LA-UR-12-23177
Approved for public release; distribution is unlimited.

Title: Mimetic Theory for Cell-Centered Lagrangian Finite Volume Formulation on General Unstructured Grids

Author(s): Sambasivan, Shiv Kumar
Shashkov, Mikhail J.
Burton, Donald E.
Christon, Mark A.

Intended for: Report

Mimetic Theory for Cell-Centered Lagrangian Finite Volume Formulation on General Unstructured Grids☆

Shiv Kumar Sambasivan[a,*], Mikhail J Shashkov[b], Donald E Burton[b], Mark A Christon[a]

[a]*Computer, Computational and Statistical Sciences Division (CCS-2), Los Alamos National Laboratory, Los Alamos, NM 87545*
[b]*X Computational Physics Group (XCP4), Los Alamos National Laboratory, Los Alamos, NM 87545*

Abstract

A finite volume cell-centered Lagrangian scheme for solving large deformation problems is constructed based on the hypo-elastic model and using the mimetic theory. Rigorous analysis in the context of gas and solid dynamics, and arbitrary polygonal meshes, is presented to demonstrate the ability of cell-centered schemes in mimicking the continuum properties and principles at the discrete level. A new mimetic formulation based gradient evaluation technique and physics-based, frame independent and symmetry preserving slope limiters are proposed. Furthermore, a physically consistent dissipation model is employed which is both robust and inexpensive to implement. The cell-centered scheme along with these additional new features are applied to solve solids undergoing elasto-plastic deformation.

1. Introduction

Tensor calculus and theory are ubiquitous in mathematical physics, science and engineering. They constitute the fundamental building blocks of governing conservation laws and many constitutive theories in continuum mechanics. The divergence, gradient and curl operators together with the tensor/vector calculus theorems and identities lie at the heart of numerous partial differential equations. Success of numerical solution to continuum principles relies on the ability to replicate the continuum properties accurately in the discrete space. A numerical scheme is considered robust and can infer realistic predictions only when it is constructed from solid mathematical theory. Such numerical schemes also preserve important properties of continuum principle at the discrete level [1]. For instance, a few of the continuum properties that are considered indispensable, in addition to several others, include the fundamental conservation laws, the underlying symmetries in the solution, symmetry of the stress tensor (a consequence of angular momentum conservation), preservation of divergence free conditions for solenoidal vectors, curl identities, etc [1]. Numerical schemes that can faithfully "mimic" such continuum properties and principles in the discrete space are termed as mimetic (compatible) methods [1, 2].

Discretization schemes are exact provided they are mimetic in formulation. Errors in mimetic formulations arise from the approximations introduced in the constitutive relations and not from the discretization technique [3]. The chief task in developing such exact discretization strategies is to formulate compatible or mimetic algebraic models that yield stable, accurate, and physically consistent approximate solutions. Existing Finite-Volume (FV), Finite-Difference (FD) and Finite-Element (FE) formulations take different paths in achieving this goal by suitably approximating the underlying divergence, gradient and curl operators. Compatibility between the discrete and continuum space for these methods have been demonstrated by exploiting the staggering of the solution space [1, 2]. This staggering of the solution space in turn restricts the placement and evolution of the field variables on the computational mesh, *viz* the dynamical quantities at the vertices of the mesh and the thermodynamical quantities at the centroid of the mesh elements. The staggering of field variables is also a common trait to FE, FV and even several FD schemes.

In spite of their sound justification, staggered formulations have found mixed success for gas dynamics and solid dynamics applications, that are of particular interest to this work. The drawback with the staggered schemes arise

☆The Los Alamos Unlimited Release number is: LA-UR XX-XXXXX
*Corresponding Author: shiv@lanl.gov

from the approximations introduced in the constitutive relations. Due to the staggering in variables, such formulations must be augmented with ad-hoc viscosity models to ensure dissipation of kinetic to internal energy [4, 5, 6, 7], techniques for damping spurious modes that would otherwise lead to mesh instabilities [8, 9], special methods for preserving symmetry in the solution [10], etc. Even the staggered grid based compatible formulations [11, 12, 13, 14] are not devoid of such issues. Alternative to staggered schemes is the cell-centered formulation, wherein all quantities are stored and evolved at the centroid of the mesh elements. After a hiatus for several decades, the cell-centered Lagrangian finite volume hydrodynamic schemes have gained renewed interest due to the novel formulations advocated in [15, 16, 17, 18, 19, 20]. Past efforts in developing cell-centered schemes [21] suffered from premature mesh entanglement issues that emanated from ad-hoc procedures utilized for extracting the vertex velocities (which in turn are required to move the mesh). In contrast, the construction of nodal forces and velocities that are consistent with Geometric Conservation Law (GCL) [22, 23] and compatible with both momentum and total energy conservation [11, 12], have made the recent formulations as promising alternatives. These cell-centered schemes do not require special techniques for evaluating vertex velocities. The vertex velocity field is now part of the solution vector. Furthermore, in contrast to staggered grid approaches, these schemes are inherently stable and robust by construction, without requiring artificial tweaking and stabilizing parameters. In addition, the new cell-centered schemes are devoid of spurious vorticity generation and mesh imprinting issues.

Past and present cell-centered Lagrangian FV schemes have been formulated based on Green's integral theorem [24, 25]. As a result, the mimetic properties for these formulations are difficult to establish. Furthermore, mimetic principles postulated in [1, 2, 14, 26] may not be directly applied as the solution space is not staggered. Therefore, in order for the cell-centered schemes to be established as standard and viable choice, it is important to ascertain their compatible or mimetic properties. This motivates the current effort.

In this work, a finite volume cell-centered Lagrangian scheme for solving large deformation problems is constructed, based on the hypo-elastic model and using the mimetic theory. Some of the contributions that are exclusive to this work are:

- Rigorous analysis corroborating the mimetic properties of cell-centered formulation - A cardinal aspect of this work is to demonstrate/prove, via comprehensive analysis, the compatible properties associated with cell-centered schemes. To this effect, the support operator method [27] is employed to derive the discrete version of the continuum governing equations and constitutive theories. The analysis presented in this work holds valid for both gas and solid dynamics, and arbitrary polygonal meshes.

- Unifying formulation for cell-centered schemes augmented with physically consistent dissipation model - Existing cell-centered schemes [15, 16, 17, 19, 20] differ in the quadrature rule employed for approximating the divergence, gradient and curl operators. As a result, these schemes employ different viscosity model for evaluating the field variables at the quadrature points for integration. In this work, a generic formulation is presented that can be used to retrieve different flavors of cell-centered schemes by simply varying the dissipation model. Furthermore, in contrast to the authors' previous work [28, 29], the dissipation model employed is physically consistent and inexpensive to implement [19].

- New mimetic formulation for evaluating gradients required for high-order reconstruction of field variables - In accordance to the recurring theme of this work, a new gradient estimation technique based on mimetic formulation is proposed. It will be shown in the results section that the gradient evaluation technique is both robust and superior to existing and established techniques such as least squares [30] and Green's integral theorem [25].

- Physics-inspired slope limiter scheme to ensure monotonicity of the reconstructed variable - A physically consistent formulation for frame independent and symmetry preserving slope limiter scheme is proposed. A slope limiter scheme constructed from the second invariant of stress tensor [31] is extended for constraining reconstructed vectors. In the case of velocity vectors, slope limiters are extracted by limiting the specific kinetic energy of the reconstructed vector field.

- Calculations on arbitrary and rough polygonal meshes in addition to extensive comparisons with representative Eulerian and Lagrangian hydrocodes - Computations on meshes comprising of polygonal elements with arbitrary edges and quality are presented for solids undergoing large deformations. The results from these calculations are compared with representative and established Eulerian and Lagrangian hydrocodes.

