H}_{\Psi}^T \mathbf{v}) \, dV_{\mathbf{X}}. \quad (2.22)$$

2.3.1 Total Lagrangian Scalar Conservation law

Considering a scalar primary variable \mathcal{U}_R , a flux vector \mathcal{F}_R , a source term \mathcal{S}_R , and a fixed control volume $\Omega_{\mathbf{X}}$ measured with an infinitesimal volume quantity dV and a material surface vector $\mathbf{N}_{\mathbf{X}}dA$. A generic conservation law in global form is defined as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathcal{U}_R \, dV + \int_{\Omega_{\mathbf{X}}} \mathcal{F}_R \cdot \mathbf{N}_{\mathbf{X}} \, dA = \int_{\Omega_{\mathbf{X}}} \mathcal{S}_R \, dV. \quad (2.23)$$

For a smooth solution, the divergence theorem can be applied to Equation (2.23) to obtain

$$\int_{\Omega_{\mathbf{X}}} \left(\frac{\partial \mathcal{U}_R}{\partial t} \Big|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot \mathcal{F}_R - \mathcal{S}_R \right) \, dV = 0, \quad (2.24)$$

⁸The divergence theorem is presented in Appendix A.4.1.

where $\nabla_{\mathbf{x}} \cdot$ is a divergence operator with respect to the material coordinates. That equation holds for every enclosed and fixed material control volume $\Omega_{\mathbf{X}}$. By considering the boundary of $\Omega_{\mathbf{X}}$ as an interface (with other control volumes or with the exterior), it is customary to define the corresponding local form stated for $\Omega_{\mathbf{X}}$ as

$$\left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot \mathcal{F}_R = \mathcal{S}_R, \quad (2.25)$$

paired with the jump condition

$$c[[\mathcal{U}_R]] = [[\mathcal{F}_R]] \cdot \mathbf{N}_{\mathbf{X}}. \quad (2.26)$$

Note that Equation (2.26) features the speed c at which the interface (or discontinuity surface), represented by its normal vector $\mathbf{N}_{\mathbf{X}}$, is propagating in the medium. Also, the jump operator $[[\cdot]]$ represents the difference between the two states of the primary variable on either side of the interface and will be discussed in Chapter 6. That jump equality is denoted as the Rankine-Hugoniot condition for the primary variable \mathcal{U}_R , expressed here in the Lagrangian form.

2.3.2 Total Lagrangian Tensorial Conservation law

Considering a tensor variable \mathbf{U}_R , a flux tensor \mathfrak{F}_R , a source term \mathcal{S}_R , and a fixed control volume $\Omega_{\mathbf{X}}$ measured with an infinitesimal volume quantity $\Omega_{\mathbf{X}}$ and surface vector $\mathbf{N}_{\mathbf{X}} dA$. A generic conservation law in global, or integral form, is defined as

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathbf{U}_R dV + \int_{\Omega_{\mathbf{X}}} \mathfrak{F}_R \mathbf{N}_{\mathbf{X}} dA = \int_{\Omega_{\mathbf{X}}} \mathcal{S}_R dV. \quad (2.27)$$

For a smooth solution, the divergence theorem can be applied (see Section A.4.1) to Equation (2.27) to obtain

$$\int_{\Omega_{\mathbf{X}}} \left(\left. \frac{\partial \mathbf{U}_R}{\partial t} \right|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot \mathfrak{F}_R - \mathcal{S}_R \right) dV = \mathbf{0}, \quad (2.28)$$

where $\nabla_{\mathbf{x}} \cdot$ is a divergence operator with respect to the aforementioned system of coordinates. As for the scalar case, the global form leads to an equivalent local form, for the volume $\Omega_{\mathbf{X}}$ stated as

$$\left. \frac{\partial \mathbf{U}_R}{\partial t} \right|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot \mathfrak{F}_R = \mathcal{S}_R, \quad (2.29)$$

with which the Rankine-Hugoniot condition reads

$$c[[\mathbf{U}_R]] = [[\mathfrak{F}_R]] \mathbf{N}_{\mathbf{X}}. \quad (2.30)$$

2.3.3 Arbitrary Lagrangian Eulerian Scalar Conservation law

A generic Total Lagrangian global scalar conservation equation can be stated as

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathcal{U}_R dV = - \int_{\partial\Omega_{\mathbf{X}}} \mathcal{F}_R \cdot \mathbf{N}_{\mathbf{X}} dA + \int_{\Omega_{\mathbf{X}}} \mathcal{S}_R dV. \quad (2.31)$$

In the generic Equation (2.31) of a conservation law expressed in the Lagrangian form, fluxes and source terms (featured in the right hand side) can be re-parametrised⁹ in terms of referential coordinates, using the definition of the material mapping $\mathbf{X} = \Psi(\boldsymbol{\chi}, t)$, as

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathcal{U}_R dV = - \int_{\partial\Omega_{\boldsymbol{\chi}}} (\mathbf{H}_{\Psi}^T \mathcal{F}_R^*) \cdot \mathbf{N}_{\boldsymbol{\chi}} dA_{\boldsymbol{\chi}} + \int_{\Omega_{\boldsymbol{\chi}}} (J_{\Psi} \mathcal{S}_R^*) dV_{\boldsymbol{\chi}}, \quad (2.32)$$

⁹Note that for a scalar conservation law, the Nanson's integration rule (2.18a) can be rewritten as shown, making use of the scalar product transpose property.

where the notation $[\cdot]_R(\mathbf{X}, t) = [\cdot]_R^*(\Psi(\chi, t), t)$ is used to emphasise the change from material to the referential coordinates. Using Reynold's transport theorem¹⁰ on the material time derivative, the left hand side of Equation (2.31) can now be expressed in terms of referential coordinates as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathcal{U}_R dV = \frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi}} (J_{\Psi} \mathcal{U}_R^*) dV_{\chi} - \int_{\partial\Omega_{\chi}} \mathcal{U}_R^* (\mathbf{H}_{\Psi}^T \mathbf{W}) \cdot \mathbf{N}_{\chi} dA_{\chi}. \quad (2.33)$$

Equations (2.32) and (2.33) can now be used to express the generic Total Lagrangian conservation law (2.31) in the referential domain, as

$$\frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi}} (J_{\Psi} \mathcal{U}_R^*) dV_{\chi} + \int_{\partial\Omega_{\chi}} [\mathbf{H}_{\Psi}^T (\mathcal{F}_R^* - \mathcal{U}_R^* \mathbf{W})] \cdot \mathbf{N}_{\chi} dA_{\chi} = \int_{\Omega_{\chi}} (J_{\Psi} \mathcal{S}_R^*) dV_{\chi}. \quad (2.34)$$

The above equation represents the Arbitrary Lagrangian Eulerian (denoted as ALE) equivalent of said generic Equation (2.31). The equivalent Arbitrary Lagrangian Eulerian local conservation law, associated to the global law (2.34), is

$$\frac{\partial (J_{\Psi} \mathcal{U}_R^*)}{\partial t} \Big|_{\chi} + \nabla_{\chi} \cdot [\mathbf{H}_{\Psi}^T (\mathcal{F}_R^* - \mathcal{U}_R^* \mathbf{W})] = J_{\Psi} \mathcal{S}_R^*, \quad (2.35)$$

and because the magnitude remains constant when the configuration is changed, the notation $[\cdot]_R(\mathbf{X}, t) = [\cdot]_R^*(\Psi(\chi, t), t)$ can be dropped to yield

$$\frac{\partial (J_{\Psi} \mathcal{U}_R)}{\partial t} \Big|_{\chi} + \nabla_{\chi} \cdot [\mathbf{H}_{\Psi}^T (\mathcal{F}_R - \mathcal{U}_R \mathbf{W})] = J_{\Psi} \mathcal{S}_R. \quad (2.36)$$

The associated jump condition is

$$c_{\chi} [J_{\Psi} \mathcal{U}_R] = -[\mathcal{F}_R - \mathcal{U}_R \mathbf{W}] \cdot \mathbf{H}_{\Psi} \mathbf{N}_{\chi}. \quad (2.37)$$

2.3.4 Arbitrary Lagrangian Eulerian Tensorial Conservation law

In this section, the previous development is generalised to second and third order tensors. The variables $\mathbf{U}_R, \mathbf{S}_R$ can either be first- or second-order tensor fields, and \mathfrak{F}_R are consequently either second- or third-order tensor fields. The notation $\mathbf{U}_R(\mathbf{X}, t) = \mathbf{U}^*(\Psi(\chi, t), t)$ is used to specify a change of variable from the material to the referential configurations. A generic Total Lagrangian global tensorial conservation equation can be stated as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathbf{u}_R dV_{\mathbf{X}} = - \int_{\partial\Omega_{\mathbf{X}}} \mathfrak{F}_R \mathbf{N}_{\chi} dA_{\mathbf{X}} + \int_{\Omega_{\mathbf{X}}} \mathbf{S}_R dV_{\mathbf{X}}. \quad (2.38)$$

The Equation (2.38) is then re-parametrised using referential coordinates, as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathbf{u}_R dV_{\mathbf{X}} = - \int_{\partial\Omega_{\chi}} (\mathfrak{F}_R^* \mathbf{H}_{\Psi}) \mathbf{N}_{\chi} dA_{\chi} + \int_{\Omega_{\chi}} (J_{\Psi} \mathbf{S}_R^*) dV_{\chi}. \quad (2.39)$$

Making use of the Reynolds' transport theorem, the left hand side is re-written as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathbf{u}_R dV_{\mathbf{X}} = \frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi}} (J_{\Psi} \mathbf{u}_R^*) dV_{\chi} - \int_{\partial\Omega_{\chi}} (\mathbf{u}_R^* \otimes \mathbf{W}) \mathbf{H}_{\Psi} \mathbf{N}_{\chi} dA_{\chi}. \quad (2.40)$$

The generic Total Lagrangian vectorial conservation law 2.38 is now expressed in the referential domain, as

$$\frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi}} (J_{\Psi} \mathbf{u}_R^*) dV_{\chi} - \int_{\partial\Omega_{\chi}} [\mathfrak{F}_R^* - (\mathbf{u}_R^* \otimes \mathbf{W})] \mathbf{H}_{\Psi} \mathbf{N}_{\chi} dA_{\chi} = \int_{\Omega_{\chi}} (J_{\Psi} \mathbf{S}_R^*) dV_{\chi}. \quad (2.41)$$

¹⁰See Appendix A.4.2 for development of Reynold's transport theorem (RTT).

By dropping the notation $[\cdot]_R(\mathbf{X}, t) = [\cdot]_R^*(\Psi(\boldsymbol{\chi}, t), t)$ as done for the scalar variable in the previous section, the equivalent Arbitrary Lagrangian Eulerian local conservation law, associated to the global law (2.41), is given as

$$\left. \frac{\partial (J_\Psi \mathcal{U}_R)}{\partial t} \right|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot ([\mathfrak{F}_R - (\mathcal{U}_R \otimes \mathbf{W})] \mathbf{H}_\Psi) = J_\Psi \mathcal{S}_R, \quad (2.42)$$

and the associated jump condition is defined as

$$c_\chi [[J_\Psi \mathcal{U}_R]] = -[[\mathfrak{F}_R - (\mathcal{U}_R \otimes \mathbf{W})]] \mathbf{H}_\Psi \mathbf{N}_\chi. \quad (2.43)$$

2.3.5 Non-conservative ALE transformations

The transformations used in the above help to obtain an ALE equation from its Total Lagrangian form. Using the same algebra, it is possible to obtain ALE equations in a non-conservative form¹¹[3–5]. For a scalar quantity \mathcal{U} , the relationship between the material and the referential time derivatives can then be written as

$$\left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\mathbf{X}} = \left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\boldsymbol{\chi}} - (\nabla_{\mathbf{x}} \mathcal{U}_R) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}). \quad (2.44)$$

For completeness, the same development can be followed for a tensorial quantity, and it gives that

$$\left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\mathbf{X}} = \left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\boldsymbol{\chi}} - (\nabla_{\mathbf{x}} \mathcal{U}_R) (\mathbf{F}_\Psi^{-1} \mathbf{W}). \quad (2.45)$$

In this case, note that the conserved variable obtained is not scaled by the material Jacobian, and is equivalent to the conserved variable of the associated Lagrangian conservation law. When solving for a set of ALE conservation laws, it is practical to also compute all the intermediate quantities such as volume sizes, gradients or the mapping velocities on the referential Ω_χ . That being said, it is possible to solve material variables, that is to say \mathcal{U}_R instead of \mathcal{U} . Using Equations (2.22), (2.44) and (2.45), conservation laws can be written in a non-conservative form.

2.4 ALE conservation laws

Using Total Lagrangian to Arbitrary Lagrangian Eulerian transformations established in the previous section, the typical conservation laws used in solid mechanics and based on material coordinates can be transformed and be expressed in referential coordinates,

¹¹Considering the RTT form stated in Equation (A.19), and the following development:

$$\begin{aligned} \left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} \mathcal{U}_R dV &= \left. \frac{\partial}{\partial t} \right|_{\boldsymbol{\chi}} \int_{\Omega_{\boldsymbol{\chi}}} \mathcal{U}_R dV - \int_{\partial \Omega_{\boldsymbol{\chi}}} \mathcal{U}_R \mathbf{W} \cdot \mathbf{N}_{\boldsymbol{\chi}} dA \\ &= \left. \frac{\partial}{\partial t} \right|_{\boldsymbol{\chi}} \int_{\Omega_{\boldsymbol{\chi}}} J_\Psi \mathcal{U}_R dV_{\boldsymbol{\chi}} - \int_{\partial \Omega_{\boldsymbol{\chi}}} \mathcal{U}_R (\mathbf{H}_\Psi^T \mathbf{W}) \cdot \mathbf{N}_{\boldsymbol{\chi}} dA_{\boldsymbol{\chi}} \\ &= \int_{\Omega_{\boldsymbol{\chi}}} \left[J_\Psi \left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\boldsymbol{\chi}} + \mathcal{U}_R \left. \frac{\partial J_\Psi}{\partial t} \right|_{\boldsymbol{\chi}} - \nabla_{\boldsymbol{\chi}} \cdot (\mathcal{U}_R \mathbf{H}_\Psi^T \mathbf{W}) \right] dV_{\boldsymbol{\chi}} \\ &= \int_{\Omega_{\boldsymbol{\chi}}} \left[J_\Psi \left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\boldsymbol{\chi}} - \nabla_{\boldsymbol{\chi}} (\mathcal{U}_R) \cdot (\mathbf{H}_\Psi^T \mathbf{W}) \right] dV_{\boldsymbol{\chi}} \\ &= \int_{\Omega_{\boldsymbol{\chi}}} \left[\left. \frac{\partial \mathcal{U}_R}{\partial t} \right|_{\boldsymbol{\chi}} - \nabla_{\boldsymbol{\chi}} (\mathcal{U}_R) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) \right] dV, \end{aligned}$$

In the above development, the conservation law on the material Jacobian is used to go from the third to the fourth line. It will be defined in the next Section in Equation (2.50).

thence constituting a set of equations expressed in the ALE viewpoint. The global form will be presented, as well as the local form and its associated jump condition. The equations will all be combined in a mixed system of conservation laws. The involutions, or compatibility conditions, will also be presented. They are mathematical constraints [2, 55] inherited from the extensions of elasticity theory into the realm of thermodynamics and electrodynamics [201], ensuring deformations are smooth and free of discontinuities¹². The compatibility condition for the deformation gradient is based on the fact that in a body, the deformation is determined by the displacement field, and states that \mathbf{F} must be a gradient field¹³. Mathematically, it means that the curl of the deformation gradient must be zero. Furthermore, the cofactor is the gradient of the determinant and thence is divergence-free. Naturally, these involutions are valid for tensors along the referential, material and spatial mappings. Their satisfaction is not necessary to close the system but they are crucial for the robustness of the numerical framework, as they let oscillations in certain directions be suppressed and they regulate wave amplitudes [201].

2.4.1 Material Geometric Conservation Law

The principle of mass conservation in a material configuration states that the total time derivative of the density of a system must remain constant, and can be formulated in Lagrangian description as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega} \rho_R dV = 0, \quad (2.46)$$

where ρ_R is the material density of the body. In this work, the density ρ_R will be considered uniformly constant in time and space. This conservation law is generally omitted in solid mechanics single-material frameworks. It can however be used to yield conservation of alternative variables in this Arbitrary Lagrangian Eulerian framework. The Geometric Conservation Law (denoted as GCL, and discussed in [113, 202–205]) is crucial for the stability and accuracy of the ALE framework and can be obtained directly from the above conservation equation (2.46). Using the ALE transformation technique, a scaling is operated from the material domain to the referential domain as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} J_{\Psi} \rho_R dV_{\mathbf{X}} - \int_{\partial\Omega_{\mathbf{X}}} (\rho_R \mathbf{H}_{\Psi}^T \mathbf{W}) \cdot \mathbf{N}_{\mathbf{X}} dA_{\mathbf{X}} = 0. \quad (2.47)$$

Remark 2.4.1 *In the general case of a multi-material domain, Equation (2.47) can be used as a continuity equation for a non-uniform density distribution. The corresponding local form is*

$$\frac{\partial J_{\Psi} \rho_R}{\partial t} \Big|_{\mathbf{X}} - \nabla_{\mathbf{X}} \cdot (\rho_R \mathbf{H}_{\Psi}^T \mathbf{W}) = 0, \quad (2.48)$$

and its associated jump condition is defined as

$$c_{\mathbf{X}} \llbracket J_{\Psi} \rho_R \rrbracket = \llbracket \rho_R \mathbf{W} \rrbracket \cdot \mathbf{H}_{\Psi} \mathbf{N}_{\mathbf{X}}. \quad (2.49)$$

¹²The compatibility conditions are important in computational mechanics because they ensure that geometrical tensors are smooth and free of singularities or discontinuities. This is important for the accurate modelling and simulation of complex systems, such as materials under different loading conditions.

¹³In other words, $\mathbf{F} = \nabla_{\mathbf{x}} \mathbf{x}$. The deformation gradient must be a single-valued tensor field, which means that it must be defined at every point in the body without any discontinuities

For a constant density ρ_R , Equation (2.47) can be expressed as a conservation equation in global form for the material Jacobian with an Arbitrary Lagrangian Eulerian viewpoint. That equation reads

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} J_{\Psi} dV_{\mathbf{x}} - \int_{\Omega_{\mathbf{x}}} \nabla_{\mathbf{x}} \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}) dV_{\mathbf{x}} = 0, \quad (2.50)$$

and the equivalent local conservation law is

$$\frac{\partial J_{\Psi}}{\partial t} \Big|_{\mathbf{x}} = \nabla_{\mathbf{x}} \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}). \quad (2.51)$$

The associated jump condition is

$$c_{\mathbf{x}} \llbracket J_{\Psi} \rrbracket = - \llbracket \mathbf{H}_{\Psi}^T \mathbf{W} \rrbracket \cdot \mathbf{N}_{\mathbf{x}}. \quad (2.52)$$

2.4.2 Conservation of Linear Momentum

The balance of linear momentum per unit of undeformed volume in Lagrangian description can be expressed in integral form as

$$\frac{d}{dt} \int_{\Omega} \mathbf{p} dV = \int_{\partial\Omega} \mathbf{P} \mathbf{N} dA + \int_{\Omega} \mathbf{f}_R dV, \quad (2.53)$$

where $\mathbf{p} = \rho_R \mathbf{v}$ is the linear momentum per unit of undeformed volume $\Omega_{\mathbf{x}}$ in the material configuration, \mathbf{f}_R are the body forces in the material configuration, \mathbf{P} is the natural first Piola Kirchhoff stress tensor, \mathbf{N} is the unit material outward normal and dA is the surface area of the boundary $\partial\Omega_{\mathbf{x}}$ in the material configuration. By using the ALE transformation technique, the equivalent Arbitrary Lagrangian Eulerian global conservation equation for (2.53) is thence

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathbf{p}_{\mathbf{x}} dV_{\mathbf{x}} - \int_{\partial\Omega_{\mathbf{x}}} [\mathbf{P} + (\mathbf{p} \otimes \mathbf{W})] \mathbf{H}_{\Psi} \mathbf{N}_{\mathbf{x}} dA_{\mathbf{x}} = \int_{\Omega_{\mathbf{x}}} \mathbf{f} dV_{\mathbf{x}}, \quad (2.54)$$

where the notation $\mathbf{p}_{\mathbf{x}} = J_{\Psi} \mathbf{p}$ denotes the linear momentum scaled by the material Jacobian J_{Ψ} and that is now expressed in referential coordinates. The body forces $\mathbf{f} = J_{\Psi} \mathbf{f}_R$ have also been scaled by the material Jacobian. An equivalent stress tensor $\mathbf{P}_{\mathbf{x}}$ is defined as

$$\mathbf{P}_{\mathbf{x}} = [\mathbf{P} + (\mathbf{p} \otimes \mathbf{W})] \mathbf{H}_{\Psi}, \quad (2.55)$$

and the corresponding local form of Equation (2.54) is

$$\frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \Big|_{\mathbf{x}} = \nabla_{\mathbf{x}} \cdot \mathbf{P}_{\mathbf{x}} + \mathbf{f}, \quad (2.56)$$

Its associated jump condition can be expressed as

$$c_{\mathbf{x}} \llbracket \mathbf{p}_{\mathbf{x}} \rrbracket = - \llbracket \mathbf{P}_{\mathbf{x}} \rrbracket \mathbf{N}_{\mathbf{x}}. \quad (2.57)$$

2.4.3 Geometric conservation laws

As presented in [198, 206, 207], the deformation gradient can be treated as an independent variable (as opposed to the classical displacement-based formulations). The conservation of the natural deformation gradient \mathbf{F} was studied in [2, 56, 75, 81]. In low order discretisation, the conservation of the deformation gradient aims at avoiding locking (especially in a nearly-incompressible scenario). This work uses a multiplicative decomposition of \mathbf{F} into a

material and spatial components, as presented in Equation (2.15b). The spatial deformation gradient \mathbf{F}_Φ relates vector quantities from the referential domain to the spatial domain. The time rate of \mathbf{F}_Φ can be obtained from its definition in Equation (2.9b), as

$$\left. \frac{\partial \mathbf{F}_\Phi}{\partial t} \right|_{\mathcal{X}} = \nabla_{\mathcal{X}} \cdot (\hat{\mathbf{v}} \otimes \mathbf{I}), \quad (2.58)$$

and it can be provided with an appropriate jump condition

$$c_{\mathcal{X}}[\mathbf{F}_\Phi] = -[[\hat{\mathbf{v}}] \otimes \mathbf{N}_{\mathcal{X}}], \quad (2.59)$$

to yield a corresponding global form

$$\left. \frac{\partial}{\partial t} \right|_{\mathcal{X}} \int_{\Omega_{\mathcal{X}}} \mathbf{F}_\Phi \, dV_{\mathcal{X}} - \int_{\partial\Omega_{\mathcal{X}}} \hat{\mathbf{v}} \otimes \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}} = 0. \quad (2.60)$$

On the other hand, the material deformation gradient \mathbf{F}_Ψ , which corresponds to the deformation gradient of the material motion (see Equation (2.7b)), can also be regarded as a conserved quantity, whose time rate is expressed as

$$\left. \frac{\partial \mathbf{F}_\Psi}{\partial t} \right|_{\mathcal{X}} = \nabla_{\mathcal{X}} \cdot (\mathbf{W} \otimes \mathbf{I}), \quad (2.61)$$

its associated jump condition is

$$c_{\mathcal{X}}[\mathbf{F}_\Psi] = -[[\mathbf{W}] \otimes \mathbf{N}_{\mathcal{X}}], \quad (2.62)$$

and the corresponding global form reads

$$\left. \frac{\partial}{\partial t} \right|_{\mathcal{X}} \int_{\Omega_{\mathcal{X}}} \mathbf{F}_\Psi \, dV_{\mathcal{X}} - \int_{\partial\Omega_{\mathcal{X}}} \mathbf{W} \otimes \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}} = 0. \quad (2.63)$$

Remark 2.4.2 *The compatibility constraints associated to the geometric deformation gradients are stated as*

$$\nabla_{\mathcal{X}} \times \mathbf{F}_\Phi = (\) = \mathbf{0}; \quad \nabla_{\mathcal{X}} \times \mathbf{F}_\Psi = \mathbf{0}; \quad , \quad (2.64a,b)$$

with the CURL operator defined using the Levi-Civita alternating tensor ε as

$$[\nabla_{\mathcal{X}} \times \mathbf{F}]_{iI} = \varepsilon_{IJK} \frac{\partial F_{iK}}{\partial \chi_J}. \quad (2.65)$$

2.4.4 Conservation of other geometric mappings

Although this work makes use of a restricted set of conservation laws, it is possible to extend that set to comprise all material $\{\mathbf{F}_\Psi, \mathbf{H}_\Psi, J_\Psi\}$ and spatial $\{\mathbf{F}_\Phi, \mathbf{H}_\Phi, J_\Phi\}$ geometrical mappings. As material and spatial deformation gradients are already stated in Equations (2.63), it is possible to bring out conservation equations for the material and spatial cofactors stated in global form. They can be respectively stated as

$$\left. \frac{\partial}{\partial t} \right|_{\mathcal{X}} \int_{\Omega_{\mathcal{X}}} \mathbf{H}_\Psi \, dV_{\mathcal{X}} - \int_{\partial\Omega_{\mathcal{X}}} \mathbf{F}_\Psi \times (\mathbf{W} \otimes \mathbf{N}_{\mathcal{X}}) \, dA_{\mathcal{X}} = 0, \quad (2.66)$$

and

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathbf{H}_{\Phi} dV_{\mathbf{x}} - \int_{\partial\Omega_{\mathbf{x}}} \mathbf{F}_{\Phi} \times (\hat{\mathbf{v}} \otimes \mathbf{N}_{\mathbf{x}}) dA_{\mathbf{x}} = 0. \quad (2.67)$$

The local forms are

$$\frac{\partial \mathbf{H}_{\Psi}}{\partial t} \Big|_{\mathbf{x}} - \mathbf{F}_{\Psi} \times \nabla_{\mathbf{x}} \mathbf{W} = 0; \quad \frac{\partial \mathbf{H}_{\Phi}}{\partial t} \Big|_{\mathbf{x}} - \mathbf{F}_{\Phi} \times \nabla_{\mathbf{x}} \hat{\mathbf{v}} = 0, \quad (2.68a,b)$$

and the corresponding jump conditions are

$$c_{\mathbf{x}}[\mathbf{H}_{\Psi}] = -\mathbf{F}_{\Psi} \times [\mathbf{W} \otimes \mathbf{N}_{\mathbf{x}}]; \quad c_{\mathbf{x}}[\mathbf{H}_{\Phi}] = -\mathbf{F}_{\Phi} \times [\hat{\mathbf{v}} \otimes \mathbf{N}_{\mathbf{x}}]. \quad (2.69a,b)$$

Additionally, the spatial Jacobian can also be treated as an variable [198, 207, 208] independent from the displacements, to provide extra flexibility. This Geometric Conservation Law (GCL) [54, 76] was introduced to improve the quality of the solution. An analogous law is yielded for the spatial mapping from the continuity Equation (2.47) as

$$\frac{\partial J_{\Phi}}{\partial t} \Big|_{\mathbf{x}} = \nabla_{\mathbf{x}} \cdot (\mathbf{H}_{\Phi}^T \hat{\mathbf{v}}), \quad (2.70)$$

and its associated jump condition is

$$c_{\mathbf{x}}[J_{\Phi}] = -[\mathbf{H}_{\Phi}^T \hat{\mathbf{v}}] \cdot \mathbf{N}_{\mathbf{x}}. \quad (2.71)$$

The corresponding global conservation equation is

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} J_{\Phi} dV_{\mathbf{x}} - \int_{\partial\Omega_{\mathbf{x}}} (\mathbf{H}_{\Phi}^T \hat{\mathbf{v}}) \cdot \mathbf{N}_{\mathbf{x}} dA_{\mathbf{x}} = 0. \quad (2.72)$$

2.5 ALE Mixed Formulation

A general system in global form can now be given for the ALE conservation laws, where the global conservation Equations (2.50), (2.54), (2.60) and (2.63) are assembled into a system of first-order equations defined as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathbf{u} dV_{\mathbf{x}} + \int_{\partial\Omega_{\mathbf{x}}} \mathcal{F}_{\mathbf{N}_{\mathbf{x}}} dA_{\mathbf{x}} = \int_{\Omega_{\mathbf{x}}} \mathcal{S}_{\mathbf{x}} dV_{\mathbf{x}}, \quad (2.73)$$

where \mathbf{u} is the vector of conserved variables, $\mathcal{F}_{\mathbf{N}_{\mathbf{x}}}$ is the vector of corresponding fluxes, and $\mathcal{S}_{\mathbf{x}}$ is the vector of corresponding source terms. A general local form can also be provided for the conservation laws in ALE form. Noticing that the system of Equations (2.73) in global form can be written as

$$\int_{\Omega_{\mathbf{x}}} \left[\frac{\partial \mathbf{u}}{\partial t} \Big|_{\mathbf{x}} + \sum_{i=1}^3 \frac{\partial \mathcal{F}_I}{\partial \chi_I} - \mathcal{S}_{\mathbf{x}} \right] dV_{\mathbf{x}} = 0, \quad (2.74)$$

where \mathcal{F}_I is the vector of corresponding fluxes in the I -th direction, represented by the Cartesian vector \mathbf{E}_I (defined in Equations A.1), and verifying $\mathcal{F}_{\mathbf{N}_{\mathbf{x}}} = \mathcal{F}_I N_{\chi_I}$ with $\mathbf{N}_{\mathbf{x}} = (N_{\chi_1}, N_{\chi_2}, N_{\chi_3})^T$. The set of first-order local equations that is obtained corresponds to the assembly of local forms (2.51), (2.56), (2.58) and (2.61), written as

$$\frac{\partial \mathbf{u}}{\partial t} \Big|_{\mathbf{x}} + \sum_{i=1}^3 \frac{\partial \mathcal{F}_I}{\partial \chi_I} = \mathcal{S}, \quad (2.75)$$

The terms featured in the ALE mixed formulation (2.75) can be expressed as

$$\mathbf{u} = \begin{bmatrix} \mathbf{p}_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \end{bmatrix}; \quad \mathcal{F}_I = - \begin{bmatrix} \mathbf{P}_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \end{bmatrix}; \quad \mathbf{S} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \end{bmatrix}, \quad (2.76\text{a,b,c})$$

with the spatial velocity vector $\hat{\mathbf{v}}$ defined in Equation (2.9a), the stress \mathbf{P}_χ defined in Equation (2.55). The associated jump conditions, defined in Equations (2.52), (2.57), (2.59) and (2.62), can be written in the compact form $c_\chi[\mathbf{u}] = \llbracket \mathcal{F}_{N_\chi} \rrbracket$ [2, 27, 56, 75, 209] as

$$c_\chi[\mathbf{p}_\chi] = -\llbracket \mathbf{P}_\chi \rrbracket \mathbf{N}_\chi, \quad (2.77\text{a})$$

$$c_\chi[\mathbf{F}_\Psi] = -\llbracket \mathbf{W} \rrbracket \otimes \mathbf{N}_\chi, \quad (2.77\text{b})$$

$$c_\chi[\mathbf{F}_\Phi] = -\llbracket \hat{\mathbf{v}} \rrbracket \otimes \mathbf{N}_\chi, \quad (2.77\text{c})$$

$$c_\chi[J_\Psi] = -\llbracket \mathbf{H}_\Psi^T \mathbf{W} \rrbracket \cdot \mathbf{N}_\chi. \quad (2.77\text{d})$$

The ALE formulation is a generalisation of the Total Lagrangian and the Eulerian formulations. It is shown in Appendix B.5 how ALE equations can degenerate to their Lagrangian and Eulerian counterparts.

Remark 2.5.1 *The Equation (2.76) represents the simplest generic ALE mixed formulation for a hyperelastic material in non-linear solid dynamics. Depending on the physical nature of the problems, it is possible to include additional conservation laws (e.g. the spatial Jacobian J_Φ for locking or incompressible scenarios, an energy-related variable when thermal processes are considered). The numerical setup of the problem may also induce the use of additional conservation laws related to the ALE mappings. These ideas are further discussed for the continuum in Section 5.3 and for the discrete domain in Section 6.6*

2.6 Conclusion

This chapter introduced in Section 2.2 the concept of referential domain and the consequent new mappings and quantities to relate that domain to the classical material and spatial domains, and all this constitutes the based of the ALE method that will be applied in this work. Using the aforementioned notation, Section 2.3 presented how to transform a generic conservation law based on the Lagrangian approach to a conservation law involving scaled variables, new advection terms, and expressed on the referential domain. In Section 2.4, the ALE transformation has then been applied on conservation laws of quantities of interest, namely the balance principles and the geometric mappings. This new set of conservation laws constitute the basis of an ALE framework and is summarised in Section 2.5. To close the system, a constitutive model has to be defined, which is the purpose of Chapter 3.

CHAPTER 3

ISOTHERMAL INELASTICITY

“Le savant doit ordonner ; on fait la science avec des faits comme une maison avec des pierres ; mais une accumulation de faits n’est pas plus une science qu’un tas de pierres n’est une maison.”

— HENRI POINCARÉ, *La science et l’hypothèse*, 1902

3.1 Introduction

For closure of system (2.75), conservation laws must be supplemented by a constitutive material law for stresses, that satisfy the objectivity requirements¹⁴ and the law of thermodynamics [2, 75, 81]. This work utilises hyperelastic models, which are used to describe elastic materials experiencing large deformations [199, 211–217], where the first Piola Kirchhoff stress \mathbf{P} must verify a distortional-volumetric decomposition. The polyconvexity condition will also be discussed for the ALE mixed system, and will be used to determine strain energy potentials [22, 75]. Then the constitutive equation is presented using work conjugacy [218], and examples of general material models are provided. The general ALE mixed system will eventually be extended to isotropic von-Mises plasticity [2, 113, 172, 195, 219].

3.2 Polyconvex elastic models

In hyperelastic materials, different constitutive restrictions on the strain energy potential were established for well-posedness¹⁵ in the sense of Hadamard of the Cauchy problem (2.76). Rank-one convexity (equivalent, for twice differentiable fields, to strong Legendre-Hadamard ellipticity [75, 81, 221–223]) leads to material stability. Moreover, the existence and propagation of real wave speeds in the material (or Hyperbolicity [224]) is an useful condition for the stability of the formulation. The condition of polyconvexity¹⁶ is also crucial as it facilitates the transformation of the mixed formulation into a symmetric system of conservation laws via the introduction of a convex entropy function [23]. Polyconvexity of the constitutive model results in local stability of the system [228]. Ellipticity difficulty holds after the onset of localized deformations, such as in the presence of high plastic strains [221, 229, 230], and this is the reason why the satisfaction of the stronger requirement of polyconvexity is clearly important.

It has been extensively mathematically studied in [8, 199, 213, 214, 231–233]. Previous works also emphasised the importance for the material model to be thermodynamically consistent (see for instance [22, 23, 58, 214, 224, 234, 235]), that is to say to establish the second law of Thermodynamics ensuring the production of entropy.

As it is well-known, the satisfaction of polyconvexity guarantees a) Rank-one convexity¹⁷ and b) Hyperbolicity in the dynamic regime [23].

To guarantee the principle of Hyperbolicity, which is equivalent to the existence of real wave speeds for any deformation, the constitutive laws to close the mixed formulations will need to satisfy the principle of tensor objectivity, and the second law of Thermodynamics will have to be established. In this context and if polyconvexity is also required, the strain energy potential can be formulated as a function of the geometric mapping denoted as $\mathcal{X} = \{\mathbf{F}, \mathbf{H}, J\}$ via a (non strictly) convex¹⁸ multivariate functional \mathcal{E} [216]. As a result,

$$\mathcal{E}(\nabla_{\mathbf{x}}\varphi) = \mathcal{E}(\mathcal{X}), \quad (3.1)$$

¹⁴Also called frame-indifference, it states that *stress components do not fundamentally change due to rigid body motions including rotations and translations* (see [8, 81, 195, 210]).

¹⁵See Equivalence theorem of Lax [220]: *For a well-posed linear initial value problem, stability is the only necessary condition for convergence provided that the numerical method is consistent.*

¹⁶For an extensive study of the concept of polyconvexity, the reader is invited to refer to [210, 213, 225–227].

¹⁷Also denoted as the condition of ellipticity: A strong elliptic system ensures a strict continuous solution well-suited for describing equilibrium states.

¹⁸The strain energy density is convex with respect to its 19 variables.

Remark 3.2.1 *Isotropic hyperelasticity describes the behaviour of isotropic materials exhibiting large deformations. In this theory, the material is modelled as a continuum and the strain energy is a function of the stretches, or equivalently of the stress invariants, which can be stated as*

$$\mathcal{E} = \hat{\mathcal{E}}(\lambda^1, \lambda^2, \lambda^3) = \tilde{E}(I_C, II_C, III_C), \quad (3.2)$$

where the invariants are defined as

$$I_C = \text{tr} \mathbf{C}; \quad II_C = \mathbf{C} : \mathbf{C}; \quad III_C = \det \mathbf{C}. \quad (3.3a,b,c)$$

Consequently \mathbf{P} is a function of the natural deformation gradient \mathbf{F} ¹⁹ that derives from a strain energy potential, describing the amount of energy required to deform the material, as

$$\mathbf{P}(\nabla_{\mathbf{x}}\varphi) = \frac{\partial \mathcal{E}}{\partial \nabla_{\mathbf{x}}\varphi}. \quad (3.4)$$

Work conjugate stresses $\Sigma_{\mathbf{F}}, \Sigma_{\mathbf{H}}, \Sigma_J$, defined for the three strain measures used to describe the energy potential in Equation (3.1) as

$$\Sigma_{\mathbf{F}}(\mathcal{X}) = \frac{\partial \mathcal{E}}{\partial \mathbf{F}}; \quad \Sigma_{\mathbf{H}}(\mathcal{X}) = \frac{\partial \mathcal{E}}{\partial \mathbf{H}}; \quad \Sigma_J(\mathcal{X}) = \frac{\partial \mathcal{E}}{\partial J}, \quad (3.5a,b,c)$$

can be used to express \mathbf{P} , the work conjugate [10, 75, 78, 196, 197] of the rate of the deformation gradient, as

$$\mathbf{P} : \dot{\mathbf{F}} = \dot{\mathcal{E}}(\mathcal{X}) = (\Sigma_{\mathbf{F}} + \Sigma_{\mathbf{H}} \times \mathbf{F} + \Sigma_J \mathbf{H}) : \dot{\mathbf{F}}, \quad (3.6)$$

where the term $\mathbf{P} : \dot{\mathbf{F}}$ refers to the total amount of mechanical work, and the expression of \mathbf{P} naturally emerges as

$$\mathbf{P}(\mathcal{X}) = \Sigma_{\mathbf{F}} + \Sigma_{\mathbf{H}} \times \mathbf{F} + \Sigma_J \mathbf{H} \quad (3.7)$$

3.3 Nearly incompressible constitutive models

Nearly incompressible materials are defined by a Poisson's ratio $\nu > 0 \rightarrow 0.5$ or a bulk to the shear modulus²⁰ ratio $\kappa/\mu \rightarrow \infty$, and therefore experience little volume change under significant load. They constitute a fundamental device to numerically model materials that are notably resistant to compression (e.g. rubbers, soft tissues), and consist in complementing the distortional strain energy with a penalty volumetric potential. This deviatoric-volumetric additive decomposition²¹ [75, 78, 197, 232, 236, 237] of the total strain energy functional is denoted as

$$\mathcal{E}(\mathcal{X}) = \mathcal{E}'(\mathcal{X}) + U(J), \quad (3.8)$$

¹⁹By extension, \mathbf{P} may also depend on temperature, plastic internal variables and other state variables (See Section 3.5 and Chapter 4)

²⁰The Lamé coefficients comprise the shear modulus and the bulk modulus, are expressed in [Pa] or [N/m²] and are respectively defined as

$$\mu = \frac{E}{1 + \nu}; \quad \lambda = \frac{\bar{E}}{3(1 - 2\nu)} = \frac{\bar{E}\nu}{(1 + \nu)(1 - 2\nu)},$$

where \bar{E} is the Young's modulus and ν is the Poisson's ratio.

²¹For hyperelastic models, \mathbf{P} can also be expressed in terms of the invariants of \mathbf{C} .

where \mathcal{E}' is the distortional/deviatoric component and $U(J)$ is the volumetric component. The decomposition of the first Piola-Kirchhoff tensor

$$\mathbf{P}(\mathcal{X}) = \mathbf{P}_{\text{dev}}(\mathcal{X}) + \mathbf{P}_{\text{vol}}(J) \quad (3.9)$$

is then obtained, using Equation (3.7), as

$$\mathbf{P}_{\text{dev}} = \Sigma_{\mathbf{F}} + \Sigma_{\mathbf{H}} \times \mathbf{F} + \hat{\Sigma}_J \mathbf{H}; \quad \mathbf{P}_{\text{vol}} = p \mathbf{H}; \quad \Sigma_J = \hat{\Sigma}_J + p, \quad (3.10\text{a,b,c})$$

where the hydrostatic pressure p and $\hat{\Sigma}_J$ being defined as

$$\hat{\Sigma}_J = \frac{\partial \mathcal{E}'}{\partial J}; \quad p = \frac{\partial U}{\partial J}. \quad (3.11\text{a,b})$$

For instance, a quadratic volumetric energy potential is defined as

$$U(J) = \frac{\kappa}{2}(J - 1)^2, \quad (3.12)$$

where $\kappa = \lambda + 2\mu/3$ is the bulk modulus of the material. Further eligible volumetric energy potentials are presented in Appendix B.3.2.

3.3.1 Isochoric Mooney Rivlin model

The nearly incompressible polyconvex Mooney-Rivlin strain energy potential is defined as [75]

$$\mathcal{E}'(\mathcal{X}) = \zeta J^{-2/3} (\mathbf{F} : \mathbf{F}) + \xi J^{-2} (\mathbf{H} : \mathbf{H})^{3/2} - 3\zeta - 3^{3/2}\xi. \quad (3.13)$$

where ζ, ξ are positive material parameters defined in terms of the shear modulus as $2\zeta + 3\sqrt{3}\xi = \mu$ [232]. The conjugate stresses are

$$\Sigma_{\mathbf{F}} = 2\zeta J^{-2/3} \mathbf{F}; \quad \Sigma_{\mathbf{H}} = 3\xi J^{-2} (\mathbf{H} : \mathbf{H})^{1/2} \mathbf{H}, \quad (3.14\text{a,b})$$

and

$$\hat{\Sigma}_J = -\frac{2}{3}\zeta J^{-5/3} (\mathbf{F} : \mathbf{F}) - 2\xi J^{-2} (\mathbf{H} : \mathbf{H})^{3/2}. \quad (3.15)$$

More specifically, the neo-Hookean model can be obtained by choosing $\zeta = \mu/2$ and $\xi = 0$ [1, 7, 56, 75, 234], and represents the hyperelastic model of choice for non thermally-coupled materials in this work. In this case, the strain energy potential is expressed as

$$\mathcal{E}'(\mathcal{X}) = \frac{\mu}{2} \left(J^{-2/3} \mathbf{F} : \mathbf{F} - 3 \right), \quad (3.16)$$

the expression of the conjugate stresses becomes

$$\Sigma_{\mathbf{F}} = \mu J^{-2/3} \mathbf{F}; \quad \Sigma_{\mathbf{H}} = \mathbf{0}; \quad \hat{\Sigma}_J = -\frac{\mu}{3} J^{-5/3} (\mathbf{F} : \mathbf{F}), \quad (3.17\text{a,b,c})$$

and consequently the deviatoric stress is

$$\mathbf{P}_{\text{dev}} = \mu J^{-2/3} \mathbf{F} - \frac{\mu}{3} J^{-5/3} (\mathbf{F} : \mathbf{F}) \mathbf{H}. \quad (3.18)$$

3.3.2 Nearly incompressible Hencky model

The Hencky model [234, 238] is a hyperelastic model based on logarithmic strains and constitutes the basis to the plasticity algorithm presented in 1. The strain energy in Equation (3.1) is defined in terms of the distortional elastic principal stretches $\hat{\lambda}_{e,\alpha} = J^{-1/3}\lambda_{e,\alpha}$, as

$$\begin{aligned}\hat{\mathcal{E}}'(\hat{\lambda}_{e,1}, \hat{\lambda}_{e,2}, \hat{\lambda}_{e,3}) &= \mu \left(\sum_{\alpha=1}^3 [\ln \hat{\lambda}_{e,\alpha}]^2 \right) \\ &= \mu \left(\sum_{\alpha=1}^3 [\ln \lambda_{e,\alpha}]^2 \right) + \frac{1}{3}\mu (\ln J)^2 - \frac{2}{3}\mu (\ln J) \left(\sum_{\alpha=1}^3 \ln \lambda_{e,\alpha} \right).\end{aligned}\quad (3.19a)$$

The first Piola-Kirchhoff stress tensor \mathbf{P} can finally be reconstructed²² as

$$\mathbf{P} = \boldsymbol{\tau} \mathbf{F}^{-T}, \quad (3.20)$$

and the decomposition in principal components of the symmetric Kirchhoff stress is given as

$$\boldsymbol{\tau} = \sum_{\alpha=1}^3 \tau'_{\alpha\alpha} \mathbf{n}_{\alpha} \otimes \mathbf{n}_{\alpha}; \quad \tau'_{\alpha\alpha} = \tau'_{\alpha\alpha} + Jp. \quad (3.21a,b)$$

For hyperelastic materials, the Hencky-based internal energy consists defined in Equation (3.8) in the sum of the deviatoric potential in Equation (3.19) and a volumetric potential. For a logarithmic volumetric energy potential (used in [239] and listed in Table B.1), it reads

$$\begin{aligned}\mathcal{E}(\boldsymbol{\mathcal{X}}) &= \left[\mu \left(\sum_{\alpha=1}^3 [\ln \hat{\lambda}_{e,\alpha}]^2 \right) \right] + \left[\frac{1}{2} \kappa (\ln J)^2 \right] \\ &= \mu \left(\sum_{\alpha=1}^3 [\ln \lambda_{e,\alpha}]^2 \right) + \frac{1}{3}\mu (\ln J)^2 - \frac{2}{3}\mu (\ln J) \left(\sum_{\alpha=1}^3 \ln \lambda_{e,\alpha} \right) \\ &= \mu \left(\sum_{\alpha=1}^3 [\ln \lambda_{e,\alpha}]^2 \right) + \frac{1}{2} \lambda (\ln J)^2,\end{aligned}\quad (3.22a)$$

by noticing that in Equation (3.22), the Jacobian can be expressed in terms of the logarithmic elastic principal stretches $\lambda_{e,\alpha}$, and that the bulk modulus κ can be expressed in terms of Lamé's parameters as

$$\ln J = \sum_{\alpha=1}^3 \ln \lambda_{e,\alpha}; \quad \kappa = \lambda + \frac{2}{3}\mu. \quad (3.23a,b)$$

3.4 Hyperbolicity

A system of conservation laws is hyperbolic if its Cauchy problem²³ is well-posed. In numerical analysis, a set of hyperbolic conservation laws can be represented by a mixed formulation, as in Equations (2.75) and (2.76), where the Jacobian matrix is diagonalisable

²²Note the relations between stress tensors:

$$\boldsymbol{\tau} = \mathbf{P} \mathbf{F}^T; \quad \tau_{\alpha\alpha} = J \sigma_{\alpha\alpha} = \tau'_{\alpha\alpha} + Jp.$$

²³In particular, an Initial Value Problems (IVP), as found in the literature [2, 240, 241].

with real eigenvalues²⁴ and independent eigenvectors. It is also worth mentioning that hyperbolicity is preserved under a change of gauge, which is analogous to considering a constant energy under orthogonal transformations. It suffices to consider the homogenous variant of the system of conservation laws to prove hyperbolicity [30, 241, 242]. The following will thence briefly remind hyperbolicity for the Total Lagrangian formulation, and present it for the Arbitrary Lagrangian Eulerian mixed formulation.

3.4.1 Hyperbolicity of the ALE system

In this section, the system (2.75) and (2.76) is considered in the context of a prescribed material motion, where the geometric material subset $\mathcal{X}_\Psi = \{\mathbf{F}_\Psi, \mathbf{H}_\Psi, J_\Psi\}$ is assumed to be given. Consequently, the system (2.76) can be reduced to

$$\mathbf{u} = \begin{bmatrix} \mathbf{p}_\chi \\ \mathbf{F}_\Phi \end{bmatrix}; \quad \mathcal{F}_I = - \begin{bmatrix} \mathbf{P}_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \end{bmatrix}; \quad \mathcal{S} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \quad (3.24a,b,c)$$

and using the characteristic equation²⁵

$$\mathbf{D}\mathcal{F}[\mathbf{u}_\alpha^R] = c_{\chi,\alpha} \mathbf{u}_\alpha^R, \quad (3.25)$$

the ALE right eigenproblem is stated as

$$c_{\chi,\alpha} \begin{bmatrix} \mathbf{p}_{\chi,\alpha}^R \\ \mathbf{F}_{\Phi,\alpha}^R \end{bmatrix} = - \begin{bmatrix} \mathbf{D}(\mathbf{P}_\chi \mathbf{N}_\chi) \begin{bmatrix} \mathbf{p}_{\chi,\alpha}^R \\ \mathbf{F}_{\Phi,\alpha}^R \end{bmatrix} \\ \mathbf{D}(\hat{\mathbf{v}} \otimes \mathbf{N}_\chi) \begin{bmatrix} \mathbf{p}_{\chi,\alpha}^R \\ \mathbf{F}_{\Phi,\alpha}^R \end{bmatrix} \end{bmatrix}. \quad (3.26)$$

The second line of Equation (3.26) is then post-multiplied by \mathbf{F}_Ψ^{-1}

$$-c_{\chi,\alpha} \mathbf{F}_\alpha^R = (\mathbf{D}(\hat{\mathbf{v}}) [\mathbf{p}_{\chi,\alpha}^R, \mathbf{F}_{\Phi,\alpha}^R] \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1}; \quad \mathbf{F}_\alpha^R = \mathbf{F}_{\Phi,\alpha}^R \mathbf{F}_\Psi^{-1}. \quad (3.27a,b)$$

Using the coefficient $\Lambda_{\mathbf{H}_\Psi}$ defined as

$$\Lambda_{\mathbf{H}_\Psi} \mathbf{N}_\chi = \mathbf{H}_\Psi \mathbf{N}_\chi; \quad \Lambda_{\mathbf{H}_\Psi} = \sqrt{\mathbf{H}_\Psi \mathbf{N}_\chi \cdot \mathbf{H}_\Psi \mathbf{N}_\chi}, \quad (3.28a,b)$$

and some algebra on Equation (3.27a), the decomposition of $\hat{\mathbf{v}} = \mathbf{v} + \mathbf{F}\mathbf{W}$ yields, for the first term,

$$\begin{aligned} (\mathbf{D}(\mathbf{v}) [\mathbf{p}_{\chi,\alpha}^R] \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1} &= (J_\Psi^{-1} \rho_R^{-1} \mathbf{p}_{\chi,\alpha}^R \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1} \\ &= (\mathbf{v}_\alpha^R \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1} \\ &= (\mathbf{v}_\alpha^R \otimes \mathbf{H}_\Psi \mathbf{N}_\chi) J_\Psi^{-1} \\ &= \frac{\Lambda_{\mathbf{H}_\Psi}}{J_\Psi} \mathbf{v}_\alpha^R \otimes \mathbf{N}_\chi, \end{aligned} \quad (3.29)$$

and for the second term

$$\begin{aligned} (\mathbf{D}(\mathbf{F}\mathbf{W}) [\mathbf{F}_{\Phi,\alpha}^R] \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1} &= (\mathbf{F}_\alpha^R \mathbf{W} \otimes \mathbf{N}_\chi) \mathbf{F}_\Psi^{-1} \\ &= (\mathbf{F}_\alpha^R \mathbf{W} \otimes \mathbf{H}_\Psi \mathbf{N}_\chi) J_\Psi^{-1} \\ &= \frac{\Lambda_{\mathbf{H}_\Psi}}{J_\Psi} \mathbf{F}_\alpha^R \mathbf{W} \otimes \mathbf{N}_\chi. \end{aligned} \quad (3.30)$$

²⁴When the eigenvalues are also distinct, the hyperbolicity is said to be strong or strict. When the flux matrix is symmetric, the hyperbolicity is symmetric.

²⁵To obtain eigenvectors, the characteristic equation can be solely used instead of the flux Jacobian matrix [2, 199] which depends on the orthogonality condition $\mathcal{R}_\alpha^T \mathcal{L}_\beta = \delta_\alpha^\beta$.

Combining Equations (3.29) and (3.30) gives

$$-c_{\chi,\alpha} \mathbf{F}_\alpha^R = \frac{\Lambda_{\mathbf{H}\Psi}}{J_\Psi} \hat{\mathbf{v}}_\alpha^R \otimes \mathbf{N}_\mathbf{X}. \quad (3.31)$$

By using the characteristic equation of the corresponding system in the Total Lagrangian formulation

$$c_\alpha \mathbf{F}_\alpha^R = -\mathbf{v}_\alpha^R \otimes \mathbf{N}_\mathbf{X}, \quad (3.32)$$

and by pre-multiplying Equation (3.31) by the inverse of \mathbf{F}_α^R , followed by multiplying by and pre-multiplying by $\mathbf{N}_\mathbf{X}$, an ALE characteristic relationship can finally be drawn, as

$$c_{\chi,\alpha} = \frac{\Lambda_{\mathbf{H}\Psi}}{J_\Psi} \left(c_\alpha - \mathbf{W} \cdot \mathbf{N}_\mathbf{X} \right). \quad (3.33)$$

Using Equation (3.32) in conjunction with the definitions introduced in Equation (3.28), Equation (3.26) is written as

$$-c_{\chi,\alpha} \mathbf{p}_{\chi,\alpha}^R = \Lambda_{\mathbf{H}\Psi} \mathbf{D}(\mathbf{P}_\chi \mathbf{N}_\mathbf{X}) [\mathbf{p}_{\chi,\alpha}^R, \mathbf{F}_{\Phi,\alpha}^R]. \quad (3.34)$$

Due to Equation (3.32), the fourth-order constitutive material tensor \mathbf{C} [2, 35, 75, 199] emerges as

$$\frac{\rho_R c_{\chi,\alpha}}{\Lambda_{\mathbf{H}\Psi}} \mathbf{v}_{\chi,\alpha}^R = \frac{1}{c_\alpha} \mathbf{C} : (\mathbf{v}_\alpha \otimes \mathbf{N}_\mathbf{X}) \mathbf{N}_\mathbf{X} - \frac{\rho_R}{J_\Psi} \mathbf{v}_{\chi,\alpha}^R (\mathbf{W} \cdot \mathbf{N}_\mathbf{X}); \quad \mathbf{C} = \frac{\partial^2 \mathcal{E}}{\partial \mathbf{F} \partial \mathbf{F}}. \quad (3.35a,b)$$

Rearranging Equation (3.35a) and multiplying it by a generic virtual velocity $\delta \mathbf{v}$ gives

$$\rho_R \left(\frac{c_{\chi,\alpha}}{\Lambda_{\mathbf{H}\Psi}} + \frac{1}{J_\Psi} \mathbf{W} \cdot \mathbf{N}_\mathbf{X} \right) J_\Psi (\mathbf{v}_\alpha^R \cdot \delta \mathbf{v}) = \frac{1}{c_\alpha} (\delta \mathbf{v} \otimes \mathbf{N}_\mathbf{X}) : \mathbf{C} : (\mathbf{v}_\alpha^R \otimes \mathbf{N}_\mathbf{X}), \quad (3.36)$$

and, by choosing $\delta \mathbf{v}$, using the newly obtained characteristic relationship (3.33), and because Equation (3.36) holds for any $\delta \mathbf{v}$ and in particular $\delta \mathbf{v} = \mathbf{v}_\alpha^R$, the ALE eigenproblem (3.26) can be reduced²⁶ to the notorious generic symmetric eigenvalue problem [2, 22, 75] that reads

$$\rho_R c_\alpha^2 = \mathbf{v}_\alpha^R \cdot (\mathbf{C}_{\mathbf{N}_\mathbf{X} \mathbf{N}_\mathbf{X}} \mathbf{v}_\alpha^R) > 0; \quad [\mathbf{C}_{\mathbf{N}_\mathbf{X} \mathbf{N}_\mathbf{X}}]_{ij} = \mathcal{C}_{iIjJ} N_{\chi,I} N_{\chi,J}, \quad (3.37a,b)$$

where the symmetric second-order tensor $\mathbf{C}_{\mathbf{N}_\mathbf{X} \mathbf{N}_\mathbf{X}}$ is the so-called acoustic tensor. A brief development on the relationship between Lagrangian and ALE eigenvalues is presented in Appendix B.4.1. The six non-zero real eigenvalues are organised in two pressure/volumetric waves (also denoted as p-waves) c_p . For nearly incompressible materials, they take the following values [81]

$$c_{1,2} = \pm c_p; \quad c_p = \sqrt{\frac{\alpha_p \Lambda_p^{-2} + \beta_p + 2\gamma_p}{\rho_R}}, \quad (3.38a,b)$$

and four shear/longitudinal waves c_s

$$c_{3,4} = c_{5,6} = \pm c_s; \quad c_s = \sqrt{\frac{\beta_p}{\rho_R}}. \quad (3.39a,b)$$

²⁶It is worth noting the following algebraic properties of the 4th-order tensor \mathbf{C} :

$$(\delta \mathbf{v} \otimes \mathbf{N}_\mathbf{X}) : \mathbf{C} : (\mathbf{v}_\alpha^R \otimes \mathbf{N}_\mathbf{X}) = \delta \mathbf{v} \cdot (\mathbf{C}_{\mathbf{N}_\mathbf{X} \mathbf{N}_\mathbf{X}} \mathbf{v}_\alpha^R) = (\delta \mathbf{v} \otimes \mathbf{v}_\alpha^R) : \mathbf{C}_{\mathbf{N}_\mathbf{X} \mathbf{N}_\mathbf{X}}.$$

with $\Lambda^{-1} = \|\mathbf{F}^{-T} \mathbf{N}_X\|$ and

$$\alpha_p = \frac{5}{9} \mu J^{-2/3} (\mathbf{F} : \mathbf{F}) + \kappa J^2; \quad \beta_p = \mu J^{-2/3}; \quad \gamma_p = -\frac{2}{3} \mu J^{-2/3}. \quad (3.40a,b,c)$$

The corresponding eigenvectors are

$$\mathbf{u}_{1,2}^R = \begin{bmatrix} \mathbf{n} \\ \pm \rho_R^{-1} c_p^{-1} \mathbf{n} \otimes \mathbf{N}_X \end{bmatrix}, \quad (3.41a)$$

$$\mathbf{u}_{3,4}^R = \begin{bmatrix} \mathbf{t}_1 \\ \pm \rho_R^{-1} c_s^{-1} \mathbf{t}_1 \otimes \mathbf{N}_X \end{bmatrix}, \quad (3.41b)$$

$$\mathbf{u}_{5,6}^R = \begin{bmatrix} \mathbf{t}_2 \\ \pm \rho_R^{-1} c_s^{-1} \mathbf{t}_2 \otimes \mathbf{N}_X \end{bmatrix}, \quad (3.41c)$$

where \mathbf{N}_X is assumed to be a principal direction of deformation, so that $\mathbf{m} = \mathbf{F} \mathbf{N}_X$ and \mathbf{n} are colinear and therefore $\mathbf{t}_i \cdot \mathbf{m} = 0$, with \mathbf{t}_i arbitrary tangential vectors orthogonal to \mathbf{N} . Moreover, the left eigenvectors are given as

$$\mathbf{u}_{1,2}^L = \begin{bmatrix} \mathbf{n} \\ \pm c_p^{-1} \mathbf{C} : (\mathbf{n} \otimes \mathbf{N}_X) \end{bmatrix}, \quad (3.42a)$$

$$\mathbf{u}_{3,4}^L = \begin{bmatrix} \mathbf{t}_1 \\ \pm c_s^{-1} \mathbf{C} : (\mathbf{t}_1 \otimes \mathbf{N}_X) \end{bmatrix}, \quad (3.42b)$$

$$\mathbf{u}_{5,6}^L = \begin{bmatrix} \mathbf{t}_2 \\ \pm c_s^{-1} \mathbf{C} : (\mathbf{t}_2 \otimes \mathbf{N}_X) \end{bmatrix}. \quad (3.42c)$$

3.5 Multiplicative plasticity

Elasto-plasticity is a widely used theory to model a variety of metals, metal powders and other plastic bodies. This section presents the main components of the isotropic von-Mises plasticity used in this work. Then, the necessary considerations to integrate plasticity yielding in the ALE framework will be shown. Permanent deformations occur when energy is locally dissipated, and constitute a plasticity phenomenon. In the following, isothermal plastic materials [75, 217, 219] is considered. Moreover, this work utilises multiplicative plasticity, a common assumption in hyperelastoplastic materials [113, 172, 195] which assumes that the natural deformation gradient \mathbf{F} can be decomposed into an elastic and a plastic components in a multiplicative [2, 58, 75, 219] manner, as

$$\mathbf{F} = \mathbf{F}_e \mathbf{F}_p. \quad (3.43)$$

For a more exhaustive insight on the plasticity theory in solid mechanics, the reader is invited to refer to the work of Oñate et al. [12] and Zienkiewicz et al. [243], as well as [1, 234, 244–249]. Additionally, the works of [113, 154] are of great interest by their use of the ALE approach in the concept of plasticity.

The stress-strain relationship is altered when a body undergoes plastic deformation, usually after significant loading. The present plasticity theory is used to predict these permanent

deformations, and in particular, hardening laws models the new stress-strain curve therefrom after plasticity is yielded, and give an expression for the inherent stress τ_y yielded due to plastic deformation. There are several types of hardening laws. An isotropic isothermal hardening law is generally defined as

$$\tau_y(\bar{\varepsilon}_p) = \tau_y^0 + H \bar{f}(\bar{\varepsilon}_p), \quad (3.44)$$

where $\bar{f}(\bar{\varepsilon}_p)$ is a function of the equivalent plastic strain $\bar{\varepsilon}_p$ that is defined, respectively for linear [250] and nonlinear [251] hardening, as

$$\bar{f}(\bar{\varepsilon}_p) = \begin{cases} \bar{f}_{\text{linear}}(\bar{\varepsilon}_p) = \bar{\varepsilon}_p \\ \bar{f}_{\text{nonlinear}}(\bar{\varepsilon}_p) = \bar{\varepsilon}_p + \frac{\tau_\infty - \tau_y^0}{H} \left(1 - \exp^{-\delta \bar{\varepsilon}_p}\right), \end{cases} \quad (3.45)$$

where the initial yield stress τ_y^0 , the hardening modulus H , the limit stress τ_∞ and the exponent δ are parameters of the material. Note that in Equation (3.45), the nonlinear hardening function $\bar{f}_{\text{nonlinear}}$ can degenerate to the linear hardening function \bar{f}_{linear} when the $\tau_\infty = \tau_y^0$. Beyond the scope of this work, kinematic hardening (where the yield surface shifts in the stress space), and mixed hardening can also be considered. Plastic behaviour is yielded after satisfaction of the loading condition

$$0 < \phi_f(\boldsymbol{\tau}, \bar{\varepsilon}_p) = \bar{\tau}(\boldsymbol{\tau}') - \tau_y(\bar{\varepsilon}_p), \quad (3.46)$$

where the ϕ_f is the yield criterion function, and the non-hydrostatic von Mises yield criterion is defined as

$$\bar{\tau}(\boldsymbol{\tau}') = \sqrt{\frac{3}{2} \boldsymbol{\tau}' : \boldsymbol{\tau}'}. \quad (3.47)$$

In particular, the plasticity yield criterion is determined by a change of sign of the yield function $\phi_f(\boldsymbol{\tau}, \bar{\varepsilon}_p)$ in Equation (3.46), which is numerically carried on by a Newton-Raphson iterative method (described in Section 3.5.3).

3.5.1 Work conjugacy in traditional work-hardening approach

The total rate of work per unit initial volume \dot{w} can be decomposed into elastic and inelastic components²⁷ as [217]

$$\dot{w} = \dot{w}_e + \dot{w}_p. \quad (3.48)$$

Moreover, using the principle of virtual work, it is shown that the Kirchhoff stress $\boldsymbol{\tau}$ and the velocity gradient $\boldsymbol{l} = \dot{\boldsymbol{F}}\boldsymbol{F}^{-1}$ are conjugate with respect to \dot{w} as

$$\dot{w} = \boldsymbol{\tau} : \boldsymbol{l}; \quad \dot{w}_e = \boldsymbol{\tau} : \boldsymbol{l}_e; \quad \dot{w}_p = \boldsymbol{\tau} : \boldsymbol{l}_p = \boldsymbol{\tau} : (\boldsymbol{l} - \boldsymbol{l}_e). \quad (3.49a,b,c)$$

The evolution of $\boldsymbol{\tau}$ is described in closed form as [113]

$$\boldsymbol{\tau} = 2 \frac{\partial w}{\partial \boldsymbol{b}_e} \boldsymbol{b}_e, \quad (3.50)$$

which, combined with Equation (3.49a) relates²⁸ the total rate of work \dot{w} to the time derivative of the left Cauchy Green tensor \boldsymbol{b}_e

$$\boldsymbol{l} = \frac{1}{2} \frac{\partial \boldsymbol{b}_e}{\partial t} \Big|_{\boldsymbol{X}, \boldsymbol{C}_p = \text{const}} \boldsymbol{b}_e^{-1}. \quad (3.51)$$

²⁷The elastic component is sometimes denoted as recoverable, and the inelastic component can be referred to as the plastic component.

²⁸The proof is discussed in [195].

The plastic flow is given by the flow rule, which relates the Lie derivative²⁹ $\mathcal{L}_v(\mathbf{b}_e)$ of \mathbf{b}_e with respect to \mathbf{v} to the flow direction $\frac{\partial \phi_f}{\partial \boldsymbol{\tau}}$ as [113]

$$\mathcal{L}_v(\mathbf{b}_e) = \left. \frac{\partial \mathbf{b}_e}{\partial t} \right|_{\mathbf{X}} - (\nabla_{\mathbf{X}} \mathbf{v}) \mathbf{b}_e - \mathbf{b}_e (\nabla_{\mathbf{X}} \mathbf{v})^T = -2\dot{\gamma} \frac{\partial \phi_f(\boldsymbol{\tau}, \bar{\varepsilon}_p)}{\partial \boldsymbol{\tau}} \mathbf{b}_e. \quad (3.52)$$

Moreover, the constitutive behaviour of isotropic inelastic materials in the spatial domain can be expressed by means of the left \mathbf{b}_e and right \mathbf{C}_p Cauchy-Green tensors as

$$\begin{aligned} \mathbf{b}_e &= \mathbf{F}_e \mathbf{F}_e^T \\ &= \mathbf{F} \mathbf{F}_p^{-1} \mathbf{F}_p^{-T} \mathbf{F}^T \\ &= \mathbf{F} \mathbf{C}_p^{-1} \mathbf{F}^T, \end{aligned} \quad (3.53)$$

or

$$\mathbf{C}_p^{-1} = \mathbf{F}^{-1} \mathbf{b}_e \mathbf{F}^{-T}. \quad (3.54)$$

The rate of the right Cauchy-Green stress tensor is characterised by the flow rule, which will be shown in Equation (3.58a). Solving that rate is the purpose of the plasticity Algorithm 1 presented in the next section.

More specifically, with regards to Equations (3.50) and (3.51), the Hencky constitutive model will be conveniently expressed in terms of the invariants of \mathbf{b}_e as seen in Equation (3.19). It follows that the velocity gradient, that is related to the elastic finger tensor, can also be related to the flow direction as [113, 217]

$$\mathbf{l}_p = -\frac{1}{2} \left. \frac{d\mathbf{b}_e}{dt} \right|_{\mathbf{F}=\text{const}} \mathbf{b}_e^{-1} = \dot{\gamma} \frac{\partial \phi_f(\boldsymbol{\tau}, \bar{\varepsilon}_p)}{\partial \boldsymbol{\tau}}. \quad (3.55)$$

In the above equation, the rate of \mathbf{b}_e with constant deformation gradient \mathbf{F} is considered. The plastic strain rate $\dot{\bar{\varepsilon}}_p$ is conjugate to the von Mises equivalent stress $\bar{\boldsymbol{\tau}}(\boldsymbol{\tau}')$ with respect to the plastic work rate w_p , which gives

$$\dot{w}_p = \bar{\boldsymbol{\tau}} \dot{\bar{\varepsilon}}_p; \quad \dot{\bar{\varepsilon}}_p = \dot{\gamma}, \quad (3.56a,b)$$

as a result, the plastic dissipation w_p can be computed via the increment Δw_p , using the equivalent von Mises stress (3.47) based on the deviatoric Kirchhoff stress, defined as

$$\Delta w_p = \bar{\boldsymbol{\tau}}(\boldsymbol{\tau}') \Delta \gamma; \quad \boldsymbol{\tau}' = \boldsymbol{\tau} - p \mathbf{I}. \quad (3.57a,b)$$

The internal plastic variables will be updated via an implicit Euler time integrator as discussed in Chapter 6. Their time integration relies on a *return mapping* class of procedure, which is featured in the plasticity algorithm 1. This algorithm makes use of a) an elastic prediction step (lines 2-4), b) a check for plasticity loading via the threshold condition at line 5, and c) a plasticity correction based on the so-called *radial return* method [95, 100, 195, 239, 244] at lines 6-10. The terms accompanied with their equation (at lines 1, 5, 7 and 19) are model-dependent.

A situation in which plasticity develops is characterised by an evolution of the plastic internal variables, consisting in a strictly positive infinitesimal plastic increment, and non trivial plastic stresses. The von-Mises plasticity theory relies on the evolution of two plastic internal variables: the plastic multiplier $\Delta \bar{\varepsilon} - p = \Delta \gamma$ and the inverse of the plastic right

²⁹This is possible due to the contravariant nature of the left Cauchy Green tensor \mathbf{b}_e [252–254]. A brief introduction to the Lie derivative for covariant and contravariant tensors is shown in Appendix B.3.3.

Cauchy-Green stress \mathbf{C}_p^{-1} . The conservation of \mathbf{C}_p^{-1} follows from Equation (3.55) and the evolution of $\Delta\bar{\varepsilon}$ is obtained via a Backward-Euler integration of the flow rule (3.56b), as

$$\left. \frac{\partial \mathbf{C}_p^{-1}}{\partial t} \right|_{\mathbf{X}} = -2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T}, \quad (3.58a)$$

$$\left. \frac{\partial \bar{\varepsilon}_p}{\partial t} \right|_{\mathbf{X}} = \dot{\gamma}. \quad (3.58b)$$

Algorithm 1: Hencky-based von Mises plasticity with isothermal hardening.

Data: $\mathbf{F}_{n+1}, J_{n+1}, \mathbf{C}_{p,n}^{-1}, \bar{\varepsilon}_{p,n}, w_{p,n}$

- 1 Pressure Correction p_{n+1} (Equation (3.11b));
- 2 Elastic finger tensor $\mathbf{b}_{e,n+1}^{trial} = \mathbf{F}_{n+1} \mathbf{C}_{p,n}^{-1} \mathbf{F}_{n+1}^T$;
- 3 Principal components decomposition $\mathbf{b}_{e,n+1}^{trial} = \sum_{i=1}^3 (\lambda_{e,i}^{trial})^2 \mathbf{n}_i^{trial} \otimes \mathbf{n}_i^{trial}$;
- 4 Trial deviatoric Kirchhoff $\boldsymbol{\tau}_{n+1}^{trial} = 2\mu \ln \sum_{i=1}^3 \left(J^{-1/3} \lambda_{e,i}^{trial} \mathbf{n}_i^{trial} \otimes \mathbf{n}_i^{trial} \right)$;
- 5 **if** $\phi_f(\boldsymbol{\tau}_{n+1}^{trial}, \bar{\varepsilon}_{p,n}) > 0$ (Loading Condition (3.46))
 - 6 Derivative of yield function $\boldsymbol{\nu}_{n+1} = \frac{\boldsymbol{\tau}_{n+1}^{trial}}{\sqrt{2/3 \boldsymbol{\tau}_{n+1}^{trial} : \boldsymbol{\tau}_{n+1}^{trial}}}$;
 - 7 Plastic Multiplier $\Delta\gamma_{n+1}$ (linear or nonlinear (3.45) hardening rule);
 - 8 Logarithmic Elastic stretches $\lambda_{e,i,n+1} = \exp \left(\ln(\lambda_{e,i}^{trial}) - \Delta\gamma \boldsymbol{\nu}_{i,n+1} \right)$;
 - 9 Deviatoric Kirchhoff $\boldsymbol{\tau}'_{i,n+1} = \boldsymbol{\tau}_{i,n+1}^{trial} - 2\mu \Delta\gamma_{n+1} \boldsymbol{\nu}_{i,n+1}$;
 - 10 Reconstruction of elastic finger $\mathbf{b}_{e,n+1} = \sum_{i=1}^3 \lambda_{e,n+1}^2 \mathbf{n}_i^{trial} \otimes \mathbf{n}_i^{trial}$;
- 11 **else**
 - 12 $\boldsymbol{\tau}'_{n+1} = \boldsymbol{\tau}_{n+1}^{trial}$;
 - 13 $\mathbf{b}_{e,n+1} = \mathbf{b}_{e,n+1}^{trial}$;
- 14 Plastic strain update $\bar{\varepsilon}_{p,n+1} = \bar{\varepsilon}_{p,n} + \Delta\gamma_{n+1}$;
- 15 Inverse of plastic right Cauchy $\mathbf{C}_{p,n+1}^{-1} = \mathbf{F}_{n+1}^{-1} \mathbf{b}_{e,n+1} \mathbf{F}_{n+1}^{-T}$;
- 16 Kirchhoff update $\boldsymbol{\tau}_{n+1} = \sum_{i=1}^3 \left(\boldsymbol{\tau}'_{ii,n+1} + J_{n+1} p_{n+1} \right) \mathbf{n}_i^{trial} \otimes \mathbf{n}_i^{trial}$;
- 17 Piola Kirchhoff update $\mathbf{P}_{n+1} = \boldsymbol{\tau}_{n+1} \mathbf{F}_{n+1}^{-T}$;
- 18 von-Mises equivalent stress $\bar{\tau}_{n+1} = \bar{\tau}(\boldsymbol{\tau}'_{n+1}) = \sqrt{\frac{3}{2} \boldsymbol{\tau}'_{n+1} : \boldsymbol{\tau}'_{n+1}}$;
- 19 Plastic dissipation $w_{p,n+1} = w_{p,n} + \bar{\tau}_{n+1} \Delta\gamma_{n+1}$ (Equation (3.57b)).

Result: $\mathbf{P}_{n+1}, \mathbf{C}_{p,n+1}^{-1}, \bar{\varepsilon}_{p,n+1}, \bar{\tau}_{n+1}, w_{p,n+1}$

3.5.2 ALE framework for plasticity

By using the non-conservative ALE transformations established in Equations (2.44) and (2.45), the governing Equations (3.58a) and (3.58b) in Lagrangian form can be expressed on a

referential domain as

$$\left. \frac{\partial \mathbf{C}_p^{-1}}{\partial t} \right|_{\mathbf{x}} - (\nabla_{\mathbf{x}} \mathbf{C}_p^{-1}) (\mathbf{F}_{\Psi}^{-1} \mathbf{W}) = -2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T}, \quad (3.59a)$$

$$\left. \frac{\partial \bar{\varepsilon}_p}{\partial t} \right|_{\mathbf{x}} - (\nabla_{\mathbf{x}} \bar{\varepsilon}_p) \cdot (\mathbf{F}_{\Psi}^{-1} \mathbf{W}) = \dot{\gamma}, \quad (3.59b)$$

These equations can be incorporated to the ALE mixed formulation by considering their respective right hand side terms as source terms solved in an implicit manner. This will be discussed in Section 6.7 dedicated to the time integration.

3.5.3 Newton-Raphson iterative procedure

Consider Equation (3.46) in Section 3.5, that can be rewritten as

$$\phi_f(\boldsymbol{\tau}', \bar{\varepsilon}_p + \Delta \bar{\varepsilon}_p) = \bar{\tau}(\boldsymbol{\tau}') - \tau_y(\bar{\varepsilon}_p + \Delta \bar{\varepsilon}_p) = 0. \quad (3.60)$$

In order to update the plastic strain to the new value $\bar{\varepsilon}_p^{n+1}$, a Newton-Raphson procedure is utilised. It is described in Algorithm 2 and serves the purpose of finding the root $\Delta \bar{\varepsilon}_p^{n+1}$ of the following

$$\phi_f(\boldsymbol{\tau}', \bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^{n+1}) = 0; \quad \bar{\varepsilon}_p^{n+1} = \bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^{n+1}. \quad (3.61a,b)$$

With the use of some algebra, the von Mises stress $\bar{\tau}$ featured in Equation (3.60) can be re-expressed as

$$\bar{\tau}(\boldsymbol{\tau}') = \bar{\tau}(\boldsymbol{\tau}'^{\text{trial}}) - 3\mu \Delta \gamma, \quad (3.62)$$

The Newton-Raphson iterative process is initialised using a linearised hardening function (see Equation (3.45)) $\bar{f}_{\text{linear}}(\bar{\varepsilon}_p) = \bar{\varepsilon}_p$, and using the fact that the plastic multiplier $\Delta \gamma$ equates to the work-hardening variable $\bar{\varepsilon}_p$,

$$\bar{\tau}(\boldsymbol{\tau}') - [\tau_y^0 + H(\bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^0)] = 0 \quad (3.63)$$

so that Equation (3.60) can be rewritten as

$$\Delta \bar{\varepsilon}_p^0 = \frac{\phi_f(\boldsymbol{\tau}', \text{trial}, \bar{\varepsilon}_p^n)}{3\mu + H}, \quad (3.64)$$

For every other iteration, the algorithm's objective is to solve

$$\Delta \bar{\varepsilon}_p^{k+1} = \Delta \bar{\varepsilon}_p^k - \frac{\text{Res}(\Delta \bar{\varepsilon}_p^k)}{\text{Res}'(\Delta \bar{\varepsilon}_p^k)} \quad (3.65)$$

with

$$T(\Delta \bar{\varepsilon}_p^k) = 3\mu \Delta \bar{\varepsilon}_p^k + \tau_y^0 + H(\bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^k) + (\tau_\infty - \tau_y^0) \left[1 - \exp^{-\delta(\bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^k)} \right] \quad (3.66a)$$

$$F = \phi_f^{\text{trial}} + \bar{\tau}_y^0 + H \bar{\varepsilon}_p + (\tau_\infty - \tau_y^0) \left[1 - e^{-\delta \bar{\varepsilon}_p} \right] \quad (3.66b)$$

$$\text{Res}(\Delta \bar{\varepsilon}_p^k) = T(\Delta \bar{\varepsilon}_p^k) - F \quad (3.66c)$$

$$\text{Res}'(\Delta \bar{\varepsilon}_p^k) = 3\mu + H + \delta (\tau_\infty - \tau_y^0) \left[1 - \exp^{-\delta(\bar{\varepsilon}_p^n + \Delta \bar{\varepsilon}_p^k)} \right] \quad (3.66d)$$

As featured in Algorithm 2, it is common to use an error tolerance TOL parameter, as well as a maximum number of iterations.

Algorithm 2: Newton-Raphson procedure.

Data: $\phi_f(\boldsymbol{\tau}^{\text{trial}}, \bar{\boldsymbol{\varepsilon}}_p^n), \bar{\boldsymbol{\varepsilon}}_p^n$

- 1 Compute F (Equation (3.66b));
- 2 Compute $\Delta \bar{\boldsymbol{\varepsilon}}_p^0$ (Equation (3.63));
- 3 **while** $\text{Err} > \text{TOL}$ **do**
- 4 Compute $\mathbf{R} = \frac{\text{Res}}{\text{Res}'} (Equation (3.66d));$
- 5 Update $\Delta \bar{\boldsymbol{\varepsilon}}_p^{k+1} = \Delta \bar{\boldsymbol{\varepsilon}}_p^k - \mathbf{R};$
- 6 Compute $\text{Err} = \|\mathbf{R}\|;$
- 7 Set $\Delta \bar{\boldsymbol{\varepsilon}}_p^{n+1} = \Delta \bar{\boldsymbol{\varepsilon}}_p^{k+1}$

Result: $\Delta \bar{\boldsymbol{\varepsilon}}_p^{n+1}$

3.6 Conclusion

In this chapter, the concept of polyconvexity was introduced in Section 3.2 as a constitutive restriction implying the Legendre-Hadamard condition (existence of travelling waves in the material in the vicinity of a stationary point). It is of paramount importance because it is strongly related to material stability and to the development of areas where ellipticity, and therefore well-posedness of the governing equations, no longer hold. Large strain elastic and inelastic isotropic materials are described by means of a hyperelastic model, and characterised by a stored energy functional which depends on the deformation gradient of the natural mapping. In Section 3.3, nearly incompressible nonlinear constitutive models are presented for modelling deformations with very little to no change in the volume of a body. They are characterised by a Poisson's ratio close or equal to 0.5. Nearly incompressible models offer the possibility to decompose the energy potential and stresses within a body into distortional and volumetric components. In Section 3.4, an eigenvalue analysis is performed on the mixed system of the ALE formulation to demonstrate hyperbolicity of the problem. It is an important property that is necessary before any discretisation method is applied. Eventually, the general ALE formulation presented in Chapter 2 has been extended to plasticity in Section 3.5 with the integration of two additional conservation laws on internal plastic variables. In the next Chapter, the ALE mixed system will be further enhanced with the introduction of thermal considerations.

CHAPTER 4

EXTENSION TO THERMAL INELASTICITY

“Rien ne se crée, ni dans les opérations de l’art, ni dans celles de la nature, et l’on peut poser en principe que, dans toute opération, il y a une égale quantité de matière avant et après l’opération; que la qualité et la quantité des principes est la même, et qu’il n’y a que des changements, des modifications.”

— ANTOINE LAVOISIER, *Traité élémentaire de chimie*, 1789

4.1 Introduction

Materials subjected to high temperatures and high loads (e.g. in aerospace, automotive, welding or casting processes), or materials sensitive to temperature-based fluctuations can experience thermal-based deformations, such as thermal expansion. Thermal expansion can cause internal stress, generated within the material by temperature gradients, which may eventually lead to significant additional large deformations. Moreover, plastic properties of the material can also be affected by internal temperature fluctuations. (e.g. temperature annealing). This section extends the present ALE formulation to thermal considerations with the introduction of the laws of thermodynamics. The conservation of energy is presented, and with the use of Calorimetry relationships, the link between the entropy, the temperature and the energy is outlined. Then, the Mie-Grüneisen equation of state is introduced and the implications on energy and stress components are discussed, as well as additional requirements for polyconvexity. Eventually, the considerations for visco-plasticity, including a thermally-coupled volumetric energy potential and a Johnson-Cook hardening law, are presented.

4.2 First Law of thermodynamics

In an irreversible process, a thermal coupling is considered where the mixed system (2.75) must include a conservation law of a new thermal variable. This work is dedicated to solving the conservation of the total energy of the system, which is represented by the first law of thermodynamics [8, 30, 241, 255]. The global form of the conservation of total energy $E(\mathbf{X}, t)$ using a Lagrangian framework is given by

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega} E dV - \int_{\partial\Omega} \mathbf{P}^T \mathbf{v} \cdot \mathbf{N} dA = - \int_{\partial\Omega} \mathbf{Q} \cdot \mathbf{N} dA + \int_{\Omega} (s + \mathbf{f}_R \cdot \mathbf{v}) dV, \quad (4.1)$$

with body forces \mathbf{f}_R , external heat source s , and the material heat flux³⁰ \mathbf{Q} defined in terms of thermal conductivity h and temperature θ as

$$\mathbf{Q} = -\mathbf{K} \nabla_{\mathbf{x}} \theta; \quad \mathbf{K} = J^{-1} \mathbf{H}^T \mathbf{k} \mathbf{H}; \quad \mathbf{k} = h \mathbf{I}, \quad (4.2a,b,c)$$

where \mathbf{k} is the isotropic thermal conductivity tensor, based on the scalar conductivity h . Locally, the conservation of the total energy is expressed, in Total Lagrangian form, as

$$\left. \frac{\partial E}{\partial t} \right|_{\mathbf{X}} - \nabla_{\mathbf{x}} \cdot (\mathbf{P}^T \mathbf{v}) + \nabla_{\mathbf{x}} \cdot \mathbf{Q} = s + \mathbf{f}_R \cdot \mathbf{v}. \quad (4.3)$$

The first law of thermodynamics, in global form, can be expressed on the referential domain by using ALE transformations on Equation (4.1), which yields

$$\begin{aligned} \left. \frac{\partial}{\partial t} \right|_{\mathbf{X}} \int_{\Omega_{\mathbf{X}}} E_{\mathbf{X}} dV_{\mathbf{X}} - \int_{\partial\Omega_{\mathbf{X}}} [(\mathbf{P} \mathbf{H}_{\Psi})^T \mathbf{v} + E \mathbf{H}_{\Psi}^T \mathbf{W} - \mathbf{Q}_{\mathbf{X}}] \cdot \mathbf{N}_{\mathbf{X}} dA_{\mathbf{X}} \\ = \int_{\Omega_{\mathbf{X}}} (\mathbf{f} \cdot \mathbf{v} + s_{\mathbf{X}}) dV_{\mathbf{X}}, \end{aligned} \quad (4.4)$$

where $E_{\mathbf{X}} = J_{\Psi} E$ is the referential energy, $\mathbf{f} = J_{\Psi} \mathbf{f}_R$ is the referential body forces

³⁰The expression of the heat flux is obtained via Fourier's law [22, 23].

$s_{\mathcal{X}} = J_{\Psi} s$ is the referential heat source and $\mathbf{Q}_{\mathcal{X}}$ is the referential heat flux obtained as

$$\begin{aligned} \int_{\partial\Omega_{\mathcal{X}}} \mathbf{Q} \cdot \mathbf{N} \, dA &= \int_{\partial\Omega_{\mathcal{X}}} (\mathbf{H}_{\Psi}^T \mathbf{Q}) \cdot \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}} \\ &= - \int_{\partial\Omega_{\mathcal{X}}} (\mathbf{H}_{\Psi}^T \boldsymbol{\kappa} \nabla_{\mathcal{X}} \theta) \cdot \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}} \\ &= - \int_{\partial\Omega_{\mathcal{X}}} (\mathbf{H}_{\Psi}^T \boldsymbol{\kappa} \mathbf{F}_{\Psi}^{-T} \nabla_{\mathcal{X}} \theta) \cdot \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}} \\ &= \int_{\partial\Omega_{\mathcal{X}}} \mathbf{Q}_{\mathcal{X}} \cdot \mathbf{N}_{\mathcal{X}} \, dA_{\mathcal{X}}, \end{aligned} \quad (4.5)$$

with $\nabla_{\mathcal{X}} = \mathbf{F}_{\Psi}^{-T} \nabla_{\mathcal{X}} \theta$, and consequently

$$\mathbf{Q}_{\mathcal{X}} = -\boldsymbol{\kappa}_{\mathcal{X}} \nabla_{\mathcal{X}} \theta; \quad \boldsymbol{\kappa}_{\mathcal{X}} = J_{\Psi}^{-1} \mathbf{H}_{\Psi}^T \boldsymbol{\kappa} \mathbf{H}_{\Psi} = h J_{\Phi} \mathbf{C}_{\Phi}^{-1}, \quad (4.6a,b)$$

where $\mathbf{C}_{\Phi} = \mathbf{F}_{\Phi}^T \mathbf{F}_{\Phi}$ is the right Cauchy-Green tensor related to the spatial deformation gradient. The local form of Equation (4.4) is

$$\left. \frac{\partial E_{\mathcal{X}}}{\partial t} \right|_{\mathcal{X}} - \nabla_{\mathcal{X}} \cdot [(\mathbf{P} \mathbf{H}_{\Psi})^T \mathbf{v} + E \mathbf{H}_{\Psi}^T \mathbf{W} - \mathbf{Q}_{\mathcal{X}}] = \mathbf{f} \cdot \mathbf{v} + s_{\mathcal{X}}, \quad (4.7)$$

together with the jump conditions

$$c_{\mathcal{X}} \llbracket E_{\mathcal{X}} \rrbracket = (\mathbf{H}_{\Psi}^T \llbracket \mathbf{P}^T \mathbf{v} + E \mathbf{W} \rrbracket - \llbracket \mathbf{Q}_{\mathcal{X}} \rrbracket) \cdot \mathbf{N}_{\mathcal{X}}. \quad (4.8)$$

4.3 Thermo-elasticity

The strain energy, originally expressed for an isothermal hyperelastic model in Equation (3.1), is postulated here as a quantity that depends on thermal variables, and potentially additional state variables that are represented by $\boldsymbol{\alpha}$. This is denoted as

$$\mathcal{E}(\nabla_{\mathcal{X}} \boldsymbol{\varphi}, \eta, \boldsymbol{\alpha}) = \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}); \quad \boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}} = \{\boldsymbol{\mathcal{X}}, \eta, \boldsymbol{\alpha}\}, \quad (4.9a,b)$$

As the first Piola Kirchhoff tensor is work conjugate with the deformation gradient, the time rate of the energy potential can be analysed using the mechanical conjugates stresses³¹ $\boldsymbol{\Sigma}_{\mathbf{F}}, \boldsymbol{\Sigma}_{\mathbf{H}}, \boldsymbol{\Sigma}_{\mathbf{J}}$ defined as

$$\boldsymbol{\Sigma}_{\mathbf{F}}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) = \frac{\partial \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}})}{\partial \mathbf{F}}; \quad \boldsymbol{\Sigma}_{\mathbf{H}}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) = \frac{\partial \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}})}{\partial \mathbf{H}}; \quad \boldsymbol{\Sigma}_{\mathbf{J}}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) = \frac{\partial \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}})}{\partial \mathbf{J}}. \quad (4.10a,b,c)$$

Moreover, the temperature is introduced by means of an energy dual conjugacy established between the entropy density as

$$\theta(\mathbf{X}, t) = \frac{\partial \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}})}{\partial \eta(\mathbf{X}, t)} = \Theta(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}), \quad (4.11)$$

where θ and Θ both denote the temperature magnitude expressed with different variables. By taking advantage of Calorimetry relationships³², the specific heat³³ at constant volume $c_v > 0$ is a coefficient that can be defined as

$$c_v = \frac{\partial \tilde{\mathcal{E}}(\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}})}{\partial \theta(\mathbf{X}, t)}, \quad (4.12)$$

³¹Conjugate stresses are defined in Equation (3.5) for an isothermal framework, and the corresponding analysis of the time rate of the energy potential is shown in Equation (3.6).

³²Extensive development of the results obtained with the Calorimetry relationships can be found in [2, 8, 10, 22, 23, 37, 249].

³³The specific heat can also be expressed per unit undeformed volume c_v can be expressed in terms of the material density ρ_R and the specific heat per unit mass C_v as $c_v = \rho_R C_v$.

where the internal energy density $\tilde{\mathcal{E}}$ and the entropy density $\tilde{\eta}$ are expressed as

$$\tilde{\mathcal{E}}(\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}}) = \mathcal{E}(\boldsymbol{\mathcal{X}}, \tilde{\eta}(\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}}), \boldsymbol{\alpha}); \quad \tilde{\eta}(\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}}) = \eta(\boldsymbol{\mathcal{X}}, t), \quad (4.13a,b)$$

and where the system of coordinates $\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}} = \{\boldsymbol{\mathcal{X}}, \theta, \boldsymbol{\alpha}\}$ now involves the temperature θ . Equation (4.12) can be integrated, with respect to the temperature and between a reference temperature θ_R and a current temperature θ (see the Appendix Section B.3.1 for more detailed development), to yield a relationship between the entropy at current temperature $\tilde{\eta}$ and the entropy $\tilde{\eta}_R$ at reference temperature, stating

$$\tilde{\eta}(\boldsymbol{\mathcal{X}}_{\theta\boldsymbol{\alpha}}) = \tilde{\eta}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha}) + c_v \ln \left(\frac{\theta(\boldsymbol{\mathcal{X}}, t)}{\theta_R} \right); \quad \tilde{\eta}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha}) = \tilde{\eta}(\boldsymbol{\mathcal{X}}, \theta = \theta_R, \boldsymbol{\alpha}). \quad (4.14a,b)$$

A direct consequence of Equation (4.14a) is to outline an expression for the current temperature magnitude, involving the difference between the entropy density η and the entropy at constant temperature $\tilde{\eta}_R$, as

$$\Theta(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) = \theta_R \exp \frac{\eta - \tilde{\eta}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha})}{c_v}. \quad (4.15)$$

In a similar fashion, Equation (4.11) is integrated between temperatures θ_R and $\Theta(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha})$ [22], and after a development detailed in Appendix Section B.3.1, a relationship for the internal energy \mathcal{E} expressed with respect to the triplet $\boldsymbol{\mathcal{X}}$, the entropy density $\eta(\boldsymbol{\mathcal{X}}, t)$, and the internal variable $\boldsymbol{\alpha}$, is obtained as

$$\mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) = \tilde{\mathcal{E}}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha}) + c_v \theta_R \left(\exp \frac{\tilde{\eta}(\boldsymbol{\mathcal{X}}, \theta, \boldsymbol{\alpha}) - \tilde{\eta}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha})}{c_v} - 1 \right) \quad (4.16a)$$

$$= \tilde{\mathcal{E}}_R(\boldsymbol{\mathcal{X}}, \boldsymbol{\alpha}) + c_v (\Theta(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}) - \theta_R), \quad (4.16b)$$

where the change of entropy in Equation (4.14a) was used to obtain the explicit expression of thermal contributions in the internal energy potential [22]. In Equation (4.16b), the internal energy $\tilde{\mathcal{E}}_R(\boldsymbol{\mathcal{X}})$ represents the energy caused by the deformation after the temperature θ returned to the reference value θ_R . In this thermo-mechanical framework, the fact that the Cauchy problem is well-posed [22, 23] results that both $\tilde{\mathcal{E}}_R(\boldsymbol{\mathcal{X}})$ and $-\tilde{\eta}_R(\boldsymbol{\mathcal{X}})$ are polyconvex³⁴. Using the chain rule, the time rate of the energy is expressed as [21, 22]

$$\begin{aligned} \left. \frac{\partial \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}})}{\partial t} \right|_{\boldsymbol{\mathcal{X}}} &= \frac{\partial \mathcal{E}}{\partial \boldsymbol{F}} : \frac{\partial \boldsymbol{F}}{\partial t} + \frac{\partial \mathcal{E}}{\partial \boldsymbol{H}} : \frac{\partial \boldsymbol{H}}{\partial t} + \frac{\partial \mathcal{E}}{\partial J} \frac{\partial J}{\partial t} + \frac{\partial \mathcal{E}}{\partial \eta} \frac{\partial \eta}{\partial t} + \frac{\partial \mathcal{E}}{\partial \boldsymbol{\alpha}} : \frac{\partial \boldsymbol{\alpha}}{\partial t} \\ &= \boldsymbol{\Sigma}_F : \nabla_{\boldsymbol{X}} \boldsymbol{v} + \boldsymbol{\Sigma}_H : (\boldsymbol{F} \times \nabla_{\boldsymbol{X}} \boldsymbol{v}) + \Sigma_J \boldsymbol{H} : \nabla_{\boldsymbol{X}} \boldsymbol{v} + \theta \left. \frac{\partial \eta}{\partial t} \right|_{\boldsymbol{\mathcal{X}}} + \frac{\partial \mathcal{E}}{\partial \boldsymbol{\alpha}} : \frac{\partial \boldsymbol{\alpha}}{\partial t} \\ &= \underbrace{[\boldsymbol{\Sigma}_F + \boldsymbol{\Sigma}_H \times \boldsymbol{F} + \Sigma_J \boldsymbol{H}]}_{\boldsymbol{P}} : \dot{\boldsymbol{F}} + \theta \left. \frac{\partial \eta}{\partial t} \right|_{\boldsymbol{\mathcal{X}}} + \frac{\partial \mathcal{E}}{\partial \boldsymbol{\alpha}} : \frac{\partial \boldsymbol{\alpha}}{\partial t}. \end{aligned} \quad (4.17)$$

Since the thermal coupling is exclusively featured in the volumetric stress contributions, the first term of Equation (4.17) corresponds to a mechanical work $\boldsymbol{P} : \dot{\boldsymbol{F}}$.

4.3.1 The Ballistic energy

The total energy E of a system is obtained as the sum of the kinetic energy, the internal energy \mathcal{E} , the force resulting of an external work, to which is subtracted energy related to thermal dissipation. Considering a body under an adiabatic process and not subjected to external work, the expression of the total energy can be stated as

$$E(X, t) = K + \mathcal{E}(\boldsymbol{\mathcal{X}}_{\eta\boldsymbol{\alpha}}), \quad (4.18)$$

³⁴That is, convex with respect to their geometric variables $\boldsymbol{\mathcal{X}} = \{\boldsymbol{F}, \boldsymbol{H}, J\}$.

where the kinetic energy is defined in terms of the linear momentum as

$$K = \frac{\mathbf{p} \cdot \mathbf{p}}{2\rho_R}. \quad (4.19)$$

Substituting Equation (4.16b) in the definition of the total energy (4.18) gives

$$\theta(\mathbf{X}, t) = \theta_R + \frac{1}{c_v} \left[E(\mathbf{X}, t) - K - \tilde{\mathcal{E}}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) \right]. \quad (4.20)$$

Furthermore and as in CFD [256], the so-called *Ballistic* free energy \mathcal{B} [21, 257] (also denoted as the Lyapunov function of the thermo-mechanical process), defined as a generalised convex entropy function³⁵ [23] and whose expression is

$$\mathcal{B}(\mathbf{X}, t) = \tilde{\mathcal{B}}(\mathbf{p}, \boldsymbol{\chi}_\eta) = K(\mathbf{p}) + \mathcal{E}(\boldsymbol{\chi}_\eta) - \eta\theta_R, \quad (4.21)$$

where $\tilde{\mathcal{B}}$ is another expression of the *Ballistic* energy with the same magnitude. It can be regarded as a minimum stable quantity constituted of the total energy (the kinetic energy K and the internal energy density $\tilde{\mathcal{E}}_R$) and thermal contributions (or thermal heat component) $\theta_R\eta$ per unit undeformed volume. In Appendix (C), it is shown how the *Ballistic* energy is utilised to obtain an expression of the second law of thermodynamics.

4.3.2 Second Law of thermodynamics

The total energy is related to the other thermal variables (entropy, temperature) with closed-form relationships. The irreversibility of the process can be characterised by the production of entropy, according to the second law of thermodynamics. This law, sometimes referred to as Clausius-Duhem inequality, states that the rate of the total entropy of a system must be greater than or equal to the transfer of entropy into the system. For a plastic internal variable $\boldsymbol{\alpha}$, the rate of physical dissipation D_{int} introduced by the physical model due to plasticity is defined as [10, 21]

$$\dot{D}_{\text{int}} = \left. \frac{\partial w_p}{\partial t} \right|_{\mathbf{X}} = - \frac{\partial \mathcal{E}(\boldsymbol{\chi}_{\eta\boldsymbol{\alpha}})}{\partial \boldsymbol{\alpha}} : \left. \frac{\partial \boldsymbol{\alpha}}{\partial t} \right|_{\mathbf{X}}, \quad (4.22)$$

where w_p is defined in Equation (3.57). Under the simple assumption that heat flows from hot to cold regions of a continuum, it is stated by convention that

$$D_{\text{int}} - \frac{1}{\theta} \mathbf{Q} \cdot \nabla_{\mathbf{x}} \theta \geq 0; \quad D_{\text{int}} = \mathbf{P} : \dot{\mathbf{F}} + \theta \dot{\eta} - \dot{\mathcal{E}}. \quad (4.23\text{a,b})$$

where D_{int} is denoted as the internal dissipation. Further independent conditions can also be stated as

$$D_{\text{int}} \geq 0; \quad \frac{1}{\theta} \mathbf{Q} \cdot \nabla_{\mathbf{x}} \theta \leq 0. \quad (4.24\text{a,b})$$

Additional contribution to the dissipation may be taken into account by further expressing the energy with other state variables $\boldsymbol{\alpha}$, for which the Equation (4.23b) becomes

$$\begin{aligned} D_{\text{int}} &= \mathbf{P} : \dot{\mathbf{F}} + \theta \dot{\eta} - \dot{\mathcal{E}}(\boldsymbol{\chi}_{\eta\boldsymbol{\alpha}}) \\ &= \mathbf{P} : \dot{\mathbf{F}} + \theta \dot{\eta} - [\boldsymbol{\Sigma}_{\mathbf{F}} + \boldsymbol{\Sigma}_{\mathbf{H}} \times \mathbf{F} + \boldsymbol{\Sigma}_{\mathbf{J}} \mathbf{H}] : \dot{\mathbf{F}} - \theta \dot{\eta} - \frac{\partial \mathcal{E}(\boldsymbol{\chi}_{\eta\boldsymbol{\alpha}})}{\partial \boldsymbol{\alpha}} : \dot{\boldsymbol{\alpha}} \\ &= - \frac{\partial \mathcal{E}(\boldsymbol{\chi}_{\eta\boldsymbol{\alpha}})}{\partial \boldsymbol{\alpha}} : \dot{\boldsymbol{\alpha}}. \end{aligned} \quad (4.25)$$

For instance, in the context of thermoplasticity, the internal energy can also be expressed in terms a plastic stress with $\boldsymbol{\alpha} = \mathbf{C}_p^{-1}$. Note that the term $\mathbf{P} - \frac{\partial \mathcal{E}}{\partial \mathbf{F}}$ in Equation (4.25) vanishes in a thermoelastic regime, where no internal variable is considered.

³⁵Hence enabling the use of the total energy density as the thermal unknown.

4.3.3 Conservation of internal energy and entropy

With the definition (4.18) of the total energy, the conservation law of E in Equation (4.7) can be subtracted with the linear momentum balance principle that is multiplied by \mathbf{v} to yield

$$\left. \frac{\partial \mathcal{E}(\mathcal{X}_{\eta\alpha})}{\partial t} \right|_{\mathbf{x}} + \nabla_{\mathbf{x}} \cdot \mathbf{Q} = \mathbf{P} : \nabla_{\mathbf{x}} \mathbf{v} + s. \quad (4.26)$$

The above equation corresponds to the conservation law of the internal energy. As shown in [21], the first law of thermodynamics can also be expressed in terms of the entropy per unit undeformed volume $\eta(\mathbf{X}, t)$, by combining Equation (4.17) and (4.26), as

$$\theta(\mathbf{X}, t) \left. \frac{\partial \eta(\mathbf{X}, t)}{\partial t} \right|_{\mathbf{x}} + \nabla_{\mathbf{x}} \cdot \mathbf{Q} = s + \dot{D}_{\text{int}}, \quad (4.27)$$

where the rate of physical dissipation \dot{D}_{int} is defined in Equation (4.22). After dividing Equation (4.27) by θ and applying a conservative ALE transformation, the conservation of scaled entropy in conservative form is expressed as

$$\left. \frac{\partial \eta_{\mathcal{X}}}{\partial t} \right|_{\mathbf{x}} = J_{\Psi} \left(s + \dot{D}_{\text{int}} \right) - \nabla_{\mathbf{x}} \cdot \left(\frac{\mathbf{Q}_{\mathcal{X}}}{\theta} - \eta \mathbf{H}_{\Psi}^T \mathbf{W} \right), \quad (4.28)$$

with $\eta_{\mathcal{X}} = J_{\Psi} \eta$. Alternatively it is interesting to consider a non-conservative ALE transformation, which leads to the following ALE form

$$\left. \frac{\partial \eta}{\partial t} \right|_{\mathbf{x}} + \frac{1}{\theta} (\nabla_{\mathbf{x}} \cdot \mathbf{Q}_{\mathcal{X}}) = (\nabla_{\mathcal{X}} \eta) \cdot (\mathbf{F}_{\Psi}^{-1} \mathbf{W}) + \frac{s + \dot{D}_{\text{int}}}{\theta}. \quad (4.29)$$

The above equation will be used in the numerical framework presented in Chapter 6.

4.4 Nearly-incompressible thermo-mechanical constitutive models

The near-incompressibility assumption (as they were established in the isothermal case in Section 3.3) lets the internal strain energy potential at reference temperature be expressed as an additive distortional-volumetric decomposition, which can be stated as

$$\tilde{\mathcal{E}}_R(\mathcal{X}, \alpha) = \tilde{\mathcal{E}}'_R(\mathcal{X}, \alpha) + \tilde{U}_R(J). \quad (4.30)$$

In Equation (4.30), $\tilde{\mathcal{E}}'_R$ is the distortional or deviatoric component and \tilde{U}_R is the volumetric component of the strain energy, and both only depend on deformation measures. For hyperelastic materials with thermal-volumetric coupling, a pure volumetric entropy density $\tilde{\eta}_R(\mathcal{X}, \alpha) \approx \tilde{\eta}_R(J, \alpha = \mathbf{0})$ is considered. This is motivated by the intuitive expectation that changes of temperature will only lead to changes in volume. Mathematically, this corresponds to a decomposition of the internal energy \mathcal{E} in an entropy-dependent volumetric component U and model-dependent isothermal deviatoric component $\tilde{\mathcal{E}}'_R$ that is analogous to its isothermal counterpart. Using the aforementioned decomposition and the expression in Equation (4.30), the Equation (4.16a) is then rewritten as

$$\begin{aligned} \mathcal{E}(\mathcal{X}_{\eta\alpha}) &= \tilde{\mathcal{E}}_R(\mathcal{X}, \alpha) + c_v \theta_R \left(\exp \frac{\eta(\mathbf{X}, t) - \tilde{\eta}_R(J)}{c_v} - 1 \right) \\ &= \tilde{\mathcal{E}}'_R(\mathcal{X}) + U(J, \eta) \\ &= \tilde{\mathcal{E}}'_R(\mathcal{X}, \alpha) + \tilde{U}_R(J) + c_v \theta_R \left(\exp \frac{\eta(\mathbf{X}, t) - \tilde{\eta}_R(J)}{c_v} \right), \end{aligned} \quad (4.31)$$

where the volumetric potential U is itself defined as the sum of a non-thermally coupled volumetric component \tilde{U}_R and a thermal contribution, stated as

$$U(J, \eta) = \tilde{U}_R(J) + c_v(\theta - \theta_R); \quad \tilde{U}_R(J) = U(J, \eta = \tilde{\eta}_R(J)). \quad (4.32a,b)$$

The first line of Equation (4.31) corresponds to the isothermal-thermal decomposition of \mathcal{E} , and the second line corresponds to the deviatoric and (thermally-coupled) volumetric decomposition of \mathcal{E} . Equation (4.31) gives the expression of \mathcal{E} in terms of model-dependent potentials.

Remark 4.4.1 *The work conjugates featured in Equation (4.17) are then expressed as*

$$\Sigma_{\mathbf{F}} = \frac{\partial \tilde{\mathcal{E}}'_R}{\partial \mathbf{F}}; \quad \Sigma_{\mathbf{H}} = \frac{\partial \tilde{\mathcal{E}}'_R}{\partial \mathbf{H}}; \quad \Sigma_J = \hat{\Sigma}'_J + p. \quad (4.33a,b,c)$$

As $\tilde{\mathcal{E}}'_R$ corresponds to the isothermal distortional energy potential, these work conjugates correspond to those of the isothermal case.

It follows from Equation (4.31) that the volumetric contribution is decomposed in a isothermal mechanical part $\hat{\Sigma}'_J$ and a thermally coupled hydrostatic pressure p as

$$\hat{\Sigma}'_J = \frac{\partial \tilde{\mathcal{E}}'_R}{\partial J}; \quad p(J, \eta) = \frac{\partial U}{\partial J} = \tilde{p}_R + p_\eta, \quad (4.34a,b,c)$$

with

$$\tilde{p}_R(J) = \frac{\partial \tilde{U}_R(J)}{\partial J}; \quad p_\eta(J, \eta) = \frac{\partial U_\eta(J, \eta)}{\partial J}. \quad (4.35a,b)$$

With this notation at hand, similar derivations can be done to obtain the components of the first Piola Kirchhoff stress tensor as for isothermal considerations in Equation (3.7). It can be seen that the first Piola Kirchhoff stress receives a thermal contribution via its volumetric component, while the conjugate stress Σ'_J featured in the deviatoric component is equivalent to that of the isothermal case.

4.4.1 Mie-Grüneisen Equation of state

The Mie-Grüneisen hydrodynamic equation of state [8, 10, 23] provides an expression for volumetric thermal coupling. It relates pressure and thermal internal energy as

$$\left. \frac{\partial p}{\partial \mathcal{E}} \right|_{J=\text{const}} = -\Gamma_0 J^{q-1}, \quad (4.36)$$

where $0 \leq q \leq 1$ is a dimensionless positive parameter (for solid materials, $q = 1$), and $\Gamma_0 > 0$ is a positive material constant. This model can also be stated using the thermal expansion coefficient $\alpha = c_v \Gamma_0 / 3\kappa$. Using the definition of the pressure (4.35), the relative entropy $\tilde{\eta}_R$ is given an expression depending on the geometric volume map defined as

$$\tilde{\eta}_R(J) = c_v \Gamma_0 \frac{J^q - 1}{q}, \quad (4.37)$$

whose convexity can be proven analytically by noticing that

$$-\frac{d^2 \tilde{\eta}_R(J)}{dJ^2} = (1 - q)c_v \Gamma_0 J^{q-2} \geq 0. \quad (4.38)$$

This relation completes an appropriate polyconvex isothermal deviatoric potential (defined in Section 3.2; see for instance Mooney-Rivlin material 3.3.1), and an appropriate volumetric potential (based, for instance, on a potential listed in Appendix B.3.2)

$$\tilde{U}_R(J) = \frac{\kappa}{2} (J-1)^2 + c_v \Gamma_0 \theta_R (J-1). \quad (4.39)$$

The Mie-Grüneisen model is provided by an universally polyconvex strain energy potential $\tilde{\mathcal{E}}'(\boldsymbol{\mathcal{X}})$ that is defined, for a general Mooney-Rivlin material (as presented in Section 3.3.1 for a non-thermally coupled material), as

$$\tilde{\mathcal{E}}'_R(\boldsymbol{\mathcal{X}}) = \zeta \left(J^{-2/3} II_{\mathbf{F}} - 3 \right) + \xi \left(J^{-2} II_{\mathbf{H}}^{3/2} - 3\sqrt{3} \right). \quad (4.40)$$

For a general hyperelastic deviatoric potential, material parameters can be formulated as

$$\mu = 2\zeta + 3\sqrt{3}\gamma; \quad \kappa = \xi + c_v \theta_R \Gamma_0 (1-q); \quad \alpha = \frac{c_v \Gamma_0}{3\kappa}. \quad (4.41a,b,c)$$

In the case of solid materials, the thermal coefficient q is equal to 1. Moreover, a neo-Hookean model type is recovered by setting $\gamma = 0$. The strain energy potential (4.40) for a Mie-Grüneisen model can then be expressed as

$$\tilde{\mathcal{E}}'_R(\boldsymbol{\mathcal{X}}) = \frac{\mu}{2} \left(J^{-2/3} II_{\mathbf{F}} - 3 \right) + \kappa \left(J^{-2} II_{\mathbf{H}}^{3/2} - 3\sqrt{3} \right). \quad (4.42)$$

If $\Gamma_0 = 0$, it results that $\kappa = 0$ and Equation (4.42) degenerates to the neo-Hookean model for non-thermally coupled material, defined in Equation (3.16).

4.5 Thermal visco-plasticity

In high-energy impact or penetration problems, large deformations are not only induced by the material's plasticity, but also by viscous effects and temperature fluctuations. Numerically, it means that plastic deformations depend on strains, strain rates and thermal softening. In addition to this, the generated heat also has effects on the hyperelastic behaviour. In this work, a thermo-hyperelastic-viscoplastic solid material, as presented in [11], is considered. It is characterised by a volumetric hyperelastic energy potential coupled with thermal effects (4.37), and by a rheological rate-dependent Johnson-Cook hardening law [26].

4.5.1 Extension to thermo-plasticity

There is a wide variety of complex models to describe the plastic deformation and failure of metals under dynamic loading conditions (e.g. Zerilli-Armstrong model, Gurson-Tvergaard-Needleman Model [25]). In this work, the isotropic nonlinear Johnson-Cook hardening law [26, 258] is considered, and can be expressed as

$$\tau_y(\bar{\varepsilon}_p, \dot{\bar{\varepsilon}}_p, \theta) = \left(\tau_y^0 + H \bar{\varepsilon}_p^{q'} \right) \left(1 + C \ln \left(\frac{\dot{\bar{\varepsilon}}_p}{\dot{\bar{\varepsilon}}_p^0} \right) \right) \left(1 - \hat{\theta}^m \right), \quad (4.43)$$

which accounts for isotropic hardening (for an initial yield stress τ_y^0 , a hardening modulus H , and a power exponent q'), strain rate $\dot{\bar{\varepsilon}}_p$ dependency (for an initial strain rate $\dot{\bar{\varepsilon}}_p^0$), a material parameter C , and thermal softening (for a material-dependent exponent m)

$$\hat{\theta} = \begin{cases} 0 & \text{if } \theta < \theta_{\text{transition}}, \\ \frac{\theta - \theta_{\text{transition}}}{\theta_{\text{melting}} - \theta_R} & \text{if } \theta_{\text{transition}} \leq \theta \leq \theta_{\text{melting}}, \\ 1 & \theta > \theta_{\text{melting}} \end{cases} \quad (4.44)$$

as three independent phenomena. In Equation (4.44), the transition temperature $\theta_{\text{transition}}$, and the melting temperature θ_{melting} are material parameters. Accordingly, the loading condition for plasticity is

$$0 < \phi_f(\boldsymbol{\tau}, \bar{\varepsilon}_p, \dot{\varepsilon}_p, \theta) = \bar{\tau}(\boldsymbol{\tau}') - \tau_y(\bar{\varepsilon}_p, \dot{\varepsilon}_p, \theta). \quad (4.45)$$

As for isothermal plasticity, the internal plastic variables are updated with an implicit Euler time integrator. The algorithmic procedure for stress evaluation of the present ALE vertex-based finite volume framework for non-linear solid dynamics in irreversible processes with thermal coupling is described in Algorithm 3. This algorithm is based on Algorithm 1 for

Algorithm 3: Hencky elasticity model with von-Mises plasticity and Johnson-Cook hardening.

Data: $\mathbf{F}_{n+1}, \mathbf{C}_{p,n}^{-1}, \bar{\varepsilon}_{p,n}, \theta_n$

- 1 Pressure Correction p_{n+1} (Equation (4.34c));
 - 2 Elastic finger tensor $\mathbf{b}_{e,n+1}^{\text{trial}} = \mathbf{F}_{n+1} \mathbf{C}_{p,n}^{-1} \mathbf{F}_{n+1}^T$;
 - 3 Principal components decomposition $\mathbf{b}_{e,n+1}^{\text{trial}} = \sum_{i=1}^3 (\lambda_{e,i}^{\text{trial}})^2 \mathbf{n}_i^{\text{trial}} \otimes \mathbf{n}_i^{\text{trial}}$;
 - 4 Trial deviatoric Kirchhoff $\boldsymbol{\tau}_{n+1}^{\text{trial}} = 2\mu \ln \sum_{i=1}^3 \left(J^{-1/3} \lambda_{e,i}^{\text{trial}} \mathbf{n}_i^{\text{trial}} \otimes \mathbf{n}_i^{\text{trial}} \right)$;
 - 5 **if** $\phi_f(\boldsymbol{\tau}_{n+1}^{\text{trial}}, \bar{\varepsilon}_{p,n}, \dot{\varepsilon}_{p,n}, \theta_n) > 0$ (Loading Condition (4.45))
 - 6 Derivative of yield function $\boldsymbol{\nu}_{n+1} = \frac{\boldsymbol{\tau}_{n+1}^{\text{trial}}}{\sqrt{\frac{2}{3} \boldsymbol{\tau}_{n+1}^{\text{trial}} : \boldsymbol{\tau}_{n+1}^{\text{trial}}}}$;
 - 7 Plastic Multiplier $\Delta\gamma$ (Johnson-Cook hardening (4.43));
 - 8 Logarithmic Elastic stretches $\lambda_{e,n+1} = \exp(\ln(\lambda_e^{\text{trial}}) - \Delta\gamma \boldsymbol{\nu}_{n+1})$;
 - 9 Deviatoric Kirchhoff $\boldsymbol{\tau}'_{n+1} = \boldsymbol{\tau}_{n+1}^{\text{trial}} - 2\mu \Delta\gamma \boldsymbol{\nu}_{n+1}$;
 - 10 Reconstruction of elastic finger $\mathbf{b}_{e,n+1} = \sum_{i=1}^3 \lambda_{e,i}^2 \mathbf{n}_i^{\text{trial}} \otimes \mathbf{n}_i^{\text{trial}}$;
 - 11 **else**
 - 12 $\boldsymbol{\tau}'_{n+1} = \boldsymbol{\tau}_{n+1}^{\text{trial}}$;
 - 13 $\mathbf{b}_{e,n+1} = \mathbf{b}_{e,n+1}^{\text{trial}}$;
 - 14 Plastic strain update $\bar{\varepsilon}_{p,n+1} = \bar{\varepsilon}_{p,n} + \Delta\gamma$;
 - 15 Inverse of plastic right Cauchy $\mathbf{C}_{p,n+1}^{-1} = \mathbf{F}_{n+1}^{-1} \mathbf{b}_{e,n+1} \mathbf{F}_{n+1}^{-T}$;
 - 16 Kirchhoff update $\boldsymbol{\tau}_{n+1} = \boldsymbol{\tau}'_{n+1} + Jp\mathbf{I}$;
 - 17 Piola update $\mathbf{P}_{n+1} = \boldsymbol{\tau}_{n+1} \mathbf{F}_{n+1}^{-T}$;
 - 18 von-Mises equivalent stress $\bar{\tau}(\boldsymbol{\tau}')$
-
- Result:** $\mathbf{P}_{n+1}, \mathbf{C}_{p,n+1}^{-1}, \bar{\varepsilon}_{p,n+1}, \bar{\tau}_{n+1}$
-

isothermal plasticity, and allows for thermo-volumetric coupling (via the thermally coupled hydrostatic pressure update) and for thermal hardening/softening. As it will be presented in Algorithm 5, it was chosen that the temperature remains unchanged throughout the plasticity update to preserve the simple structure of the algorithm. In absence of thermal expansion ($\Gamma_0 = 0$), the reference entropy is null. Additionally, a constant temperature $\theta = \theta_R$ implies that the entropy (4.15) is also constant, the volumetric potential U in

Equation (4.37) degenerates to its pure isothermal counterpart, and the thermal softening in Equation (4.43) vanishes.

4.6 Conclusion

In this Chapter, the two laws of thermodynamics were introduced and incorporated in the numerical framework; firstly, the first law of thermodynamics, featured in Section 4.2, provided a conservation law on the total energy that can be used to measure the evolution of the energy distribution of a system, whether the considered body account for thermal effects or not. In Section 4.3, the Calorimetry relationships were used to outline relations between the energy, the temperature and the entropy, and the second law of thermodynamics provided an expression for the evolution of the numerical dissipation under the Clausius-Duhem inequality. Nearly incompressible models are established for thermo-mechanical models, and more especially the volumetric coupling is presented therein, and the Mie-Grüneisen Equation of state was described in Section 4.4. Eventually, and following the development of Chapter 3, the extension of the present ALE formulation to thermo-visco-plasticity is conducted in Section 4.5 with the association of the energy conservation equation, the thermo-mechanical volumetric coupling and the Johnson-Cook rate-dependent visco-plastic hardening law.

So far, the ALE formulation has been extended to a mixed system incorporating the equations of thermo-visco-plasticity. The scope of this present work is restricted to using the energy equation, a volumetric coupling and a Johnson-Cook hardening law, but the numerical framework can be easily adapted to different considerations in many aspects. For instance more complex kinematic hardening law, thermo-mechanical coupling or the entropy equation could be taken into account without real implementation issue, due to the clear modularity of the pieces constituting this ALE framework. In fact, those are physical considerations that will not any impact on the mesh motion technique presented in the following Chapter.

CHAPTER 5

MATERIAL MOTION

“A mathematician who is not somewhat of a poet, will never be a perfect mathematician.”

— KARL WEIERSTRASS, 1883

5.1 Introduction

The choice of the kinematical description of the continuum is crucial in numerical simulations where the mesh plays a major role³⁶. In an attempt to handle greater distortions than with a Lagrangian method and with more resolution than with an Eulerian method, those two approaches are generalised as the ALE method where the mesh displacements are solved as an independent quantity. Inherently, the re-meshing technique is the main determinant of the success of the ALE framework. A thorough review of ALE mesh-update techniques in Computational Mechanics was done by [113]. Implicit methods are traditionally used to update the mesh: Mesh refinement trade coarse mesh granularity for higher local resolution and higher computational cost [163, 167], while Lagrangian smoothing [149] or mid-area averaging [259] is a simple technique that may fail on complex geometries. The mesh-update technique presented in this work is based on an explicit unsplit approach. This present method circumvents the need to split the solution process into material and convection stages, and relies on the introduction of suitable mappings between the new referential domain and the respective material and spatial domains. As presented in Chapter 2, the natural deformation gradient is obtained via a multiplicative decomposition $\mathbf{F} = \mathbf{F}_\Phi \mathbf{F}_\Psi^{-1}$, and the material gradient \mathbf{F}_Ψ will be solved from the referential domain together with the material velocity \mathbf{W} using a conservation law closed with a hyperelastic constitutive model [6]. The aim of this technique is to better accept mesh distortions, thence focusing on the quality of the spatial mesh [178], and propose mesh motion equations that can be integrated into the mixed formulation (2.75).

5.1.1 Geometrical requirements

A first series of geometrical requirements was developed in Chapter 2, taking the form of a rate-form conservation equation for the material deformation gradient \mathbf{F}_Ψ and the material Jacobian J_Ψ . Note that as it was done in Lagrangian in [56, 75, 111], it is also possible to derive a conservation law for the material area map \mathbf{H}_Ψ , leading to a potentially more accurate compressible mesh motion [78, 111]. In this work, the material mesh motion will be assumed constitutively hyperelastic and solved in an advection problem. Henceforth, a conservation law will govern the evolution of the material linear momentum $\mathbf{p}_W = \rho_R \mathbf{W}$ using a material stress \mathbf{P}_W deriving from a hyperelastic potential [3–6]. The new conservation law will be presented in Section 5.2, and appropriate constitutive models will be presented in Section 5.2.1

5.2 Material Balance law

Following the constitutive approach in [6], the mesh motion can be solved via a conservation law on the material linear momentum \mathbf{p}_W . The local form and the associated jump condition are expressed as

$$\left. \frac{\partial \mathbf{p}_W}{\partial t} \right|_{\mathbf{x}} - \nabla_{\mathbf{x}} \cdot \mathbf{P}_W = \mathbf{0}, \quad (5.1)$$

with the jump condition

$$c_{\mathbf{x}} \llbracket \mathbf{p}_W \rrbracket = - \llbracket \mathbf{P}_W \rrbracket \mathbf{N}_{\mathbf{x}}. \quad (5.2)$$

Artificial damping is a technique often used [43, 260–262] to reduce high-frequencies oscillations and improve stability of the computation in the context of explicit dynamic frameworks. The conservation of the material linear momentum (5.1) can be provided with

³⁶For instance, large mesh distortions, material interface tracking and mobile boundaries.

artificial damping coefficient as

$$\left. \frac{\partial \mathbf{p}_W}{\partial t} \right|_{\mathbf{x}} - \nabla_{\mathbf{x}} \cdot \mathbf{P}_W = -\nu_{\Psi} \mathbf{p}_W, \quad (5.3)$$

where $0 \leq \nu_{\Psi} [\text{s}^{-1}] < 1$ is a viscosity-like coefficient controlling the amount of numerical damping for the conservation of \mathbf{p}_W . This conservation law can be stated in global form as

$$\left. \frac{\partial}{\partial t} \right|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathbf{p}_W \, dV_{\mathbf{x}} - \int_{\partial\Omega_{\mathbf{x}}} \mathbf{t}_W \, dA_{\mathbf{x}} = - \int_{\Omega_{\mathbf{x}}} \nu_{\Psi} \mathbf{p}_W \, dV_{\mathbf{x}}, \quad (5.4)$$

and can be incorporated to the mixed system (2.75).

To ensure accurate tracking of the body's moving boundaries, they are necessarily treated as Lagrangian surfaces [113]. Consequently boundary nodes have to physically remain on these boundaries and are only allowed to move tangentially which leads to the boundary condition $\mathbf{W} \cdot \mathbf{N}_{\mathbf{X}}$ for the mesh velocity. To alleviate spatial mesh distortions, the mesh motion machinery must take into account spatial geometric variables \mathcal{X}_{Φ} . The keystone of the ALE methodology in this work is to use an energy potential W_{sm} defined as a convex sum of a material and a scalar potential

$$W_{\text{sm}}(\mathcal{X}_{\Psi}, \tilde{\mathcal{X}}_{\Phi}) = (1 - \alpha_{\text{ALE}})W(\mathcal{X}_{\Psi}) + \alpha_{\text{ALE}}W(\tilde{\mathcal{X}}_{\Phi}), \quad (5.5)$$

with $0 \leq \alpha_{\text{ALE}} \leq 1$ the averaging parameter, W a hyperelastic strain energy potential, $\tilde{\mathbf{F}}$ a linear function of the natural deformation gradient \mathbf{F} , and with the material and spatial triplets respectively defined as

$$\mathcal{X}_{\Psi} = \{\mathbf{F}_{\Psi}, \mathbf{H}_{\Psi}, J_{\Psi}\}; \quad \tilde{\mathcal{X}}_{\Phi} = \{\tilde{\mathbf{F}}_{\Phi}, \tilde{\mathbf{H}}_{\Phi}, \tilde{J}_{\Phi}\}. \quad (5.6a,b)$$

The spatial mappings are reconstructed from the natural deformation gradient \mathbf{F} and the material mappings as

$$\tilde{\mathbf{F}}_{\Phi} = \tilde{\mathbf{F}}(\mathbf{F})\mathbf{F}_{\Psi}; \quad \tilde{J}_{\Phi} = \det(\tilde{\mathbf{F}}\mathbf{F}_{\Psi}); \quad \tilde{\mathbf{H}}_{\Phi} = \tilde{\mathbf{H}}\mathbf{H}_{\Psi}. \quad (5.7a,b,c)$$

The time derivative of Equation (5.5) enables the definition of a specific material stress tensor that, using the general relationships of a hyperelastic model presented in Equation (3.6), reads

$$\mathbf{P}_W : \nabla_{\mathbf{x}} \mathbf{W} = \left. \frac{\partial W_{\text{sm}}}{\partial t} \right|_{\mathbf{x}} = \left[(1 - \alpha_{\text{ALE}})\mathbf{P}(\mathcal{X}_{\Psi}) + \alpha_{\text{ALE}}\tilde{\mathbf{F}}^T \mathbf{P}(\tilde{\mathcal{X}}_{\Phi}) \right] : \nabla_{\mathbf{x}} \mathbf{W}, \quad (5.8)$$

which yields an expression for the material first Piola-Kirchhoff stress tensor

$$\begin{aligned} \mathbf{P}_W &= (1 - \alpha_{\text{ALE}})\mathbf{P}(\mathcal{X}_{\Psi}) + \alpha_{\text{ALE}}\tilde{\mathbf{F}}^T \mathbf{P}(\tilde{\mathcal{X}}_{\Phi}) \\ &= (1 - \alpha_{\text{ALE}})\left(\Sigma_{\mathbf{F}_{\Psi}} + \Sigma_{\mathbf{H}_{\Psi}} \times \mathbf{F}_{\Psi} + \Sigma_{J_{\Psi}} \mathbf{H}_{\Psi}\right) \\ &\quad + \alpha_{\text{ALE}}\tilde{\mathbf{F}}^T \left(\Sigma_{\tilde{\mathbf{F}}_{\Phi}} + \Sigma_{\tilde{\mathbf{H}}_{\Phi}} \times \tilde{\mathbf{F}}_{\Phi} + \Sigma_{\tilde{J}_{\Phi}} \tilde{\mathbf{H}}_{\Phi}\right). \end{aligned} \quad (5.9)$$

An appropriate constitutive model for the deviatoric stress \mathbf{P}_{dev} is presented in Section 5.2.1. The volumetric stress component \mathbf{P}_{vol} can be modelled by one of the potentials shown in Appendix B.3.2.

Algorithm 4: Update of \mathbf{P}_W

-
- 1 Construction of natural deformation gradient $\tilde{\mathbf{F}}(\mathbf{F})$
 - 2 Reconstruction of the spatial deformation gradient $\tilde{\mathbf{F}}_{\Phi} = \tilde{\mathbf{F}}(\mathbf{F})\mathbf{F}_{\Psi}$ (Equation (5.7a))
 - 3 Choice of material potential W_{sm} (Equation (5.5)) and computation to the corresponding stress \mathbf{P}_W (Equation (5.9))
-

5.2.1 Neo-Hookean mesh model

The nearly incompressible neo-Hookean energy potential presented in Section 3.3.1 is used for the modelling the material first Piola Kirchhoff stresses in the material balance law. The strain energy potential $W(\mathcal{X}_{\Psi}) = W(\mathbf{F}_{\Psi})$ for the material deformation, and the strain energy potential $W(\mathcal{X}_{\Phi}) = W(\mathbf{F}_{\Phi})$ for the spatial deformation are defined as

$$W(\mathbf{F}_{\Psi}) = \frac{\mu}{2} \left[\left(J^{-2/3} (\mathbf{F}_{\Psi} : \mathbf{F}_{\Psi}) \right) - 3 \right]; \quad W(\mathbf{F}_{\Phi}) = \frac{\mu}{2} \left[\left(J^{-2/3} (\mathbf{F}_{\Phi} : \mathbf{F}_{\Phi}) \right) - 3 \right]. \quad (5.10a,b)$$

The stress tensors respectively associated to $W(\mathcal{X}_{\Phi})$ and $W(\mathcal{X}_{\Psi})$ are defined as

$$\mathbf{P}_{\text{dev}}(\mathbf{F}_{\Psi}) = \mu J^{-2/3} \mathbf{F}_{\Psi} - \frac{\mu}{3} J^{-5/3} (\mathbf{F}_{\Psi} : \mathbf{F}_{\Psi}) \mathbf{F}_{\Psi} \quad (5.11a)$$

$$\mathbf{P}_{\text{dev}}(\mathbf{F}_{\Phi}) = \mu J^{-2/3} \mathbf{F}_{\Phi} - \frac{\mu}{3} J^{-5/3} (\mathbf{F}_{\Phi} : \mathbf{F}_{\Phi}) \mathbf{F}_{\Phi}. \quad (5.11b)$$

Note that the stress tensors defined in Equations (5.11a) and (5.11b) are purely deviatoric. If the mesh motion is considered hyperelastic, a volumetric potential can be used to complete the model as for the first Piola Kirchhoff stress \mathbf{P} applied to the physical body. Candidates for volumetric potential are shown in Appendix B.3.2. In the case where the potential W is neo-Hookean, the conjugate stresses required for the expression of the first Piola Kirchhoff stress defined in Equation (5.9) are expressed as

$$\boldsymbol{\Sigma}_{\mathbf{F}_{\Psi}} = 2\zeta J^{-2/3} \mathbf{F}_{\Psi}, \quad (5.12a)$$

$$\boldsymbol{\Sigma}_{\mathbf{H}_{\Psi}} = 3\xi J^{-2} (\mathbf{H}_{\Psi} : \mathbf{H}_{\Psi})^{1/2} \mathbf{F}_{\Psi}, \quad (5.12b)$$

$$\hat{\Sigma}_{J_{\Psi}} = -\frac{2}{3}\zeta J^{-5/3} (\mathbf{F}_{\Psi} : \mathbf{F}_{\Psi}), \quad (5.12c)$$

and

$$\boldsymbol{\Sigma}_{\mathbf{F}_{\Phi}} = 2\zeta J^{-2/3} \mathbf{F}_{\Phi}, \quad (5.12d)$$

$$\boldsymbol{\Sigma}_{\mathbf{H}_{\Phi}} = 3\xi J^{-2} (\mathbf{H}_{\Phi} : \mathbf{H}_{\Phi})^{1/2} \mathbf{F}_{\Phi}, \quad (5.12e)$$

$$\hat{\Sigma}_{J_{\Phi}} = -\frac{2}{3}\zeta J^{-5/3} (\mathbf{F}_{\Phi} : \mathbf{F}_{\Phi}). \quad (5.12f)$$

5.3 Conclusion

The core of the ALE numerical framework resides in the mesh motion technique, as it is built around the conservation law of the material linear momentum \mathbf{p}_W . The most common difficulties usually encountered when using a Lagrangian formulation and a static mesh are outstanding distortions. The conservation law of \mathbf{p}_W , or material balance law, presented in Section 5.2 of this Chapter, resorts to a material stress tensor \mathbf{P}_W that is computed via a conservative law and expressed in terms of the material deformation gradient \mathbf{F}_{Ψ} . The new material balance law is hyperbolic and can be integrated to the ALE mixed system. Additionally, artificial damping can be added to the mesh motion to alleviate

potential spurious oscillations in the material motion. The magnitude of the mesh motion is characterised by the coefficient $0 \leq \alpha_{\text{ALE}} \leq 1$. If $\alpha_{\text{ALE}} = 0$, the material momentum becomes null, the material geometric mappings verify $\{\mathbf{F}_\Psi, \mathbf{H}_\Psi, J_\Psi\} = \{\mathbf{I}, \mathbf{I}, 1\}$. In the case of α_{ALE} always equal to 0, the ALE mixed formulation degenerates to its Total Lagrangian equivalent. Recalling the form (2.75), the balance laws, geometrical conservation laws, conservation laws for plastic internal variables, the conservation of the spatial volume map and the entropy-related equation are all combined into the set of ALE conservation laws defined in the following:

$$\mathbf{u} = \begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ C_p^{-1} \\ \bar{\varepsilon}_p \\ \eta \\ J_\Phi \\ p_W \end{bmatrix}; \mathcal{F}_I = - \begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ C_p^{-1} [J_\Psi^{-1} \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I] \\ \bar{\varepsilon}_p [J_\Psi^{-1} \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I] \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \\ P_W \mathbf{E}_I \end{bmatrix}; \mathcal{S} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ -2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \tau} \mathbf{b}_e \mathbf{F}^{-T} \\ \dot{\gamma} \\ -\frac{1}{\theta} (\nabla_\chi \cdot \mathbf{Q}_\chi) + (\nabla_\chi \eta) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + \frac{s + \dot{D}_{\text{int}}}{\theta} \\ 0 \\ \mathbf{0} \end{bmatrix}. \quad (5.13\text{a,b,c})$$

Alternatively, the energy-based ALE mixed formulation is defined as

$$\mathbf{u} = \begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ C_p^{-1} \\ \bar{\varepsilon}_p \\ E_\chi \\ J_\Phi \\ p_W \end{bmatrix}; \mathcal{F}_I = - \begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ C_p^{-1} [J_\Psi^{-1} \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I] \\ \bar{\varepsilon}_p [J_\Psi^{-1} \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I] \\ [P_\chi^T \mathbf{v} + E_R \mathbf{H}_\Psi^T \mathbf{W} - \mathbf{Q}_\chi] \cdot \mathbf{E}_I \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \\ P_W \mathbf{E}_I \end{bmatrix}; \mathcal{S} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ -2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \tau} \mathbf{b}_e \mathbf{F}^{-T} \\ \dot{\gamma} \\ \mathbf{f} \cdot \mathbf{v} + s_\chi \\ 0 \\ \mathbf{0} \end{bmatrix}. \quad (5.14\text{a,b,c})$$

5.3.1 Different ALE frameworks

The enforcement of suitable kinematic restrictions enables the above ALE system (5.13) to degenerate into three alternative systems of first-order conservation equations for fast solid dynamics:

- The Total Lagrangian approach can be recovered by ensuring that both referential and material domains overlap. It corresponds to enforcing $\mathbf{W} = \mathbf{0}$ and $\mathbf{F}_\Psi = \mathbf{I}$, which in turn yields $\chi = \mathbf{X}$, $\hat{\mathbf{v}} = \mathbf{v}$ and $\mathbf{F}_\Phi = \mathbf{F}$.
- The respective Eulerian system is obtained when the referential domain coincides with the spatial domain. This is achieved by setting $\hat{\mathbf{v}} = \mathbf{0}$, and $\mathbf{F}_\Phi = \mathbf{I}$, leading to $\chi = \mathbf{x}$, $\mathbf{W} = -\mathbf{F}^{-1} \mathbf{v}$ and $\mathbf{F}_\Psi = \mathbf{F}^{-1}$.

- A proposed Updated Reference Lagrangian formulation [20, 37], based on incremental kinematics, can be obtained via the introduction of a intermediate configuration for the multiplicative decomposition of the conservation variables. By setting $\mathbf{W} = \mathbf{0}$, and with given (or prescribed) material mapping functions such as $\mathbf{F}_\Psi = \bar{\mathbf{F}}_\Psi$ (which implies $\boldsymbol{\chi} = \bar{\boldsymbol{\chi}}, \hat{\mathbf{v}} = \mathbf{v}$ and $\mathbf{F} = \mathbf{F}_\Phi (\bar{\mathbf{F}}_\Psi)^{-1}$, the so-called corresponding Updated Lagrangian system is obtained.

Alternatively, the ALE system (5.13) can be modified by simplifying the complexity of the numerical problem. On one hand, a mixed formulation based on different physics can be obtained by considering the adequate equations (e.g. a simple hyperelastic problem can be modelled with a $\mathbf{p} - \mathbf{F}$ formulation). On the other hand, the same choices can be made for the mesh motion model.

\mathbf{u}	\mathcal{F}_I	\mathbf{s}
\mathbf{p}-\mathbf{F} formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}$
\mathbf{p}-\mathbf{F} formulation with material \mathbf{F}-\mathbf{J} formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \end{bmatrix}$
\mathbf{p}-\mathbf{F} formulation with von-Mises plasticity and material \mathbf{F}-\mathbf{J} formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ C_p^{-1} \\ \bar{\varepsilon}_p \\ J_\Phi \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ \mathbf{0} \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ (\nabla_\chi C_p^{-1}) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) - 2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T} \\ (\nabla_\chi \bar{\varepsilon}_p) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + \dot{\gamma} \\ 0 \end{bmatrix}$
\mathbf{p}-\mathbf{F}-η formulation with material \mathbf{F}-\mathbf{J} formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ \eta \\ J_\Phi \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} \mathbf{f} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ \frac{1}{\theta} (\nabla_\chi \cdot \mathbf{Q}_\chi) + (\nabla_\chi \eta) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + J_\Psi \frac{s + \dot{D}_{\text{int}}}{\theta} \\ 0 \end{bmatrix}$

TABLE 5.1: Examples of ALE mixed formulations with a prescribed material motion.

The components of the alternative systems are shown in Tables tables 5.1 and 5.2 (The velocity-deformation gradient formulation is abbreviated \mathbf{p} - \mathbf{F}). The motivation to use non-conservative forms is discussed in Section 6.6 of Chapter 6.

\mathcal{U}	\mathcal{F}_I	\mathcal{S}
p-F formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ p_W \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ P_W \mathbf{N}_\chi \end{bmatrix}$	$\begin{bmatrix} f \\ \mathbf{0} \\ 0 \end{bmatrix}$
p-F formulation with material p-F-J formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ p_W \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ P_W \mathbf{N}_\chi \end{bmatrix}$	$\begin{bmatrix} f \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ 0 \end{bmatrix}$
p-F formulation with von-Mises plasticity and material p-F-J formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ C_p^{-1} \\ \bar{\varepsilon}_p \\ J_\Phi \\ p_W \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ 0 \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \\ P_W \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} f \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ (\nabla_\chi C_p^{-1}) (\mathbf{F}_\Psi^{-1} \mathbf{W}) - 2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T} \\ (\nabla_\chi \bar{\varepsilon}_p) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + \dot{\gamma} \\ 0 \\ 0 \end{bmatrix}$
p-F-η formulation with material F-J formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ \eta \\ J_\Phi \\ p_W \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \\ P_W \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} f \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ \frac{1}{\theta} (\nabla_\chi \cdot \mathbf{Q}_\chi) + (\nabla_\chi \eta) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + J_\Psi \frac{s + \dot{D}_{\text{int}}}{\theta} \\ 0 \\ 0 \end{bmatrix}$
p-F-η formulation with material F-J formulation		
$\begin{bmatrix} p_\chi \\ \mathbf{F}_\Phi \\ \mathbf{F}_\Psi \\ J_\Psi \\ C_p^{-1} \\ \bar{\varepsilon}_p \\ \eta \\ J_\Phi \\ p_W \end{bmatrix}$	$-\begin{bmatrix} P_\chi \mathbf{E}_I \\ \hat{\mathbf{v}} \otimes \mathbf{E}_I \\ \mathbf{W} \otimes \mathbf{E}_I \\ \mathbf{H}_\Psi^T \mathbf{W} \cdot \mathbf{E}_I \\ 0 \\ 0 \\ 0 \\ \mathbf{H}_\Phi^T \hat{\mathbf{v}} \cdot \mathbf{E}_I \\ P_W \mathbf{E}_I \end{bmatrix}$	$\begin{bmatrix} f \\ \mathbf{0} \\ \mathbf{0} \\ 0 \\ (\nabla_\chi C_p^{-1}) (\mathbf{F}_\Psi^{-1} \mathbf{W}) - 2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T} \\ (\nabla_\chi \bar{\varepsilon}_p) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + \dot{\gamma} \\ \frac{1}{\theta} (\nabla_\chi \cdot \mathbf{Q}_\chi) + (\nabla_\chi \eta) \cdot (\mathbf{F}_\Psi^{-1} \mathbf{W}) + J_\Psi \frac{s + \dot{D}_{\text{int}}}{\theta} \\ 0 \\ 0 \end{bmatrix}$

TABLE 5.2: Examples of ALE mixed formulations with solved material mesh motion.

CHAPTER 6

NUMERICAL DISCRETISATION

“In mathematics as in other fields, to find one self lost in wonder at some manifestation is frequently the half of a new discovery.”

— JOHANN PETER GUSTAV LEJEUNE DIRICHLET, *Werke*, 1897

6.1 Introduction

In Chapter 5, the complete set of first order hyperbolic equations that constitute the ALE formulation was introduced in the continuum, including the conservation of the internal plastic variables and of the energy, and the conservation and model of the mesh motion. The work makes use of the Vertex Centred Finite Volume Method (VCFVM) which gives a methodology to spatially discretise the equations for the numerical solver. This chapter will introduce the main concepts utilised in VCFVM, including the use of a dual mesh (see Section 6.2), the general finite volume method (see Section 6.3), the Riemann solver based on a Godunov-type upwinding stabilisation technique (see Section 6.4), and the linear reconstruction procedure [75, 263] used in the gradient operator (see Section 6.5). Furthermore, the set of semi-discretised equations will be presented in Section 6.6, together with the different boundary conditions [56, 81] in Section 6.6.1. Eventually, this chapter will also shed light on the time integrator in Section 6.7.

6.2 Dual mesh

In this work, the (overlapping-free) median dual tessellation [81, 264–266] will be used on a tetrahedral mesh. Edge midpoints are connected to face centroids and elements centroids. This work will use the notation featured in [35, 264]. For a given node a , the set of nodes connected to it through an edge is denoted by Λ_a , and the subset of nodes connected to a through a boundary edge is denoted by Λ_a^B . The mean area vector \mathbf{C}^{ab} associated to the edge ab connecting a and b is defined as

$$\mathbf{C}_\chi^{ab} = \sum_{k \in \Gamma_{ab}} A_k \mathbf{N}_\chi^k = -\mathbf{C}_\chi^{ba}, \quad (6.1)$$

where Γ_{ab} is the set of facets belonging to edge ab , A_k is the area of facet k and \mathbf{N}_k the normal vector of k .

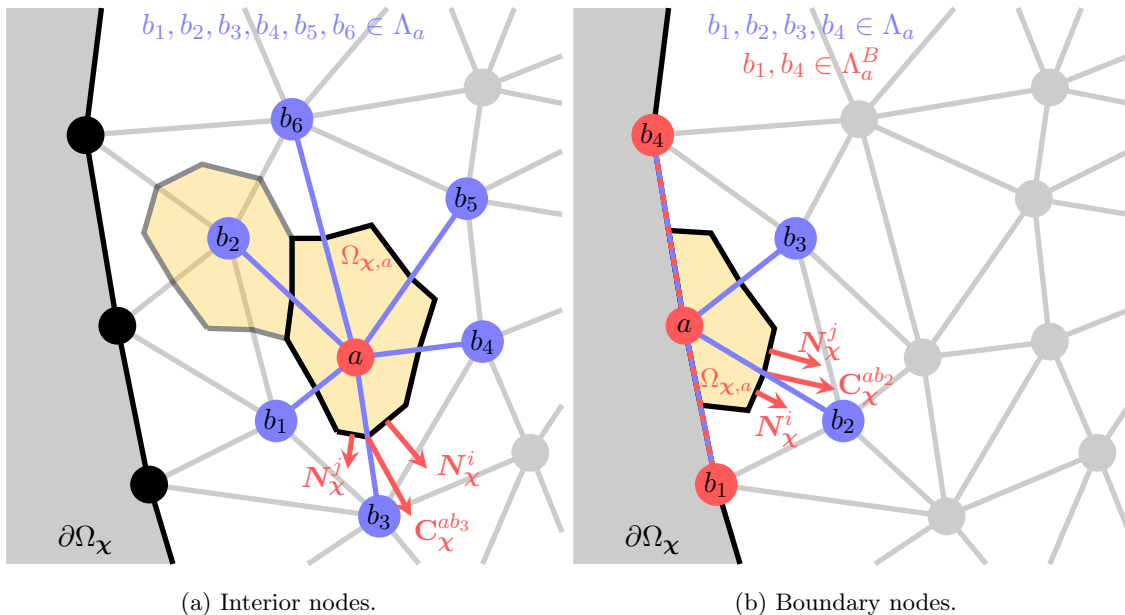


FIGURE 6.1: Control volumes for (a) an interior node and (b) a boundary node for a median dual tessellation visualised in 2D.

A two dimensional depiction of the construction of the dual mesh is shown in Figures 6.1, where both the case of an interior node and a boundary node a (red) is considered. The

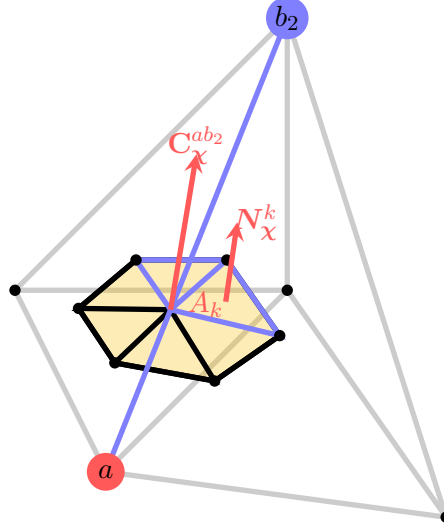


FIGURE 6.2: Area vector and its local contribution, for an edge ab_2 of a three dimensional tetrahedral mesh.

interior nodes associated to a , or in the first layer of the vicinity Λ_a of a , are shown in blue while the boundary nodes associated to a are shown in red. Note that the neighbours of a are nodes in the first layer of the vicinity of a (i.e. connected to a by one edge) regardless of their position in the interior or on the boundary. As a consequence, nodes b_1 and b_4 in Figure 6.1b belong to the set of boundary nodes λ_a^B of a and to the set of interior neighbours Λ_a of a , and they will be used to compute boundary face values as well as the interior face values.

It can be seen on Figure 6.1a that control volumes associated to node a and b_2 are connected via a face whose area vector $\mathbf{C}_\chi^{ab_2}$ is reconstructed using Equation (6.1). A three dimensional visualisation of the reconstructed area vector is shown in Figure 6.2.

6.3 Finite Volume Method

In the Finite Volume Method (FVM), a set of conservation law is integrated on a continuum partitioned in arbitrary control volumes [265, 267] and is often the method of choice in CFD due to its flexibility (it can be applied to many types of complex structured/unstructured grids) and easiness of implementation, and its local discrete conservation properties [30]. It is based on conservative discretisation of conservation laws in integral forms, which ensures the local conservation of solved variables at discrete level.

A set of conservation laws (2.73) is integrated onto an arbitrary volume $\Omega_{\chi,a}$ as

$$\frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi,a}} \mathbf{u} \, dV_{\chi} = - \int_{\partial\Omega_{\chi,a}} \mathcal{F}_{N_{\chi}} \, dA_{\chi} + \int_{\Omega_{\chi,a}} \mathcal{S} \, dV_{\chi}, \quad (6.2)$$

where $\partial\Omega_{\chi,a}$ is the boundary of $\Omega_{\chi,a}$. The combination of all non-overlapping control volumes recover the integral form (2.73) on the whole domain as

$$\sum_a \frac{\partial}{\partial t} \Big|_{\chi} \int_{\Omega_{\chi,a}} \mathbf{u} \, dV_{\chi} = - \sum_a \int_{\partial\Omega_{\chi,a}} \mathcal{F}_{N_{\chi}} \, dA_{\chi} + \sum_a \int_{\Omega_{\chi,a}} \mathcal{S} \, dV_{\chi}, \quad (6.3)$$

and the conservative property requires $\mathcal{F}_{N_{\chi}} = -\mathcal{F}_{-N_{\chi}}$. The average value of \mathbf{u} and \mathcal{S} at $\Omega_{\chi,a}$ is given by

$$\mathbf{u}_a = \frac{1}{\Omega_{\chi,a}} \int_{\Omega_{\chi,a}} \mathbf{u} \, dV; \quad \mathcal{S}_a = \frac{1}{\Omega_{\chi,a}} \int_{\Omega_{\chi,a}} \mathcal{S} \, dV. \quad (6.4a,b)$$

6.3.1 Godunov-type approach

Godunov-type methods are used to give an expression to numerical fluxes and are stated on a Riemann problem, which is composed on a system form (2.73) with discontinuities. These problems are linearised and solved by means of approximate Riemann solvers (e.g. Roe [268–270], Osher [271]) using the eigenstructure of the system (2.73) and designed for less diffusive solutions in the context of discontinuity/wave propagation phenomena. In the discretised form (6.3), the flux term $\left(\mathcal{F}_{N_{\mathbf{x}}}^C\right)_k$ passing through facet k can be expressed, using dual control volumes, as a summation over interior edges together with a sum over boundary faces, as

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{u}_a}{\partial t} \Big|_{\mathbf{x}} = - \left[\sum_{b \in \Lambda_a} \sum_{k \in \Gamma_a^b} \left(\mathcal{F}_{N_{\mathbf{x}}}^C\right)_k A_k + \sum_{\gamma \in \Lambda_a^B} \mathcal{F}_a^\gamma N_{\mathbf{x}}^\gamma \frac{A_\gamma}{3} \right] + \Omega_{\mathbf{x},a} \mathcal{S}_a, \quad (6.5)$$

with $\left(\mathcal{F}_{N_{\mathbf{x}}}^C\right)_k = \mathcal{F}_{N_{\mathbf{x}}}^C(\mathbf{u}_k^-, \mathbf{u}_k^+, N_{\mathbf{x}}^k)$ represents the numerical flux at facet k and based on reconstructed left and right values/states of \mathbf{u} . Mean area vectors (6.1) emerge in Equation (6.5) as

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{u}_a}{\partial t} \Big|_{\mathbf{x}} = - \left[\sum_{b \in \Lambda_a} \left(\mathcal{F}_{N_{\mathbf{x}}}^C\right)_{ab} \|C_{\mathbf{x}}^{ab}\| + \sum_{\gamma \in \Lambda_a^B} \mathcal{F}_a^\gamma C_{\mathbf{x}}^\gamma \right] + \Omega_{\mathbf{x},a} \sum_{b \in \Lambda_a} \mathcal{S}_a, \quad (6.6)$$

with $C_{\mathbf{x}}^\gamma = N_{\mathbf{x}}^\gamma \frac{A_\gamma}{3}$ the area vector at the boundary face γ , and where $\left(\mathcal{F}_{N_{\mathbf{x}}}^C\right)_{ab} = \mathcal{F}_{N_{\mathbf{x}}}^C(\mathbf{u}_{ab}^-, \mathbf{u}_{ab}^+, N_{\mathbf{x}}^{ab})$ is the numerical flux at midpoint of edge ab , expressed in terms of the normal $N_{\mathbf{x}}^{ab}$ associated to this flux, and $\mathbf{u}_{ab}^-, \mathbf{u}_{ab}^+$ the reconstructed left and right values of \mathbf{u} at edge ab (discussed in the next section). Moreover, the contribution of node a for the flux over the three-dimensional surface γ (composed of three nodes a, b, c) will be given by a weighted average stencil [272]

$$\mathcal{F}_a^\gamma = \frac{6\mathcal{F}_a + \mathcal{F}_b + \mathcal{F}_c}{8}. \quad (6.7)$$

To avoid checker-boarding errors³⁷, the expression of flux terms can be obtained through several methodologies. For instance, the Jameson-Schmidt-Turkel (JST) scheme is very popular in CFD and was also implemented for solid mechanics [81, 82]. Moreover, Rhie-Chow [13, 273–275] stabilisation provides a way to detect high-frequency oscillations by adding a term to the divergence of stresses.

The framework presented in this work makes use of a central difference scheme [276], together with numerical dissipation. This artificial dissipation is required to circumvent numerical instabilities and in particular the appearance of numerical modes [267]. In the conservation of linear momentum $\mathbf{p}_{\mathbf{x}}$ and of the material linear momentum $\mathbf{p}_{\mathbf{W}}$, traction terms emerge in the right hand side as the product of a stress-like tensor with a surface norm vector, and are defined as

$$\mathbf{t}_{\mathbf{x}} = \left(\mathbf{P} + \left(\frac{\mathbf{p}_{\mathbf{x}}}{J_{\Psi}} \otimes \mathbf{W} \right) \right) \mathbf{H}_{\Psi} N_{\mathbf{x}}; \quad \mathbf{t}_{\mathbf{W}} = \mathbf{P}_{\mathbf{W}} N_{\mathbf{x}}. \quad (6.8a,b)$$

The contact fluxes of the discretised Riemann problem (6.6) can be generalised to the case where elastic, thermo-mechanical and plastic variables are considered together with the

³⁷Also referred to as zero-energy modes or hour-glassing.

conservation of the mesh motion (presented in Equation (5.3)), as

$$\mathcal{F}_{N_{\chi,ab}}^C = - \begin{bmatrix} t_{\chi,ab}^C \\ \hat{\mathbf{v}}_{ab}^{\text{ave}} \otimes N_{\chi}^{ab} \\ W_{ab}^{\text{ave}} \otimes N_{\chi}^{ab} \\ W_{ab}^{\text{ave}} \cdot H_{\Psi,ab}^{\text{ave}} N_{\chi}^{ab} \\ C_{p,ab}^{-1,\text{ave}} \left(J_{F_{\Psi,a}}^{-1} H_{\Psi,a}^T W_a N_{\chi}^{ab} \right) \\ \bar{\varepsilon}_{p,ab}^{\text{ave}} \left(J_{F_{\Psi,a}}^{-1} H_{\Psi,a}^T W_a N_{\chi}^{ab} \right) \\ 0 \\ H_{\Phi,a}^{\text{ave}} \hat{\mathbf{v}}_{ab}^C \otimes N_{\chi}^{ab} \\ t_{W,ab}^C \end{bmatrix}, \quad (6.9)$$

where the average operator is defined as $[\bullet]_{ab}^{\text{ave}} = \frac{1}{2}([\bullet]_a + [\bullet]_b)$ and corresponds to a central difference scheme. Contact fluxes components in the above equation will therefore be composed of an unstable (average) term and a stabilisation term. It has to be noted that due to the satisfaction of involutions, the fluxes corresponding to the deformation gradients F_{Ψ} and F_{Φ} (second and third components) do not include stabilisation. Moreover, as remarked in [24], the use of a mixed formulation outperforms any stabilisation technique, and thence no stabilisation is required for the plastic internal variables, which is the reason why the 5th and 6th component of the contact flux in Equation (6.9) does only feature averaging³⁸.

6.4 Godunov upwinding stabilisation

Solutions of hyperbolic equations³⁹ can be visualised as propagating waves. Hence in nonlinear systems, discontinuities are generated and/or propagated, and uniqueness of the solution is lost. This corresponds to a violation of the entropy condition presented in Section 4.3.2 and studied in Appendix C.. It is therefore necessary for the numerical framework to include a dissipation term⁴⁰ to single out admissible weak solutions. Upwinding stabilisation (also known as high order correction) has extensively been used with the Godunov approach (see for instance [20, 75]) and will be used in this work. Using this technique enables the numerical framework to recover second-order accuracy, ipso facto circumventing the inf-sup Ladyzhenskaya-Babuška-Brezzi (LBB) condition [84, 283], and it motivates the use of linear tetrahedral elements.

6.4.1 Contact scenario

In this section, the impact between two bodies is considered, where two types of elastic shock waves originate from the contact point and propagate through both bodies. More especially, a frictionless impact will result in a surface traction with no tangential component, and the corresponding wave speeds will travel with a volumetric speed $c_{\chi,p}$. In addition, under infinite friction contact, shock waves will propagate with a shear wave speed $c_{\chi,s}$. In this

³⁸Upwinding stabilisation could be considered also in this case but it is not in the scope of this work.

³⁹Hyperbolic conservation laws are denoted as quasilinear hyperbolic systems in divergence forms (see the work of [30, 214, 277–279] for extended studies). Conservative symmetric hyperbolic equations (in the sense of Einstein-Friedrich [280–282]) describe wave propagation phenomena.

⁴⁰Incorporating the second law of Thermodynamics, or an artificial numerical dissipation, are a classical technique hitherto.

work, normal vectors will be assumed collinear on either surfaces of the interface, as well as tangential vectors. Following the development in the works of Kluth et al. [284] and Maire et al. [95], the normal and tangential operators are applied to the jump of linear momentum (2.57) which, at the interface edge ab , yields, in linear elasticity ($c_{\chi,p} = c_{\chi,p}^+ = c_{\chi,p}^-$ and $c_{\chi,s} = c_{\chi,s}^+ = c_{\chi,s}^-$)

$$c_{\chi,p}\rho_{\chi}(\mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) \llbracket \mathbf{v} \rrbracket_{ab} = -(\mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) \llbracket \mathbf{t}_{\chi,ab} \rrbracket, \quad (6.10a)$$

$$c_{\chi,s}\rho_{\chi}(\mathbf{I} - \mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) \llbracket \mathbf{v} \rrbracket_{ab} = -(\mathbf{I} - \mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) \llbracket \mathbf{t}_{\chi,ab} \rrbracket, \quad (6.10b)$$

where $\rho_{\chi} = \rho_R J_{\Psi}$ is the density projected to the material domain, and where the normals \mathbf{n} are simply a push-forward mapping of the referential normal vectors as

$$\mathbf{n}_{ab} = \frac{\mathbf{F}_{\Phi,ab}^{-T,ave} \mathbf{N}_{\chi}^{ab}}{\|\mathbf{F}_{\Phi,ab}^{-T,ave} \mathbf{N}_{\chi}^{ab}\|}. \quad (6.11)$$

Following the work of [2, 75, 82], linear momentum and traction contact fluxes are expressed as

$$\mathbf{v}_{ab}^C = \mathbf{v}_{ab}^{ave} + \mathbf{v}_{ab}^{stab}; \quad \mathbf{t}_{\chi,ab}^C = \mathbf{t}_{\chi,ab}^{ave} + \mathbf{t}_{\chi,ab}^{stab}. \quad (6.12a,b)$$

The first term on the right hand side in Equations (6.12b) and (6.12a) constitute an unstable flux⁴¹ whose expression is given by a simple central difference scheme as

$$\mathbf{v}_{ab}^{ave} = \frac{1}{2}(\mathbf{v}_a + \mathbf{v}_b); \quad \mathbf{t}_{\chi,ab}^{ave} = \frac{1}{2}(\mathbf{t}_{\chi,a} + \mathbf{t}_{\chi,b}). \quad (6.13a,b)$$

On the other hand, the second term on the right hand side of Equations (6.12b) and (6.12a) constitute the upwinding stabilisation terms and, in the context of a homogeneous material where properties are uniform across the continuum, are defined as

$$\mathbf{v}_{ab}^{stab} = \frac{1}{2\rho_{\chi}^{ave}} \mathbf{S}_v^{ab} (p_{ab}^+ - p_{ab}^-) (\mathbf{H}_{\Psi})_{ab}^{ave} \mathbf{N}_{\chi}^{ab}, \quad (6.14a)$$

$$\mathbf{t}_{\chi,ab}^{stab} = \frac{1}{2} \rho_{\chi,ab}^{ave} \mathbf{S}_t^{ab} (\mathbf{v}_{ab}^+ - \mathbf{v}_{ab}^-), \quad (6.14b)$$

and with (positive definite) stabilisation matrices defined as

$$\mathbf{S}_v^{ab} = \frac{1}{c_{\chi,p}} (\mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) + \frac{1}{c_{\chi,s}} (\mathbf{I} - \mathbf{n}_{ab} \otimes \mathbf{n}_{ab}), \quad (6.15a)$$

$$\mathbf{S}_t^{ab} = c_{\chi,p} (\mathbf{n}_{ab} \otimes \mathbf{n}_{ab}) + c_{\chi,s} (\mathbf{I} - \mathbf{n}_{ab} \otimes \mathbf{n}_{ab}). \quad (6.15b)$$

Equation (6.12b) represents the stabilised flux term in the conservation of linear momentum, and Equation (6.12a) represents the stabilised flux term in the conservation of the deformation gradient. Under the same considerations as in the previous section, the Rankine-Hugoniot condition (5.2) for the conservation of \mathbf{p}_W is projected normally and tangentially. The contact flux is then proposed to be built following the same structure as Equation (6.12b) as

$$\mathbf{t}_{W,ab}^C = \mathbf{t}_{W,ab}^{ave} + \mathbf{t}_{W,ab}^{stab}, \quad (6.16)$$

where the $\mathbf{t}_{W,ab}^{ave}$ is an average value corresponding to the central difference scheme, and where $\mathbf{t}_{W,ab}^{stab}$ is a stabilisation term sharing its structure with the contact flux in Equation (6.15b), which yields

$$\mathbf{t}_{W,ab}^{ave} = \frac{1}{2} (\mathbf{P}_{W,a} + \mathbf{P}_{W,b}) \mathbf{N}_{\chi}^{ab}; \quad \mathbf{t}_{W,ab}^{stab} = \frac{1}{2} (\rho_R)_{ab}^{ave} \mathbf{S}_W^{ab} (\mathbf{W}_{ab}^+ - \mathbf{W}_{ab}^-), \quad (6.17a,b)$$

⁴¹The referred terms can be seen as a FE discretisation of a linear tetrahedral mesh [75, 93].

based on the stabilisation term

$$\mathbf{S}_{\mathbf{W}}^{ab} = c_{\chi,p} \left(\mathbf{N}_{\chi}^{ab} \otimes \mathbf{N}_{\chi}^{ab} \right) + c_{\chi,s} \left(\mathbf{I} - \mathbf{N}_{\chi}^{ab} \otimes \mathbf{N}_{\chi}^{ab} \right). \quad (6.18)$$

A stabilisation procedure is also proposed for the spatial velocity $\hat{\mathbf{v}}$ based on the stabilisation term (6.12a) as

$$\hat{\mathbf{v}}_{ab}^C = \hat{\mathbf{v}}_{ab}^{\text{ave}} + \hat{\mathbf{v}}_{ab}^{\text{stab}}, \quad (6.19)$$

where each term is defined as

$$\hat{\mathbf{v}}_{ab}^{\text{ave}} = \frac{1}{2} (\hat{\mathbf{v}}_a + \hat{\mathbf{v}}_b); \quad \hat{\mathbf{v}}_{ab}^{\text{stab}} = \frac{1}{2\rho_{\chi}^{\text{ave}}} \mathbf{S}_v^{ab} (p_{ab}^+ - p_{ab}^-) \mathbf{H}_{\Phi,ab}^{\text{ave}} \mathbf{N}_{\chi}^{ab}. \quad (6.20a,b)$$

In the context of homogenous linear elasticity, and taking into consideration the expression of the linear momenta $\mathbf{p}_{\chi} = J_{\Psi} \rho_R \mathbf{v}$ and $\mathbf{p}_{\mathbf{W}} = \rho_R \mathbf{W}$ and the contact fluxes $\mathbf{t}_{\chi} = (\mathbf{P} + \mathbf{p}_R \otimes \mathbf{W}) \mathbf{H}_{\Psi}$ and $\mathbf{t}_{\mathbf{W}} = \mathbf{P}_{\mathbf{W}} \mathbf{N}_{\chi}$, the contact fluxes can be stated in the following compact forms

$$\mathbf{p}_{\chi}^C = \frac{1}{2} (\mathbf{p}_{\chi}^+ + \mathbf{p}_{\chi}^-) + \frac{1}{2J_{\Psi}} \mathbf{S}_v^{\text{ave}} (p^+ - p^-) (\mathbf{H})^{\text{ave}} \mathbf{N}_{\chi}, \quad (6.21a)$$

$$\hat{\mathbf{v}}^C = \hat{\mathbf{v}}^{\text{ave}} + \frac{1}{2\rho_{\chi}^{\text{ave}}} \mathbf{S}_v (p^+ - p^-) \mathbf{H}_{\Phi}^{\text{ave}} \mathbf{N}_{\chi}, \quad (6.21b)$$

$$\mathbf{t}_{\chi}^C = [(\mathbf{P} + \mathbf{p}_R \otimes \mathbf{W}) \mathbf{H}_{\Psi}]^{\text{ave}} + \frac{1}{2} \rho_{\chi}^{\text{ave}} \mathbf{S}_t (\mathbf{v}^+ - \mathbf{v}^-), \quad (6.21c)$$

$$\mathbf{t}_{\mathbf{W}}^C = \frac{1}{2} \mathbf{P}_{\mathbf{W}}^{\text{ave}} \mathbf{N}_{\chi} + \frac{1}{2} \rho_R^{\text{ave}} \mathbf{S}_{\mathbf{W}} (\mathbf{W}^+ - \mathbf{W}^-). \quad (6.21d)$$

Equations (6.21a–6.21d) can be integrated in the discretised ALE formulation, and can degenerate to TLF contact fluxes already presented in [2, 75].

6.5 Linear reconstruction

The Godunov method is a monotone linear scheme where the unknown variables are reconstructed as a piecewise constant at the grid elements, and can only be first-order accurate in that regard. Indeed, the diffusivity of the truncation error leads to a downgrade of the spatial and time accuracy to first-order. To overcome these drawbacks, several modifications and extensions to the Godunov method have been developed. One approach is to use adaptive mesh refinement techniques, which can dynamically adjust the grid resolution to focus computational effort where it is most needed. On the other hand, computational effort can be made in the expression of contact fluxes by using higher-order numerical schemes. One can cite the WENO scheme and the MUSCL-Hancock scheme [2, 55], which can reduce numerical oscillations and improve the accuracy of the solution.

In this work, a local piecewise linear reconstruction procedure is performed to obtain the values of the solution at either side of a discontinuity or interface, thereby leading to a second-order accurate solution. Mathematically, it reads [107, 285]

$$\mathcal{U}_{ab}^+ = \mathcal{U}_a + \mathbf{G}_a \cdot \mathbf{d}_{ea}; \quad \mathcal{U}_{ab}^- = \mathcal{U}_b + \mathbf{G}_b \cdot \mathbf{d}_{eb} \quad (6.22a,b)$$

where \mathbf{G}_a is the gradient of \mathcal{U} at node a , and $\mathbf{d}_{ea} = \chi_e - \chi_a$ corresponds, for an edge e composed of nodes a and b , to the distance between the midpoint and the node a .

An advantage of this method is that gradients can all be computed together, prior to the update of primary variables, and used in the expression of the right hand sides. Using the Green-Gauss procedure [266]

$$\mathbf{G}_a = \frac{1}{\Omega_{\chi,a}} \left[\sum_{b \in \Lambda_a} \mathcal{U}_{ab}^{\text{ave}} \mathbf{C}_{\chi}^{ab} + \sum_{\gamma \in \Lambda_a^B} \mathcal{U}_a^{\gamma} \mathbf{C}_{\chi}^{\gamma} \right], \quad (6.23)$$

where \mathbf{u}_a^γ is obtained via the weighted average stencil defined in Equation (6.7).

It is important to note that the above procedure constitutes the prediction step of the Total Variation Diminishing Monotone Upstream Scheme for Conservation Laws technique, or TVD-MUSCL, as originally presented in Sweby [286] and Leer et al. [287], and used in the works of Lee et al. [285]. Because of the potential presence of overshoots and undershoots in the computation of fluxes [2, 30, 263], the predicted gradient operator has to be corrected by employing a slope limiter to respect the maximum principle.

Remark 6.5.1 *In the present work, only minor discontinuities are present, thereby preserving the quality of the reconstituted gradient is good enough and no filter is needed. Nevertheless, the reader is invited to refer to the previous publication of our group [2, 75] for an implementation of the Barth et al. [263] limiter.*

6.6 Discretisation of the mixed formulation

The ALE mixed formulation, stated in Equation (5.13), can now be stated in their semi-discretised form, featuring the unstable and stabilised fluxes. The main equations of the set are spatially discretised as

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{p}_{\mathbf{x},a}}{\partial t} \Big|_{\mathbf{x}} - \left(\sum_{b \in \Lambda_a} t_{\mathbf{x},ab}^C \|\mathbf{C}_{\mathbf{x}}^{ab}\| + \sum_{\gamma \in \Lambda_a^B} t_{\mathbf{x}}^\gamma \|\mathbf{C}_{\mathbf{x}}^\gamma\| + \Omega_{\mathbf{x},a} \mathbf{f}_a \right) = \mathbf{0} \quad (6.24a)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{F}_{\Phi,a}}{\partial t} \Big|_{\mathbf{x}} - \left(\sum_{b \in \Lambda_a} \hat{\mathbf{v}}_{ab}^{\text{ave}} \otimes \mathbf{C}_{\mathbf{x}}^{ab} + \sum_{\gamma \in \Lambda_a^B} \hat{\mathbf{v}}^\gamma \otimes \mathbf{C}_{\mathbf{x}}^\gamma \right) = \mathbf{0} \quad (6.24b)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{F}_{\Psi,a}}{\partial t} \Big|_{\mathbf{x}} - \left(\sum_{b \in \Lambda_a} \mathbf{W}_{ab}^{\text{ave}} \otimes \mathbf{C}_{\mathbf{x}}^{ab} + \sum_{\gamma \in \Lambda_a^B} \mathbf{W}^\gamma \otimes \mathbf{C}_{\mathbf{x}}^\gamma \right) = \mathbf{0} \quad (6.24c)$$

$$\Omega_{\mathbf{x},a} \frac{\partial J_{\Psi,a}}{\partial t} \Big|_{\mathbf{x}} - \left(\sum_{b \in \Lambda_a} (\mathbf{H}_{\Psi}^T \mathbf{W})_{ab}^{\text{ave}} \cdot \mathbf{C}_{\mathbf{x}}^{ab} \right) = 0 \quad (6.24d)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{C}_{p,a}^{-1}}{\partial t} \Big|_{\mathbf{x}} = \sum_{b \in \Lambda_a} (\mathbf{C}_p^{-1})_{ab}^{\text{ave}} \left[\mathbf{F}_{\Psi,a}^{-1} \mathbf{W}_a \right] \cdot \mathbf{C}_{\mathbf{x}}^{ab} + \Omega_{\mathbf{x},a} \left(-2\dot{\gamma} \mathbf{F}^{-1} \frac{\partial \phi_f}{\partial \boldsymbol{\tau}} \mathbf{b}_e \mathbf{F}^{-T} \right)_a \quad (6.24e)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \bar{\varepsilon}_{p,a}}{\partial t} \Big|_{\mathbf{x}} = \sum_{b \in \Lambda_a} (\bar{\varepsilon}_p)_{ab}^{\text{ave}} \left[\mathbf{F}_{\Psi,a}^{-1} \mathbf{W}_a \right] \cdot \mathbf{C}_{\mathbf{x}}^{ab} + \Omega_{\mathbf{x},a} \dot{\gamma}_a \quad (6.24f)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \eta_a}{\partial t} \Big|_{\mathbf{x}} + \sum_{b \in \Lambda_a} \left[\frac{\mathbf{Q}_{\mathbf{x},ab}^{\text{ave}}}{\theta_a} \cdot \mathbf{C}_{\mathbf{x}}^{\text{ave}} \right] = \sum_{b \in \Lambda_a} \eta_{ab}^{\text{ave}} \left[\mathbf{F}_{\Psi,a}^{-1} \mathbf{W}_a \right] \cdot \mathbf{C}_{\mathbf{x}}^{ab} + \frac{\dot{D}_{\text{int},a}}{\theta_a} \quad (6.24g)$$

$$\Omega_{\mathbf{x},a} \frac{\partial J_{\Phi,a}}{\partial t} \Big|_{\mathbf{x}} - \left(\mathbf{H}_{\Phi,a} : \left[\sum_{b \in \Lambda_a} \hat{\mathbf{v}}_{ab}^{\text{ave}} \otimes \mathbf{C}_{\mathbf{x}}^{ab} + \sum_{\gamma \in \Lambda_a^B} \hat{\mathbf{v}}^\gamma \otimes \mathbf{C}_{\mathbf{x}}^\gamma \right] + \sum_{b \in \Lambda_a} \hat{\mathbf{v}}_{ab}^{\text{stab}} \cdot \mathbf{C}_{\mathbf{x}}^{ab} \right) = 0 \quad (6.24h)$$

$$\Omega_{\mathbf{x},a} \frac{\partial \mathbf{p}_{\mathbf{W},a}}{\partial t} \Big|_{\mathbf{x}} - \left(\sum_{b \in \Lambda_a} t_{\mathbf{W},ab}^C \|\mathbf{C}_{\mathbf{x}}^{ab}\| + \sum_{\gamma \in \Lambda_a^\gamma} t_{\mathbf{W}}^\gamma \|\mathbf{C}_{\mathbf{x}}^\gamma\| - \nu_{\Psi,a} \mathbf{p}_{\mathbf{W},a} \right) = \mathbf{0}. \quad (6.24i)$$

As is was shown in this section, the selected approach for solving the mixed formulation is to correct the interface numerical fluxes to ensure non-negative numerical entropy production. In plasticity equations, the dominance in magnitude of the plasticity term over the ALE correction term motivated the use of a non-conservative form (see transformation in Equation (2.44)). Furthermore, the plasticity equations mentioned earlier are solved implicitly. This choice was made for ease of integration within the proposed ALE framework, which has the capability to degenerate into a Total Lagrangian approach. Different implementations for these equations can be considered; the use of an explicit integration is possible but led to substantial numerical instabilities in this work. Additionally, it is important to mention that while the plasticity equations in the mixed formulation (6.24) are presented in non-conservative form, it is crucial that their structure is coherent with the conservation of material Jacobian (6.24d). That is to say, they must be all implemented in either a conservative or a non-conservative form in the framework.

Together with the above discretised system, the conservation equation for the plastic dissipation, stated in Equation (4.22), is also required in the context of plasticity. Its spatially discretised form is

$$\Omega_{\mathcal{X},a} \left. \frac{\partial w_{p,a}}{\partial t} \right|_{\mathcal{X}} = \sum_{b \in \Lambda_a} (w_p)_{ab}^{\text{ave}} \left[J_{\mathbf{F}_{\Psi,a}}^{-1} \mathbf{H}_{\Psi,a}^T \mathbf{W}_a \right] \cdot \mathbf{C}_{\mathcal{X}}^{ab} + \Omega_{\mathcal{X},a} \dot{D}_{\text{int},a}. \quad (6.25)$$

In the above equation, the second term of the right-hand side is computed within the Lagrangian plasticity update, and the first term on the right-hand side corresponds to the ALE component responsible for adjusting the value after mesh motion. In a Total Lagrangian context, the latter term vanishes and the update of the plastic dissipation is fully conducted in the plasticity update. It must be emphasised that Equation (6.25) is redundant and is exclusively used for plotting purposes. In thermo-mechanics, and to account for possibly strong thermally-coupled scenarios, the spatially discretised system (6.24) was provided with the first law of thermodynamics written in terms of the entropy η . Alternatively, that system can instead be supplemented with the conservation of the scaled total energy $E_{\mathcal{X}}$ [21, 22]. In discretised form, it reads

$$\begin{aligned} \Omega_{\mathcal{X},a} \left. \frac{\partial E_{\mathcal{X},a}}{\partial t} \right|_{\mathcal{X}} &= \sum_{b \in \Lambda_a} \left[\mathbf{v}_{ab}^C \cdot \mathbf{t}_{\mathcal{X},ab}^{\text{ave}} \|\mathbf{C}_{\mathcal{X}}^{ab}\| + \left([E_R \mathbf{H}_{\Psi}^T \mathbf{W}]_{ab}^{\text{ave}} - \mathbf{Q}_{\mathcal{X},ab}^{\text{ave}} \right) \cdot \mathbf{C}_{\mathcal{X}}^{ab} \right] \\ &\quad + \sum_{b \in \Lambda_a^B} \mathbf{v}^\gamma \cdot \mathbf{t}_{\mathcal{X}}^\gamma \|\mathbf{C}_{\mathcal{X}}^\gamma\| + \Omega_{\mathcal{X},a} (\mathbf{f}_a \cdot \mathbf{v}_a + s_{\mathcal{X},a}). \end{aligned} \quad (6.26)$$

The discretisation of the flux term in the balance of linear momentum (2.54) is given in Equation (6.24a) using the linear momentum contact flux (6.12b). While upwinding stabilisation can be inserted in the discrete conservation of geometric mappings⁴² [56], this work will only make use of upwinding stabilisation for the spatial volume map $J_{\mathbf{F}}$. No numerical stabilisation is introduced in the discretisation of the material (6.24c) and spatial (6.24b) deformation gradients. The discretisation of the conservation of η [22] is given in Equation (6.24g). The discretisation of the material velocity \mathbf{W} is given in Equation (6.24i). Following the semi-discrete version of the classical Coleman-Noll⁴³ procedure [212, 224], the stabilisation terms in Equations (6.15a), (6.15b) and (6.18) ensure the production of numerical entropy.

⁴²Although upwinding stabilisation can be given to the conservation equation of the deformation gradient and its cofactor, those mappings are naturally curl-free and do not require stabilisation [56, 75].

⁴³The Coleman-Noll procedure is presented for the system at a continuum level in Appendix C.

6.6.1 Boundary conditions

It is clear from Equations (6.12) that expressions need to be provided for the linear momentum and the traction on the boundaries⁴⁴. Moreover, the importance of a symmetric boundary condition for the material velocity \mathbf{W} was highlighted in Chapter 5. Flux terms require enforcement of their value on boundary faces (see Table 6.1), whereas spatial and material linear momenta can be prescribed on the boundary faces or boundary nodes (see Tables 6.1 and 6.2). This is due to the fact that \mathbf{p}_χ and \mathbf{W} are both appearing in a flux term, and are conserved quantities.

The interface along the boundary is such that the value in the interior corresponds to the negative superscript value and the value in the outer domain corresponds to the positive superscript. Consequently, the different boundary conditions can be brought out by applying proper values in the outer domain for contact fluxes (6.21a–6.21d).

Moving surface

A boundary that is considered moving at a velocity \mathbf{v}_B can be mathematically stated, for a face γ , as

$$\mathbf{v}_\gamma^+ = \mathbf{v}_B. \quad (6.27)$$

Note that in the case of a fixed boundary, also denoted as sticking surface (wall or no-slip condition in CFD), the velocity at the boundary is restricted to $\mathbf{0}$ and the above Equation becomes $\mathbf{p}_{\chi,\gamma} = \mathbf{0}$. As a consequence, no local deformation is allowed in any direction and the wave speeds tend to infinity as

$$c_{p,\chi}^+ = c_{s,\chi}^+ = \infty. \quad (6.28)$$

Substituting above equations in Equations (6.21a) and (6.21c), it yields

$$\mathbf{v}^C = \mathbf{v}_B; \quad \mathbf{t}_\chi^C = \mathbf{t}_\chi^- + \rho_\chi \mathbf{S}_v (\mathbf{v}_B - \mathbf{v}^-). \quad (6.29a,b)$$

Traction on a free surface

When a traction \mathbf{t}_B is applied on a surface, it can be expressed as

$$\mathbf{t}_{\chi,\gamma}^+ = \mathbf{t}_B, \quad (6.30)$$

and the wave speeds can be expressed as [2]

$$c_{p,\chi}^+ = c_{s,\chi}^+ = 0. \quad (6.31)$$

Using the same substitution process, the contact fluxes are

$$\mathbf{v}^C = \mathbf{v}^- + \frac{1}{\rho_{\chi,\gamma}} \mathbf{S}_t (\mathbf{t}_B - \mathbf{t}^-); \quad \mathbf{t}_\chi^C = \mathbf{t}_B. \quad (6.32a,b)$$

Symmetric surface

When a surface (denoted as sliding or symmetric) is considered a roller support, the velocity is only allowed to slide along the boundary and is therefore restricted to its tangential component, which can be stated as

$$\mathbf{v}_\gamma^+ \cdot \mathbf{n} = 0. \quad (6.33)$$

⁴⁴Boundary conditions can be denoted as weak and strong when they are respectively applied on fluxes and on nodes). The traction is only weakly imposed while the linear momenta is weakly and/or strongly imposed.