In essence, the following is performed for capturing the response of solids undergoing large deformation. Since solid materials can sustain significant shear deformation, evolution equations for stress and strain fields are solved in addition to mass, momentum and energy conservation laws. In order to evolve the momentum equation, an area weighted [10] formulation of the discrete divergence and gradient operators for the stress and velocity gradient tensors are employed. Area weighted formulations have long been used for problems where it is desired that perfect one-dimensional spherical symmetry be preserved as a possible limiting case in two-dimensional cylindrical geometry [10, 12]. To evolve the total energy equation, a control volume formulation [10, 12] of the discrete divergence operator for the stress tensor is utilized. Due to the later choice, compatibility with the total energy conservation is established with errors incurred in the momentum conservation. On the solid modeling side, the total strain-rate realized in the material is split into the elastic and plastic response. The elastic and plastic components in turn are modeled using the hypo-elastic theory. In accordance with the hypo-elastic model, a predictor-corrector algorithm is employed for evolving the deviatoric component of the stress tensor. A trial elastic deviatoric stress state is obtained by integrating a rate equation, cast in an objective derivative form, based on Hooke's law [32]. The dilatational response of the material is modeled using an equation of state of the Mie-Grüneisen form. The plastic deformation is accounted for using an iterative radial return algorithm [33, 34] constructed from the J_2 von Mises yield condition.

2. Constitutive Relations and Governing Equations

2.1. Governing Equations in Differential Form

The differential form of the governing equations cast in the Lagrangian frame of reference and written in semi-Lagrangian coordinates, take the following form:

$$\frac{dm}{dt} = 0 \tag{1}$$

$$\rho\frac{d\boldsymbol{u}}{dt} - \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} = 0 \tag{2}$$

$$\rho\frac{dE}{dt} - \boldsymbol{\nabla}\cdot(\boldsymbol{\sigma}\boldsymbol{u}) = 0 \tag{3}$$

$$\frac{1}{V}\frac{dV}{dt} - \boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 \tag{4}$$

where m is the mass of the material, ρ is the material density, V is the material volume, $\boldsymbol{\sigma} = \boldsymbol{S} - P\boldsymbol{I}$ is the symmetric Cauchy stress tensor with $\boldsymbol{S}$ & P denoting the symmetric deviatoric stress tensor and pressure respectively. $E = e + \frac{1}{2}\boldsymbol{u}\cdot\boldsymbol{u}$ is the specific total energy with e defined as the specific internal energy and $\boldsymbol{u}$ is the velocity vector. Equations (1), (2) & (3) constitute the mass, momentum and energy conservation laws and Eq (4) represents the volumetric evolution equation. In combination with the kinematic equation

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{u}(\boldsymbol{x}(t), t), \boldsymbol{x}(0) = \boldsymbol{x}, \tag{5}$$

the GCL ensures the compatibility between the volume evolution equation (Eq (4)) and the volume computed from geometry [23, 35].

Assuming Y to be the axis of (rotational) symmetry, both planar (XY) and axisymmetric (RZ) geometries can be succinctly represented in a pseudo-Cartesian frame by introducing a pseudo-radius $R(x)$ [21]

$$R(x) = 1 - \beta + \beta x \tag{6}$$

where $\beta = 1$ for cylindrical geometry and $\beta = 0$ otherwise (Figure 1). Therefore, the divergence operator for the

velocity vector $\boldsymbol{u}$ and stress tensor $\boldsymbol{\sigma}$ in the governing equations reads as

$$\boldsymbol{\nabla}\cdot\boldsymbol{u} = \frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\beta\frac{u}{R} \tag{7}$$

$$= \frac{1}{R}(\frac{\partial Ru}{\partial x}+\frac{\partial Rv}{\partial y}) \tag{8}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} = \begin{pmatrix} \frac{\partial\sigma_{xx}}{dx}+\frac{\partial\sigma_{xy}}{dy} \\ \frac{\partial\sigma_{xy}}{dx}+\frac{\partial\sigma_{yy}}{dy} \end{pmatrix} + \frac{\beta}{R}\begin{pmatrix} \sigma_{xx}-\sigma_{zz} \\ \sigma_{xy} \end{pmatrix} \tag{9}$$

$$= \frac{1}{R}\begin{pmatrix} \frac{\partial(R\sigma_{xx})}{dx}+\frac{\partial(R\sigma_{xy})}{dy} \\ \frac{\partial(R\sigma_{xy})}{dx}+\frac{\partial(R\sigma_{yy})}{dy} \end{pmatrix} - \frac{\beta}{R}\begin{pmatrix} \sigma_{zz} \\ 0 \end{pmatrix} \tag{10}$$

Equations (7) and (9) and Eqs (8) and (10) correspond to the form of the divergence operator employed in what is called the area-weighted and control volume representations of the integral form of the governing equations, respectively. In combination with the vector identity

$$\boldsymbol{\nabla}\cdot(\boldsymbol{\sigma u}) = \boldsymbol{u}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{\sigma})+\boldsymbol{\sigma}:\boldsymbol{\nabla u} \tag{11}$$

and using Eqs (8) and (10), the following expression is obtained for the divergence of stress power:

$$\boldsymbol{\nabla}\cdot\boldsymbol{(\sigma u)} = \frac{1}{R}\left((\frac{d}{dx}(Ru\sigma_{xx}+Rv\sigma_{xy})+\frac{d}{dy}(Ru\sigma_{xy}+Rv\sigma_{yy}))\right) \tag{12}$$

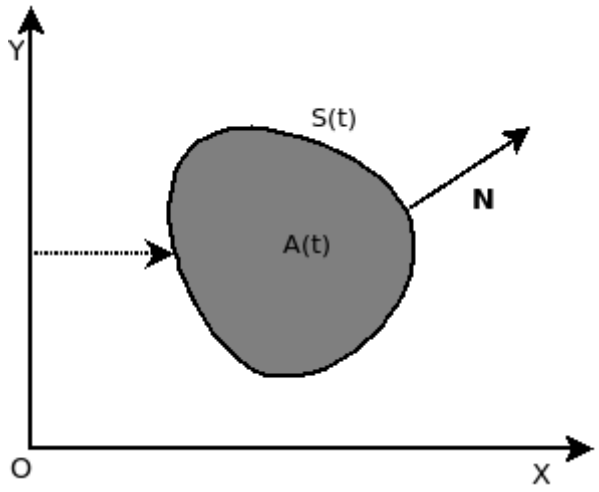


Figure 1: Cartesian frame (O,x,y) with y denoting the the axis of revolution

2.2. Constitutive Equations

Following the conventional theory of hypo-elasticity, an additive decomposition of strain-rate rule is invoked to model the strain-rate tensor:

$$\boldsymbol{D} = \boldsymbol{D}^{E}+\boldsymbol{D}^{PL} \tag{13}$$

where $\boldsymbol{D}$ is the strain-rate tensor given as,

$$\boldsymbol{D} = \frac{1}{2}(\boldsymbol{\nabla u}+\boldsymbol{\nabla u}^{T}) \tag{14}$$

and $\boldsymbol{D}^{E}$ and $\boldsymbol{D}^{PL}$ are the elastic and plastic strain-rate tensors respectively. Assuming incompressibility of the plastic flow ($tr(\boldsymbol{D}^{PL})=0$), the volumetric or dilatational response is governed by an equation of state while the deviatoric

response obeys a conventional flow theory of plasticity [36]. Hence, the total stress tensor $\boldsymbol{\sigma}$ in the continua can be expressed as

$$\boldsymbol{\sigma} = \boldsymbol{S} - P\boldsymbol{I} \tag{15}$$

where $\boldsymbol{S}$ is the traceless deviatoric component, $P\boldsymbol{I}$ is the isotropic, dilatational component taken to be positive in compression. Using Eq (13), the rate of change of deviatoric stress tensor can be modeled using the hypo-elastic stress-strain relation (in the limit of small deformation):

$$\overset{\triangledown}{\boldsymbol{S}} = 2G(\bar{\boldsymbol{D}} - \boldsymbol{D}^{PL}) \tag{16}$$

where G is the shear modulus and $\overset{\triangledown}{\boldsymbol{S}}$ is the Jaumann derivative [37]

$$\overset{\triangledown}{\boldsymbol{S}} = \dot{\boldsymbol{S}} + \boldsymbol{S}\boldsymbol{\Omega} - \boldsymbol{\Omega}\boldsymbol{S} \tag{17}$$

and $\boldsymbol{\Omega}$ is the spin tensor. The Jaumann derivative is used to ensure objectivity of the stress tensor with respect to rotation. The spin tensor used in Eq (17) is given by:

$$\boldsymbol{\Omega} = \frac{1}{2}(\boldsymbol{\nabla u} - \boldsymbol{\nabla u}^T) \tag{18}$$