	$\mathbf{t}_{\mathcal{X},a}^\gamma$	\mathbf{v}_a^γ	\mathbf{W}_a^γ
Traction	\mathbf{t}_B^γ	\mathbf{v}_a^γ	
Moving	$\mathbf{t}_{\mathcal{X},a}^\gamma$	\mathbf{v}_B^γ	$(\mathbf{I} - \mathbf{N}_{\mathcal{X}}^\gamma \otimes \mathbf{N}_{\mathcal{X}}^\gamma) \mathbf{W}_a^\gamma$
Symmetric	$(\mathbf{N}_{\mathcal{X}}^\gamma \otimes \mathbf{N}_{\mathcal{X}}^\gamma) \mathbf{t}_{\mathcal{X},a}^\gamma$	$(\mathbf{I} - \mathbf{N}_{\mathcal{X}}^\gamma \otimes \mathbf{N}_{\mathcal{X}}^\gamma) \mathbf{v}_a$	
Anti-symmetric	$(\mathbf{I} - \mathbf{N}_{\mathcal{X}}^\gamma \otimes \mathbf{N}_{\mathcal{X}}^\gamma) \mathbf{t}_{\mathcal{X},a}^\gamma$	$(\mathbf{N}_{\mathcal{X}}^\gamma \otimes \mathbf{N}_{\mathcal{X}}^\gamma) \mathbf{v}_a$	

TABLE 6.1: Facial contribution to a boundary node a for a boundary condition on spatial and material quantities.

	\mathbf{v}_a	\mathbf{W}_a
Traction	\mathbf{v}_a	
Moving	\mathbf{v}_B	$(\mathbf{I} - \mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \mathbf{W}_a$
Symmetric	$(\mathbf{I} - \mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \mathbf{v}_a$	
Anti-symmetric	$(\mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \mathbf{v}_a$	

TABLE 6.2: Nodal contribution to a boundary node a for a boundary condition on spatial and material velocities.

and consequently,

$$c_{p,\mathcal{X}}^+ \equiv \infty; \quad c_{s,\infty}^+ = 0. \quad (6.34a,b)$$

Moreover, if a traction \mathbf{t}_B is applied on that boundary, Equation (6.30) holds and contact fluxes become

$$\mathbf{v}^C = (\mathbf{I} - \mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \left[\mathbf{v}^- + \frac{1}{\rho_{\mathcal{X},\gamma} c_{s,\mathcal{X}}^-} (\mathbf{t}_B - \mathbf{t}_{\mathcal{X}}^-) \right]; \quad (6.35a)$$

$$\mathbf{t}_{\mathcal{X}}^C = (\mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) [\mathbf{t}^- - u_{p,\mathcal{X}}^- \mathbf{p}_{\mathcal{X}}^-] + (\mathbf{I} - \mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \mathbf{t}_B. \quad (6.35b)$$

Skew-symmetric surface

In the case of a skew-symmetric surface, the velocity is restricted to its normal component

$$\mathbf{v}_\gamma^+ = \mathbf{p}_\gamma^+ \cdot \mathbf{n}, \quad (6.36)$$

and as a result

$$c_{p,\mathcal{X}}^+ = 0; \quad c_{s,\infty}^+ \equiv \infty. \quad (6.37a,b)$$

Contact fluxes are defined as

$$\mathbf{v}^C = (\mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \left[\mathbf{v}^- + \frac{1}{\rho_{\mathcal{X},\gamma} c_{s,\mathcal{X}}^-} (\mathbf{t}_B - \mathbf{t}_{\mathcal{X}}^-) \right] \quad (6.38a)$$

$$\mathbf{t}_{\mathcal{X}}^C = (\mathbf{I} - \mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) [\mathbf{t}^- - u_{p,\mathcal{X}}^- \mathbf{p}_{\mathcal{X}}^-] + (\mathbf{N}_{\mathcal{X}} \otimes \mathbf{N}_{\mathcal{X}}) \mathbf{t}_B. \quad (6.38b)$$

Note that in Table 6.1, the velocity \mathbf{v} is unaltered along free boundaries while the traction is unaltered along fixed (or clamped) boundaries. Moreover, the value of those two quantities is only partially altered on symmetric and anti-symmetric by the appropriate normal or tangential operator. The base value of those two quantities, at face γ , is given by the

weighted average stencil defined in Equation (6.12). A free boundary is a boundary where the prescribed traction $\mathbf{t}_{\mathcal{X}}^B = \mathbf{0}$. A fixed boundary is a moving boundary with velocity $\mathbf{v}^B = 0$. Boundary conditions can be strongly applied at boundary nodes, as presented in Table 6.2.

6.7 Time-marching scheme

To preserve second-order accuracy offered by the upwinding stabilisation, the time-marching scheme in this work will rely on an high-order explicit time integrator. This Section presents the Total Variation Diminishing one step multistage Runge-Kutta method (TVD-RK) [288, 289] used in this work for the conservation variables \mathcal{U} stated in Equation (5.13) and the geometric material \mathbf{x}_{Ψ} and spatial \mathbf{x}_{Φ} positions vector, which is very well suited for fast-transient problems where small time increments may have to be used to accurately capture physical phenomena such as plasticity yielding and nearly incompressible materials undergoing important load. This method has already been explored in [2, 35, 56, 75, 81, 107, 290, 291] and is overall a popular choice due to its high accuracy and implementation. The 2-stage Runge-Kutta time integration for a variable \mathcal{U} is defined as

$$\mathbf{u}_a^* = \mathbf{u}_a^n + \Delta t \dot{\mathbf{u}}_a^n(\mathbf{u}_a^n), \quad (6.39a)$$

$$\mathbf{u}_a^{n+1} = \frac{1}{2}\mathbf{u}_a^n + \frac{1}{2}\left(\mathbf{u}_a^* + \Delta t \dot{\mathbf{u}}_a^*(\mathbf{u}_a^*)\right), \quad (6.39b)$$

where \mathbf{u}^* represents an intermediate state, or stage, computed from the solution \mathbf{u}^n at timestep n , used to obtain the result at timestep $n + 1$. Note that in the case of the 2-stage Runge-Kutta method, Equation (6.39a) corresponds to a classical forward Euler time integration. Additionally, the 3-stage Runge-Kutta time integration reads

$$\mathbf{u}_a^* = \mathbf{u}_a^n + \Delta t \dot{\mathbf{u}}_a^n(\mathbf{u}_a^n), \quad (6.40a)$$

$$\mathbf{u}_a^{**} = \frac{3}{4}\mathbf{u}_a^n + \frac{1}{4}\left(\mathbf{u}_a^* + \Delta t \dot{\mathbf{u}}_a^*(\mathbf{u}_a^*)\right), \quad (6.40b)$$

$$\mathbf{u}_a^{n+1} = \frac{1}{3}\mathbf{u}_a^n + \frac{2}{3}\left(\mathbf{u}_a^{**} + \Delta t \dot{\mathbf{u}}_a^{**}(\mathbf{u}_a^{**})\right), \quad (6.40c)$$

where \mathbf{u}_a^* and \mathbf{u}_a^{**} are two intermediate stages to advance from timestep n to timestep $n + 1$. For a given stage, These integration procedures involve a weighted average both the solution \mathbf{u}^n at original timestep n , and the sum of the solution at the previous stage⁴⁵ and their corresponding right hand side. Note that this RK integrator can easily reduce to the forward Euler method consisting of only one stage, expressed as

$$\mathbf{u}_a^{n+1} = \mathbf{u}_a^n + \Delta t \dot{\mathbf{u}}_a^n(\mathbf{u}_a^n). \quad (6.41)$$

The computation of the time increment Δt is based on a CFL number $0 < \alpha_{\text{CFL}} < 1$ and is defined as

$$\Delta t = \alpha_{\text{CFL}} \min \left(\frac{h\lambda_{\Phi}}{c_{\mathcal{X},p}} \right), \quad (6.42)$$

where h is the minimum edge length in the reference mesh, $c_{\mathcal{X},p}$ is the p -wave speed, and λ_{Φ} is the spatial stretch obtained as the minimum eigenvalue of the spatial right Cauchy-Green tensor \mathbf{C}_{Φ} at node a . This condition, whose bounded values are found in the Von-Neumann stability analysis, is crucial for explicit time integrators as it is designed to ensure numerical stability, henceforth avoiding the appearance and propagation of spurious oscillations.

Eventually, the update of the non-linear ALE wave speeds $c_{\mathcal{X},p}$ and $c_{\mathcal{X},s}$ is established in Equation (3.33).

⁴⁵Stage 0 corresponds to the solution at timestep n .

6.7.1 Implicit source terms

In the discrete set of ALE equations in Section 6.6, the terms were arranged so that the left hand side is composed of the conserved variable and the term that is explicitly computed in every RK iteration. That term corresponds to the convective component of each equation, which is represented by the divergence operator in the continuum. On the other hand, the right hand side is composed of the terms that are treated as source for each equation. There are source terms for the plastic internal variables, the entropy equation and the evolution of the plastic dissipation. They are treated as implicit terms in the algorithm. For example, the conservation of plastic internal variables can be fully discretised as

$$\begin{aligned} (\mathbf{C}_p^{-1})_a^{n+1} &= (\mathbf{C}_p^{-1})_a^n + \Delta t \sum_{b \in \Lambda_a} (\mathbf{C}_p^{-1})_{ab}^{\text{ave},n} \left([(\mathbf{F}_\Psi^{-1})_a^{n+1} \mathbf{W}_a^{n+1}] \cdot \mathbf{C}_\chi^{ab} \right) \\ &\quad - 2\Delta t \dot{\gamma}_a^{n+1} (\mathbf{F}^{-1})_a^{n+1} \frac{\partial (\phi_f)_a^{n+1}}{\partial \boldsymbol{\tau}_a^{n+1}} (\mathbf{b}_e)_a^{n+1} (\mathbf{F}^{-T})_a^{n+1} \end{aligned} \quad (6.43a)$$

$$(\bar{\varepsilon}_p)_a^{n+1} = (\bar{\varepsilon}_p)_a^n + \Delta t \sum_{b \in \Lambda_a} \left[(\bar{\varepsilon}_p)_{ab}^{\text{ave},n} \left([(\mathbf{F}_\Psi^{-1})_a^{n+1} \mathbf{W}_a^{n+1}] \cdot \mathbf{C}_\chi^{ab} \right) + \dot{\gamma}_a^{n+1} \right]. \quad (6.43b)$$

For post-processing purposes, the material Ψ and spatial Φ mappings can be obtained by integrating their respective momenta, whose balance is solved explicitly in this framework. That integration is herein performed by means of a backward Euler method. In the case of a Total Lagrangian framework (that is, $\mathbf{W} = 0$), the time update of the plasticity model exclusively relies on the predictor step (which corresponds to the Lagrangian phase). This update for plastic variables follows the implicit Backward Euler method, a technique well-established in earlier works [2, 81, 217].

6.8 Conclusion

This Chapter introduced the spatial and time discretisation methodologies used in this work for the implementation of the numerical framework. The domain is discretised following the Vertex-Centred Finite Volume Method (VCFVM). It is primarily characterised by a dual mesh obtained via median tessellation, as described in Section 6.2. Then, the notation and the discretised ALE equation were presented in 6.3. Contact fluxes were conferred therein, and in the context of this work, the Godunov method with upwind stabilisation was used in Chapter 6.4 to compute those fluxes and recover second-order accuracy. This methodology requires the reconstruction of left and right values of quantities at the interface between two cells, which is taken care of by a linear reconstruction procedure based on the nodal gradient operator and shown in Section 6.5. All the previous considerations were incorporated in the ALE framework in Section 6.6, where the semi-discretised form of every conservation law of the ALE mixed system is presented. The boundary values require special treatment that is featured in Section 6.6.1. Eventually, Section 6.7 presents the methodology used to integrate the equations in time, using the TVD-RK schemes. In the next Chapter, the implementation of the present ALE framework in a dedicated OpenFOAM package will be presented.

CHAPTER 7  OPENFOAM IMPLEMENTATION

*“Theory and practice sometimes clash. And when that happens,
theory loses. Every single time.”*

— LINUS TORVALD, 2009

7.1 Introduction

OpenFOAM⁴⁶ (Open-source Field Operation And Manipulation) [39, 75, 292] is an open-source software C++ package originally designed for computational fluid dynamics (CFD) that is widely used in a variety of industries. It is commonly used in

- aerospace engineering, for the simulation of aircraft and rocket propulsion systems, as well as for the design of aircraft and spacecraft components,
- the automotive industry, for the simulation of internal combustion engines, exhaust systems, and cooling systems,
- in chemical engineering, for the simulation of chemical reactions, mixing, and heat transfer in industrial processes,
- biomedical engineering, for simulation blood flows in the human cardiovascular system,
- civil and marine engineering, electrical engineering, mechanical engineering and more.

OpenFOAM is mainly based on the Cell-Centred Finite Volume Method (CCFVM) and provides a wide range of solvers (e.g. PISO, SIMPLE, PIMPLE) for simulating different types of fluid flows, combustion, heat transfers including laminar, turbulent, and multiphase flows, and many more. Its parallel computing component is based on the Message Passing Interface (MPI). The software also includes a wide range of utility programs for pre- and post-processing, including data visualization, mesh generation, and data conversion. OpenFOAM software also comes with a extensive list of quality requirements⁴⁷ such as modularity, design patterns [293], maintainability and easy deployment.

The selection of the OpenFOAM platform as the primary tool for generating results in this work is well-founded, owing to its myriad advantages in CFD. An essential strength of OpenFOAM lies in its status as an open-source solution, which enables the scientific community to scrutinize, modify, and enhance the underlying algorithms and numerical schemes, thus fostering a collaborative and transparent research environment. OpenFOAM's active and extensive user community contributes to its continual development and enhancement. Frequent updates and improvements ensure that the software remains at the forefront of computational dynamics research.

Moreover, OpenFOAM exhibits inherent flexibility and adaptability, rooted in its object-oriented architecture. This feature facilitates seamless customization and extensibility, enabling researchers to tailor the software to their specific research requirements. Consequently, complex and novel CFD simulations are regularly implemented, while maintaining a high degree of numerical accuracy and reliability. The software's underlying numerical methods and solvers are well-vetted, demonstrating robustness and accuracy across a diverse range of fluid dynamics problems. Its support for various turbulence models, discretization schemes, and boundary conditions further accentuates its proficiency in capturing intricate flow phenomena encountered in practical engineering applications. Its validation against benchmark experiments and industrial data instils confidence in its applicability to real-world engineering challenges. Additionally, OpenFOAM boasts a comprehensive suite of pre-processing and post-processing utilities, facilitating the efficient setup of simulation cases and the subsequent analysis and visualization of results. For all these aforementioned reasons, OpenFOAM appeared as a tool of choice⁴⁸ to implement the ALE formulation.

This chapter will present the solver made from scratch that implements the novel ALE formulation for fast solid dynamics, as well as the code workflow.

⁴⁶Further information about the OpenFOAM platform can be viewed in [48, 66, 97, 292].

⁴⁷See OpenFOAM Foundation's Code quality guide at <https://openfoam.org/dev/code-quality/>.

⁴⁸The benefits of using OpenFOAM are in [75].

7.1.1 OpenFOAM versions

It is important to keep in mind that OpenFOAM is a trademark related to an open-source initiative. It has therefore undergone many forks in its extensive lifetime. In other words, several projects have been releasing their own version of the OpenFOAM code package. At the time of this work, the most outstanding contributors are ESI's OpenCFD⁴⁹, the OpenFOAM Foundation⁵⁰, and the foam-extend project. This work is based on OpenFOAM-6, the version of the OpenFOAM Foundation that came out in August 2018, and is compatible with OpenFOAM v1806, the version of OpenCFD that was released in June 2018.

7.2 Simulation workflow

In the following section, the code structure is presented, as well as the setup of a numerical experiment and the simulation workflow. A focus will be put on the algorithmic description of the solver, and specifically how the different numerical considerations mentioned in the previous chapter are used.

7.2.1 Code structure

The solver files are organised following OpenFOAM's standards with the use of header files (*.H* files), and is represented in Figure 7.1. The structure comes with 3 main elements: python scripts to manipulate data (in *scripts* folder), *initialConditions* (in *utility* folder) to set up simple or complex initial distributions, and the solver's binary file (*vcALEFoam*, in the eponym folder). The latter contains files representing all the necessary steps for the algorithm⁵¹. The content of the solver's main file can be viewed in Listing 7.1. The

```

1  /*-----*\
2
3  \ \ / / F i e l d           |   OpenFOAM: The Open Source CFD Toolbox
4  \ / \ / O peration        |   Website:  https://openfoam.org
5  / \ / \ A n d             |   Copyright (C) 2011-2018 OpenFOAM Foundation
6  /___\ M anipulation       |
7  -----*/
8
9  License
10   This file is part of OpenFOAM.
11
12   OpenFOAM is free software: you can redistribute it and/or modify it
13   under the terms of the GNU General Public License as published by
14   the Free Software Foundation, either version 3 of the License, or
15   (at your option) any later version.
16
17   OpenFOAM is distributed in the hope that it will be useful, but WITHOUT
18   ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
19   FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License
20   for more details.
21
22   You should have received a copy of the GNU General Public License
23   along with OpenFOAM. If not, see <http://www.gnu.org/licenses/>.
24
25  Application
26   vcALEFoam
27
28  Description
29   Main Algorithm
30
31  \*-----*/
32
33  #include "fvCFD.H"
34  #include "solidModel.H"
35  #include "mechanics.H"
36  #include "operations.H"
37  #include "aleModel.H"
38
39  #include <chrono>
40  #include "side.H"
41  #include "labelListIOList.H"
42  #include "vectorListIOList.H"

```

⁴⁹OpenCFD's version is also denoted as openfoam.com.

⁵⁰The OpenFOAM Foundation's version is also denoted as openfoam.org.

⁵¹The header files included before the main function are necessary for the solver to recognise OpenFOAM objects as well as the custom libraries.

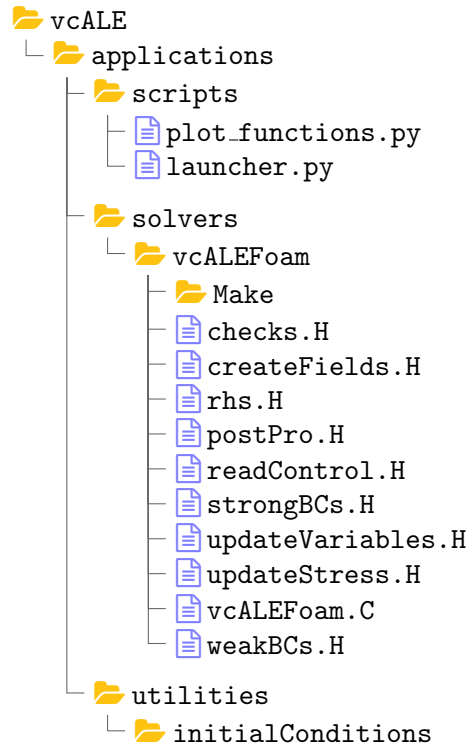


FIGURE 7.1: Solver structure: files related to the main solver; files related to the compilation are omitted.

```

43 #include "gradientSchemes.H"
44
45 // #include "primitivePatchInterpolation.H"
46 // #include "symmetricPatchPointPatchField.H"
47
48 /* * * * * * */
49 int main(int argc, char *argv[]){
50     #include "setRootCase.H"
51     #include "createTime.H"
52     #include "createMesh.H"
53     #include "readControls.H"
54     #include "createFields.H"
55
56     while (runTime.loop()) { // time loop
57         /* Compute time values * * * * * */
58         stretch = mech.stretch(spatF);    minStretch = gMin(stretch);
59         if (timeStepping == "variable") {
60             deltaT = mech.getNewDT(stretch, x, spatF, model.Up().value(), cfl);
61             runTime.setDeltaT(deltaT);
62         }
63         Up = model.Up().value()/minStretch; Us = model.Us().value()/minStretch;
64         t += deltaT;    tstep++;
65         Info << nl << "Time_Step_" << tstep << nl
66             << "deltaT_" << deltaT.value() << "s" << nl
67             << "Time_" << t.value() << "s" << nl;
68
69         /* check for mesh motion (isALE) * * * * * */
70         meshUpdate = meshUpdateCheck(model, aleModel_, minStretch);
71
72         #include "saveOldTime.H" /* save old variables * * * * * */
73
74         for(RKstage=0; RKstage<nbRKstages; RKstage++) { /* RK loop * * * * * */
75             isLastStage = (RKstage==(nbRKstages-1));
76             #include "gradients.H" // Compute gradients
77             #include "rhs.H" // Compute interior of RHS
78
79             #include "calculFlux.H"
80
81             #include "weakBCs.H" // Compute boundary of RHS
82
83             #include "integrator.H" // Advance conserved vars
84
85             #include "updateVariables.H" // correct ALE and F,H,J,v,vHat
86             #include "strongBCs.H" // applys BCs
87             #include "updateStress.H" // compute Piola tensors
88         } /* end RK loop * * * * * */
89
90         if ((not usePstar) and meshUpdate) aleModel_.correct();
91         W = aleModel_.wDot_noref();
92         #include "backwardEuler.H" // only for source terms
  
```

```

93
94     sprintf(percentageCompleted, "%.3f", (t.value()/endTime)*100.);
95     Info << "Simulation completed" << percentageCompleted << "%\n"
96         << "ClockTime=" << runTime.elapsedClockTime() << "s" << nl;
97
98     //if (FSI){
99         //u = x - X;
100        //updateCellDisp(cellDisp, u);
101        //}
102
103        if (runTime.writeTime() {
104            #include "postPro.H"
105        }
106
107    } // end time loop
108
109    std::time_t end_time = getDatetime();
110    Info << "\nExecutionTime=" << runTime.elapsedCpuTime() << "\ns"
111        << "\nClockTime=" << runTime.elapsedClockTime() << "\ns"
112        << "\nEndTime=" << std::ctime(&end_time) ;
113    return EXIT_SUCCESS;
114 } /* ** ** ** **

```

Listing 7.1: vcALEFoam.C

```

1     //if (isThermo()) p_ -= (cv_ * Gamma0_) * (theta - thetaR_psf());//
2     if (isThermo()) basicThermo.correctPressure(p_, theta);
3     P_ = (mu_ * op.pow(J_, -2./3.) * F_) // SigmaF
4         - ( ((mu_/3.)*op.pow(J_, -5./3.)) * ((F_&&F_) * H_) ) // SigmaJ' H
5         + (p_*H_);
6
7     } else if ( elasticityModel.Deviatoric().getName() == "linearelastic" ){ //(model_ == "linearelastic") {
8         p_ = kappa_ * (J_ - op.ID()); // wrong but unused (1/3 of cauchy)
9         P_ = (mu_*(F_ + F_.T()) - 2*op.IDtensor())
10            + (lambda_ * ((tr(F_) - 3*op.ID()) * op.IDtensor()))
11        ;
12    } else
13        /*
14        FatalErrorInFunction << "error_solidModel::correct()" << abort(FatalError);
15    }
16
17    // ** ** **
18
19    void solidModel::correct4(
20        pointTensorField& P,           // Piola tensor
21        pointScalarField& p,           // pressure
22        const pointTensorField& F_,     // natural F
23        const pointScalarField& newJ_,  // natural J
24        pointTensorField& CpInv,       // inverse of plastic right Cauchy
25        pointScalarField& strain_p,    // plastic strain
26        const pointScalarField& T,     // temperature
27        pointScalarField& eps_dot,     // time rate of plastic strain
28        pointScalarField& coeffTau,   // coeff to scale tauDev
29        pointScalarField& vMises_,    // von Mises stress
30        pointScalarField& intDEV,
31        pointScalarField& intVOL,
32        bool isLastStage,             // is it the last RK stage?
33        const scalar& dt
34    ){
35        const scalar sqrt15 = Foam::sqrt(1.5);
36        const scalar mu = mu_.value();
37        const pointScalarField JF = Foam::det(F_);
38        const pointTensorField Finv = Foam::inv(F_).ref();
39        p = elasticityModel.Volumetric().p(newJ_);
40        if (isThermo()) basicThermo.correctPressure(p, T); //p-=cvGamma0(T-TR);
41
42        forAll(mesh_.points(), n){
43            /* Jacobi Spectral decomposition ** ** ** */
44            tensor be = (F_[n] & CpInv[n]) & F_[n].T();
45            tensor eVec = op.eigenS_sqrt(be, eStretch[n]); // eigenvect and sqrt eigenval
46            const tensor n0xn0 = (vector(1,0,0) & eVec) * (vector(1,0,0) & eVec); //n0*n0;
47            const tensor n1xn1 = (vector(0,1,0) & eVec) * (vector(0,1,0) & eVec); //n1*n1;
48            const tensor n2xn2 = (vector(0,0,1) & eVec) * (vector(0,0,1) & eVec); //n2*n2;
49
50            /* Trial Deviatoric Kirchoff Stress Vector ** ** ** */
51            vector tauDev = 2.* mu * op.log(Foam::pow(JF[n], -1./3.) * eStretch[n]);
52            const scalar normTauDev = Foam::sqrt(tauDev & tauDev);
53
54            /* Yield Criterion ** ** ** */
55            scalar plasticM_ ;
56            scalar tauY = 0;
57            // compute f and tauY
58            scalar f = plasticityModel.phi2(tauY, tauDev, strain_p[n], eps_dot[n], T[n]);
59            if ( f > 0.0) and isLastStage ){ /* von-Mises plasticity yield ** ** ** */
60                const vector directionV = tauDev * sqrt15 / normTauDev;
61
62                // Compute Plastic Multiplier
63                plasticM_ = plasticityModel.Newton3(f, mu, strain_p[n], T[n], eps_dot[n], tauY/*sigmaY*/);
64
65                // Correct elastic stretches
66                eStretch[n][0] /= Foam::exp( plasticM_*directionV[0] );
67                eStretch[n][1] /= Foam::exp( plasticM_*directionV[1] );
68                eStretch[n][2] /= Foam::exp( plasticM_*directionV[2] );
69
70                // Principle Deviatoric Kirchoff Stress Tensor
71                coeffTau[n] = 1. - ( Foam::sqrt(6.) * mu*plasticM_ / normTauDev );
72                tauDev *= coeffTau[n];
73            } else { /* end von-Mises plasticity yield ** ** ** */

```

```

74     coeffTau[n] = 1.; plasticM_ = 0;
75   }
76   /* Update Kirchhoff and Piola Stresses * * * * * */
77   const scalar JFp_n = JF[n] * p[n];
78   const tensor Kirch = ((tauDev[0] + JFp_n) * n0xn0)
79                       + ((tauDev[1] + JFp_n) * n1xn1)
80                       + ((tauDev[2] + JFp_n) * n2xn2) ;
81   P[n] = Kirch & Finv_[n].T();
82
83   /* Compute von-Mises * * * * * */
84   const tensor devK = Kirch - (JFp_n * tensor::I);
85   vMises_[n] = sqrt15 * Foam::sqrt( devK && devK );
86   if ( (f > 0.0) and isLastStage ) { /* Compute be, and CpInv, eps, epsdot * * * * * */
87     be = ((eStretch[n][0]*eStretch[n][0]) * n0xn0)

```

Listing 7.2: solidModel.C

Furthermore, the solver is accompanied by a set of libraries: the *solidModels* library, which implements the necessary routines to update the physical stress. The *aleModels* library, which is in charge of computing or updating all the quantities impacted by mesh motion in the ALE formulation. The *volumetricEnergy* and *deviatoricEnergy* libraries, which entail models of volumetric and deviatoric energy potentials together with their derived stress tensors. The libraries implementing energy potentials are used both for the solid model and for the ALE model due to the complex expression of the considered Piola-Kirchhoff stress \mathbf{P} .

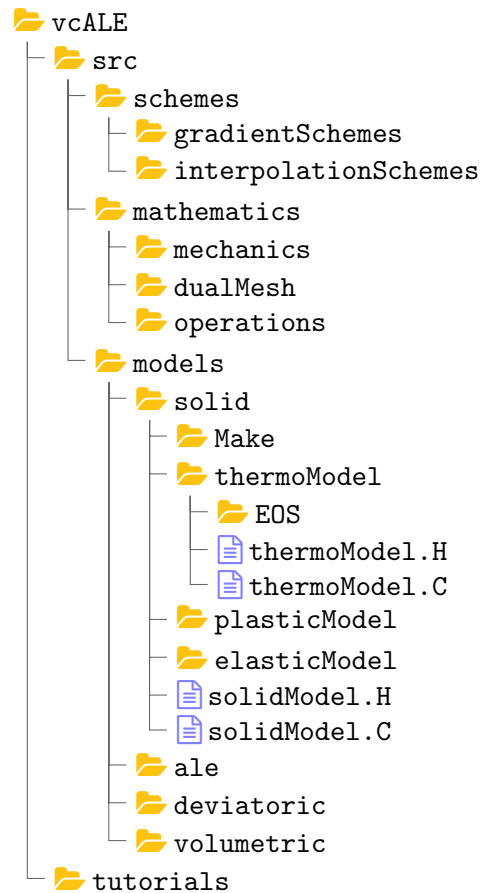


FIGURE 7.2: Solver structure: library sources; files related to the compilation are omitted.

7.2.2 Setup of the numerical simulation

The description of the simulation workflow respected by the author is described in the following and summarised in Figure 7.3. Before conducting a numerical simulation with OpenFOAM, a comprehensive pre-processing step involving geometry design is required.

This stage encompasses the use of various tools and software to create a suitable geometry for analysis. For instance, CAD software, such as SolidWorks, Autodesk Inventor, CATIA, Salome or FreeCAD, can be employed to design the initial 3D geometry and generally offer user-friendly interfaces for creating complex shapes and structures, as well as to perform geometry clean-ups. Alternatively, GMSH is a mesh generation software extensively used within the OpenFOAM community, which enables the generation of structured or unstructured meshes, and which relies on scripts and parameters like cell size, boundary layers, and refinement zones to accurately represent the geometry. Furthermore, the OpenFOAM suite comes with in-house tools such as *blockMesh* to create structured geometries, and *snappyHexMesh* to refine and capture intricate geometry design.

The obtained geometry serves as the foundation for the subsequent numerical setup in OpenFOAM and therefore requires careful attention. The boundaries will be associated with boundary conditions for the spatial and material linear momenta and/or velocities, as well as for the spatial and material tractions.

For commercial software, the attribution of boundary conditions is done together with the geometry design. When both are ready, the mesh can be generated.

Accurately defining the physical model and material properties is a critical step for conducting the numerical simulation. It involves specifying the governing equations, and material properties that govern the behaviour of the system under analysis. The choice of governing equations depends on the specific physics being simulated, and properties such as density, viscosity, thermal conductivity, and specific heat capacity must be defined based on the physical properties of the materials involved.

The numerical scheme plays a vital role in achieving accurate and stable results, and governs how the governing equations are discretised and solved numerically. For instance, OpenFOAM natively proposes common discretisation schemes such as first-order upwind, second-order central differencing, third-order QUICK, and higher-order schemes like MUSCL. OpenFOAM also employs various gradient calculation schemes such as the least squares method, and linear or cubic Gauss methods. OpenFOAM supports different time integration schemes, such as explicit methods (Euler and Runge-Kutta methods), and implicit methods (Crank-Nicolson).

After the solver has finished the simulation, effective visualization techniques enable the extraction of meaningful insights. For instance, post-processing tools such as *Paraview* can be used to visualize simulation results. Also, time-varying plots of quantitative data such as averages can also be conducted with the use of custom-made Python scripts.

The structure of a numerical experiment is presented in Figure 7.4. Generically, a *clean* and *run* scripts are present to clean the folder, and prepare and run the numerical solver. The empty file *case.foam* is a mock up for *Paraview* to be able to read mesh information, and the *specimen.msh* file is a GMSH generated mesh file that can be converted to OF format. The mesh information, after being generated, is located in the *constant/polyMesh* folder and usually comprises the files *boundary*, *pointZones*, *cellZones*, *owner*, *faceZones*, *faces*, *neighbour*, *points*, *sets*, *volume* (refer to [50] for further explanation). The file *constant/mechanicalProperties* contains the selected models and relevant parameters used in the simulation, and the file *constant/runParameters* contains technical information such as the name of meshfiles. In *0.orig*, the user includes the initial conditions and boundary types for all relevant fields (usually \mathbf{p}_x denoted as *lm* and θ denoted as *theta*). Eventually, the directory *system* contains the file *controlDict* that is a file dictionary containing data such as start and end times of the simulation, number of timesteps or the value of the CFL number. In classical OF solvers, that folder also contains the files *fvSchemes* and *fvSolution* that are used to set which numerical solver will be used for divergence and gradient computation.

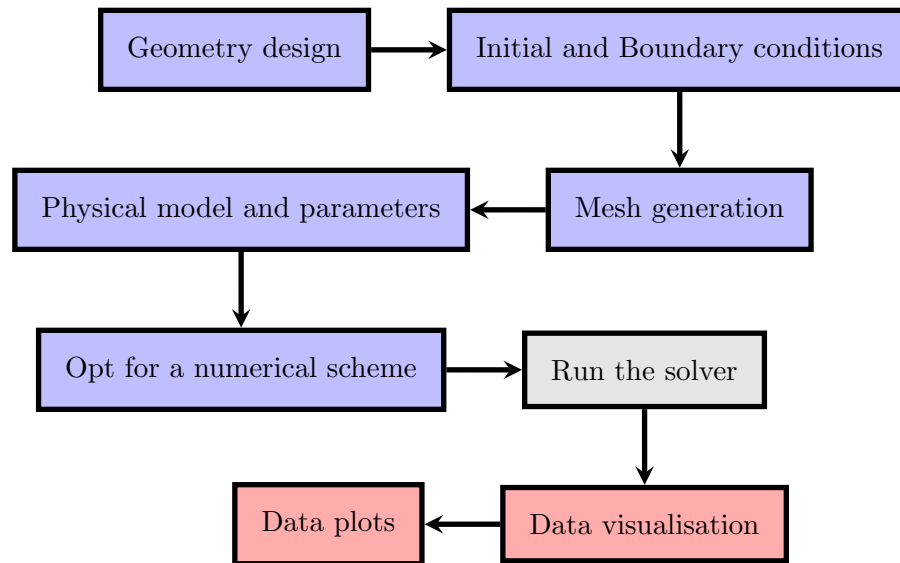


FIGURE 7.3: Workflow of an OpenFOAM simulation. The pre-processing steps are represented in blue, the post-processing steps are represented in red and the solving step is represented in grey.

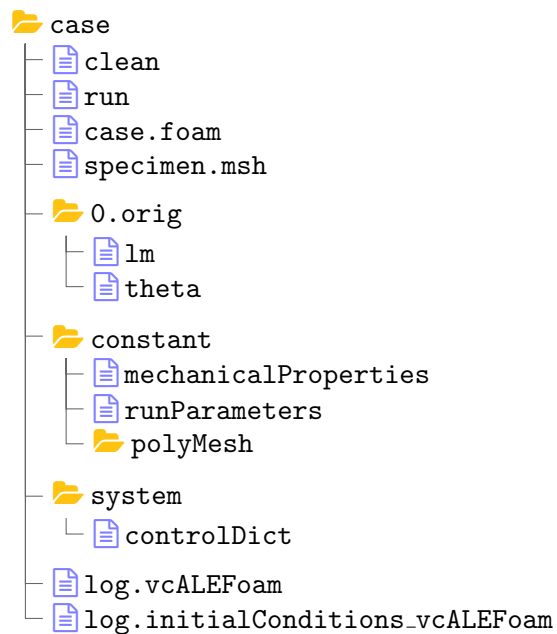


FIGURE 7.4: Case directory structure.

7.2.3 Algorithm workflow

The workflow is presented in Algorithm 5. The term RHS refers to the right hand side expression of any solved conservation law. The algorithm is followed regardless of the time integrator. The spatial λ_{Φ} and material λ_{Φ} stretches, as well as the plasticity algorithm, features the computation of eigenstructures taken care of by the Jacobi iterative algorithm [294].

Algorithm 5: Workflow of the visco-plasticity algorithm, featuring the multi-stage Runge-Kutta time integrator.

- 1 Compute spatial stretch λ_{Φ} ;
 - 2 Compute timestep increment Δt ;
 - 3 Save values of previous timestep;
 - 4 **for** Runge-Kutta stage k {
 - 5 Obtain integration coefficients;
 - 6 Compute gradients $\nabla_{\mathbf{x}}\mathbf{v}, \nabla_{\mathbf{x}}\mathbf{W}, \nabla_{\mathbf{x}}p$;
 - 7 Compute RHS interior and boundary;
 - 8 Update displacements Ψ, Φ and conserved variables \mathbf{U} ;
 - 9 Compute intermediate variables $\mathbf{F}, \mathbf{H}, J, \mathbf{v}, \hat{\mathbf{v}}$;
 - 10 Apply strong boundary corrections for $\mathbf{p}_{\chi}, \mathbf{W}, \hat{\mathbf{v}}$;
 - 11 Update stresses $\mathbf{P}, \mathbf{P}_{\mathbf{W}}$ and temperature θ ;
 - 12 Compute plastic dissipation w_p ;
 - 13 Compute implicit convective term for plastic internal variables;
-

Remark 7.2.1 *Generic OpenFOAM solvers usually make use of a list of commands each performing an action on an entire field; for instance, `pointTensorField invF = Foam::inv(F)` computes the inverse of the deformation gradient tensor \mathbf{F} at every node, and stores the variable in `invF`. The author opted for a different approach that is more optimised in the case of the ALE solver because it results in minimising the loops over edges, faces and nodes.*

7.3 Conclusion

This chapter presented how the numerical schemes and the set of conservations laws are put together in a framework for further use in numerical experiments. It encompasses the resolution of the ALE equations presented in the previous chapters, as well as numerical techniques adapted for fast solid dynamics, for the first time on the OpenFOAM platform. The power of the C language together with the flexibility of the formulation and an optimised implementation, makes this framework a serious alternative in fast solid dynamics, as demonstrated by the numerical examples in the next chapters.

CHAPTER 8

NUMERICAL SIMULATIONS: ISOTHERMAL CASES

“Qui dit préjugé dit une opinion qu’on a reçue sans examen; parce qu’elle ne se soutiendrait pas.”

— EMILIE DU CHÂTELET, *Discours sur le bonheur*, 1779

8.1 Introduction

The objective of this Chapter is to present a series of numerical examples in order to show the applicability of the ALE computational framework as well as to prove its robustness, its accuracy and its capability to preserve momentum. Scenarios involving polyconvex hyperelastic isothermal constitutive models for nearly incompressible materials will be considered. The mixed system considered in this Chapter comprises the general ALE equations presented in Equations (2.76), and are supplemented by the plasticity equation in the context of isothermal plasticity. The resulting framework used in this chapter hence relies on the (isothermal) elastoplastic mixed systems reported in Section (5.3.1) (that is, the first and second mixed formulations presented in Tables ??). All examples are simulated in a three dimensional space using the proposed ALE solid mechanics solver *vcALEFoam* whose structure is presented in Chapter 7. The ALE numerical framework can be referred to as *vc-ALE*.

The body forces are considered to be neglected, unless stated otherwise. The energy components may be reported but the energy remains uncoupled from the mixed system. In the case of material undergoing reversible (elastic) deformation, the prescription of a mesh motion is applied unless stated otherwise. On the other hand, situation involving permanent (plastic) deformations will demonstrate the potential of the ALE framework by solving the material motion based on the conservation law introduced in Chapter 5.

For several numerical cases, an analytically prescribed motion is used to benchmark the ALE formulation. That prescribed motion is composed of sinusoidal function and can be viewed in Section 8.1.1. Moreover, the ALE formulation is sometimes compared with its equivalent Total Lagrangian form. Such TLF equivalent is obtained by means of degenerating the ALE equations, and is discussed in Appendix B.5. In the following, the material and spatial cofactor will always be treated as intermediate variables, while the importance to solving the other geometric mappings will be assessed in Section 8.2.

8.1.1 Analytical prescription of ALE motion

The material motion can be artificially prescribed by an analytical expression, provided geometrical requirements mentioned in Section 5.1.1 are satisfied. In the following, an example of prescribed material mapping is represented by a plane sinusoidal-type function defined for a rectangular domain of dimensions $\chi_{1,R}$ and $\chi_{2,R}$ and of centre (χ_1^C, χ_2^C) as

$$\Psi^\beta = \chi + \begin{bmatrix} \beta_x \sin^2\left(\frac{\pi t}{T_x}\right) \sin\left(\frac{\tilde{\chi}_1}{\chi_{1,R}^R}\right) \left[\cos\left(\frac{\tilde{\chi}_2}{\chi_{2,R}^R}\right) + \sin\left(\frac{\tilde{\chi}_2}{\chi_{2,R}^R}\right) \right] \\ \beta_y \sin^2\left(\frac{\pi t}{T_y}\right) \sin\left(\frac{\tilde{\chi}_2}{\chi_{2,R}^R}\right) \left[\cos\left(\frac{\tilde{\chi}_1}{\chi_{1,R}^R}\right) + \sin\left(\frac{\tilde{\chi}_1}{\chi_{1,R}^R}\right) \right] \\ 0 \end{bmatrix}, \quad (8.1)$$

where $\tilde{\chi}_1 = \frac{\chi_1 + \chi_1^C}{2\pi}$ and $\tilde{\chi}_2 = \frac{\chi_2 + \chi_2^C}{2\pi}$ are coordinates shifted with respect to the centre of the domain, β_x, β_y and $\beta[\text{m/s}] > 0$ are parameters determining the magnitude of the mesh motion, and T_x, T_y and T are the periods⁵² of the sinusoid functions. Due to the sinusoidal structure of the motion (8.1), nodes oscillate either vertically or horizontally. Note that the symmetry condition $\mathbf{W} \cdot \mathbf{N}_\chi$ is always fulfilled. Moreover, the conservation of geometrical mappings χ_Ψ is satisfied by construction. This material motion is depicted in Figure 8.1 on an unit cube for a magnitude $\beta = 10^{-2}\text{m}$, where a selection of nodes is highlighted in red colour so their displacements can be followed. Note that in addition to the nature of the prescribed mesh mapping, the quality of the mesh is further impacted by the use of tetrahedral elements, and in particular their orientation.

⁵²Most of the numerical cases imply $\beta_y = \frac{\chi_{2,R}^R}{\chi_{1,R}^R} \beta_x = \frac{\chi_{2,R}^R}{\chi_{1,R}^R} \beta$, and $T_x = T_y/2 = T$.

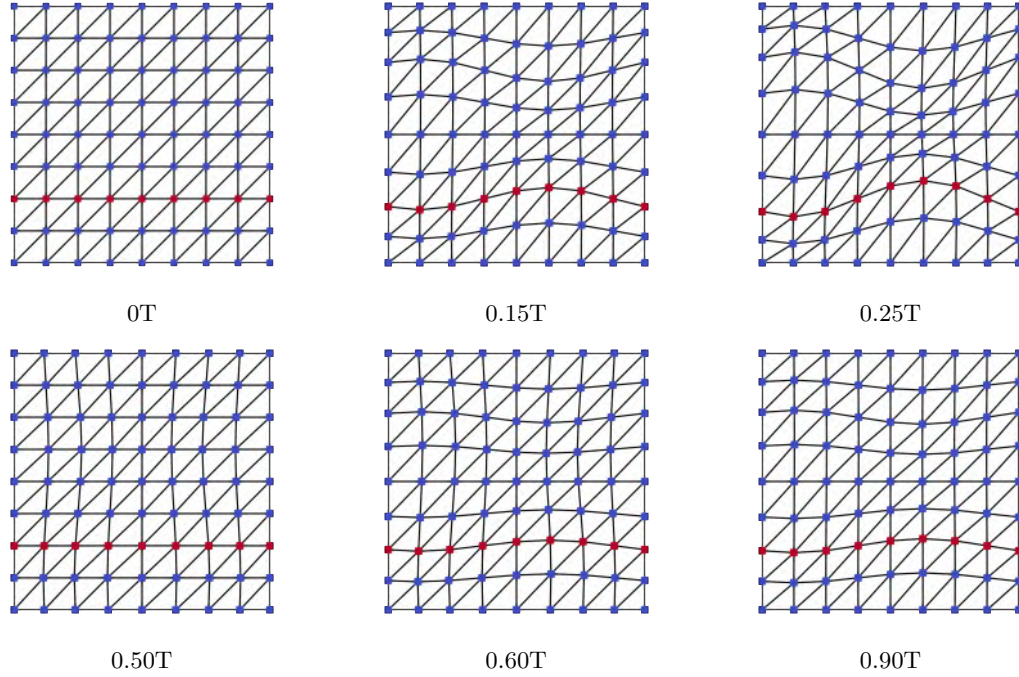


FIGURE 8.1: Visualisation of the analytical material motion (8.1) on a unit cube with a magnitude $\beta = 10^{-2}\text{m}$. A range of 8 nodes is highlighted in red so their displacements can be followed.

8.2 Satisfaction of geometric conservation laws: Patch test

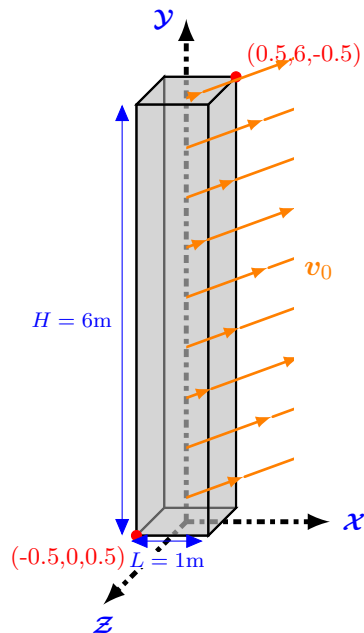


FIGURE 8.2: Translation: Problem setup.

all faces	free boundary (6.32)
-----------	-------------------------

TABLE 8.1: Summary of boundary conditions.

This first numerical example consists of a three-dimensional patch test in order to assess the correctness of the computational implementation, and the importance of fulfilling the discrete geometric conservation laws. This problem was already presented in [195, 199, 219, 232, 295]. The considered specimen is a $1 \times 6 \times 1\text{m}$ square column presented in Figure 8.2. The polyconvex hyperelastic constitutive model considered is the neo-Hookean model together with the volumetric potential presented in [239] and featured in Table B.1. The material parameters are listed in Table 8.2. Velocities are uniformly initialised

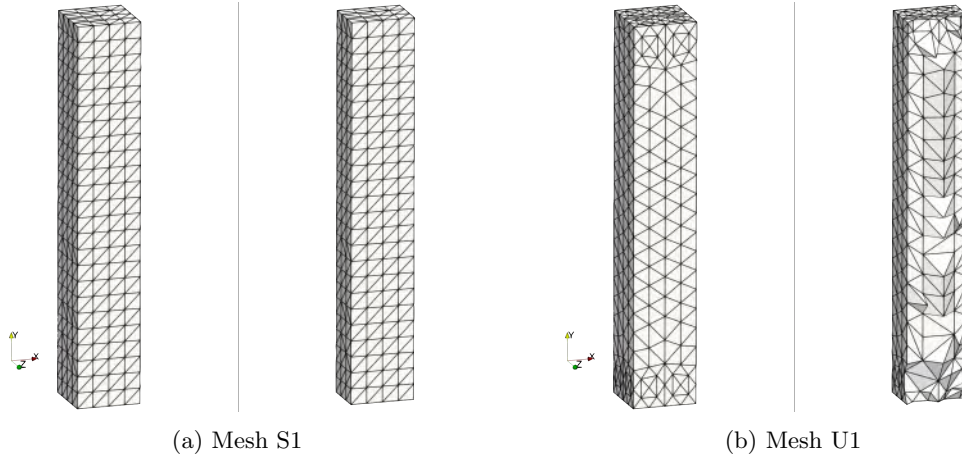


FIGURE 8.3: Translation:(a) Structured (S1) and (b) unstructured (U1) meshes. An overview of the mesh is presented on the left and a crinkle clip of half of the column is presented on the right.

to $v_0 = [3, 1, 0]^T$ m/s in the body and the boundaries are all left free (see Table 8.1) so a straight uniform displacement is expected across the whole specimen. In addition, the natural deformation gradient \mathbf{F} is expected to remain the identity mapping, the first Piola-Kirchhoff stress tensor \mathbf{P} is expected to remain equal to $\mathbf{0}$ and the natural Jacobian J is expected to remain equal to 1 throughout the numerical simulation. The volumetric potential is based on the [239] model defined in Appendix B.3.2.

Young's modulus	E [MPa]	17
density	ρ_R [kg/m ³]	1100
Poisson's ratio	ν	0.45

TABLE 8.2: Translation: Table of physical parameters.

period	T [s]	0.5
magnitude	β [m/s]	0.02

TABLE 8.3: Translation: Table of parameters for the prescribed mesh motion in Equation (8.1).

First of all, a structured mesh⁵³ S1 of $4 \times 24 \times 4$ tetrahedral elements (2304 elements, 625 nodes) is compared to an unstructured mesh U1 (1864 elements, 551 nodes). The objective is to demonstrate that the same result is obtained in both cases using the Total Lagrangian Formulation, and therefore that the implementation of the numerical framework does not depend on the structure of the mesh. In Figure 8.4, the error in J is and the error in the first and second velocity components are presented for the L^2 norm. As expected the two meshes provide very similar results that only differ due to machine accuracy.

Secondly, the mesh motion presented in Equation (8.1) is prescribed to the numerical simulation with parameters listed in the Table 8.3, and the results are compared when using a different number of geometric conservation laws, on the S1 mesh. The objective is to demonstrate that when introducing a non negligible mesh motion, the numerical framework must account for additional (geometric) conservation laws to recover the accuracy of the Total Lagrangian simulation. A collection of snapshots can be viewed in Figure 8.7 at

⁵³Note that results of this numerical example are shown with the present mesh because it is reused throughout this chapter. For this patch test, similar results can be obtained with a coarser mesh.

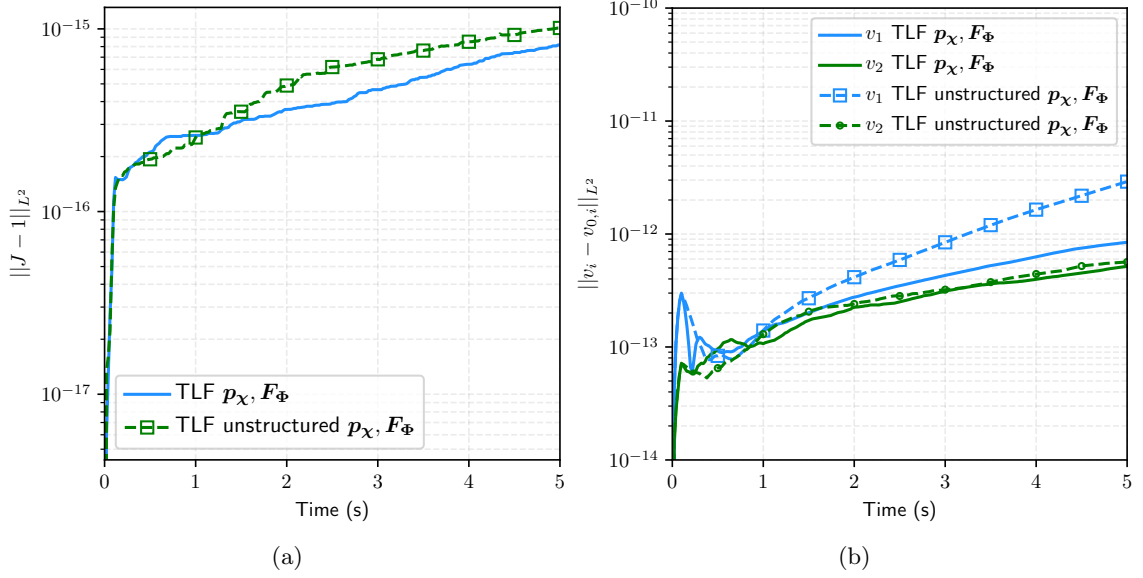


FIGURE 8.4: Translation: Time evolution of the L^2 error in Jacobian and velocity components when using a Total Lagrangian Formulation and solving for $\{p_\chi, F_\Phi\}$ on a structured (blue) and on an unstructured (green) mesh. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using a discretisation of $4 \times 24 \times 4$ elements across on a tetrahedral mesh. A neo-Hookean model is used with parameters listed in Table 8.2.

time $t = 0, 0.8, 1.6, 2.4, 3.2, 4.0, 4.8$ s for 3 cases: a) The formulation p_χ, F_Φ is used and no geometric conservation laws are solved, b) the formulation p_χ, F_Φ is solved, including the material deformation gradient, and c) the formulation $p_\chi, F_\Phi, F_\Psi, J_\Psi$ is solved, including the material deformation gradient F_Ψ and material Jacobian J_Ψ . It can be seen in the first scenario that the straight trajectory of the body is lost, and strong instabilities in the Jacobian appeared. In the second scenario, the addition of the conservation law on the material deformation gradient lets the body keep its original straight trajectory but it is still smeared with consequent Jacobian instabilities. In the third and final scenario, the addition of the conservation law on the material Jacobian lets the body keep its straight line and addresses the instabilities in the natural Jacobian. Those visual observations are accompanied by a study of the L^2 -error in the natural Jacobian 8.6 and velocity components in Figure 8.6. The first scenario where no geometric conservation law is solved corresponds to the blue curve and it is clear that the error in Jacobian and in the velocity components is of a too great factor. In the second scenario, which corresponds to the introduction of the conservation law of the material deformation gradient F_Ψ and that is represented by the green curve, both errors decrease and the deviation of the column is no longer visible (as noted in Figure 8.7). However, the best quality results are obtained when solving for the two geometric conservation laws (scenario 3) and are represented by the red curve. In this last example, the errors are comparable to those reported for the Total Lagrangian case (see Figure 8.4).

As a conclusion, the implementation of the numerical framework was proven to be adapted to structured and unstructured meshes for simple geometries. The accuracy obtained via a simulation based on the Total Lagrangian formulation can be obtained when using the Arbitrary Lagrangian Eulerian method, provided the two geometric conservation laws are solved. If the conservation of the material Jacobian is not solved, the results can still be accepted but a significant loss of accuracy is observed. However, the conservation of the material deformation gradient must be solved when using the Arbitrary Lagrangian Eulerian method to avoid important accuracy losses that are very likely to affect the stability of the numerical framework.

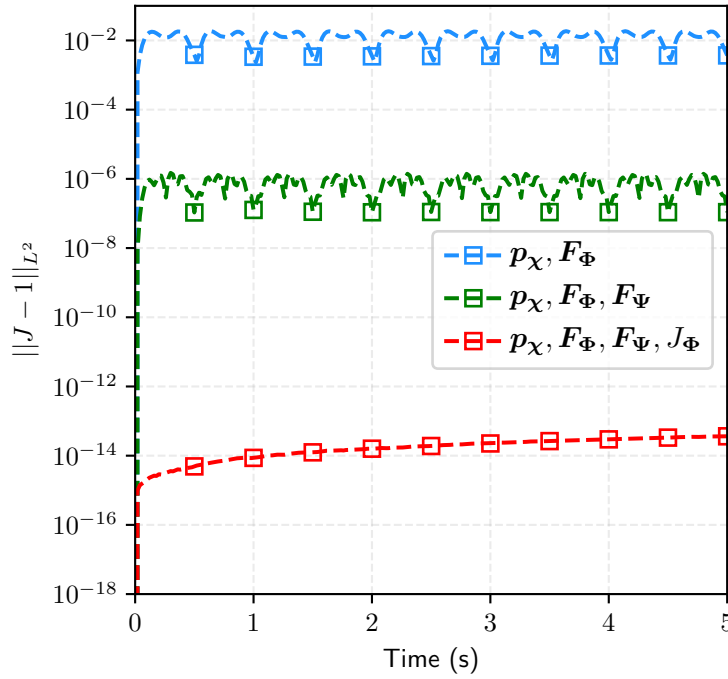


FIGURE 8.5: Translation: Time evolution of the L^2 -error in Jacobian $\|J - 1\|_{L^2}$ when solving for $\{p_\chi, F_\Phi\}$ (blue), $\{p_\chi, F_\Phi, F_\Psi\}$ (green) and $\{p_\chi, F_\Phi, F_\Psi, J_\Phi\}$ (red). Results obtained with velocity $v_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

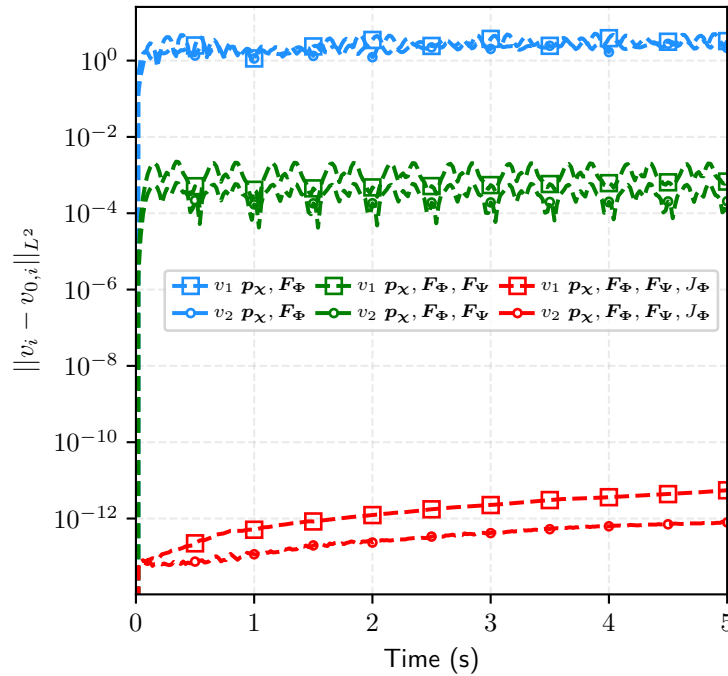


FIGURE 8.6: Translation: Time evolution of the L^2 -error in velocity components $\|v_i - v_{0,i}\|_{L^2}$ when solving for $\{p_\chi, F_\Phi\}$ (blue), $\{p_\chi, F_\Phi, F_\Psi\}$ (green) and $\{p_\chi, F_\Phi, F_\Psi, J_\Phi\}$ (red). Results obtained with velocity $v_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

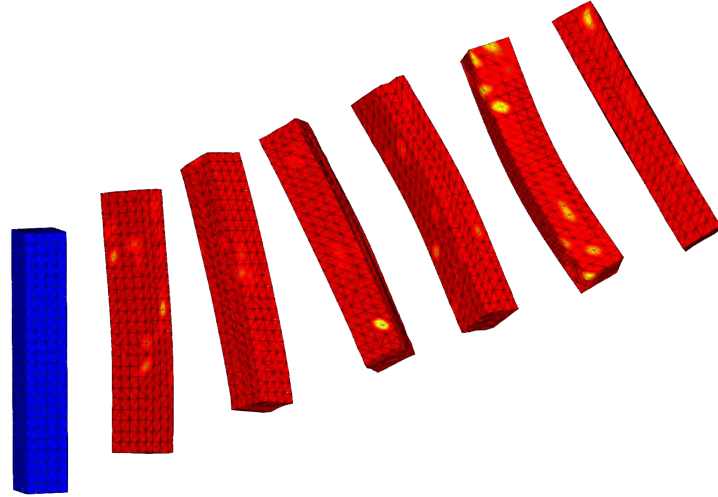
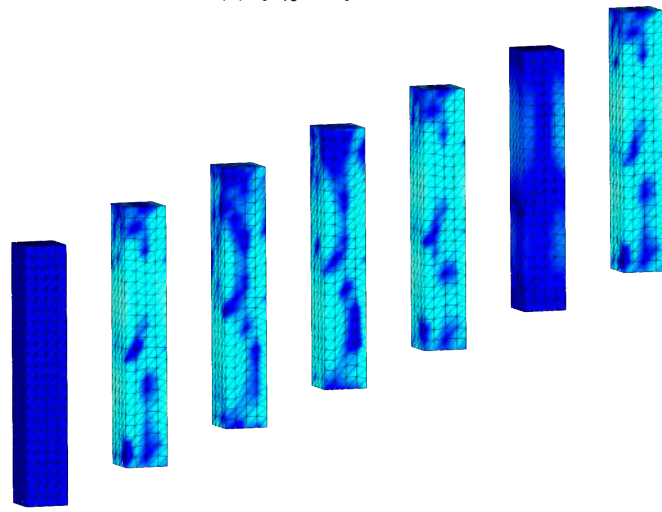
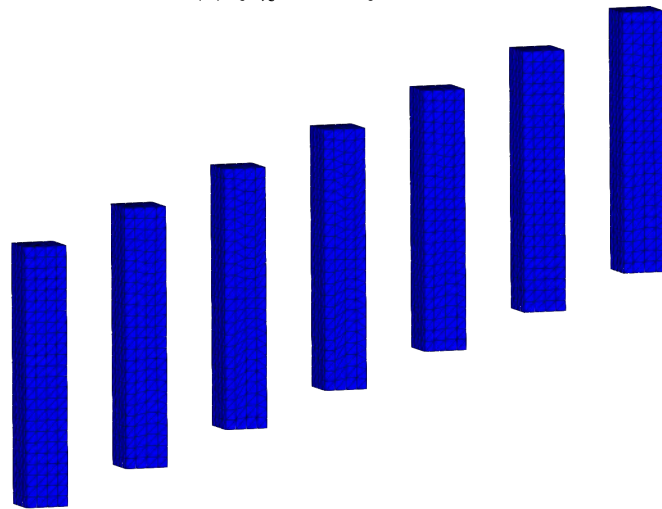
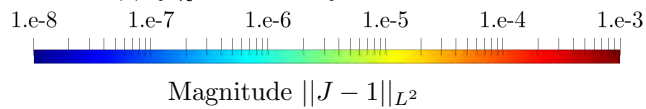
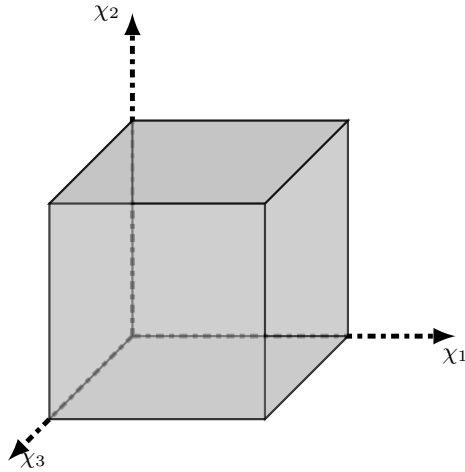
(a) $\{p_\chi, F_\Phi\}$ formulation.(b) $\{p_\chi, F_\Phi, F_\Psi\}$ formulation.(c) $\{p_\chi, F_\Phi, F_\Psi, J_\Psi\}$ formulation.

FIGURE 8.7: Translation: Snapshots of the deformation at time $t = 0, 0.8, 1.6, 2.4, 3.2, 4.0, 4.8$ s, along with distribution $\|J - 1\|_{L^2}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model is used with parameters listed in Table 8.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

8.3 Mesh convergence: Low dispersion swinging cube



faces $\chi_1 = 0$, $\chi_2 = 0$ and $\chi_3 = 0$	symmetric boundary (6.35)
faces $\chi_1 = 1$, $\chi_2 = 1$ and $\chi_3 = 1$	anti-symmetric boundary (6.38)

TABLE 8.4: Summary of boundary conditions.

FIGURE 8.8: Swinging cube: Geometry.

The objective of this numerical experiment is to assess the spatial convergence of the proposed numerical framework, based on the Vertex-Centred Finite Volume Methodology (VCFVM) and on the implementation of the general ALE mixed system (2.76), including the linear momentum \mathbf{p}_χ , the spatial deformation gradient \mathbf{F}_Φ and the two material mappings \mathbf{J}_Ψ and \mathbf{F}_Ψ . This numerical example was introduced for a two-dimensional geometry in [296], and further studied in [285]. More recently, it has been studied for a three-dimensional geometry using a cell-centred [75] and a vertex-centred [56] approach, as well as for a SPH [109] approach. The considered geometry consists in a three-dimensional unit cube (see Figure 8.8) with symmetric boundary conditions at faces $\chi_1 = 0$, $\chi_2 = 0$ and $\chi_3 = 0$, and skew-symmetric boundary conditions at faces $\chi_1 = 1$, $\chi_2 = 1$ and $\chi_3 = 1$ (which is summarised in Table 8.4). The constitutive model used for the simulations is a hyperelastic neo-Hookean deviatoric model paired with a quadratic potential (see the Table B.1 of volumetric potentials). For small deformations, the constitutive behaviour of the material can be approximated by the linear elasticity theory, and the problem has a closed-form solution characterised by displacements defined as [58, 75, 76, 78–81, 107, 108, 197, 285]

$$\mathbf{u}(\boldsymbol{\chi}, t) = U_0 \cos\left(\frac{\sqrt{3}}{2} c_d t\right) \begin{bmatrix} A \sin\left(\frac{\pi\chi_1}{2}\right) \cos\left(\frac{\pi\chi_2}{2}\right) \cos\left(\frac{\pi\chi_3}{2}\right) \\ B \cos\left(\frac{\pi\chi_1}{2}\right) \sin\left(\frac{\pi\chi_2}{2}\right) \cos\left(\frac{\pi\chi_3}{2}\right) \\ C \cos\left(\frac{\pi\chi_1}{2}\right) \cos\left(\frac{\pi\chi_2}{2}\right) \sin\left(\frac{\pi\chi_3}{2}\right) \end{bmatrix}; \quad c_d = \sqrt{\frac{\lambda + 2\mu}{\rho_R}}. \quad (8.2a,b)$$

In the closed-form solution (8.2), the arbitrary parameters A, B, C verify $A = B = C$ to ensure the existence of a non-zero⁵⁴ pressure field. Moreover, the magnitude U_0 must be small enough for the solution to be considered linear and the closed-form solution to hold. Consequently, in this convergence study, the initialisation of the numerical example is based on the closed-form (8.2), which can then be used to obtain the initial geometric mappings

$$\mathbf{F}_0 = \mathbf{I} + \nabla_{\boldsymbol{\chi}} \mathbf{u}_0; \quad \mathbf{H}_0 = \frac{1}{2} \mathbf{F}_0 \times \mathbf{F}_0; \quad J_0 = \frac{1}{6} (\mathbf{F}_0 \times \mathbf{F}_0) : \mathbf{F}_0, \quad (8.3a,b,c)$$

⁵⁴Alternatively, this numerical example was studied in [35, 58, 76, 78, 197] with parameters verifying $A + B + C = 0$ and $c_d = \sqrt{\frac{\mu}{\rho_R}}$, which leads to a non-volumetric deformation field.

and the initial displacement field

$$\mathbf{u}_0 \equiv \mathbf{u}(\boldsymbol{\chi}, 0). \quad (8.4)$$

Young's modulus	E [MPa]	17
Density	ρ [kg/m ³]	1100
Poisson's ratio	ν	0.45

TABLE 8.5: Swinging cube: Table of parameters for the physical model and for the closed-form solution.

Period	T [s]	1×10^{-3}
Magnitude	β [m/s]	1×10^{-4}

TABLE 8.6: Swinging cube: Table of parameters for the prescribed mesh motion in Equation (8.1).

The values of the parameters for the closed-form solution, as well as the values of the physical parameters, are summarised in Table 8.5. The convergence study is performed with the general ALE mixed system (2.76) with a mesh motion prescribed by Equation (8.1) and parameters defined in Table 8.6. The results are obtained at the time $t = 1$ ms and the time is integrated by a 2-stage Runge-Kutta scheme to aim at second-order accuracy.

A convergence study is carried out for the general ALE mixed formulation (2.76), when the material mesh motion is prescribed via the displacement in Equation (8.1) and with the parameters in Table 8.6, and based on the L^1 and L^2 norm of the error for the velocity components and the diagonal components of the first Piola Kirchhoff stress. It is performed on four meshes: S1 (343 nodes, 1296 tetrahedra, 6 elements across each edge), S2 (2197 nodes, 10368 tetrahedra, 12 elements across each edge), S3 (15625 nodes, 82944 tetrahedra, 24 elements across each edge), and S4 (117649 nodes, 663552 tetrahedra, 48 elements across each edge). First of all, Tables 8.7 and 8.8 list the error figures using the L^1 norm, respectively for the three components of \mathbf{v} and the diagonal components of \mathbf{P} . Then, Tables 8.9 and 8.10 show the error figures using the L^2 norm, respectively for the three components of \mathbf{v} and the diagonal components of \mathbf{P} . Eventually, the convergence rates of the aforementioned errors are depicted, using both the L^1 and the L^2 norms, on Figures

L^1 error	v_x	v_y	v_z
1/6	3.189×10^{-2}	3.177×10^{-2}	3.059×10^{-2}
1/12	7.718×10^{-3}	7.876×10^{-3}	7.578×10^{-3}
1/24	1.902×10^{-3}	1.934×10^{-3}	1.880×10^{-3}
1/48	4.641×10^{-4}	4.641×10^{-4}	4.627×10^{-4}
conv. rate	2.035	2.059	2.023

TABLE 8.7: Swinging cube: numerical values for the L^1 -error of the velocity components, based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

L^1 error	P_{xx}	P_{yy}	P_{zz}
1/6	1.881×10^{-2}	1.881×10^{-2}	1.881×10^{-2}
1/12	4.928×10^{-3}	4.928×10^{-3}	4.926×10^{-3}
1/24	1.252×10^{-3}	1.252×10^{-3}	1.252×10^{-3}
1/48	3.163×10^{-4}	3.163×10^{-4}	3.163×10^{-4}
conv. rate	1.985	1.985	1.985

TABLE 8.8: Swinging cube: numerical values for the L^1 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

L^2 error	v_x	v_y	v_z
1/6	2.716×10^{-2}	2.709×10^{-2}	2.645×10^{-2}
1/12	6.678×10^{-3}	6.769×10^{-3}	6.569×10^{-3}
1/24	1.654×10^{-3}	1.676×10^{-3}	1.641×10^{-3}
1/48	4.065×10^{-4}	4.064×10^{-4}	4.042×10^{-4}
conv. rate	2.025	2.044	2.021

TABLE 8.9: Swinging cube: numerical values for the L^2 -error of the velocity components, based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

L^2 error	P_{xx}	P_{yy}	P_{zz}
1/6	1.798×10^{-2}	1.798×10^{-2}	1.797×10^{-2}
1/12	4.661×10^{-3}	4.661×10^{-3}	4.660×10^{-3}
1/24	1.171×10^{-3}	1.171×10^{-3}	1.171×10^{-3}
1/48	2.932×10^{-4}	2.932×10^{-4}	2.932×10^{-4}
conv. rate	1.997	1.997	1.997

TABLE 8.10: Swinging cube: numerical values for the L^2 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the ALE mixed formulation (2.76). Convergence rates are calculated using meshes S3 and S4. A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

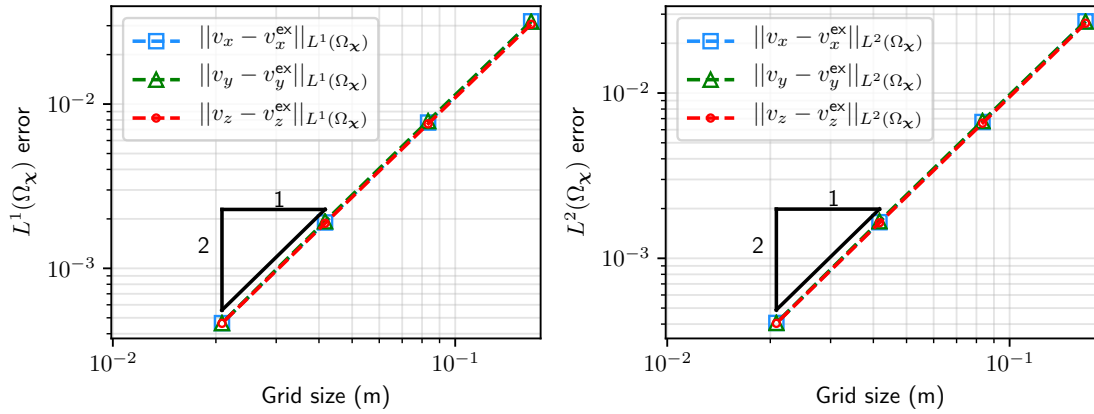


FIGURE 8.9: Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the velocity, based on the closed-form (8.2) and the ALE mixed formulation (2.76). A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

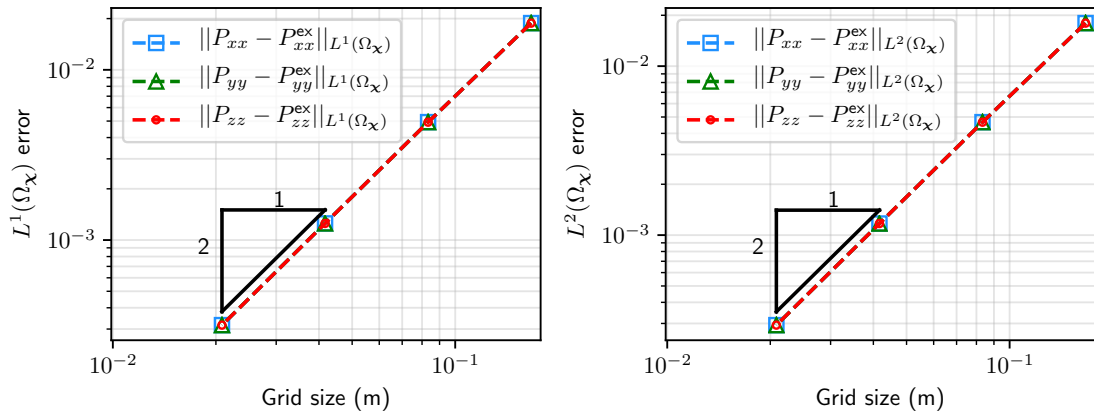


FIGURE 8.10: Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the first Piola Kirchhoff stress tensor, based on the closed-form (8.2) and the ALE mixed formulation (2.76). A neo-Hookean model is used with parameters listed in Table 8.5. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.6.

8.9 and 8.10, for respectively the velocity components and the diagonal components of \mathbf{P} . This convergence study clearly shows the capability of the *vc-ALE* numerical framework to perform a second-order spatial convergence for both velocities and stresses, as compared to the analytical closed-form solution described in Equation (8.2).

8.4 L-shaped block

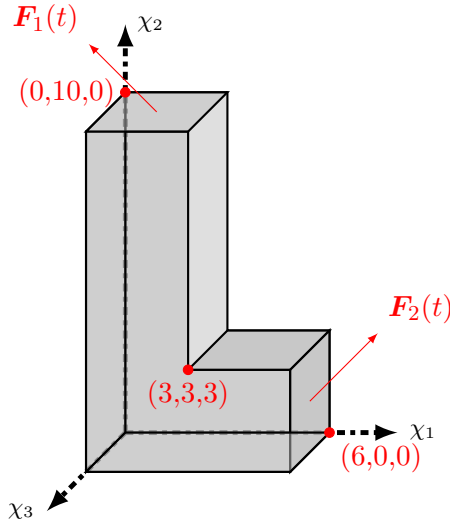


FIGURE 8.11: L-shaped block: Problem setup.

As studied in [58, 75, 108], the objective of this classical benchmark problem is to measure the capability of the proposed method to preserve both linear and angular momenta, as well as total energy over a long term response. A L-shaped specimen (see Figure 8.11) is subjected to an external torque consisting of time-varying tractions applied on two boundary faces, and described as

$$\mathbf{F}_1(t) = -\mathbf{F}_2(t) = \begin{bmatrix} 150 \\ 300 \\ 450 \end{bmatrix} f(t), \quad f(t) = \begin{cases} t & \text{if } 0 \leq t < 2.5 \\ t - 5 & \text{if } 2.5 \leq t < t \\ 0 & \text{else} \end{cases} \quad (8.5a,b)$$

The time span of 5s at the beginning of the simulation hence corresponds to a loading phase where energy is introduced in the system. After $t = 5s$, the specimen is left tumbling in space with free boundary conditions (see Table 8.11) and suffers from finite deformations and large rotations. A hyperelastic neo-Hookean model is considered with parameters summarised in Table 8.12.

Young's modulus	E [kPa]	50.05
density	ρ_R [kg/m ³]	1000
Poisson's ratio	ν	0.3

TABLE 8.12: L-shaped block: Table of parameters.

In this numerical example, the results will be compared for the Total Lagrangian $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ formulation, and for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. In the case of the ALE formulation, the mesh motion is prescribed with Equation (8.1), where parameters are listed in Table 8.13. The mesh motion is concealed to a rectangular area delimited by the points (0, 10, 0) and (1, 0, 1), and can be viewed in Figure 8.14. The volumetric potential is based on the [239] model defined in Appendix B.3.2.

Several meshes will be considered: $\{S1, S2, S3\}$ comprises respectively $\{5616, 18954, 44928\}$ cells and $\{1323, 4000, 8957\}$ particles. Those meshes are reported in Figure 8.12. The simulations are conducted with a 2-stage RK integrator, and a fixed CFL number $\alpha_{CFL} = 0.3$

face $\chi_2 = 10$	traction boundary (6.32)
face $\chi_1 = 6$	traction boundary (6.32)
all remaining faces	free boundary (6.32)

TABLE 8.11: Summary of boundary conditions.

Period	T [s]	1
Magnitude	β [m/s]	2×10^{-2}

TABLE 8.13: L-shaped block: Table of parameters for the mesh motion.

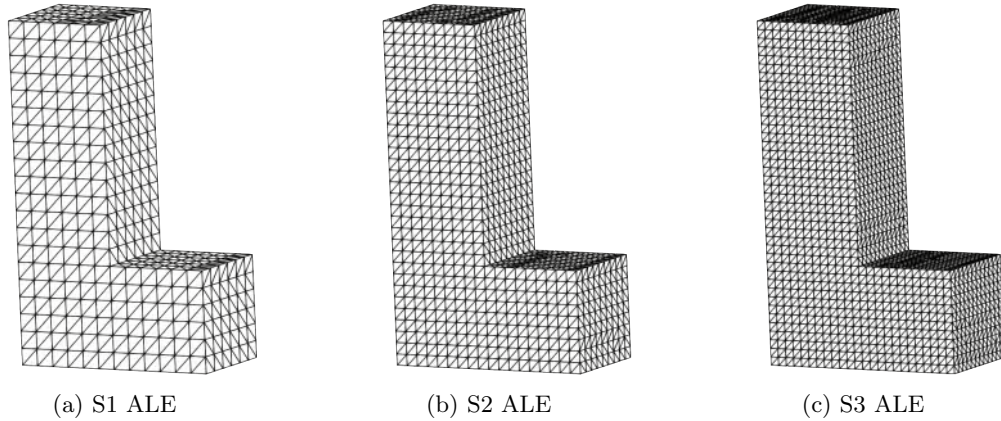


FIGURE 8.12: L-shaped block: Presentation of the S1, S2 and S3 meshes.

was used. A refinement analysis is shown in Figure 8.13 for the 3 meshes using the ALE formulation and the Total Lagrangian formulation (TLF) at time 24s. The deformation pattern as well as the pressure distribution for a small number of particles agree very well with the results obtained using finer discretisations. Results when using the ALE mixed formulation and when using TLF are also in very good agreement. As compared with previous results (SPH [37], CCFVM [75] and Petrov-Galerkin [78]), the deformation as well as the pressure field are visually matching with those reported using the ALE mixed formulation.

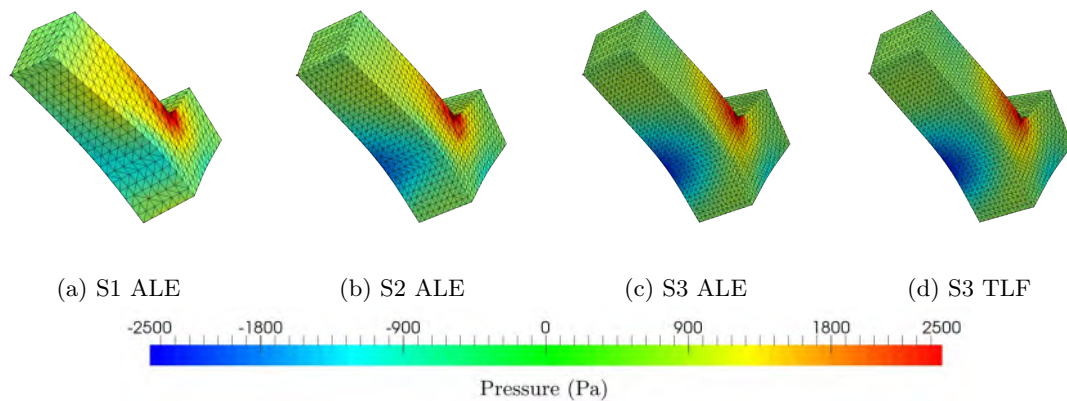


FIGURE 8.13: L-shaped block: Pressure distribution at time $t = 24$ s using, from the left to the right, the S1, S2 and S3 meshes together with the proposed ALE method and the S3 mesh together with the TLF method. Results are obtained with the impulse boundary condition (8.5). A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

In Figure 8.15, snapshots of deformations are shown together with the pressure distribution. The three components of the global linear and angular momenta are shown respectively in Figures 8.16 and 8.16 to study the ability of the proposed ALE mixed formulation to preserve them. The global linear momentum closely oscillates around zero machine accuracy as the specimen spins around its motionless centre of mass, and is conserved

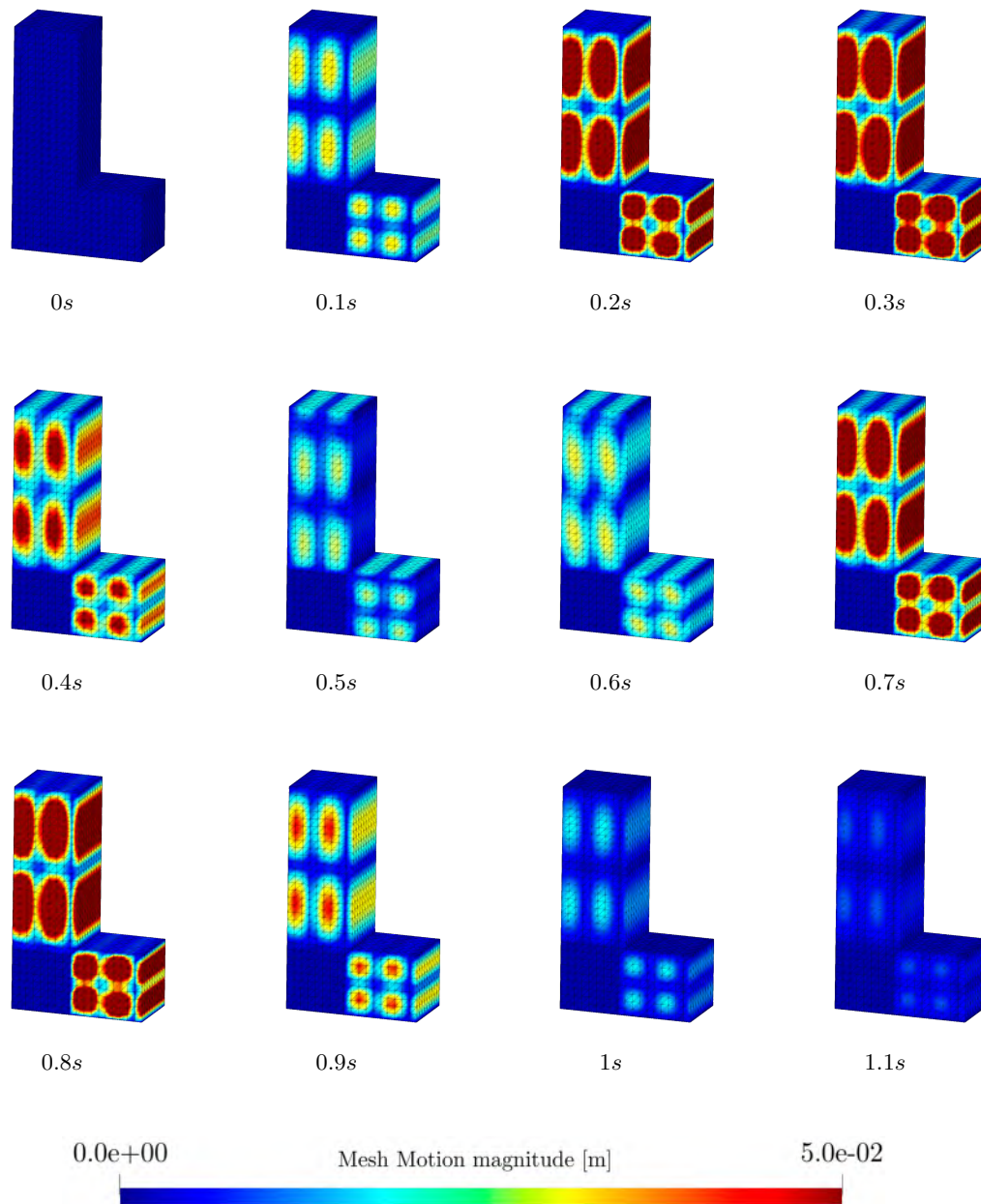


FIGURE 8.14: L-shaped block: Time evolution of the mesh deformation at time $t = 0.1, 0.2, 0.3, \dots, 1.0, 1.1$ s (left to right, top to bottom), along with contours of $\|\mathbf{u}_\Psi\|$, when solving for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S2. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

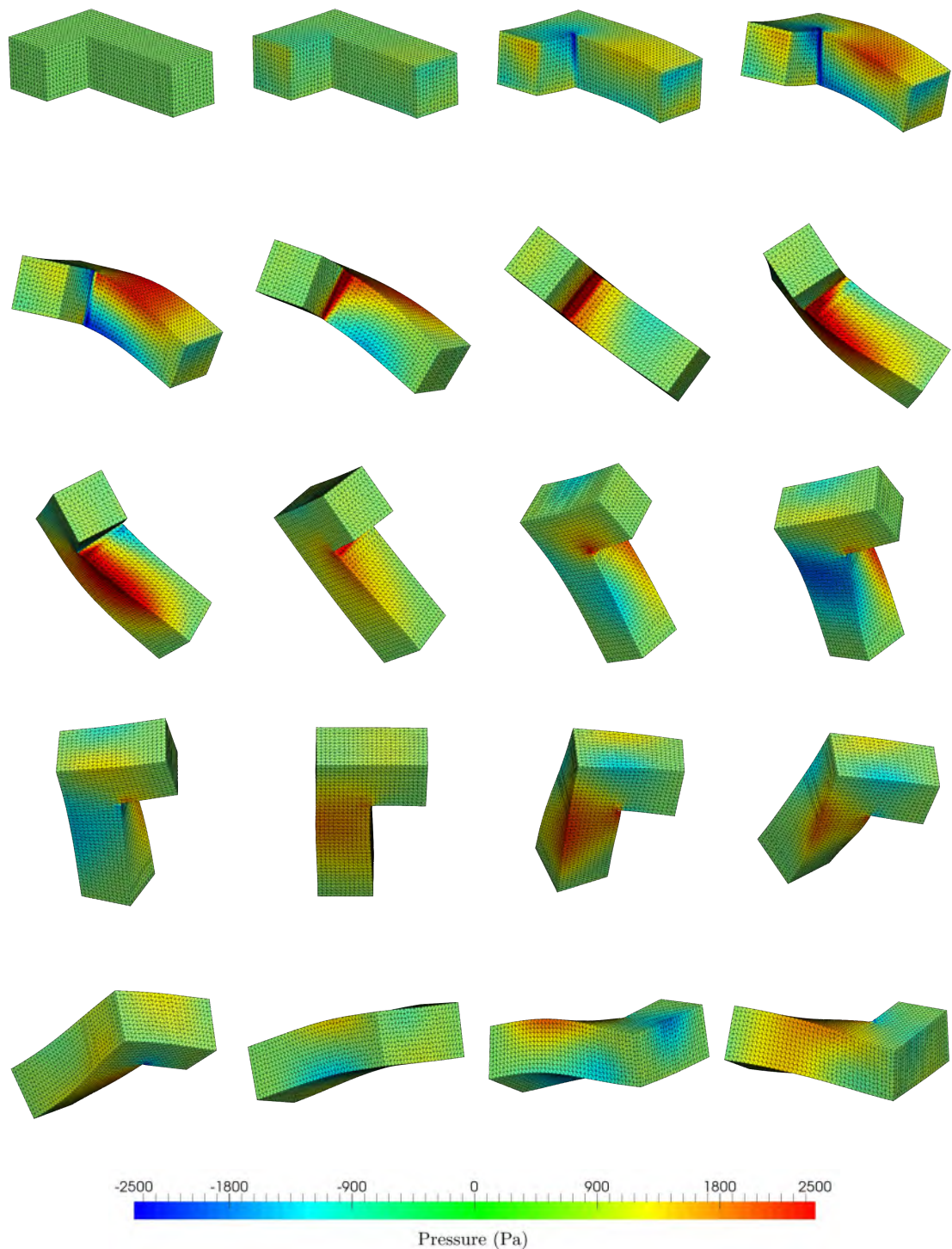


FIGURE 8.15: L-shaped block: Time evolution of the deformation at time $t = 0, 1, 2, 3, \dots, 19$ s (left to right, top to bottom), along with pressure contours, when solving for the $\{p_{\chi}, F_{\Phi}, F_{\Psi}, J_{\Psi}\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

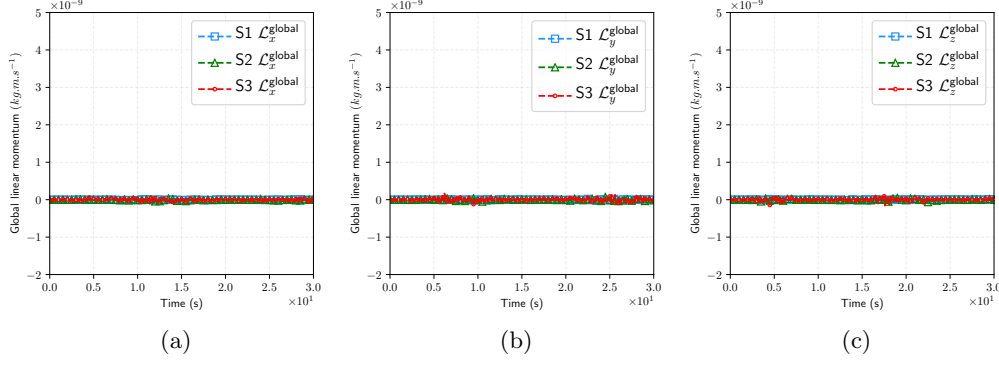


FIGURE 8.16: L-shaped block: Time evolution of the global linear momentum for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, \mathbf{J}_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

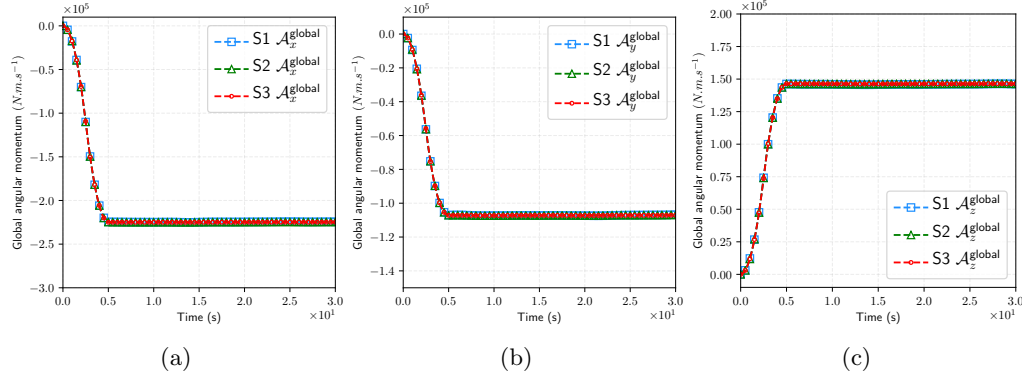


FIGURE 8.17: L-shaped block: Time evolution of the global angular momentum for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, \mathbf{J}_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

after the loading phase. On these diagrams, $\mathcal{L}_x^{\text{global}}$, $\mathcal{L}_y^{\text{global}}$ and $\mathcal{L}_z^{\text{global}}$ refer respectively to the first, second and third component of the global linear momentum. Moreover, $\mathcal{A}_x^{\text{global}}$, $\mathcal{A}_y^{\text{global}}$ and $\mathcal{A}_z^{\text{global}}$ refer respectively to the first, second and third component of the global angular momentum.