The deviatoric strain-rate component $\bar{\boldsymbol{D}}$ in Eq (16) is given by:

$$\bar{\boldsymbol{D}} = \boldsymbol{D} - \frac{1}{3}(\nabla \cdot \boldsymbol{U})\boldsymbol{I} \tag{19}$$

2.2.1. Predictor-Corrector Algorithm and the Plasticity Model

In accordance with the hypo-elastic theory, the stress tensor is evolved by a predictor-corrector algorithm. A predicted trial elastic state of stress $\boldsymbol{S}_{tr}$ is determined by assuming a pure elastic deformation *i.e.* by setting $\boldsymbol{D}^{PL} = 0$ in (Eq (16)), to obtain:

$$\dot{\boldsymbol{S}}_{tr} + \boldsymbol{S}_{tr}\boldsymbol{\Omega} - \boldsymbol{\Omega}\boldsymbol{S}_{tr} = 2G\bar{\boldsymbol{D}} \tag{20}$$

The final state of stress is determined by accounting for plastic deformation via a radial return algorithm (section Appendix B). The isochoric plastic deformation ($\bar{\boldsymbol{D}}^{PL} = \boldsymbol{D}^{PL}$) in Eq (13) is modeled assuming a coaxial flow theory (Druckers' postulate) for strain hardening materials [32]:

$$\boldsymbol{D}^{PL} = \dot{\Lambda}\boldsymbol{N} \tag{21}$$

where $\boldsymbol{N} = \frac{\boldsymbol{S}}{\sqrt{\boldsymbol{S}:\boldsymbol{S}}}$ is the unit outward normal to the yield surface and $\dot{\Lambda}$ is a proportional positive scalar factor called the consistency parameter [36]. The consistency parameter $\dot{\Lambda}$ is determined using the J_2 von Mises yield condition. The effective stress (S_{eff}) and the effective plastic strain-rate ($\dot{\bar{\epsilon}}_p$) are given by:

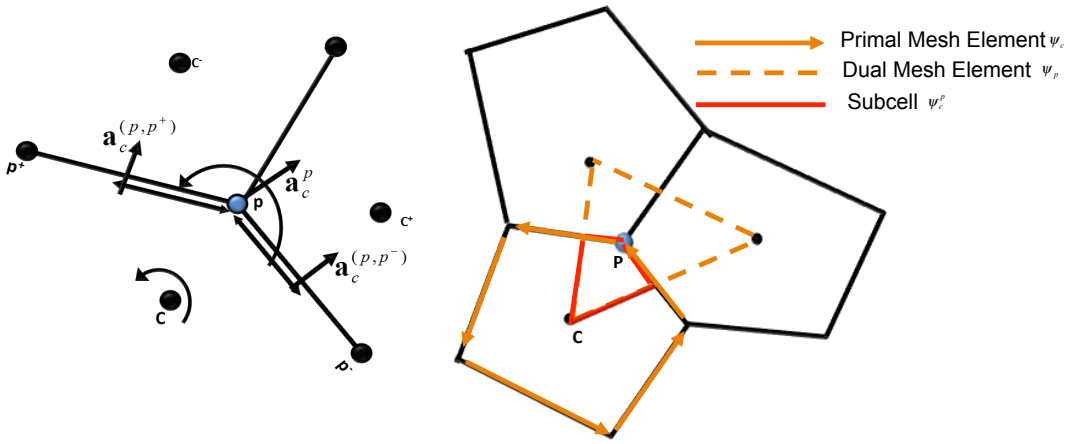
$$S_{eff}^2 = \frac{3}{2}\boldsymbol{S}:\boldsymbol{S} \tag{22}$$

$$(\dot{\bar{\epsilon}}_p)^2 = \frac{2}{3}\boldsymbol{D}^{PL}:\boldsymbol{D}^{PL} \tag{23}$$

$$= \frac{2}{3}\dot{\Lambda}^2 \tag{24}$$

Details on the constitutive relations, material models and the return mapping algorithm employed in this work, in addition to the equation of state for computing the dilatational response of the material, can be found in [38, 39, 40, 36]. For the sake of the readers' convenience, a concise discussion on these topics are presented in the appendix of this paper.

3. Notations



(a) Notations (b) Primal, Dual and Subcell Elements

Figure 2: (a) Figure showing the vertices p, p^- and p^+ in the counter-clockwise ordered list $\mathscr{P}(c)$. Likewise, the cells c, c^- and c^+ in the counter-clockwise ordered list $\mathscr{C}(p)$ are also shown. The arrows pointing between the midpoint of the edges to the vertex p denote the half-edge regions. The half-edge area vectors $\boldsymbol{a}_c^{(p,p^\pm)}$ and the corner area vector $\boldsymbol{a}_c^p$ for the vertex p with respect to the cell c are also displayed. (b) Figure shows a fragment of the computational mesh with regions corresponding to the primal, dual and subcell elements

Without loss of generality, the existence of a discrete space $\varpi(t)$, defined by a continuous one-to-one linear map $\varpi(t) = \chi(\varpi(0))$ and filled with non-overlapping arbitrary polygonal cells Ψ_c, encompassing the physical space is assumed. In what follows, an arbitrary polygonal cell Ψ_c in $\varpi(t)$ is denoted by an unique index c and an arbitrary vertex (also called node/point) in $\varpi(t)$ is denoted by p (Figure 2(a)). For any cell c, $\mathscr{P}(c)$ represents the set of counter-clockwise ordered (list of) vertices defining the cell c (Figure 2(a)). For any vertex p, the vertices p^- and p^+ denote the preceding and succeeding vertices to p in the ordered list $\mathscr{P}(c)$. Similarly, $\mathscr{C}(p)$ corresponds to the set of counter-clockwise ordered (list of) cells surrounding the vertex p. The set $\mathscr{S}(c)$ is also introduced to denote the counter-clockwise ordered list of edges defining the cell c. Scalar, vector and tensor variables are denoted as $\phi, \boldsymbol{\phi}, \boldsymbol{\Phi}$ respectively. For the sake of convenience, the total time derivative $\frac{d\boldsymbol{\Phi}}{dt}$ is often written as $\dot{\boldsymbol{\Phi}}$.

Because of repeated use of subscripts and superscripts, it is important to state clearly the nomenclature rules assumed throughout this work. The actual computational mesh associated with the cell Ψ_c is denoted as the primal mesh. Similarly, there exists a dual mesh associated with vertex p (Figure 2(b)). In the dual mesh, the vertex p resides at the centroid of the dual mesh element (Figure 2(b)). Mesh elements in the dual mesh are denoted by Ψ_p. The indices corresponding to vertex and centroid are denoted by superscripts and subscripts respectively (Figure 2(a)). For instance, the area of the primal and dual mesh elements with respect to the primal mesh are denoted as A_c and A^p respectively. Alternatively, area of primal and dual mesh elements with respect to the dual mesh are denoted as A^c and A_p respectively. Nevertheless, $A_c = A^c$ and $A_p = A^p$ as they both refer to the same regions in $\varpi(t)$. The dual mesh and elements introduced here are never used in the computation. The actual computation is performed on the primal mesh. As will be shown later, the dual mesh and elements come handy when deriving the discrete equations and demonstrating the compatibility of the cell-centered schemes. Therefore, unless otherwise stated, the primal mesh is always the reference mesh. Summations are always performed over the superscripts (with subscripts held constant). In other words, the subscripts indicate the reference mesh about which the summations are performed.

A subcell Ψ_c^p is formed by the intersection of the cells in the primal and the dual mesh *i.e.* $\Psi_c^p = \Psi_c \cap \Psi_p$ (Figure 2(b)). The area of the subcell Ψ_c^p is denoted as A_c^p (when viewed from the primal mesh). The following

identities immediately follow from the definition of subcell:

$$A^p = \sum_{\mathscr{C}(p)} A_c^p \tag{25}$$

$$A_c = \sum_{\mathscr{P}(c)} A_c^p \tag{26}$$

The same holds true for the volumes of the primal and dual mesh elements.

4. Governing Equations Revisited

The purpose of this section is to derive the discrete counterpart for the evolution equations. The equations that are evolved in time constitute the mass, momentum, energy and volumetric conservation laws listed in Eqs (1), (2), (3) & (4) respectively, in addition to the stress evolution equation presented in Eq (16). The evolution equations are solved in the discrete space $\varpi(t)$ with each arbitrary polygonal cell Ψ_c (also denoted as c for convenience) denoting a control volume/mass. To facilitate the derivation of the discrete expressions for the evolution equations, the integral form of the governing equations is first formulated in these control volumes.

4.1. Control Volume and area weighted Formulations of Governing Equations

An elemental control volume dV, for both planar and cylindrical geometries, can be written as $dV = RdA$ with dA defined in the Cartesian frame as $dA = dxdy$. Using this definition and exploiting the Gauss-divergence integral identity, the integral form of the evolution equations (Eqs (1), (2), (3), (4) & (16)) can be written as follows:

$$\frac{d}{dt}\int_{V(t)} dM = 0 \tag{27}$$

$$\frac{d}{dt}\int_{A(t)} \boldsymbol{u} dM - \int_{\partial A} \boldsymbol{\sigma}\cdot\boldsymbol{n} R dl + \beta \int_{A(t)} \begin{pmatrix} \sigma_{zz} \\ 0 \end{pmatrix} dA = 0 \tag{28}$$

$$\frac{d}{dt}\int_{V(t)} E dM - \int_{\partial A} \boldsymbol{n}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{u}) R dl = 0 \tag{29}$$

$$\frac{d}{dt}\int_{A(t)} \boldsymbol{S} dM + \int_{A(t)} (\boldsymbol{S\Omega} - \boldsymbol{\Omega S}) dM - 2G\int_{\partial A} \rho(\frac{1}{2}\boldsymbol{n}\otimes\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}\otimes\boldsymbol{n} - \frac{1}{3}\boldsymbol{u}\cdot\boldsymbol{n}\boldsymbol{I}) R dl = 0 \tag{30}$$

$$\frac{d}{dt}\int_{V(t)} dV - \int_{\partial A} \boldsymbol{u}\cdot\boldsymbol{n} R dl = 0 \tag{31}$$

where $dM = \rho R dA$ is the elemental mass of the control volume $V(t)$ and $\boldsymbol{n}$ denotes the outward unit normal vector to the boundary ∂A of A. The set of equations displayed above correspond to the control volume representation of the governing equations. The integral form of the stress evolution equation is obtained by integrating the constitutive equation (Eq (20)) in combination with the Reynold's transport theorem.

Defining an averaged pseudo-radius $\grave{R} = \frac{1}{A}\int_{A(t)} R dA$ and utilizing the expressions for the divergence operator (Eqs (8) & (9)), the area weighted representation of the momentum and stress evolution equations in Cartesian reference frame can be written as follows:

$$\frac{d}{dt}\int_{A(t)} \boldsymbol{u} d\mu - \int_{\partial A} \boldsymbol{\sigma}\cdot\boldsymbol{n} dl + \frac{\beta}{\grave{R}}\int_{A(t)} \begin{pmatrix} \sigma_{xx} - \sigma_{zz} \\ \sigma_{xy} \end{pmatrix} dA = 0 \tag{32}$$

$$\frac{d}{dt}\int_{A(t)} (\boldsymbol{S} + \boldsymbol{S\Omega} - \boldsymbol{\Omega S}) d\mu - 2G\int_{\partial A} \rho(\frac{1}{2}\boldsymbol{n}\otimes\boldsymbol{u} + \frac{1}{2}\boldsymbol{u}\otimes\boldsymbol{n} - \frac{1}{3}\boldsymbol{u}\cdot\boldsymbol{n}) dl = 0 \tag{33}$$

where $d\mu = \rho dA$ is the planar inertia [17]. Note that, when $\beta = 0$, $R = \grave{R} = 1$. Thus it is recognized that in both control volume and area weighted representations, the governing equations corresponding to planar geometry

are recovered when $\beta = 0$. In light of the importance of preserving symmetry (if any) in the flow field, it is desirable to work with area weighted representations of the momentum and stress evolution equations [12]. Hence, Eqs (27), (32), (29), (33) & (31) constitute the set of equations (governing equations augmented with the evolution equation for deviatoric part of stress tensor) that must be evolved in time.

4.2. Green's Integral Identity for Discrete Representation of Governing Equations

Following the analysis presented in [41, 17], the discrete counterpart of the integral formulations of governing equations (Eqs (27), (32), (29), (33), & (31)) can be approximated, by defining suitable averaged quantities, as follows:

$$\frac{dm_c}{dt} = 0 \tag{34}$$

$$\mu_c \frac{d\boldsymbol{u}_c}{dt} + \sum_{\mathscr{S}(c)} \boldsymbol{\sigma}_c^s \cdot \boldsymbol{n}_c^s R^s l^s - \frac{\beta}{\grave{R}_c} \begin{pmatrix} \sigma_{xx,c} - \sigma_{zz,c} \\ \sigma_{xy,c} \end{pmatrix} = 0 \tag{35}$$

$$\frac{dE_c}{dt} + \sum_{\mathscr{S}(c)} \boldsymbol{n}_c^s \cdot (\boldsymbol{\sigma}_c^s \cdot \boldsymbol{u}_c^s) R^s l^s = 0 \tag{36}$$

$$\mu_c \left(\frac{d\boldsymbol{S}_c}{dt} + \boldsymbol{S}_c \boldsymbol{\Omega}_c - \boldsymbol{\Omega}_c \boldsymbol{S}_c\right) - 2\rho_c G \sum_{\mathscr{S}(c)} (\boldsymbol{n}_c^s \otimes \boldsymbol{u}_c^s + \boldsymbol{u}_c^s \otimes \boldsymbol{n}_c^s - \frac{1}{3} \boldsymbol{n}_c^s \cdot \boldsymbol{u}_c^s) l^s = 0 \tag{37}$$

$$\frac{dV_c}{dt} - \sum_{\mathscr{S}(c)} \boldsymbol{n}_c^s \cdot \boldsymbol{u}_c^s R^s l^s = 0 \tag{38}$$

where $\mathscr{S}(c)$ is the set of counter-clockwise ordered list of edges (the discrete analog of the boundary ∂A of integration) for cell Ψ_c, $\boldsymbol{n}_c^s$ is the normal vector to the edges in $\mathscr{S}(c)$, and the superscript s in $\boldsymbol{u}_c^s, \boldsymbol{\sigma}_c^s$ denotes the approximation for the surface integral over an arbitrary edge $s \in \mathscr{S}(c)$:

$$\boldsymbol{\Phi}_c^s = \frac{1}{l_s} \int_s \boldsymbol{\Phi} dl \tag{39}$$

and the variables $\boldsymbol{u}_c$, E_c, $\boldsymbol{\Omega}_c$ & $\boldsymbol{S}_c$ denote the volume averaged velocity vector, total energy, spin and shear stress tensor for the cell c:

$$\boldsymbol{\Phi}_c = \frac{1}{V_c} \int_{\Psi_c} \boldsymbol{\Phi} dV \tag{40}$$

The system of equations presented above introduces the cell-centered formulation for the governing equations in the discrete space. The primitive variables namely $\rho_c, \boldsymbol{u}_c, E_c$ & $\boldsymbol{\sigma}_c$ are stored and evolved at the cell-center, once the approximations to the surface fluxes in the discrete system are determined. It is important to recognize that the formulation presented above is a direct consequence of the Green's integral identity. This is the formulation that has been employed in the existing cell-centered schemes [21, 15, 16, 17, 18, 19, 20]. Although such an approach may be found adequate, however, the formulation becomes questionable as the validity of the tensor operators (gradient and divergence), theorems and identities in the discrete space cannot be ascertained. Hence, the present work departs from the existing notion with formulating the discrete equations for conservation laws. In this work, the discrete analog for the governing equations are derived by invoking the mimetic theory and this is the subject of analysis for the rest of this section.