On the other hand, it can be seen that the global angular momentum is building during the loading phase and preserved then very accurately. In Figure 8.18a, the energy components are compared for mesh S3 when using the ALE mixed formulation, and the Total Lagrangian formulation. The dissipation is shown for several discretisations in Figure 8.18b. The kinetic energy, the internal energy and their summation, obtained using the Total Lagrangian and the ALE formulations are featured for mesh S3. The stability of the ALE method can be appreciated by how similar the evolution of the energy components are. Moreover, in both cases a slight decrease in the total energy, due to the use of upwinding stabilisation, is observed after the loading phase. This can be related to the numerical dissipation that is reported in Figure 8.18b, where the quality of the result is proportional to the level of discretisation. Third, and for qualitative comparison purposes, the velocity components at points $A = (0, 10, 0)^T$, $B = (6, 0, 0)^T$, and $C = (3, 3, 3)^T$ (see Figure 8.11) is reported in Figure 8.19 for the ALE method using different meshes and for the Total Lagrangian formulation. It is clear that the solution converges with a progressive level of refinement.

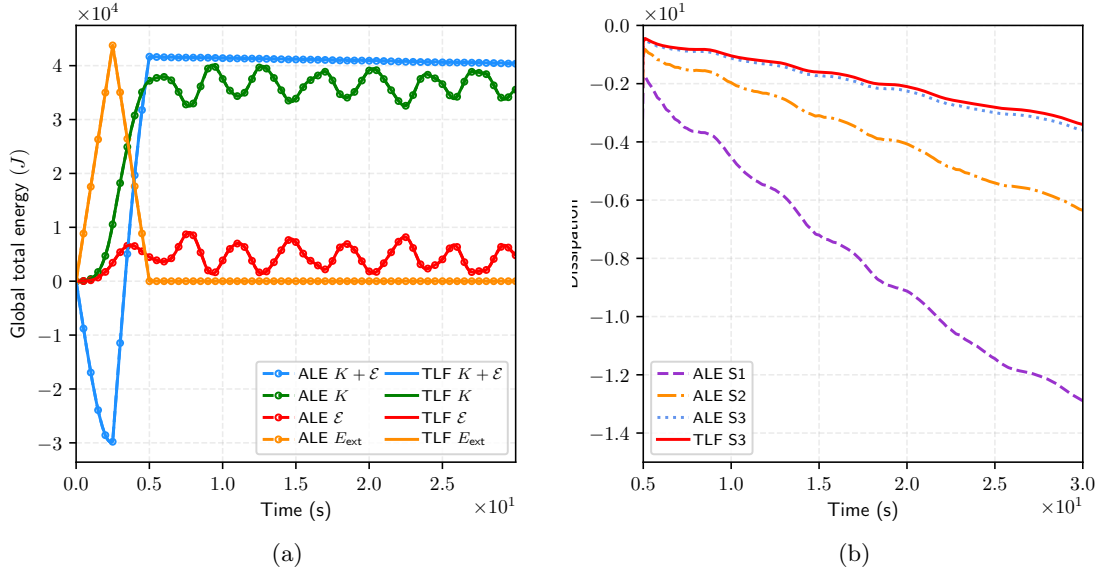


FIGURE 8.18: L-shaped block: Time evolution of a) the global energy components for mesh S3, and b) the dissipation for meshes S1, S2 and S3, and the ALE $\{p_\chi, F_\Phi, F_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition 8.5. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

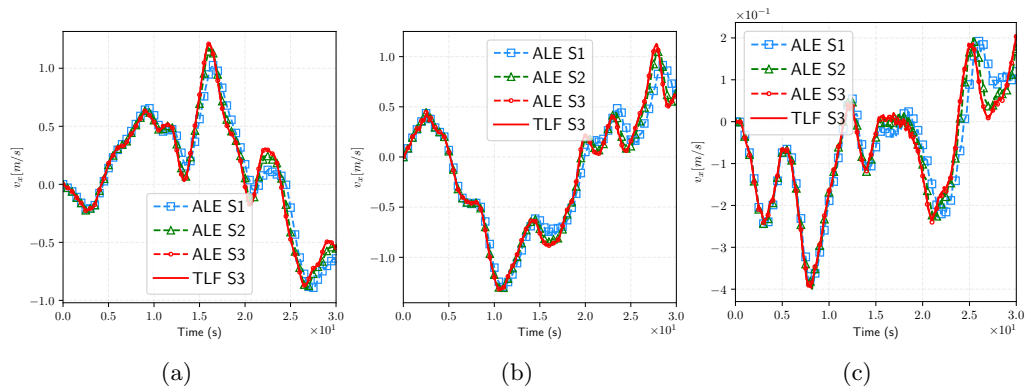
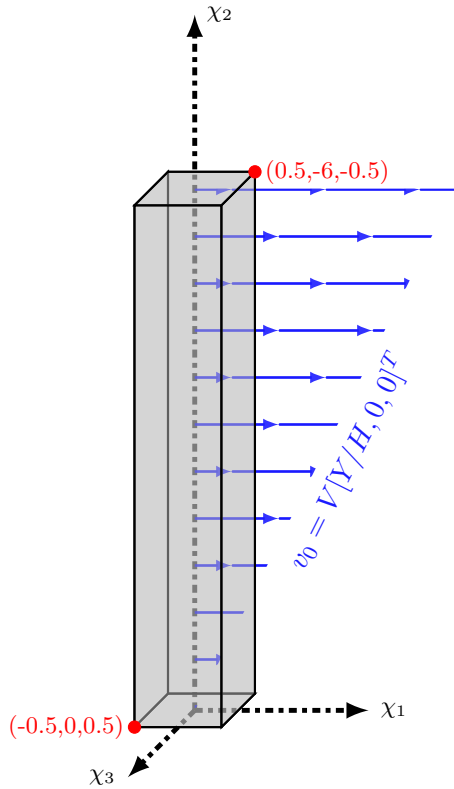


FIGURE 8.19: L-shaped block: Time evolution of the velocity components at a) point $A = (0, 10, 0)^T$, b) point $B = (6, 0, 0)^T$, and c) point $C = (3, 3, 3)^T$ for the ALE $\{p_\chi, F_\Phi, F_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition 8.5, and using meshes S1, S2 and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

8.5 Nearly incompressible bending column



face $\chi_2 = 0$	Moving boundary (6.29)
all remaining faces	free boundary (6.32)

TABLE 8.14: Summary of boundary conditions.

FIGURE 8.20: Bending column: Problem setup.

The main objective of this example is to demonstrate the performance of the proposed scheme in nearly incompressible bending dominated scenarios. A 1 m squared cross section column (see Figure 8.20) clamped at the bottom and free on all other sides is presented. The column is subjected to a bending deformation by means of an initial linearly varying profile in the X-Y plane given by $\mathbf{v}_0 = V[Y/H, 0, 0]^T$ m/s, where $V = 10$ m/s and $H = 6$ m is the height of the column. A polyconvex hyperelastic neo-Hookean model is considered with parameters summarised in Table 8.15. The volumetric potential is based on [239] defined in Appendix B.3.2.

Young's modulus	E [MPa]	17
density	ρ_R [kg/m ³]	1100
Poisson's ratio	ν	0.45

TABLE 8.15: Bending column: Table of parameters.

The results will be compared for the Total Lagrangian $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ formulation, and for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. In the case of the ALE formulation, the mesh motion is prescribed with Equation (8.1), where parameters are listed in Table 8.16.

period	T [s]	1
magnitude	β [m/s]	1×10^{-2}

TABLE 8.16: Translation: Table of parameters for the prescribed mesh motion in Equation (8.1).

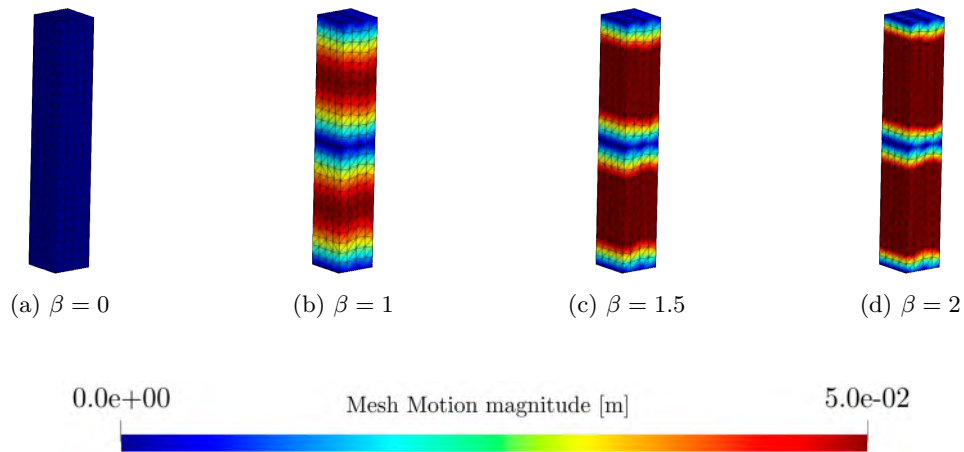


FIGURE 8.21: Bending column: Snapshots of the mesh deformation at time $t = 0.25$ s for different values of the parameter $\beta = 0, 1, 1.5, 2$ (left to right, top to bottom) along with contours of the mesh motion magnitude when solving for the $\{p_{\chi}, \mathbf{F}_{\Phi}, \mathbf{F}_{\Psi}, J_{\Psi}\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

Figure 8.21 shows the spatial deformation of the bending column at time $t = 0.25$ s with different magnitude of mesh motion. Due to the mesh motion contour plot, it can clearly be seen that increasing the value of the β coefficient to 2 leads to a situation where the mesh motion is non null almost everywhere. In Figure 8.15, snapshots of deformations are shown together with the pressure distribution. In Figure 8.23a, the total energy E , kinetic energy K and internal energy \mathcal{E} are compared using mesh S3 for the ALE formulation and the equivalent Total Lagrangian formulation. It can be seen that the numerical dissipation is comparable in both cases, despite the introduction of mesh motion. In Figure 8.23b, the evolution of the total energy is compared for the three meshing using the ALE formulation, and for the third mesh using the equivalent Total Lagrangian formulation. The numerical dissipation in the ALE scenarios converge to what is obtained in the Total Lagrangian scenario. For comparison purposes, the spatial velocity and displacement x component at points $A = (0.5, 5, 0.5)^T$ (see Figure 8.20) is reported in Figure 8.24 for the ALE method using different meshes and for the Total Lagrangian formulation.

In this example, it was shown that in the presence of a mesh motion, results are satisfyingly converging to what can be obtained with a Total Lagrangian formulation. In particular, the energy is conserved and measured quantities such as local velocities converge when the mesh is refined.

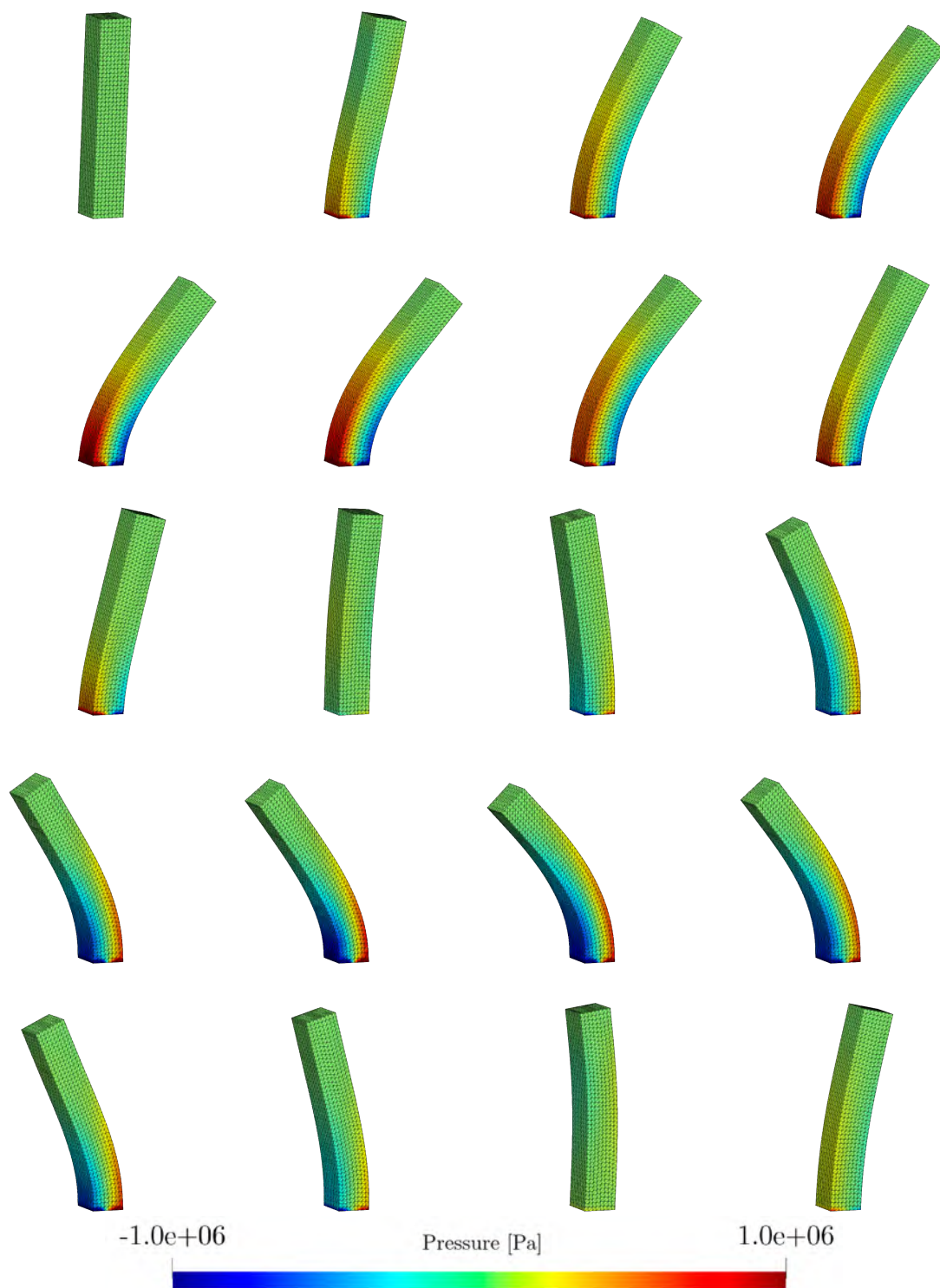


FIGURE 8.22: Bending column: Time evolution of the deformation at time $t = 0, 0.1, 0.2, \dots, 1.9$ s (left to right, top to bottom) along with the pressure distribution when solving for the $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained with the impulse boundary condition (8.5), and using mesh S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

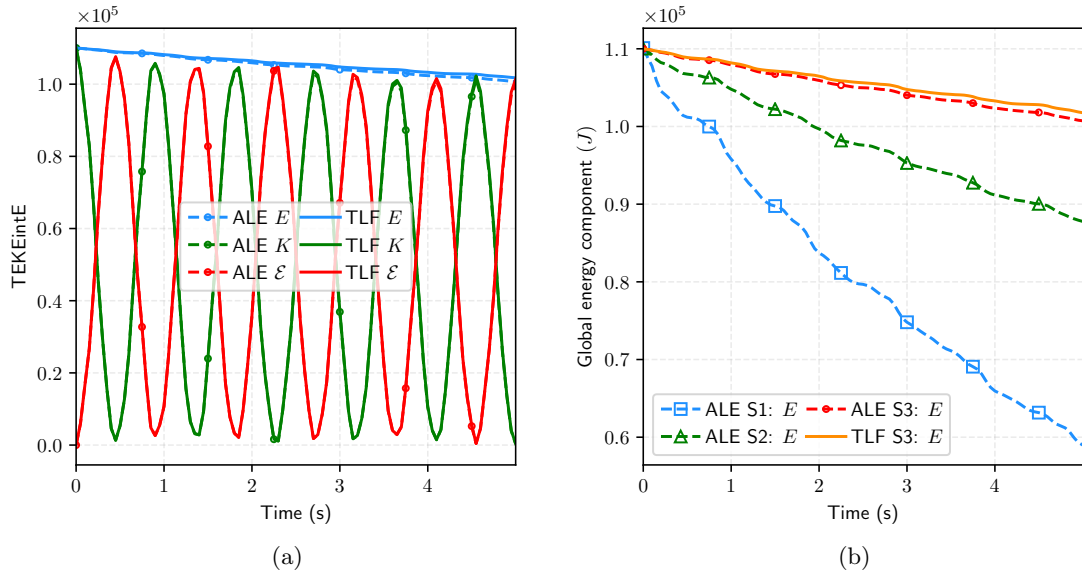


FIGURE 8.23: Bending column: Time evolution of a) the global energy components on mesh S3, and b) the total energy on meshes S1, S2 and S3, for the Total Lagrangian formulation and the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. A neo-Hookean model and the volumetric potential in [239] is used with parameters in Table 8.15. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.16.

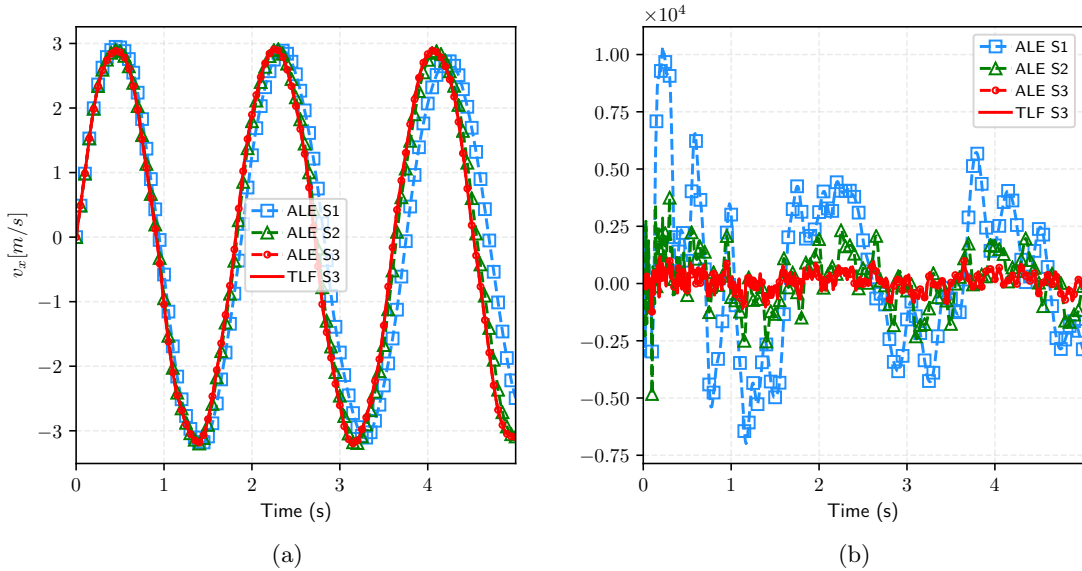


FIGURE 8.24: Bending column: Time evolution of the spatial a) velocity x component and b) the displacement x component at point $A = (0.5, 6, 0.5)^T$ for the Total Lagrangian formulation and for the ALE $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ formulation. Results are obtained on meshes S2, S2 and S3. A neo-Hookean model and the volumetric potential in [239] is used with parameters in Table 8.15. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.16.

8.6 Frictionless impact of a copper bar on a rigid wall.

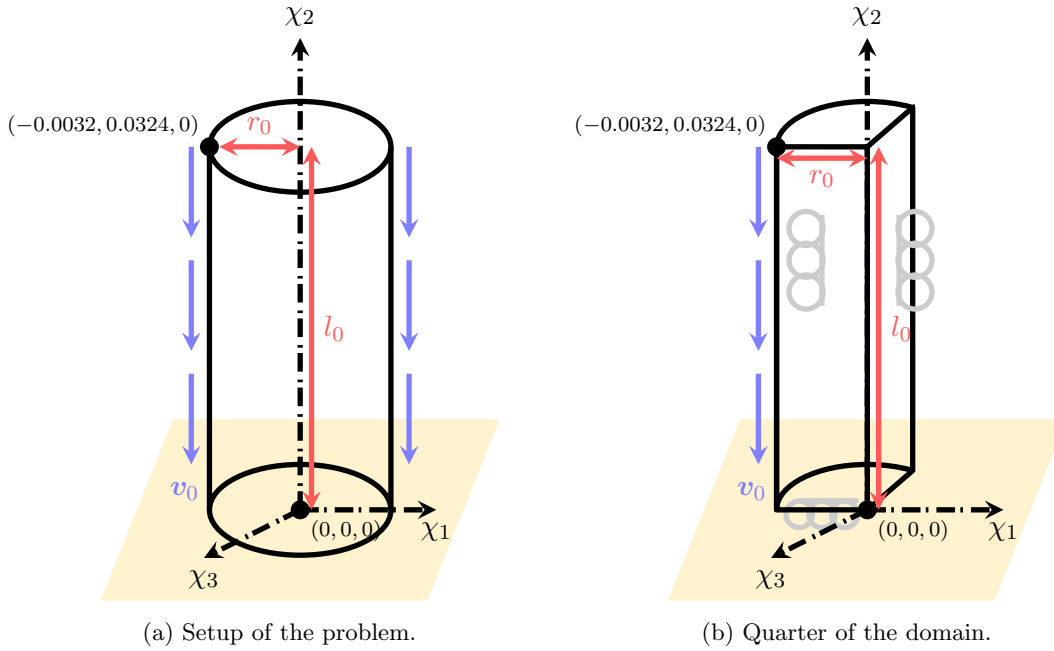


FIGURE 8.25: Taylor impact bar: Sketch of the problem.

face $\chi_1 = 0$ and $\chi_3 = 0$	symmetric boundary (6.35)
all remaining faces	free boundary (6.32)

TABLE 8.17: Summary of boundary conditions.

The Taylor impact is a classical benchmark inspired from pioneering works [297] in the study of yield stress of materials. It has been numerically studied [79, 85, 152, 193, 248, 298] extensively, especially on the context of the Finite Volume Method (FVM) [35, 42, 75, 95, 299] and elasto-plasticity [6, 300]. This problem is also featured in some industrial codes benchmark tests [301, 302]. A cylindrical copper bar of radius $r_0 = 3.2\text{mm}$ and height $l_0 = 32.4\text{mm}$ is impacted to a frictionless wall at time $t = 0\text{s}$ with an initial high velocity $\mathbf{v}_0 = [0, -227, 0]^T$ m/s, as depicted in Figure 8.25a. The material is modelled by von-Mises plasticity, based on the Hencky logarithmic strains, a quadratic volumetric potential (see Table B.1) and isotropic linear hardening. All the parameters are listed in Table 8.18.

By making use of the various axis of symmetry of the problem, only one quarter of the domain is considered and appropriate symmetric boundary conditions are applied accordingly, while the remaining boundary is left free (see Figure 8.25b and Table 8.17). There are three meshes used in this case (see Figure 8.26): U1 (1887 nodes and 8100 elements), U2 (3721 nodes and 17280 elements) and U3 (6461 nodes and 31500 elements). A 2-stage Runge-Kutta time integrator and a CFL number of 0.4 were used in the simulations. In this example, the mesh motion is driven by a pure deviatoric \mathbf{P}^* tensor (originally defined in Equation (5.9)), and can be expressed as

$$\mathbf{P}_W = \tilde{\mathbf{F}}^T \mathbf{P}_{\text{dev}}(\tilde{\mathbf{F}}_\Phi); \quad \tilde{\mathbf{F}}_\Phi = \tilde{\mathbf{F}} \mathbf{F}_\Psi, \quad (8.6a,b)$$

Young's modulus	E [MPa]	117
Material density	ρ [kg/m ³]	8930
Poisson's ratio	ν	0.35
Initial Yield Stress	$\bar{\tau}_y^0$ [GPa]	0.4
Hardening Modulus	H [GPa]	0.1

TABLE 8.18: Taylor impact bar: Table of parameters.

where \mathbf{P}_{dev} is governed by a deviatoric neo-Hookean constitutive law using the physical parameters listed in table 8.19, and where a vertical axial restriction is applied in the natural deformation gradient as

$$\tilde{\mathbf{F}} = \mathbf{I} + (F_{yy} - 1)\mathbf{E}_2 \otimes \mathbf{E}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & F_{yy} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (8.7)$$

ALE parameter 1	α_{ALE}	1
ALE parameter 2	μ_{ALE}	0.02 μ
ALE parameter 3	ϵ_{ALE}	0.001

TABLE 8.19: Taylor bar impact: Table of parameters for the mesh motion.

The conservation of the material velocity \mathbf{W} featured in Equation (5.3) is exploited here in order to minimise spatial mesh distortion. The computation of the stresses is balanced between the (solved) material \mathbf{F}_{Ψ} and spatial \mathbf{F}_{Φ} deformation gradients to aim at an improved solution accuracy at a potentially lower cost. In Figure 8.27, the spatial and the material deformation of mesh U1 is reported for several values of the ALE parameter α_{ALE} . The case where $\alpha_{\text{ALE}} = 1$ has the most amount of mesh motion and will be the case studied in this example, as stated in Table 8.19.

A refinement study is done in Figure 8.28, where a snapshot of the deformation at the final time $t = 80 \mu\text{s}$ is shown for the ALE formulation using meshes U1, U2 and U3.

	final Δt	speed-up ratio
TLF U1	5.822×10^{-10}	
TLF U2	3.267×10^{-10}	
TLF U3	2.031×10^{-10}	
ALE U1	3.407×10^{-9}	5.851
ALE U2	1.895×10^{-9}	5.802
ALE U3	1.078×10^{-9}	5.308

TABLE 8.21: Taylor bar impact: speed-up chart comparing the values of time increments for the Total Lagrangian and the ALE formulation. Figures are rounded to the third decimal. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

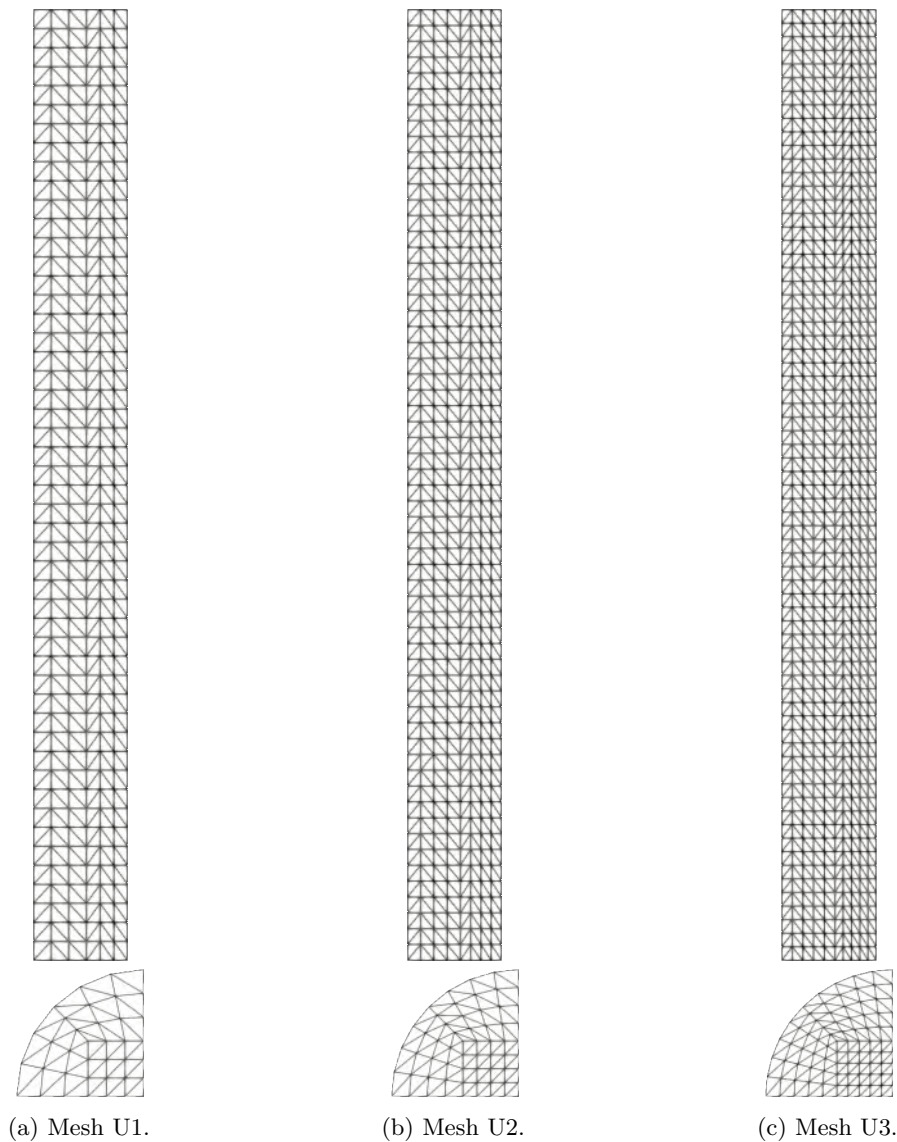


FIGURE 8.26: Taylor impact bar: Meshes U1, U2, and U3.

Results are very sensitive to the efficiency of the formulations in recovering the high amount of plasticity in the vicinity of the impact plane. In this experiment, the deformation of the specimen is based on linear hardening and can be observed through the evolution of plastic internal variables such as the equivalent plastic strain, as well as load measures such as the von-Mises stress and the hydrostatic pressure. These measures are crucial to assess the accuracy and robustness of the ALE formulation and the numerical stabilisation. Figure fig. 8.30 show the evolution of the von-Mises stress for several time snapshots, comparing the Total Lagrangian formulation (right side) with the ALE formulation (left side) for meshes U1, U2 and U3. Figure fig. 8.29 show the evolution of the equivalent plastic strain for several time snapshots, comparing the Total Lagrangian formulation (right side) with the ALE formulation (left side) for meshes U1, U2 and U3. The evolution of the pressure field can be viewed via the snapshots reported in Figure fig. 8.31 for the three respective meshes. The results using the Total Lagrangian formulation (right side) are compared to those using the ALE formulation (left side). For the different meshes of the quarter cylinder, and using both the Total Lagrangian formulation and the ALE formulation, the total energy is featured in Figure 8.32a, and a breakdown of its components, namely the kinetic energy K , the elastic strain potential \mathcal{E} and the plastic dissipation w_p , is shown in

Standard FEM [206]	4-node tetrahedra	5.5
Standard FEM [206]	8-node tetrahedra	6.95
Average nodal pressure FEM [206]	4-node tetrahedra	6.99
Newark FEM [303]	hexahedra (972)	≈ 7
Split FEM [304]	tetrahedra	7.07-7.33
Mixed FEM [304]	hexahedra	7.11
Mixed JST VCFVM [35]	4-node tetrahedra	6.98
Mixed PG-FEM [79]	4-node tetrahedra	7.00
Mixed CCFVM [107]	hexahedra (480)	6.88-7.11
SPH Reference Updated Lagrangian [37]	SPH 1560 particles	6.75
SPH Reference Updated Lagrangian [37]	SPH 3744 particles	6.84
SPH Reference Updated Lagrangian [37]	SPH 7280 particles	6.89
VCFVM ALE	tetrahedra mesh U1	6.720
VCFVM ALE	tetrahedra mesh U2	6.797
VCFVM ALE	tetrahedra mesh U3	6.824

TABLE 8.20: Taylor bar impact: final bottom radii chart for the Total Lagrangian and the ALE formulation using different meshes. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

Figure 8.32b. The kinetic energy decreases upon impact because it first transforms into elastic strain energy, and is rapidly converted into plastic dissipation when plasticity is yielded. Naturally, a minor amount of kinetic energy is also dissipated due to the proposed numerical algorithm. An important parameter to measure the deformation of the bar after impact is the radius of the bar throughout the simulation. It is measured with tracking the displacement magnitude at the same bottom extremity node for all meshes and using both simulation. The evolution of that bottom radius is reported in Figure 8.33a and the final value of that radius can be compared with other published results in Table 8.20. The results, obtained with both formulations using the solver presented in this work, are exempt of volumetric locking that can sometimes be present in standard and/or commercial solutions. Additionally, the evolution of the time increment featured in Figure 8.33b can be used to appreciate the gain in temporal resources offered by the ALE formulation. The time speed-up is reported in Table 8.21.

The ALE formulation was shown to capture the high plastic strain in the impact region as well as the Total Lagrangian formulation. This shows that the numerical discretisation is perfectly capable of dealing with problems with high plasticity without numerical instabilities or volumetric locking. In fact, the use of the ALE formulation leads to smoother deformation patterns, and if the control of mesh deformations can be left to the user's appreciation via the ALE parameters, it detects in the areas with high plastic loads and performs the mesh motion automatically. As the geometry is refined, the results converge to values coherent with the those reported in the literature and the numerical dissipation is reduced. This validates the efficiency and consistency of the stabilisation technique implemented in the algorithm.

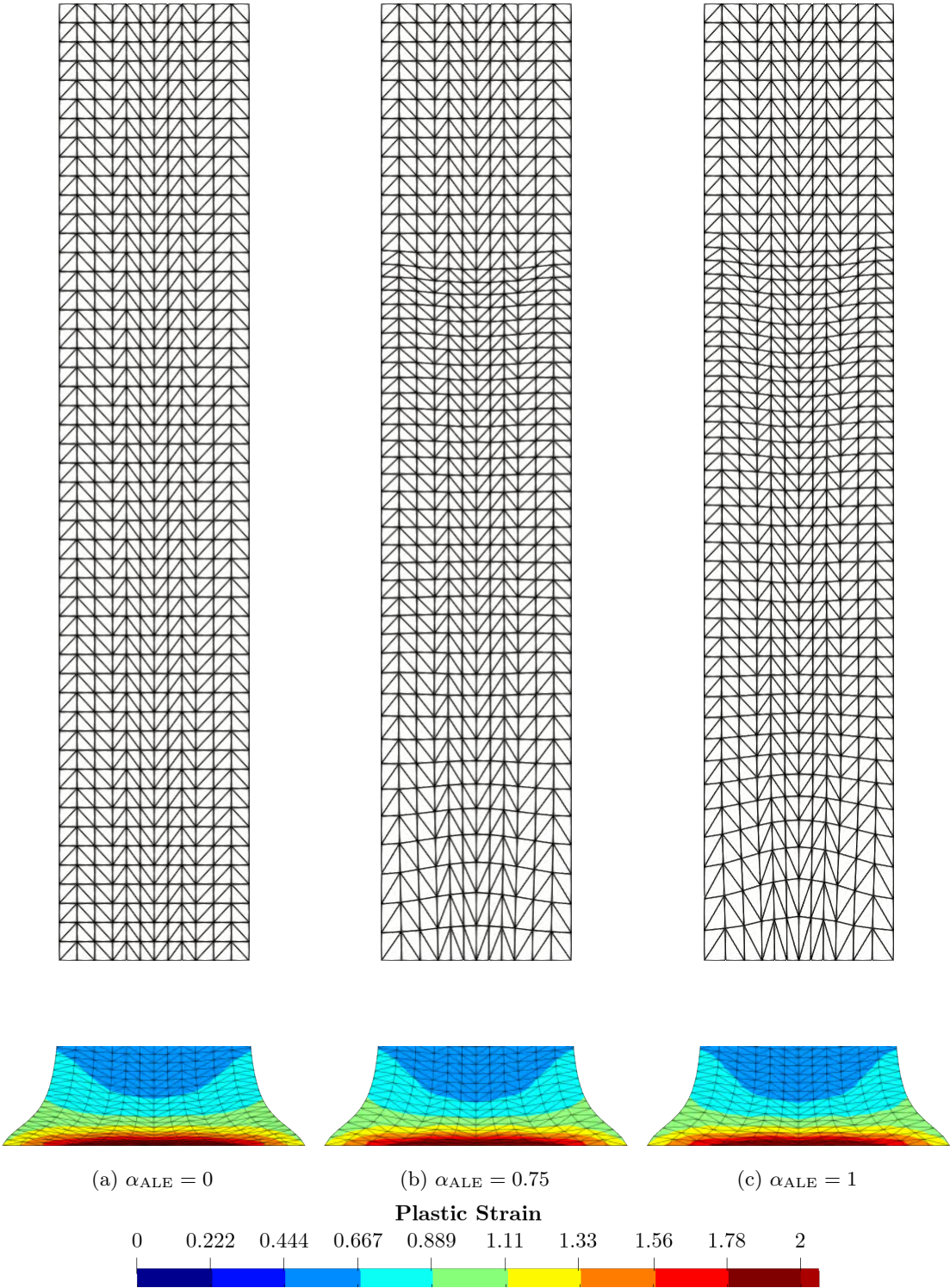


FIGURE 8.27: Taylor bar impact: Snapshots of the material deformation (top row) and spatial deformation (bottom row, zoomed view of the lower section) using the ALE formulation for $\alpha_{ALE} = 0, 0.5, 0.75, 1$ on mesh U1. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

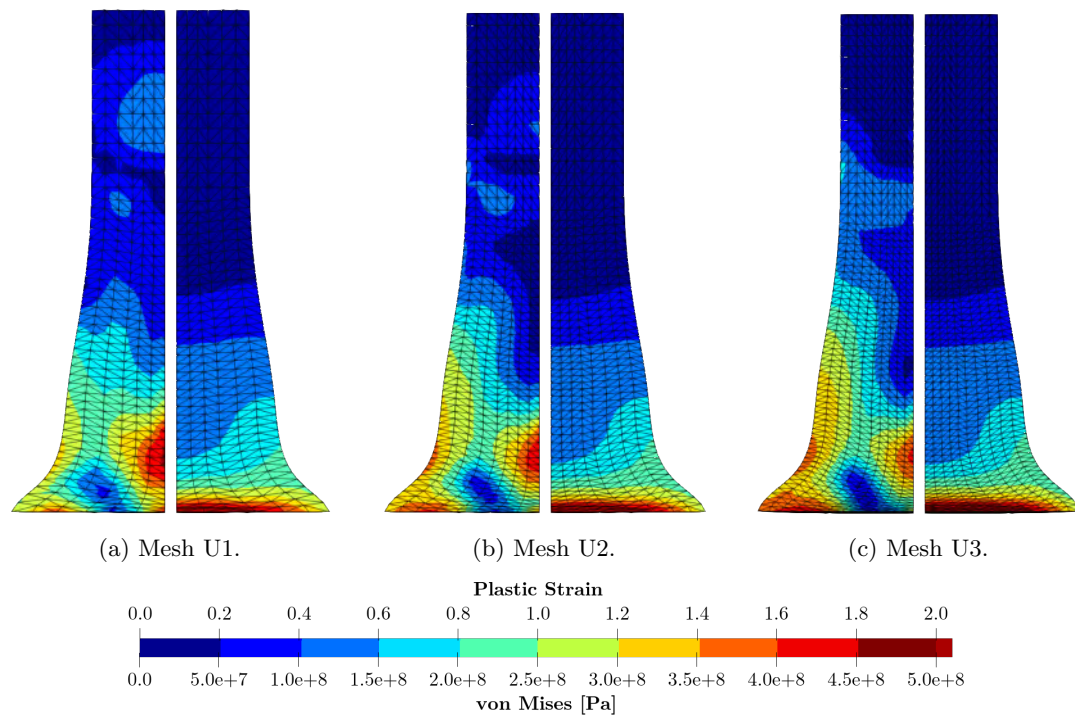


FIGURE 8.28: Taylor bar impact: Refinement analysis at time $t = 80 \mu\text{s}$ using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

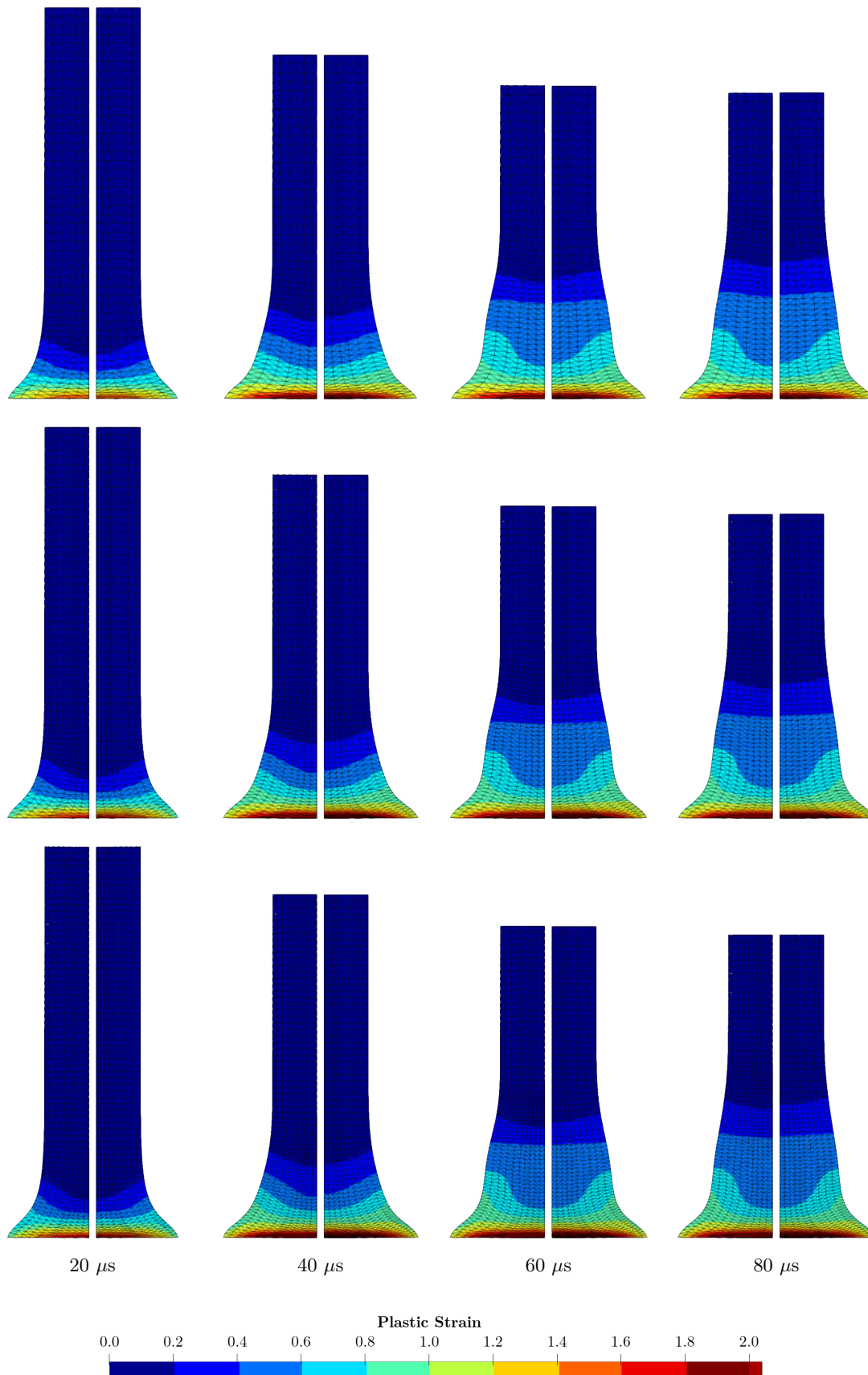


FIGURE 8.29: Taylor bar impact: Snapshots of the equivalent plastic strain at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

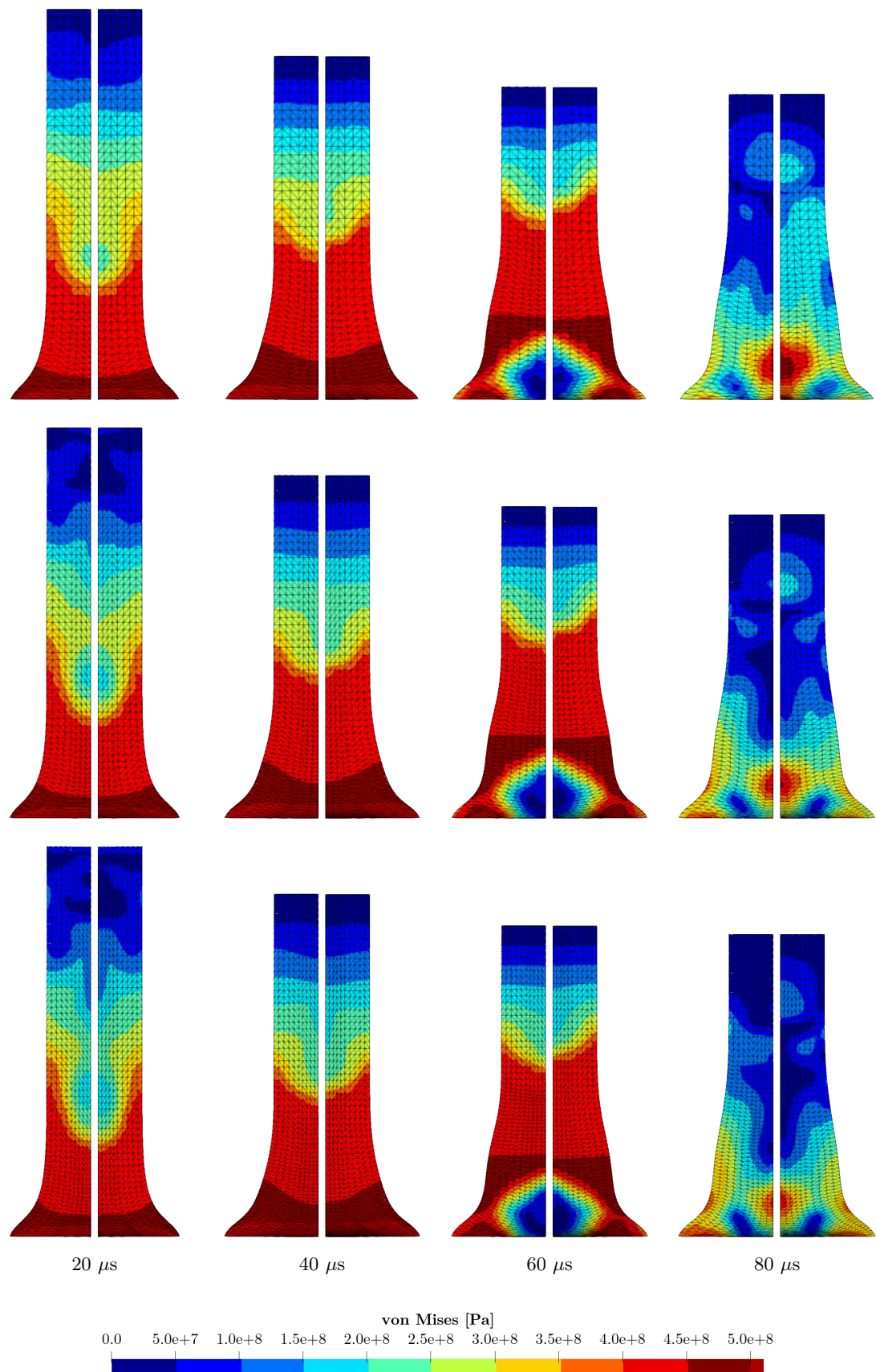


FIGURE 8.30: Taylor bar impact: Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu\text{s}$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

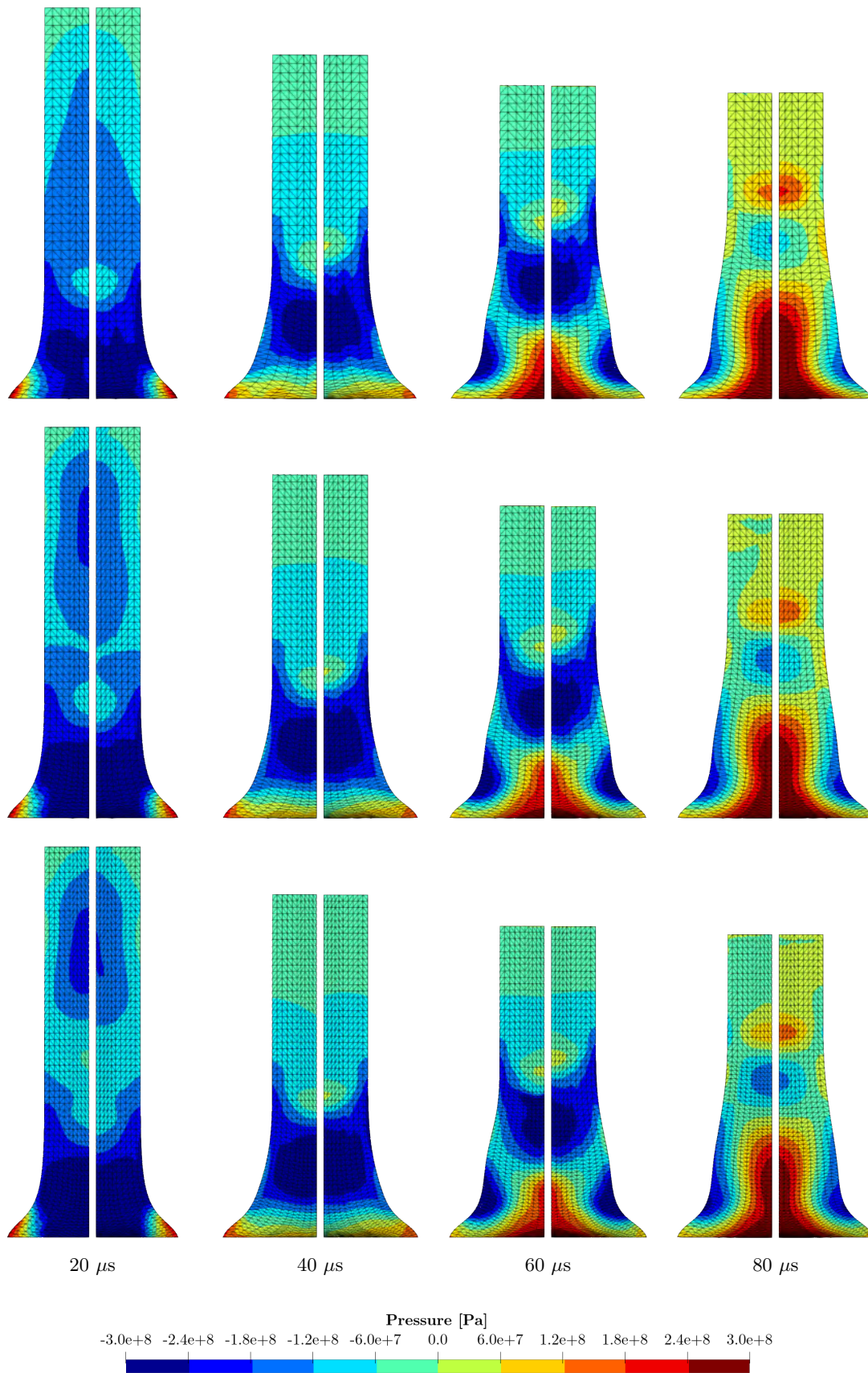


FIGURE 8.31: Taylor bar impact: Snapshots of the deformation with pressure contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

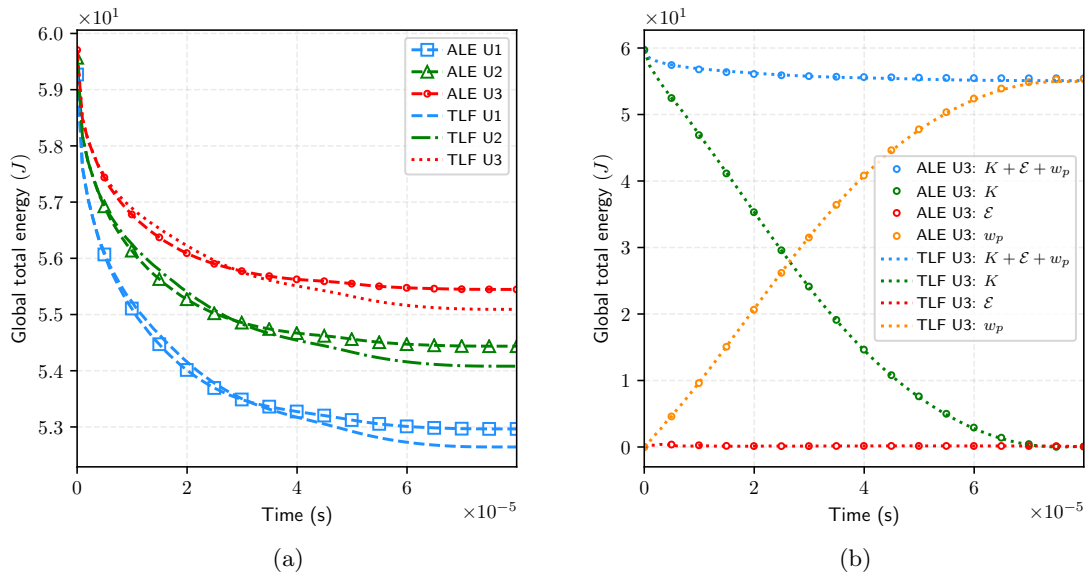


FIGURE 8.32: Taylor bar impact: (a) Total energy $E = E_{\chi}/J_{\Psi}$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

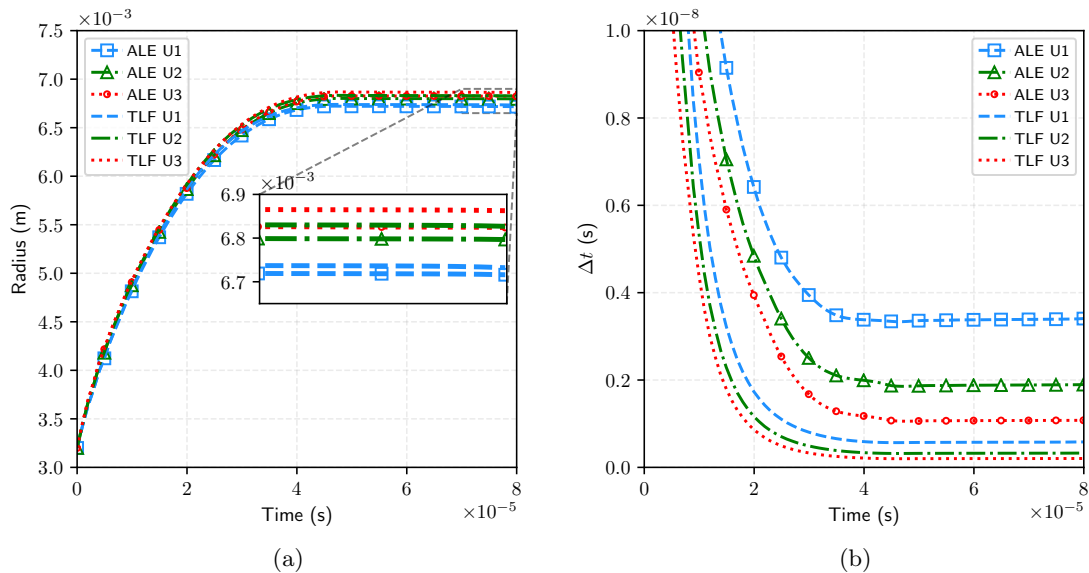


FIGURE 8.33: Taylor bar impact: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

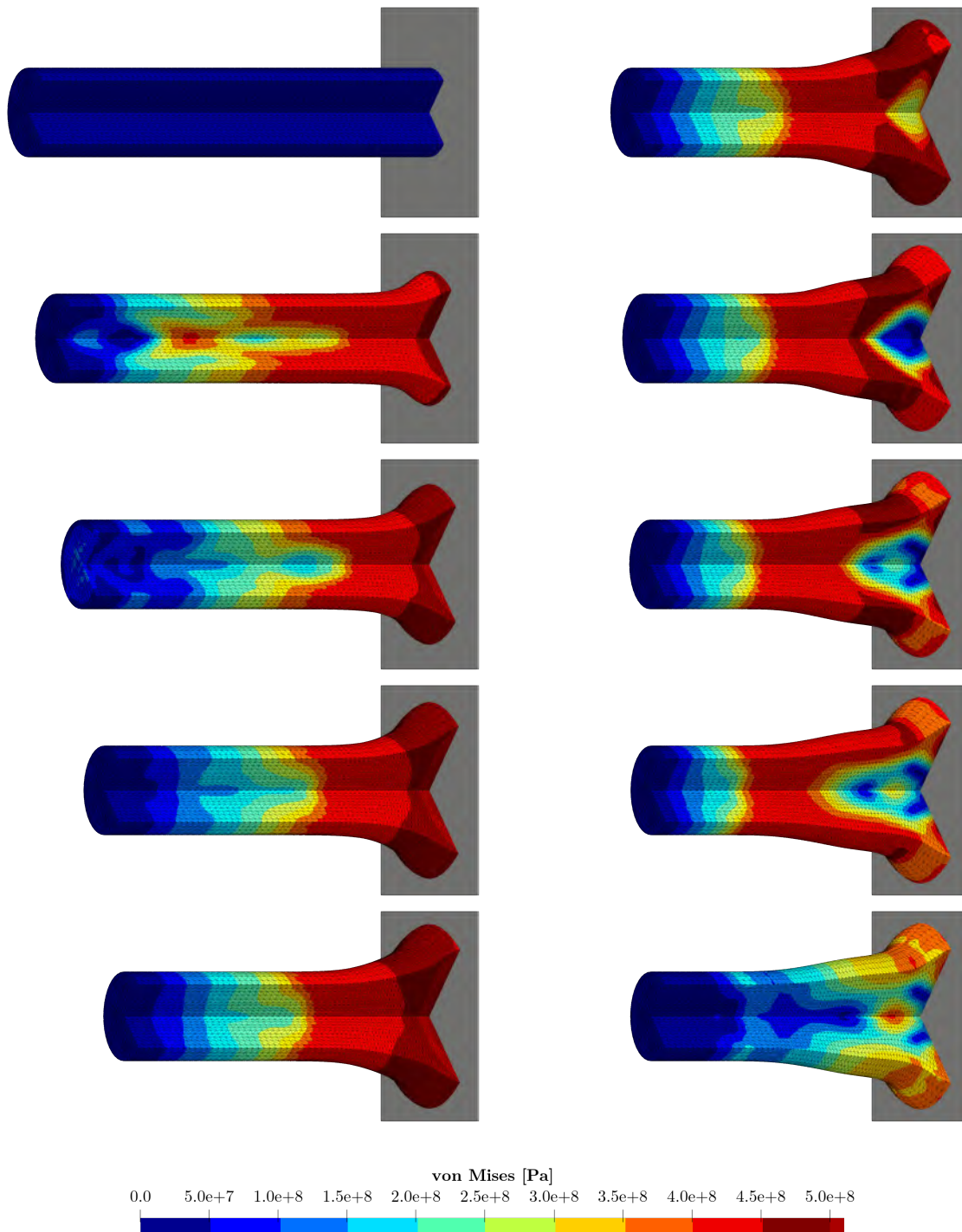


FIGURE 8.34: Taylor bar impact: Comparison of spatial deformation together with von-Mises contour plot at time $t = 0, 10, 20, 30, 40 \mu\text{s}$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu\text{s}$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, nonlinear hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

8.7 Necking of a circular copper bar

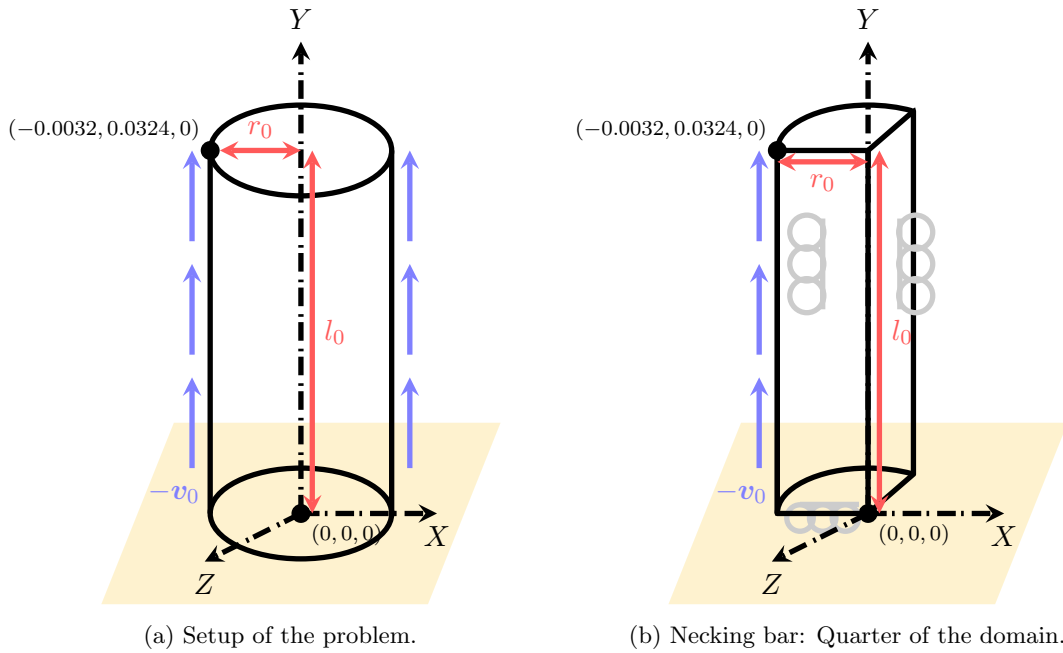


FIGURE 8.35: Necking bar: Sketch of the problem.

faces $\chi_1 = 0$ and $\chi_3 = 0$	symmetric boundary (6.35)
all remaining faces	free boundary (6.32)

TABLE 8.22: Summary of boundary conditions.

In this example, a cylindrical bar is pulled from the top to induce necking in the bottom area, denoted as the neck. In this setup, the necking is created by a fast dynamic pull of the top surface, initialised with a horizontal velocity $-\mathbf{v}_0 = -V[0, 1, 0]^T$ with $-V = 227\text{m/s}$. The Hencky-based von-Mises plasticity is used to model the body, with parameters equivalent to the impact case summarised in Table 8.18. The volumetric potential is based on the [239] model defined in Appendix B.3.2. As only a quarter of the domain is considered for obvious symmetry reasons, the boundary conditions are equivalent to those used for the dynamic impact problem, and are shown in Table 8.22.

The mesh motion, whose parameters are listed in Table 8.23, is governed by a deviatoric neo-Hookean incompressible potential, and the function $\tilde{\mathbf{F}}$ presented in Equation (8.7) is used. The simulations are conducted with a 2-stage RK integrator, and a fixed CFL

ALE parameter 1	α_{ALE}	1
ALE parameter 2	μ_{ALE}	0.02μ
ALE parameter 3	ϵ_{ALE}	0.001

TABLE 8.23: Necking bar: Table of parameters for the mesh motion in Equation (8.1).

number $\alpha_{CFL} = 0.4$ was used.

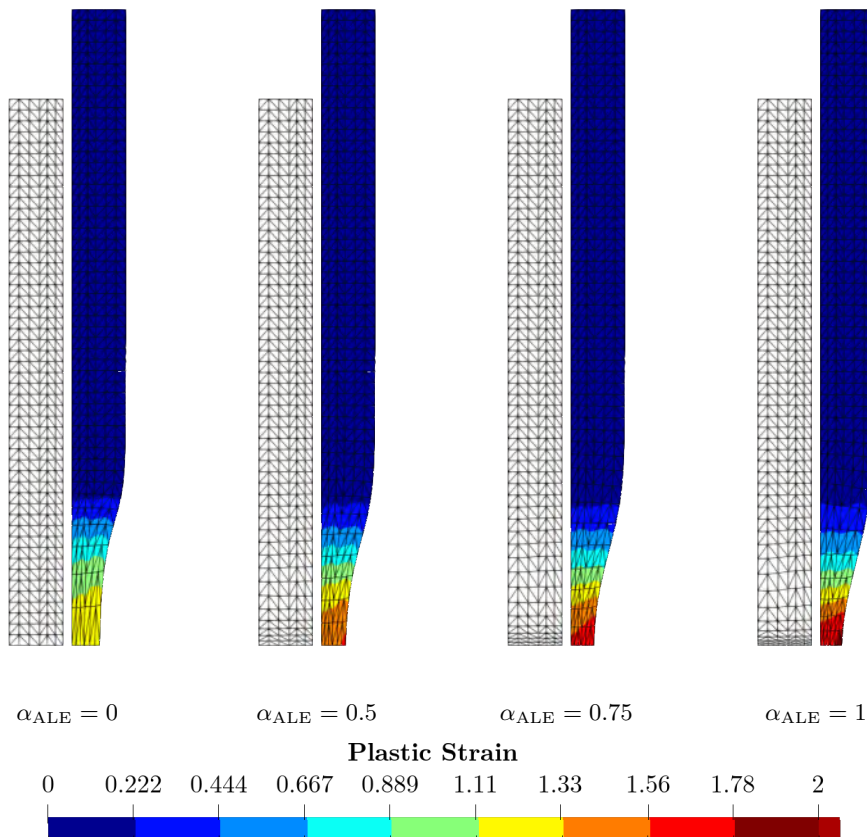


FIGURE 8.36: Dynamic axisymmetric necking: Spatial (right side) and material (left) deformation at final time $t = 25 \mu\text{s}$, using the ALE formulation with an ALE parameter $\alpha_{ALE} = 0, 0.5, 0.75, 1$ and $\mu_{ALE} = 1$, on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

First of all, the mesh deformation is studied for several values of the parameter α_{ALE} in Figure 8.36. As that coefficient grows, it can clearly be seen that elements are pulled back to the necking region. Moreover, the neck profile improves with higher values of that coefficient, and more plastic strain is recovered in the vicinity of the neck. These observations motivated the use of the value $\alpha_{ALE} = 1$ for the rest of the study. In Figure 8.37, the deformation along with the plastic strain and von-Mises contours can be observed on mesh U1. Snapshots are shown for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row). There is a significant difference in the deformation pattern because of the presence of more elements in the neck region. Moreover, the distribution of plastic variables is different. The same observations can be made for both mesh U2 and mesh U3. Figure 8.38 and on Figure 8.39. Note that when using the Total Lagrangian formulation, elements are distorted and lose quality in the neck region. By using the ALE formulation, the elements keep a higher quality for a longer time, and the results can be exploited for a longer physical time. The presence of numerous elements in the neck region results in a finer integration of the energy in that area, as if the mesh was locally refined. The evolution of the energy components is shown in Figure 8.41a, and the evolution of the total energy is shown in Figure 8.41b. It can be seen that the conservation of energy is of better quality when using the ALE formulation. The components of the energy, namely the kinetic energy, the internal energy, and the plastic dissipation, are conserved in a good manner for both formulations. The deformation around the neck region, and more especially the shape of the boundary, is clearly improved with the ALE

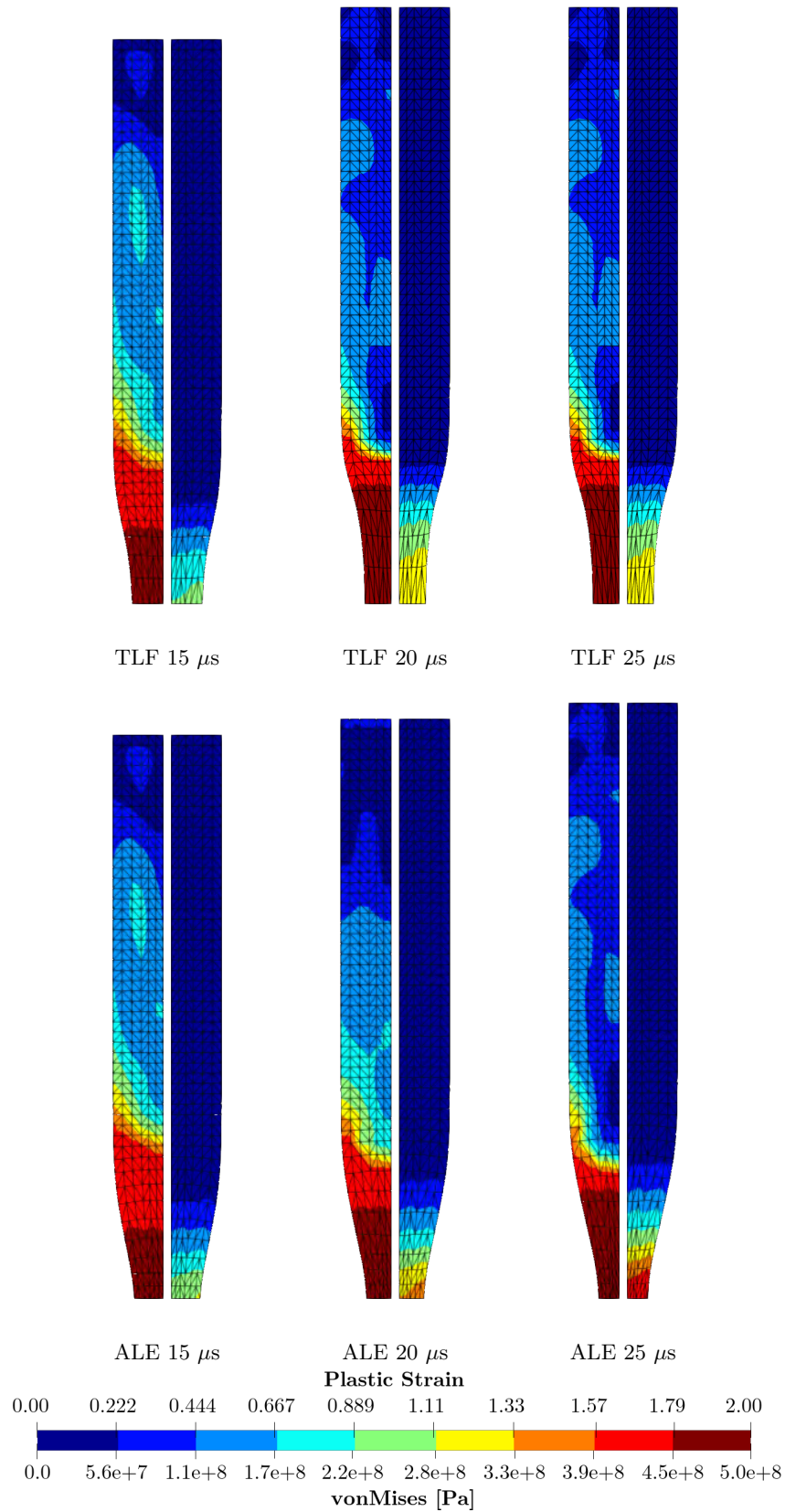


FIGURE 8.37: Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times 15, 20, 25 μ s, using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

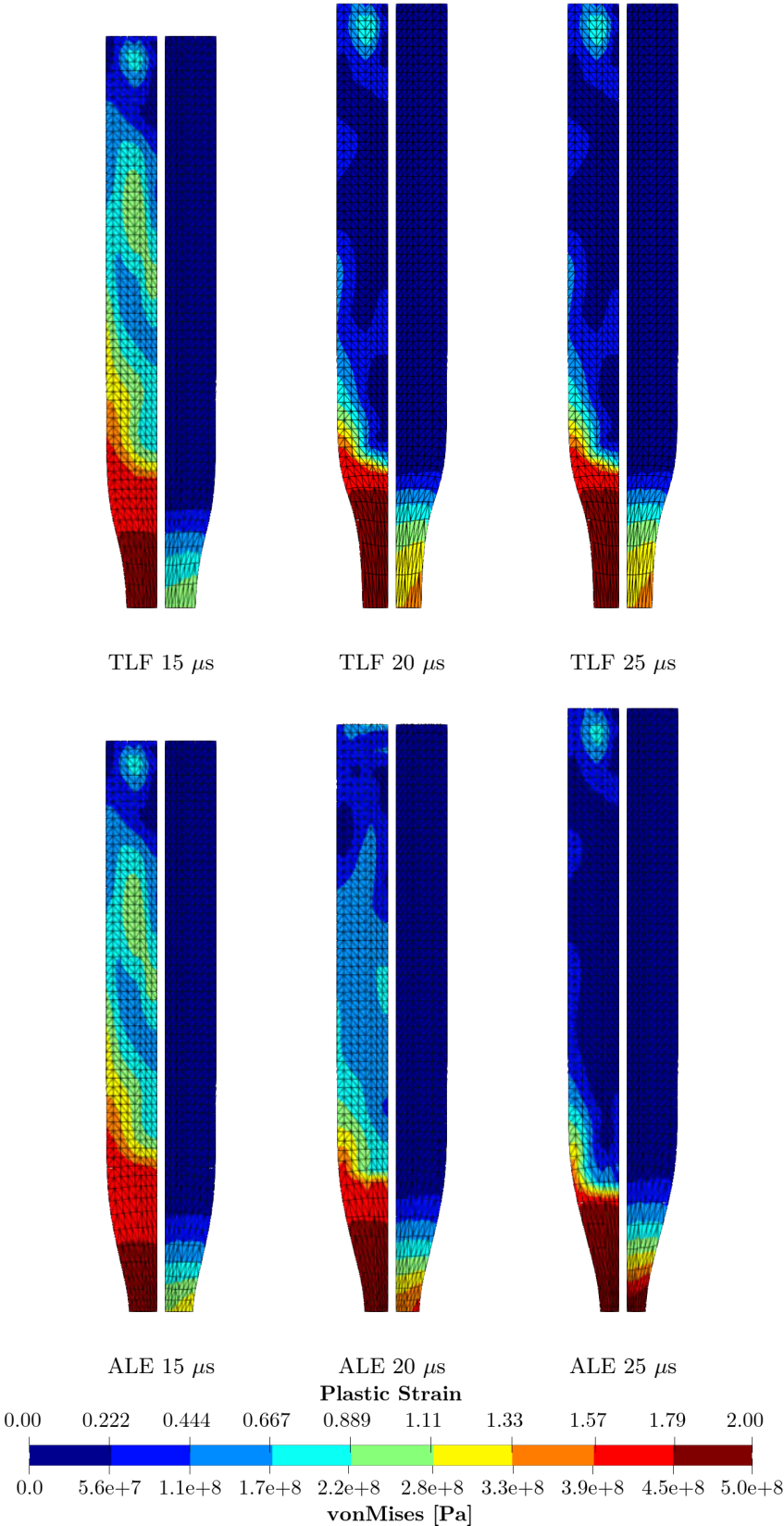


FIGURE 8.38: Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times 15, 20, 25 μs , using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U2. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

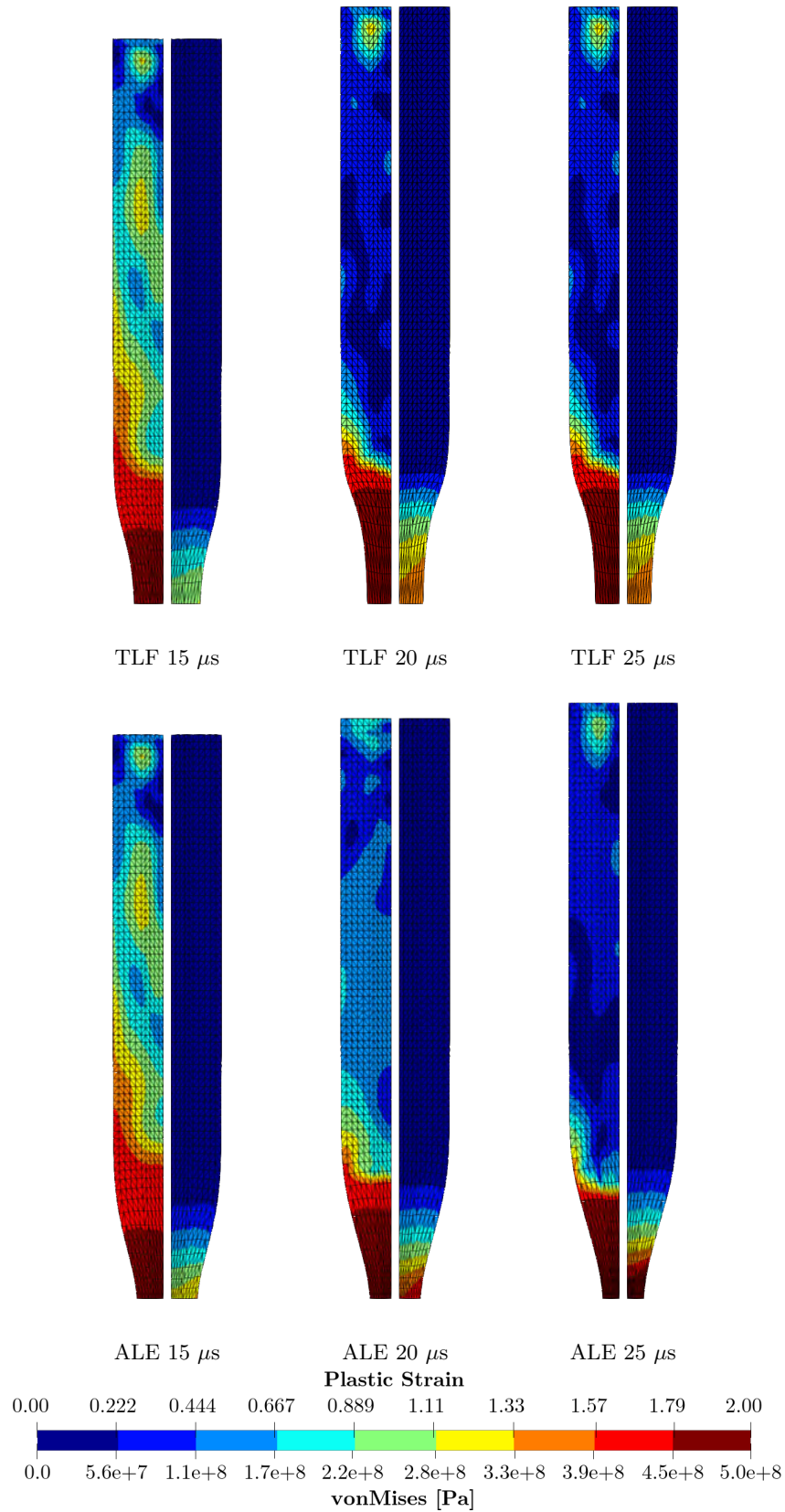


FIGURE 8.39: Dynamic axisymmetric necking: Snapshots of von-Mises stress (left side) and equivalent plastic strain (right side) contours at times 15, 20, 25 μs , using the Total Lagrangian formulation (top row) and the ALE formulation (bottom row) on mesh U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

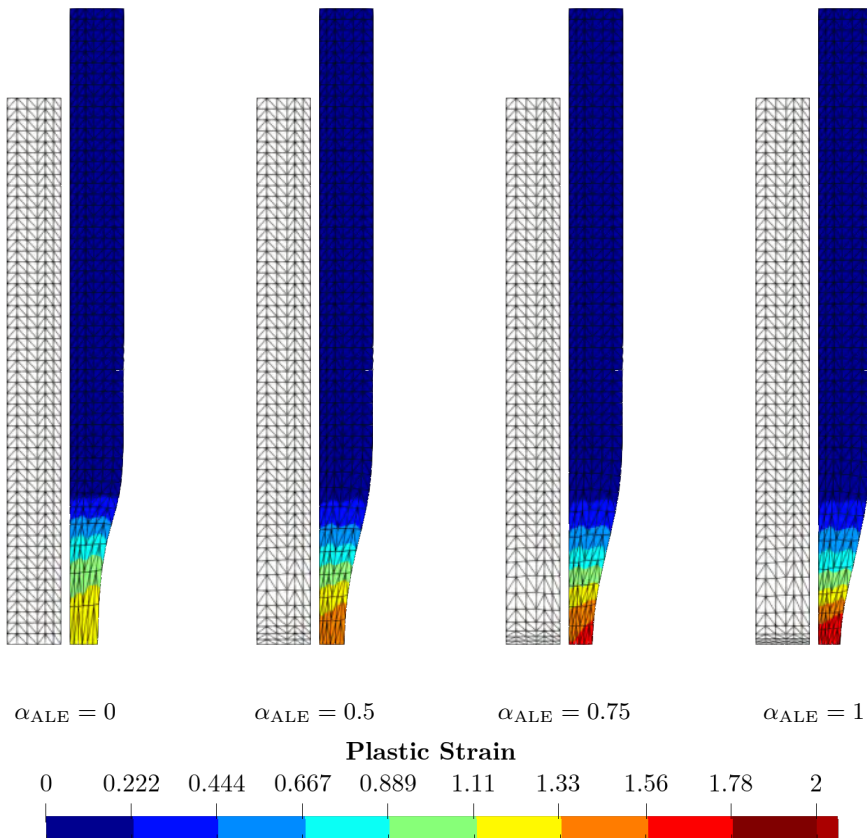


FIGURE 8.40: Dynamic axisymmetric necking: Spatial (right side) and material (left) deformation at final time $t = 25 \mu s$, using the ALE formulation with an ALE parameter $\alpha_{ALE} = 0, 0.5, 0.75, 1$ and $\mu_{ALE} = 0.02$, on mesh U1. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

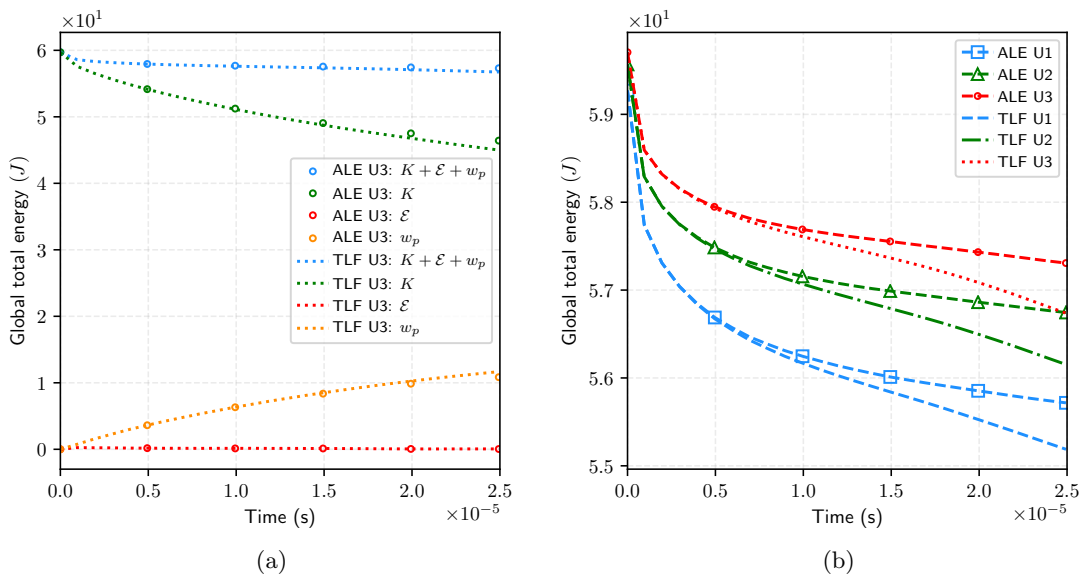


FIGURE 8.41: Dynamic axisymmetric necking: (a) energy components, for the Total Lagrangian and the ALE formulation on mesh U3, and (b) Total energy $E = E_{\chi} / J_{\Psi}$ on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

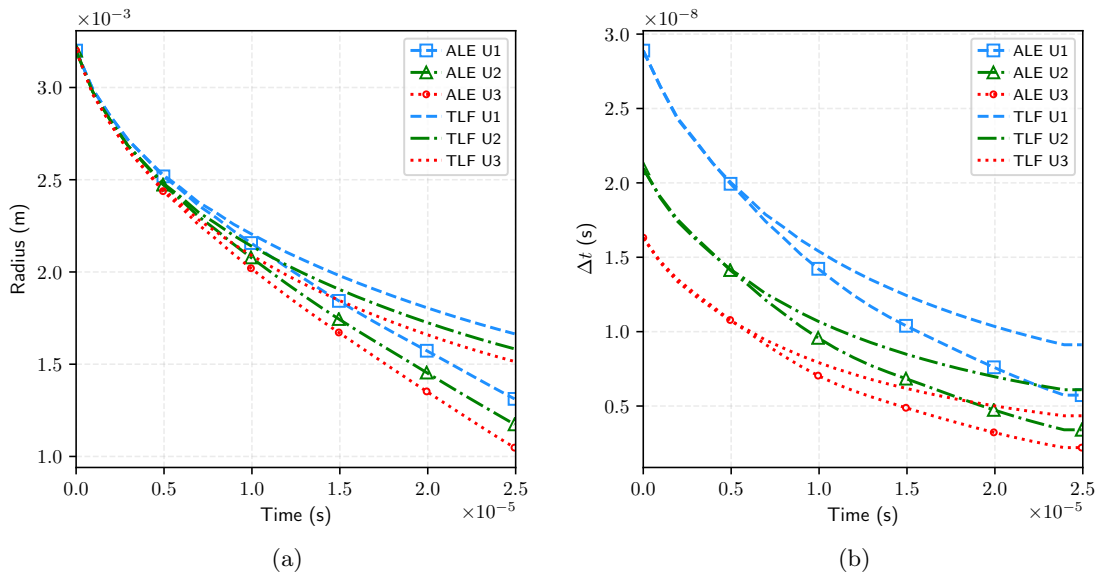


FIGURE 8.42: Dynamic axisymmetric necking: Time evolution of a) bottom radius, and b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain and von-Mises plasticity with material parameters summarised in Table 8.18.

formulation. On Figure 8.42a, the radius is shown to keep decreasing in a linear manner as the bar is pulled. Naturally, if more elements are kept in the neck region, they are smaller than in the Total Lagrangian scenario. As a consequence, the time increment is expected to be smaller than for a Total Lagrangian simulation, as observed on Figure 8.42b. To conclude the observations on this dynamic necking case, it was observed that the behaviour of the ALE formulation is the opposite behaviour as what was observed in the impact case for the same ALE parameters. The elements are pulled downward to the neck region. As a consequence, the boundary has a better shape but the computational time will suffer from smaller elements that will be linked to smaller time increments. Nevertheless, the ALE formulation was shown to convergence in this necking scenario, and to agree well with the results obtained with the Total Lagrangian equivalent formulation. The advantage of using ALE here resides in the possibility to get the advantage of a local refinement dynamically.

CHAPTER 9

NUMERICAL SIMULATIONS: THERMOELASTICITY AND THERMOPLASTICITY

“Mi verdad básica es que todo tiempo es un ahora en expansión”

— SEVERO OCHOA

9.1 Introduction

In the following, thermo-mechanical numerical examples are simulated. Isothermal cases presented in Chapter 8 are converted to scenarios with thermal effects. The thermo-elastic cases are based on the Mie-Grüneisen equation of state, and make use of a neo-Hookean model for the non thermally-coupled deviatoric stress. The volumetric potentials are thermally coupled. For plastic materials, a thermal visco-plastic Johnson-Cook model is also used.

9.2 Translation column

This translation problem is adapted from the isothermal case: the mechanical parameters are the same as in isothermal and a uniform temperature, equal to a reference temperature, is considered in the specimen. The parameters are summarised in the table 9.1. The value of the coefficient Γ_0 , equal to 0.0255, is taken from the L-Shaped block problem in Bonet et al. [23]. The thermal expansion coefficient is obtained as $\alpha = c_v \Gamma_0 / 3\kappa$. The thermally-coupled volumetric potential is based on the [239] model defined in Appendix B.3.2.

Young's modulus	E [MPa]	17
density	ρ_R [kg/m ³]	1100
Poisson's ratio	ν	0.45
thermal conductivity	h [W.m ⁻¹ K ⁻¹]	10
specific heat capacity	C_v [J kg ⁻¹ K ⁻¹]	1
thermal expansion coefficient	α [K ⁻¹]	4.95×10^{-7}
reference temperature	θ_R [K]	293.15

TABLE 9.1: Translation with heat: Table of physical parameters.

This example is the thermo-mechanical equivalent of the Case 8.2. The objective of this example is to demonstrate the importance of the solving the discrete geometric conservation laws in the context of a thermo-mechanical deformation. Henceforth, the specimen geometry is presented in Figure 8.2, the constitutive model considered is a polyconvex hyperelastic neo-Hookean model with the volumetric potential presented in [239] and featured in Table B.1 Mie-Grüneisen Equation of state. The required physical parameters are presented in Table 9.1. Velocities are uniformly initialised to $v_0 = [3, 1, 0]^T$ m/s and the boundary surfaces are all left free. The structured mesh S1 presented in Figure 8.3 is used in the numerical simulations. The mesh motion is prescribed according to Equation (8.1), together with the parameters in Table 8.3. A comparison of the results is made when solving for a different number of geometric conservation laws. The objective is to demonstrate that in the context of a non-zero material mesh motion, the numerical framework must solve additional geometric conservation laws to recover the accuracy of the equivalent simulation using a Total Lagrangian formulation (i.e. without mesh motion).