4.3. Mimetic Theory for Discrete Divergence and Gradient Operators

To establish the discrete version of the governing equations, the discrete analogs for the gradient and divergence operators must be determined at the outset. In accordance to the mimetic theory, these operators are formulated by mimicking some of the crucial conservation laws and vector identities as accurately as possible in the discrete space.

To this end, a first-order finite volume approximation for the discrete operators are defined as given below:

$$[\boldsymbol{\nabla} \odot \boldsymbol{\Phi}]_c = \frac{1}{V_c} \int_{V_c} \boldsymbol{\nabla} \odot \boldsymbol{\Phi} dV \tag{41}$$

$$[\boldsymbol{\nabla} \odot \boldsymbol{\Phi}]_p = \frac{1}{V^p} \int_{V^p} \boldsymbol{\nabla} \odot \boldsymbol{\Phi} dV \tag{42}$$

where $\odot$ denotes dot, cross or tensor product operator permissible on Φ [42]. The symbol $[\ ]_{c|p}$ indicates that the operator is evaluated on the primal (c) or the dual (p) mesh. It is now required to determine $[\boldsymbol{\nabla} \odot \boldsymbol{\Phi}]_c$ to evolve the governing equations.

Support operator method [27, 43], often used to construct mimetic schemes, provides a systematic approach for constructing discrete analogs for the divergence and gradient operators [14]. The discrete operators constructed using the support operator method satisfy relationships and identities that are intimately tied to the conversations laws and physical principles. In what follows, the support operator method will be used to determine the functional form for the discrete divergence and gradient operators.

The methodology adopted in this work follows the discussion outlined in [14]. The four key steps that are used in the support operator method are as follows:

1. Determine the prime operator
2. Identify the derived/target operator (gradient of scalar/vector, divergence of tensor) for which the mimetic expression is sought
3. Based on the target operator, select important identities and conservation laws that must be satisfied
4. Enforce the identities and conservation laws via summation over the entire domain (in the discrete space) to obtain the expression for the target/derived operator

As noted in [14], the prime operator is chosen based on the application and details of the discretization. As the divergence of the velocity field is directly related to the volumetric strain-rate and since the volume of a cell $V_c = V_c(\boldsymbol{x}^p)$ can be expressed as a computable function of the position vector of the vertices, $\boldsymbol{x}^p \in \mathscr{P}(c)$ [35], it is logical to choose the divergence of vector as the prime operator [14]. Furthermore, there is also a freedom of choosing the prime operator with respect to the primal or dual mesh as the conservation laws apply equally on both reference meshes. As will be shown in the following sections, this fact is exploited to its fullest potential in deriving the target operators. This is the most important step which distinguishes and immensely simplifies (without the need for introducing and transforming to curvilinear coordinates [13]) the analysis presented in this work from [14, 13, 1, 2].

4.3.1. Discrete Operator for Divergence of Vectors - Prime Operator

The prime operator is the key to establishing the functional form of the target/derived operators. As an evolution equation for volume is not explicitly solved, it is important to formulate the discretization scheme such that it remains compatible with the volume evolution equation (and consequently with the stress-volume work). Since the divergence of velocity vector is related to the volumetric strain-rate, expressing the rate of change of volume of a cell as:

$$\frac{d}{dt} V_c(\boldsymbol{x}^p) = \boldsymbol{\nabla} V_c(\boldsymbol{x}^p) \cdot \frac{d\boldsymbol{x}^p}{dt} \ \forall\, p \in \mathscr{P}(c) \tag{43}$$

one can immediately guess how this relation can be used to extract the expression for the discrete divergence operator for vectors. By performing triangular decomposition of the cell and by virtue of Pappus' rule, the analytical expression for the volume of an arbitrary polygonal cell Ψ_c can be expressed as:

$$V_c = \frac{1}{2(2+\beta)} \sum_{\mathscr{P}(c)} (R^p + R^{p^-})(\boldsymbol{x}^{p^-} \times \boldsymbol{x}^p) \cdot \boldsymbol{k} \tag{44}$$

where $\boldsymbol{k}$ is the axis perpendicular to the plane of the paper (taken positive coming out of the paper) and the last term corresponds to the planar area for the triangular segment formed by the triplet $(O, \boldsymbol{x}^{p^-}, \boldsymbol{x}^p)$. Inserting the expression

for volume in Eq (43), the following is obtained:

$$\frac{d}{dt}V_c = \frac{1}{2(2+\beta)}\sum_{\mathscr{P}(c)}\begin{bmatrix}\left(\dot{R}^p+\dot{R}^{p^-}\right)(\boldsymbol{x}^{p^-}\times\boldsymbol{x}^p)+\\ (R^p+R^{p^-})(\dot{\boldsymbol{x}}^{p^-}\times\boldsymbol{x}^p+\boldsymbol{x}^{p^-}\times\dot{\boldsymbol{x}}^p)\end{bmatrix}\cdot\boldsymbol{k} \tag{45}$$

$$= \frac{1}{2(2+\beta)}\sum_{\mathscr{P}(c)}\begin{bmatrix}\dot{R}^p(\boldsymbol{x}^{(p)}\times\boldsymbol{x}^{p^+}+\boldsymbol{x}^{p^-}\times\boldsymbol{x}^p)+\\ (R^p+R^{p^+})(\dot{\boldsymbol{x}}^p\times\boldsymbol{x}^{p^+})+\\ (R^p+R^{p^-})(\boldsymbol{x}^{p^-}\times\dot{\boldsymbol{x}}^p)\end{bmatrix}\cdot\boldsymbol{k} \tag{46}$$

where in the last expression, the indices corresponding to terms containing $\dot{\boldsymbol{x}}^{p^-}$ & $\dot{R}^{p^-}$ were shifted from $p^- \rightarrow p$. In the above expressions, the term containing $\dot{R}^p$ corresponds to the contribution from the axisymmetric model. Noting that $\dot{\boldsymbol{x}}^{(p^\pm)} = \left(u^{(p^\pm)}, v^{(p^\pm)}\right)$ and $\dot{R}^{(p^\pm)} = \beta\dot{u}^{(p^\pm)}$, the equations displayed above can be simplified to obtain:

$$\frac{d}{dt}V_c = \frac{1}{2(2+\beta)}\sum_{\mathscr{P}(c)}\begin{bmatrix}u^p\left\{\begin{matrix}\left(R^p+R^{p^+}\right)\left(y^{p^+}-y^p\right)+\\ \left(R^p+R^{p^-}\right)\left(y^p-y^{p^-}\right)+\\ \left(R^{p^+}-R^{p^-}\right)y^p\end{matrix}\right\}-\\ v^p\left\{\begin{matrix}\left(R^p+R^{p^+}\right)\left(x^{p^+}-x^p\right)+\\ \left(R^p+R^{p^-}\right)\left(x^p-x^{p^-}\right)+\\ \left(R^{p^+}-R^{p^-}\right)x^p\end{matrix}\right\}+\\ \dot{R}_p\left\{x^p(y^{p^+}-y^{p^-})-y^p(x^{p^+}-x^{p^-})\right\}\end{bmatrix} \tag{47}$$