In Figures 9.1 and 9.2, snapshots of the simulation at time $t = 0, 0.8, 1.6, 2.4, 3.2, 4.0, 4.8$ s can be viewed based on a) the formulation $\{p_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi\}$ and b) $\{p_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, J_\Phi\}$. This case is similar the translation test done in Chapter 8.2, and is conducted with a framework that now includes thermal effects. The results are reported for a scenario where the spatial Jacobian J_Φ is solved, so the natural Jacobian is reconstructed as $J = J_\Phi J_\Psi^{-1}$. However, the incorporation of the conservation law on the spatial Jacobian is not important

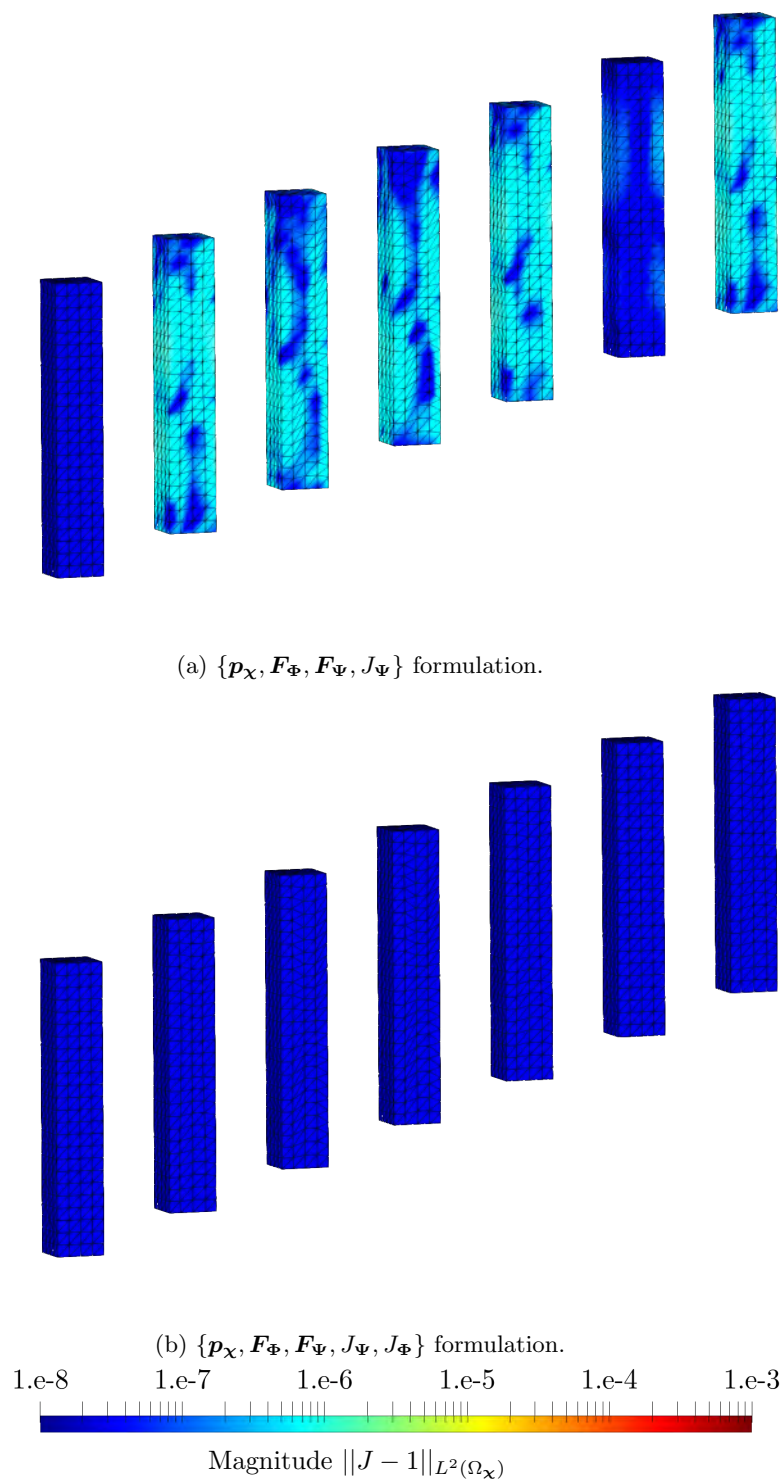


FIGURE 9.1: Translation with heat: Snapshots of the deformation at time $t = 0, 0.8, \dots, 4.8$ s, along with distribution $\|J - 1\|_{L^2(\Omega_x)}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

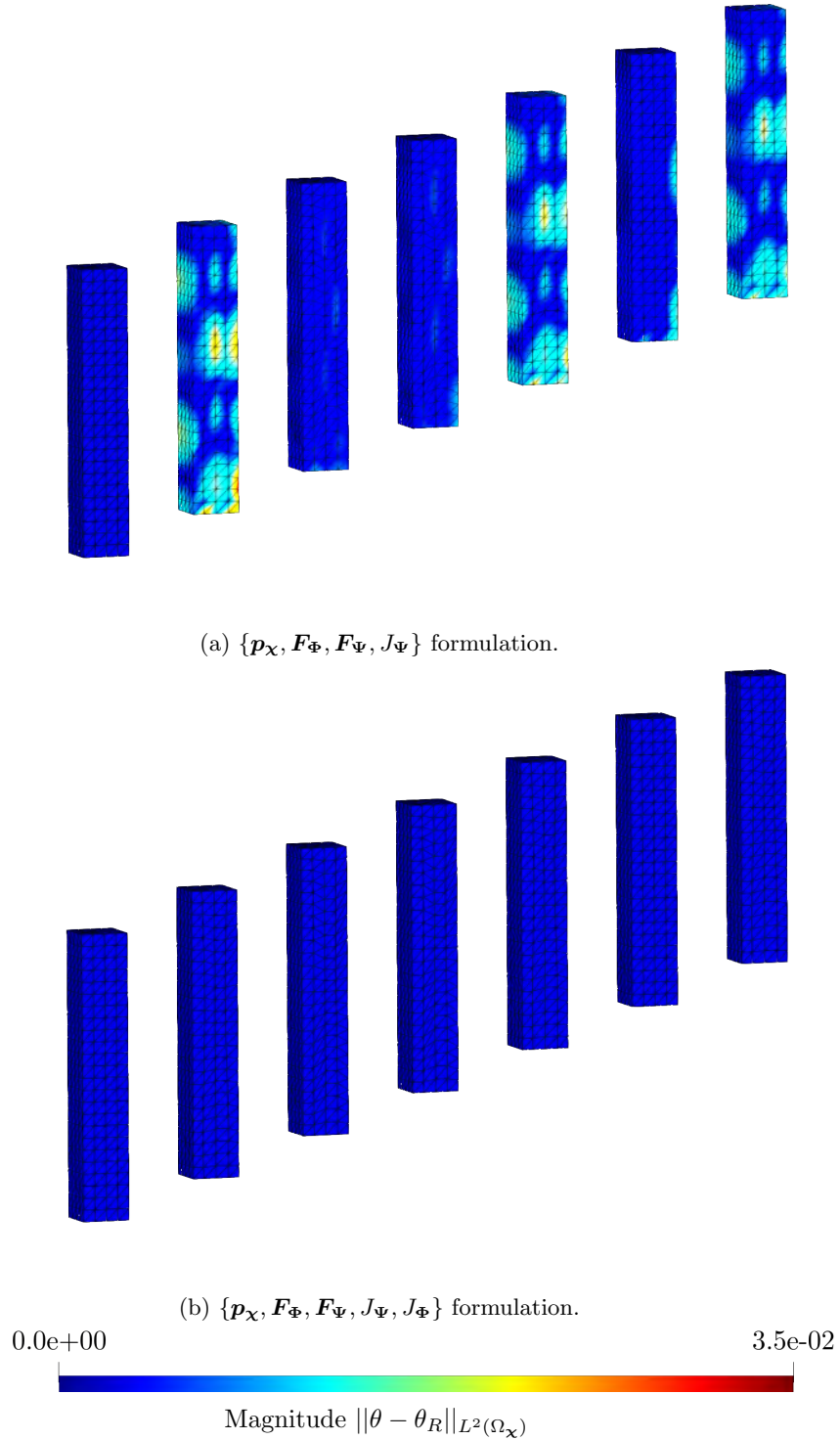


FIGURE 9.2: Translation with heat: Snapshots of the deformation at time $t = 0, 0.8, \dots, 4.8$ s, along with distribution $\|\theta - \theta_R\|_{L^2(\Omega_x)}$ when solving for different conservation laws. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

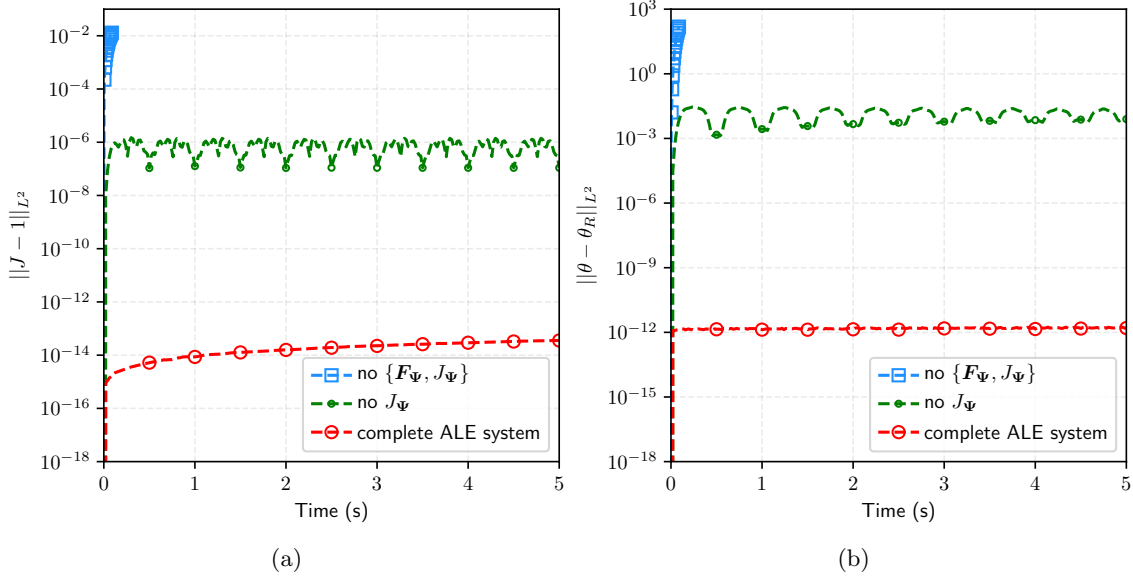


FIGURE 9.3: Translation with heat: Time evolution of the L^2 -error in Jacobian and temperature when using (red) the complete ALE system, (green) a reduced system without solving variable J_Ψ and (blue) a further reduced system without solving variables F_Ψ, J_Ψ . Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

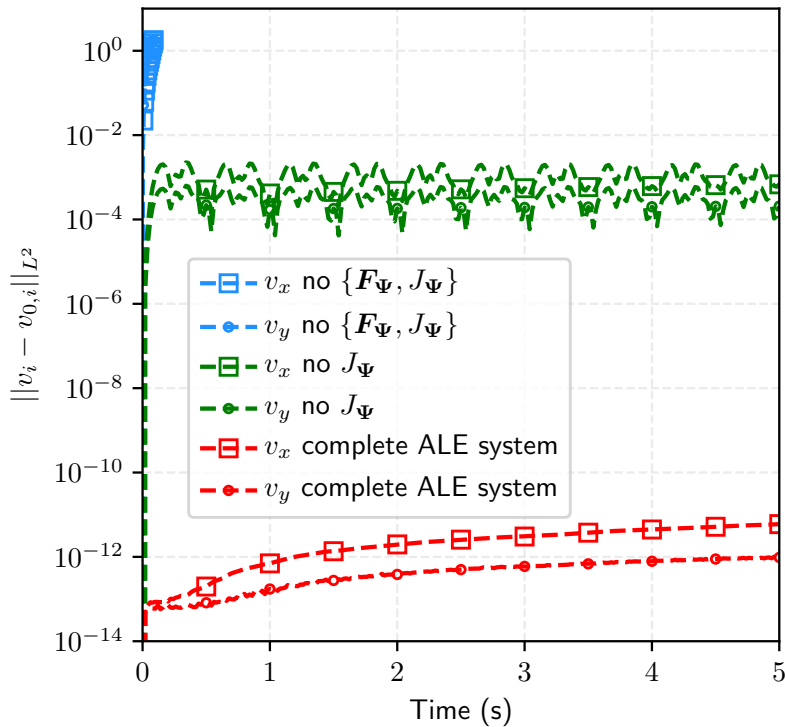


FIGURE 9.4: Translation with heat: Time evolution of the L^2 error in velocity components when using (red) the complete ALE system, (green) a reduced system without solving variable J_Ψ and (blue) a further reduced system without solving variables F_Ψ, J_Ψ . Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

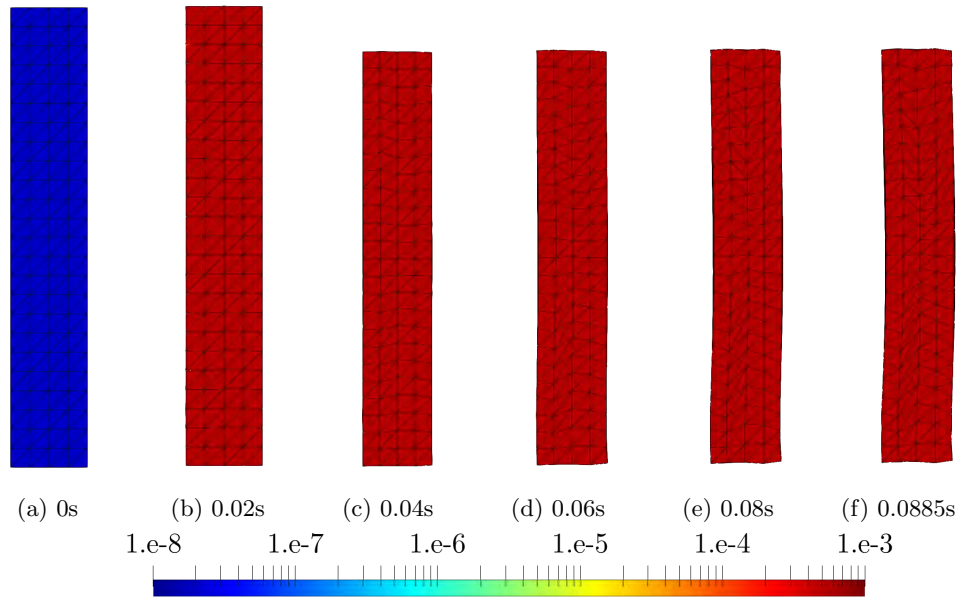


FIGURE 9.5: Translation with heat: Snapshots of the deformation at time $t = 0, 0.02, 0.04, 0.06, 0.08, 0.0885$ s, along with distribution $\|J - 1\|_{L^2(\Omega_\chi)}$ when solving for $\{\mathbf{p}_\chi \mathbf{F}_\Phi J_\Phi\}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

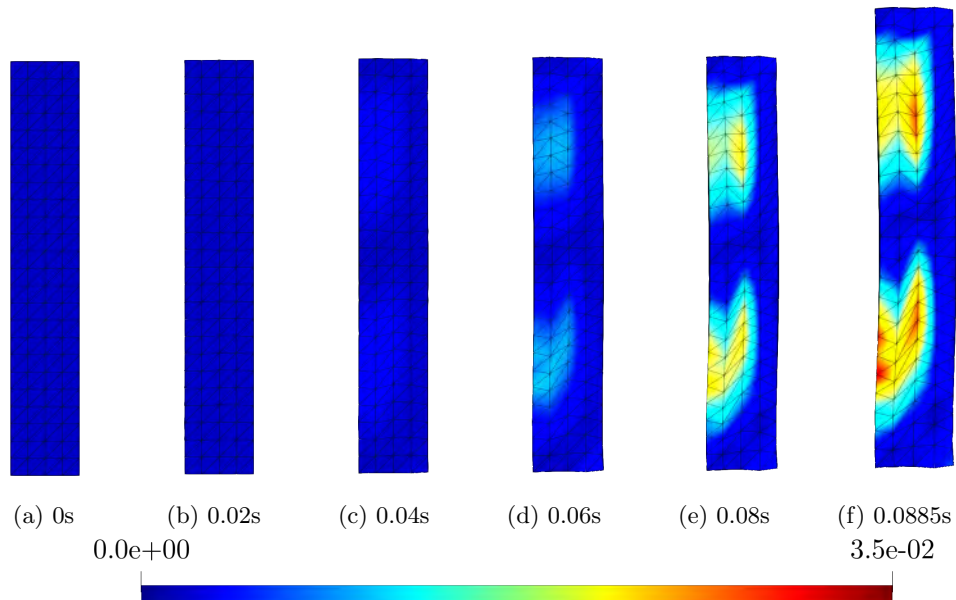


FIGURE 9.6: Translation with heat: Snapshots of the deformation at time $t = 0, 0.02, 0.04, 0.06, 0.08, 0.0885$ s, along with distribution $\|\theta - \theta_0\|_{L^2(\Omega_\chi)}$ when solving for $\{\mathbf{p}_\chi \mathbf{F}_\Phi J_\Phi\}$. Results obtained with velocity $\mathbf{v}_0 = [3, 1, 0]^T$ m/s using mesh S1. A neo-Hookean model and a Mie-Grüneisen equation of state are used with parameters listed in Table 9.1. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.3.

and does not alter this test. In the second scenario, it is clear that solving for the material deformation gradient lets the numerical simulation perform until the end, with notable instabilities in the pressure distribution and the velocity components. In the third scenario, solving for both the material deformation gradient and the material Jacobian lets the body keep its initial straight trajectory as well as minimum instabilities in the Jacobian distribution. Those visual observations are accompanied by a study of the L^2 -error in the natural Jacobian, the distribution of temperature magnitude in Figure 9.3 and the velocity components in Figure 9.4. The first scenario where no geometric conservation law is solved corresponds to the blue curve, the second scenario where the material deformation gradient is solved corresponds to the green curve, and the third scenario where both the material deformation gradient and the material Jacobian are solved corresponds to the red curve. As it is the case in the isothermal problem (see Case 8.2), the best quality results are obtained in the third scenario. When only solving for the $\{\mathbf{p}_\chi, \mathbf{F}_\Phi\}$ formulation, it is clear that the accuracy and the stability are greatly reduced, and the numerical framework is unable to further compute the results in a quick manner. The Figures 9.5 and 9.6 shows the snapshots of the simulation in this case. As it can be seen, instabilities instantly occur throughout the body and the simulation cannot continue.

As a conclusion, the addition of a thermo-mechanical coupling does not alter the requirement to solve for the geometric conservation laws to recover the accuracy proposed by a Total Lagrangian simulation. Moreover, in this thermo-mechanical context, the numerical framework is more sensitive to the inclusion of the Discrete Geometric Conservation Law (DGCL). Indeed, if the simulation could still complete in the isothermal Case 8.2 while still being of bad quality when the material Jacobian is not solved, a setting involving the present thermo-mechanical coupling requires the use of a formulation incorporating the conservation laws for both \mathbf{F}_Ψ J_Ψ . It is clear that the discrete geometric conservation law is required for the ALE framework to be stable, as reported in Farhat et al. [205].

9.3 Swinging cube with heat

This example is a reproduction of the isothermal Swinging Cube problem 8.3, with the additional consideration of thermal effects. This numerical scenario was previously studied in [22]. The objective of this example is to assert the second-order spatial convergence of this ALE framework for thermo-elastic problems. The geometry is a unit length cube presented in Figure 8.8 with the same symmetric and anti-symmetric boundary conditions. However, the polyconvex hyperelastic neo-Hookean model used here is accompanied by a Mie-Grüneisen equation of state, and the volumetric coupling presented in Equation (4.32) is considered. Therefore, the present configuration of the ALE framework accounts for solving the energy conservation law (4.7). This thermoelastic body is subjected to a swing motion identical to that of the isothermal case and that is repeated for convenience as

$$\mathbf{u}(\boldsymbol{\chi}, t) = U_0 \cos(\xi t) \begin{bmatrix} A \sin(\frac{\pi\chi_1}{2}) \cos(\frac{\pi\chi_2}{2}) \cos(\frac{\pi\chi_3}{2}) \\ B \cos(\frac{\pi\chi_1}{2}) \sin(\frac{\pi\chi_2}{2}) \cos(\frac{\pi\chi_3}{2}) \\ C \cos(\frac{\pi\chi_1}{2}) \cos(\frac{\pi\chi_2}{2}) \sin(\frac{\pi\chi_3}{2}) \end{bmatrix}; \quad \xi = \frac{\sqrt{3}}{2} \sqrt{\frac{2\mu + \lambda}{\rho_R} + \frac{\theta_R \Gamma_0^2 c_v}{\rho_R}}. \quad (9.1a,b)$$

Note that the wave speed defined in Equation (9.1b) is different than the wave speed used in the isothermal case, defined in Equation (8.2b), in that an additional term appears because of thermal considerations; if there is no thermal expansion, $\alpha = \Gamma_0 = 0$. The exact temperature profile is then defined as

$$\theta^{\text{exact}}(\boldsymbol{\chi}, t) = \theta_R \left(1 - \frac{3\pi}{2} U_0 \cos(\xi \pi t) \cos\left(\frac{\pi\chi_1}{2}\right) \cos\left(\frac{\pi\chi_2}{2}\right) \cos\left(\frac{\pi\chi_3}{2}\right) \right), \quad (9.2)$$

and other necessary quantities are computed from the two Equations (9.1) and (9.2). The condition $U_0 < 10^{-3}\text{m}$ must be still satisfied for the problem to be considered linear, and for the closed-form solution (9.1) and (9.2) to be a valid approximation of the solution. The parameters used in the physical model are summarised in table 9.2. The thermally-coupled volumetric potential is based on the quadratic model defined in Appendix B.3.2.

Young's modulus	E [MPa]	17
density	ρ [kg/m ³]	1100
Poisson's ratio	ν	0.45
thermal conductivity	h [W/m ⁻¹ K ⁻¹]	10
specific heat capacity	C_v [J kg ⁻¹ K ⁻¹]	1
thermal expansion coefficient	α [K ⁻¹]	4.95×10^{-7}
reference temperature	θ_R [K]	293.15
Magnitude	U_0 [m]	5×10^{-4}
Lame first parameter	λ [MPa]	9.8077
Lame shear modulus	μ [MPa]	6.5385
Mie-Grüneisen coefficient		8.5889

TABLE 9.2: Swinging cube with heat: Table of parameters for the physical model and for the closed-form solution.

The convergence study is performed with the general ALE mixed formulation (2.76) with a mesh motion prescribed by Equation (8.1) and parameters defined in Table 9.3. The

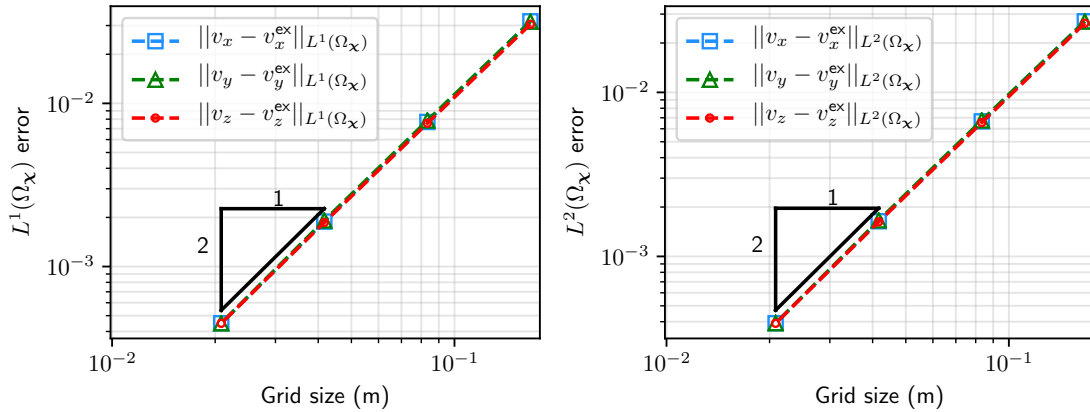


FIGURE 9.7: Swinging cube with heat: L^1 and L^2 global convergence analysis at time $t = 1\text{ms}$ for the components of the velocity, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

results are obtained at the time $t = 1\text{ms}$ and the time is integrated by a 2-stage RK scheme, thence enabling a second-order temporal accuracy.

Period	T [s]	1×10^{-3}
Magnitude	β [m/s]	1×10^{-4}

TABLE 9.3: Swinging cube: Table of parameters for the prescribed mesh motion in Equation (8.1).

In this numerical example, the total energy is used as the conserved thermal variable. The ALE formulation will be compared to its Total Lagrangian formulation.

The convergence rates are reported in Tables 9.4 to 9.7 and show the errors between the results and the analytical closed-form (9.1) and (9.2) for the L^1 and the L^2 norms. As for the isothermal case 8.3, the proposed computational framework achieves equal second order convergence for the velocity, the stress tensor, and additionally the temperature. Also, the meshes S1, S2, S3 and S4 presented in the isothermal case 8.3 are also used in this example. This equal order convergence for all derived variables is one of the advantages of the proposed framework. Using the ALE formulation, the convergence figures using the L_1 norm are reported for the three velocity components and for the temperature in Table 9.4 and for the diagonal components of the first Piola Kirchhoff stress tensor in Table 9.5. Under the same setting, the convergence figures using the L_2 norm are reported for the velocity components and the temperature in Table 9.6 and for the diagonal components of the Piola Kirchhoff stress tensor is shown in Table 9.7. The convergence analysis is plotted in Figure 9.7 for the velocity components, in Figure 9.8 for the diagonal components of \mathbf{P} , and in Figure 9.9 for the temperature. To conclude, the second order in spatial and temporal discretisation is achieved in this thermo-mechanical example, for all the solved variables $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, E\}$, namely the velocity, stresses, and energy. It can be also be noted that similar results are obtained when J_Φ is solved. These results agree well with the literature [22, 23]. The equal order of convergence for all derived variables was an objective of this thesis and motivated the development of the discrete ALE mixed formulation. This is a clear advantage of the proposed framework.

L^1 error	v_x	v_y	v_z	θ
1/6	3.184×10^{-2}	3.171×10^{-2}	3.053×10^{-2}	8.211×10^{-5}
1/12	7.692×10^{-3}	7.851×10^{-3}	7.550×10^{-3}	2.134×10^{-5}
1/24	1.886×10^{-3}	1.916×10^{-3}	1.863×10^{-3}	5.276×10^{-6}
1/48	4.492×10^{-4}	4.491×10^{-4}	4.487×10^{-4}	1.197×10^{-6}
conv. rate	2.070	2.093	2.054	2.141

TABLE 9.4: Swinging cube with heat: numerical values for the dimensionless L^1 -error of the velocity components and the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

L^1 error	P_{xx}	P_{yy}	P_{zz}
1/6	1.861×10^{-2}	1.861×10^{-2}	1.860×10^{-2}
1/12	4.847×10^{-3}	4.847×10^{-3}	4.845×10^{-3}
1/24	1.206×10^{-3}	1.207×10^{-3}	1.206×10^{-3}
1/48	2.817×10^{-4}	2.817×10^{-4}	2.816×10^{-4}
conv. rate	2.098	2.098	2.098

TABLE 9.5: Swinging cube with heat: numerical values for the dimensionless L^1 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

L^2 error	v_1	v_2	v_3	θ
1/6	2.711×10^{-2}	2.704×10^{-2}	2.640×10^{-2}	1.071×10^{-4}
1/12	6.654×10^{-3}	6.746×10^{-3}	6.544×10^{-3}	2.755×10^{-5}
1/24	1.638×10^{-3}	1.660×10^{-3}	1.625×10^{-3}	6.683×10^{-6}
1/48	3.917×10^{-4}	3.916×10^{-4}	3.909×10^{-4}	1.465×10^{-6}
conv. rate	2.064	2.084	2.056	2.189

TABLE 9.6: Swinging cube with heat: numerical values for the dimensionless L^2 -error of the velocity components and the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

L^2 error	P_{xx}	P_{yy}	P_{zz}
1/6	1.777×10^{-2}	1.777×10^{-2}	1.776×10^{-2}
1/12	4.574×10^{-3}	4.574×10^{-3}	4.574×10^{-3}
1/24	1.119×10^{-3}	1.119×10^{-3}	1.119×10^{-3}
1/48	2.536×10^{-4}	2.536×10^{-4}	2.536×10^{-4}
conv. rate	2.141	2.141	2.141

TABLE 9.7: Swinging cube with heat: numerical values for the dimensionless L^2 -error of the diagonal components of \mathbf{P} , based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

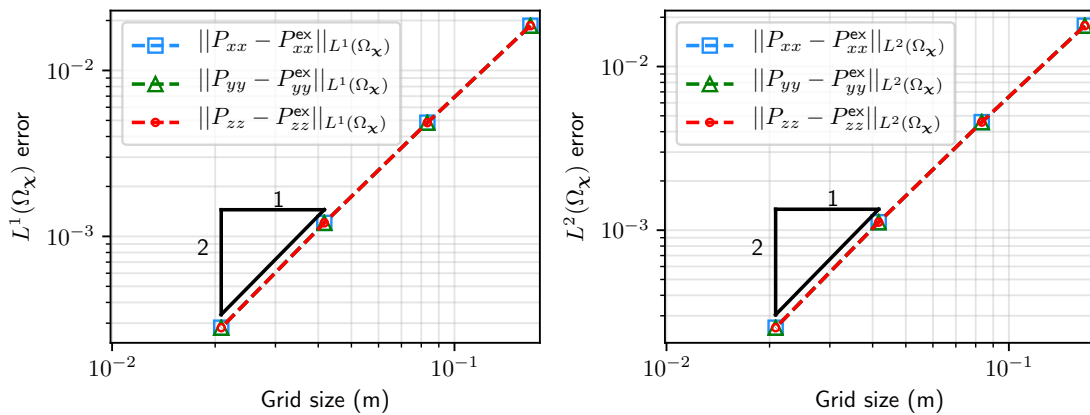


FIGURE 9.8: Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1$ ms for the components of the first Piola Kirchhoff stress tensor, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

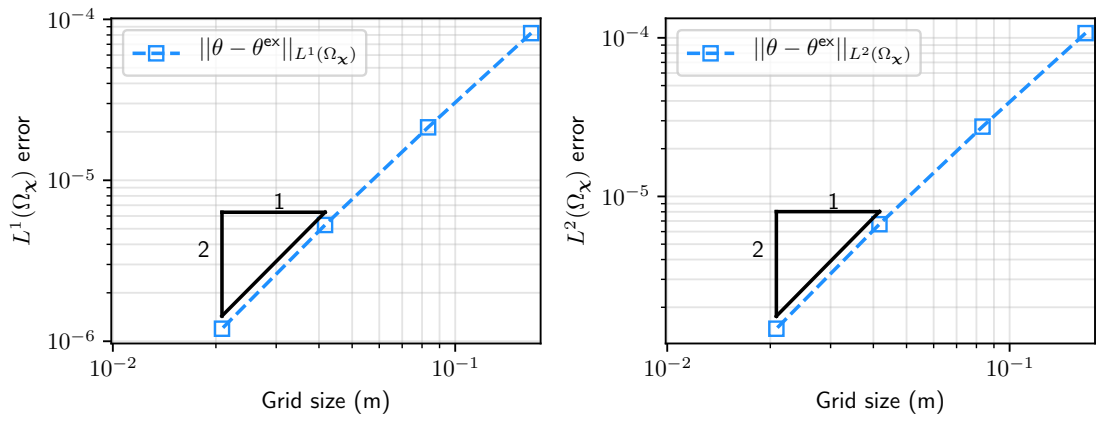


FIGURE 9.9: Swinging cube: L^1 and L^2 global convergence analysis at time $t = 1\text{ms}$ for the temperature, based on the closed-form (8.2) and the generic ALE mixed formulation (2.76). Convergence rates are calculated using the results of the two finest meshes. A neo-Hookean model with a Mie-Grüneisen equation of state is used with parameters listed in Table 9.2. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.3.

9.4 L-Shaped block (heat)

This case was previously studied in an isothermal context in [58, 75, 108] and in the present work in Case 8.4, and is now studied in the context of a thermo-mechanical deformation [7, 22, 23, 37, 233, 305, 306]. The main objective of this benchmark example is to demonstrate the capability of the ALE method to preserve linear and angular momenta of a system, as well as total energy over a long term response and in a thermo-mechanical context. An L-shaped specimen (see Figure 8.11) is subjected to an external torque consisting of time-varying tractions applied on two boundary faces, that are recalled for convenience as

$$\mathbf{F}_1(t) = -\mathbf{F}_2(t) = \begin{bmatrix} 150 \\ 300 \\ 450 \end{bmatrix} f(t), \quad f(t) = \begin{cases} t & \text{if } 0 \leq t < 2.5 \\ t - 5 & \text{if } 2.5 \leq t < t . \\ 0 & \text{else} \end{cases} \quad (9.3a,b)$$

This problem will be studied along with two different types of temperature distribution: a discontinuous distribution and a linear distribution. As for the isothermal scenario, the block is loaded during 5 seconds, and then left tumbling in space with free boundaries. A polyconvex hyperelastic neo-Hookean model is considered with parameters summarised in Table 9.8. The thermally-coupled volumetric potential is based on the [239] model defined in Appendix B.3.2.

Young's modulus	E [kPa]	50.05
density	ρ_R [kg/m ³]	1000
Poisson's ratio	ν	0.3
thermal conductivity	h [W/m ⁻¹ K ⁻¹]	10
specific heat capacity	C_v [J kg ⁻¹ K ⁻¹]	1
thermal expansion coefficient	α [K ⁻¹]	2.223×10^{-4}
reference temperature	θ_R [K]	293.15

TABLE 9.8: L-shaped block (heat): Table of parameters.

As done in the isothermal study, the results will be compared between the Total Lagrangian formulation and the ALE formulation. The mesh motion is the one presented in Equation (8.1) and can be visualised on Figure 9.10. The parameters of the prescribed ALE motion are presented in Table 9.9. The meshes S1, S2, and S3 of the isothermal Case 8.4 presented in Figure 8.12 will be also used in this example. The simulations are conducted with a 2-stage RK integrator, and a fixed CFL number $\alpha_{CFL} = 0.3$ was used.

period	T [s]	1
magnitude	β [m/s]	0.002

TABLE 9.9: L-shaped block (heat): Table of parameters for the prescribed mesh motion in Equation (8.1).

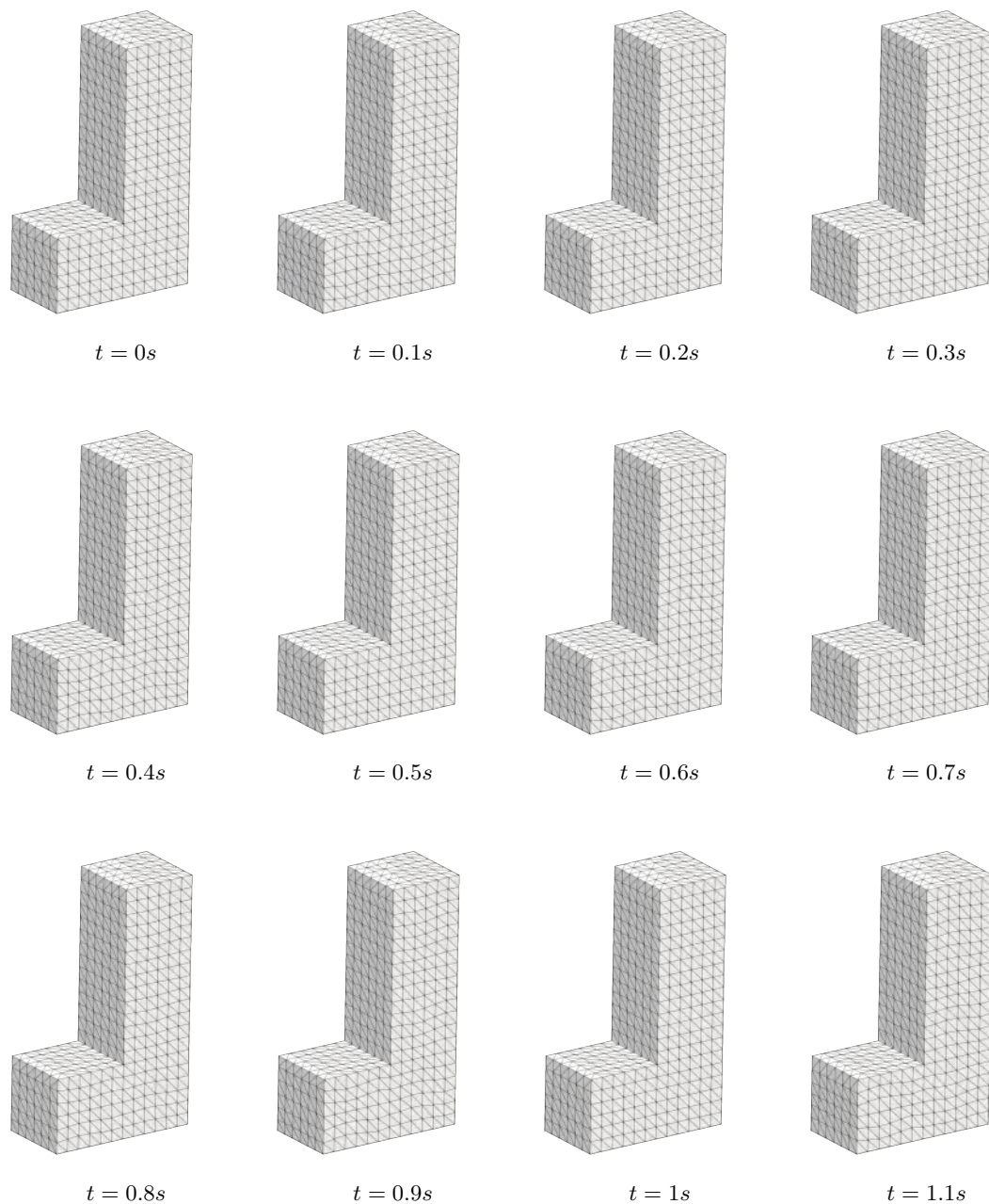


FIGURE 9.10: L-shaped block (heat): Snapshots of mesh deformation at time $t = 0, 0.1, \dots, 1, 1.1s$ (left to right, top to bottom) the ALE formulation with a prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9 and using mesh S1. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8.

9.4.1 Discontinuous temperature distribution

In this case, the initial temperature profile is initialised as

$$\theta|_{t=0} = \begin{cases} 300 \text{ K} & \text{if } Y = 10 \text{ m,} \\ 250 \text{ K} & \text{if } X = 6 \text{ m,} \\ \theta_R & \text{elsewhere.} \end{cases} \quad (9.4)$$

In the following, the ALE formulation $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, \eta, J_\Phi\}$ (with prescribed motion) is compared to the equivalent Total Lagrangian formulation $\{\mathbf{p}, \mathbf{F}, \eta, J\}$.

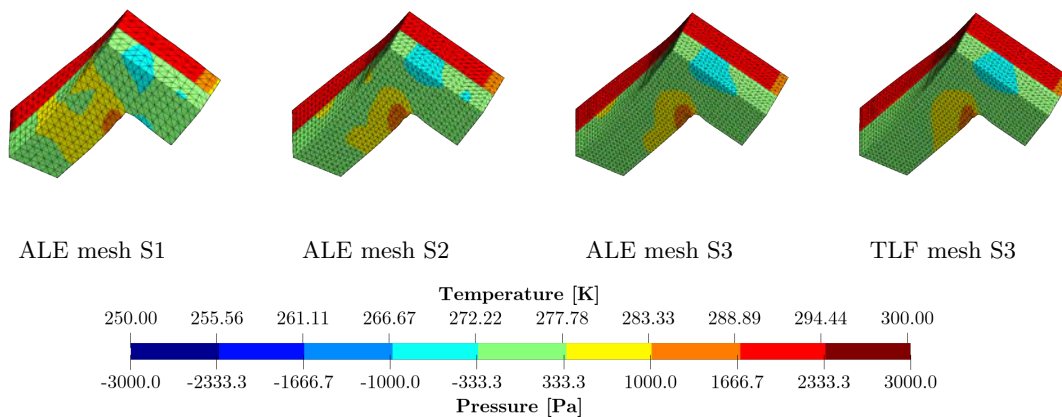


FIGURE 9.11: L-shaped block (heat): Refinement analysis at time $t = 23\text{s}$ for the ALE formulation using meshes S1, S2 and S3, and for the Total Lagrangian formulation using mesh S3 (from left to right). Representation of the temperature contours (top half) and pressure contours (bottom half). Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

A refinement study is shown in Figure 9.11, where the results using the ALE formulation are reported for the meshes S1, S2 and S3 at time $t = 23\text{s}$, and compared to the Total Lagrangian formulation for mesh S3. The results using the ALE formulation seem to converge to the results obtained with the equivalent Total Lagrangian simulation. Additionally, snapshots of the deformation at times $t = 0, 10, 20, 25, 30\text{s}$ together with pressure contours are reported in Figure 9.12 for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) for mesh S3. The evolution of the deformation, as well as the evolution of the pressure profile, are very similar for both cases and compares well with reported results in the literature. Same observation can be made in the case of the temperature contours presented in Figure 9.13. The evolution of various energy components are presented in Figure 9.14a for mesh S3. Among the reported results, the total energy E , the kinetic energy K , the mechanical component of the internal energy and the heat component of the internal energy, and the *Ballistic* energy \mathcal{B} are similar for both the Total Lagrangian and the ALE formulation. Nevertheless, a coarse mesh will result in a temperature profile that is not smooth enough. That numerical experiment is performed using the conservation of entropy. The total energy is measured as the summation of the kinetic energy and the internal energy. The latter is composed of a thermoelastic distribution (especially yielding an Equation of State) summed to a plastic (stored) contribution. The total energy undergoes an unavoidable slight decrease due to the use of upwinding-based stabilisation dissipation. The global total energy of the system is reported for the three meshes

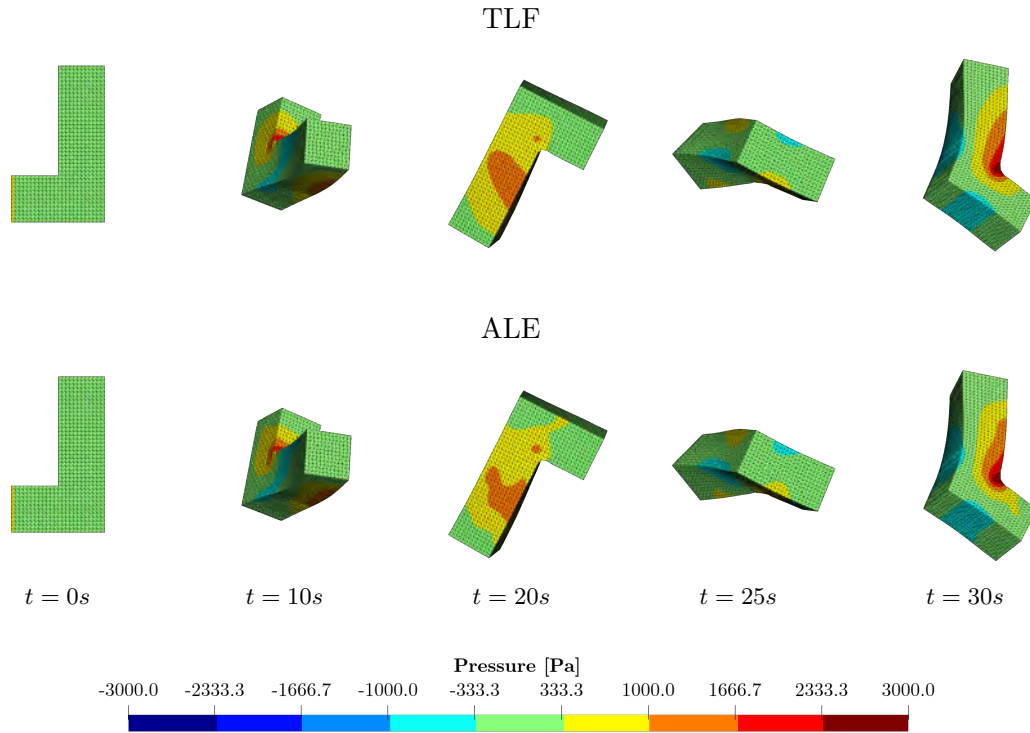


FIGURE 9.12: L-shaped block (heat): Sequence of deformation with pressure contours at times $t = 0, 20, 20, 25, 30$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

using ALE and for mesh S3 using the Total Lagrangian formulation on Figure 9.14b. In addition, The evolution of the global linear and angular momenta are respectively reported in Figure figs. 9.15 and 9.16. As for the isothermal case, $\mathcal{L}_x^{\text{global}}$, $\mathcal{L}_y^{\text{global}}$ and $\mathcal{L}_z^{\text{global}}$ refer respectively to the first, second and third component of the global linear momentum on these diagrams. Moreover, $\mathcal{A}_x^{\text{global}}$, $\mathcal{A}_y^{\text{global}}$ and $\mathcal{A}_z^{\text{global}}$ refer respectively to the first, second and third component of the global angular momentum. It is worth mentioning that like in the isothermal experiment, the global linear momentum is still equal to 0 and the global angular momentum is preserved after the forces F_1 and F_2 stopped acting (these forces are equal to 0 past 5s). This is obviously due to the fact that only torque type work is practised on the block, and the temperature has no impact on the conservation of momenta. The evolution of the velocity components at point $A = (0, 10, 0)^T$, $B = (6, 0, 0)^T$ and $C = (3, 3, 3)^T$ are displayed in Figure 9.17, and the evolution of the temperature at these points is shown in Figure 9.18.

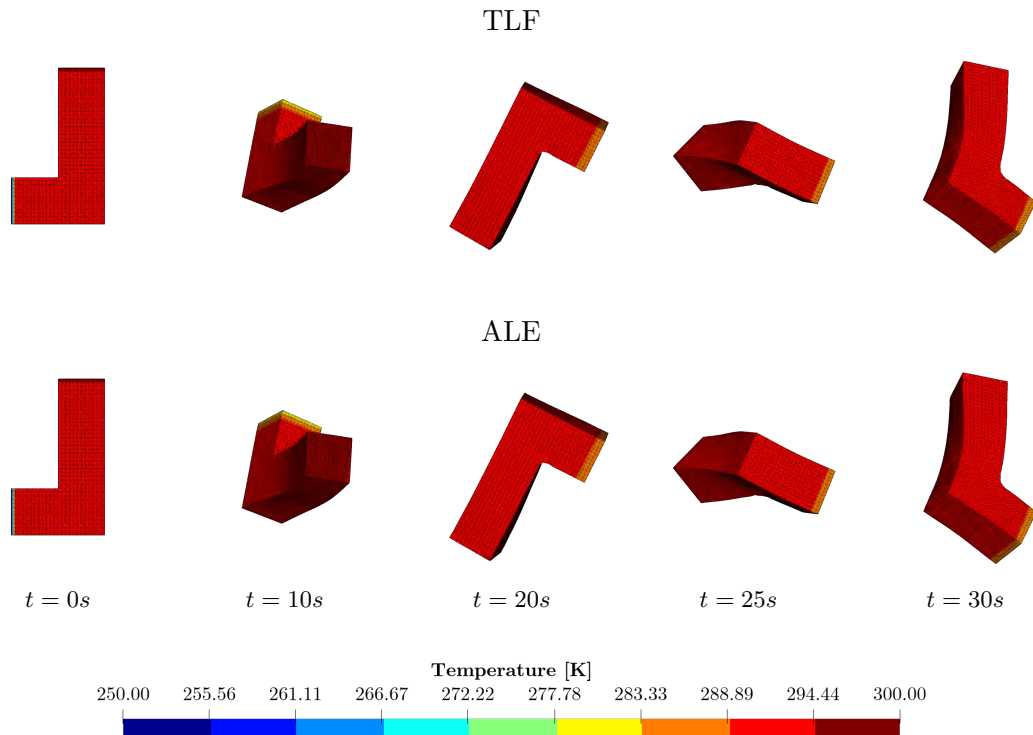


FIGURE 9.13: L-shaped block (heat): Sequence of deformation with temperature contours at times $t = 0, 20, 20, 25, 30$ s for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

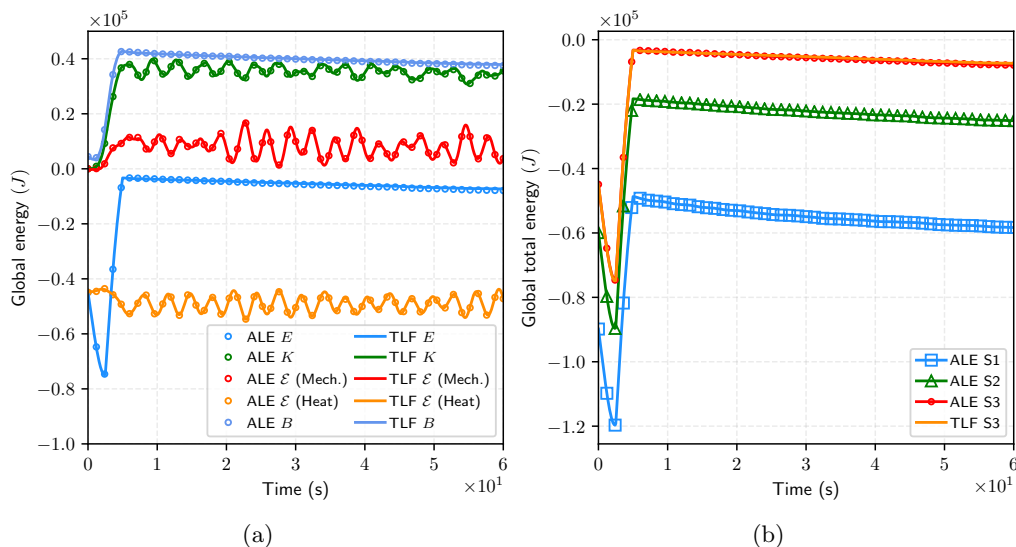


FIGURE 9.14: L-shaped block (heat): Time evolution of the (a) energy components on mesh S3 using the Total Lagrangian formulation and the ALE formulation, and (b) the total energy on meshes S1, S2 and S3 using the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

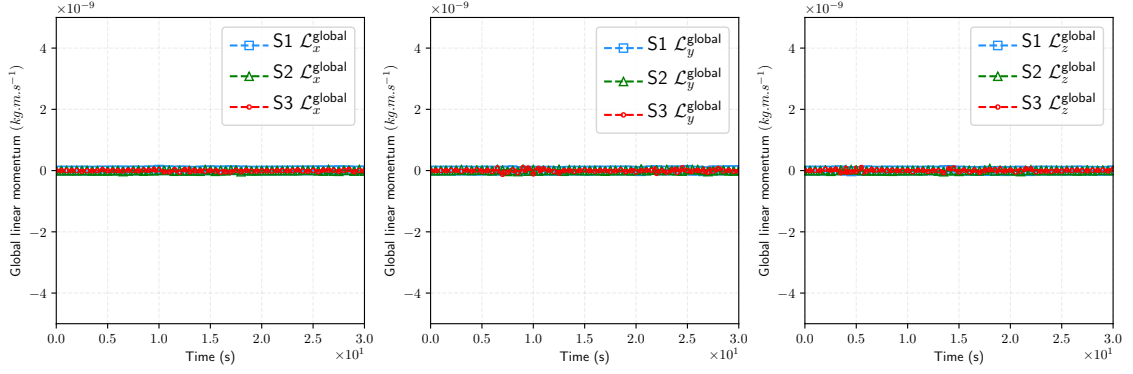


FIGURE 9.15: L-shaped block (heat): Time evolution of the components of the global linear momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

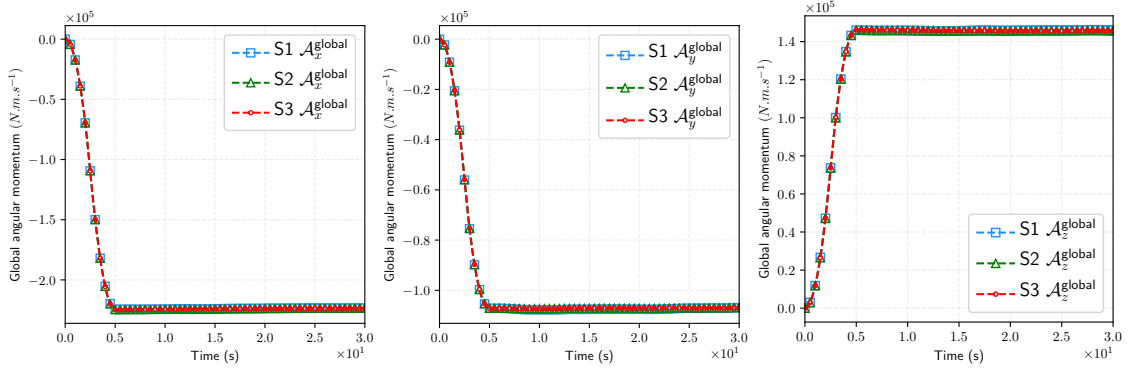


FIGURE 9.16: L-shaped block (heat): Time evolution of the components of the global angular momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), and the discontinuous temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

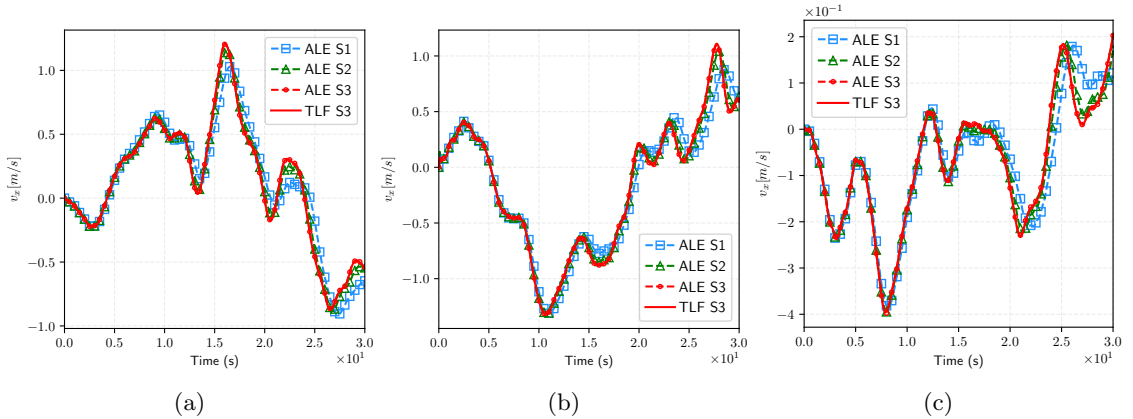


FIGURE 9.17: L-shaped block: Time evolution of the velocity components at point (a) $A = (0, 10, 0)^T$, (b) $B = (6, 0, 0)^T$, and (c) $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, and the discontinuous temperature profile (9.4), and using meshes S1, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

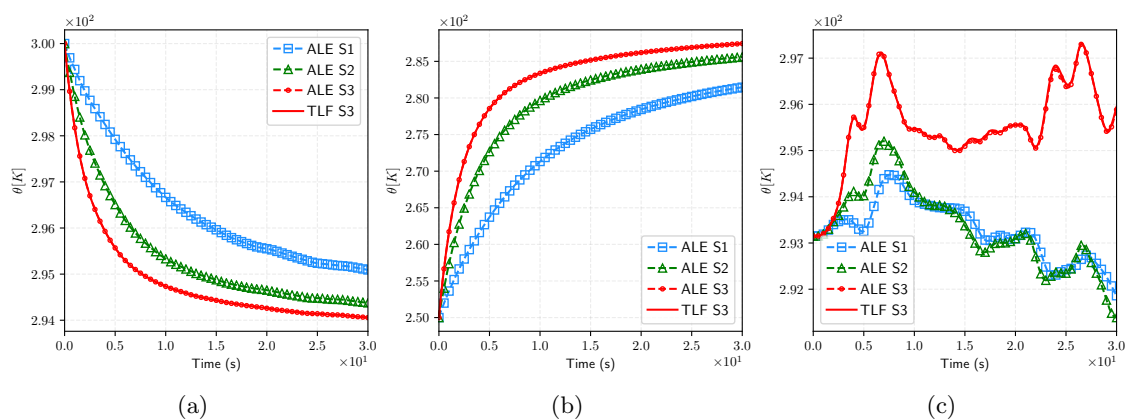


FIGURE 9.18: L-shaped block: Time evolution of the temperature at point (a) point $A = (0, 10, 0)^T$, (b) point $B = (6, 0, 0)^T$, and (c) point $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, and the discontinuous temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

9.4.2 Linear temperature distribution

The same studies are now realised for a linear temperature profile expressed as

$$\theta|_{t=0} = \begin{cases} -(43.15/3.0)X + 336.3 & \text{if } X \leq 6, X > 3 \\ (6.85/7.0)Y + 290.214286 & \text{if } Y \leq 10, Y > 3. \end{cases} \quad (9.5)$$

Hereafter, results obtained with the linear temperature profile in Equation (9.5) can be compared with the case of a discontinuous profile in Equation (9.5). The refinement study is presented in Figure 9.19 and snapshots of the evolution of the deformation together with the pressure and the temperature contours are respectively presented in Figures figs. 9.20 and 9.21.

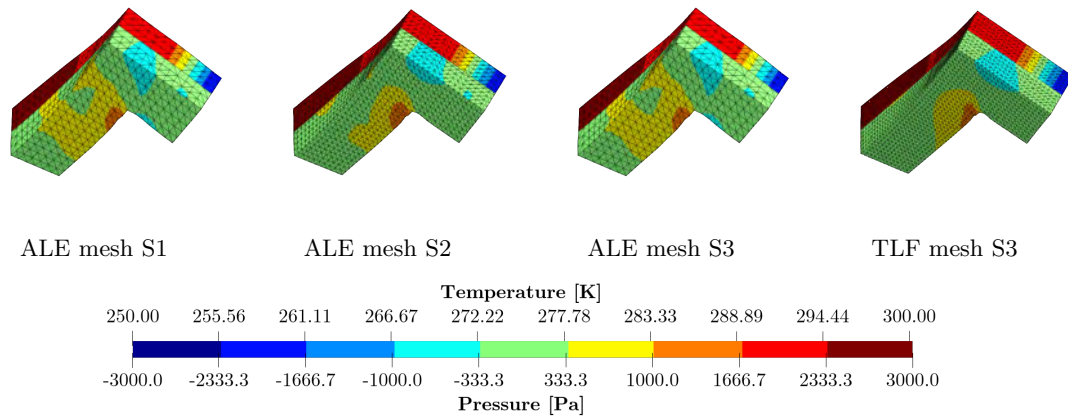


FIGURE 9.19: L-shaped block (heat): Refinement analysis at time $t = 23\text{s}$ for the ALE formulation using meshes S1, S2 and S3, and for the Total Lagrangian formulation using mesh S3 (from left to right). Representation of the temperature contours (top half) and pressure contours (bottom half). Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

In this scenario, the convergence of the total energy is affected by the initial temperature profile. On Figure 9.22a, it can be seen that the energy components for both formulation agree very well. On Figure 9.22b, it can be seen that the total energy converges in a very similar way for both formulation, and less energy is dissipation due to the absence of sharp temperature gradients. The linear and angular momenta, reported in Figure figs. 9.23 and 9.24, are unaffected by the change of temperature profile as expected. The deformation is also hardly change, as it can be seen with the evolution of the three velocity components at points A , B and C in Figure 9.25. However, since there is no more sharp gradients of temperature in the body, it can be seen in Figure 9.26 that the evolution of the temperature is significantly more linear than in the previous scenario.

The robustness of the ALE formulation was put at test through the use of a prescribed mesh motion, and with several profiles of initial temperature. No spurious mechanism was introduced. The results are as good as the ones obtained with the Total Lagrangian formulation for the variables studied. The *Ballistic* energy is smooth and decreasing with the two temperature profile. Moreover, the conservation of angular and linear momenta is ensured with both formulations: the linear momentum is close to zero machine and the angular momentum is preserved. Considering the non natural aspect of the mesh motion

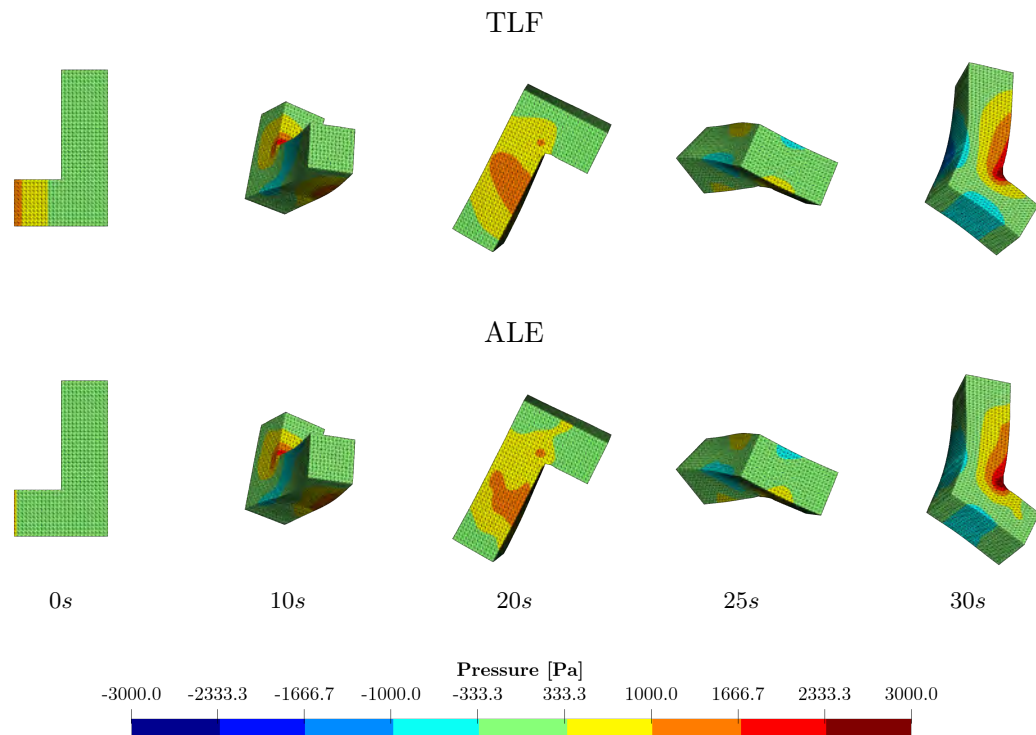


FIGURE 9.20: L-shaped block (heat): Time evolution of the deformation with pressure contours at time $t = 23\text{s}$ for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

introduced in this example, it is remarkable that the kinematics and the behaviour of the studied primary variables are well computed for the considered array of meshes.

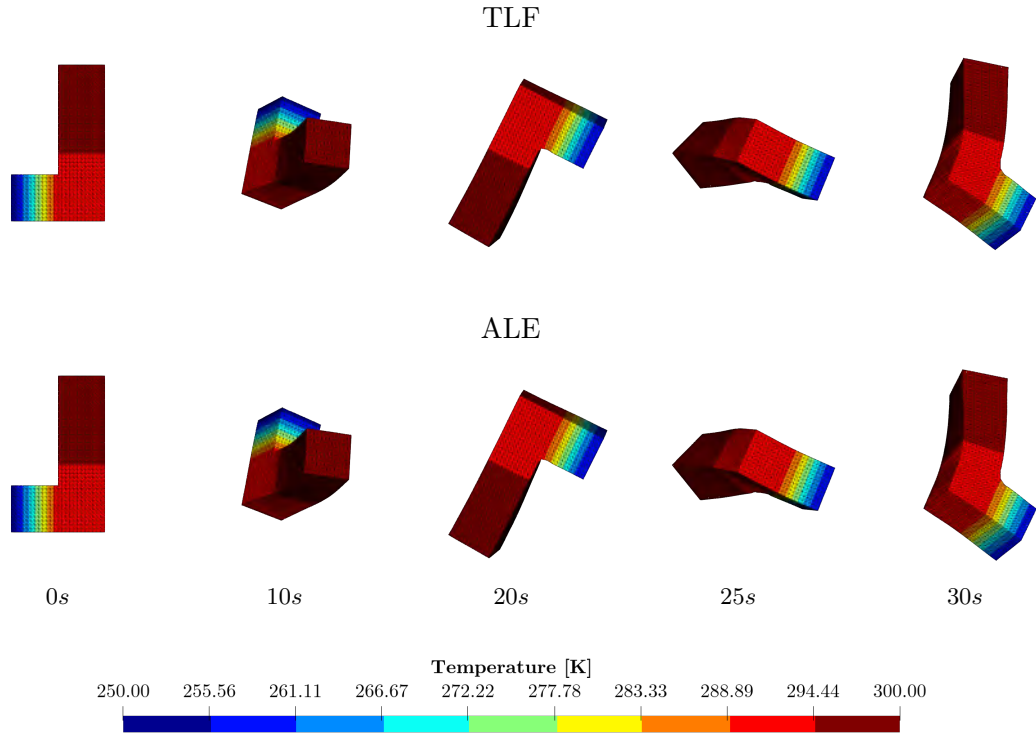


FIGURE 9.21: L-shaped block (heat): Time evolution of the deformation with temperature contours at time $t = 23s$ for the Total Lagrangian formulation (top row) and for the ALE formulation (bottom row) using mesh S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5). A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 9.9.

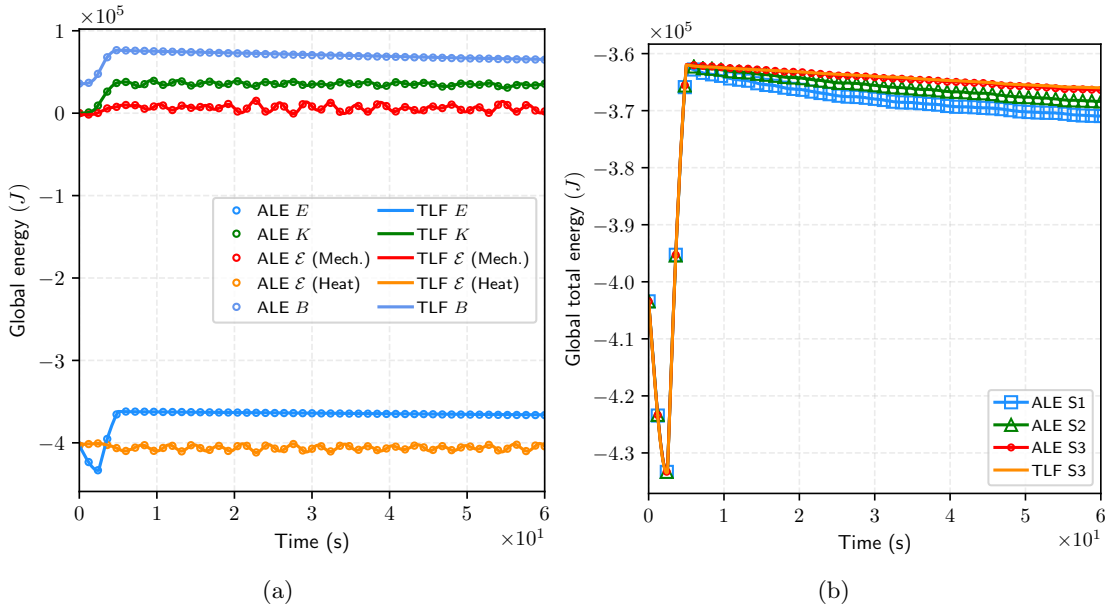


FIGURE 9.22: L-shaped block (heat): Time evolution of the (a) energy components on mesh S3 using the Total Lagrangian formulation and the ALE formulation, and (b) the total energy on meshes S1, S2 and S3 using the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.4), and using mesh S3. A neo-Hookean model with a Mie-Grüneisen equation of state and a quadratic volumetric potential is used with parameters listed in Table 9.8. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

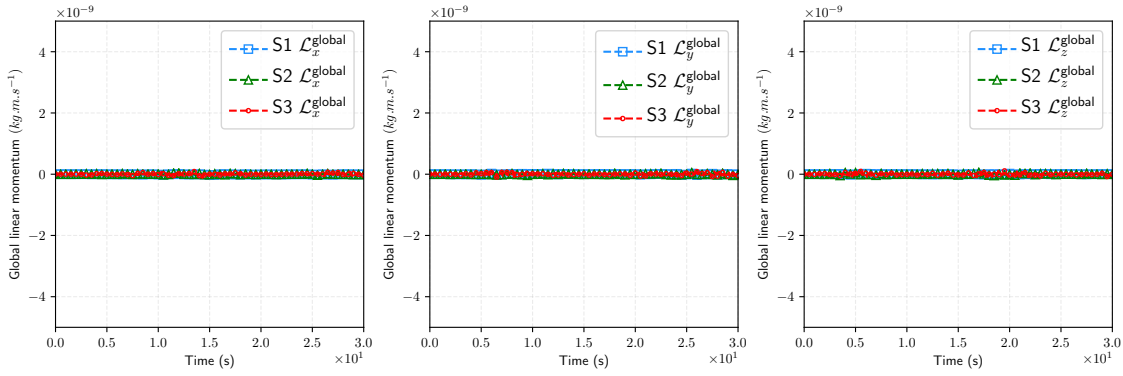


FIGURE 9.23: L-shaped block (heat): Time evolution of the components of the global linear momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.5), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

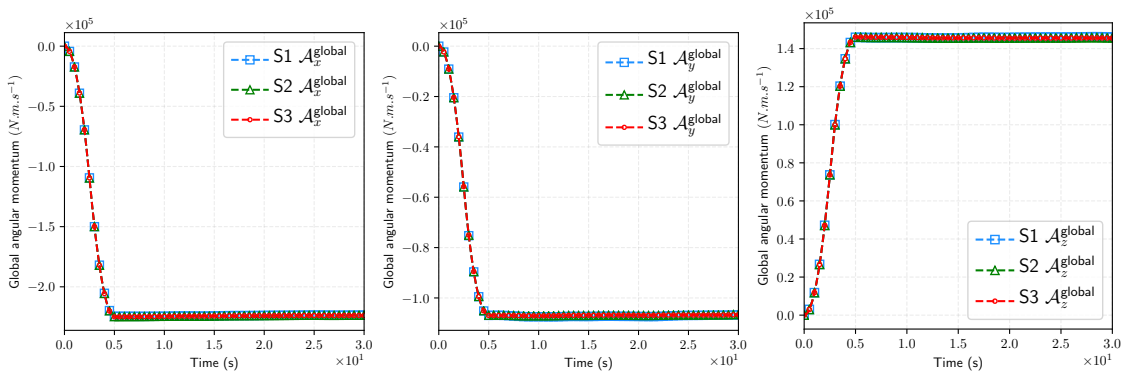


FIGURE 9.24: L-shaped block (heat): Time evolution of the components of the global angular momentum for the ALE formulation on meshes S1, S2 and S3. Results are obtained with the impulse boundary condition (9.3), the linear temperature profile (9.4), and using mesh S3. A neo-Hookean model and a quadratic volumetric potential is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

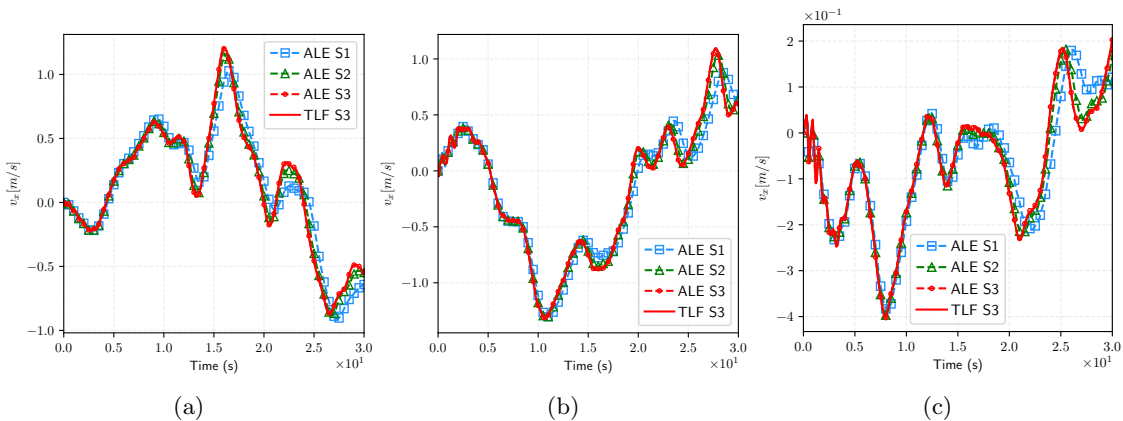


FIGURE 9.25: L-shaped block: Time evolution of the velocity components at point (a) $A = (0, 10, 0)^T$, (b) $B = (6, 0, 0)^T$, and (c) $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, the linear temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

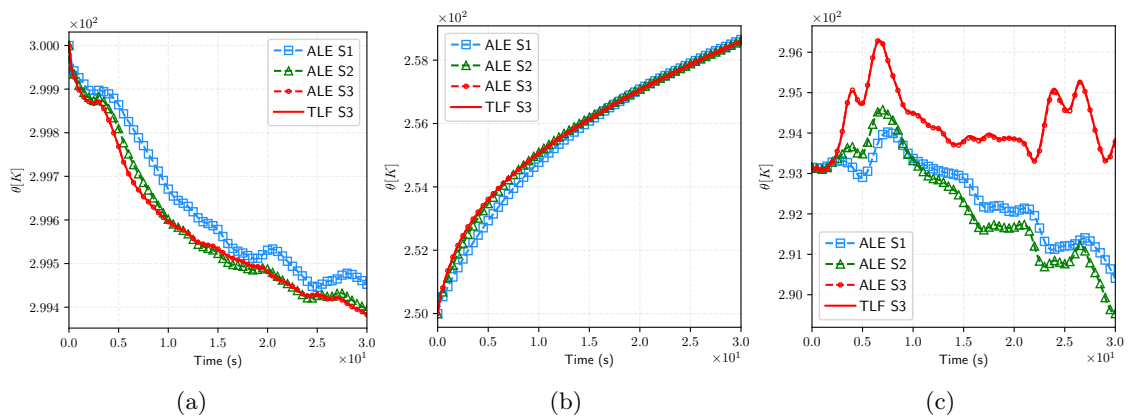


FIGURE 9.26: L-shaped block: Time evolution of the temperature at point (a) point $A = (0, 10, 0)^T$, (b) point $B = (6, 0, 0)^T$, and (c) point $C = (3, 3, 3)^T$ for the Total Lagrangian formulation and the ALE formulation. Results are obtained with the impulse boundary condition 9.3, the linear temperature profile (9.4), and using meshes S2, S2, and S3. A neo-Hookean model is used with parameters in Table 8.12. The prescribed mesh motion is presented in Equation (8.1) with parameters listed in Table 8.13.

9.5 Taylor impact with thermal coupling

Young's modulus	E [MPa]	124
Material density	ρ [kg/m ³]	8960
Poisson's ratio	ν	0.34
Yield stress	A [MPa]	90
Hardening modulus	B [MPa]	292
Hardening exponent	q	0.31
Melting temperature	θ_{melt} [K]	1356
Transition temperature	$\theta_{\text{transition}}$ [K]	298.15
Strain rate coefficient	C	0.025
Initial strain rate	$\dot{\epsilon}_{p,0}$ [s ⁻¹]	1
Temperature exponent	m	1.09
Reference temperature	θ_{ref} [K]	298.15
Thermal conductivity	h [W m ⁻¹ K ⁻¹]	286
Specific heat capacity	C_v [J kg ⁻¹ K ⁻¹]	383
Thermal expansion rate	α [K ⁻¹]	5×10^{-5}

TABLE 9.10: Taylor impact with thermal coupling: Table of parameters.

The Taylor bar problem 8.6 is now investigated in an adiabatic thermal-stress analysis; In this context, temperature changes are induced by mechanical deformations, and heat has time to diffuse through the material. The geometry considered in this model is equivalent to the isothermal scenario presented in Section 8.6, and is depicted in Figure 8.25. The Hencky-based von-Mises plasticity model is accompanied by a rate- and thermal-dependent Johnson-Cook hardening. The temperature is initialised uniformly across the specimen as $\theta_0 = 298.15\text{K}$. The material constants are given for a reference temperature $\theta_R = 298.15\text{K}$. The parameters are summarised in Table 9.10. The boundary conditions are equivalent to those presented in the Section 8.6. The mesh motion setup is exactly the same as the one used in the isothermal in Section 8.6 including the \tilde{F} function defined in Equation (8.7), and the parameters can thence be consulted in Table 8.19. The thermally-coupled volumetric potential is based on the [239] model defined in Appendix B.3.2. The simulations are conducted with a 3-stage RK integrator, and a fixed CFL number $\alpha_{CFL} = 0.4$ was used.

9.5.1 Case 1

The specimen is initialised with a uniform temperature profile of $\theta_0 = 298.15$ K, so that no important variations should be observed in the mechanical properties or in the deformation due to thermal-coupling. A refinement study is done in Figure 9.27, where a snapshot of the deformation at the final time $t = 80 \mu\text{s}$ is shown for the ALE formulation using meshes U1, U2 and U3.

Figure fig. 9.30 show the evolution of the von-Mises stress for several time snapshots, comparing the Total Lagrangian formulation (right side) with the ALE formulation (left side) for meshes U1, U2 and U3. Figure fig. 9.29 show the evolution of the equivalent

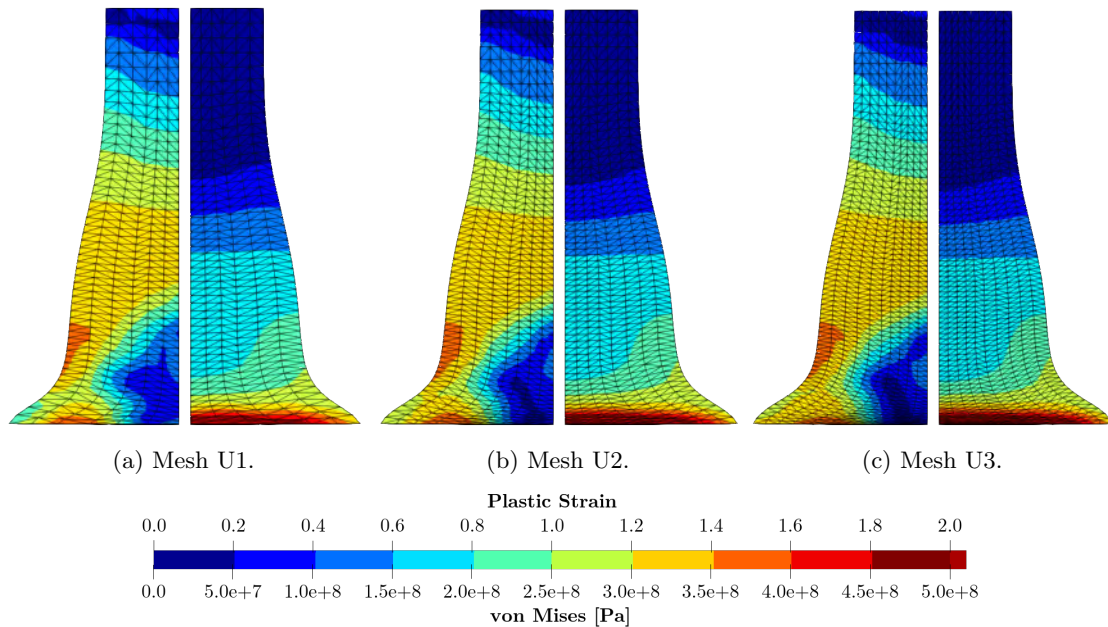


FIGURE 9.27: Taylor bar impact with thermal coupling: Refinement analysis at time $t = 80 \mu\text{s}$ with contour plots of von-Mises stress magnitude (left) and equivalent plastic strain (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

plastic strain for several time snapshots, comparing the Total Lagrangian formulation (right side) with the ALE formulation (left side) for meshes U1, U2 and U3. The evolution of the pressure field can be viewed via the snapshots reported in Figure fig. 9.31 for the three respective meshes. The results using the Total Lagrangian formulation (right side) are compared to those using the ALE formulation (left side). Similarly, the evolution of the temperature profile can be viewed in Figure fig. 9.32.

Figure 9.28 shows the evolution of the material mesh motion for different snapshots.

For the different meshes of the quarter cylinder, and using both the Total Lagrangian formulation and the ALE formulation, the total energy is featured in Figure 9.33a, and a breakdown of its components, namely the kinetic energy K , the elastic strain potential \mathcal{E} and the plastic dissipation w_p , is shown in Figure 9.33b. The kinetic energy decreases upon impact because it first transforms into elastic strain energy, and is rapidly converted into plastic dissipation when plasticity is yielded. Naturally, a minor amount of kinetic energy is also dissipated due to the proposed numerical algorithm.

On Figure 9.35, the deformation is shown for different snapshots together with the temperature contour plot.