Analyzing the last term in the previous expression,

$$\begin{aligned}\left\{x^p(y^{p^+}-y^{p^-})-y^p(x^{p^+}-x^{p^-})\right\}\boldsymbol{k} &= \boldsymbol{x}^p\times\boldsymbol{x}^{p^+}+\boldsymbol{x}^{p^-}\times\boldsymbol{x}^p\\ &= 2\boldsymbol{x}^p\times\frac{(\boldsymbol{x}^{p^+}+\boldsymbol{x}^p)}{2}+2\frac{(\boldsymbol{x}^{p^-}+\boldsymbol{x}^p)}{2}\times\boldsymbol{x}^p\\ &= 2\boldsymbol{x}^p\times\boldsymbol{x}^{(\frac{p,p^+}{2})}+2\boldsymbol{x}^{(\frac{p,p^-}{2})}\times\boldsymbol{x}^p\\ &= 6A_c^p\boldsymbol{k}-2A_c^p\boldsymbol{k}-2(\boldsymbol{x}^{(\frac{p,p^+}{2})}\times\boldsymbol{x}_c+\boldsymbol{x}_c\times\boldsymbol{x}^{(\frac{p,p^-}{2})}))\end{aligned} \tag{48}$$

the contribution from the area of the subcell A_c^p formed by the cell c (with $\boldsymbol{x}_c$ denoting the centroid), the vertex p and the position vectors of the mid point ($\boldsymbol{x}^{(\frac{p,p^\pm}{2})} = \frac{\left(\boldsymbol{x}^p+\boldsymbol{x}^{p^\pm}\right)}{2}$) of the edges $p, p^\pm$ can be recognized (Figure 3):

$$4A_c^p\boldsymbol{k} = \boldsymbol{x}^p\times\boldsymbol{x}^{p^+}+(\boldsymbol{x}^{p^+}-\boldsymbol{x}^{p^-})\times\boldsymbol{x}_c+\boldsymbol{x}^{p^-}\times\boldsymbol{x}^p \tag{49}$$

Substituting this expression for the last term and rearranging the first term in Eq (47), the rate of change of volume can be written as

$$\frac{d}{dt}V_c = \frac{1}{2(2+\beta)}\sum_{\mathscr{P}(c)}\begin{bmatrix}u^p\left\{\begin{matrix}\left(2R^p+R^{p^+}\right)\left(y^{p^+}-y^p\right)+\\ \left(2R^p+R^{p^-}\right)\left(y^p-y^{p^-}\right)-6\beta A_c^p\end{matrix}\right\}-\\ v^p\left\{\begin{matrix}\left(2R^p+R^{p^+}\right)\left(x^{p^+}-x^p\right)+\\ \left(2R^p+R^{p^-}\right)\left(x^p-x^{p^-}\right)\end{matrix}\right\}+\\ 6\dot{R}_pA_c^p\end{bmatrix} \tag{50}$$

As will be shown later, Eq (50) will be critical in obtaining an expression for the discrete gradient operator for vectors.

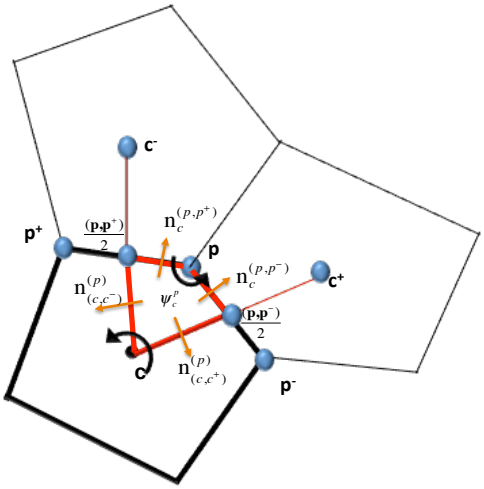


Figure 3: Figure shows the subcell Ψ_c^p formed by the centroid c, the vertex p and the midpoint of the edges $\frac{(p,p^{\pm})}{2}$. The figure also shows the normals $\boldsymbol{n}^p_{(c,c^{\pm})}$ to the edges in the dual mesh.

Simplifying Eq (50) yields the following relation [23, 35, 17]:

$$\frac{d}{dt}V_c = \sum_{\mathscr{P}(c)} \boldsymbol{a}_c^p \cdot \boldsymbol{u}^p \tag{51}$$

where

$$\boldsymbol{a}_c^p = \boldsymbol{a}_c^{(p,p^-)} + \boldsymbol{a}_c^{(p,p^+)} \tag{52}$$

is the area-vector [23] (also called as the corner vector [11, 12, 16]), and

$$\boldsymbol{a}_c^{(p,p^{\pm})} = \frac{1}{2+\beta}(2R^p + R^{(p^{\pm})})\frac{1}{2}l^{(p,p^{\pm})}\boldsymbol{n}_c^{(p,p^{\pm})} \tag{53}$$

$$\boldsymbol{n}_c^{(p,p^+)} = \frac{(\boldsymbol{x}^{p^+} - \boldsymbol{x}^p)}{l^{(p,p^+)}} \times \boldsymbol{k} \tag{54}$$

$$\boldsymbol{n}_c^{(p,p^-)} = \frac{(\boldsymbol{x}^p - \boldsymbol{x}^{p^-})}{l^{(p,p^-)}} \times \boldsymbol{k} \tag{55}$$

are the half edge area and normal vectors associated with each edge emanating from $p \in \mathscr{P}(c)$. One can immediately recognize the following standard identities associated with the corner vector [16]:

$$\sum_{\mathscr{P}(c)} \boldsymbol{a}_c^p = \beta A_c \boldsymbol{r} \tag{56}$$

$$\sum_{\mathscr{C}(p)} \boldsymbol{a}_c^p = 0 \tag{57}$$

where $A_c = \sum_{\mathscr{P}(c)} A_c^p$ is the planar area of the cell c. The geometrical interpretation of the corner vector presented in [16] also readily extends to $\boldsymbol{a}_c^p$. Thus, the discrete divergence operator for the velocity vector can be expressed as:

$$\begin{aligned}
\left[\boldsymbol{\nabla}\cdot\boldsymbol{u}\right]_c &= \sum_{\mathscr{P}(c)} \mathscr{D}^p_{(c,x)} u^p + \mathscr{D}^p_{(c,y)} v^p + \mathscr{D}^p_{(c,R)} \dot{R}^p \\
&= \frac{1}{V_c} \sum_{\mathscr{P}(c)} \boldsymbol{a}_c^p \cdot \boldsymbol{u}^p \\
&= \frac{1}{V_c} \sum_{\mathscr{P}(c)} (\boldsymbol{a}_c^{(p,p^-)} + \boldsymbol{a}_c^{(p,p^+)}) \cdot \boldsymbol{u}^p
\end{aligned} \tag{58}$$

and therefore the components $(\mathscr{D}_{x,vector}, \mathscr{D}_{y,vector})$ of the discrete divergence operator take the form given below:

$$\mathscr{D}_{x,vector} = \frac{1}{V_c} \sum_{\mathscr{P}(c)} \mathscr{D}^p_{(c,x)} = \frac{1}{V_c} \sum_{\mathscr{P}(c)} \left(\boldsymbol{a}_{c,x}^{(p,p^-)} + \boldsymbol{a}_{c,x}^{(p,p^+)} - \frac{6\beta}{2\left(2+\beta\right)} A_c^p \right) \tag{59}$$

$$\mathscr{D}_{y,vector} = \frac{1}{V_c} \sum_{\mathscr{P}(c)} \mathscr{D}^p_{(c,y)} = \frac{1}{V_c} \sum_{\mathscr{P}(c)} \boldsymbol{a}_{c,y}^{(p,p^-)} + \boldsymbol{a}_{c,y}^{(p,p^+)} \tag{60}$$

$$\mathscr{D}_{R,vector} = \frac{1}{V_c} \sum_{\mathscr{P}(c)} \mathscr{D}^p_{(c,R)} = \frac{6\beta}{2\left(2+\beta\right) V_c} \sum_{\mathscr{P}(c)} A_c^p \tag{61}$$

Examining Eq (58), it can be inferred that the discrete divergence operator for any arbitrary vector $\boldsymbol{w}$ is solely a function of the geometrical quantities associated with the vertices. Hence, the first-order finite volume approximation for $\boldsymbol{\nabla}_{div\ \boldsymbol{w}}$ is given as follows:

$$\left[\boldsymbol{\nabla}_{div\ \boldsymbol{w}}\right]_c = \frac{1}{V_c} \sum_{\mathscr{P}(c)} (\boldsymbol{a}_c^p)\cdot = \frac{1}{V_c} \sum_{\mathscr{P}(c)} (\boldsymbol{a}_c^{(p,p^-)} + \boldsymbol{a}_c^{(p,p^+)})\cdot \tag{62}$$

Eq (62) suggests that the quadrature points for the divergence operator for a vector could either reside on the half edges (namely (p,p^-) & (p,p^+)) associated with each vertex $p \in \mathscr{P}(c)$ or on the vertex p itself. Since the kinematic condition (Eq (5)) demands the velocity vectors to be defined uniquely at the vertices ($\boldsymbol{u}^p$), the most appropriate choice for the quadrature points are the vertices of the mesh. This will become clear when the discrete gradient of scalar and vector operators are determined. In fact, the degree of freedom in choosing the quadrature points is restricted to the vertices of the mesh by the kinematic condition together with the curl identity.

The consistency of the above formulation can be demonstrated for an uniform flow in planar geometry. For such flows, the volumetric strain rate $\boldsymbol{\nabla}\cdot\boldsymbol{u}$ reduces to zero and the same is mimicked at the discrete level due to the identity listed in Eq (56). Comparing Eqs (51) & (38), the surface approximations at the quadrature points can be obtained:

$$\begin{aligned}
\boldsymbol{u}_c^s &= \boldsymbol{u}^p \text{ and} \\
\boldsymbol{n}_c^s l^s R^s &= \boldsymbol{a}_c^p
\end{aligned}$$

and the summation is now carried over the vertices defining the cell Ψ_c.

4.3.2. Discrete Operator for Gradient of Vectors

In this section, the discrete operator corresponding to gradient of a vector (target operator) is derived. To determine this operator, the vector identity $tr\left(\boldsymbol{\nabla w}\right) = \boldsymbol{\nabla}\cdot\boldsymbol{w}$ that holds true for any vector $\boldsymbol{w}$ is enforced at the discrete level. Selecting $\boldsymbol{w}$ to be the velocity vector $\boldsymbol{u}$, one obtains:

$$tr\left(\left[\boldsymbol{\nabla u}\right]_c\right) = \left[\boldsymbol{\nabla}\cdot\boldsymbol{u}\right]_c \tag{63}$$

Therefore, from the expression for volumetric strain-rate displayed in Eq (50), the components of discrete gradient operator can be identified by comparing the coefficients of $(u^p, v^p, \dot{R}^p)$ in $[\boldsymbol{\nabla} \cdot \boldsymbol{u}]_c$:

$$\sum_{\mathscr{P}(c)} \mathscr{G}^p_{x,vector} u^p + \mathscr{G}^p_{y,vector} v^p + \mathscr{G}^p_{R,vector} \dot{R}^p = \sum_{\mathscr{P}(c)} \mathscr{D}^p_{x,vector} u^p + \mathscr{D}^p_{y,vector} v^p + \mathscr{D}^p_{R,vector} \dot{R}^p \tag{64}$$

$$\mathscr{G}_{x,vector} = \frac{1}{2(2+\beta)V_c} \sum_{\mathscr{P}(c)} \left\{ \begin{array}{c} \left(2R^p + R^{p^+}\right) l^{(p,p^+)} n^{(p,p^+)}_{c,x} + \\ \left(2R^p + R^{p^-}\right) l^{(p,p^+)} n^{(p,p^-)}_{c,x} - \\ 6\beta A^p_c \end{array} \right\} \tag{65}$$

$$\mathscr{G}_{y,vector} = \frac{1}{2(2+\beta)V_c} \sum_{\mathscr{P}(c)} \left\{ \begin{array}{c} \left(2R^p + R^{p^+}\right) l^{(p,p^+)} n^{(p,p^+)}_{c,y} + \\ \left(2R^p + R^{p^-}\right) l^{(p,p^+)} n^{(p,p^-)}_{c,y} \end{array} \right\} \tag{66}$$

$$\mathscr{G}_{R,vector} = \frac{6\beta}{2(2+\beta)V_c} \sum_{\mathscr{P}(c)} A^p_c \tag{67}$$

Therefore, the strain-rate tensor $\boldsymbol{D}$ and spin tensor $\boldsymbol{\Omega}$, that are required to evolve the governing equations, can be expressed as given below:

$$\boldsymbol{D} = \frac{1}{2(2+\beta)V_c} \begin{pmatrix} \sum_{\mathscr{P}(c)} \mathscr{G}^p_x u^p & \frac{1}{2}\sum_{\mathscr{P}(c)} (\mathscr{G}^p_y u^p + \mathscr{G}^p_x v^p) & 0 \\ \frac{1}{2}\sum_{\mathscr{P}(c)} (\mathscr{G}^p_y u^p + \mathscr{G}^p_x v^p) & \sum_{\mathscr{P}(c)} \mathscr{G}^p_y v^p & 0 \\ 0 & 0 & \sum_{\mathscr{P}(c)} \mathscr{G}_R \dot{R}^p \end{pmatrix} \tag{68}$$

$$\boldsymbol{\Omega} = \frac{1}{2(2+\beta)V_c} \begin{pmatrix} 0 & \frac{1}{2}\sum_{\mathscr{P}(c)} (\mathscr{G}^p_y u^p - \mathscr{G}^p_x v^p) & 0 \\ -\frac{1}{2}\sum_{\mathscr{P}(c)} (\mathscr{G}^p_y u^p - \mathscr{G}^p_x v^p) & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{69}$$

For the special case when $\boldsymbol{u} = \boldsymbol{x}$ and $\beta = 1$, the mimetic formulation for gradient operator reduces to

$$\sum_{\mathscr{P}(c)} \mathscr{G}^p_x x^p = \frac{1}{6V_c} \sum_{\mathscr{P}(c)} \left(R^p + R^{p^+}\right) (x^p y^{p^+} - x^{p^+} y^p) = 1 \tag{70}$$

$$\sum_{\mathscr{P}(c)} \mathscr{G}^p_y y^p = \frac{1}{6V_c} \sum_{\mathscr{P}(c)} \left(R^p + R^{p^+}\right) (x^{p^+} y^p - x^p y^{p^+}) = 1 \tag{71}$$

$$\sum_{\mathscr{P}(c)} \mathscr{G}^p_R x^p = \frac{1}{6V_c} \sum_{\mathscr{P}(c)} 6x^p \left\{ x^p y^{p^+} - y^p x^{p^+} \right\} = 1 \tag{72}$$

$$\sum_{\mathscr{P}(c)} \mathscr{G}^p_x y^p = \sum_{\mathscr{P}(c)} \mathscr{G}^p_y x^p = 0 \tag{73}$$

From Eq (63) it is easy to see that

$$[\boldsymbol{\nabla} \cdot \boldsymbol{x}]_c = 3 \tag{74}$$

Similarly it can be shown that for a pure rigid body translation motion, the discrete velocity gradient and spin tensors vanish.

Remark 1. It is worth noting that, for axisymmetric geometries, the discrete expression for gradient of vector operator presented in this section has not been reported in the past. The terms in the expression are fairly intuitive and the key step that lead to this expression is the intermediate form of the divergence of vector operator displayed in Eq (50). In Eq (50), the contribution from the axisymmetric term were kept segregated from the rest of the terms and therefore a term by term comparison was made feasible.

4.3.3. Discrete Operator for Gradient of Scalars

There are multiple ways to deduce the discrete operator for gradient of scalars [44, 14]. The most simplest and primitive approach is to obtain the expression by dotting the expression for gradient of vector with an unit vector ($\boldsymbol{i}$, $\boldsymbol{j}$ & $\boldsymbol{r}$ in this case) to obtain the discrete operator in the direction of that unit vector. However, such an approach will defeat the purpose of demonstrating the compatibility of the scheme. Therefore, in this section, a different approach is adopted which leads to the aforementioned trivial outcome. This is primarily done to demonstrate the essence and consistency of the mimetic formulation. Furthermore, it is believed that the steps outlined in this section may serve as a set of general guidelines that could be potentially exploited in constructing discrete (mimetic) operators for other applications. At this juncture, it is worth re-iterating that the vast amount of literature that exist on mimetic methods are limited to staggered grid techniques. The cell-centered schemes that have been developed thus far have extended the discrete formulation for $\frac{\partial}{\partial x}, \frac{\partial}{\partial y}$ operators, determined from the divergence (prime) operator, directly to the derived operator. The validity of such an assumption is established through the analysis presented in this and subsequent sections.