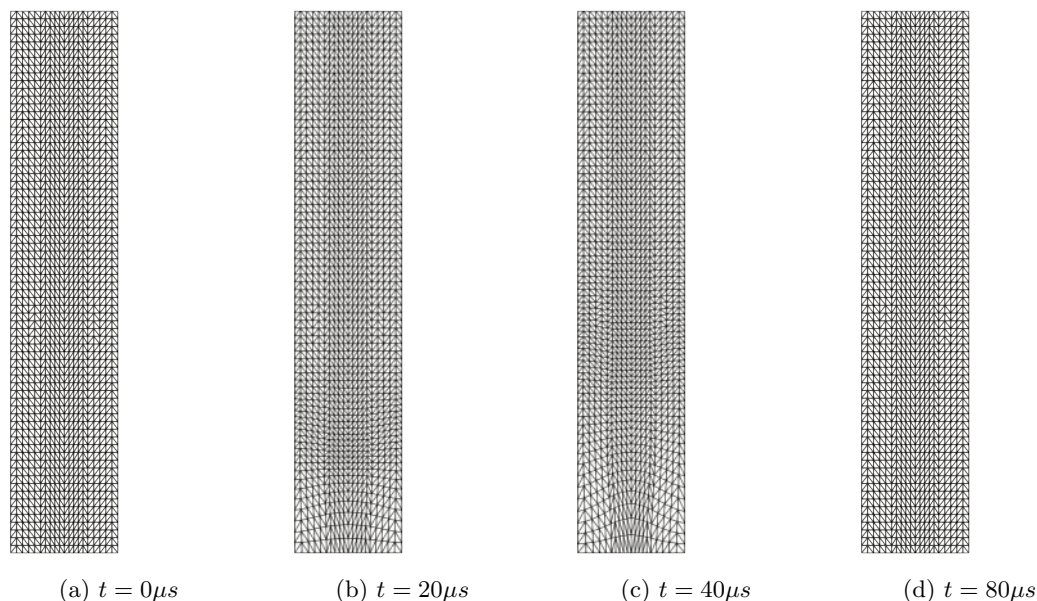


FIGURE 9.28: Taylor bar impact with thermal coupling: Comparison of material deformation together at time $t = 0, 20, 40, 80 \mu s$ using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

	final Δt	speedup ratio
TLF U1	8.665×10^{-10}	
TLF U2	5.080×10^{-10}	
TLF U3	3.328×10^{-10}	
ALE U1	3.550×10^{-9}	4.097
ALE U2	1.988×10^{-9}	3.913
ALE U3	1.208×10^{-9}	3.630

TABLE 9.11: Taylor bar impact with thermal coupling: speed-up chart comparing the values of time increments for the Total Lagrangian and the ALE formulation. Figures are rounded to the third decimal. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

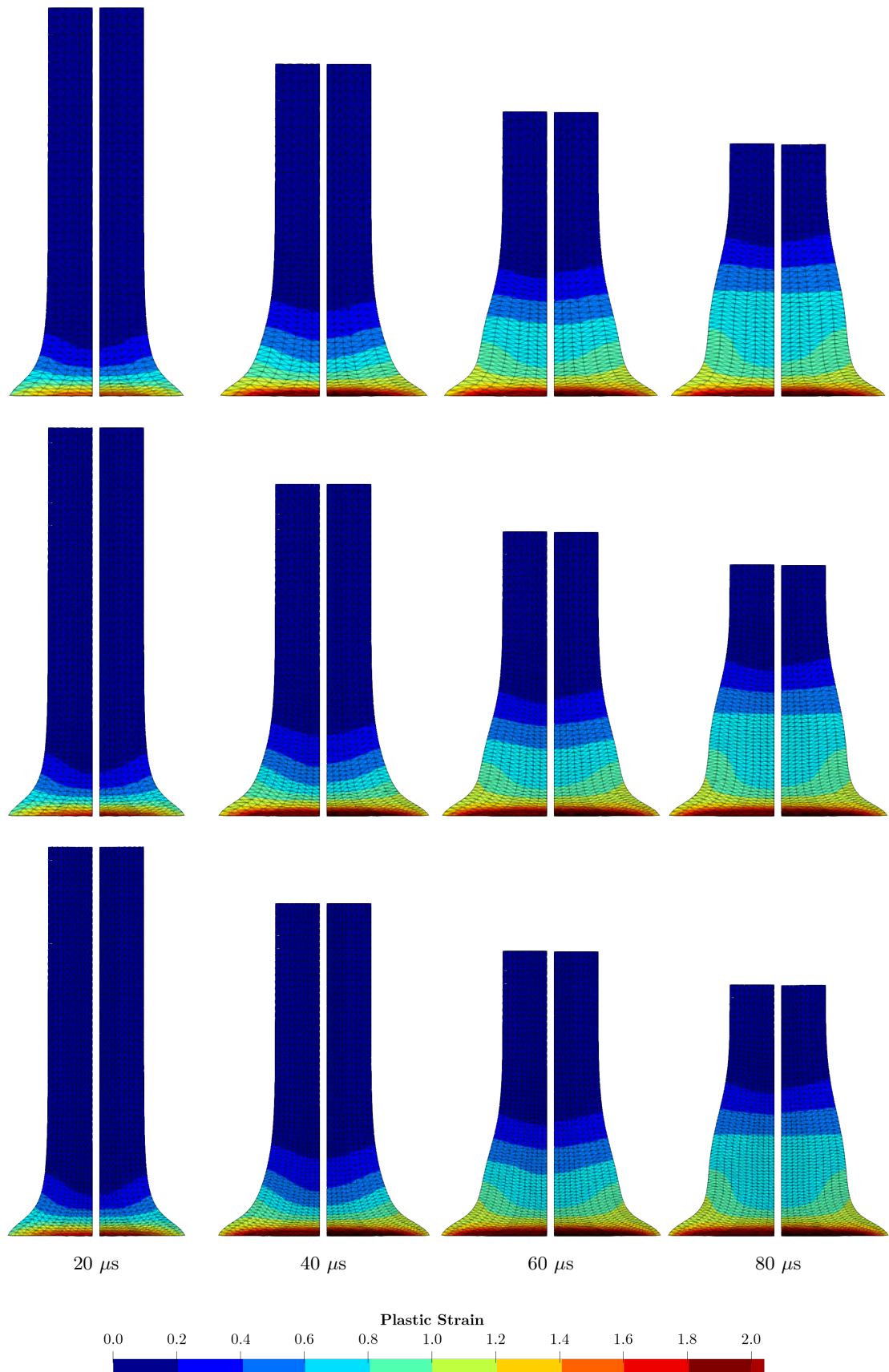


FIGURE 9.29: Taylor bar impact with thermal coupling: Snapshots of the equivalent plastic strain at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

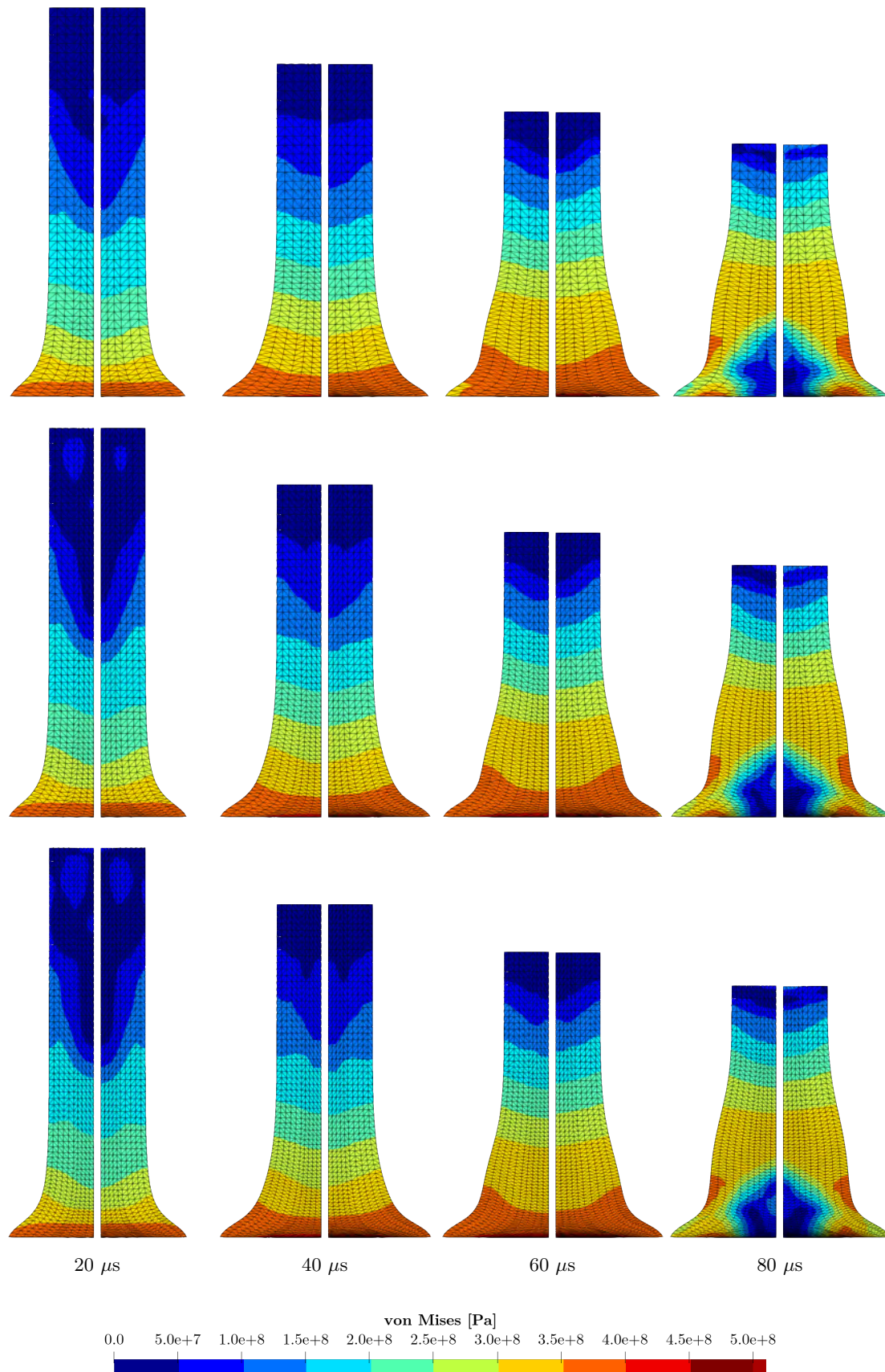


FIGURE 9.30: Taylor bar impact with thermal coupling: Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

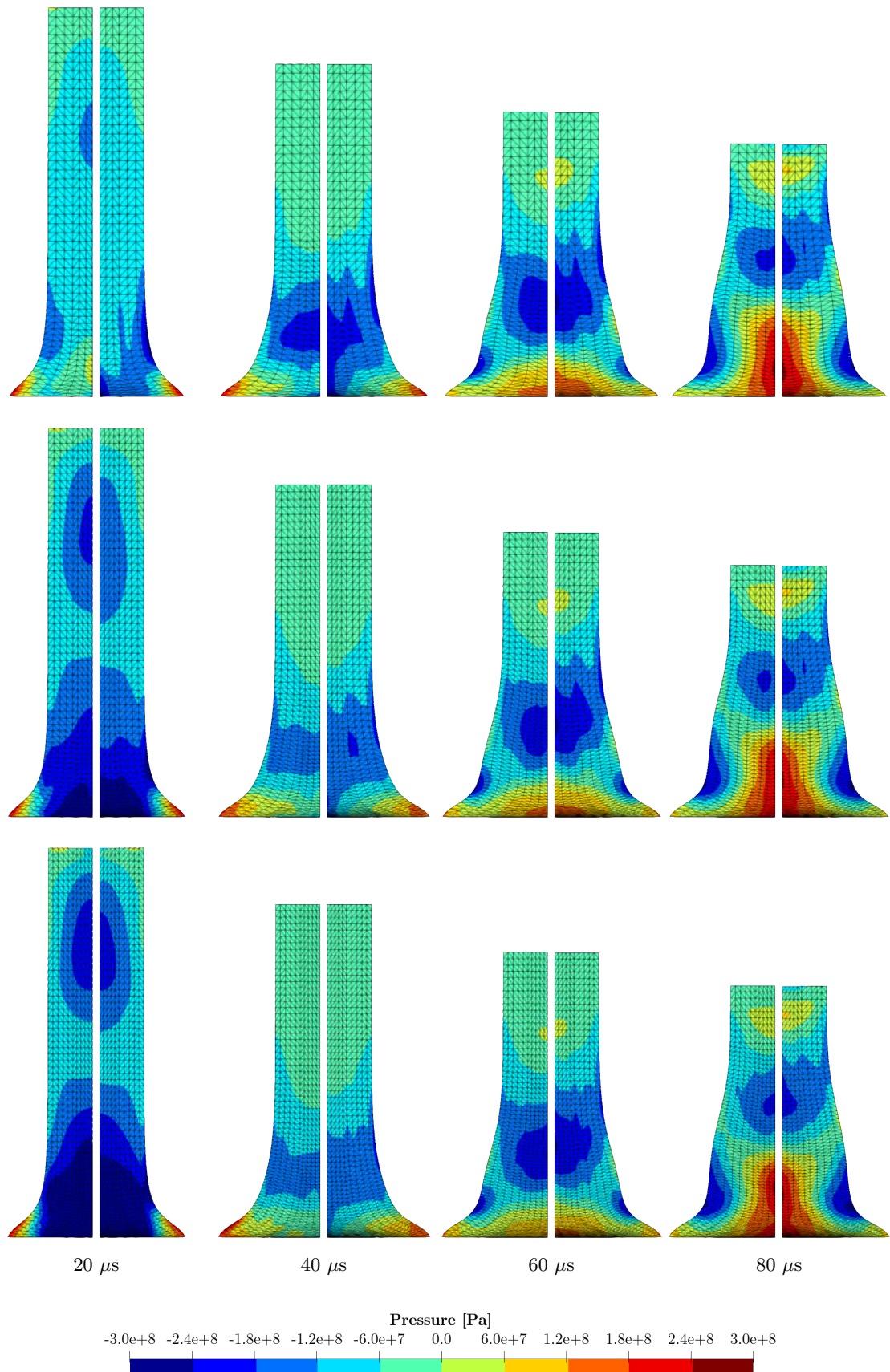


FIGURE 9.31: Taylor bar impact with thermal coupling: Snapshots of the deformation with pressure contours at times $t = 20, 40, 60, 80 \mu\text{s}$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

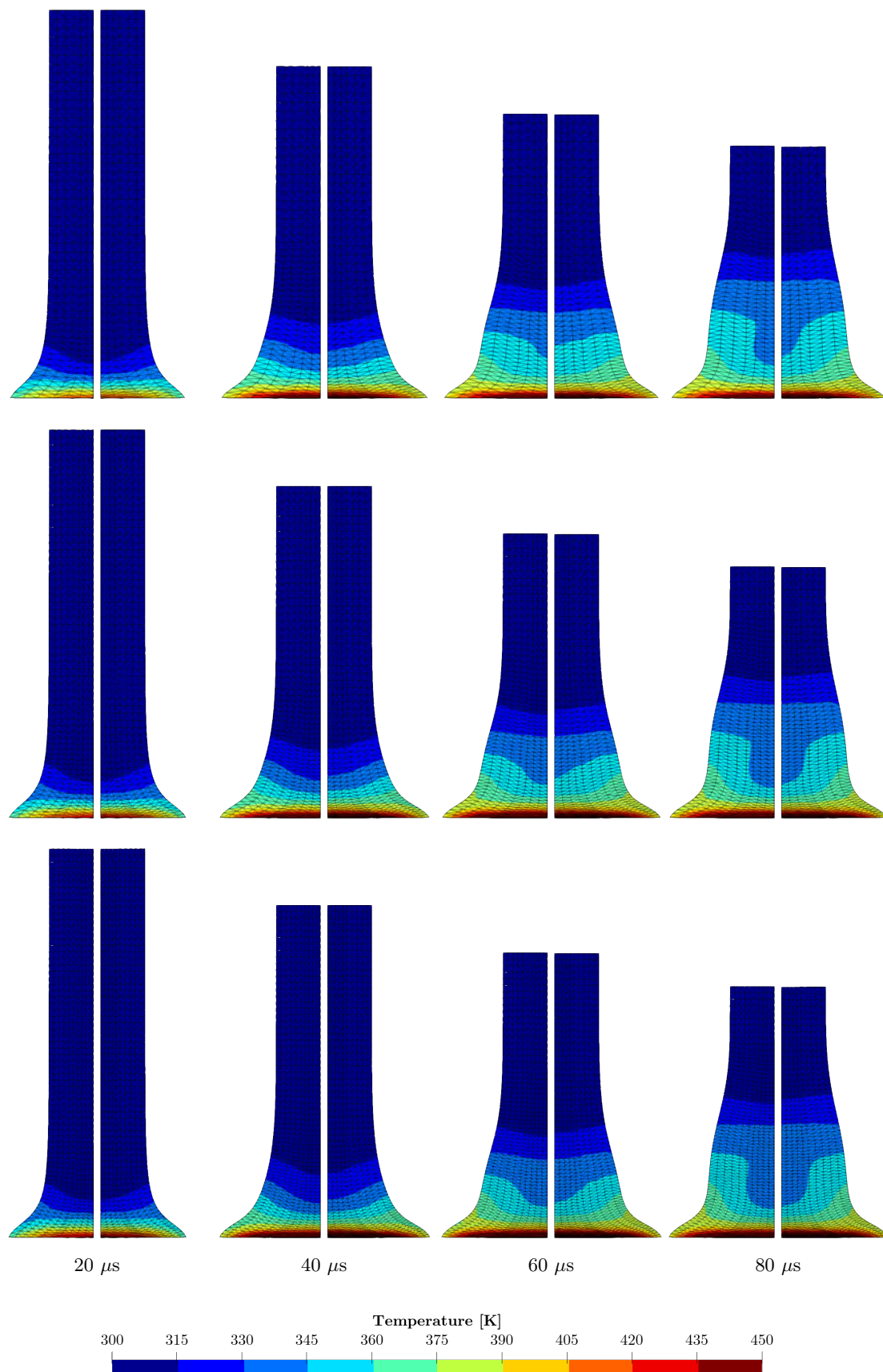


FIGURE 9.32: Taylor bar impact with thermal coupling: Snapshots of the deformation with temperature contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

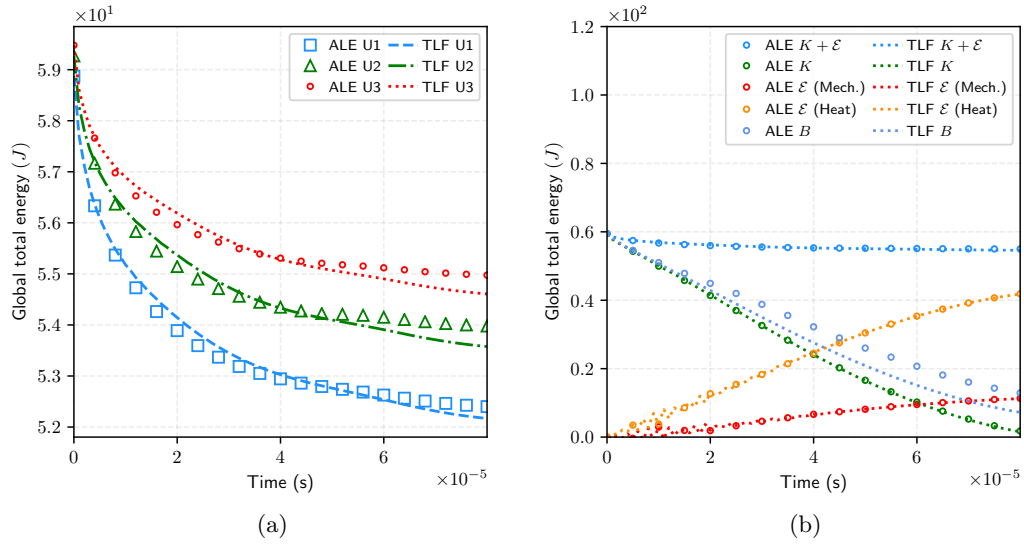


FIGURE 9.33: Taylor bar impact with thermal coupling: (a) Total energy $E = E_{\chi}/J_{\Psi}$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

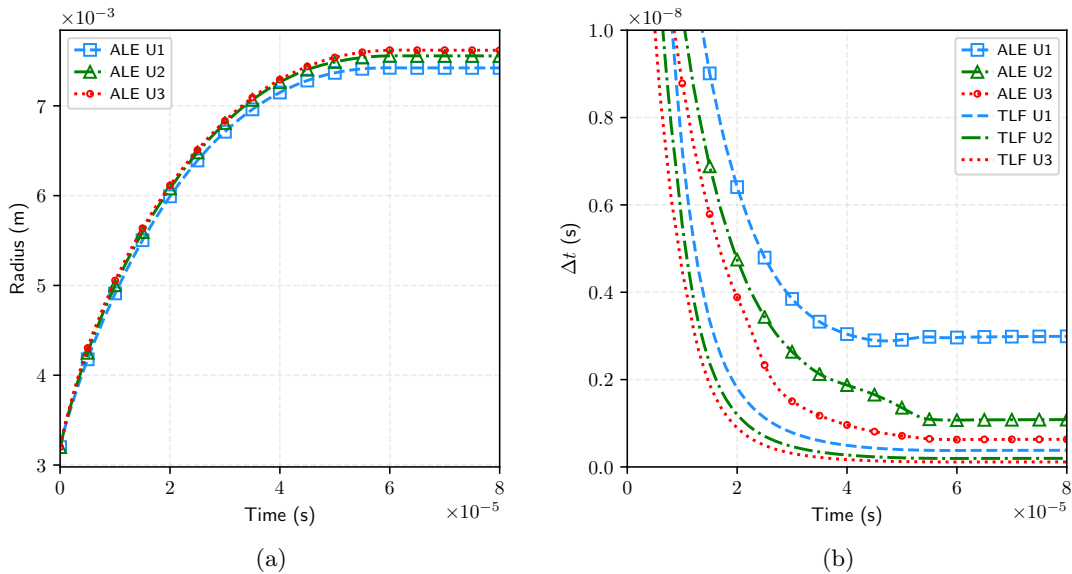


FIGURE 9.34: Taylor bar impact with thermal coupling: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

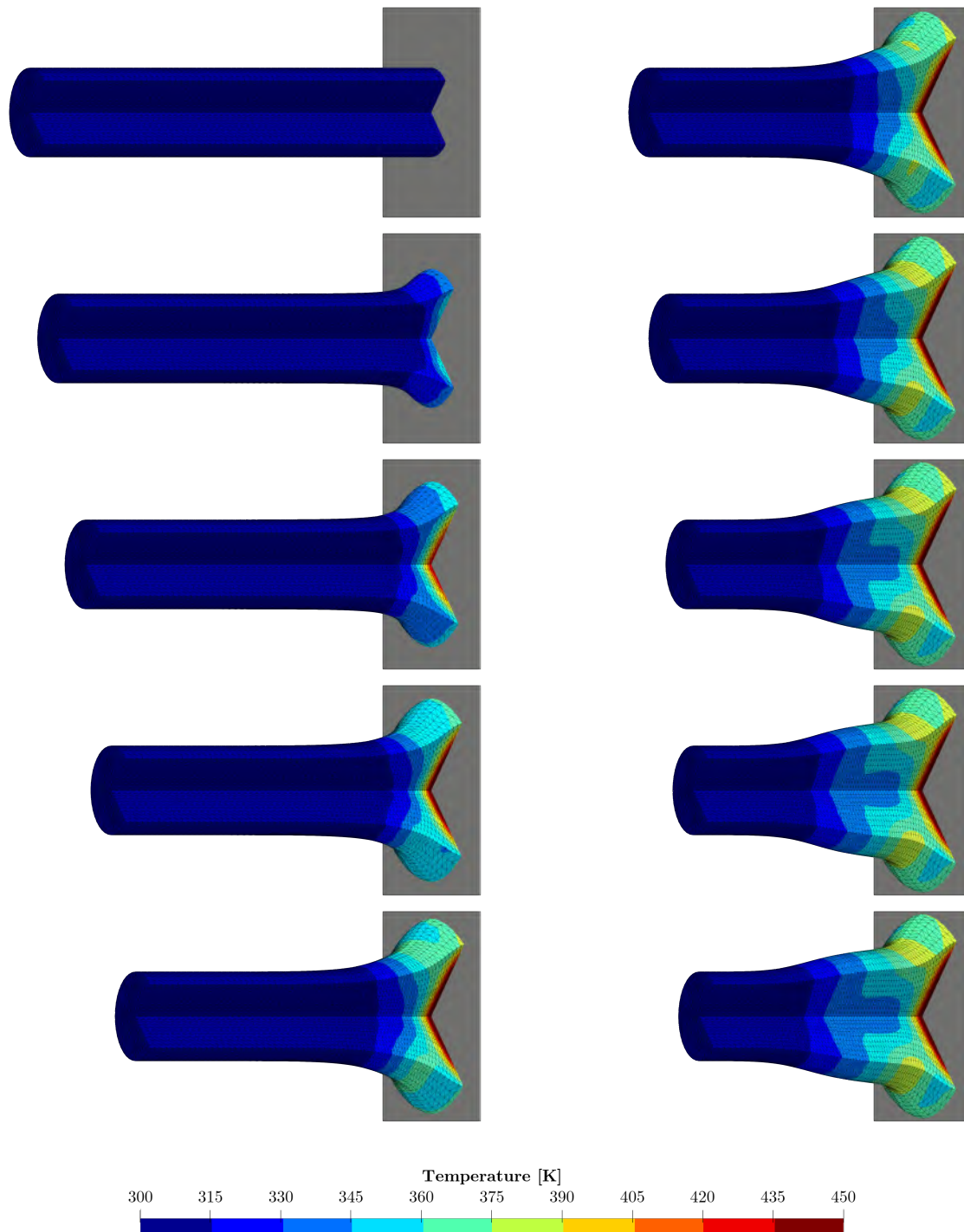


FIGURE 9.35: Taylor bar impact with thermal coupling: Comparison of spatial deformation together with temperature contour plot at time $t = 0, 10, 20, 30, 40 \mu\text{s}$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu\text{s}$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

9.5.2 Case 2

In this section, the specimen is initialised with a uniform temperature profile $\theta_0 = 573.15$ K. This will enable thermal softening, thence leading to new contours and deformation because of the high temperature. This phenomenon can be observed in metal forming and tempering processes, welding, glass manufacturing and other thermoplastic experiments. Apart from the change of initial temperature, the rest of the physical parameters remain identical to those reported in Table 9.10. The final deformation for meshes U1, U2 and U3 is shown in Figure 9.27, together with von-Mises contour plot on the left hand side and temperature contours on the right hand side.

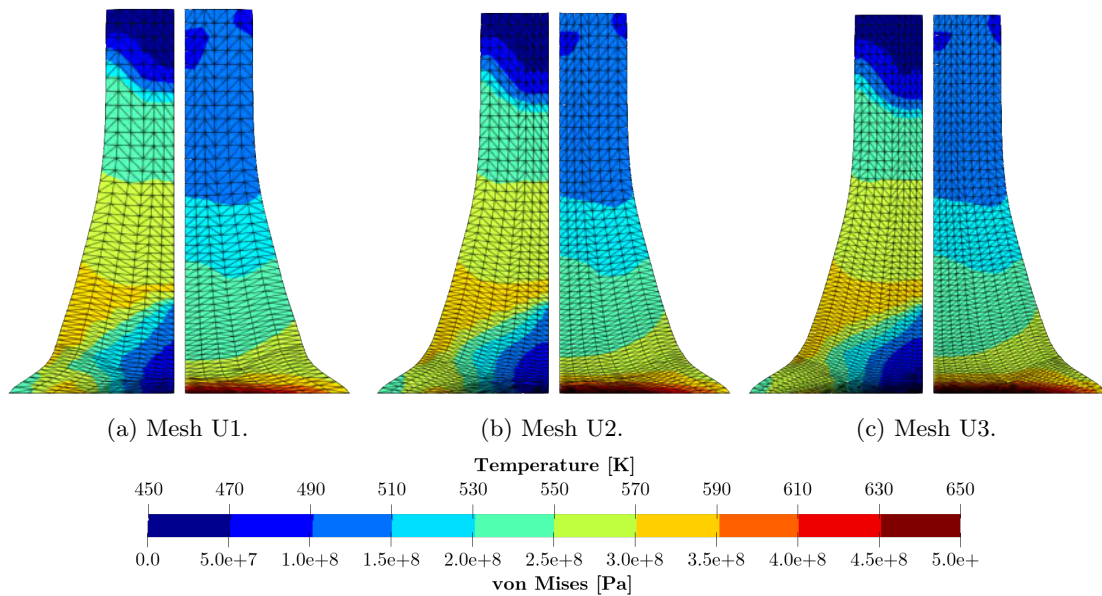


FIGURE 9.36: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Refinement analysis at time $t = 80 \mu s$ with contour plots of von-Mises stress magnitude (left) and temperature (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

The evolution of the deformation together with the von-Mises contours and with the temperature contours is respectively shown in Figures 9.37 and 9.38. On Figure 9.39a, the total energy is compared for ALE formulation and for the equivalent Total Lagrangian formulation on the three meshes. Energy components are shown on Figure 9.39b, including the kinetic energy KE , the mechanical and heat-related contributions of the internal energy, and the so-called *Ballistic* energy. Note that the energy relative to external work is equal to zero and therefore not shown. As it was done in case 1, the convergence of the radius for the ALE formulation on the different meshes is shown on Figure 9.40a, and the evolution of the time increment Δt is shown on Figure 9.40b. A sequence of snapshots of the deformation, together with temperature contours, is shown in Figure 9.41 for a

extensive selection of time slots.

To conclude, this case has shown that during impact, a large amount of the kinetic energy is converted to irrecoverable heat dissipation and plastic dissipation, and a small amount is converted to elastic strain energy. As the global energy decreases over time, the global entropy will increase, hence ensuring long time stability. The radius of the specimen on the impact plane does converge when using the ALE simulation. The results agree very well with what is reported in the literature [6, 20, 75]. This is remarkable as the mesh motion does not depend on an *ad hoc* technique. It has also been seen that the deformation pattern and the contours of temperature, von-Mises stresses and plastic strain agree extremely well as the mesh is refined. When the numerical experiment is reproduced with a higher initial temperature, thermal softening occurs as expected and identical remarks can be made: results agree extremely well as the mesh refines, and contours plots of temperature, pressure and plastic variables are smooth. The similarity between the evolution of the plastic strain and the temperature is also observed with the ALE formulation.

Moreover, as it was observed for the isothermal scenario, the effects of using the ALE formulation are especially visible in this example. First, it is noted that the mesh motion patterns of Armero [6] were replicated. Second, more elements are pulled away from the bottom of the specimen throughout the simulation. As a consequence, the contours are of better quality and converge faster than with the Total Lagrangian formulation in that area. Third, the elements around the impact plane have a better shape overall, and the time increment is directly impacted: Δt is generally $4\times$ as high as in the Total Lagrangian scenario.

A clear advantage of using the ALE formulation in this impact case is that for the cost of solving extra variables, the mesh quality is automatically improved in area with high plasticity, and time increments are increased for a faster simulation.

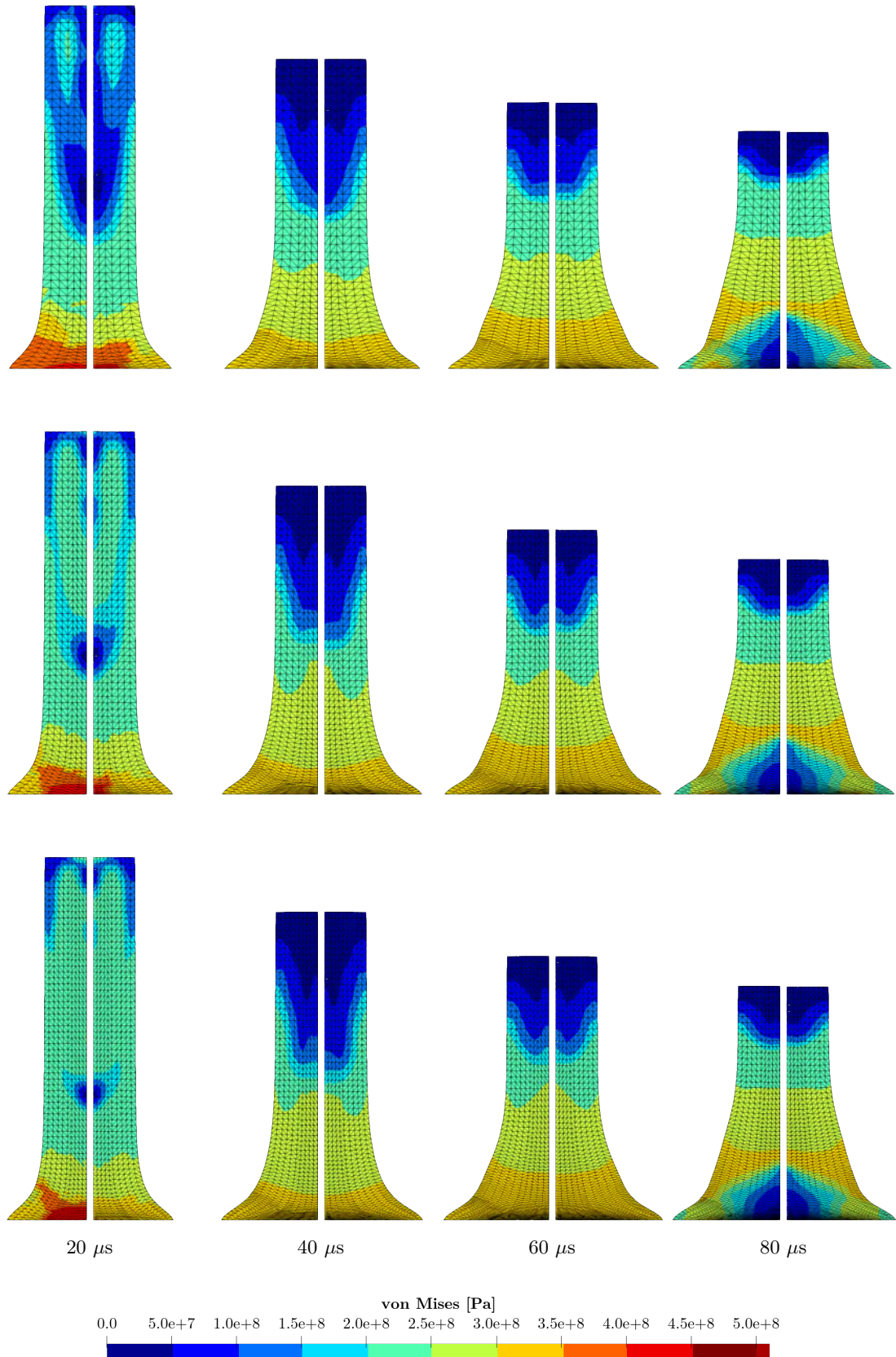


FIGURE 9.37: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Snapshots of the von-Mises stress at times $t = 20, 40, 60, 80 \mu s$, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

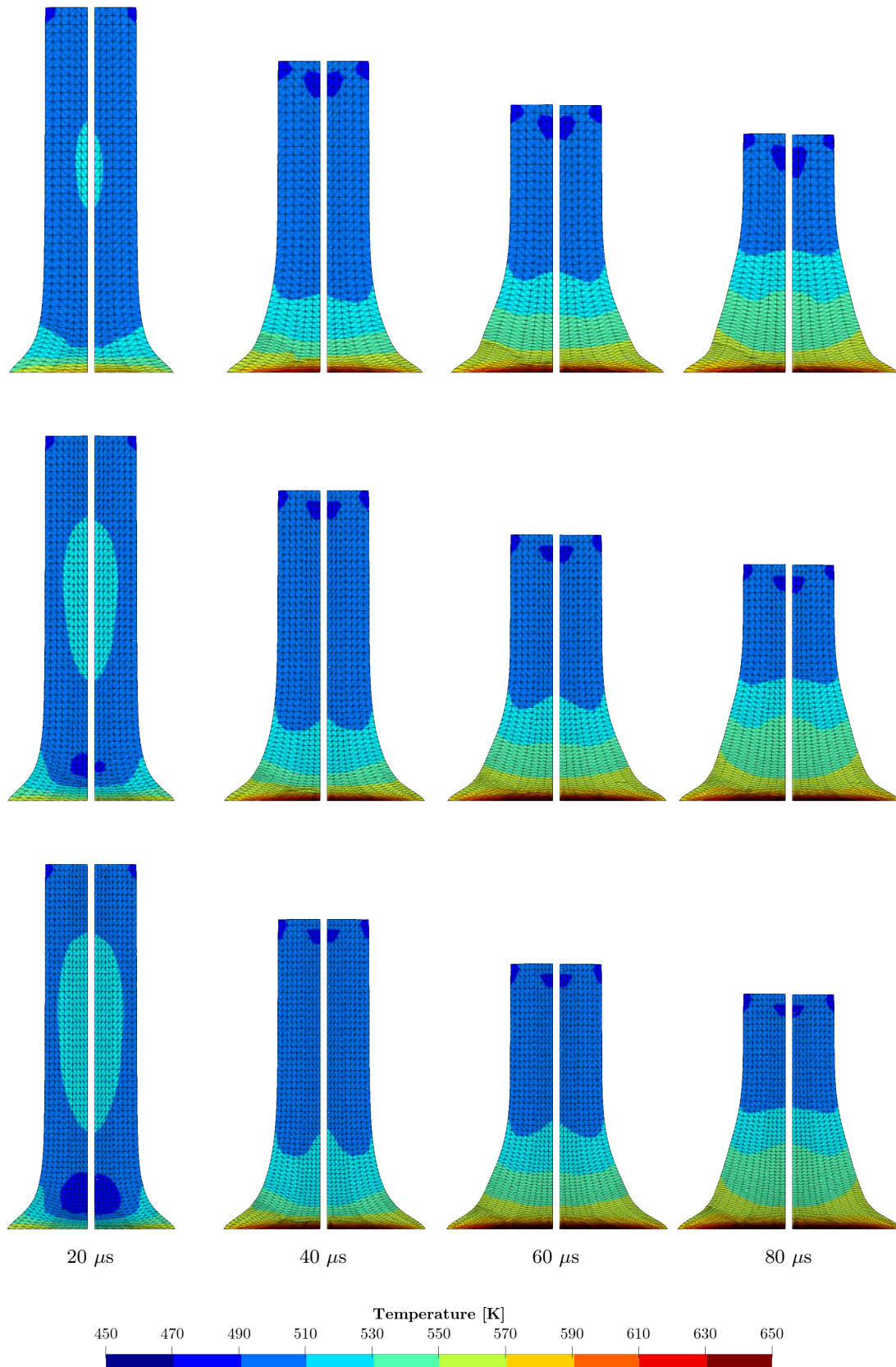


FIGURE 9.38: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Snapshots of the deformation with temperature contours at times $t = 20, 40, 60, 80 \mu s$, using the Total Lagrangian formulation (right side) and the ALE formulation (left side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

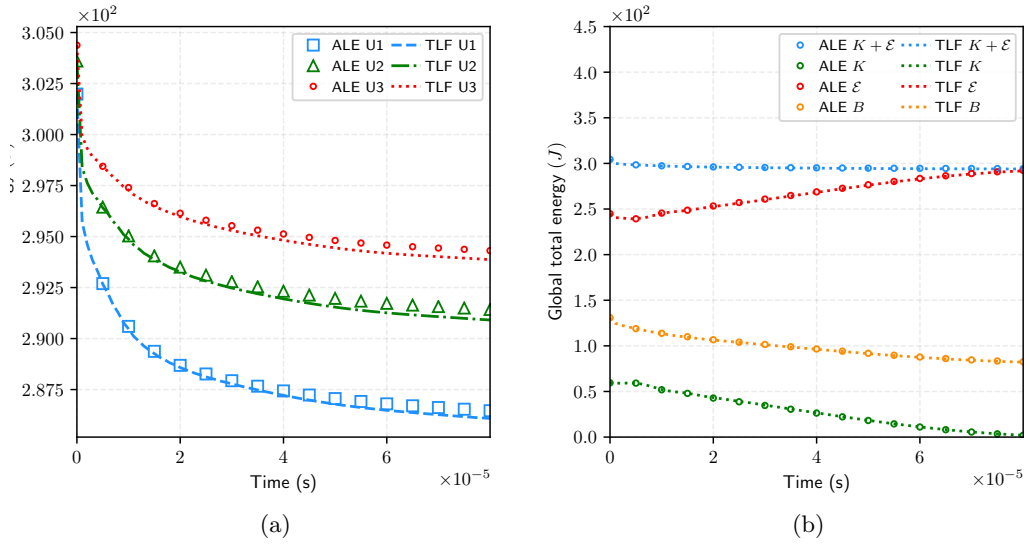


FIGURE 9.39: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): (a) Total energy $E = E_\chi / J_\Psi$, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

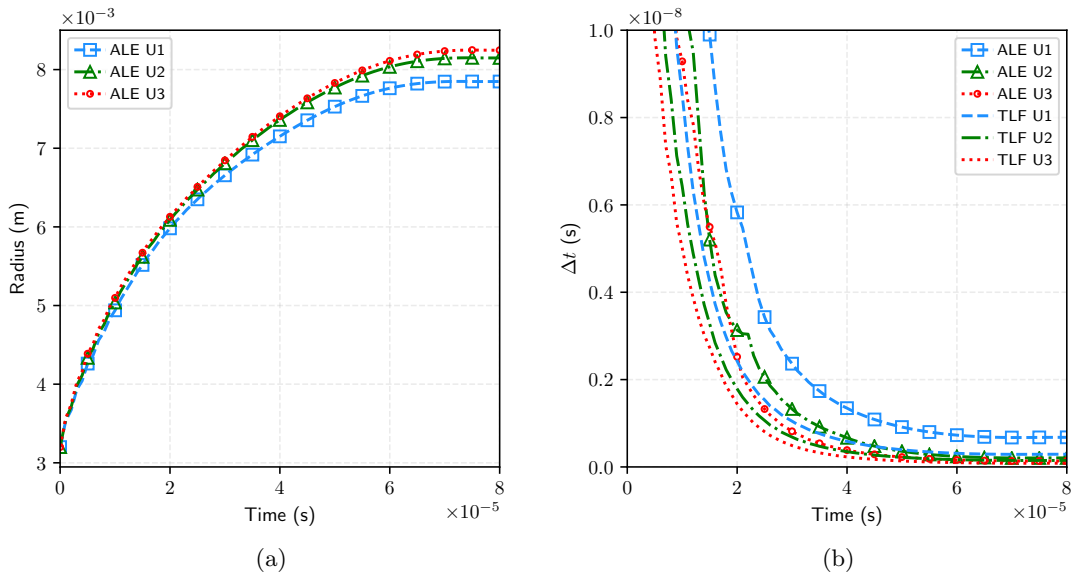


FIGURE 9.40: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

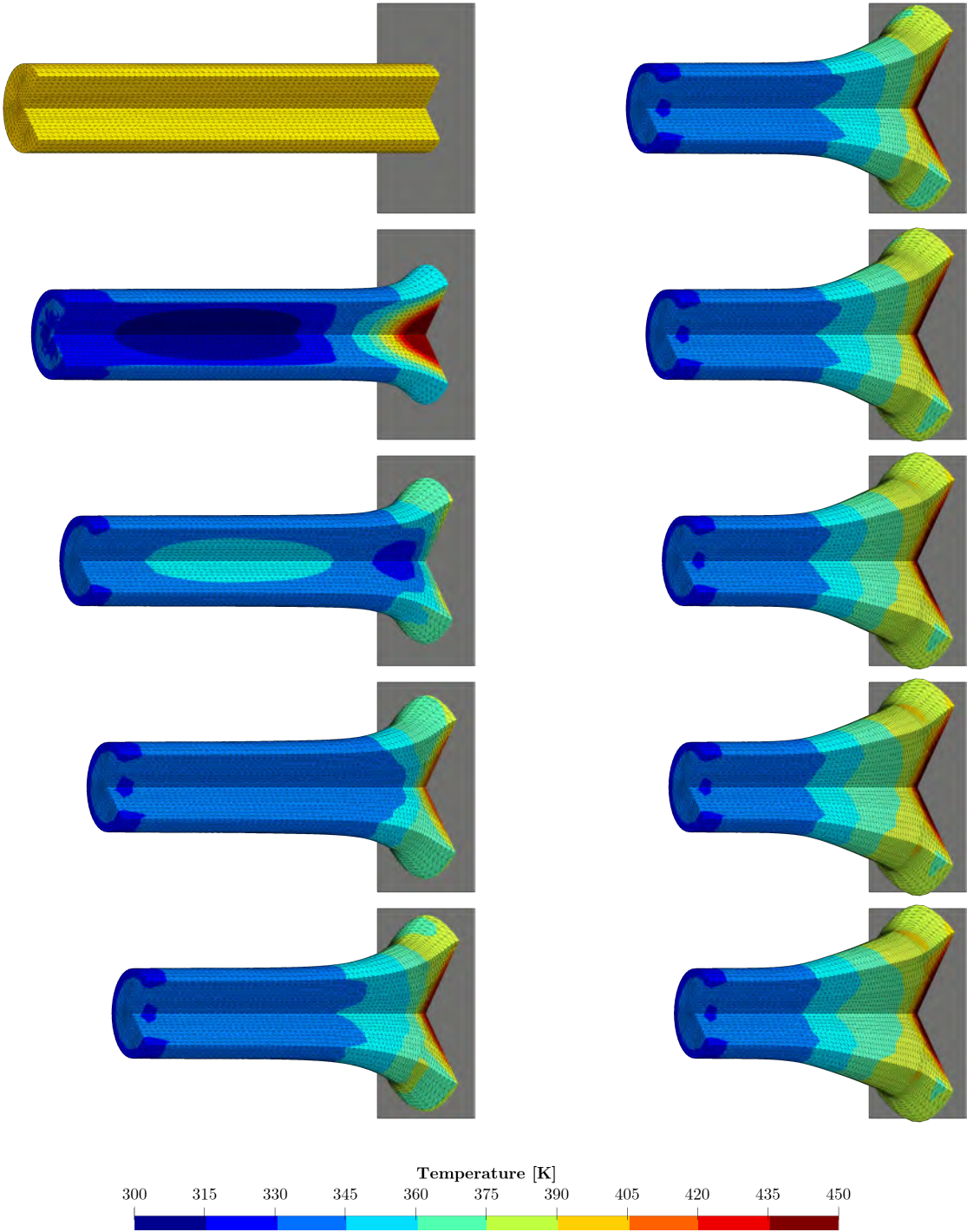


FIGURE 9.41: Taylor bar impact with thermal coupling ($\theta_0 = 573.15K$): Comparison of spatial deformation together with temperature contour plot at time $t = 0, 10, 20, 30, 40 \mu s$ (first column, top to bottom) and $t = 50, 60, 70, 75, 80 \mu s$ (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

9.6 Necking with thermal effects

The setup of the Taylor impact with thermal coupling 9.5 is reproduced identically with the exception of a reversed initial velocity, as in [21]. The objective of this numerical example is to demonstrate the capacity of the proposed ALE framework to alleviate spurious mode in the presence of massive stretching and thermal softening. As seen in the isothermal case 8.7, the use of the ALE method is expected to improve the shape of the neck region, and help recover stronger plasticity values. By the several physical phenomena at stake and by the characteristic mesh distortion that happens in this scenario, the ALE framework is an interesting candidate for the simulation. A Hencky-based von-Mises plasticity model is used in conjunction with a rate- and thermal-dependent Johnson-Cook hardening law. The material properties are those of the impact bar case with heat, and are presented in Table 9.10. To account for thermal effects, the temperature will be initialised as $\theta_0 = 593.15K$. The thermally-coupled volumetric potential is based on the [239] model defined in Appendix B.3.2. The mesh motion is taken from Cases sections 8.6 and 9.5, including the $\tilde{\mathbf{F}}$ function defined in Equation (8.7) and $\alpha_{ALE} = 1$. The simulations are conducted with a 3-stage RK integrator, and a fixed CFL number $\alpha_{CFL} = 0.4$ was used. As for the Taylor impact, the three meshes U1, U2 and U3 represents one eighth of the physical bar structure.

A refinement analysis of the necking using the ALE formulation is shown in Figure 9.42. It is clear that the stress and temperature contours are converging. note that the deformation and the number of elements in the neck region are comparable with the isothermal case. In Figure 9.43, the deformation can be observed for snapshots at time $t = 15, 20, 25, 30ms$ for meshes U1 (first row), U2 (second row) and U3 (third row). The contours von-Mises stress magnitude are compared between the ALE formulation (left side) and the equivalent Total Lagrangian formulation (right side). It can be seen despite the different nature of the two methods and of the deformation at the neck region, the contours are remarkably comparable. Moreover, the mesh motion seems to alleviate spurious modes in the coarse mesh. Exact same observations can be made about the pressure contours in Figure 9.44, the contour of equivalent plastic strain in Figure 9.45, and the temperature contours in Figure 9.46. As it is the case in the isothermal scenario, the ALE formulation recovers more plasticity in the necking area. Consequently, more temperature is also observed in that region. The evolution of the total energy is shown in Figure 9.47a for the two formulations using the three meshes, and the energy components are plotted for the finest meshes in Figure 9.47b. The evolution of components, including the *Ballistic* energy \mathcal{B} , for both formulations agree very well. For comparison purposes, the evolution of the radius of the neck is reported in Figure 9.48a. The evolution of the time increment in Figure 9.48b shows that, like it is the case in the equivalent isothermal process, the presence of more elements in the neck leads to smaller time increments. For comparison purposes, a series of snapshots are presented in Figure 9.49, where the plastic strain contour is shown together with the temperature contour.

To conclude, this simulation of a bar necking with thermal softening is the continuity of the isothermal necking presented in Case 8.7. The ALE formulation shows the same advantage in pulling the elements in the neck region. This results in a automatic and dynamic mesh motion that mimics a local refinement. Additionally, it is shown that elastic, plastic and thermal variables are exempt of spurious instabilities.

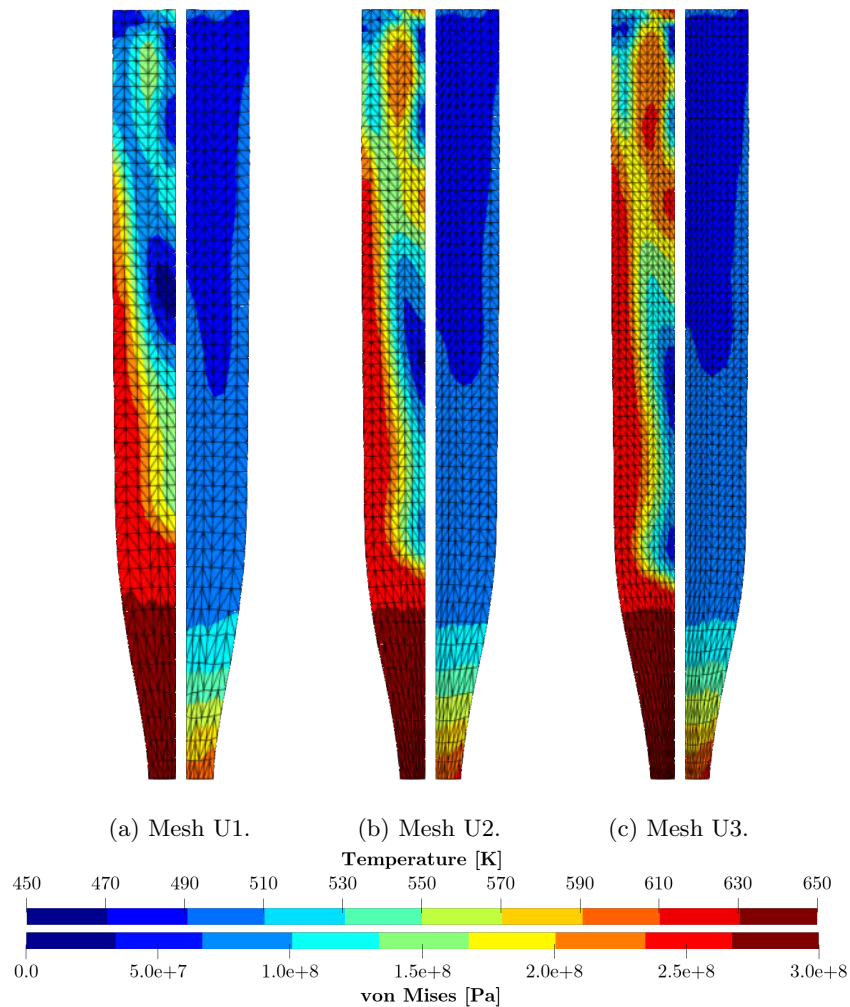


FIGURE 9.42: Bar necking with thermal coupling: Refinement analysis at time $t = 30$ ms with contour plots of von-Mises stress magnitude (left) and temperature (right), using the ALE formulation on meshes (a) U1, (b) U2, and (c) U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

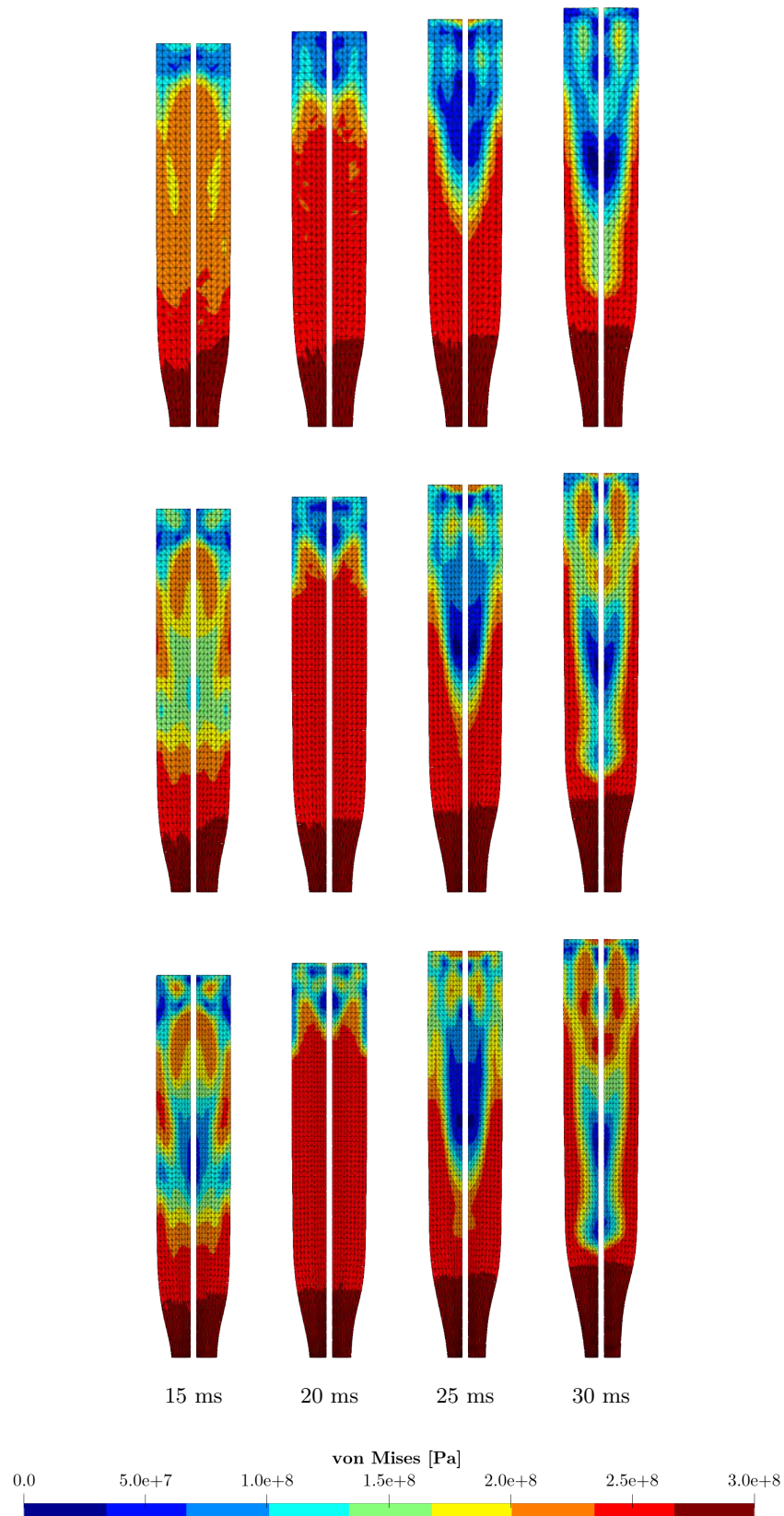


FIGURE 9.43: Bar necking impact with thermal coupling: Snapshots of the von-Mises contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

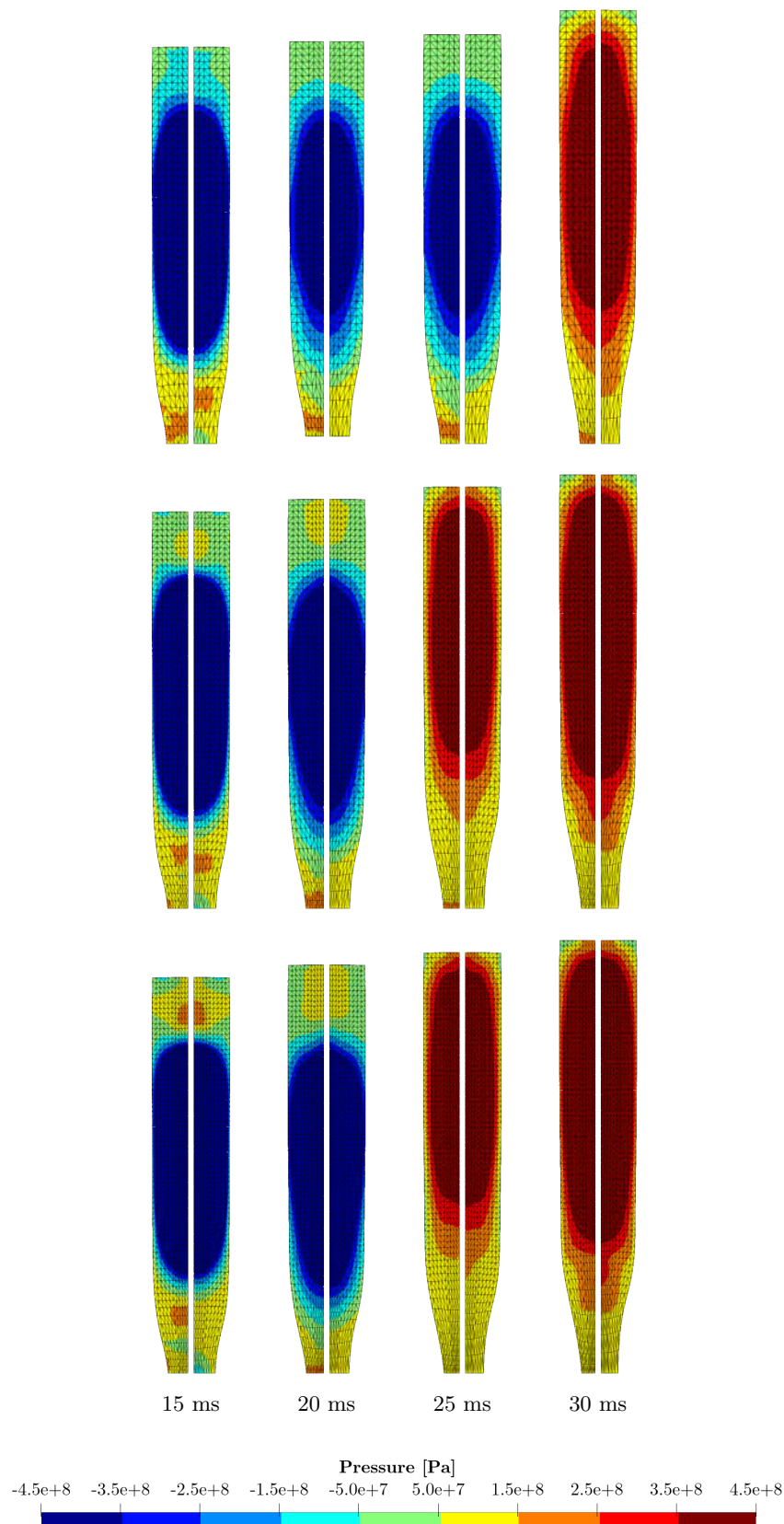


FIGURE 9.44: Bar necking impact with thermal coupling: Snapshots of the pressure contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

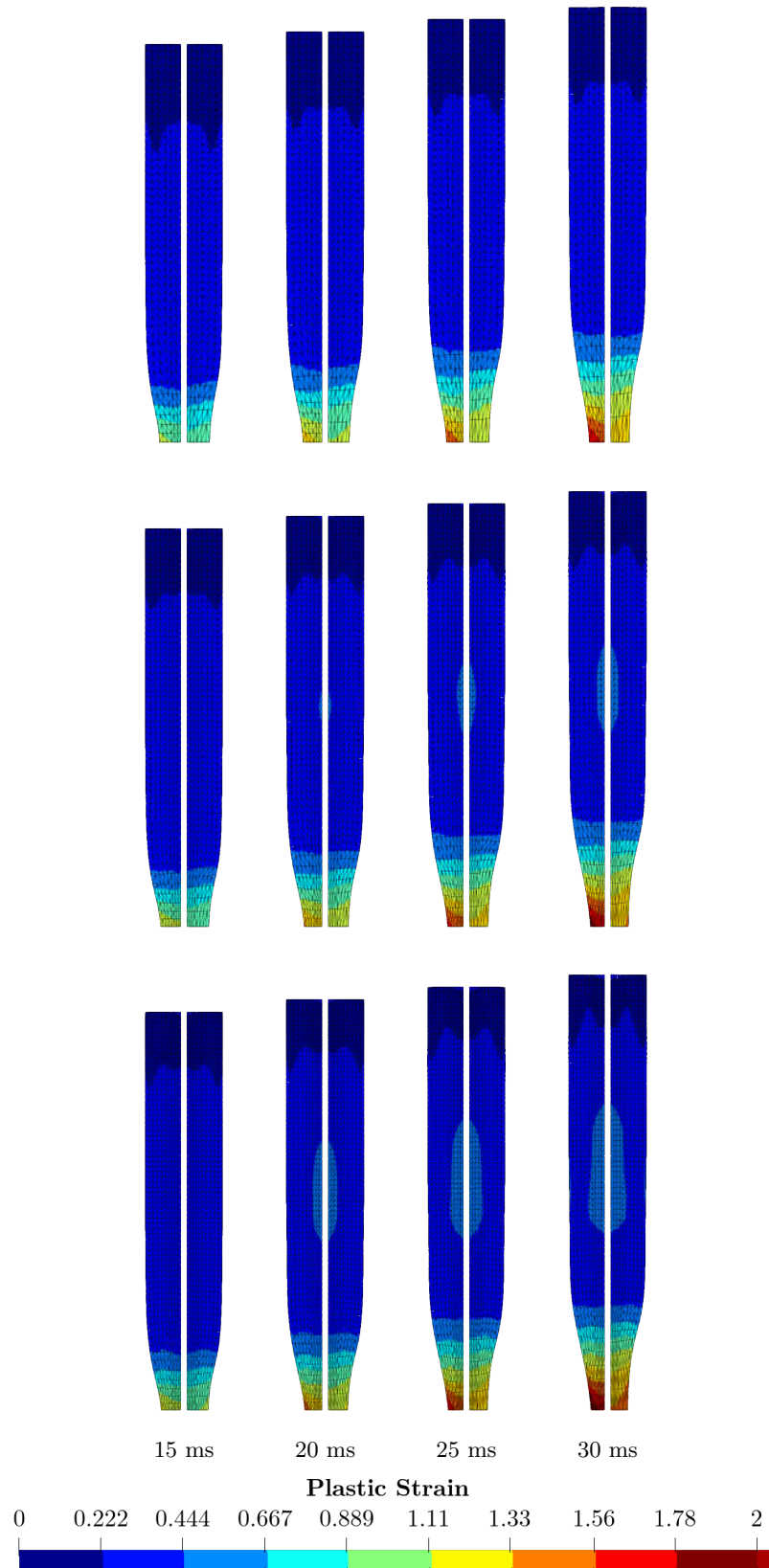


FIGURE 9.45: Bar necking impact with thermal coupling: Snapshots of the equivalent plastic strain at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

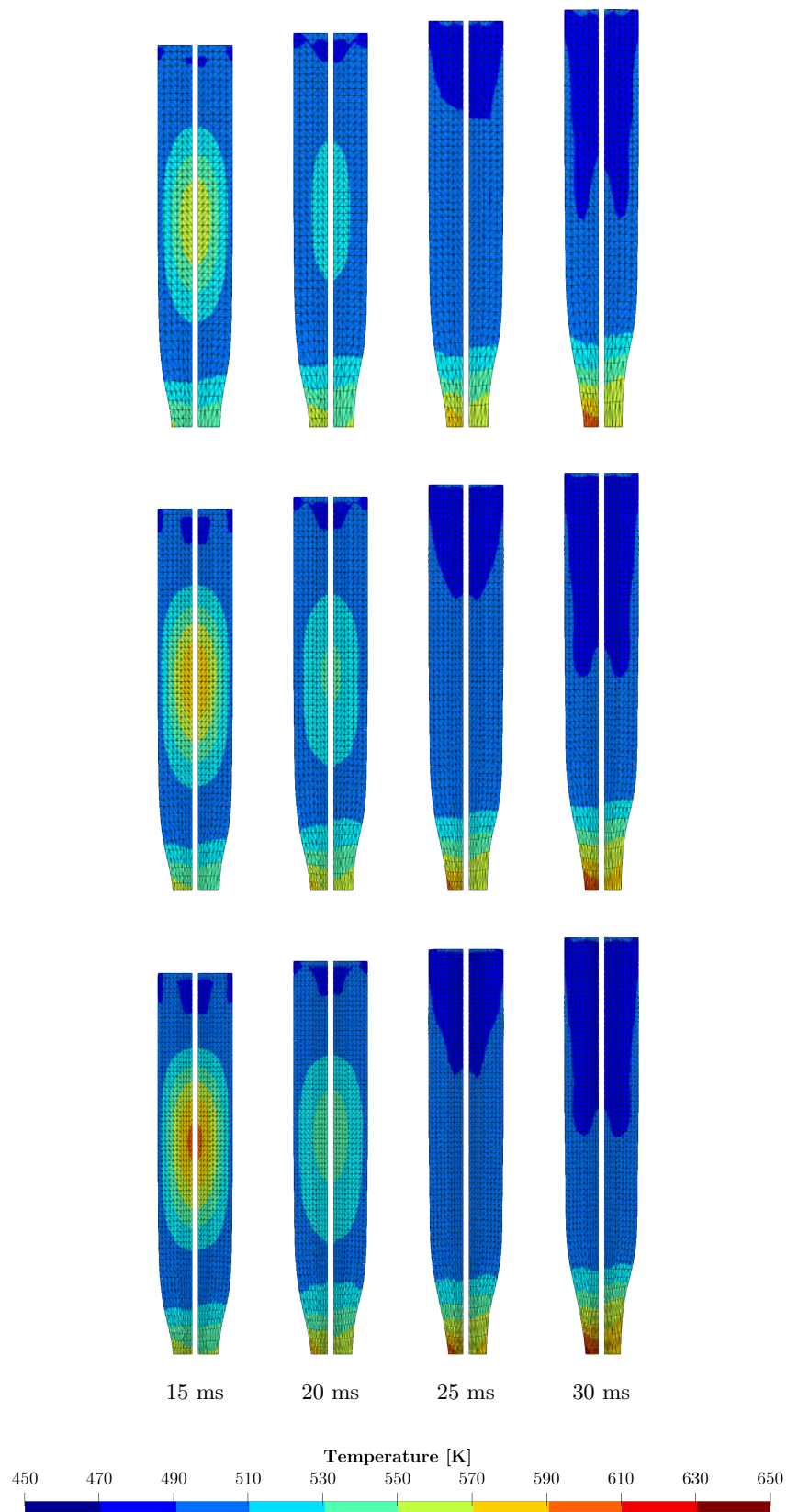


FIGURE 9.46: Bar necking impact with thermal coupling: Snapshots of the temperature contours at times $t = 15, 20, 25, 30$ ms, using the ALE formulation (left side) and the Total Lagrangian formulation (right side) on mesh U1 (first row), U2 (second row) and U3 (third row). Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

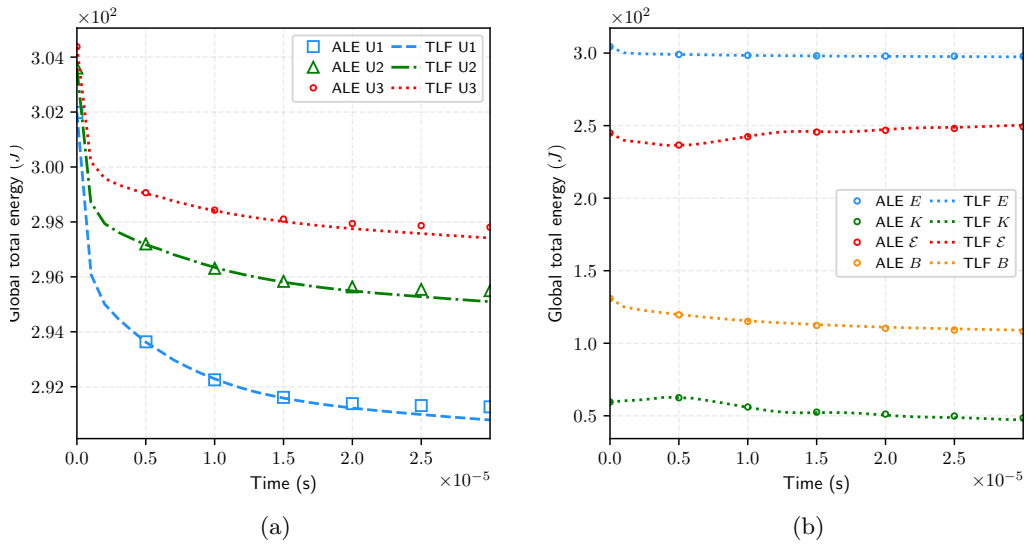


FIGURE 9.47: Bar necking with thermal coupling: (a) Total energy, and (b) energy components, for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 8.18.

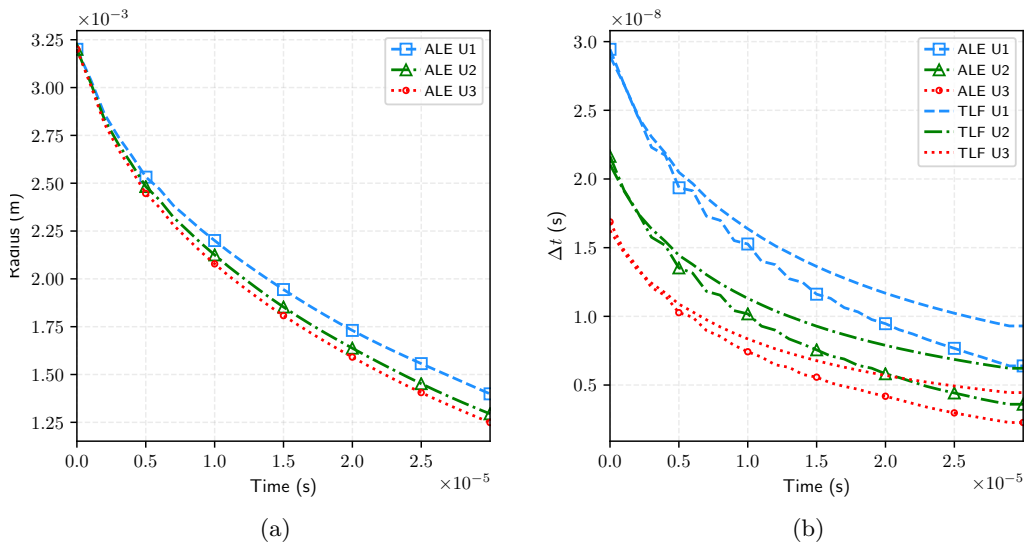


FIGURE 9.48: Bar necking with thermal coupling: Time evolution of (a) bottom radius, and (b) Δt , for the Total Lagrangian and the ALE formulation and on meshes U1, U2 and U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

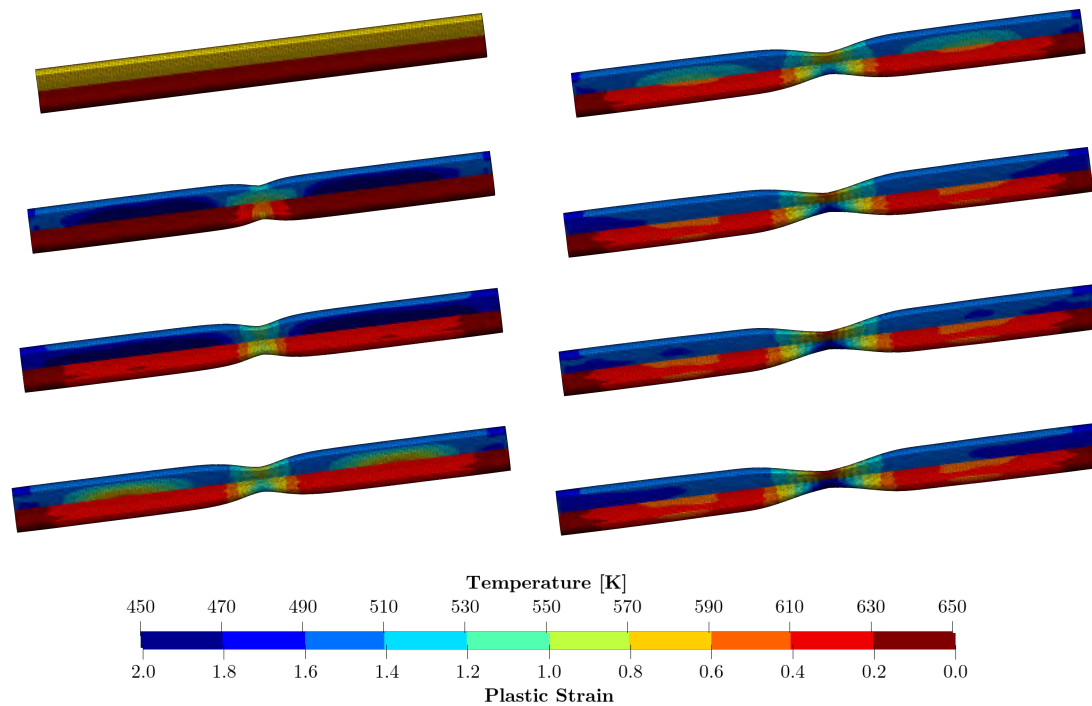


FIGURE 9.49: Bar necking with thermal coupling: Snapshots spatial deformation together with temperature and plastic strain contours at time $t = 0, 5, 10, 15$ ms (first column, top to bottom) and $t = 20, 25, 28, 30$ ms (second column, from top to bottom) using the ALE formulation on mesh U3. Results obtained using Hencky logarithmic strain, Johnson-Cook hardening and von-Mises plasticity with material parameters summarised in Table 9.10.

CHAPTER 10

CONCLUDING REMARKS

“The time is always ripe for the re-interpretation of theories in the light of new vision and of new facts. This is the very province of science.”

— CHARLES DARWIN, *The Life and Letters of Charles Darwin*, 1887

10.1 General conclusions

A novel Arbitrary Lagrangian-Eulerian framework for large strain solid dynamics was presented in this work. In particular, the cases of elasticity and plasticity were considered in isothermal scenarios, and with thermal-coupling. The objective of this framework is to be a robust alternative to the existing technology for simulating fast solid dynamics problems.

To begin with, the motivation for developing the ALE formulation came from the observation that challenging problems in Computational Structural Dynamics often display heavily distorted mesh, due to the presence of strong plastic deformations. Methodologies to alleviate numerical instabilities coming from heavy mesh distortions are plethora, as these phenomena are observed in a large quantity of real-life applications. The approach used herein consists in embedding the balance of material linear momentum in a variational framework [3–5]. That conservation law is closed by means of a hyperelastic constitutive model on the material stresses [6]. Moreover, the natural (or physical) deformation gradient \mathbf{F} is recovered as a multiplicative decomposition of the spatial deformation gradient and the inverse of the material deformation gradient. The geometric mappings between the referential domain and the two other domains are also provided with a conservative equation that can be integrated in the variational formulation. These equations are all encompassed into the mixed-based methodology developed by our group [2, 22, 55, 58, 77] to yield a system of first order hyperbolic conservation laws expressed on the referential domain. Inasmuch as the velocities and stresses, both material and spatial, are governed by balance laws integrated in the mixed formulation, the convergence rate of all these primary variables is equivalent. The primary unknown variables that are considered include the linear momentum \mathbf{p}_χ , the spatial deformation gradient \mathbf{F}_Φ , the material deformation gradient \mathbf{F}_Ψ , and the material Jacobian J_Ψ . In addition to that, ALE conservation equations were also given for the entropy η and the spatial deformation gradient J_Φ ; conservative forms can also be considered for the material cofactor \mathbf{H}_Ψ , the spatial cofactor \mathbf{H}_Φ and for the total Energy⁵⁵ E .

In the viewpoint of this new ALE formulation for solids, the balance of the linear momentum \mathbf{p}_χ depends on a measure of stress, as it is the case for the equivalent Lagrangian form of the mixed formulation. Therefore, constitutive law are also required for closure of the system. In this work, the polyconvex Mooney-Rivlin model is introduced to provide a closed-form relation between the first Piola Kirchhoff stress tensor \mathbf{P} and the triplet of deformation measure \mathcal{X} . Additionally, an isotropic von-Mises plasticity model is also presented. For completeness, an analysis of the eigenproblem is also performed to ensure the existence of real wave speeds, and consequently material stability and the satisfaction of the rank one convexity condition. The relationship between ALE wave speeds and wave speeds obtained in Lagrangian form are highlighted.

For closure of a thermo-mechanical system, the equations have also been provided with a Mie-Grüneisen equation of state. In the case of visco-plasticity, the Johnson-Cook hardening law is provided. The thermal considerations are possible due to the incorporation in the mixed system of the conservation of entropy. Alternatively, the conservation of energy is also provided. As for the other primary variables, the conserved thermal variables are provided with a new ALE convection term, and a modified heat flux.

The resolution of the mesh motion is the cornerstone of the ALE machinery presented in this work. The mesh motion is obtained via a conservation law for the material linear momentum, where stresses are obtained by making use of a constitutive closed form for

⁵⁵These additional conservation laws are shown in Appendix C. More particularly, the conservation law for the total energy is given in terms of the scaled energy $E_\chi = J_\Psi E$.

hyperelastic materials. In as much that the local distortions of the spatial mesh are captured, the ALE approach aims at distributing the deformation via the computation of the material stress. Typically, the mesh motion will pull elements in a direction that is opposed to that of the spatial mesh. As the quality of elements in the current configuration is improved, better physical responses are recovered for quantities such as the pressure and plastic internal variables. That approach also limits the need for local geometric rezoning, and gives access to the material mapping at no extra cost. The latter is useful to study the caveats of the mesh motion, as material deformation gradients may still suffer from excessive local distortion. The flexibility of the proposed ALE methodology, combined with the riddance of an *ad hoc* procedure for determining the mesh motion, makes it a competitive alternative for problems with strong and permanent distortions such as plasticity.

The ALE formulation is spatially discretised using the Vertex-Centred Finite Volume Method with linear tetrahedra. The solution of the present ALE framework is dramatically improved in the vicinity of sharp gradients by taking advantage of shock capturing techniques: suitable numerical interface fluxes are obtained via an upwinding stabilisation technique based on an acoustic Riemann solver, a methodology borrowed from the CFD community as for the Lagrangian frameworks previously developed in the research group [2, 37, 75]. In order to achieve second order spatial accuracy, a local piecewise linear reconstruction procedure is carried out to compute slopes within elements. The conserved material and spatial momenta, as well as their fluxes and the thermal variables, are provided with boundary conditions expressed in the referential domain. In addition to this, the temporal discretisation is handled by a monolithic multi-stage Total Variation Diminishing Runge-Kutta scheme. The CFL condition and the spatial stretch are used to compute the time increments. The staggered approach is another advantage of that framework: it is easy to implement and the ALE framework can degenerate to a Total Lagrangian framework in a simple manner by deactivating the advection step.

The implementation of that ALE framework was carried out on the open-source package OpenFOAM under the name *vcALEFoam*. The main objective was to further continue the endeavour of [75] to improve the capabilities of solid solvers on the platform. As OpenFOAM is originally a Cell-Centred Finite Volume based environment for Computational Fluid Dynamics, the implementation was made from scratch. However, an effort was put in using as much of the already existing OpenFOAM machinery as possible. This, combined with the fact that the set of first-order hyperbolic ALE equations are stabilised with an acoustic Riemann solver, aims to bridge the gap between Computational Fluid Dynamics and Computational Solid Dynamics.

Numerical examples were used to assess the ALE framework. It has been showed that in the context of elasticity, plasticity, thermo-elasticity with volumetric thermal coupling and thermo visco-plasticity, the ALE framework was proven to verify a series of properties, and showed an array of advantages. For stability of the material motion, the resolution of the DGCL was shown to be of paramount importance. One clear advantage of the proposed ALE framework is the equal second-order of convergence for all derived variables, in the context of elasticity and thermoelasticity. The robustness was proven in an isothermal and a thermally-coupled scenario with a tumbling L-shaped block after external work was applied. Also, the linear momentum is successfully preserved, and the angular momenta shows little numerical dissipation despite the absence of solving the respective conservation law. The bending of near incompressible materials was also proven to be possible. In impact problems, the ALE formulation was shown to produce a deformation pattern and contours plots that are coherent with the literature and with the results using the Total Lagrangian formulation. Additionally, it was observed that the method was capable of automatically moving the mesh to improve its quality in crucial high-plasticity areas. In

<ul style="list-style-type: none"> • Mixed formulation of first-order conservation laws in ALE form for thermal visco-plasticity, including balance laws and geometric conservation equations for the spatial and material quantities. 	Chapters chapters 2 to 5
<ul style="list-style-type: none"> • Explicit numerical framework based on Vertex-centred Finite Volume discretisation and multistage Runge Kutta time integrator, making use of tailor-made upwind stabilisation. 	Chapter chapter 6
<ul style="list-style-type: none"> • Demonstration of Hyperbolicity and of the fulfilment of the second law of thermodynamics in an ALE setting. 	Chapter 3, Appendix C
<ul style="list-style-type: none"> • Full OpenFOAM implementation. 	Chapter 7

TABLE 10.1: Novelties of this thesis

these scenarios, an added benefit is that the time increments are higher due to the better overall shape of elements. For necking scenarios, the use of ALE offers the possibility of a trade off between smaller time increments and a finer definition of the specimen in the neck region, and more especially regarding the shape of the boundary. By activating the mesh motion, the ALE framework was able to pull the elements down as if a local refinement was applied dynamically. These novelties are summarised in Table 10.1.

10.2 Future works

Following on what has been established the research group [2, 20, 56, 58, 75, 199, 307], this author of this thesis also reckons the outstanding science contributions of *Huerta* [113, 124, 160, 161, 177, 255, 308] and *Bonet* [10, 195–197, 206, 207, 232], as well as many others [3–5, 13, 42, 49, 66, 93, 292, 309]. Furthermore, future research lines have emerged and pointed out in the following:

- **Alternative ALE motion and mesh adaptivity:**

The ALE motion used in this work relies on a constitutive neo-Hookean model. In Appendix C, other techniques to determine material stresses were presented: a generalised neo-Hookean model, a constitutive law based on linear elasticity, and a plasticity-based stress model. Also, the deformation of the material mesh can be considered in specific areas of the body. The current ALE method can be compared with already existing ALE technology: Radial based functions, Laplacian smoothing, or analytically prescribed mesh motion. Eventually, and by exploiting the structure of the mesh motion machinery in this work, the material stresses could theoretically be computed with any relevant measure (e.g. equivalent plastic strain in the context of inelastic deformations).

- **Extension to Fluid-Structure Interactions:**

Further physical considerations can make use of the present ALE approach. The capability of the ALE numerical framework to allow for sliding nodes on the boundary can be exploited in Fluid-Structures Interactions problems. In popular approaches such as *preCICE* [63, 310–312], a fixed-point problem is solved for an array of variables at the interface between two (fluid and/or structural) bodies. Hessesenthaler et al. [313] is resorting to a monolithic ALE FSI solver for non-conforming interfaces. In FSI

problems, the complex nature of boundaries implies that the meshes at the interface are highly non-conforming. This makes the ALE framework easily compatible with the aforementioned interface solver.

- **Exploitation of advanced interface solvers:**

In this work, an acoustic Riemann solver was implemented in the ALE framework, for the evaluation of contact fluxes. This technology is borrowed from CFD and has recently been applied to solid dynamics [20, 37, 75, 110]. More advanced alternatives such as Roe's [268] or Osher's [271] Riemann solvers can be integrated to the framework for better shock capturing capabilities.

- **Parallelisation and optimisation:**

The ALE framework in this work is implemented in the OpenFOAM software. In the endeavour to better comply with OpenFOAM's community requirements, the code could make more use of design patterns [314] such as templates and observers. Further optimisation of the present code, through the parallelisation of loops like the construction of the right hand sides, would be a welcome improvement. As another option, the exploration of an implementation using different programming languages could be interesting from the pure computational point of view. While C++ is fast and object-oriented, the use of Fortran in specific parts of the codes with high computational cost could potentially make the code faster. The languages Julia (used in data science and scientific computing) and Rust (which prioritise performance and safety) is also an appealing idea.

APPENDIX A

MATHEMATICAL FOUNDATIONS

“Number is the ruler of forms and ideas.”

— PYTHAGORAS

A.1 Introduction

This appendix ratifies the nomenclature used in this thesis, and summarises the useful general mathematical concepts. Further developments can be found in [1, 2, 10, 195, 224, 315]. First, generic objects and useful results of linear algebra will be recalled in Section A.2. Then, Section A.3 will present the differential operators that are essential to derive conservation laws in solid mechanics. Throughout this chapter, the results may be either expressed in the referential configuration $\Omega_{\boldsymbol{\chi}}$ by means of the referential position notation $\boldsymbol{\chi}$ (e.g. the gradient $\nabla_{\boldsymbol{\chi}}$), the spatial configuration $\Omega_{\boldsymbol{x}}$ by means of the spatial position \boldsymbol{x} (e.g. the scalar function $f(\boldsymbol{x}, t)$), or the material configuration $\Omega_{\boldsymbol{X}}$ by means of the material position \boldsymbol{X} (e.g. the scalar function $f^{**}(\boldsymbol{X}, t)$). The choice of a specific configuration to express a property is made for convenience, and since it is crucial to understand how each configuration is linked to one another, the reader may refer to the introduction of those configurations and the three mappings in Chapter 2. Also, the focus is on properties that are useful in the developments in this thesis, and that are non-trivial.

A.2 Useful Linear Algebra

Let \boldsymbol{I} be the second-order identity tensor and $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ be second-order tensors. Let $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ be first-order tensors defined, with a Cartesian base \boldsymbol{E}_I , as

$$\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = u_I \boldsymbol{E}_I; \quad \boldsymbol{E}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \quad \boldsymbol{E}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}; \quad \boldsymbol{E}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (\text{A.1a,b,c,d})$$

A.2.1 Useful Algebraic properties

Using Einstein's notation, the scalar product \cdot , the cross product \times , the dyadic product \otimes and the norm are respectively defined for first-order tensors as

$$\boldsymbol{u} \cdot \boldsymbol{v} = u_i v_i; \quad \boldsymbol{u} \times \boldsymbol{v} = \epsilon_{ijk} u_j v_k; \quad (\boldsymbol{u} \otimes \boldsymbol{v})_{ij} = u_i v_j; \quad \|\boldsymbol{u}\| = \sqrt{\boldsymbol{u} \cdot \boldsymbol{u}}; \quad (\text{A.2a,b})$$

with ϵ_{ijk} the Levi-Civita permutation tensor. By introducing the double dot contraction operator $:$, it is possible to define the Euclidean norm as

$$\boldsymbol{A} : \boldsymbol{B} = A_{ij} B_{ij}; \quad \|\boldsymbol{A}\| = \sqrt{\boldsymbol{A} : \boldsymbol{A}}. \quad (\text{A.3a,b})$$

Some properties of the contraction operator \cdot (or dot product) and of the double contraction operator are presented in the following set of equations:

$$\boldsymbol{u} \cdot (\boldsymbol{A}\boldsymbol{v}) = \boldsymbol{v} \cdot (\boldsymbol{A}^T \boldsymbol{u}), \quad (\text{A.4a})$$

$$\text{tr}(\boldsymbol{u} \otimes \boldsymbol{v}) = \boldsymbol{u} \cdot \boldsymbol{v}, \quad (\text{A.4b})$$

$$\boldsymbol{A} : (\boldsymbol{u} \otimes \boldsymbol{v}) = \boldsymbol{u} \cdot (\boldsymbol{A}\boldsymbol{v}), \quad (\text{A.4c})$$

$$\boldsymbol{A} : \boldsymbol{B} = \text{tr}(\boldsymbol{A}^T \boldsymbol{B}) = \text{tr}(\boldsymbol{B}\boldsymbol{A}^T) = \text{tr}(\boldsymbol{B}^T \boldsymbol{A}) = \text{tr}(\boldsymbol{A}\boldsymbol{B}^T). \quad (\text{A.4d})$$

This thesis makes use of the tensorial cross product $\boldsymbol{\times}$ introduced in [111, 196, 197, 200, 232] and further used in [20, 75]. Some useful properties are

$$(\boldsymbol{A} \boldsymbol{\times} \boldsymbol{B})_{ij} = \epsilon_{ikl} \epsilon_{jmn} A_{km} B_{ln}; \quad (\boldsymbol{A} \boldsymbol{\times} \boldsymbol{u})_{ij} = \epsilon_{jkl} A_{ik} u_l; \quad (\boldsymbol{u} \boldsymbol{\times} \boldsymbol{A})_{ij} = \epsilon_{ikl} u_k A_{lj}. \quad (\text{A.5a,b,c})$$

In the following, $\boldsymbol{v}, \boldsymbol{w}$, are first-order tensors from one configuration Ω and $\boldsymbol{V}, \boldsymbol{W}$ are first-order tensors from another configuration $\Omega_{\boldsymbol{X}}$; a, b, c are scalars. The operator $\boldsymbol{\times}$ refers

to the classical vectorial cross product. The symbol δ represents the Kronecker delta operator, \det represents the determinant and tr represents the trace of a second-order tensor.

$$\mathbf{A} \times \mathbf{B} = \mathbf{B} \times \mathbf{A}, \quad (\text{A.6a})$$

$$\mathbf{A} \times \mathbf{B} = \mathbf{A}^T \times \mathbf{B}^T, \quad (\text{A.6b})$$

$$a\mathbf{A} \times (b\mathbf{B} + c\mathbf{C}) = ab(\mathbf{A} \times \mathbf{B}) + ac(\mathbf{A} \times \mathbf{C}), \quad (\text{A.6c})$$

$$(\mathbf{v} \otimes \mathbf{V}) \times (\mathbf{w} \otimes \mathbf{W}) = (\mathbf{v} \times \mathbf{w}) \otimes (\mathbf{V} \times \mathbf{W}), \quad (\text{A.6d})$$

$$\mathbf{v} \times (\mathbf{A} \times \mathbf{V}) = (\mathbf{v} \times \mathbf{A}) \times \mathbf{V} = \mathbf{v} \times \mathbf{A} \times \mathbf{V} = -\mathbf{A} \times (\mathbf{v} \otimes \mathbf{V}), \quad (\text{A.6e})$$

$$(\mathbf{A} \times \mathbf{B}) : \mathbf{C} = (\mathbf{B} \times \mathbf{C}) : \mathbf{A} = (\mathbf{A} \times \mathbf{C}) : \mathbf{B}, \quad (\text{A.6f})$$

$$(\mathbf{A} \times \mathbf{B})(\mathbf{V} \times \mathbf{W}) = (\mathbf{A}\mathbf{V}) \times (\mathbf{B}\mathbf{W}) + (\mathbf{B}\mathbf{W}) \times (\mathbf{A}\mathbf{W}), \quad (\text{A.6g})$$

$$\mathbf{A} \times \mathbf{I} = (\text{tr}\mathbf{I})\mathbf{A} - \mathbf{A}^T, \quad (\text{A.6h})$$

$$(\mathbf{A} \times \mathbf{A}) : \mathbf{A} = 6 \det \mathbf{A}, \quad (\text{A.6i})$$

$$(\mathbf{A}\mathbf{C}) \times (\mathbf{B}\mathbf{C}) = \frac{1}{2}(\mathbf{A} \times \mathbf{B})(\mathbf{C} \times \mathbf{C}). \quad (\text{A.6j})$$

A.3 Differential calculus

In the following, a succinct presentation of the operators generally found in partial differential equations will be presented. Further discussion can be found for instance in [8, 10, 196, 316]. In the following, tensors are considered differentiable enough for the application of the differential operators, and are expressed at the position $\boldsymbol{\chi} \in \Omega_{\boldsymbol{\chi}}$. The notation introduced in the previous section is also used here: f be a zero-order tensor (a scalar field $\Omega_{\boldsymbol{\chi}}$), \mathbf{v} a first-order tensor (a vector field in $\Omega_{\boldsymbol{\chi}}$) and \mathbf{A} a second-order tensor (a matrix field in $\Omega_{\boldsymbol{\chi}}$). The notation of this section applies directly to the referential domain $\Omega_{\boldsymbol{\chi}}$ of the ALE formulation, and can nonetheless be applied to other domain, particularly those mappable to the $\Omega_{\boldsymbol{\chi}}$.

A.3.1 Differential operators

The gradient $\nabla_{\boldsymbol{\chi}} f(\boldsymbol{\chi}_0) \cdot \mathbf{u}$ of f at the point $\boldsymbol{\chi}_0$ for an arbitrary incremental vector \mathbf{u} is defined as the directional derivative of f at $\boldsymbol{\chi}_0$ in the direction \mathbf{u}

$$\nabla_{\boldsymbol{\chi}} f(\boldsymbol{\chi}_0) \cdot \mathbf{u} = Df(\boldsymbol{\chi}_0)[\mathbf{u}] = \left. \frac{d}{d\epsilon} f(\boldsymbol{\chi}_0 + \epsilon\mathbf{u}) \right|_{\epsilon=0} = \sum_{i=1}^3 u_i \left. \frac{\partial f}{\partial \chi_i} \right|_{\chi_i=\chi_{0,i}}, \quad (\text{A.7})$$

which can also be written for $i \in \{1, 2, 3\}$ as

$$\nabla_{\boldsymbol{\chi}} f = \sum_{i=1}^3 \frac{\partial f}{\partial \chi_i} \mathbf{E}_i = \frac{\partial f}{\partial \boldsymbol{\chi}}; \quad (\nabla_{\boldsymbol{\chi}} f)_i = \frac{\partial f}{\partial \chi_i}. \quad (\text{A.8a,b})$$

Similarly, the gradient of the first-order tensor \mathbf{v} verifies

$$\nabla_{\boldsymbol{\chi}} \mathbf{v}(\boldsymbol{\chi}_0) \mathbf{u} = D\mathbf{v}(\boldsymbol{\chi}_0)[\mathbf{u}]; \quad (\nabla_{\boldsymbol{\chi}} \mathbf{v})_{ij} = \frac{\partial v_i}{\partial \chi_j}, \quad (\text{A.9a,b})$$

and the gradient of the second-order tensor \mathbf{A} verifies

$$\nabla_{\boldsymbol{\chi}} \mathbf{A}(\boldsymbol{\chi}_0) \mathbf{u} = D\mathbf{A}(\boldsymbol{\chi}_0)[\mathbf{u}]; \quad (\nabla_{\boldsymbol{\chi}} \mathbf{A})_{ijk} = \frac{\partial A_{ij}}{\partial \chi_k}, \quad (\text{A.10a,b})$$

The divergence operator can then be derived, for first-order tensors, as

$$\nabla_{\mathbf{x}} \cdot \mathbf{v} = \nabla_{\mathbf{x}} \mathbf{v} : \mathbf{I} = \text{tr} \nabla_{\mathbf{x}} \mathbf{v}; \quad (\nabla_{\mathbf{x}} \cdot \mathbf{v}) = \frac{\partial v_i}{\partial \chi_i}, \quad (\text{A.11a,b})$$

and for second-order tensors as

$$\nabla_{\mathbf{x}} \cdot \mathbf{A} = \nabla_{\mathbf{x}} \mathbf{A} : \mathbf{I}; \quad (\nabla_{\mathbf{x}} \cdot \mathbf{A})_i = \frac{\partial A_{ij}}{\partial \chi_j}, \quad (\text{A.12a,b})$$

The rotational operator, or curl operator, can eventually be defined for a first- and second-order tensor as

$$(\nabla_{\mathbf{x}} \times \mathbf{v})_i = \epsilon_{ijk} \frac{\partial v_j}{\partial \chi_k}; \quad (\nabla_{\mathbf{x}} \times \mathbf{A})_{ij} = \epsilon_{jkl} \frac{\partial A_{il}}{\partial \chi_k}. \quad (\text{A.13a,b})$$

As a general rule of thumb, it is useful to remember that for a (n) -order tensor, the gradient operator corresponds to a $(n+1)$ -order tensor, while the divergence operator corresponds to a $(n-1)$ -order tensor and the curl operator (of a first- or second-order tensor) corresponds to a tensor of the same order. The following properties of the differential operators are used throughout this work:

$$\nabla_{\mathbf{x}}(f\mathbf{v}) = f\nabla_{\mathbf{x}}\mathbf{v} + \mathbf{v} \otimes \nabla_{\mathbf{x}}f; \quad (\text{A.14a})$$

$$\nabla_{\mathbf{x}} \cdot (f\mathbf{v}) = f\nabla_{\mathbf{x}} \cdot \mathbf{v} + \mathbf{v} \cdot \nabla_{\mathbf{x}}f; \quad (\text{A.14b})$$

$$\nabla_{\mathbf{x}}(\mathbf{v} \cdot \mathbf{w}) = (\nabla_{\mathbf{x}}\mathbf{v})^T \mathbf{w} + (\nabla_{\mathbf{x}}\mathbf{w})^T \mathbf{v}; \quad (\text{A.14c})$$

$$\nabla_{\mathbf{x}} \cdot (\mathbf{v} \otimes \mathbf{w}) = \mathbf{v} \nabla_{\mathbf{x}} \cdot \mathbf{w} + (\nabla_{\mathbf{x}}\mathbf{w}); \quad (\text{A.14d})$$

$$\nabla_{\mathbf{x}} \cdot (\mathbf{A}^T \mathbf{v}) = \mathbf{A} : \nabla_{\mathbf{x}}\mathbf{v} + \mathbf{v} \cdot \nabla_{\mathbf{x}} \cdot \mathbf{A}; \quad (\text{A.14e})$$

$$\nabla_{\mathbf{x}} \cdot (f\mathbf{A}) = f\nabla_{\mathbf{x}} \cdot \mathbf{A} + \mathbf{A} \nabla_{\mathbf{x}}f; \quad (\text{A.14f})$$

$$\nabla_{\mathbf{x}}(f\mathbf{A}) = f\nabla_{\mathbf{x}}\mathbf{A} + \mathbf{A} \otimes \nabla_{\mathbf{x}}f. \quad (\text{A.14g})$$

A.4 Transformation of equations in the continuum

Considering the mapping⁵⁶ relations between the different referential, material and spatial domains, there exist a relationship⁵⁷ between scalar functions defined as

$$f(\mathbf{x}, t) = f^*(\boldsymbol{\chi}, t) = f^{**}(\mathbf{X}, t), \quad (\text{A.15})$$

Such functions, as all the conserved quantities utilised in this work, are assumed to satisfy the requirement of being twice continuously differentiable for the Schwarz symmetry theorem.

A.4.1 On Green-Ostrogradsky theorem

The Green-Ostrogradsky theorem, also denoted as the divergence theorem, relates the integral of the divergence of a first-order tensor \mathbf{v} or a second-order tensor \mathbf{A} on the continuum $\Omega_{\boldsymbol{\chi}}$, with an integral on the surface $\partial\Omega_{\boldsymbol{\chi}}$ of that continuum, as

$$\int_{\Omega_{\boldsymbol{\chi}}} \nabla_{\mathbf{x}} \cdot \mathbf{v} \, dV_{\boldsymbol{\chi}} = \int_{\partial\Omega_{\boldsymbol{\chi}}} \mathbf{v} \cdot \mathbf{N}_{\boldsymbol{\chi}} \, dA_{\boldsymbol{\chi}}; \quad \int_{\Omega_{\boldsymbol{\chi}}} \nabla_{\mathbf{x}} \cdot \mathbf{A} \, dV_{\boldsymbol{\chi}} = \int_{\partial\Omega_{\boldsymbol{\chi}}} \mathbf{A} \mathbf{N}_{\boldsymbol{\chi}} \, dA_{\boldsymbol{\chi}}. \quad (\text{A.16a,b})$$

⁵⁶Refer to Figure 2.1 in Section 2.2 for visual support.

⁵⁷See [3, 4] for further development.

A.4.2 On Reynold's Transport theorem

Using the notation of the previous section, the following development shows how to transform a scalar quantity f , integrated on a spatial domain, onto a material domain:

$$\begin{aligned}
\frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_x} f(\mathbf{x}, t) dv_x &= \frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_X} f^{**}(\mathbf{X}, t) J^{**}(\mathbf{X}, t) dV \\
&= \int_{\Omega_X} \frac{\partial}{\partial t} \Big|_{\mathbf{X}} (f^{**}(\mathbf{X}, t) J^{**}(\mathbf{X}, t)) dV \\
&= \int_{\Omega_X} \left[\frac{\partial f^{**}}{\partial t} \Big|_{\mathbf{X}} J^{**}(\mathbf{X}, t) + \frac{\partial J^{**}}{\partial t} \Big|_{\mathbf{X}} f^{**}(\mathbf{X}, t) \right] dV \\
&= \int_{\Omega_X} \left[\frac{\partial f^{**}}{\partial t} + f^{**} \nabla_{\mathbf{x}} \cdot \mathbf{v} \right] J^{**} dV \\
&= \int_{\Omega_x} \frac{\partial f^{**}}{\partial t} + f^{**} \nabla_{\mathbf{x}} \cdot \mathbf{v} dv \\
&= \int_{\Omega_x} \frac{\partial f^{**}}{\partial t} \Big|_{\mathbf{x}} + \nabla_{\mathbf{x}} f \cdot \mathbf{v} + f \nabla_{\mathbf{x}} \cdot \mathbf{v} dv \\
&= \int_{\Omega_x} \frac{\partial f^{**}}{\partial t} \Big|_{\mathbf{x}} + \nabla_{\mathbf{x}} \cdot (f \mathbf{v}) dv \\
&= \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_x} f(\mathbf{x}, t) dv + \int_{\partial \Omega_x} f \mathbf{v} \cdot \mathbf{n} da. \tag{A.17a}
\end{aligned}$$

This relation is very useful to transform equations written with an Eulerian framework to equations written with a Lagrangian framework. Using the same algebra, it is also possible to recover the following relation on the integral form

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_x} f(\mathbf{x}, t) dv = \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_x} f(\mathbf{x}, t) dv + \int_{\partial \Omega_x} f \hat{\mathbf{v}} \cdot \mathbf{n} dv. \tag{A.18}$$

The Reynold's transport theorem can be usefully stated for the material time rate and the referential time rate of a scalar variable \mathcal{U}_R as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_X} \mathcal{U}_R dV = \frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_X} \mathcal{U}_R dV + \int_{\partial \Omega_X} \mathcal{U}_R \mathbf{W} \cdot \mathbf{N}_X dA, \tag{A.19a}$$

$$= \frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_X} \mathcal{U}_R dV + \int_{\partial \Omega_X} (\mathcal{U}_R \mathbf{H}_{\Psi}^T \mathbf{W}) \cdot \mathbf{N}_X dA_X, \tag{A.19b}$$

$$= \int_{\Omega_X} \left[\frac{\partial \mathcal{U}_R}{\partial t} \Big|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot \mathcal{U}_R \mathbf{W} \right] dV, \tag{A.19c}$$

and for a tensorial variable \mathbf{u}_R as

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_X} \mathbf{u}_R dV = \frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_X} \mathbf{u}_R dV + \int_{\partial \Omega_X} (\mathbf{u}_R \otimes \mathbf{W}) \mathbf{N}_X dA, \tag{A.20a}$$

$$= \frac{\partial}{\partial t} \Big|_{\mathbf{X}} \int_{\Omega_X} \mathbf{u}_R dV + \int_{\partial \Omega_X} (\mathbf{u}_R \otimes \mathbf{W}) \mathbf{H}_{\Psi}^T \mathbf{N}_X dA_X, \tag{A.20b}$$

$$= \int_{\Omega_X} \left[\frac{\partial \mathbf{u}_R}{\partial t} \Big|_{\mathbf{X}} + \nabla_{\mathbf{x}} \cdot (\mathbf{u}_R \otimes \mathbf{W}) \right] dV. \tag{A.20c}$$

Note that in Equations (A.19) and (A.20), the second line is obtained by using Nanson's rule on the second term of the first line, and the third line is obtained by regrouping the terms of the first line under a single integral.

A.4.3 On Euler's theorem

By using algebra, the following relationships arise:

$$\begin{aligned} \frac{\partial f^{**}}{\partial t}(\mathbf{X}, t) &= \frac{\partial f^*}{\partial t}(\boldsymbol{\chi} = \boldsymbol{\Psi}^{-1}(\mathbf{X}, t), t) \\ &= \frac{\partial f^*}{\partial \boldsymbol{\chi}}(\boldsymbol{\chi}, t) \cdot \frac{\partial \boldsymbol{\Psi}^{-1}}{\partial t} + \frac{\partial f^*}{\partial t}(\boldsymbol{\chi}, t) = \nabla_{\boldsymbol{x}} f^* \cdot \mathbf{W} + \frac{\partial f^*}{\partial t}. \end{aligned} \quad (\text{A.21})$$

This transformation is very useful in solid mechanics because it enables time derivatives in Lagrangian formulations to be rewritten in a referential domain. A similar relation can be obtained between the spatial and the material domains as

$$\frac{\partial f^{**}}{\partial t}(\mathbf{X}, t) = \frac{\partial f^*}{\partial t}(\boldsymbol{x} = \boldsymbol{\varphi}(\mathbf{X}, t), t) = \nabla_{\boldsymbol{x}} f \cdot \boldsymbol{v} + \frac{\partial f}{\partial t}, \quad (\text{A.22})$$

and between the spatial and the material domain as

$$\frac{\partial f^*}{\partial t}(\boldsymbol{\chi}, t) = \frac{\partial f}{\partial t}(\boldsymbol{x} = \boldsymbol{\Phi}(\boldsymbol{\chi}, t), t) = \nabla_{\boldsymbol{x}} f \cdot \hat{\boldsymbol{v}} + \frac{\partial f}{\partial t}. \quad (\text{A.23})$$

Equations section A.4.3 and (A.22) and (A.23) can alternatively be written as

$$\left. \frac{\partial f}{\partial t} \right|_{\mathbf{X}} = \left. \frac{\partial f}{\partial t} \right|_{\boldsymbol{\chi}} + \nabla_{\boldsymbol{x}} f \cdot \mathbf{W}; \quad \left. \frac{\partial f}{\partial t} \right|_{\mathbf{X}} = \left. \frac{\partial f}{\partial t} \right|_{\boldsymbol{x}} + \nabla_{\boldsymbol{x}} f \cdot \boldsymbol{v}; \quad \left. \frac{\partial f}{\partial t} \right|_{\boldsymbol{\chi}} = \left. \frac{\partial f}{\partial t} \right|_{\boldsymbol{x}} + \nabla_{\boldsymbol{x}} f \cdot \hat{\boldsymbol{v}}. \quad (\text{A.24a,b,c})$$

These equations can be combined to obtain

$$\left. \frac{\partial f}{\partial t} \right|_{\mathbf{X}} = \left. \frac{\partial f}{\partial t} \right|_{\boldsymbol{x}} + \nabla_{\boldsymbol{x}} f \cdot (\boldsymbol{v} - \hat{\boldsymbol{v}}), \quad (\text{A.25})$$

and constitute Euler's theorem. Alternatively, using Equation (A.19), Euler's theorem can be stated for a scalar quantity \mathcal{U} as

$$\left. \frac{\partial J_{\boldsymbol{\Psi}} \mathcal{U}}{\partial t} \right|_{\boldsymbol{\chi}} = \left. \frac{\partial \mathcal{U}}{\partial t} \right|_{\mathbf{X}} + \nabla_{\boldsymbol{x}} \cdot (\mathcal{U} \mathbf{W}), \quad (\text{A.26})$$

APPENDIX B

FUNDAMENTALS OF SOLID MECHANICS

“I do not define time, space, place, and motion, as being well known to all.”

— ISAAC NEWTON, *Philosophiæ Naturalis Principia Mathematica*,
1687

B.1 Introduction

In this appendix, the general principles of large strain, or large deformation, mechanics applicable to deformable solid bodies are used in this work. Additionally, the considerations that are made to utilise a thermo-mechanical framework are presented. The common volumetric energy potentials are also presented.

B.2 Prerequisites on classical nonlinear solid mechanics

This section will go along the course of Chapter 2 and complete it with more general notions or classical nonlinear solid mechanics. This section strongly relies on the work of [10, 195].

B.2.1 Kinematics

Chapter 2 introduced the required kinematics notion to apprehend the ALE formulation. In particular, the quantities associated to the natural mapping φ are those of the classical Lagrangian formulation. More generally, when the mappings Ψ and Φ are both equal to the identity mapping, the formulation and all the quantities boil down to the Total Lagrangian Formulation. In that context, φ and its associated quantity, such as the natural velocity \mathbf{v} and the natural Jacobian J , will be referred to just as *velocity* and *jacobian* in this section. As Equation (2.2a) presents the velocity \mathbf{v} , the displacement vector \mathbf{u} can also be defined as

$$\mathbf{u}(\mathbf{X}, t) = \varphi(\mathbf{X}, t) - \mathbf{X}. \quad (\text{B.1})$$

In the context of large deformations, the displacement field is important quantity that characterises the change of configuration of the body. As the distance between two particles evolves throughout the deformation, that relative spatial position after deformation can be expressed in terms of their initial material position. By noting that $\mathbf{x} + d\mathbf{x} = \varphi(\mathbf{X} + d\mathbf{X}, t)$ and for a fixed t , a Taylor's series development of $\varphi(\mathbf{X} + d\mathbf{X}, t)$ in the neighbourhood of \mathbf{X} for a small arbitrary spatial vector $d\mathbf{X}$ yields

$$\mathbf{u}(\mathbf{X} + d\mathbf{X}, t) \approx \varphi(\mathbf{X}, t) + \frac{\partial \varphi(\mathbf{X}, t)}{\partial \mathbf{X}} d\mathbf{X} - (\mathbf{X} + d\mathbf{X}), \quad (\text{B.2})$$

and it is clear that

$$d\mathbf{x} = d\mathbf{X} + \mathbf{u}(\mathbf{X} + d\mathbf{X}, t) - \mathbf{u}(\mathbf{X}, t). \quad (\text{B.3})$$

Combining Equations (B.1) and (B.2) in (B.3), it is then possible to construct the deformation gradient \mathbf{F} , presented in Equation (2.2b), as

$$d\mathbf{x} = \mathbf{F} d\mathbf{X}; \quad \mathbf{F} = \frac{\partial \varphi(\mathbf{X}, t)}{\partial \mathbf{X}} = \nabla_{\mathbf{x}} \varphi; \quad F_{iI} = \frac{\partial x_i}{\partial X_I}. \quad (\text{B.4a,b,c})$$

In Equation (B.4c), the indexing suggests that \mathbf{F} is a two-point tensor, that is to say it transforms quantities from the material configuration to the spatial configuration. The time derivative of the material and spatial deformation gradients are already used in the ALE formulation as seen in Section 2.4. When considering the mapping φ , it can be stated using the Schwarz theorem as

$$\dot{\mathbf{F}} = \frac{d}{dt} \left(\frac{\partial \varphi}{\partial \mathbf{X}} \right) = \frac{\partial}{\partial \mathbf{X}} \left(\frac{\partial \varphi}{\partial t} \right) = \nabla_{\mathbf{x}} \mathbf{v} = \mathbf{l} = \dot{\mathbf{F}} \mathbf{F}^{-1}. \quad (\text{B.5})$$

where \mathbf{l} is the velocity gradient. The expression of the cofactor and the Jacobian are recalled as

$$J = \det \mathbf{F} = \frac{1}{6} (\mathbf{F} \times \mathbf{F}) : \mathbf{F} > 0; \quad \mathbf{H} = J \mathbf{F}^{-T} = \frac{1}{2} (\mathbf{F} \times \mathbf{F}) \quad (\text{B.6a,b})$$

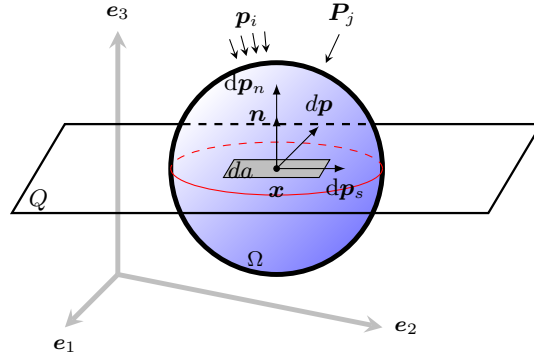


FIGURE B.1: Local traction and normal vector.

The non-objective right Cauchy-Green tensor \mathbf{C} characterises the deformation before any rotation. \mathbf{C} and the left Cauchy-Green or Finger tensor \mathbf{b} are defined as

$$\mathbf{C} = \mathbf{F}^T \mathbf{F} = \mathbf{C}^T; \quad \mathbf{b} = \mathbf{F} \mathbf{F}^T = \mathbf{b}^T. \quad (\text{B.7a,b})$$

Since $J > 0$, \mathbf{F} can be written as a multiplicative polar decomposition of a symmetric tensor and an orthogonal tensor, and as a consequence, spectral decomposition of \mathbf{F} and \mathbf{C} are related as

$$\mathbf{F} = \sum_{i=1}^3 \lambda_i \mathbf{N}_i \otimes \mathbf{N}_i; \quad \mathbf{C} = \sum_{i=1}^3 \lambda_i^2 \mathbf{N}_i \otimes \mathbf{N}_i, \quad (\text{B.8a,b})$$

for eigenvalues λ_i and eigenvectors \mathbf{N}_i .

B.2.2 Stress and equilibrium

In accordance with the infinitesimal strain theory, stress and equilibrium concepts can now be introduced on a given configuration. For a more detailed and exhaustive presentation, refer to the work of [195]. Consider a general body Ω illustrated in Figure B.1, upon which forces \mathbf{p}_i are acting, and which is cut by a fictitious plane Q . As it is the case in Fluid Mechanics, fluxes play an important role in the discretisation of conservation laws in the Finite Volume Method. Consider therefore an infinitesimal element of area da normal to the vector \mathbf{n} in the neighbourhood of a point \mathbf{x} , and undergoing a resultant force $d\mathbf{p}$. The traction vector \mathbf{t} corresponding to the normal \mathbf{n} at \mathbf{x} is defined as follows

$$\mathbf{t}(\mathbf{n}) = \lim_{da \rightarrow 0} \frac{d\mathbf{p}}{da} = \boldsymbol{\sigma} \mathbf{n}; \quad t_i = \sigma_{ij} n_j, \quad (\text{B.9a,b})$$

and constitutes the basis for assuming the existence of the symmetric second-order tensor $\boldsymbol{\sigma}$, known as the Cauchy stress tensor. The traction \mathbf{t} verifies the principle of action and reaction (Newton's third law)

$$\mathbf{t}(\mathbf{n}) = -\mathbf{t}(-\mathbf{n}). \quad (\text{B.10})$$

By studying the expression of the \mathbf{t} , the columns of $\boldsymbol{\sigma}$ can be viewed as traction vectors acting on the normal vectors \mathbf{e}_i

$$\boldsymbol{\sigma} = \sum_{i=1}^3 \mathbf{t}(\mathbf{e}_i) \otimes \mathbf{e}_i; \quad \mathbf{t}(\mathbf{n}) = \sum_{i,j=1}^3 \sigma_{ij} (\mathbf{e}_j \cdot \mathbf{n}) \mathbf{e}_i = \sum_{i,j=1}^3 \sigma_{ij} (\mathbf{e}_i \otimes \mathbf{e}_j) \mathbf{n}, \quad (\text{B.11a,b})$$

and it transpires that it will have three real eigenvalues, or principal stresses. Note that the Cauchy stress is generally expressed for the (deformed) current area. It is of common

practice in solid mechanics to re-express the previous relations with an alternative stress representation, the Kirchhoff stresses, that are acting upon the undeformed body. Thus, it is possible to introduce a traction force \mathbf{t}_0 per unit undeformed area such that

$$d\mathbf{p} = \mathbf{t}da = \boldsymbol{\sigma}\mathbf{n}da = \mathbf{t}_0dA. \quad (\text{B.12})$$

Using the Nanson's rule on the area vector $\mathbf{n}da$, the above equation yields

$$\mathbf{t}_0 = J\boldsymbol{\sigma}\mathbf{F}^{-T}\mathbf{N} = \boldsymbol{\sigma}\mathbf{H} = \mathbf{P}\mathbf{N} = \mathbf{F}\mathbf{S}\mathbf{N}, \quad (\text{B.13})$$

which serves as the basis to define the first and second Piola Kirchhoff stress tensor, \mathbf{P} and \mathbf{S} , acting on a material normal \mathbf{N} in the undeformed configuration

$$\mathbf{P} = J\boldsymbol{\sigma}\mathbf{F}^{-T}; \quad \mathbf{S} = \mathbf{F}^{-1}\mathbf{P} = J\mathbf{F}^{-1}\boldsymbol{\sigma}\mathbf{F}^{-T}. \quad (\text{B.14a,b})$$

Note that \mathbf{P} is asymmetric.

B.3 Large Strain thermo-elasticity

In the following, a summary of the considerations important for this work, in the context of large strain thermo-elasticity is made. Then, the potentials and other thermal-related quantities that are required in the derivation of the ALE formulation and for the algorithm are presented. For a more comprehensive development, it is recommended to refer to [10, 21–23, 195]. Finally, a collection of some common volumetric potential is shown.

B.3.1 Calorimetry relationships

The derivation of Equation (4.12) with respect to the temperature between θ and θ_R is done in the following [22]. First of all, notice that the following fraction can be expressed in an alternative way by using the chain rule as

$$\frac{c_v}{\theta(\mathbf{X}, t)} = \frac{1}{\theta(\mathbf{X}, t)} \frac{\partial \tilde{\mathcal{E}}(\boldsymbol{\chi}_{\theta\alpha})}{\partial \theta(\mathbf{X}, t)} = \frac{1}{\theta(\mathbf{X}, t)} \frac{\partial \tilde{\mathcal{E}}(\boldsymbol{\chi}_{\eta\alpha})}{\partial \eta(\mathbf{X}, t)} \frac{\partial \tilde{\eta}(\boldsymbol{\chi}_{\theta\alpha})}{\partial \theta(\mathbf{X}, t)} = \frac{\partial \tilde{\eta}(\boldsymbol{\chi}_{\theta\alpha})}{\partial \theta(\mathbf{X}, t)}. \quad (\text{B.15})$$

Let T be a variable for integration of the entropy in terms of the temperature as

$$\int_{\theta_R}^{\theta} \frac{\partial \tilde{\eta}(\boldsymbol{\chi}_{T\alpha})}{\partial T} = \tilde{\eta}(\boldsymbol{\chi}_{\theta\alpha}) - \tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}); \quad \int_{\theta_R}^{\theta} \frac{c_v}{T} = c_v \ln \left(\frac{\theta}{\theta_R} \right). \quad (\text{B.16a,b})$$

with $\tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) = \tilde{\eta}(\boldsymbol{\chi}, \theta = \theta_R, \boldsymbol{\alpha})$. It is then possible, using the notation defined in Chapter 4 to bring out two relationships involving the entropy and the temperature, stated as

$$\tilde{\eta}(\boldsymbol{\chi}_{\theta\alpha}) = \tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) + c_v \ln \left(\frac{\theta}{\theta_R} \right); \quad \Theta(\boldsymbol{\chi}_{\eta\alpha}) = \theta_R e^{\frac{\tilde{\eta}(\boldsymbol{\chi}_{\theta\alpha}) - \tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha})}{c_v}}. \quad (\text{B.17a,b})$$

Moreover, it is possible to express the temperature, using the value Θ_0 in the absence of entropy production, as

$$\Theta(\boldsymbol{\chi}_{\eta\alpha}) = \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) e^{\frac{\tilde{\eta}}{c_v}}; \quad \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) = \theta_R e^{-\frac{\tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha})}{c_v}} = \Theta(\boldsymbol{\chi}, \eta=0, \boldsymbol{\alpha}) \quad (\text{B.18a,b})$$

Then consider the integration of the temperature definition (4.11) between the absence and the production of entropy:

$$\int_0^{\eta} \frac{\partial \mathcal{E}(\boldsymbol{\chi}_{n\alpha})}{\partial n(\mathbf{X}, t)} = \mathcal{E}(\boldsymbol{\chi}_{\eta\alpha}) - \mathcal{E}_0(\boldsymbol{\chi}, \boldsymbol{\alpha}); \quad \mathcal{E}_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) = \mathcal{E}(\boldsymbol{\chi}, \eta=0, \boldsymbol{\alpha}). \quad (\text{B.19a,b})$$

Using the above results, the energy potential can be written as

$$\mathcal{E}(\boldsymbol{\chi}_{\eta\alpha}) = \mathcal{E}_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) + c_v \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) \left[\exp^{\frac{\tilde{\eta}(\boldsymbol{\chi}, \theta, \boldsymbol{\alpha})}{c_v}} - 1 \right]. \quad (\text{B.20})$$

Now the integration of (4.12) for a temperature that varies between $\Theta_0(\boldsymbol{\chi})$ and θ_R gives

$$\int_{\Theta_0(\boldsymbol{\chi})}^{\theta_R} c_v d\theta = c_v (\theta_R - \Theta_0(\boldsymbol{\chi})), \quad (\text{B.21})$$

and

$$\int_{\Theta_0(\boldsymbol{\chi})}^{\theta_R} \frac{\partial \tilde{\mathcal{E}}(\boldsymbol{\chi}, \theta, \boldsymbol{\alpha})}{\partial \theta(\boldsymbol{\chi}, t)} d\theta = \tilde{\mathcal{E}}(\boldsymbol{\chi}, \theta = \theta_R, \boldsymbol{\alpha}) - \tilde{\mathcal{E}}(\boldsymbol{\chi}, \theta = \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha}), \boldsymbol{\alpha}), \quad (\text{B.22})$$

which eventually gives

$$\tilde{\mathcal{E}}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) = \mathcal{E}_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) + c_v (\theta_R - \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha})) \quad (\text{B.23a})$$

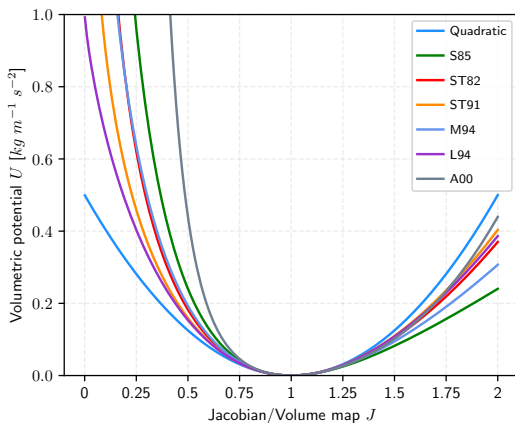
$$\tilde{\mathcal{E}}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) = \tilde{\mathcal{E}}(\boldsymbol{\chi}, \theta = \theta_R, \boldsymbol{\alpha}) = \mathcal{E}(\boldsymbol{\chi}, \tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}), \boldsymbol{\alpha}). \quad (\text{B.23b})$$

Combining Equations (B.20) and (B.23) and (B.18b) yields the following relationship

$$\begin{aligned} \mathcal{E}(\boldsymbol{\chi}_{\eta\alpha}) &= \tilde{\mathcal{E}}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) - c_v (\theta_R - \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha})) + c_v \Theta_0(\boldsymbol{\chi}, \boldsymbol{\alpha}) \left[\exp^{\frac{\tilde{\eta}(\boldsymbol{\chi}, \theta, \boldsymbol{\alpha})}{c_v}} - 1 \right] \\ &= \tilde{\mathcal{E}}_R(\boldsymbol{\chi}, \boldsymbol{\alpha}) + c_v \theta_R \left(\exp^{\frac{\tilde{\eta}(\boldsymbol{\chi}, \theta, \boldsymbol{\alpha}) - \tilde{\eta}_R(\boldsymbol{\chi}, \boldsymbol{\alpha})}{c_v}} - 1 \right). \end{aligned} \quad (\text{B.24})$$

B.3.2 Volumetric energies

An array of alternative volumetric potential to account for stress caused by volume distortion, originally presented in Rossi et al. [317], is summarised in Table B.1, and depicted in Figure B.2.



Quadratic	$U = \frac{\kappa}{2} [J - 1]^2$
S85 [239]	$U = \frac{\kappa}{2} \ln^2 J$
ST82 [318]	$U = \frac{\kappa}{4} [(J - 1)^2 + \ln^2 J]$
ST91 [319]	$U = \frac{\kappa}{4} [J^2 - 1 - 2 \ln J]$
M94 [320]	$U = \kappa [J - \ln J - 1]$
L94 [321]	$U = \kappa [J \ln J - J + 1]$
A00 Ansys [44]	$U = \frac{\kappa}{32} [J^2 - J^{-2}]^2$

FIGURE B.2: Comparison of volumetric potentials using models presented in Table B.1.

TABLE B.1: Volumetric energy potentials.

This is justified by the isochoric-volumetric decomposition of the energy potential, studied in [236, 237]. Volumetric potentials must verify certain requirements such as the incompressibility condition $U(1) = 0$.

B.3.3 On the Lie time derivative

The Lie time derivative of a tensor \mathbf{g} over a mapping φ is defined as the push-forward⁵⁸ of the time derivative of the pull-back of \mathbf{g} [217] and is noted as

$$\mathcal{L}_\varphi(\mathbf{g}) = \varphi_* \left[\frac{d}{dt} (\varphi_*^{-1} \mathbf{g}) \right]. \quad (\text{B.25})$$

The Lie derivative can easily be expressed for covariant and contravariant quantities [322]. For a covariant tensor such as the right Cauchy-Green tensor \mathbf{C} , and for a contravariant tensor such as the left Cauchy-Green tensor, the Lie derivative⁵⁹ with regards to the velocity vector \mathbf{v} can be expressed as

$$\mathcal{L}_\mathbf{v}(\mathbf{C}) = \frac{\partial \mathbf{C}}{\partial t} \Big|_{\mathbf{X}} + (\nabla_{\mathbf{X}} \mathbf{v})^T \mathbf{C} + \mathbf{C} \nabla_{\mathbf{X}}, \quad (\text{B.26a})$$

$$\mathcal{L}_\mathbf{v}(\mathbf{b}) = \frac{\partial \mathbf{b}}{\partial t} \Big|_{\mathbf{X}} - (\nabla_{\mathbf{X}} \mathbf{v}) \mathbf{b} - \mathbf{b} (\nabla_{\mathbf{X}} \mathbf{v})^T \quad (\text{B.26b})$$

In particular, the Lie derivative (B.26b) of the contravariant left Cauchy-Green tensor with respect to \mathbf{v} is used to get an evolution equation on the internal elastic stresses, as shown in Section 3.5.

B.4 Tools for the Arbitrary Lagrangian Eulerian approach

This section is dedicated to further discuss the ALE methodology in the context of solid mechanics. Section B.4.1 shows a development to derive the ALE wave speeds to complement the one presented in Chapter 5. Then, Section B.5 shows how to degenerate ALE equations to their TLF equivalent equations.

B.4.1 Wave speeds connections

In the homogenous ALE system

$$\frac{\partial \mathbf{U}}{\partial t} \Big|_{\mathbf{x}} + \nabla_{\mathbf{x}} \cdot [\mathcal{F}] = \mathbf{0}; \quad \mathbf{U} = J_\Psi \mathbf{U}_R; \quad \mathbf{S} = J_\Psi \mathbf{S}; \quad \mathcal{F} = [\mathcal{F}_R - (\mathbf{U}_R \otimes \mathbf{W})] \mathbf{H}_\Psi, \quad (\text{B.27a,b,c,d})$$

normal fluxes can be expanded as

$$\mathcal{F}_{N_{\mathbf{X}}} = \mathcal{F} N_{\mathbf{X}} = \mathcal{F}_R \mathbf{H}_\Psi N_{\mathbf{X}} - \mathbf{U}_R (\mathbf{W} \cdot \mathbf{H}_\Psi N_{\mathbf{X}}). \quad (\text{B.28})$$

The directional derivative \mathcal{F} , in the direction of an eigenvector \mathbf{U}_α^R , is linked to that of \mathcal{F}_R , using Equation (B.28) and change of variables $J_\Psi = \partial \mathbf{U} / \partial \mathbf{U}_R$ as

$$\begin{aligned} \text{D}\mathcal{F}[\mathbf{U}_\alpha^R] &= \left(\frac{\partial \mathcal{F}}{\partial \mathbf{U}_R} \frac{\partial \mathbf{U}_R}{\partial \mathbf{U}} \right) \mathbf{U}_\alpha^R = \left(\Lambda_{\mathbf{H}_\Psi} \frac{\partial \mathcal{F}_R}{\partial \mathbf{U}_R} - \mathbf{W} \cdot \mathbf{H}_\Psi N_{\mathbf{X}} \right) \mathbf{U}_\alpha^R \\ &= \Lambda_{\mathbf{H}_\Psi} (\text{D}\mathcal{F}_R[\mathbf{U}_\alpha^R] - (\mathbf{W} \cdot N_{\mathbf{X}}) \mathbf{U}_\alpha^R). \end{aligned} \quad (\text{B.29a})$$

with $\Lambda_{\mathbf{H}_\Psi}$ defined in Equation (3.28). Using the ALE characteristic equation (3.32), it yields

$$\left(\frac{c^\alpha J_\Psi}{\Lambda_{\mathbf{H}_\Psi}} + \mathbf{W} \cdot N_{\mathbf{X}} \right) \mathbf{U}_R^\alpha = \text{D}\mathcal{F}_R[\mathbf{U}_R^\alpha]. \quad (\text{B.30})$$

⁵⁸The push forward of that mapping is denoted as φ_* and the pull-forward is denoted as φ_*^{-1} .

⁵⁹In the expressions of the Lie time derivative (B.26), the first line (B.26a) can also be denoted the Cottler-Rivlin rate, and the second line (B.26b) can be denoted as the upper Oldroyd rate.

By identification, the parenthesis in the left hand side of Equation (B.30) corresponds wave speeds c_α of a TLF eigenproblem and the ALE wave speed expression (3.33) is recovered. The ALE problem is therefore also hyperbolic. This result is fortunately compatible with degenerating a set of ALE equations into a set of TLF equations, as addressed in Section B.5.

B.5 On the degeneration to Lagrangian and Eulerian formulations

It is interesting to notice that ALE equations can degenerate to Lagrangian and Eulerian forms. By considering

$$\mathbf{W} = \mathbf{0}; \quad \mathbf{F}_\Psi = \mathbf{I}. \quad (\text{B.31a,b})$$

The scaled variables and intermediate quantities are equal to

$$\chi = \mathbf{X}; \quad \hat{\mathbf{v}} = \mathbf{v}; \quad \mathbf{F}_\Phi = \mathbf{F}. \quad (\text{B.32a,b,c})$$

Doing this, enables the ALE system (2.75) to be rewritten as

$$\left. \frac{\partial \mathbf{U}}{\partial t} \right|_{\mathbf{X}} + \sum_{i=1}^3 \frac{\partial \mathcal{F}_I}{\partial X_I} = \mathcal{S}, \quad (\text{B.33})$$

with the following components

$$\mathbf{U} = \begin{bmatrix} \rho_R \\ \mathbf{p} \\ \mathbf{F} \\ E \end{bmatrix}; \quad \mathcal{F}_I = - \begin{bmatrix} 0 \\ \mathbf{P} \mathbf{E}_I \\ \mathbf{v} \otimes \mathbf{E}_I \\ \mathbf{P}^T \mathbf{v} \cdot \mathbf{E}_I \end{bmatrix}; \quad \mathcal{S} = \begin{bmatrix} 0 \\ \mathbf{f}_R \\ \mathbf{0} \\ \mathbf{f}_R \cdot \mathbf{v} \end{bmatrix}. \quad (\text{B.34a,b,c})$$

This system, labelled in this work as the equivalent TLF mixed formulation. Furthermore, the respective Eulerian equivalent system can also be obtained by enforcing the referential domain to coincide with the spatial domain by setting

$$\hat{\mathbf{v}} = \mathbf{0}; \quad \mathbf{F}_\Phi = \mathbf{I}. \quad (\text{B.35a,b})$$

As a consequence,

$$\chi = \mathbf{x}; \quad \mathbf{W} = -\mathbf{F}^{-1} \mathbf{v}; \quad \mathbf{F}_\Psi = \mathbf{F}^{-1}, \quad (\text{B.36a,b,c})$$

and the system (2.75) can be rewritten as

$$\left. \frac{\partial \mathbf{U}}{\partial t} \right|_{\mathbf{x}} + \sum_{i=1}^3 \frac{\partial \mathcal{F}_I}{\partial x_I} = \mathcal{S}, \quad (\text{B.37})$$

with the following components

$$\mathbf{U} = \begin{bmatrix} H^{-1} \rho_R \\ J^{-1} \mathbf{p} \\ \mathbf{F}^{-1} \\ J^{-1} E \end{bmatrix}; \quad \mathcal{F}_I = - \begin{bmatrix} J^{-1} \rho_R \mathbf{v} \cdot \mathbf{E}_I \\ [\boldsymbol{\sigma} - J^{-1} \mathbf{v} \otimes \mathbf{v}] \mathbf{E}_I \\ (\mathbf{F}^{-1} \mathbf{v}) \otimes \mathbf{E}_I \\ [\boldsymbol{\sigma}^T \mathbf{v} - (J^{-1} E \mathbf{v})] \cdot \mathbf{E}_I \end{bmatrix}; \quad \mathcal{S} = \begin{bmatrix} 0 \\ \mathbf{f}_R \\ \mathbf{0} \\ \mathbf{f}_R \cdot \mathbf{v} \end{bmatrix}, \quad (\text{B.38a,b,c})$$

where $\boldsymbol{\sigma} = \mathbf{P} \mathbf{H}^{-1}$ is the Cauchy stress tensor.

APPENDIX C

SECOND LAW OF THERMODYNAMICS

*“Just as the eye was made to see colours, and the ear to hear sounds,
so the human mind was made to understand.”*

— JOHANNES KEPLER

In order to provide a proper physical meaning to the conjugate fields of the first-order system of primary variables $\{\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, \eta\}$, the *Ballistic* energy (also known as the Lyapunov function of the thermo-mechanical process) is introduced. The Ballistic energy per unit material volume, described by Ericksen [257], is expressed as

$$\begin{aligned} \mathcal{B}(\mathbf{X}, t) &= \tilde{\mathcal{B}}(\mathbf{p}, \boldsymbol{\chi}_\eta) = K(\mathbf{p}) + \mathcal{E}(\boldsymbol{\chi}_\eta) - \theta_R \eta \\ &= K(\mathbf{p}) + \Psi(\boldsymbol{\chi}_\eta) + \vartheta \eta, \end{aligned} \quad (\text{C.1})$$

with the difference of temperature $\vartheta = \theta - \theta_R$, where \mathcal{B} and $\tilde{\mathcal{B}}$ are alternative functional of the same magnitude, and where Ψ is the Helmholtz free energy expressed as

$$\Psi(\boldsymbol{\chi}_\eta) = \mathcal{E}(\boldsymbol{\chi}_\eta) - \Theta(\boldsymbol{\chi}_\eta)\eta, \quad (\text{C.2})$$

In the rest of this development, it is convenient to define the Ballistic energy \mathcal{B}_χ per unit of referential volume as

$$\begin{aligned} \mathcal{B}_\chi(\boldsymbol{\chi}, t) &= \hat{\mathcal{B}}_\chi(\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, \eta_\chi) = J_\Psi \tilde{\mathcal{B}}(\mathbf{p}, \boldsymbol{\chi}_{\eta\alpha}) \\ &= \frac{1}{2\rho_\chi} \mathbf{p}_\chi \cdot \mathbf{p}_\chi + J_\Psi \mathcal{E}(\mathbf{F}(\mathbf{F}_\Phi, \mathbf{F}_\Psi), J_\Psi^{-1} \eta_\chi) - \theta_R \eta_\chi, \end{aligned} \quad (\text{C.3})$$

with $\rho_\chi = J_\Psi \rho_R$ and $\mathcal{B}_\chi(\boldsymbol{\chi}, t)$ and $\hat{\mathcal{B}}_\chi(\mathbf{p}_\chi, \mathbf{F}_\Phi, \mathbf{F}_\Psi, J_\Psi, \eta_\chi)$ represent alternative functional representations of the same magnitude. In the above equation, the first term of the right hand side represents the kinetic energy, the second term represents internal energy and the third term represents the heat component expressed in terms of the scaled entropy $\eta_\chi = J_\Psi \eta$. It is now appropriate to define energy conjugates for the two deformation measures, the spatial and material deformation gradients $\{\mathbf{F}_\Phi, \mathbf{F}_\Psi\}$, that are defined as

$$\mathbf{P}_\Phi = \frac{\partial \mathcal{E}(\mathbf{F}(\mathbf{F}_\Phi, \mathbf{F}_\Psi), J_\Psi^{-1} \eta_\chi)}{\partial \mathbf{F}_\Phi}; \quad \mathbf{P}_\Psi = \frac{\partial \mathcal{E}(\mathbf{F}(\mathbf{F}_\Phi, \mathbf{F}_\Psi), J_\Psi^{-1} \eta_\chi)}{\partial \mathbf{F}_\Psi}. \quad (\text{C.4a,b})$$

Moreover, the definition of the temperature in Equation (4.11) can also be seen as an energy conjugacy relationship between the entropy η and the temperature θ , stated as

$$\theta = \frac{\partial \mathcal{E}(\mathbf{F}(\mathbf{F}_\Phi, \mathbf{F}_\Psi), J_\Psi^{-1} \eta_\chi)}{\partial \eta}. \quad (\text{C.5})$$

Following Hamilton's principle, it is now possible to obtain the set of associated work conjugates \mathcal{V}_χ as [2, 23]

$$\mathcal{V}_\chi = \frac{\partial \hat{\mathcal{B}}_\chi}{\partial \boldsymbol{\mathcal{U}}_\chi} = \begin{bmatrix} \frac{\partial \hat{\mathcal{B}}_\chi}{\partial \mathbf{p}_\chi} \\ \frac{\partial \hat{\mathcal{B}}_\chi}{\partial \mathbf{F}_\Phi} \\ \frac{\partial \hat{\mathcal{B}}_\chi}{\partial \mathbf{F}_\Psi} \\ \frac{\partial \hat{\mathcal{B}}_\chi}{\partial J_\Psi} \\ \frac{\partial \hat{\mathcal{B}}_\chi}{\partial \eta_\chi} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{P}_\Phi \\ \mathbf{P}_\Psi \\ -\mathcal{L} \\ \vartheta \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \mathbf{P} \mathbf{H}_\Psi \\ -\mathbf{F}^T \mathbf{P} \mathbf{H}_\Psi \\ -\mathcal{L} \\ \vartheta \end{bmatrix}. \quad (\text{C.6})$$

Here, the functional

$$\mathcal{L} = K - (\mathcal{E}(\mathbf{F}, \eta) - \theta \eta) \quad (\text{C.7})$$

denotes the (thermal-mechanical based) Lagrangian function, defined as a combination of the kinetic energy $K = \frac{1}{2} \rho_R (\mathbf{v} \cdot \mathbf{v})$, the internal energy \mathcal{E} and the heat dissipative energy $\theta \eta$.

It is instructive to revisit the global version of the second law of thermodynamics when written in terms of the *Ballistic* energy density $\hat{\mathcal{B}}$. The time derivative of the *Ballistic* energy is obtained via the chain rule as

$$\begin{aligned}
& \left. \frac{\partial}{\partial t} \right|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathcal{B}_{\mathbf{x}} dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}(\mathbf{p}_{\mathbf{x}}, \mathbf{F}_{\Phi}, \mathbf{F}_{\Psi}, J_{\Psi}, \eta_{\mathbf{x}})}{\partial t} \right|_{\mathbf{x}} dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left(\left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}}{\partial \mathbf{p}_{\mathbf{x}}} \cdot \frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} + \left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}}{\partial \mathbf{F}_{\Phi}} : \frac{\partial \mathbf{F}_{\Phi}}{\partial t} \right|_{\mathbf{x}} + \left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}}{\partial \mathbf{F}_{\Psi}} : \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} + \left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}}{\partial J_{\Psi}} \frac{\partial J_{\Psi}}{\partial t} \right|_{\mathbf{x}} + \left. \frac{\partial \hat{\mathcal{B}}_{\mathbf{x}}}{\partial \eta_{\mathbf{x}}} \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right) dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left(\mathbf{v} \cdot \left. \frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} + \mathbf{P}_{\Phi} : \left. \frac{\partial \mathbf{F}_{\Phi}}{\partial t} \right|_{\mathbf{x}} + \mathbf{P}_{\Psi} : \left. \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} - \mathcal{L} \left. \frac{\partial J_{\Psi}}{\partial t} \right|_{\mathbf{x}} + \vartheta \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right) dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left(\mathbf{v} \cdot \left. \frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} + (\mathbf{P}\mathbf{H}_{\Psi}) : \left. \frac{\partial \mathbf{F}_{\Phi}}{\partial t} \right|_{\mathbf{x}} - (\mathbf{F}^T \mathbf{P}\mathbf{H}_{\Psi}) : \left. \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} - \mathcal{L}\mathbf{H}_{\Psi} : \left. \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} + \vartheta \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right) dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left(\mathbf{v} \cdot \left. \frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} + (\mathbf{P}\mathbf{H}_{\Psi}) : \left. \frac{\partial \mathbf{F}_{\Phi}}{\partial t} \right|_{\mathbf{x}} + (\boldsymbol{\Sigma}\mathbf{H}_{\Psi}) : \left. \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} + \vartheta \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right) dV_{\mathbf{x}}, \quad (\text{C.8})
\end{aligned}$$

where, Equations (C.4–C.5) have been substituted in the third line of (C.8), respectively. The pairs

$$\left\{ \mathbf{v}, \left. \frac{\partial \mathbf{p}_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right\}, \left\{ \mathbf{P}\mathbf{H}_{\Psi}, \left. \frac{\partial \mathbf{F}_{\Phi}}{\partial t} \right|_{\mathbf{x}} \right\}, \left\{ \boldsymbol{\Sigma}\mathbf{H}_{\Psi}, \left. \frac{\partial \mathbf{F}_{\Psi}}{\partial t} \right|_{\mathbf{x}} \right\}, \left\{ \vartheta, \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} \right\} \quad (\text{C.9})$$

are said to be dual or work conjugate with respect to the referential volume in the sense that their inner product yields work rate per unit of referential volume. For instance, the energy conjugate field to the time rate of the material-based deformation gradient is the classical (material-based) Eshelby stress tensor defined as $\boldsymbol{\Sigma} = -(\mathbf{F}^T \mathbf{P} + \mathcal{L}\mathbf{I})$. Consequently, we can substitute the linear momentum balance law (2.56) and the geometric conservation equations (2.58) and (2.61) into the fifth line of (C.8), and after some algebraic manipulations, to give

$$\begin{aligned}
& \left. \frac{\partial}{\partial t} \right|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathcal{B}_{\mathbf{x}} dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left[\text{DIV}_{\mathbf{x}} \left((\mathbf{P}\mathbf{H}_{\Psi})^T \mathbf{v} + 2K\mathbf{H}_{\Psi}^T \mathbf{W} \right) + \mathbf{v} \cdot \mathbf{f} - \nabla_{\mathbf{x}} K \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}) \right] dV_{\mathbf{x}} \\
&\quad + \int_{\Omega_{\mathbf{x}}} \left[(\mathbf{P}\mathbf{H}_{\Psi}) : \nabla_{\mathbf{x}} \mathbf{v} + (\mathbf{P}\mathbf{H}_{\Psi}) : \nabla_{\mathbf{x}} \hat{\mathbf{v}} + (\boldsymbol{\Sigma}\mathbf{H}_{\Psi}) : \nabla_{\mathbf{x}} \mathbf{W} \right] dV_{\mathbf{x}} + \int_{\Omega_{\mathbf{x}}} \vartheta \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} dV_{\mathbf{x}} \\
&= \int_{\Omega_{\mathbf{x}}} \left[\text{DIV}_{\mathbf{x}} \left((\mathbf{P}\mathbf{H}_{\Psi})^T \mathbf{v} + 2K\mathbf{H}_{\Psi}^T \mathbf{W} \right) + \mathbf{v} \cdot \mathbf{f} - \nabla_{\mathbf{x}} K \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}) \right] dV_{\mathbf{x}} \\
&\quad + \int_{\Omega_{\mathbf{x}}} \left[\nabla_{\mathbf{x}} \mathcal{E} \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}) - \mathcal{L}\mathbf{H}_{\Psi} : \nabla_{\mathbf{x}} \mathbf{W} \right] dV_{\mathbf{x}} + \int_{\Omega_{\mathbf{x}}} \vartheta \left. \frac{\partial \eta_{\mathbf{x}}}{\partial t} \right|_{\mathbf{x}} dV_{\mathbf{x}}, \quad (\text{C.10})
\end{aligned}$$

It is worth pointing out that the term on the second line of the equation above can only be obtained by strongly enforce the curl-free conditions. Further expanding the term

$$\mathcal{L}\mathbf{H}_{\Psi} : \nabla_{\mathbf{x}} \mathbf{W} = \text{DIV}_{\mathbf{x}} (\mathcal{L}\mathbf{H}_{\Psi}^T \mathbf{W}) - (\nabla_{\mathbf{x}} \mathcal{L}) \cdot (\mathbf{H}_{\Psi}^T \mathbf{W}), \quad (\text{C.11})$$

and some term arrangement, Equation (C.10) reduces to

$$\begin{aligned} & \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathcal{B}_{\mathbf{x}} dV_{\mathbf{x}} \\ &= \int_{\Omega_{\mathbf{x}}} \left[\text{DIV}_{\mathbf{x}} \left((\mathbf{P}\mathbf{H}_{\Psi})^T \mathbf{v} + (2K - \mathcal{L}) \mathbf{H}_{\Psi}^T \mathbf{W} \right) + \mathbf{v} \cdot \mathbf{f} + (\nabla_{\mathbf{x}} \theta) \cdot (\eta \mathbf{H}_{\Psi}^T \mathbf{W}) \right] dV_{\mathbf{x}} \\ & \quad + \int_{\Omega_{\mathbf{x}}} \vartheta \frac{\partial \eta_{\mathbf{x}}}{\partial t} \Big|_{\mathbf{x}} dV_{\mathbf{x}}. \end{aligned} \quad (\text{C.12})$$

Recalling the ALE entropy density rate in conservative form, stated in Equation (4.28), the heat component in Equation (C.10) is given as

$$\begin{aligned} \int_{\Omega_{\mathbf{x}}} \vartheta \frac{\partial \eta_{\mathbf{x}}}{\partial t} \Big|_{\mathbf{x}} dV_{\mathbf{x}} &= \int_{\Omega_{\mathbf{x}}} \vartheta \text{DIV}_{\mathbf{x}} (\eta \mathbf{H}_{\Psi}^T \mathbf{W}) dV_{\mathbf{x}} - \int_{\partial V_{\mathbf{x}}} \vartheta \left(\frac{q_B}{\theta_B} \right) dA_{\mathbf{x}} + \int_{\Omega_{\mathbf{x}}} \vartheta \frac{s_{\mathbf{x}}}{\theta} dV_{\mathbf{x}} \\ & \quad + \int_{\Omega_{\mathbf{x}}} \frac{\theta_R}{\theta^2} \mathbf{Q}_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} \theta dV_{\mathbf{x}}. \end{aligned} \quad (\text{C.13})$$

By noting that

$$\vartheta \text{DIV}_{\mathbf{x}} (\eta \mathbf{H}_{\Psi}^T \mathbf{W}) = \text{DIV}_{\mathbf{x}} (\vartheta \eta \mathbf{H}_{\Psi}^T \mathbf{W}) - \nabla_{\mathbf{x}} \theta (\eta \mathbf{H}_{\Psi}^T \mathbf{W}), \quad (\text{C.14})$$

the substitution of Equation (C.13) into the expression of the Ballistic rate in Equation (C.12), which becomes

$$\begin{aligned} \frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathcal{B}_{\mathbf{x}} dV_{\mathbf{x}} - \dot{\Pi}_{\mathbf{x}}^{\text{ext}} - \mathcal{Q}_{\mathbf{x}}^{\text{ext}} &= \int_{\Omega_{\mathbf{x}}} \nabla_{\mathbf{x}} \cdot (\mathcal{B} \mathbf{H}_{\Psi}^T \mathbf{W}) dV_{\mathbf{x}} \\ &= \int_{\Omega_{\mathbf{x}}} \frac{\theta_R}{\theta^2} \mathbf{Q}_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} \theta dV_{\mathbf{x}}. \end{aligned} \quad (\text{C.15})$$

The first term on the right hand side vanishes due to the enforcement of suitable boundary condition, that is

$$\mathbf{H}_{\Psi}^T \mathbf{W} \cdot \mathbf{N}_{\mathbf{x}} = \mathbf{W} \cdot (\mathbf{H}_{\Psi} \mathbf{N}_{\mathbf{x}}) = \mathbf{W} \cdot \mathbf{N}_{\mathbf{x}} = 0. \quad (\text{C.16})$$

In Equation (C.15), the term $\dot{\Pi}_{\mathbf{x}}^{\text{ext}}$ represents the mechanical power associated with external forces and is expressed as

$$\dot{\Pi}_{\mathbf{x}}^{\text{ext}} = \int_{\Omega_{\mathbf{x}}} \mathbf{v} \cdot \mathbf{f} dV_{\mathbf{x}} + \int_{\partial V_{\mathbf{x}}} \mathbf{v}_B \cdot \mathbf{t}_B dA_{\mathbf{x}}, \quad (\text{C.17})$$

with $\mathbf{t}_B = \mathbf{P}\mathbf{H}_{\Psi}\mathbf{N}_{\mathbf{x}}$, the term $\mathcal{Q}_{\mathbf{x}}^{\text{ext}}$ represents the heat source and heat flux added (removed) to (from) the system and is expressed

$$\mathcal{Q}_{\mathbf{x}}^{\text{ext}} = \int_{\Omega_{\mathbf{x}}} \frac{\vartheta}{\theta} r_{\mathbf{x}} dV_{\mathbf{x}} - \int_{\partial V_{\mathbf{x}}} \frac{\vartheta}{\theta} q_B dA_{\mathbf{x}}. \quad (\text{C.18})$$

Recalling the Fourier's law of heat conduction [21–23], the last term on the left hand side of Equation (C.15) is non-positive, which is demonstrated as below

$$\mathbf{Q}_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} \theta = -(\mathbf{K}_{\mathbf{x}} \nabla_{\mathbf{x}} \theta) \cdot \nabla_{\mathbf{x}} \theta = -\mathbf{K}_{\mathbf{x}} : (\nabla_{\mathbf{x}} \theta \otimes \nabla_{\mathbf{x}} \theta) \leq 0. \quad (\text{C.19})$$

With all this at hand, Equation (C.15) can finally be transformed into the following inequality

$$\frac{\partial}{\partial t} \Big|_{\mathbf{x}} \int_{\Omega_{\mathbf{x}}} \mathcal{B}_{\mathbf{x}} dV_{\mathbf{x}} - \dot{\Pi}_{\mathbf{x}}^{\text{ext}} - \mathcal{Q}_{\mathbf{x}}^{\text{ext}} \leq 0. \quad (\text{C.20})$$

This represents a valid expression for the second law of thermodynamics of a system. Satisfaction of inequality (C.20) is a necessary *ab initio* condition to ensure stability, otherwise referred to as the classical Coleman-Noll procedure. When introducing Riemann-based numerical dissipation to the finite volume spatial discretisation, this concept can be further exploited, which is beyond the scope of this thesis.

Remark C.0.1 *It is interesting to notice that when no thermal considerations is present in the formulation, the Ballistic energy degenerates to the Hamiltonian functional \mathcal{H} defined as*

$$\mathcal{H} = K + \mathcal{E}. \quad (\text{C.21})$$


Consequently, it is possible to bring out from this development the Hamiltonian equations of continuum mechanics.

Remark C.0.2 *It is shown in Expression (C.9) that the work conjugate relative to the energy variables is represented by the rate of scaled entropy density $\eta_{\mathbf{x}}$ and the difference of temperature ϑ . It is possible to consider the alternative work pair*

$$\left\{ \vartheta_{\mathbf{x}}, \left. \frac{\partial \eta}{\partial t} \right|_{\mathbf{x}} \right\}, \quad (\text{C.22})$$

that combines the scaled difference of temperature $\vartheta_{\mathbf{x}} = J_{\Psi}(\theta - \theta_R)$ and the rate of the entropy.

Remark C.0.3 *As said in the above development, it is possible to adapt the Coleman-Noll procedure to the spatially-discretised mixed system in order to prove the production of entropy, thence the presence of numerical dissipation. For the framework to preserve energy and momentum, it is necessary that the fully-discretised ALE mixed system is qualified as Hamiltonian [323]. This is achieved with the use of a symplectic time integrator [324, 325], which will not be discussed in this thesis.*

APPENDIX D  OPENFOAM COMPONENTS

“Ideas are cheap. It’s only what you do with them that counts.”

— ISAAC ASIMOV, *The Secrets of the Universe*, 1989

This appendix contains the list of the main files used in the *vcALEFoam* solver. They are organised in the following listing. They are compatible with the version 6 of *OpenFOAM.org*. That version only contains a few elements for solid dynamics. Moreover, it contains a mesh motion motion that implement several techniques for Fluid Dynamics. They can be used in the study of shock, or in turbomachinery. Further discussions upon the available techniques to move the mesh can be found in [49, 66, 292, 309].

```

1  /*-----*\
2  Description
3  Store the values of previous timestep for conserved variables
4  \*-----*/
5
6  x.oldTime();
7  lm.oldTime();
8  spatF.oldTime();
9  if (solveSpatH) { spatH.oldTime(); }
10 if (solveSpatJ) { spatJ.oldTime(); }
11
12 if (meshUpdate) {
13   if (solveMatF) { matF.oldTime(); }
14   //if (solveMatH) { matH.oldTime(); }
15   if (solveMatJ) { matJ.oldTime(); }
16   W.oldTime();
17   xw.oldTime();
18 }
19
20 if (isPlastic) {
21   CpInv.oldTime();
22   eps.oldTime();
23   epsDot.oldTime();
24   wp.oldTime();
25   coeffTau.oldTime();
26 }
27
28 if (Emethod) { E.oldTime(); } else { eta.oldTime(); }

```

Listing D.1: saveOldTime.H

```

1  /*-----*\
2  Description
3  Computes the gradients of W, v, p and T
4  \*-----*/
5
6  if (meshUpdate) wGrad = tensor::zero;
7  vGrad = tensor::zero;
8  pGrad = vector::zero;
9  if (isThermo) gradT = vector::zero;
10
11 forAll(edges, edge) { /* edge loop * * * * * */
12   const label& a = edges[edge][0];
13   const label& b = edges[edge][1];
14   const vector& Cab = Sf[edge];
15   if (meshUpdate) {
16     const tensor wGrad_ = 0.5 * (W[a] + W[b]) * Cab;
17     wGrad[a] += wGrad_;          wGrad[b] -= wGrad_; // W
18   }
19
20   const vector va = lm[a]/rhoR[a];
21   const vector vb = lm[b]/rhoR[b];
22   const tensor vGrad_ = 0.5 * (va + vb) * Cab;
23   vGrad[a] += vGrad_;          vGrad[b] -= vGrad_; // v
24
25   const vector pGrad_ = 0.5 * (p[a] + p[b]) * Cab;
26   pGrad[a] += pGrad_;          pGrad[b] -= pGrad_; // p
27
28   if (isThermo) {
29     const vector tGrad_ = 0.5 * (T[a] + T[b]) * Cab;
30     gradT[a] += tGrad_;        gradT[b] -= tGrad_; // T
31   }
32 } /* end of edge loop * * * * * */
33 forAll(mesh.boundary(), patch) { /* boundary loop * * * * * */
34   forAll(patches[patch], facei) { // face loop
35     const vector& Af = Sf_p.boundaryField()[patch][facei];
36     const label& face = patches[patch].start() + facei;
37     const labelList& myface = faces[face];
38     const label& nodeA = myface[0], nodeB = myface[1], nodeC = myface[2];
39
40     if (meshUpdate) {
41       wGrad[nodeA] += ((6.*W[nodeA]) + W[nodeB] + W[nodeC]) * Af / 24.;
42       wGrad[nodeB] += (W[nodeA] + (6.*W[nodeB]) + W[nodeC]) * Af / 24.;
43       wGrad[nodeC] += (W[nodeA] + W[nodeB] + (6.*W[nodeC])) * Af / 24.;
44     }
45
46     const vector va = lm[nodeA]/rhoR[nodeA];
47     const vector vb = lm[nodeB]/rhoR[nodeB];
48     const vector vc = lm[nodeC]/rhoR[nodeC];
49     vGrad[nodeA] += ((6.*va) + vb + vc) * Af / 24.;
50     vGrad[nodeB] += (va + (6.*vb) + vc) * Af / 24.;
51     vGrad[nodeC] += (va + vb + (6.*vc)) * Af / 24.;
52     pGrad[nodeA] += ((6.*p[nodeA]) + p[nodeB] + p[nodeC]) * Af / 24.;
53     pGrad[nodeB] += (p[nodeA] + (6.*p[nodeB]) + p[nodeC]) * Af / 24.;
54     pGrad[nodeC] += (p[nodeA] + p[nodeB] + (6.*p[nodeC])) * Af / 24.;
55

```



```

56     if (isThermo) {
57         gradT[nodeA] += ((6.*T[nodeA]) + T[nodeB] + T[nodeC]) * Af / 24.;
58         gradT[nodeB] += (T[nodeA] + (6.*T[nodeB]) + T[nodeC]) * Af / 24.;
59         gradT[nodeC] += (T[nodeA] + T[nodeB] + (6.*T[nodeC])) * Af / 24.;
60     }
61 } // end of face loop
62 } /* end of boundary loop * * * * * */
63 if (meshUpdate) wGrad = wGrad / Vol;
64 vGrad = vGrad / Vol;
65 pGrad = pGrad / Vol;
66 if (isThermo) gradT = gradT / Vol;

```

Listing D.2: gradients.H

```

1  /-----*\
2  Description
3  Compute interior contribution for the RHS
4  \-----*/
5
6  /* Reinitialise RHS * * * * * */
7  rhsLm = vector::zero;
8  rhsSpatF = tensor::zero;
9  if (solveSpatH) rhsSpatH = tensor::zero;
10 if (solveSpatJ) rhsSpatJ = 0;
11 if (meshUpdate) {
12     if (solveMatF) rhsMatF = tensor::zero;
13     if (solveMatJ) rhsMatJ = 0;
14     rhsW = vector::zero;
15 }
16 rhsE = 0;
17 if (isThermo) rhsEthermo = 0;
18
19 forAll(edges, edge) { /* loop over edges * * * * * */
20     const label& a = edges[edge][0];
21     const label& b = edges[edge][1];
22     const vector Xe_Xa = 0.5 * (meshPoints[b] - meshPoints[a]);
23     const vector Xe_Xb = -Xe_Xa;
24     const vector va = lm[a] / rhoR[a];
25     const vector vb = lm[b] / rhoR[b];
26
27     const vector vM_ = va + (vGrad[a] & (Xe_Xa)); // reconstruction velocity
28     const vector vP_ = vb + (vGrad[b] & (Xe_Xb)); //,
29     const vector dV = vP_ - vM_;
30
31     // referential normals
32     const vector& Cab = Sf[edge] ; //,          Nab = N[edge];
33     const scalar& normCab = magSf[edge];
34
35     /* spatial normals, referential density and wave speeds * * * * * */
36     const vector c_ave_ab = (0.5 * (spatH[a] + spatH[b])) & Cab;
37     const vector n_ab = c_ave_ab / Foam::mag(c_ave_ab);
38     const tensor nxn = n_ab * n_ab;
39     const tensor Inxn = tensor::I - nxn;
40     const scalar rho_ave = 0.5 * (rhoR[a] + rhoR[b]);
41     const scalar Up_ave = 0.5 * (Up[a] + Up[b]);
42     const scalar Us_ave = 0.5 * (Us[a] + Us[b]);
43
44     // Stabilisation linear momentum
45     const tensor Sp_ab = 0.5 * normCab * ((Up_ave*nxn) + (Us_ave*Inxn));
46     const vector stabC = Sp_ab & dV ;
47
48     /* * * * * * */
49     // RHS of p_{cchi}
50     const vector pR_a = lm[a] / matJ[a];
51     const vector pR_b = lm[b] / matJ[b];
52     const tensor PH_a = (P[a]+(pR_a*W[a])) & matH[a];
53     const tensor PH_b = (P[b]+(pR_b*W[b])) & matH[b];
54     const vector t_ab = (0.5 * (PH_a + PH_b)) & Cab;
55     const vector rhsLm_e = t_ab + (rho_ave * stabC);
56     rhsLm[a] += rhsLm_e;          rhsLm[b] -= rhsLm_e;
57
58     // RHS of Spatial F
59     const tensor rhsSpatF_e = (0.5 * (vHat[a]+vHat[b])) * Cab;
60     rhsSpatF[a] += rhsSpatF_e;
61     rhsSpatF[b] -= rhsSpatF_e;
62
63     if (solveSpatH) { // RHS of Spatial H
64         rhsSpatH[a] += Tcross(spatF[a], rhsSpatF_e);
65         rhsSpatH[b] -= Tcross(spatF[b], rhsSpatF_e);
66     }
67
68     if (solveSpatJ){ // RHS of Spatial J
69         const scalar pM_ = p[a] + (pGrad[a] & (Xe_Xa)); //reconstruction p
70         const scalar pP_ = p[b] + (pGrad[b] & (Xe_Xb));
71         const scalar dP = pP_ - pM_;
72
73         // Stabilisation of Spatial J
74         const tensor Sj_ab = (0.5/rho_ave) * ((nxn/Up_ave) + (Inxn/Us_ave));
75         const vector v_stab = dP * ( Sj_ab & n_ab ) ;
76         const scalar stabJC = v_stab & c_ave_ab;
77
78         rhsSpatJ[a] += (spatH[a] && rhsSpatF_e) + stabJC;
79         rhsSpatJ[b] -= (spatH[b] && rhsSpatF_e) + stabJC;
80     }
81
82     // RHS of E (Mechanical)
83     const vector v_ab = 0.5 * (va + vb);
84     scalar rhsE_ea = 0;

```

```

85 scalar rhsE_eb = 0;
86 if (Emethod) {
87     rhsE_ea = (v_ab & t_ab) * normCab;
88     rhsE_eb = -rhsE_ea;
89 }
90
91 // RHS of E (Thermal)
92 if (isThermo) {
93     const scalar& h_ = model.getThermoModel().h().value();
94     const tensor spatC_a = spatF[a].T() & spatF[a];
95     const tensor spatC_b = spatF[b].T() & spatF[b];
96     const vector Q_a = - h_ * spatJ[a] * (Foam::inv(spatC_a) & gradT[a]);
97     const vector Q_b = - h_ * spatJ[b] * (Foam::inv(spatC_b) & gradT[b]);
98     const vector Q_ave = 0.5 * (Q_a + Q_b);
99     rhsEthermo[a] += -Q_ave & Cab;
100    rhsEthermo[b] += Q_ave & Cab;
101 }
102
103 if (meshUpdate) {
104     const vector HTW_a = matH[a].T() & W[a];
105     const vector HTW_b = matH[b].T() & W[b];
106     const scalar HTW_ab_C_ab = (0.5*(HTW_a + HTW_b)) & Cab;
107
108     const tensor rhsMatF_e = (0.5 * (W[a]+W[b])) * Cab;
109     if (solveMatF) { // RHS of Material F
110         rhsMatF[a] += rhsMatF_e;
111         rhsMatF[b] -= rhsMatF_e;
112     }
113
114     if (solveMatJ) { // RHS of Material J
115         rhsMatJ[a] += HTW_ab_C_ab;
116         rhsMatJ[b] -= HTW_ab_C_ab;
117     }
118
119     /* material normals and wave speeds * * * * * */
120     const vector c_ave_ab_X = (0.5 * (matH[a] + matH[b])) & Cab;
121     const vector n_ab_X = c_ave_ab_X / Foam::mag(c_ave_ab_X);
122     const tensor nxn_X = n_ab_X * n_ab_X;
123     const tensor Inxn_X = tensor::I - nxn_X;
124
125     // reconstruction W
126     const vector wM_ = W[a] + (wGrad[a] & (Xe_Xa));
127     const vector wP_ = W[b] + (wGrad[b] & (Xe_Xb));
128     const vector dW = wP_ - wM_;
129
130     // Stabilisation of W
131     const tensor SpX_ab = 0.5 * normCab * ((Up_ave*nxn_X) + (Us_ave*Inxn_X));
132     const vector stabWC = 0.5 * (SpX_ab & dW);
133     /* * * * * * */
134     // RHS of p_{W}
135     const vector tW_ab = (0.5 * (starP[a] + starP[b]))/rho.value() & Cab;
136     const vector rhsW_e = tW_ab + stabWC;
137     rhsW[a] += rhsW_e;
138     rhsW[b] -= rhsW_e;
139
140     // RHS of E (Convection)
141     scalar E_a = E[a] / matJ[a];
142     scalar E_b = E[b] / matJ[b];
143     if (Emethod) {
144         scalar rhsE_e_convect = (0.5 * ((E_a*HTW_a) + (E_b*HTW_b))) & Cab;
145         rhsE_ea += rhsE_e_convect;
146         rhsE_eb -= rhsE_e_convect;
147     }
148 } // end of dynamic array update
149 rhsE[a] += rhsE_ea;
150 rhsE[b] += rhsE_eb;
151 /* end of edge loop * * * * * */
152 if (meshUpdate) rhsW -= (W * Vol) * dampingCoeff / deltaT.value();
153 /* * * * * * */
154
155 if (isThermo) {
156     if (Emethod) { rhsE = rhsE + rhsEthermo; }
157     else { rhsE = rhsEthermo/T; }
158 }

```

Listing D.3: rhs.H

```

1  /-----*\
2  Description
3  Compute boundary contributions for the RHS
4  \-----*/
5
6  forAll(mesh.boundary(), patch) { // loop over patches
7      const fvPatch& patx = mesh.boundary()[patch];
8      const word& PATCHNAME = patx.name();
9      forAll(patches[patch], facei) { // loop over faces
10         const label& face = patches[patch].start() + facei;
11         const vector& Sfpf = Sf_p.boundaryField()[patch][facei];
12         const scalar& Af = magSf_p.boundaryField()[patch][facei];
13         const vector& Npf = N_p.boundaryField()[patch][facei];
14         const tensor NxN = Npf*Npf, INxN = tensor::I - NxN;
15         const labelList& this_face = faces[facei];
16         const label& nodeA = this_face[0], nodeB=this_face[1], nodeC=this_face[2];
17
18         // contact velocities
19         vector vC1 = lm[nodeA]/rhoR[nodeA],
20                vC2 = lm[nodeB]/rhoR[nodeB],
21                vC3 = lm[nodeC]/rhoR[nodeC];
22
23         vector vHC1 = vHat[nodeA], vHC2 = vHat[nodeB], vHC3 = vHat[nodeC],

```

```

24     wC1 = W[nodeA],      wC2 = W[nodeB],      wC3 = W[nodeC];
25
26 // tractions
27 vector tC1 = (P[nodeA] & math[nodeA]) & Npf,
28          tC2 = (P[nodeB] & math[nodeB]) & Npf,
29          tC3 = (P[nodeC] & math[nodeC]) & Npf,
30          tWC1 = starP[nodeA] & Npf,
31          tWC2 = starP[nodeB] & Npf,
32          tWC3 = starP[nodeC] & Npf;
33
34 if (isBC(PATCHNAME, "free")) {
35     tC1 = vector::zero;    tWC1 = NxN & tWC1;
36     tC2 = vector::zero;    tWC2 = NxN & tWC2;
37     tC3 = vector::zero;    tWC3 = NxN & tWC3;
38     /*vHC1 = vHC1;*/      wC1 = INxN & wC1;
39     /*vHC2 = vHC2;*/      wC2 = INxN & wC2;
40     /*vHC3 = vHC3;*/      wC3 = INxN & wC3;
41 } else if (isBC(PATCHNAME, "roller")) {
42     tC1 = NxN & tC1;      tWC1 = NxN & tWC1;
43     tC2 = NxN & tC2;      tWC2 = NxN & tWC2;
44     tC3 = NxN & tC3;      tWC3 = NxN & tWC3;
45     vHC1 = INxN & vHC1;  wC1 = INxN & wC1;
46     vHC2 = INxN & vHC2;  wC2 = INxN & wC2;
47     vHC3 = INxN & vHC3;  wC3 = INxN & wC3;
48 } else if (isBC(PATCHNAME, "skew")) {
49     tC1 = INxN & tC1;    tWC1 = NxN & tWC1;
50     tC2 = INxN & tC2;    tWC2 = NxN & tWC2;
51     tC3 = INxN & tC3;    tWC3 = NxN & tWC3;
52     vHC1 = NxN & vHC1;  wC1 = INxN & wC1;
53     vHC2 = NxN & vHC2;  wC2 = INxN & wC2;
54     vHC3 = NxN & vHC3;  wC3 = INxN & wC3;
55 } else if (isBC(PATCHNAME, "traction")) {
56     tC1 = traction;      tWC1 = NxN & tWC1;
57     tC2 = traction;      tWC2 = NxN & tWC2;
58     tC3 = traction;      tWC3 = NxN & tWC3;
59     /*vH1 = (lm1/rhoR1) + (F1 & w1);*/ wC1 = INxN & wC1;
60     /*vH2 = (lm2/rhoR2) + (F2 & w2);*/ wC2 = INxN & wC2;
61     /*vH3 = (lm3/rhoR3) + (F3 & w3);*/ wC3 = INxN & wC3;
62 } else if (isBC(PATCHNAME, "timeTraction1")) {
63     scalar& tv = t.value();
64     vector tract =
65         vector(150,300,450) * ( tv*(tv<2.5) + (5-tv)*((tv>=2.5)&&(tv<5)) );
66     //
67     tC1 = tract;        tWC1 = NxN & tWC1;
68     tC2 = tract;        tWC2 = NxN & tWC2;
69     tC3 = tract;        tWC3 = NxN & tWC3;
70     /*vH1 = (lm1/rhoR1) + (F1 & w1);*/ wC1 = INxN & wC1;
71     /*vH2 = (lm2/rhoR2) + (F2 & w2);*/ wC2 = INxN & wC2;
72     /*vH3 = (lm3/rhoR3) + (F3 & w3);*/ wC3 = INxN & wC3;
73 } else if (isBC(PATCHNAME, "timeTraction2")) {
74     scalar& tv = t.value();
75     vector tract =
76         -vector(150,300,450) * ( tv*(tv<2.5) + (5-tv)*((tv>=2.5)&&(tv<5)) );
77     //
78     tC1 = tract;        tWC1 = NxN & tWC1;
79     tC2 = tract;        tWC2 = NxN & tWC2;
80     tC3 = tract;        tWC3 = NxN & tWC3;
81     /*vH1 = (lm1/rhoR1) + (F1 & w1);*/ wC1 = INxN & wC1;
82     /*vH2 = (lm2/rhoR2) + (F2 & w2);*/ wC2 = INxN & wC2;
83     /*vH3 = (lm3/rhoR3) + (F3 & w3);*/ wC3 = INxN & wC3;
84 } else if (isBC(PATCHNAME, "fixedStatic")) { // fixed
85     /*tC1 = tC1;*/      tWC1 = /*NxN &*/ tCC1;
86     /*tC2 = tC2;*/      tWC2 = /*NxN &*/ tWC2;
87     /*tC3 = tC3;*/      tWC3 = /*NxN &*/ tWC3;
88     vHC1 = vector::zero;    wC1 = vector::zero;
89     vHC2 = vector::zero;    wC2 = vector::zero;
90     vHC3 = vector::zero;    wC3 = vector::zero;
91 } else if (isBC(PATCHNAME, "fixed")) { // moving // to check
92     /*tC1 = tC1;*/      tWC1 = NxN & tWC1;
93     /*tC2 = tC2;*/      tWC2 = NxN & tWC2;
94     /*tC3 = tC3;*/      tWC3 = NxN & tWC3;
95     vHC1 = VV;            wC1 = vector::zero; //INxN & w1;
96     vHC2 = VV;            wC2 = vector::zero; //INxN & w2;
97     vHC3 = VV;            wC3 = vector::zero; //INxN & w3;
98 }
99
100 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
101 rhsLm[nodeA] += ((6.*tC1) + tC2 + tC3) * Af / 24.; // p_{cchi}
102 rhsLm[nodeB] += ((6.*tC2) + tC3 + tC1) * Af / 24.;
103 rhsLm[nodeC] += ((6.*tC3) + tC1 + tC2) * Af / 24.;
104
105 // Spatial F
106 const tensor gVH1 = ((6.*vHC1) + vHC2 + vHC3) * Sfpf / 24.; // Spatial F
107 const tensor gVH2 = ((6.*vHC2) + vHC3 + vHC1) * Sfpf / 24.;
108 const tensor gVH3 = ((6.*vHC3) + vHC1 + vHC2) * Sfpf / 24.;
109 rhsSpatF[nodeA] += gVH1;
110 rhsSpatF[nodeB] += gVH2;
111 rhsSpatF[nodeC] += gVH3;
112
113 if (solveSpatH) { // Spatial H
114     rhsSpatH[nodeA] += Tcross( spatF[nodeA] , gVH1 );
115     rhsSpatH[nodeB] += Tcross( spatF[nodeB] , gVH2 );
116     rhsSpatH[nodeC] += Tcross( spatF[nodeC] , gVH3 );
117 }
118
119 if (solveSpatJ) { // Spatial J
120     rhsSpatJ[nodeA] += spatH[nodeA] && gVH1;
121     rhsSpatJ[nodeB] += spatH[nodeB] && gVH2;
122     rhsSpatJ[nodeC] += spatH[nodeC] && gVH3;

```

```

123     }
124
125     if (Emethod) { // E insulated
126         rhsE[nodeA] += ((6.*tC1&vC1) + (tC2&vC2) + (tC3&vC3)) * Af / 24.;
127         rhsE[nodeB] += ((6.*tC2&vC2) + (tC3&vC3) + (tC1&vC1)) * Af / 24.;
128         rhsE[nodeC] += ((6.*tC3&vC3) + (tC1&vC1) + (tC2&vC2)) * Af / 24.;
129     }
130
131     if (meshUpdate) {
132         const tensor gw1 = ((6.*wC1) + wC2 + wC3) * Sfpf / 24.;
133         const tensor gw2 = ((6.*wC2) + wC3 + wC1) * Sfpf / 24.;
134         const tensor gw3 = ((6.*wC3) + wC1 + wC2) * Sfpf / 24.;
135
136         if (solveMatF) { // Material F
137             rhsMatF[nodeA] += gw1;
138             rhsMatF[nodeB] += gw2;
139             rhsMatF[nodeC] += gw3;
140         }
141
142         //if (solveMatH) { // Material H
143             // rhsMatH[nodeA] += Tcross( matF[nodeA] , gw1 );
144             // rhsMatH[nodeB] += Tcross( matF[nodeB] , gw2 );
145             // rhsMatH[nodeC] += Tcross( matF[nodeC] , gw3 );
146         //}
147
148         if (usePstar and (not useFEM_RHS)) { // p_{W}
149             rhsW[nodeA] += ((6.*tWC1) + tWC2 + tWC3) * Af / (24.*rho.value());
150             rhsW[nodeB] += ((6.*tWC2) + tWC3 + tWC1) * Af / (24.*rho.value());
151             rhsW[nodeC] += ((6.*tWC3) + tWC1 + tWC2) * Af / (24.*rho.value());
152         }
153     }
154     /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
155 } // end of faces loop
156 } // end of patch loop

```

Listing D.4: weakBCs.H

```

1  /*-----*\
2  Description
3  Perform time update for the current RK stage
4  \*-----*/
5
6  scalar a = RKcoef2[RKstage][0],
7          b = RKcoef2[RKstage][1],
8          c = RKcoef2[RKstage][2] * deltaT.value();
9  // Update mesh coords and vars
10 x      = (a*x.oldTime()) + (b*x) + (c*vHat);
11 xw     = (a*xw.oldTime()) + (b*xw) + (c*W);
12 lm     = (a*lm.oldTime()) + (b*lm) + (c*rhsLm/Vol);
13 spatF  = (a*spatF.oldTime()) + (b*spatF) + (c*rhsSpatF/Vol);
14 if (meshUpdate) {
15     if (solveMatF) matF = (a*matF.oldTime()) + (b*matF) + (c*rhsMatF/Vol);
16     //if (solveMatH) matH = (a*matH.oldTime()) + (b*matH) + (c*rhsMatH/Vol);
17     if (solveMatJ) matJ = (a*matJ.oldTime()) + (b*matJ) + (c*rhsMatJ/Vol);
18     if (usePstar) W = (a*W.oldTime()) + (b*W) + (c*rhsW/Vol);
19 }
20 if (solveSpatH) spatH = (a*spatH.oldTime()) + (b*spatH) + (c*rhsSpatH/Vol);
21 if (solveSpatJ) spatJ = (a*spatJ.oldTime()) + (b*spatJ) + (c*rhsSpatJ/Vol);
22
23 if (Emethod) { E = (a*E.oldTime()) + (b*E) + (c*rhsE/Vol); }
24 else { eta = (a*eta.oldTime()) + (b*eta) + (c*rhsE/Vol); }
25
26 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```

Listing D.5: integrator.H

```

1  /*-----*\
2  Description
3  Computes intermediate variables: mat F H J, nat F J, rhoR, v, vHat
4  \*-----*/
5
6  if (meshUpdate){
7      // Update Material Jacobian and Deformation Gradient (if not solved)
8      if (not solveMatF) matF = aleModel_.defGrad();
9      //if (not solveMatH) matH = Foam::det(matF) * Foam::inv(matF.T());
10     if (not solveMatJ) matJ = aleModel_.Jac();
11
12     //if (correctWP) // Correct Wave Speeds
13     //aleModel_.correctWaveSpeeds(Up,Us,model.Up(), model.Us(), matH, W, matJ, N);
14 }
15 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
16 trueF = meshUpdate ? (spatF&Foam::inv(matF)).ref() : spatF; // natural F
17 trueJ = solveSpatJ ? (spatJ/matJ).ref() : Foam::det(trueF).ref(); // natural J
18 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
19 if (not solveSpatH) spatH = Foam::det(spatF) * Foam::inv(spatF.T()); // spat H
20 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
21 rhoR = rho * matJ;
22 v_ = lm / rhoR;
23 vHat = v_ + (trueF & W);
24 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

```

Listing D.6: updateVariables.H

```

1  /-----*\
2  Description
3  Impose strong boundary conditions on W, lm and vHat
4
5  /\'strongBCs.H' MUST be applied at last, because the BCs passed from the
6  file dict (e.g. '0/lm/') likely to be wrong (most of the time)
7  \-----*/
8
9  forAll(mesh.boundary(), patch) { /* * * * * * */
10 const fvPatch& patx = mesh.boundary()[patch];
11 string PATCHNAME = patx.name();
12 if (0) { // mock
13 } else if (isBC(PATCHNAME, "free")) {
14     forAll(patches[patch], facei) {
15         const label& face = patches[patch].start() + facei;
16         const vector Npf = N_p.boundaryField()[patch][facei];
17         const tensor INxN = tensor::I - (Npf*Npf);
18         const labelList& this_face = faces[face];
19
20         // connectivity
21         const label& i1 = this_face[0];
22         const label& i2 = this_face[1];
23         const label& i3 = this_face[2];
24         W[i1] = INxN & W[i1];
25         W[i2] = INxN & W[i2];
26         W[i3] = INxN & W[i3];
27     }
28 } else if (isBC(PATCHNAME, "roller")) {
29     forAll(patches[patch], facei) {
30         const label& face = patches[patch].start() + facei;
31         const vector Npf = N_p.boundaryField()[patch][facei];
32         const tensor INxN = tensor::I - (Npf*Npf);
33         const labelList& this_face = faces[face];
34         // connectivity
35         const label& i1 = this_face[0];
36         const label& i2 = this_face[1];
37         const label& i3 = this_face[2];
38
39         W[i1] = INxN & W[i1];
40         W[i2] = INxN & W[i2];
41         W[i3] = INxN & W[i3];
42
43         lm[i1] = INxN & lm[i1];
44         lm[i2] = INxN & lm[i2];
45         lm[i3] = INxN & lm[i3];
46
47         vHat[i1] = INxN & vHat[i1];
48         vHat[i2] = INxN & vHat[i2];
49         vHat[i3] = INxN & vHat[i3];
50     }
51 } else if (isBC(PATCHNAME, "skew")) {
52     forAll(patches[patch], facei) {
53         const label& face = patches[patch].start() + facei;
54         vector Npf = N_p.boundaryField()[patch][facei];
55         tensor NxN = Npf*Npf;
56         tensor INxN = tensor::I - NxN;
57         const labelList& this_face = faces[face];
58         // connectivity
59         const label& i1 = this_face[0];
60         const label& i2 = this_face[1];
61         const label& i3 = this_face[2];
62
63         W[i1] = INxN & W[i1];
64         W[i2] = INxN & W[i2];
65         W[i3] = INxN & W[i3];
66
67         lm[i1] = NxN & lm[i1];
68         lm[i2] = NxN & lm[i2];
69         lm[i3] = NxN & lm[i3];
70
71         vHat[i1] = NxN & vHat[i1];
72         vHat[i2] = NxN & vHat[i2];
73         vHat[i3] = NxN & vHat[i3];
74     }
75 } else if (isBC(PATCHNAME, "fixedLoad_PST")) {
76     scalar t = runTime.value();
77     scalar bV = 0, V1 = 1, t1=0.005, t2 =0.01, xi = 0;
78     if (t <= t1) {
79         xi = t/t1;
80         bV = V1*xi*xi*(10.-(15.*xi)+(6.*xi*xi));
81     } else {
82         xi = (t-t1)/(t2-t1);
83         bV = V1 - V1*(xi*xi*xi*(10.-(15.*xi)+(6.*xi*xi)));
84     }
85     forAll(patches[patch], facei) {
86         const label& face = patches[patch].start() + facei;
87         vector Npf = N_p.boundaryField()[patch][facei];
88         tensor NxN = Npf*Npf;
89         tensor INxN = tensor::I - NxN;
90         const labelList& this_face = faces[face];
91         // connectivity
92         const label& i1 = this_face[0];
93         const label& i2 = this_face[1];
94         const label& i3 = this_face[2];
95
96         W[i1] = INxN & W[i1];
97         W[i2] = INxN & W[i2];
98         W[i3] = INxN & W[i3];
99     }

```

```

100
101     lm[i1][1] = bV * rhoR[i1];
102     lm[i2][1] = bV * rhoR[i2];
103     lm[i3][1] = bV * rhoR[i3];
104
105     vHat[i1][1] = bV + ((vector(0,1,0)&>trueF[i1])&W[i1]);
106     vHat[i2][1] = bV + ((vector(0,1,0)&>trueF[i2])&W[i2]);
107     vHat[i3][1] = bV + ((vector(0,1,0)&>trueF[i3])&W[i3]);
108 }
109 } else if (isBC(PATCHNAME, "fixedLoad_NECKING")) {
110     scalar t = runTime.value();
111     scalar bV = 0, V1 = 10, t1= 7e-4, t2 = 14e-4, xi = 0;
112     if (t <= t1) {
113         xi = t/t1;
114         bV = V1*xi*xi*xi*(10.-(15.*xi)+(6.*xi*xi));
115     } else if ((t <= t2) && (t > t1)) {
116         xi = (t-t1)/(t2-t1);
117         bV = V1 - (V1*xi*xi*xi*(10.-(15.*xi)+(6.*xi*xi)));
118     } else {}
119
120     forAll(patch, facei) {
121         const label& face = patches[patch].start() + facei;
122         const labelList& this_face = faces[face];
123         // connectivity
124         const label& i1 = this_face[0];
125         const label& i2 = this_face[1];
126         const label& i3 = this_face[2];
127
128         W[i1] = vector::zero;
129         W[i2] = vector::zero;
130         W[i3] = vector::zero;
131
132         lm[i1] = vector::zero;
133         lm[i2] = vector::zero;
134         lm[i3] = vector::zero;
135         vHat[i1] = vector::zero;
136         vHat[i2] = vector::zero;
137         vHat[i3] = vector::zero;
138
139         label dir = 1; // y-dir
140         lm[i1][dir] = bV * rhoR[i1];
141         lm[i2][dir] = bV * rhoR[i2];
142         lm[i3][dir] = bV * rhoR[i3];
143
144         vHat[i1][dir] = bV;
145         vHat[i2][dir] = bV;
146         vHat[i3][dir] = bV;
147     }
148 } else if (isBC(PATCHNAME, "fixedStatic")) { // fixed
149     forAll(patch, facei) {
150         const label& face = patches[patch].start() + facei;
151         forAll(faces[face], nodei) {
152             const label& node = faces[face][nodei];
153             lm[node] = vector::zero;
154             W[node] = vector::zero;
155             vHat[node] = vector::zero;
156         }
157     }
158 } else if (isBC(PATCHNAME, "fixed")) { // moving
159     forAll(patch, facei) {
160         const label& face = patches[patch].start() + facei;
161         forAll(faces[face], nodei) {
162             const label& node = faces[face][nodei];
163             lm[node] = VV * rhoR[node];
164             W[node] = vector::zero;
165             vHat[node] = VV;
166         }
167     }
168 }
169 }
170 /* * * * * *
171 // do we force 2d motion ?
172 if (xyMotion) lm = tensor(1,0,0,0,1,0,0,0,0) & lm;
173 if (xyMotionW) W = tensor(1,0,0,0,1,0,0,0,0) & W;
174 /* * * * * *

```

Listing D.7: strongBCs.H

```

1  /-----*\
2  Description
3  Update Stresses and internal variables
4  \-----*/
5
6  if (isPlastic) { /* plasticity * * * * * */
7      model.correct4(
8          P, p, // these are updated
9          trueF, trueJ,
10         CpInv, eps, T, epsDot, coeffTau, vonMises, intDEV, intVOL, // these are updated
11         isLastStage, deltaT.value()
12     );
13     //
14     if (not isLastStage) {
15         CpInv = CpInv.oldTime();
16         eps = eps.oldTime();
17         epsDot = epsDot.oldTime();
18         wp = wp.oldTime();
19         coeffTau = coeffTau.oldTime();
20     }

```

```

21
22 } else {
23   model.correctElastic(P, p, trueF, trueJ, T, intDEV, intVOL, isLastStage);
24 } /* * * * * * */
25
26 if ( meshUpdate and usePstar ) { /* update Pstar * * * * * */
27   aleModel_.computePstar(starP, trueF, matF, spatF, eps, CpInv );
28 }
29
30 /* update Temperature * * * * * */
31 K = 0.5*rho.value()* (1m/rhoR) & (1m/rhoR);
32 if (Emethod) ER = E / matJ;
33
34 if (isThermo) {
35   etaR = cv * Gamma0 * (trueJ - op.ID());
36   intVOLHeat = op.dimensionedMult(TR , etaR);
37   if (Emethod) {
38     pointScalarField tildeE_R = intDEV + intVOL + intVOLHeat;
39     T = TR + ((ER - K - tildeE_R) / cv);
40   } else {
41     T = op.dimensionedMult(TR , op.exp((eta - etaR)/cv) );
42   }
43 }

```

Listing D.8: updateStress.H

```

1  /*-----*\
2  Description
3  Compute source terms for internal variables, plastic dissipation
4  and energy
5  \*-----*/
6
7  if (isPlastic) { /* Plasticity * * * * * */
8    wpdot = deltaT.value()*op.dimensionedMult(vonMises, epsDot);
9    wp += wpdot;
10
11    if (isThermo and (not Emethod)){
12      eta += (wpdot / T);
13    }
14 } // end of Plasticity * * * * * */
15
16
17 if (meshUpdate) { // ALE: integration of source terms for internal variables
18   rhsD = tensor::zero;      rhsA = 0; rhsWp = 0;          // reset RHS
19   rhsE = 0;
20   forAll(edges, edge) { // edge loop
21     const label& a = edges[edge][0];
22     const label& b = edges[edge][1];
23
24     // referential normals
25     const vector& Cab = Sf[edge];
26     const vector HTW_a = matH[a].T() & W[a];
27     const vector HTW_b = matH[b].T() & W[b];
28     const scalar HTWC_Ja = (HTW_a & Cab) / Foam::det(matF[a]); //
29     const scalar HTWC_Jb = (HTW_b & Cab) / Foam::det(matF[b]); //
30
31     if (isPlastic) {
32       // RHS of CpInv
33       const tensor rhsD_ea = (0.5 * (CpInv[a] + CpInv[b])) * HTWC_Ja;
34       const tensor rhsD_eb = (0.5 * (CpInv[a] + CpInv[b])) * HTWC_Jb;
35       rhsD[a] += rhsD_ea;      rhsD[b] -= rhsD_eb;
36
37       // RHS of eps_p
38       const scalar rhsA_ea = (0.5 * (eps[a] + eps[b])) * HTWC_Ja;
39       const scalar rhsA_eb = (0.5 * (eps[a] + eps[b])) * HTWC_Jb;
40       rhsA[a] += rhsA_ea;      rhsA[b] -= rhsA_eb;
41
42       // RHS of wp
43       const scalar rhsWp_ea = (0.5 * (wp[a] + wp[b])) * HTWC_Ja;
44       const scalar rhsWp_eb = (0.5 * (wp[a] + wp[b])) * HTWC_Jb;
45       rhsWp[a] += rhsWp_ea;    rhsWp[b] -= rhsWp_eb;
46     }
47
48     if (isThermo) {
49       // RHS of eta
50       const scalar rhsETA_ea = (0.5 * (eta[a] + eta[b])) * HTWC_Ja;
51       const scalar rhsETA_eb = (0.5 * (eta[a] + eta[b])) * HTWC_Jb;
52       rhsE[a] += rhsETA_ea;    rhsE[b] -= rhsETA_eb;
53     }
54   } // end of edge loop
55
56   // get particle id to move ALE - and update position
57   forAll(meshPoints, pos) { // loop over nodes
58     if ( (1. - coeffTau[ pos ]) > tolerance ) {
59       scalar dt_over_V = deltaT.value() / Vol[pos];
60       CpInv[pos] = CpInv[pos] + (rhsD[pos] * dt_over_V);
61       eps[pos] = eps[pos] + (rhsA[pos] * dt_over_V);
62       wp[pos] = wp[pos] + (rhsWp[pos] * dt_over_V);
63       eta[pos] = eta[pos] + (rhsE[pos] * dt_over_V);
64     } else W[ pos ] = vector::zero;
65   } // end of node loop
66 } // end of meshUpdate
67 /* * * * * * */

```

Listing D.9: backwardEuler.H

References

- [1] E. A. de Souza Neto, D. Peri, and D. R. J. Owen. *Computational Methods for Plasticity*. Publication Title: Computational Methods for Plasticity. Chichester, UK: John Wiley and Sons, Ltd, Oct. 2008. DOI: [10.1002/9780470694626](https://doi.org/10.1002/9780470694626).
- [2] C. H. Lee, A. J. Gil, and J. Bonet. “Development of a cell centred upwind finite volume algorithm for a new conservation law formulation in structural dynamics”. In: *Computers and Structures* 118 (2013), pp. 13–38. DOI: [10.1016/j.compstruc.2012.12.008](https://doi.org/10.1016/j.compstruc.2012.12.008).
- [3] E. Kuhl, H. Askes, and P. Steinmann. “An ALE formulation based on spatial and material settings of continuum mechanics. Part 1: Generic hyperelastic formulation”. In: *Computer Methods in Applied Mechanics and Engineering* 193.39-41 SPEC. ISS. (2004), pp. 4207–4222. DOI: [10.1016/j.cma.2003.09.030](https://doi.org/10.1016/j.cma.2003.09.030).
- [4] H. Askes, E. Kuhl, and P. Steinmann. “An ALE formulation based on spatial and material settings of continuum mechanics. Part 2: Classification and applications”. In: *Computer Methods in Applied Mechanics and Engineering* 193.39-41 SPEC. ISS. (2004), pp. 4223–4245. DOI: [10.1016/j.cma.2003.09.031](https://doi.org/10.1016/j.cma.2003.09.031).
- [5] E. Kuhl and P. Steinmann. “A hyperelastodynamic ALE formulation based on referential, spatial and material settings of continuum mechanics”. In: *Acta Mechanica* 174.3-4 (2005), pp. 201–222. DOI: [10.1007/s00707-004-0200-4](https://doi.org/10.1007/s00707-004-0200-4).
- [6] F. Armero and E. Love. “An arbitrary Lagrangian-Eulerian finite element method for finite strain plasticity”. In: *International Journal for Numerical Methods in Engineering* 57.4 (2003), pp. 471–508. DOI: [10.1002/nme.684](https://doi.org/10.1002/nme.684).
- [7] R. Ortigosa et al. “A new energy–momentum time integration scheme for non-linear thermo-mechanics”. In: *Computer Methods in Applied Mechanics and Engineering* 372 (2020). DOI: [10.1016/j.cma.2020.113395](https://doi.org/10.1016/j.cma.2020.113395).
- [8] G. A. Holzapfel. “Nonlinear Solid Mechanics: A Continuum Approach”. In: (2007). ISBN: 0471823198, p. 455.
- [9] J. Degroote et al. “Partitioned simulation of the interaction between an elastic structure and free surface flow”. In: *Computer Methods in Applied Mechanics and Engineering* 199.33-36 (2010). Publisher: Elsevier B.V., pp. 2085–2098. DOI: [10.1016/j.cma.2010.02.019](https://doi.org/10.1016/j.cma.2010.02.019).
- [10] J. Bonet, A. J. Gil, and R. D. Wood. *Nonlinear Solid Mechanics for Finite Element Analysis: Dynamics*. Publication Title: Nonlinear Solid Mechanics for Finite Element Analysis: Dynamics. Cambridge University Press, Mar. 2021. DOI: [10.1017/9781316336083](https://doi.org/10.1017/9781316336083).
- [11] T. Heuzé and L. Stainier. “A variational formulation of thermomechanical constitutive update for hyperbolic conservation laws A variational formulation of thermomechanical constitutive update for hyperbolic A variational formulation of thermomechanical constitutive update for hyperbolic conservation laws”. In: *Computer Methods in Applied Mechanics and Engineering* 394 (2022), p. 114893. DOI: [10.1016/j.cma.2022.114893](https://doi.org/10.1016/j.cma.2022.114893).
- [12] E. Oñate et al. *Advances in Computational Plasticity*. Ed. by E. Oñate et al. Vol. 46. Series Title: Computational Methods in Applied Sciences. Cham: Springer International Publishing, 2018. DOI: [10.1007/978-3-319-60885-3](https://doi.org/10.1007/978-3-319-60885-3).

- [13] P. Cardiff et al. “A Lagrangian cell-centred finite volume method for metal forming simulation”. In: *International Journal for Numerical Methods in Engineering* 109.13 (2017), pp. 1777–1803. DOI: [10.1002/nme.5345](https://doi.org/10.1002/nme.5345).
- [14] M. Kuroda and V. Tvergaard. “Shear band development predicted by a non-normality theory of plasticity and comparison to crystal plasticity predictions”. In: *International Journal of Solids and Structures* 38.50-51 (Dec. 2001), pp. 8945–8960. DOI: [10.1016/S0020-7683\(01\)00166-4](https://doi.org/10.1016/S0020-7683(01)00166-4).
- [15] F. Ciardo et al. “A fast boundary element based solver for localized inelastic deformations”. In: *International Journal for Numerical Methods in Engineering* 121.24 (Dec. 2020). Publisher: John Wiley and Sons Ltd, pp. 5696–5718. DOI: [10.1002/nme.6520](https://doi.org/10.1002/nme.6520).
- [16] V. Tvergaard, A. Needleman, and K. K. Lo. “Flow localization in the plane strain tensile test”. In: *Journal of the Mechanics and Physics of Solids* 29.2 (1981), pp. 115–142. DOI: [10.1016/0022-5096\(81\)90019-3](https://doi.org/10.1016/0022-5096(81)90019-3).
- [17] Y. Suzuki and S. Koshizuka. “A Hamiltonian particle method for non-linear elastodynamics”. en. In: *International Journal for Numerical Methods in Engineering* 74.8 (May 2008), pp. 1344–1373. DOI: [10.1002/nme.2222](https://doi.org/10.1002/nme.2222).
- [18] H. M. Koh, H. S. Lee, and R. B. Haber. “Dynamic crack propagation analysis using Eulerian-Lagrangian kinematic descriptions”. In: *Computational Mechanics* 3.3 (1988), pp. 141–155. DOI: [10.1007/BF00297441](https://doi.org/10.1007/BF00297441).
- [19] T. Rabczuk et al. “A simple and robust three-dimensional cracking-particle method without enrichment”. In: *Computer Methods in Applied Mechanics and Engineering* 199.37-40 (2010). Publisher: Elsevier B.V., pp. 2437–2455. DOI: [10.1016/j.cma.2010.03.031](https://doi.org/10.1016/j.cma.2010.03.031).
- [20] P. Roberto and R. De Campos. *A New Updated Reference Lagrangian Smooth Particle Hydrodynamics Framework for Large Strain Solid Dynamics and its Extension to Dynamic Fracture*. Tech. rep. 2022.
- [21] C. H. Lee et al. “An entropy-stable updated reference Lagrangian smoothed particle hydrodynamics algorithm for thermo-elasticity and thermo-visco-plasticity”. en. In: *Computational Particle Mechanics* (Apr. 2023). DOI: [10.1007/s40571-023-00564-3](https://doi.org/10.1007/s40571-023-00564-3).
- [22] A. Ghavamian et al. “An entropy-stable Smooth Particle Hydrodynamics algorithm for large strain thermo-elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 379 (2021). Publisher: Elsevier B.V., p. 113736. DOI: [10.1016/j.cma.2021.113736](https://doi.org/10.1016/j.cma.2021.113736).
- [23] J. Bonet et al. “A first order hyperbolic framework for large strain computational solid dynamics. Part III: Thermo-elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 373 (2021). Publisher: Elsevier B.V., p. 113505. DOI: [10.1016/j.cma.2020.113505](https://doi.org/10.1016/j.cma.2020.113505).
- [24] M. Cervera, M. Chiumenti, and C. A. de Saracibar. “Softening, localization and stabilization: Capture of discontinuous solutions in J2 plasticity”. In: *International Journal for Numerical and Analytical Methods in Geomechanics* 28.5 (Apr. 2004), pp. 373–393. DOI: [10.1002/nag.341](https://doi.org/10.1002/nag.341).
- [25] D. Samantaray, S. Mandal, and A. K. Bhaduri. “A comparative study on Johnson Cook, modified Zerilli-Armstrong and Arrhenius-type constitutive models to predict elevated temperature flow behaviour in modified 9Cr-1Mo steel”. In: *Computational Materials Science* 47.2 (Dec. 2009), pp. 568–576. DOI: [10.1016/j.commatsci.2009.09.025](https://doi.org/10.1016/j.commatsci.2009.09.025).
- [26] G. R. Johnson and W. H. Cook. “Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures”. In: *Engineering Fracture Mechanics* 21.1 (1985), pp. 31–48. DOI: [10.1016/0013-7944\(85\)90052-9](https://doi.org/10.1016/0013-7944(85)90052-9).
- [27] G. H. Miller. “An iterative Riemann solver for systems of hyperbolic conservation laws, with application to hyperelastic solid mechanics”. In: *Journal of Computational Physics* 193.1 (2004), pp. 198–225. DOI: [10.1016/j.jcp.2003.08.005](https://doi.org/10.1016/j.jcp.2003.08.005).
- [28] D. S. Lemons and C. M. Lund. “Thermodynamics of high temperature, Mie-Grüneisen solids”. en. In: *American Journal of Physics* 67.12 (Dec. 1999), pp. 1105–1108. DOI: [10.1119/1.19091](https://doi.org/10.1119/1.19091).

- [29] M. S. Chafi, G. Karami, and M. Ziejewski. “Numerical analysis of blast-induced wave propagation using FSI and ALE multi-material formulations”. In: *International Journal of Impact Engineering* 36.10-11 (2009). Publisher: Elsevier Ltd, pp. 1269–1275. DOI: [10.1016/j.ijimpeng.2009.03.007](https://doi.org/10.1016/j.ijimpeng.2009.03.007).
- [30] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Publication Title: Finite Volume Methods for Hyperbolic Problems Issue: 1 ISSN: 0521810876, 0521009243, 9780521810876, 9780511042195. 2002. DOI: [10.1017/cbo9780511791253](https://doi.org/10.1017/cbo9780511791253).
- [31] G. Pijaudier-Cabot, L. Bodé, and A. Huerta. “Arbitrary Lagrangian-Eulerian finite element analysis of strain localization in transient problems”. In: *International Journal for Numerical Methods in Engineering* 38.24 (Dec. 1995), pp. 4171–4191. DOI: [10.1002/nme.1620382406](https://doi.org/10.1002/nme.1620382406).
- [32] A. Kurganov, S. Noelle, and G. Petrova. *Semidiscrete Central-Upwind Schemes for Hyperbolic Conservation laws and Hamilton-Jacobi Equations*. Vol. 23. Publication Title: SIAM Journal on Scientific Computing Issue: 3. 2001.
- [33] J. D. Achenbach. *Wave propagation in elastic solids*. en. North-Holland series in applied mathematics and mechanics v. 16. Amsterdam New York: North-Holland Pub. Co. American Elsevier Pub. Co, 1973.
- [34] G. C. Sih and J. F. Loeber. “Wave propagation in an elastic solid with a line of discontinuity or finite crack”. en. In: *Quarterly of Applied Mathematics* 27.2 (1969), pp. 193–213. DOI: [10.1090/qam/99830](https://doi.org/10.1090/qam/99830).
- [35] M. Aguirre et al. “A vertex centred Finite Volume Jameson-Schmidt-Turkel (JST) algorithm for a mixed conservation formulation in solid dynamics”. In: *Journal of Computational Physics* 259 (2014). Publisher: Elsevier Inc., pp. 672–699. DOI: [10.1016/j.jcp.2013.12.012](https://doi.org/10.1016/j.jcp.2013.12.012).
- [36] J. Haider et al. “An upwind cell centred Total Lagrangian finite volume algorithm for nearly incompressible explicit fast solid dynamic applications”. In: *Computer Methods in Applied Mechanics and Engineering* 340 (2018). Publisher: Elsevier B.V., pp. 684–727. DOI: [10.1016/j.cma.2018.06.010](https://doi.org/10.1016/j.cma.2018.06.010).
- [37] P. R. de Campos et al. “A New Updated Reference Lagrangian Smooth Particle Hydrodynamics algorithm for isothermal elasticity and elasto-plasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 392 (Mar. 2022). Publisher: Elsevier B.V. DOI: [10.1016/j.cma.2022.114680](https://doi.org/10.1016/j.cma.2022.114680).
- [38] A. Gargallo-Peiró et al. “Optimization of a regularized distortion measure to generate curved high-order unstructured tetrahedral meshes”. In: *International Journal for Numerical Methods in Engineering* 103.5 (2015), pp. 342–363. DOI: [10.1002/nme.4888](https://doi.org/10.1002/nme.4888).
- [39] C. Geuzaine and J. F. Remacle. “Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities”. In: *International Journal for Numerical Methods in Engineering* 79.11 (Sept. 2009), pp. 1309–1331. DOI: [10.1002/nme.2579](https://doi.org/10.1002/nme.2579).
- [40] V. H. Bhusare et al. “CFD simulations of a bubble column with and without internals by using OpenFOAM”. In: *Chemical Engineering Journal* 317 (2017). Publisher: Elsevier B.V., pp. 157–174. DOI: [10.1016/j.cej.2017.01.128](https://doi.org/10.1016/j.cej.2017.01.128).
- [41] J. Vila-Pérez, M. Giacomini, and A. Huerta. “Benchmarking the face-centred finite volume method for compressible laminar flows”. In: (Aug. 2022). arXiv: 2208.02848.
- [42] P. Cardiff et al. “A block-coupled Finite Volume methodology for linear elasticity and unstructured meshes”. In: *Computers and Structures* 175 (2016), pp. 100–122. DOI: [10.1016/j.compstruc.2016.07.004](https://doi.org/10.1016/j.compstruc.2016.07.004).
- [43] *Abaqus/CAE User’s Manual Abaqus 6.11 Abaqus/CAE User’s Manual*.
- [44] *ANSYS Modeling and Meshing Guide*. Tech. rep.
- [45] *HyperWorks 2017 RADIOSS User Guide*. Tech. rep. 1986.
- [46] *COMSOL Multiphysics Reference Manual*. Tech. rep. 2019.
- [47] ESI. *Pam-Crash*. 2022.
- [48] C. Greenshields. *OpenFOAM v6 User Guide*. London, UK: The OpenFOAM Foundation, 2018.

- [49] H. Jasak. “OpenFOAM: Open source CFD in research and industry”. In: *International Journal of Naval Architecture and Ocean Engineering* 1.2 (Dec. 2009). Publisher: Elsevier BV, pp. 89–94. DOI: [10.2478/ijnaoe-2013-0011](https://doi.org/10.2478/ijnaoe-2013-0011).
- [50] OpenCFD Limited. *OpenFOAM User Guide*. Issue: December. 2020.
- [51] *SALOME: The Open Source Integration Platform for Numerical Simulation*. Tech. rep. 2022.
- [52] F Hecht. “New development in FreeFem++”. In: *J. Numer. Math.* 20.3-4 (2012), pp. 251–265.
- [53] S. Turek et al. “Numerical simulation and benchmarking of a monolithic multigrid solver for fluid-structure interaction problems with application to hemodynamics”. In: *Lecture Notes in Computational Science and Engineering* 73 LNCSE (2010). ISBN: 9783642142055, pp. 193–220. DOI: [10.1007/978-3-642-14206-2_8](https://doi.org/10.1007/978-3-642-14206-2_8).
- [54] A. J. Gil et al. “The immersed structural potential method for haemodynamic applications”. In: *Journal of Computational Physics* 229.22 (2010). Publisher: Elsevier Inc., pp. 8613–8641. DOI: [10.1016/j.jcp.2010.08.005](https://doi.org/10.1016/j.jcp.2010.08.005).
- [55] J. Haider. “OpenFOAM course for beginners : Hands-on training OpenFOAM course for beginners : Hands-on training”. In: September (2018). DOI: [10.5281/zenodo.1435601](https://doi.org/10.5281/zenodo.1435601).
- [56] O. Ibrahim and I. Hassan. “A vertex centred Finite Volume algorithm for fast solid dynamics : Total and Updated Lagrangian descriptions”. In: September (2018).
- [57] C. Farhat et al. “Robust and provably second-order explicit-explicit and implicit-explicit staggered time-integrators for highly non-linear compressible fluid-structure interaction problems: Staggered Algorithms for non-linear Fluid-Structure Problems”. en. In: *International Journal for Numerical Methods in Engineering* 84.1 (Oct. 2010), pp. 73–107. DOI: [10.1002/nme.2883](https://doi.org/10.1002/nme.2883).
- [58] M. Aguirre et al. “An upwind vertex centred Finite Volume solver for Lagrangian solid dynamics”. In: *Journal of Computational Physics* 300 (2015). Publisher: Elsevier Inc., pp. 387–422. DOI: [10.1016/j.jcp.2015.07.029](https://doi.org/10.1016/j.jcp.2015.07.029).
- [59] B. Howell and G. Ball. “A Free-Lagrange Augmented Godunov Method for the Simulation of Elastic–Plastic Solids”. en. In: *Journal of Computational Physics* 175.1 (Jan. 2002), pp. 128–167. DOI: [10.1006/jcph.2001.6931](https://doi.org/10.1006/jcph.2001.6931).
- [60] A. Khayyer et al. “A 3D SPH-based entirely Lagrangian meshfree hydroelastic FSI solver for anisotropic composite structures”. en. In: *Applied Mathematical Modelling* 112 (Dec. 2022), pp. 560–613. DOI: [10.1016/j.apm.2022.07.031](https://doi.org/10.1016/j.apm.2022.07.031).
- [61] S. Nicolici and R. M. Bilegan. “Fluid structure interaction modeling of liquid sloshing phenomena in flexible tanks”. In: *Nuclear Engineering and Design* 258 (2013). Publisher: Elsevier B.V., pp. 51–56. DOI: [10.1016/j.nucengdes.2012.12.024](https://doi.org/10.1016/j.nucengdes.2012.12.024).
- [62] S. T. Miller et al. “An Overset Grid Method for Fluid-Structure Interaction”. In: *World Journal of Mechanics* 04.07 (2014), pp. 217–237. DOI: [10.4236/wjm.2014.47023](https://doi.org/10.4236/wjm.2014.47023).
- [63] B. Gatzhammer. “Efficient and Flexible Partitioned Simulation of Fluid-Structure Interactions”. In: (2014). arXiv: cond-mat/0402594v3 ISBN: 0950-5849, pp. 30–31. DOI: [10.1016/j.infsof.2008.09.005](https://doi.org/10.1016/j.infsof.2008.09.005).
- [64] K. Takizawa et al. “Methods for FSI modeling of spacecraft parachute dynamics and cover separation”. In: *Mathematical Models and Methods in Applied Sciences* 23.2 (2013), pp. 307–338. DOI: [10.1142/S0218202513400058](https://doi.org/10.1142/S0218202513400058).
- [65] K. Takizawa and T. E. Tezduyar. *Space-time fluid-structure interaction methods*. Vol. 22. Publication Title: Mathematical Models and Methods in Applied Sciences Issue: SUPPL.2 ISSN: 02182025. 2012. DOI: [10.1142/S0218202512300013](https://doi.org/10.1142/S0218202512300013).
- [66] H. Jasak and Z. Tukovic. “Dynamic mesh handling in OpenFOAM applied to fluid-structure interaction simulations”. In: *European Conference on Computational Fluid Dynamics* June (2010), pp. 1–19.
- [67] P Cignoni et al. “MeshLab: an Open-Source Mesh Processing Tool”. en. In: ()
- [68] T. Schneider et al. “A Large-Scale Comparison of Tetrahedral and Hexahedral Elements for Solving Elliptic PDEs with the Finite Element Method”. In: *ACM Transactions on Graphics* 41.3 (June 2022). arXiv: 1903.09332 Publisher: Association for Computing Machinery. DOI: [10.1145/3508372](https://doi.org/10.1145/3508372).

- [69] G. Scovazzi et al. “A simple, stable, and accurate linear tetrahedral finite element for transient, nearly, and fully incompressible solid dynamics: A dynamic variational multiscale approach”. In: *International Journal for Numerical Methods in Engineering* 106.10 (June 2016). Publisher: John Wiley and Sons Ltd, pp. 799–839. DOI: [10.1002/nme.5138](https://doi.org/10.1002/nme.5138).
- [70] F. M. Pires, E. A. de Souza Neto, and J. L. de la Cuesta Padilla. “An assessment of the average nodal volume formulation for the analysis of nearly incompressible solids under finite strains”. In: *Communications in Numerical Methods in Engineering* 20.7 (2004), pp. 569–583. DOI: [10.1002/cnm.697](https://doi.org/10.1002/cnm.697).
- [71] K. J. Bathe. *Finite Element Procedures*. Publication Title: Englewood Cliffs New Jersey, 1996.
- [72] Y. Onishi and K. Amaya. “Performance evaluation of the selective smoothed finite element methods using tetrahedral elements with deviatoric/hydrostatic split in large deformation analysis”. In: *Theoretical and Applied Mechanics Japan* 63 (2015), pp. 55–65. DOI: [10.11345/nctam.63.55](https://doi.org/10.11345/nctam.63.55).
- [73] L. R. G. Treloar. *The physics of rubber elasticity: by L.R.G. Treloar*. en. 3rd ed. Oxford classic texts in the physical sciences. Oxford : New York: Clarendon Press ; Oxford University Press, 2005.
- [74] D. J. Payen and K.-J. Bathe. “Improved stresses for the 4-node tetrahedral element”. en. In: *Computers & Structures* 89.13-14 (July 2011), pp. 1265–1273. DOI: [10.1016/j.compstruc.2011.02.009](https://doi.org/10.1016/j.compstruc.2011.02.009).
- [75] J. Haider. “An upwind cell centred finite volume method for large strain explicit solid dynamics in OpenFOAM”. In: *TDX (Tesis Doctorals en Xarxa)* M.June (2018).
- [76] A. J. Gil et al. “A stabilised Petrov–Galerkin formulation for linear tetrahedral elements in compressible, nearly incompressible and truly incompressible fast dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 276 (July 2014), pp. 659–690. DOI: [10.1016/j.cma.2014.04.006](https://doi.org/10.1016/j.cma.2014.04.006).
- [77] A. J. Gil et al. “A discrete geometric conservation law (DGCL) for a cell vertex finite-volume algorithm on moving domains”. In: *International Journal for Numerical Methods in Biomedical Engineering* 26.6 (June 2010), pp. 770–779. DOI: [10.1002/cnm.1180](https://doi.org/10.1002/cnm.1180).
- [78] A. J. Gil et al. “A first order hyperbolic framework for large strain computational solid dynamics. Part II: Total Lagrangian compressible, nearly incompressible and truly incompressible elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 300.November 2015 (2016). Publisher: Elsevier Ltd, pp. 146–181. DOI: [10.1016/j.cma.2015.11.010](https://doi.org/10.1016/j.cma.2015.11.010).
- [79] C. H. Lee, A. J. Gil, and J. Bonet. “Development of a stabilised Petrov-Galerkin formulation for conservation laws in Lagrangian fast solid dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 268 (2014). Publisher: Elsevier B.V., pp. 40–64. DOI: [10.1016/j.cma.2013.09.004](https://doi.org/10.1016/j.cma.2013.09.004).
- [80] C. H. Lee et al. “A new Jameson–Schmidt–Turkel Smooth Particle Hydrodynamics algorithm for large strain explicit fast dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 311 (Nov. 2016). Publisher: Elsevier B.V., pp. 71–111. DOI: [10.1016/j.cma.2016.07.033](https://doi.org/10.1016/j.cma.2016.07.033).
- [81] M. Aguirre. “A vertex centred Finite Volume Method for solid dynamics, PhD thesis”. In: *Civil Engineering Dept. Swansea University* M (2014).
- [82] O. I. Hassan et al. “An upwind vertex centred finite volume algorithm for nearly and truly incompressible explicit fast solid dynamic applications: Total and Updated Lagrangian formulations”. In: *Journal of Computational Physics: X* 3 (June 2019). Publisher: Academic Press Inc. DOI: [10.1016/j.jcpx.2019.100025](https://doi.org/10.1016/j.jcpx.2019.100025).
- [83] N. Abboud and G. Scovazzi. “Elastoplasticity with linear tetrahedral elements: A variational multiscale method”. In: (2018). DOI: [10.1002/nme.5831](https://doi.org/10.1002/nme.5831).
- [84] M. Chiumenti et al. “A stabilized formulation for incompressible plasticity using linear triangles and tetrahedra”. In: *International Journal of Plasticity* 20.8-9 (Aug. 2004), pp. 1487–1504. DOI: [10.1016/j.ijplas.2003.11.009](https://doi.org/10.1016/j.ijplas.2003.11.009).

- [85] L. Noels, L. Stainier, and J. P. Ponthot. “An energy-momentum conserving algorithm for non-linear hypoelastic constitutive models”. In: *International Journal for Numerical Methods in Engineering* 59.1 (2004), pp. 83–114. DOI: [10.1002/nme.869](https://doi.org/10.1002/nme.869).
- [86] E. A. De Souza Neto et al. “Design of simple low order finite elements for large strain analysis of nearly incompressible solids”. In: *International Journal of Solids and Structures* 33.20-22 (1996), pp. 3277–3296. DOI: [10.1016/0020-7683\(95\)00259-6](https://doi.org/10.1016/0020-7683(95)00259-6).
- [87] A. Pérez-Foguet, A. Rodríguez-Ferran, and A. Huerta. “Consistent tangent matrices for substepping schemes”. In: *Computer Methods in Applied Mechanics and Engineering* 190.35-36 (2001), pp. 4627–4647. DOI: [10.1016/S0045-7825\(00\)00336-4](https://doi.org/10.1016/S0045-7825(00)00336-4).
- [88] T. Heuzé. “Simulation of impacts on elastic–viscoplastic solids with the flux-difference splitting finite volume method applied to non-uniform quadrilateral meshes”. en. In: *Advanced Modeling and Simulation in Engineering Sciences* 5.1 (Dec. 2018), p. 9. DOI: [10.1186/s40323-018-0101-z](https://doi.org/10.1186/s40323-018-0101-z).
- [89] M. A. Puso and J. Solberg. “A stabilized nodally integrated tetrahedral”. en. In: *International Journal for Numerical Methods in Engineering* 67.6 (Aug. 2006), pp. 841–867. DOI: [10.1002/nme.1651](https://doi.org/10.1002/nme.1651).
- [90] E. Onate et al. “Non Linear Dynamic Analysis of Solids using Linear Triangles and Tetrahedra”. en. In: ().
- [91] N. Molino, R. Bridson, and R. Fedkiw. “Tetrahedral Mesh Generation for Deformable Bodies”. en. In: ().
- [92] Q. Du and D. Wang. “Recent progress in robust and quality Delaunay mesh generation”. en. In: *Journal of Computational and Applied Mathematics* 195.1-2 (Oct. 2006), pp. 8–23. DOI: [10.1016/j.cam.2005.07.014](https://doi.org/10.1016/j.cam.2005.07.014).
- [93] P. Cardiff et al. “An open-source finite volume toolbox for solid mechanics and fluid-solid interaction simulations”. In: (2018). arXiv: 1808.10736, pp. 1–45.
- [94] R. Eymard, T. Gallouët, and R. Herbin. “Finite Volume Methods”. en. In: (2019).
- [95] P. H. Maire et al. “A nominally second-order cell-centered Lagrangian scheme for simulating elastic-plastic flows on two-dimensional unstructured grids”. In: *Journal of Computational Physics* 235 (2013). Publisher: Elsevier Inc., pp. 626–665. DOI: [10.1016/j.jcp.2012.10.017](https://doi.org/10.1016/j.jcp.2012.10.017).
- [96] Z. Tukovic et al. “Openfoam finite volume solver for fluid-solid interaction”. In: *Transactions of Famena* 42.3 (2018), pp. 1–31. DOI: [10.21278/TOF.42301](https://doi.org/10.21278/TOF.42301).
- [97] J. Haider et al. “Development of a high performance parallel computing toolkit for fast solid dynamics in OpenFOAM”. In: (2018), pp. 1–26.
- [98] A. Afzal et al. “Parallelization Strategies for Computational Fluid Dynamics Software: State of the Art Review”. en. In: *Archives of Computational Methods in Engineering* 24.2 (Apr. 2017), pp. 337–363. DOI: [10.1007/s11831-016-9165-4](https://doi.org/10.1007/s11831-016-9165-4).
- [99] R. L. Taylor. *FEAP-A Finite Element Analysis Program*. Tech. rep. 2007.
- [100] M. L. Wilkins. *Calculation of Elastic-Plastic flow distribution*. Tech. rep. 1963.
- [101] J. A. Trangenstein and P. Colella. “A higher-order Godunov method for modeling finite deformation in elastic-plastic solids”. In: *Communications on Pure and Applied Mathematics* 44.1 (Jan. 1991), pp. 41–100. DOI: [10.1002/cpa.3160440103](https://doi.org/10.1002/cpa.3160440103).
- [102] J. A. Trangenstein and R. B. Pember. “The Riemann Problem for Longitudinal Motion in an Elastic-Plastic Bar”. In: *SIAM Journal on Scientific and Statistical Computing* 12.1 (Jan. 1991), pp. 180–207. DOI: [10.1137/0912010](https://doi.org/10.1137/0912010).
- [103] J. A. Trangenstein and R. B. Pember. “Numerical algorithms for strong discontinuities in elastic—plastic solids”. In: *Journal of Computational Physics* 103.1 (Nov. 1992), pp. 63–89. DOI: [10.1016/0021-9991\(92\)90326-T](https://doi.org/10.1016/0021-9991(92)90326-T).
- [104] J. A. Trangenstein. “A second-order Godunov algorithm for two-dimensional solid mechanics”. In: *Computational Mechanics* 13.5 (Sept. 1994), pp. 343–359. DOI: [10.1007/BF00512588](https://doi.org/10.1007/BF00512588).

- [105] N. C. Nguyen and J. Peraire. “Hybridizable discontinuous Galerkin methods for partial differential equations in continuum mechanics”. In: *Journal of Computational Physics* 231.18 (July 2012). Publisher: Academic Press Inc., pp. 5955–5988. DOI: [10.1016/j.jcp.2012.02.033](https://doi.org/10.1016/j.jcp.2012.02.033).
- [106] I. A. Karim et al. “A two-step taylor-galerkin formulation for fast dynamics”. In: *Engineering Computations (Swansea, Wales)* 31.3 (2014). Publisher: Emerald Group Publishing Ltd., pp. 366–387. DOI: [10.1108/EC-12-2012-0319](https://doi.org/10.1108/EC-12-2012-0319).
- [107] J. Haider et al. “A first-order hyperbolic framework for large strain computational solid dynamics: An upwind cell centred Total Lagrangian scheme”. In: *International Journal for Numerical Methods in Engineering* 109.3 (2017), pp. 407–456. DOI: [10.1002/nme.5293](https://doi.org/10.1002/nme.5293).
- [108] C. H. Lee et al. “A variationally consistent Streamline Upwind Petrov–Galerkin Smooth Particle Hydrodynamics algorithm for large strain solid dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 318 (2017). Publisher: Elsevier Ltd, pp. 514–536. DOI: [10.1016/j.cma.2017.02.002](https://doi.org/10.1016/j.cma.2017.02.002).
- [109] C. H. Lee et al. “A Total Lagrangian upwind Smooth Particle Hydrodynamics algorithm for large strain explicit solid dynamics”. In: *Computer Methods in Applied Mechanics and Engineering* 344 (2019). Publisher: Elsevier B.V., pp. 209–250. DOI: [10.1016/j.cma.2018.09.033](https://doi.org/10.1016/j.cma.2018.09.033).
- [110] C. J. Runcie et al. “An acoustic Riemann solver for large strain computational contact dynamics”. In: *International Journal for Numerical Methods in Engineering* (Sept. 2022). Publisher: Wiley. DOI: [10.1002/nme.7085](https://doi.org/10.1002/nme.7085).
- [111] J. Bonet et al. “A first order hyperbolic framework for large strain computational solid dynamics. Part I: Total Lagrangian isothermal elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 283 (2015). Publisher: Elsevier Ltd, pp. 689–732. DOI: [10.1016/j.cma.2014.09.024](https://doi.org/10.1016/j.cma.2014.09.024).
- [112] T. Heuzé and L. Stainier. “A variational formulation of thermomechanical constitutive update for hyperbolic conservation laws”. en. In: *Computer Methods in Applied Mechanics and Engineering* 394 (May 2022), p. 114893. DOI: [10.1016/j.cma.2022.114893](https://doi.org/10.1016/j.cma.2022.114893).
- [113] J. Donea et al. “Arbitrary Lagrangian–Eulerian Methods”. In: *Encyclopedia of Computational Mechanics Second Edition* (2017). ISBN: 0470846992, pp. 1–23. DOI: [10.1002/9781119176817.ecm2009](https://doi.org/10.1002/9781119176817.ecm2009).
- [114] C. Farhat. “CFD-Based Nonlinear Computational Aeroelasticity”. In: *Encyclopedia of Computational Mechanics Second Edition* (2017). ISBN: 9781119176817, pp. 1–21. DOI: [10.1002/9781119176817.ecm2063](https://doi.org/10.1002/9781119176817.ecm2063).
- [115] S. Shahmiri. “A Hybrid ALE-Fixed-Grid Approach for Fluid-Structure Interaction”. In: 22.22 (2014).
- [116] A. J. Barlow et al. “Arbitrary Lagrangian–Eulerian methods for modeling high-speed compressible multimaterial flows”. In: *Journal of Computational Physics* 322 (Oct. 2016). Publisher: Academic Press Inc., pp. 603–665. DOI: [10.1016/j.jcp.2016.07.001](https://doi.org/10.1016/j.jcp.2016.07.001).
- [117] T. J. Hughes, W. K. Liu, and T. K. Zimmermann. “Lagrangian–Eulerian finite element formulation for incompressible viscous flows”. In: *Computer Methods in Applied Mechanics and Engineering* 29.3 (Dec. 1981), pp. 329–349. DOI: [10.1016/0045-7825\(81\)90049-9](https://doi.org/10.1016/0045-7825(81)90049-9).
- [118] W. K. Liu, T. Belytschko, and H. Chang. “An arbitrary lagrangian-eulerian finite element method for path-dependent materials”. In: *Computer Methods in Applied Mechanics and Engineering* 58.2 (Oct. 1986), pp. 227–245. DOI: [10.1016/0045-7825\(86\)90097-6](https://doi.org/10.1016/0045-7825(86)90097-6).
- [119] J. Huétink, P. T. Vreede, and J. van der Lugt. “Progress in mixed Eulerian–Lagrangian finite element simulation of forming processes”. In: *International Journal for Numerical Methods in Engineering* 30.8 (1990), pp. 1441–1457. DOI: [10.1002/nme.1620300808](https://doi.org/10.1002/nme.1620300808).
- [120] H. M. Koh and R. B. Haber. “Elastodynamic Formulation of the Eulerian–Lagrangian Kinematic Description”. In: *Journal of Applied Mechanics* 53.4 (Dec. 1986), pp. 839–845. DOI: [10.1115/1.3171868](https://doi.org/10.1115/1.3171868).

- [121] R. B. Haber. “A mixed eulerian-lagrangian displacement model for large-deformation analysis in solid mechanics”. In: *Computer Methods in Applied Mechanics and Engineering* 43.3 (May 1984), pp. 277–292. DOI: [10.1016/0045-7825\(84\)90068-9](https://doi.org/10.1016/0045-7825(84)90068-9).
- [122] R. B. Haber and B. H. Hariandja. “Computational strategies for nonlinear and fracture mechanics problems. An Eulerian-Lagrangian finite element approach to large deformation frictional contact”. In: *Comput. Struct* 20.1-3 (1985), pp. 193–201.
- [123] H. M. Koh and R. B. Haber. *A Mixed Eulerian-Lagrangian Model for the Analysis of Dynamic Fracture*. Tech. rep. ISSN: 0069-4274. 1986.
- [124] A. Huerta and W. K. Liu. “Viscous flow with large free surface motion”. In: *Computer Methods in Applied Mechanics and Engineering* 69.3 (Aug. 1988), pp. 277–324. DOI: [10.1016/0045-7825\(88\)90044-8](https://doi.org/10.1016/0045-7825(88)90044-8).
- [125] W. J. Gordon and C. A. Hall. “Construction of curvilinear co-ordinate systems and applications to mesh generation”. In: *International Journal for Numerical Methods in Engineering* 7.4 (1973), pp. 461–477. DOI: [10.1002/nme.1620070405](https://doi.org/10.1002/nme.1620070405).
- [126] O. Haber and J. F. Abel. “Discrete transfinite mappings for the description and meshing of three-dimensional surfaces using interactive computer graphics”. In: *International Journal for Numerical Methods in Engineering* 18.1 (Jan. 1982), pp. 41–66. DOI: [10.1002/nme.1620180105](https://doi.org/10.1002/nme.1620180105).
- [127] L.-E. Eriksson. “Practical Three-Dimensional Mesh Generation Using Transfinite Interpolation”. In: *SIAM Journal on Scientific and Statistical Computing* 6.3 (July 1985), pp. 712–741. DOI: [10.1137/0906049](https://doi.org/10.1137/0906049).
- [128] J.-P. Ponthot. “The Use of the Eulerian-Lagrangian FEM with Adaptive Mesh. Applications to Metal Forming Simulation”. English. In: *Proceedings of Complas III, Computational Plasticity. Fundamentals and Applications*. 1992.
- [129] T. Yamada and F. Kikuchi. “An arbitrary Lagrangian-Eulerian finite element method for incompressible hyperelasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 102.2 (Jan. 1993), pp. 149–177. DOI: [10.1016/0045-7825\(93\)90106-8](https://doi.org/10.1016/0045-7825(93)90106-8).
- [130] M. Gadala and J. Wang. “ALE formulation and its application in solid mechanics”. In: *Computer Methods in Applied Mechanics and Engineering* 167.1-2 (Dec. 1998). ISBN: 0045.7825/98, pp. 33–55. DOI: [10.1016/S0045-7825\(98\)00107-8](https://doi.org/10.1016/S0045-7825(98)00107-8).
- [131] M. S. Gadala and J. Wang. “Simulation of metal forming processes with finite element methods”. In: *International Journal for Numerical Methods in Engineering* 44.10 (Apr. 1999). Publisher: John Wiley & Sons, pp. 1397–1428. DOI: [10.1002/\(SICI\)1097-0207\(19990410\)44:10<1397::AID-NME496>3.0.CO;2-3](https://doi.org/10.1002/(SICI)1097-0207(19990410)44:10<1397::AID-NME496>3.0.CO;2-3).
- [132] M. Gadala, M. Movahhedy, and J. Wang. “On the mesh motion for ALE modeling of metal forming processes”. In: *Finite Elements in Analysis and Design* 38.5 (Mar. 2002), pp. 435–459. DOI: [10.1016/S0168-874X\(01\)00080-4](https://doi.org/10.1016/S0168-874X(01)00080-4).
- [133] J.-L. Chenot and F. Bay. “An overview of numerical modelling techniques”. en. In: *Journal of Materials Processing Technology* 80-81 (Aug. 1998), pp. 8–15. DOI: [10.1016/S0924-0136\(98\)00205-2](https://doi.org/10.1016/S0924-0136(98)00205-2).
- [134] A. M. Winslow. “Numerical solution of the quasilinear poisson equation in a nonuniform triangle mesh”. In: *Journal of Computational Physics* 135.2 (1997), pp. 128–138. DOI: <https://doi.org/10.1006/jcph.1997.5698>.
- [135] A. M. Winslow. *Equipotential Zoning of Two-Dimensional Meshes*. Tech. rep. arXiv: Winslow1963. 1963.
- [136] P. Hansbo. “Generalized Laplacian smoothing of unstructured grids”. en. In: *Communications in Numerical Methods in Engineering* 11.5 (May 1995), pp. 455–464. DOI: [10.1002/cnm.1640110510](https://doi.org/10.1002/cnm.1640110510).
- [137] J. Hermansson and P. Hansbo. “A variable diffusion method for mesh smoothing”. In: *Communications in Numerical Methods in Engineering* 19.11 (Nov. 2003), pp. 897–908. DOI: [10.1002/cnm.639](https://doi.org/10.1002/cnm.639).

- [138] R. Löhner and C. Yang. “Improved ALE mesh velocities for moving bodies”. In: *Communications in Numerical Methods in Engineering* 12.10 (Oct. 1996), pp. 599–608. DOI: [10.1002/\(SICI\)1099-0887\(199610\)12:10<599::AID-CNMI1>3.0.CO;2-Q](https://doi.org/10.1002/(SICI)1099-0887(199610)12:10<599::AID-CNMI1>3.0.CO;2-Q).
- [139] A. Masud, M. Bhanabhagwanwala, and R. A. Khurram. “An adaptive mesh rezoning scheme for moving boundary flows and fluid–structure interaction”. In: *Computers and Fluids* 36.1 (Jan. 2007), pp. 77–91. DOI: [10.1016/j.compfluid.2005.07.013](https://doi.org/10.1016/j.compfluid.2005.07.013).
- [140] H. Kanchi and A. Masud. “A 3D adaptive mesh moving scheme”. In: *International Journal for Numerical Methods in Fluids* 54.6-8 (June 2007), pp. 923–944. DOI: [10.1002/flid.1512](https://doi.org/10.1002/flid.1512).
- [141] J. Brackbill and J. Saltzman. “Adaptive zoning for singular problems in two dimensions”. en. In: *Journal of Computational Physics* 46.3 (June 1982), pp. 342–368. DOI: [10.1016/0021-9991\(82\)90020-1](https://doi.org/10.1016/0021-9991(82)90020-1).
- [142] B. T. Helenbrook. “Mesh deformation using the biharmonic operator”. In: *International Journal for Numerical Methods in Engineering* 56.7 (Feb. 2003), pp. 1007–1021. DOI: [10.1002/nme.595](https://doi.org/10.1002/nme.595).
- [143] T. J. Hughes, K. S. Pister, and R. L. Taylor. “Implicit-explicit finite elements in nonlinear transient analysis”. en. In: *Computer Methods in Applied Mechanics and Engineering* 17-18 (Jan. 1979), pp. 159–182. DOI: [10.1016/0045-7825\(79\)90086-0](https://doi.org/10.1016/0045-7825(79)90086-0).
- [144] T. Belytschko and R. Mullen. “Stability of explicit-implicit mesh partitions in time integration”. en. In: *International Journal for Numerical Methods in Engineering* 12.10 (1978), pp. 1575–1586. DOI: [10.1002/nme.1620121008](https://doi.org/10.1002/nme.1620121008).
- [145] S. Ghosh and N. Kikuchi. “An arbitrary Lagrangian-Eulerian finite element method for large deformation analysis of elastic-viscoplastic solids”. In: *Computer Methods in Applied Mechanics and Engineering* 86.2 (Mar. 1991), pp. 127–188. DOI: [10.1016/0045-7825\(91\)90126-Q](https://doi.org/10.1016/0045-7825(91)90126-Q).
- [146] T. J. Hughes and R. L. Taylor. “Unconditionally stable algorithms for quasi-static elasto/viscoplastic finite element analysis”. In: *Computers & Structures* 8.2 (1978), pp. 169–173. DOI: [https://doi.org/10.1016/0045-7949\(78\)90019-6](https://doi.org/10.1016/0045-7949(78)90019-6).
- [147] J. H. Argyris, L. E. Vaz, and K. J. Willam. “Integrated finite-element analysis of coupled thermoviscoplastic problems”. In: *Journal of Thermal Stresses* 4.2 (1981). Publisher: Taylor & Francis tex.eprint: <https://doi.org/10.1080/01495738108909960>, pp. 121–153. DOI: [10.1080/01495738108909960](https://doi.org/10.1080/01495738108909960).
- [148] M. Ortiz and E. P. Popov. “Accuracy and stability of integration algorithms for elastoplastic constitutive relations”. en. In: *International Journal for Numerical Methods in Engineering* 21.9 (Sept. 1985), pp. 1561–1576. DOI: [10.1002/nme.1620210902](https://doi.org/10.1002/nme.1620210902).
- [149] D. J. Benson. “An efficient, accurate, simple ale method for nonlinear finite element programs”. In: *Computer Methods in Applied Mechanics and Engineering* 72.3 (Mar. 1989), pp. 305–350. DOI: [10.1016/0045-7825\(89\)90003-0](https://doi.org/10.1016/0045-7825(89)90003-0).
- [150] D. J. Benson. “Vectorization techniques for explicit arbitrary Lagrangian-Eulerian calculations”. In: *Computer Methods in Applied Mechanics and Engineering* 96.3 (May 1992), pp. 303–328. DOI: [10.1016/0045-7825\(92\)90068-U](https://doi.org/10.1016/0045-7825(92)90068-U).
- [151] D. J. Benson. “Computational methods in Lagrangian and Eulerian hydrocodes”. In: *Computer Methods in Applied Mechanics and Engineering* 99.2-3 (Sept. 1992), pp. 235–394. DOI: [10.1016/0045-7825\(92\)90042-I](https://doi.org/10.1016/0045-7825(92)90042-I).
- [152] Wing Kam Liu et al. “Arbitrary lagrangian-eulerian petrov-galerkin finite elements for nonlinear continua”. In: *Computer Methods in Applied Mechanics and Engineering* 68.3 (June 1988), pp. 259–310. DOI: [10.1016/0045-7825\(88\)90011-4](https://doi.org/10.1016/0045-7825(88)90011-4).
- [153] W. K. Liu et al. “Adaptive ALE finite elements with particular reference to external work rate on frictional interface”. In: *Computer Methods in Applied Mechanics and Engineering* 93.2 (Dec. 1991), pp. 189–216. DOI: [10.1016/0045-7825\(91\)90151-U](https://doi.org/10.1016/0045-7825(91)90151-U).
- [154] J. Donea, S. Giuliani, and J. Halleux. “An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions”. In: *Computer Methods in Applied Mechanics and Engineering* 33.1-3 (Sept. 1982), pp. 689–723. DOI: [10.1016/0045-7825\(82\)90128-1](https://doi.org/10.1016/0045-7825(82)90128-1).

- [155] J. T. Batina. “Implicit flux-split Euler schemes for unsteady aerodynamic analysis involving unstructured dynamic meshes”. In: *AIAA Journal* 29.11 (Nov. 1991), pp. 1836–1843. DOI: [10.2514/3.10808](https://doi.org/10.2514/3.10808).
- [156] J. Y. Trepanier et al. “Unsteady Euler solutions for arbitrarily moving bodies and boundaries”. In: *AIAA Journal* 31.10 (Oct. 1993), pp. 1869–1876. DOI: [10.2514/3.11861](https://doi.org/10.2514/3.11861).
- [157] S. Ghosh and S. Raju. “R-adapted arbitrary lagrangian-eulerian finite-element method in metal-forming simulation”. In: *International Journal for Numerical Methods in Engineering* 39.19 (Oct. 1996). ISBN: 39,32473272, pp. 3247–3272. DOI: [10.1002/\(SICI\)1097-0207\(19961015\)39:19<3247::AID-NME998>3.0.CO;2-1](https://doi.org/10.1002/(SICI)1097-0207(19961015)39:19<3247::AID-NME998>3.0.CO;2-1).
- [158] J. Aymone, E. Bittencourt, and G. Creus. “Simulation of 3D metal-forming using an arbitrary Lagrangian-Eulerian finite element method”. In: *Journal of Materials Processing Technology* 110.2 (Mar. 2001). arXiv: 10.1016/S0924-0136(00)00886-4, pp. 218–232. DOI: [10.1016/S0924-0136\(00\)00886-4](https://doi.org/10.1016/S0924-0136(00)00886-4).
- [159] H. T. Ahn, L. Branets, and G. F. Carey. “Moving boundary simulations with dynamic mesh smoothing”. In: *International Journal for Numerical Methods in Fluids* 64.8 (Nov. 2010), pp. 887–907. DOI: [10.1002/flid.2180](https://doi.org/10.1002/flid.2180).
- [160] A. Huerta et al. “Adaptive finite element strategies based on error assessment”. In: *1097-0207* 46.10 (1999), pp. 1803–1818.
- [161] H. Askes, A. Rodríguez-Ferran, and A. Huerta. “Adaptive analysis of yield line patterns in plates with the arbitrary Lagrangian–Eulerian method”. In: *Computers and Structures* 70.3 (Feb. 1999), pp. 257–271. DOI: [10.1016/S0045-7949\(98\)00188-6](https://doi.org/10.1016/S0045-7949(98)00188-6).
- [162] H. Askes, L. Sluys, and B. de Jong. “Remeshing techniques for r-adaptive and combined h/r-adaptive analysis with application to 2D/3D crack propagation”. In: *Structural Engineering and Mechanics* 12.5 (Nov. 2001), pp. 475–490. DOI: [10.12989/sem.2001.12.5.475](https://doi.org/10.12989/sem.2001.12.5.475).
- [163] H. Askes and A. Rodríguez-Ferran. “A combined rh-adaptive scheme based on domain subdivision. Formulation and linear examples”. In: *International Journal for Numerical Methods in Engineering* 51.3 (May 2001), pp. 253–273. DOI: [10.1002/nme.142](https://doi.org/10.1002/nme.142).
- [164] J. P. Cescutti and J. L. Chenot. “A geometrical continuous remeshing procedure for application to finite element calculation of non-steady state forming processes”. In: *Numerical techniques for engineering analysis and design*. Ed. by G. N. Pande and J. Middleton. Dordrecht: Springer Netherlands, 1987, pp. 677–686.
- [165] P. Knupp, L. G. Margolin, and M. Shashkov. “Reference Jacobian Optimization-Based Rezone Strategies for Arbitrary Lagrangian Eulerian Methods”. en. In: *Journal of Computational Physics* 176.1 (Feb. 2002), pp. 93–128. DOI: [10.1006/jcph.2001.6969](https://doi.org/10.1006/jcph.2001.6969).
- [166] M. Shashkov and K. Patrick. “Optimization-based reference-matrix rezone strategies for arbitrary lagrangian-eulerian methods on unstructured meshes”. In: *Selçuk Journal of Applied Mathematics* 3 (Jan. 2002).
- [167] W. B. a. M. Shashkov. “R-Adaptive Reconnection-based Arbitrary Lagrangian Eulerian Method-R-ReALE”. In: *Journal of Mathematical Study* 48.2 (June 2015). Publisher: Global Science Press, pp. 125–167. DOI: [10.4208/jms.v48n2.15.03](https://doi.org/10.4208/jms.v48n2.15.03).
- [168] A. M. Winslow. “Numerical solution of the quasilinear Poisson equation in a nonuniform triangle mesh”. In: *Journal of computational physics* 1.2 (1966). Publisher: Elsevier, pp. 149–172.
- [169] P. Schreurs, F. Veldpaus, and W. Brekelmans. “Simulation of forming processes, using the arbitrary eulerian-lagrangian formulation”. In: *Computer Methods in Applied Mechanics and Engineering* 58.1 (Oct. 1986), pp. 19–36. DOI: [10.1016/0045-7825\(86\)90076-9](https://doi.org/10.1016/0045-7825(86)90076-9).
- [170] J. Chenot and Bellet M. “The ALE method for the numerical simulation of material forming processes”. In: *Simulation of Materials Processing: Theory, Methods and Applications* (1995), pp. 39–48.
- [171] H Dormohammadi. “Arbitrary Lagrangian – Eulerian method in plasticity of pressure-sensitive material : application to powder forming processes”. In: (2008), pp. 13–38. DOI: [10.1007/s00466-007-0232-4](https://doi.org/10.1007/s00466-007-0232-4).

- [172] T. Belytschko et al. *Nonlinear Finite Elements for Continua and Structures*. 2013.
- [173] F. Casadei. “New ale applications in nonlinear fast-transient solid dynamics”. In: *Engineering Computations* 11.4 (1994), pp. 317–345. DOI: [10.1108/02644409410799317](https://doi.org/10.1108/02644409410799317).
- [174] I. Zreid, R. Behnke, and M. Kaliske. “ALE formulation for thermomechanical inelastic material models applied to tire forming and curing simulations”. In: *Computational Mechanics* 67.6 (2021). Publisher: Springer Berlin Heidelberg, pp. 1543–1557. DOI: [10.1007/s00466-021-02005-5](https://doi.org/10.1007/s00466-021-02005-5).
- [175] I. Wollny and M. Kaliske. “Numerical simulation of pavement structures with inelastic material behaviour under rolling tyres based on an arbitrary Lagrangian Eulerian (ALE) formulation”. en. In: *Road Materials and Pavement Design* 14.1 (Mar. 2013), pp. 71–89. DOI: [10.1080/14680629.2012.735800](https://doi.org/10.1080/14680629.2012.735800).
- [176] H. N. Bayoumi and M. S. Gadala. “A complete finite element treatment for the fully coupled implicit ALE formulation”. en. In: *Computational Mechanics* 33.6 (May 2004), pp. 435–452. DOI: [10.1007/s00466-003-0544-y](https://doi.org/10.1007/s00466-003-0544-y).
- [177] A. Huerta and F. Casadei. “New ALE applications in non-linear fast-transient solid dynamics”. In: *Engineering Computations* 11.4 (Apr. 1994), pp. 317–345. DOI: [10.1108/02644409410799317](https://doi.org/10.1108/02644409410799317).
- [178] A. Rodríguez-Ferran, A. Pérez-Foguet, and A. Huerta. “Arbitrary Lagrangian-Eulerian (ALE) formulation for hyperelastoplasticity”. In: *International Journal for Numerical Methods in Engineering* 53.8 (Mar. 2002), pp. 1831–1851. DOI: [10.1002/nme.362](https://doi.org/10.1002/nme.362).
- [179] M. Kucharik et al. “Hybrid remap for multi-material ALE”. en. In: *Computers & Fluids* 46.1 (July 2011), pp. 293–297. DOI: [10.1016/j.compfluid.2010.08.004](https://doi.org/10.1016/j.compfluid.2010.08.004).
- [180] D. Wang et al. “Large deformation finite element analyses in geotechnical engineering”. In: *Computers and Geotechnics* 65 (Apr. 2015). Publisher: Elsevier Ltd, pp. 104–114. DOI: [10.1016/j.compgeo.2014.12.005](https://doi.org/10.1016/j.compgeo.2014.12.005).
- [181] Y. Hu and M. F. Randolph. “A practical numerical approach for large deformation problems in soil”. In: *International Journal for Numerical and Analytical Methods in Geomechanics* 22.5 (May 1998). Publisher: John Wiley & Sons, pp. 327–350. DOI: [10.1002/\(SICI\)1096-9853\(199805\)22:5<327::AID-NAG920>3.0.CO;2-X](https://doi.org/10.1002/(SICI)1096-9853(199805)22:5<327::AID-NAG920>3.0.CO;2-X).
- [182] D. Wang, M. F. Randolph, and D. J. White. “A dynamic large deformation finite element method based on mesh regeneration”. In: *Computers and Geotechnics* 54 (Oct. 2013), pp. 192–201. DOI: [10.1016/j.compgeo.2013.07.005](https://doi.org/10.1016/j.compgeo.2013.07.005).
- [183] M. Nazem et al. “Arbitrary Lagrangian-Eulerian method for large-strain consolidation problems”. In: *International Journal for Numerical and Analytical Methods in Geomechanics* 32.9 (June 2008), pp. 1023–1050. DOI: [10.1002/nag.657](https://doi.org/10.1002/nag.657).
- [184] M. Nazem, J. P. Carter, and D. W. Airey. “Arbitrary Lagrangian-Eulerian method for non-linear problems of geomechanics”. In: *IOP Conference Series: Materials Science and Engineering* 10.1 (June 2010). Publisher: Institute of Physics Publishing, p. 012074. DOI: [10.1088/1757-899X/10/1/012074](https://doi.org/10.1088/1757-899X/10/1/012074).
- [185] M. Nazem et al. “Dynamic analysis of a smooth penetrometer free-falling into uniform clay”. In: *Géotechnique* 62.10 (Oct. 2012), pp. 893–905. DOI: [10.1680/geot.10.P.055](https://doi.org/10.1680/geot.10.P.055).
- [186] L. Yu et al. “Numerical study of spudcan penetration in loose sand overlying clay”. In: *Computers and Geotechnics* 46 (Nov. 2012), pp. 1–12. DOI: [10.1016/j.compgeo.2012.05.012](https://doi.org/10.1016/j.compgeo.2012.05.012).
- [187] P. Hu et al. “Predicting the resistance profile of a spudcan penetrating sand overlying clay”. In: *Canadian Geotechnical Journal* 51.10 (Oct. 2014). Publisher: Canadian Science Publishing, pp. 1151–1164. DOI: [10.1139/cgj-2013-0374](https://doi.org/10.1139/cgj-2013-0374).
- [188] T. Pucker and J. Grabe. “Numerical simulation of the installation process of full displacement piles”. In: *Computers and Geotechnics* 45 (Sept. 2012), pp. 93–106. DOI: [10.1016/j.compgeo.2012.05.006](https://doi.org/10.1016/j.compgeo.2012.05.006).
- [189] G. Qiu and J. Grabe. “Numerical investigation of bearing capacity due to spudcan penetration in sand overlying clay”. In: *Canadian Geotechnical Journal* 49.12 (Dec. 2012), pp. 1393–1407. DOI: [10.1139/t2012-085](https://doi.org/10.1139/t2012-085).

- [190] R. Sevilla, A. J. Gil, and M. Weberstadt. “A high-order stabilised ALE finite element formulation for the Euler equations on deformable domains”. In: *Computers and Structures* 181 (2017). Publisher: The Authors, pp. 89–102. DOI: [10.1016/j.compstruc.2016.11.019](https://doi.org/10.1016/j.compstruc.2016.11.019).
- [191] A. Dervieux et al. “Total energy conservation in ALE schemes for compressible flows”. In: *European Journal of Computational Mechanics* 19.4 (Jan. 2010). Publisher: Lavoisier, pp. 337–363. DOI: [10.3166/ejcm.19.337-363](https://doi.org/10.3166/ejcm.19.337-363).
- [192] C. Farhat and P. Geuzaine. “Design and analysis of robust ALE time-integrators for the solution of unsteady flow problems on moving grids”. In: *Computer Methods in Applied Mechanics and Engineering* 193.39-41 (Oct. 2004), pp. 4073–4095. DOI: [10.1016/j.cma.2003.09.027](https://doi.org/10.1016/j.cma.2003.09.027).
- [193] F Armero. “Volume-preserving energy – momentum schemes for isochoric multiplicative plasticity”. In: 196 (2007), pp. 4130–4159. DOI: [10.1016/j.cma.2007.04.002](https://doi.org/10.1016/j.cma.2007.04.002).
- [194] A. J. Gil et al. “An enhanced Immersed Structural Potential Method for fluid-structure interaction”. In: *Journal of Computational Physics* 250 (2013). Publisher: Elsevier Inc., pp. 178–205. DOI: [10.1016/j.jcp.2013.05.011](https://doi.org/10.1016/j.jcp.2013.05.011).
- [195] J. Bonet, A. J. Gil, and R. D. Wood. *Nonlinear solid mechanics for finite element analysis: Statics*. Publication Title: Nonlinear Solid Mechanics for Finite Element Analysis: Statics. Cambridge: Cambridge University Press, 2016. DOI: [10.1017/9781316336144](https://doi.org/10.1017/9781316336144).
- [196] J. Bonet, A. J. Gil, and R. Ortigosa. “On a tensor cross product based formulation of large strain solid mechanics”. In: *International Journal of Solids and Structures* 84 (2016). Publisher: Elsevier Ltd, pp. 49–63. DOI: [10.1016/j.ijsolstr.2015.12.030](https://doi.org/10.1016/j.ijsolstr.2015.12.030).
- [197] J. Bonet et al. “A first order hyperbolic framework for large strain computational solid dynamics. Part I: Total Lagrangian isothermal elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 283 (2015). Publisher: Elsevier B.V., pp. 689–732. DOI: [10.1016/j.cma.2014.09.024](https://doi.org/10.1016/j.cma.2014.09.024).
- [198] J. Bonet, H. Marriott, and O. Hassan. “An averaged nodal deformation gradient linear tetrahedral element for large strain explicit dynamic applications”. In: *Communications in Numerical Methods in Engineering* 17.8 (2001), pp. 551–561. DOI: [10.1002/cnm.429](https://doi.org/10.1002/cnm.429).
- [199] R. O. Mart. “On a new variational and computational framework for polyconvex nonlinear continuum mechanics and convex multi-variable nonlinear electro-elasticity”. In: (2016).
- [200] R. Boer. *Vektor- und Tensorrechnung für Ingenieure*. Publication Title: Vektor- und Tensorrechnung für Ingenieure. Springer-Verlag Berlin Heidelberg, 1982. DOI: [10.1007/978-3-642-81901-8](https://doi.org/10.1007/978-3-642-81901-8).
- [201] C. M. Dafermos. “Quasilinear hyperbolic systems with involutions”. In: *Archive for Rational Mechanics and Analysis* 94.4 (1986), pp. 373–389. DOI: [10.1007/BF00280911](https://doi.org/10.1007/BF00280911).
- [202] M. Lesoinne and C. Farhat. “Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations”. In: *Computer Methods in Applied Mechanics and Engineering* 134.1-2 (1996), pp. 71–90. DOI: [10.1016/0045-7825\(96\)01028-6](https://doi.org/10.1016/0045-7825(96)01028-6).
- [203] B. Koobus and C. Farhat. “Second-order time-accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes”. In: *Computer Methods in Applied Mechanics and Engineering* 170.1-2 (Feb. 1999), pp. 103–129. DOI: [10.1016/S0045-7825\(98\)00207-2](https://doi.org/10.1016/S0045-7825(98)00207-2).
- [204] H. Guillard and C. Farhat. “On the significance of the geometric conservation law for flow computations on moving meshes”. In: *Computer Methods in Applied Mechanics and Engineering* 190.11-12 (Dec. 2000), pp. 1467–1482. DOI: [10.1016/S0045-7825\(00\)00173-0](https://doi.org/10.1016/S0045-7825(00)00173-0).
- [205] C. Farhat, P. Geuzaine, and C. Grandmont. “The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids”. In: *Journal of Computational Physics* 174.2 (Dec. 2001). Publisher: Academic Press Inc. Citation Key: pp. 669–694. DOI: [10.1006/jcph.2001.6932](https://doi.org/10.1006/jcph.2001.6932).

- [206] J Bonet and A. J. Burton. “A simple average nodal pressure tetrahedral element for incompressible and nearly incompressible dynamic explicit applications”. In: 449.October 1997 (1998), pp. 437–449. DOI: [10.1002/\(SICI\)1099-0887\(199805\)14:5<437::AID-CNM162>3.0.CO;2-W](https://doi.org/10.1002/(SICI)1099-0887(199805)14:5<437::AID-CNM162>3.0.CO;2-W).
- [207] J. Bonet, H. Marriott, and O. Hassan. “Stability and comparison of different linear tetrahedral formulations for nearly incompressible explicit dynamic applications”. In: *International Journal for Numerical Methods in Engineering* 50.1 (Jan. 2001), pp. 119–133. DOI: [10.1002/1097-0207\(20010110\)50:1<119::AID-NME24>3.0.CO;2-C](https://doi.org/10.1002/1097-0207(20010110)50:1<119::AID-NME24>3.0.CO;2-C).
- [208] M. W. Gee et al. “A uniform nodal strain tetrahedron with isochoric stabilization”. In: *International Journal for Numerical Methods in Engineering* 78.4 (Apr. 2009), pp. 429–443. DOI: [10.1002/nme.2493](https://doi.org/10.1002/nme.2493).
- [209] R. Abedi and R. B. Haber. “Riemann solutions and spacetime discontinuous Galerkin method for linear elastodynamic contact”. In: *Computer Methods in Applied Mechanics and Engineering* 270 (2014). Publisher: Elsevier B.V., pp. 150–177. DOI: [10.1016/j.cma.2013.11.021](https://doi.org/10.1016/j.cma.2013.11.021).
- [210] C Truesdell and W Noll. “The Non-Linear Field Theories of Mechanics / Die Nicht-Linearen Feldtheorien der Mechanik”. In: (2013). ISBN: 3642460151, p. 602.
- [211] R. Hill. “On uniqueness and stability in the theory of finite elastic strain”. In: *Journal of the Mechanics and Physics of Solids* 5.4 (1957), pp. 229–241. DOI: [10.1016/0022-5096\(57\)90016-9](https://doi.org/10.1016/0022-5096(57)90016-9).
- [212] B. D. Coleman and W. Noll. “On the thermostatics of continuous media”. In: *Archive for Rational Mechanics and Analysis* 4.1 (Jan. 1959), pp. 97–128. DOI: [10.1007/BF00281381](https://doi.org/10.1007/BF00281381).
- [213] J. M. Ball. “Convexity conditions and existence theorems in nonlinear elasticity”. In: *Archive for Rational Mechanics and Analysis* 63.4 (Dec. 1976), pp. 337–403. DOI: [10.1007/BF00279992](https://doi.org/10.1007/BF00279992).
- [214] J. a. H. Marsden. “Mathematical foundations of elasticity”. In: *The Mathematical Gazette* (1983). DOI: [10.2307/3617651](https://doi.org/10.2307/3617651).
- [215] P. G. Ciarlet. *Three-dimensional elasticity*. Elsevier, 1988.
- [216] O. Gonzalez and A. M. Stuart. *A First Course in Continuum Mechanics*. Cambridge University Press, Jan. 2001. DOI: [10.1017/CB09780511619571](https://doi.org/10.1017/CB09780511619571).
- [217] J. Bonet, A. J. Gil, and R. D. Wood. *Nonlinear solid mechanics for finite element analysis: Statics*. Publication Title: Nonlinear Solid Mechanics for Finite Element Analysis: Statics Issue: May. 2016. DOI: [10.1017/9781316336144](https://doi.org/10.1017/9781316336144).
- [218] Z. P. Bazant. “A Correlation Study of Formulations of Incremental Deformation and Stability of Continuous Bodies”. In: *Journal of Applied Mechanics* 38.4 (Dec. 1971), pp. 919–928. DOI: [10.1115/1.3408976](https://doi.org/10.1115/1.3408976).
- [219] J. Bonet and S. Kulasegaram. “Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations”. In: *International Journal for Numerical Methods in Engineering* 47.6 (2000), pp. 1189–1214. DOI: [10.1002/\(SICI\)1097-0207\(20000228\)47:6<1189::AID-NME830>3.0.CO;2-I](https://doi.org/10.1002/(SICI)1097-0207(20000228)47:6<1189::AID-NME830>3.0.CO;2-I).
- [220] R. D. Richtmyer and K. Morton. *Difference methods for initial-value problems*. 1994.
- [221] R. J. Martin et al. “Rank-one convexity vs. ellipticity for isotropic functions”. In: (Aug. 2020). arXiv: 2008.11631.
- [222] J. Kristensen. “On the non-locality of quasiconvexity”. In: *Annales de l’Institut Henri Poincaré C, Analyse non linéaire* 16.1 (Feb. 1999), pp. 1–13. DOI: [10.1016/s0294-1449\(99\)80006-7](https://doi.org/10.1016/s0294-1449(99)80006-7).
- [223] J. Sivaloganathan. “Implications of rank one convexity”. In: *Annales de l’Institut Henri Poincaré C, Analyse non linéaire* 5.2 (Apr. 1988), pp. 99–118. DOI: [10.1016/s0294-1449\(16\)30351-1](https://doi.org/10.1016/s0294-1449(16)30351-1).
- [224] E. F. Morton E. Gurtin Lallit Anand. *The Mechanics and Thermodynamics of Continua*. Cambridge University Press, Dec. 2013.

- [225] Y. C. Chen. “On strong ellipticity and the Legendre-Hadamard condition”. In: *Archive for Rational Mechanics and Analysis* 113.2 (1991), pp. 165–175. DOI: [10.1007/BF00380415](https://doi.org/10.1007/BF00380415).
- [226] D. J. Steigmann. “Applications of polyconvexity and strong ellipticity to nonlinear elasticity and elastic plate theory”. In: 2010, pp. 265–299. DOI: [10.1007/978-3-7091-0174-2_8](https://doi.org/10.1007/978-3-7091-0174-2_8).
- [227] C. Hesch et al. “A framework for polyconvex large strain phase-field methods to fracture”. In: *Computer Methods in Applied Mechanics and Engineering* 317 (Apr. 2017). Publisher: Elsevier B.V., pp. 649–683. DOI: [10.1016/j.cma.2016.12.035](https://doi.org/10.1016/j.cma.2016.12.035).
- [228] Z Kewei. “Polyconvexity and Stability of Equilibria in Nonlinear Elasticity”. In: *The Quarterly Journal of Mechanics and Applied Mathematics* 43.2 (1990), pp. 215–221. DOI: [10.1093/qjmam/43.2.215](https://doi.org/10.1093/qjmam/43.2.215).
- [229] I.-D. Ghiba, R. J. Martin, and P. Neff. “Rank-one convexity implies polyconvexity in isotropic planar incompressible elasticity”. en. In: *Journal de Mathématiques Pures et Appliquées* 116 (Aug. 2018), pp. 88–104. DOI: [10.1016/j.matpur.2018.06.009](https://doi.org/10.1016/j.matpur.2018.06.009).
- [230] R. De Borst and H.-B. Muhlhaus. “Gradient-dependent plasticity: Formulation and algorithmic aspects”. In: *International Journal for Numerical Methods in Engineering* 35 (), p. 992.
- [231] S. Hartmann and P. Neff. “Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility”. In: *International Journal of Solids and Structures* 40.11 (2003), pp. 2767–2791. DOI: [10.1016/S0020-7683\(03\)00086-6](https://doi.org/10.1016/S0020-7683(03)00086-6).
- [232] J. Bonet, A. J. Gil, and R. Ortigosa. “A computational framework for polyconvex large strain elasticity”. In: *Computer Methods in Applied Mechanics and Engineering* 283 (2015). Publisher: Elsevier B.V., pp. 1061–1094. DOI: [10.1016/j.cma.2014.10.002](https://doi.org/10.1016/j.cma.2014.10.002).
- [233] M. Franke et al. “An energy momentum consistent integration scheme using a polyconvexity-based framework for nonlinear thermo-elastodynamics”. In: *International Journal for Numerical Methods in Engineering* 115.5 (2018), pp. 549–577. DOI: [10.1002/nme.5816](https://doi.org/10.1002/nme.5816).
- [234] V. A. Lubarda. *Elastoplasticity theory*. CRC Press, 2002.
- [235] F. Capaldi. *Continuum Mechanics: Constitutive Modeling of Structural and Biological Materials*. Cambridge University Press, 2012.
- [236] P. J. Flory. “Thermodynamic relations for high elastic materials”. In: *Transactions of the Faraday Society* 57 (1961), p. 829. DOI: [10.1039/tf9615700829](https://doi.org/10.1039/tf9615700829).
- [237] C. Sansour. “On the physical assumptions underlying the volumetric-isochoric split and the case of anisotropy”. In: *European Journal of Mechanics, A/Solids* 27.1 (Jan. 2008), pp. 28–39. DOI: [10.1016/j.euromechsol.2007.04.001](https://doi.org/10.1016/j.euromechsol.2007.04.001).
- [238] H. Xiao and L. S. Chen. “Hencky’s elasticity model and linear stress-strain relations in isotropic finite hyperelasticity”. In: *Acta Mechanica* 157.1-4 (Mar. 2002), pp. 51–60. DOI: [10.1007/BF01182154](https://doi.org/10.1007/BF01182154).
- [239] J. C. Simo and R. L. Taylor. *Consistent Tangent Operators for Rate-Independent Elastoplasticity*. Tech. rep. Computer Methods in Applied Mechanics and Engineering Volume: 4. 1985, pp. 101–118.
- [240] C. William Gear. *Numerical initial value problems in ordinary differential equations*. Prentice-Hall, 1971.
- [241] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Publication Title: Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. DOI: [10.1007/b79761](https://doi.org/10.1007/b79761).
- [242] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Publication Title: Mathematics of Computation ISSN: 00255718. Birkhäuser Basel, 1990. DOI: [10.2307/2938728](https://doi.org/10.2307/2938728).
- [243] O. Zienkiewicz, R. Taylor, and D. Fox. “Inelastic and Nonlinear Materials”. en. In: *The Finite Element Method for Solid and Structural Mechanics*. Elsevier, 2014, pp. 75–145. DOI: [10.1016/B978-1-85617-634-7.00004-1](https://doi.org/10.1016/B978-1-85617-634-7.00004-1).
- [244] T. Heuzé. *Plasticité des structures*. 2015.
- [245] R. M. Jones. *Deformation theory of plasticity*. Blacksburg, Va, 2009.

- [246] J. Lubliner and B. Moran. “Plasticity Theory”. In: *Journal of Applied Mechanics* 59.1 (Mar. 1992), pp. 245–246. DOI: [10.1115/1.2899459](https://doi.org/10.1115/1.2899459).
- [247] Fionn Dunne and Nik Petrinic. *Introduction to Computational Plasticity*. Oxford University Press, 2005.
- [248] J. C. Simo. “Numerical analysis and simulation of plasticity”. In: *Handbook of Numerical Analysis* 6.Part 3 (1998), pp. 183–499. DOI: [10.1016/S1570-8659\(98\)80009-4](https://doi.org/10.1016/S1570-8659(98)80009-4).
- [249] T. J. R. H. J. C. Simo. *Computational Inelasticity*. Springer-Verlag New York, 1998.
- [250] J. Chaboche. “Constitutive equations for cyclic plasticity and cyclic viscoplasticity”. In: *International Journal of Plasticity* 5.3 (Jan. 1989), pp. 247–302. DOI: [10.1016/0749-6419\(89\)90015-6](https://doi.org/10.1016/0749-6419(89)90015-6).
- [251] E VOCE. “A practical strain hardening function”. In: *Metallurgia* 51 (1955), pp. 219–226.
- [252] D. Aubram. “Notes on rate equations in nonlinear continuum mechanics”. In: (Sept. 2017). arXiv: 1709.10048.
- [253] B. Kolev and R. Desmorat. “Objective rates as covariant derivatives on the manifold of Riemannian metrics”. In: (June 2021). arXiv: 2106.01126.
- [254] G. P. Leborgne. “Virtual Power Principle: A Lie Covariant Approach. Applications to Non-Linear Elasticity, Turbulence, Visco-elasticity”. In: (Aug. 2022). arXiv: 2208.10780.
- [255] J. Donea and A. Huerta. *Introduction and Preliminaries*. Publication Title: Finite Element Methods for Flow Problems. 2005. DOI: [10.1002/0470013826.ch1](https://doi.org/10.1002/0470013826.ch1).
- [256] R. Huang et al. “Direct observation of the full transition from ballistic to diffusive Brownian motion in a liquid”. en. In: *Nature Physics* 7.7 (July 2011), pp. 576–580. DOI: [10.1038/nphys1953](https://doi.org/10.1038/nphys1953).
- [257] J. L. Ericksen. *Introduction to the Thermodynamics of Solids*. en. Ed. by J. E. Marsden and L. Sirovich. Vol. 131. Applied Mathematical Sciences. New York, NY: Springer New York, 1998. DOI: [10.1007/978-1-4612-1614-8](https://doi.org/10.1007/978-1-4612-1614-8).
- [258] G. R. Johnson and W. H. Cook. “A Computational Constitutive Model and Data for Metals Subjected to Large Strain, High Strain Rates and High Pressures”. In: *the Seventh International Symposium on Ballistics* (1983), pp. 541–547.
- [259] A. R. Khoei, M. Anahid, and K. Shahim. “An extended arbitrary Lagrangian-Eulerian finite element method for large deformation of solid mechanics”. In: *Finite Elements in Analysis and Design* 44.6-7 (2008), pp. 401–416. DOI: [10.1016/j.finel.2007.12.005](https://doi.org/10.1016/j.finel.2007.12.005).
- [260] G. R. Johnson and S. R. Beissel. “Damping algorithms and effects for explicit dynamics computations”. In: *International Journal of Impact Engineering* 25.9 (Oct. 2001), pp. 911–925. DOI: [10.1016/S0734-743X\(01\)00019-7](https://doi.org/10.1016/S0734-743X(01)00019-7).
- [261] G. Noh and K.-J. Bathe. “An explicit time integration scheme for the analysis of wave propagations”. In: *Computers and Structures* 129 (Dec. 2013), pp. 178–193. DOI: [10.1016/j.compstruc.2013.06.007](https://doi.org/10.1016/j.compstruc.2013.06.007).
- [262] L. Maheo, V. Grolleau, and G. Rio. “Numerical damping of spurious oscillations: a comparison between the bulk viscosity method and the explicit dissipative Tchamwa–Wielgosz scheme”. In: *Computational Mechanics* 51.1 (Jan. 2013). Publisher: Springer Verlag, pp. 109–128. DOI: [10.1007/s00466-012-0708-8](https://doi.org/10.1007/s00466-012-0708-8).
- [263] T. Barth and D. Jespersen. “The design and application of upwind schemes on unstructured meshes”. en. In: *27th Aerospace Sciences Meeting*. Reno,NV,U.S.A.: American Institute of Aeronautics and Astronautics, Jan. 1989. DOI: [10.2514/6.1989-366](https://doi.org/10.2514/6.1989-366).
- [264] K. A. Sørensen et al. “A multigrid accelerated hybrid unstructured mesh method for 3D compressible turbulent flow”. In: *Computational Mechanics* 31.1-2 SPEC. (2003). Publisher: Springer Verlag, pp. 101–114. DOI: [10.1007/s00466-002-0397-9](https://doi.org/10.1007/s00466-002-0397-9).
- [265] H. K. Versteeg and W Malalasekera. *An Introduction to Computational Fluid Dynamics Second Edition*. Tech. rep. 2007.
- [266] J Blazek. *Computational Fluid Dynamics: Principles and Applications*. Elsevier, 2015. DOI: [10.1016/C2013-0-19038-1](https://doi.org/10.1016/C2013-0-19038-1).

- [267] C. Hirsch. “Finite Volume Method and Conservative Discretization with an Introduction to Finite Element Method”. In: *Numerical Computation of Internal and External Flows M* (2007), pp. 203–248. DOI: [10.1016/b978-075066594-0/50046-1](https://doi.org/10.1016/b978-075066594-0/50046-1).
- [268] P. L. Roe. “Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes”. In: *Journal of Computational Physics* 43 (1981), pp. 357–372.
- [269] P. L. Roe and J Pike. “Efficient construction and utilisation of approximate Riemann solutions”. In: *Computing methods in applied sciences and engineering, VI* September (1984). ISBN: 0-444-87597-2, pp. 499–518.
- [270] P. Roe. “Characteristic-Based Schemes for the Euler Equations”. In: *Annual Review of Fluid Mechanics* 18.1 (1986). ISBN: 0824307186, pp. 337–365. DOI: [10.1146/annurev.fluid.18.1.337](https://doi.org/10.1146/annurev.fluid.18.1.337).
- [271] S. Osher and F. Solomon. “Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws”. In: *Mathematics of Computation* 38.158 (Apr. 1982), p. 339. DOI: [10.2307/2007275](https://doi.org/10.2307/2007275).
- [272] H. Luo, J. D. Baum, and R. Löhner. “An improved finite volume scheme for compressible flows on unstructured grids”. In: *33rd Aerospace Sciences Meeting and Exhibit* (1995). DOI: [10.2514/6.1995-348](https://doi.org/10.2514/6.1995-348).
- [273] C. RHIE and W. CHOW. “A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation”. In: *3rd Joint Thermophysics, Fluids, Plasma and Heat Transfer Conference*. ISSN: 01463705. Reston, Virginia: American Institute of Aeronautics and Astronautics, June 1982. DOI: [10.2514/6.1982-998](https://doi.org/10.2514/6.1982-998).
- [274] I. Demirdžić and S. Muzaferija. “Numerical method for coupled fluid flow, heat transfer and stress analysis using unstructured moving meshes with cells of arbitrary topology”. In: *Computer Methods in Applied Mechanics and Engineering* 125.1-4 (Sept. 1995). arXiv: 10.1016/0045-7825(95)00800-G, pp. 235–255. DOI: [10.1016/0045-7825\(95\)00800-G](https://doi.org/10.1016/0045-7825(95)00800-G).
- [275] A. Nouri-Borujerdi and A. Kebriaee. “Numerical Simulation of Laminar and Turbulent Two-Phase Flow in Pressure-Swirl Atomizers”. In: *AIAA Journal* 50.10 (Oct. 2012), pp. 2091–2101. DOI: [10.2514/1.J051331](https://doi.org/10.2514/1.J051331).
- [276] D. J. Mavriplis. “Unstructured Grid Techniques”. In: *Annual Review of Fluid Mechanics* 29.1 (Jan. 1997), pp. 473–514. DOI: [10.1146/annurev.fluid.29.1.473](https://doi.org/10.1146/annurev.fluid.29.1.473).
- [277] S. K. Godunov and E. I. Romenskii. *Elements of Continuum Mechanics and Conservation Laws*. Publication Title: Elements of Continuum Mechanics and Conservation Laws. 2003. DOI: [10.1007/978-1-4757-5117-8](https://doi.org/10.1007/978-1-4757-5117-8).
- [278] V. Araújo and M. Viana. “Hyperbolic Dynamical Systems”. In: *Mathematics of Complexity and Dynamical Systems*. New York, NY: Springer New York, 2012, pp. 740–754. DOI: [10.1007/978-1-4614-1806-1_45](https://doi.org/10.1007/978-1-4614-1806-1_45).
- [279] C. M. Dafermos. *Hyperbolic Conservation Laws in Continuum Physics*. Series Title: Grundlehren der mathematischen Wissenschaften. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016. DOI: [10.1007/978-3-662-49451-6](https://doi.org/10.1007/978-3-662-49451-6).
- [280] H. Friedrich. “On the hyperbolicity of Einstein’s and other gauge field equations”. In: *Communications in Mathematical Physics* 100.4 (Dec. 1985). arXiv: 1104114004, pp. 525–543. DOI: [10.1007/BF01217728](https://doi.org/10.1007/BF01217728).
- [281] S. Frittelli and O. A. Reula. “First-order symmetric-hyperbolic Einstein equations with arbitrary fixed gauge”. In: *Physical Review Letters* (May 1996). arXiv: gr-qc/9605005. DOI: [10.1103/PhysRevLett.76.4667](https://doi.org/10.1103/PhysRevLett.76.4667).
- [282] R. Beig. “Concepts of Hyperbolicity and Relativistic Continuum Mechanics”. In: *Analytical and Numerical Approaches to Mathematical Relativity*. Berlin/Heidelberg: Springer-Verlag, pp. 101–116. DOI: [10.1007/3-540-33484-X_5](https://doi.org/10.1007/3-540-33484-X_5).
- [283] R. Codina. “Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods”. In: *Computer Methods in Applied Mechanics and Engineering* 190.13-14 (Dec. 2000), pp. 1579–1599. DOI: [10.1016/S0045-7825\(00\)00254-1](https://doi.org/10.1016/S0045-7825(00)00254-1).

- [284] G. Kluth and B. Després. “Discretization of hyperelasticity on unstructured mesh with a cell-centered Lagrangian scheme”. In: *Journal of Computational Physics* 229.24 (Dec. 2010). Publisher: Academic Press Inc., pp. 9092–9118. DOI: [10.1016/j.jcp.2010.08.024](https://doi.org/10.1016/j.jcp.2010.08.024).
- [285] C. H. Lee, A. J. Gil, and J. Bonet. “Development of a cell centred upwind finite volume algorithm for a new conservation law formulation in structural dynamics”. In: *Computers and Structures* 118 (2013), pp. 13–38. DOI: [10.1016/j.compstruc.2012.12.008](https://doi.org/10.1016/j.compstruc.2012.12.008).
- [286] P. K. Sweby. “High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws”. en. In: *SIAM Journal on Numerical Analysis* 21.5 (Oct. 1984), pp. 995–1011. DOI: [10.1137/0721062](https://doi.org/10.1137/0721062).
- [287] B. van Leer and C.-H. Tait. “Design of optimally smoothing multi-stage schemes for the Euler equations”. en. In: (1993). DOI: [10.2514/6.1989-1933](https://doi.org/10.2514/6.1989-1933).
- [288] W Kutta. “Beitrag zur näherungsweise Integration totaler Differentialgleichungen”. In: *Zeit. Math. Phys.* 46 (1901), pp. 435–453.
- [289] C. Runge. “Ueber die numerische Auflösung von Differentialgleichungen”. In: *Mathematische Annalen* 46.2 (1895), pp. 167–178. DOI: [10.1007/BF01446807](https://doi.org/10.1007/BF01446807).
- [290] B. Engquist and S. Osher. “One-Sided Difference Approximations for Nonlinear Conservation Laws”. In: *Mathematics of Computation* 36.154 (1981), p. 321. DOI: [10.2307/2007646](https://doi.org/10.2307/2007646).
- [291] P. O. Persson, J. Bonet, and J. Peraire. “Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains”. In: *Computer Methods in Applied Mechanics and Engineering* 198.17-20 (2009). Publisher: Elsevier B.V. ISBN: 1563478900, pp. 1585–1595. DOI: [10.1016/j.cma.2009.01.012](https://doi.org/10.1016/j.cma.2009.01.012).
- [292] H. Jasak. “Dynamic mesh handling in OpenFOAM”. In: *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition* (2009). ISBN: 9781563479694, pp. 1–10. DOI: [10.2514/6.2009-341](https://doi.org/10.2514/6.2009-341).
- [293] E. Gamma et al. *Design patterns: elements of reusable object-oriented software*. Pearson Deutschland GmbH, 1995.
- [294] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. en. Second. Society for Industrial and Applied Mathematics, Jan. 2003. DOI: [10.1137/1.9780898718003](https://doi.org/10.1137/1.9780898718003).
- [295] J. Schröder, P. Wriggers, and D. Balzani. “A new mixed finite element based on different approximations of the minors of deformation tensors”. In: *Computer Methods in Applied Mechanics and Engineering* 200.49-52 (Dec. 2011), pp. 3583–3600. DOI: [10.1016/j.cma.2011.08.009](https://doi.org/10.1016/j.cma.2011.08.009).
- [296] J Peraire et al. “A Discontinuous Galerkin Formulation for Lagrangian Dynamic Analysis of Hyperelastic Materials”. In: *VII World Congress of computational Mechanics, Los Angeles*. 2006.
- [297] G. Taylor and P. R. S. L. A. “The use of flat-ended projectiles for determining dynamic yield stress I. Theoretical considerations”. In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 194.1038 (1948), pp. 289–299. DOI: [10.1098/rspa.1948.0081](https://doi.org/10.1098/rspa.1948.0081).
- [298] F. Casadei. “New ale applications in nonlinear fast-transient solid dynamics”. In: *Engineering Computations* 11.4 (1994), pp. 317–345. DOI: [10.1108/02644409410799317](https://doi.org/10.1108/02644409410799317).
- [299] D. E. Burton et al. “A cell-centered Lagrangian Godunov-like method for solid dynamics”. In: *Computers and Fluids* 83 (2013). Publisher: Elsevier Ltd, pp. 33–47. DOI: [10.1016/j.compfluid.2012.09.008](https://doi.org/10.1016/j.compfluid.2012.09.008).
- [300] J. C. Simo and F. Armero. “Geometrically non-linear enhanced strain mixed methods and the method of incompatible modes”. In: *International Journal for Numerical Methods in Engineering* 33.7 (1992), pp. 1413–1449. DOI: [10.1002/nme.1620330705](https://doi.org/10.1002/nme.1620330705).
- [301] A. Kamoulakos. “A simple benchmark for impact”. In: *Benchmark* February (1990), pp. 31–35. DOI: [10.13140/RG.2.2.30118.55364](https://doi.org/10.13140/RG.2.2.30118.55364).
- [302] R. Whirley and J. Hallquist. “DYNA-3D User Manual”. In: Technical Report UCRL-MA-107254 (1991).

- [303] J. C. Simo. “Algorithms for static and dynamic multiplicative plasticity that preserve the classical return mapping schemes of the infinitesimal theory”. In: 99 (1992), pp. 61–112. DOI: [10.1016/0045-7825\(92\)90123-2](https://doi.org/10.1016/0045-7825(92)90123-2).
- [304] O. C. Zienkiewicz et al. “Triangles and Tetrahedra in Explicit Dynamic Codes for Solids”. In: 583. January (1998), pp. 565–583. DOI: [10.1002/\(SICI\)1097-0207\(19981015\)43:3<565::AID-NME454>3.0.CO;2-9](https://doi.org/10.1002/(SICI)1097-0207(19981015)43:3<565::AID-NME454>3.0.CO;2-9).
- [305] M. Schiebl and P. Betsch. “Structure-preserving space-time discretization of large-strain thermo-viscoelasticity in the framework of GENERIC”. In: *International Journal for Numerical Methods in Engineering* 122.14 (2021), pp. 3448–3488. DOI: [10.1002/nme.6670](https://doi.org/10.1002/nme.6670).
- [306] P. Betsch and M. Schiebl. “Energy-momentum-entropy consistent numerical methods for large-strain thermoelasticity relying on the GENERIC formalism”. In: *International Journal for Numerical Methods in Engineering* 119.12 (2019). Publisher: John Wiley and Sons Ltd, pp. 1216–1244. DOI: [10.1002/nme.6089](https://doi.org/10.1002/nme.6089).
- [307] M. S. Choucino. “3D Simulation of Magneto-Mechanical Coupling in MRI Scanners Using High Order FEM and POD”. en. In: ().
- [308] A. Huerta. *Progress in Arbitrary Lagrangian Eulerian Analysis of Fluid and Solid Problems*. Tech. rep. Stuttgart: Fluid Structures Interface SMiRT 12 - Stuttgart, 1993.
- [309] H. Jasak and Z. Tukovic. “Automatic mesh motion for the unstructured Finite Volume Method”. In: *Transactions of Famena* 30.2 (2006), pp. 1–20.
- [310] G. Chourdakis. “Computational Science and Engineering (International Master ’ s Program) Master ’ s Thesis A general OpenFOAM adapter for the coupling library preCICE Gerasimos Chourdakis”. In: October (2017).
- [311] H. J. Bungartz et al. “preCICE – A fully parallel library for multi-physics surface coupling”. In: *Computers and Fluids* 141 (2016). Publisher: Elsevier Ltd, pp. 250–258. DOI: [10.1016/j.compfluid.2016.04.003](https://doi.org/10.1016/j.compfluid.2016.04.003).
- [312] B. Uekermann. “Partitioned Fluid-Structure Interaction on Massively Parallel Systems”. In: *Dissertation* (2016).
- [313] A. Henthaler, O. Röhrle, and D. Nordsletten. “Validation of a non-conforming monolithic fluid-structure interaction method using phase-contrast MRI”. In: *International Journal for Numerical Methods in Biomedical Engineering* 33.8 (2017), pp. 1–35. DOI: [10.1002/cnm.2845](https://doi.org/10.1002/cnm.2845).
- [314] Tomislav Maric. “Software Design Patterns in Research Software with examples from OpenFOAM”. en. In: (2022).
- [315] Anthony James Merrill Spencer. *Continuum Mechanics*. Dover: Courier Corporation, 2004.
- [316] G. T. Mase. *Continuum Mechanics for Engineers*. Publication Title: New York. 1999.
- [317] S. Rossi, N. Abboud, and G. Scovazzi. “Implicit finite incompressible elastodynamics with linear finite elements: A stabilized method in rate form”. In: *Computer Methods in Applied Mechanics and Engineering* 311 (2016). Publisher: Elsevier Ltd, pp. 208–249. DOI: [10.1016/j.cma.2016.07.015](https://doi.org/10.1016/j.cma.2016.07.015).
- [318] J. Simo and R. Taylor. “Penalty function formulations for incompressible nonlinear elastostatics”. In: *Computer Methods in Applied Mechanics and Engineering* 35.1 (Oct. 1982), pp. 107–118. DOI: [10.1016/0045-7825\(82\)90035-4](https://doi.org/10.1016/0045-7825(82)90035-4).
- [319] J. C. Simo and R. L. Taylor. “Quasi-incompressible finite elasticity in principal stretches. continuum basis and numerical algorithms”. In: *Computer Methods in Applied Mechanics and Engineering* 85.3 (Feb. 1991), pp. 273–310. DOI: [10.1016/0045-7825\(91\)90100-K](https://doi.org/10.1016/0045-7825(91)90100-K).
- [320] C. Miehe. “Aspects of the formulation and finite element implementation of large strain isotropic elasticity”. In: *International Journal for Numerical Methods in Engineering* 37.12 (June 1994), pp. 1981–2004. DOI: [10.1002/nme.1620371202](https://doi.org/10.1002/nme.1620371202).
- [321] C. H. Liu, G. Hofstetter, and H. A. Mang. “3D Finite element analysis of rubber-like materials at finite strains”. In: *Engineering Computations* 11.2 (Feb. 1994), pp. 111–128. DOI: [10.1108/02644409410799236](https://doi.org/10.1108/02644409410799236).

-
- [322] Kelly. “Solid mechanics part III: Foundations of Continuum Mechanics. Solid mechanics lecture notes”. In: 2013.
- [323] M. A. Sánchez et al. “Symplectic Hamiltonian finite element methods for linear elastodynamics”. en. In: *Computer Methods in Applied Mechanics and Engineering* 381 (Aug. 2021), p. 113843. DOI: [10.1016/j.cma.2021.113843](https://doi.org/10.1016/j.cma.2021.113843).
- [324] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*. en. Springer series in computational mathematics 31. Berlin: Springer, 2006.
- [325] C. Kane et al. “Variational integrators and the Newmark algorithm for conservative and dissipative mechanical systems”. en. In: *International Journal for Numerical Methods in Engineering* 49.10 (Dec. 2000), pp. 1295–1325. DOI: [10.1002/1097-0207\(20001210\)49:10<1295::AID-NME993>3.0.CO;2-W](https://doi.org/10.1002/1097-0207(20001210)49:10<1295::AID-NME993>3.0.CO;2-W).