Following [44, 14], the discrete operator is established by exploiting the adjointness property of gradient and divergence operators, defined by the vector identity written for an arbitrary volume $\mathscr{V}(t)$ and for a scalar φ:

$$\int_{\mathscr{V}(t)} \boldsymbol{\nabla}\varphi \cdot \boldsymbol{w} dV = \int_{\partial\mathscr{V}(t)} \varphi \boldsymbol{n} \cdot \boldsymbol{w} R dl - \int_{\mathscr{V}(t)} \varphi \boldsymbol{\nabla} \cdot \boldsymbol{w} dV \tag{75}$$

Assuming a zero boundary integral (with φ vanishing at the boundary), the discrete analog for the integral identity shown above can be written as

$$\sum_c [\boldsymbol{\nabla}\varphi \cdot \boldsymbol{w}]_c V_c = -\sum_p [\varphi \boldsymbol{\nabla} \cdot \boldsymbol{w}]_P V^p \tag{76}$$

where the summation in the left and the right hand side are performed over the cells c and vertices p in $\varpi(t)$ respectively, V^p denotes the volume of the dual mesh associated with the vertex p and the operator $[\,]_p$ is the dual mesh based volume averaged quantity. To be more specific, $[\,]_p$ indicates an "operation" over quantities stored at the cell center (of the primal mesh) to yield an "operator" centered at the vertex for the corresponding dual mesh (common to staggered grid methods). Since the quantity of interest is the gradient of scalar operating from vertices to cell $[\boldsymbol{\nabla}\varphi]_c$, the prime operator in this case is the discrete divergence of vector $[\boldsymbol{\nabla} \cdot \boldsymbol{w}]_p$ operating from cells to vertex. To this degree, it is first required to devise a technique to evaluate the prime operator $[\boldsymbol{\nabla} \cdot \boldsymbol{w}]_p$ operating on the dual mesh. In retrospect of Eq (43) and the subsequent steps leading to Eq (58), choosing $\boldsymbol{w}$ to be $\boldsymbol{u}$ and performing the sequel of steps for the volume of dual mesh (written as a function of centroid coordinates *i.e.* $V^p(\boldsymbol{x}_c)$) lead to the following expression for discrete divergence of vector $[\boldsymbol{\nabla} \cdot \boldsymbol{w}]_p$:

$$\begin{aligned}
[\boldsymbol{\nabla} \cdot \boldsymbol{u}]_p &= \sum_{\mathscr{C}(p)} \begin{bmatrix} \frac{1}{(2+\beta)}(2R^c + R^{c^+})\frac{1}{2} l^{(c,c^+)} \boldsymbol{n}_p^{(c,c^+)} + \\ \frac{1}{(2+\beta)}(2R^c + R^{c^-})\frac{1}{2} l^{(c,c^-)} \boldsymbol{n}_p^{(c,c^-)} \end{bmatrix} \cdot \boldsymbol{u}_c && (77) \\
&= \sum_{\mathscr{C}(p)} \left(\boldsymbol{a}_p^{c^-} + \boldsymbol{a}_p^{c^+} \right) \cdot \boldsymbol{u}_c && (78) \\
&= \sum_{\mathscr{C}(p)} \boldsymbol{a}_p^c \cdot \boldsymbol{u}_c && (79)
\end{aligned}$$

where $l^{(c,c^\pm)}$ denote the length of the edges in the dual mesh and $\boldsymbol{n}_p^{(c,c^\pm)}$ are the normals to these edges (Figure 3). Using the identity listed in Eq (56), the following can be deduced for a subcell Ψ_c^p:

$$\begin{aligned}
(\quad & 2R^c + R^{c^+})\frac{1}{2} l^{(c,c^+)} \boldsymbol{n}_p^{(c,c^+)} + (2R^c + R^{c^-})\frac{1}{2} l^{(c,c^-)} \boldsymbol{n}_p^{(c,c^-)} = 6\beta A_c^p \boldsymbol{r} - \\
(\quad & 2R^p + R^{p^+})\frac{1}{2} l^{(p,p^+)} \boldsymbol{n}_c^{(p,p^+)} + (2R^p + R^{p^-})\frac{1}{2} l^{(p,p^-)} \boldsymbol{n}_c^{(p,p^-)}
\end{aligned} \tag{80}$$

i.e.

$$\boldsymbol{a}_p^c = 6\beta A_c^p \boldsymbol{r} - \boldsymbol{a}_c^p \tag{81}$$

Therefore, the discrete divergence operator in the dual mesh is given as

$$[\boldsymbol{\nabla}\cdot\boldsymbol{u}]_p = \frac{1}{2(2+\beta)V^p}\sum_{\mathscr{C}(p)}\left\{6\beta A_c^p \dot{R}_c - \left[\begin{array}{c}(2R^p+R^{p^+})l^{(p,p^+)}\boldsymbol{n}_c^{(p,p^+)}+\\(2R^p+R^{p^-})l^{(p,p^-)}\boldsymbol{n}_c^{(p,p^-)}\end{array}\right]\cdot\boldsymbol{u}_c\right\} \tag{82}$$

Inserting Eq (82) in Eq (76) and comparing the coefficients of $\boldsymbol{u}_c$, the components $(\mathscr{G}_{(x,scalar)}, \mathscr{G}_{(y,scalar)}, \mathscr{G}_{(r,scalar)})$ of the discrete gradient of scalar operator can be identified:

$$\mathscr{G}_{(x,scalar)} = \mathscr{G}_{x,vector} = \frac{1}{2(2+\beta)V_c}\sum_{\mathscr{P}(c)}\left[\begin{array}{c}(2R^p+R^{p^+})l^{(p,p^+)}n_{(c,x)}^{(p,p^+)}+\\(2R^p+R^{p^-})l^{(p,p^-)}n_{(c,x)}^{(p,p^-)}\end{array}\right] \tag{83}$$

$$\mathscr{G}_{(y,scalar)} = \mathscr{G}_{y,vector} = \frac{1}{2(2+\beta)V_c}\sum_{\mathscr{P}(c)}\left[\begin{array}{c}(2R^p+R^{p^+})l^{(p,p^+)}n_{(c,y)}^{(p,p^+)}+\\(2R^p+R^{p^-})l^{(p,p^-)}n_{(c,y)}^{(p,p^-)}\end{array}\right]$$

$$\mathscr{G}_{(R,scalar)} = \mathscr{G}_{R,vector} = \frac{6\beta}{2(2+\beta)V_c}\sum_{\mathscr{P}(c)} A_c^p \tag{84}$$

The consistency of the discrete gradient operator can be ascertained by confirming that, for a constant scalar field φ, the discrete divergence of the scalar reduces to zero. Substituting $\varphi_c = \varphi^p = \varphi$ in the discrete gradient operator, the following can be inferred:

$$\begin{aligned}[\boldsymbol{\nabla}\varphi]_c &= \frac{\varphi}{2(2+\beta)V_c}\sum_{\mathscr{P}(c)}\left[\begin{array}{c}(2R^p+R^{p^+})l^{(p,p^+)}\boldsymbol{n}_c^{(p,p^+)}+\\(2R^p+R^{p^-})l^{(p,p^-)}\boldsymbol{n}_c^{(p,p^-)}\end{array}\right] - \frac{\varphi}{2(2+\beta)V_c}\sum_{\mathscr{P}(c)} 6\beta A_c^p \\ &= \frac{6\beta\varphi}{2(2+\beta)V_c}A_c - \frac{6\beta\varphi}{2(2+\beta)V_c}A_c = 0\end{aligned}$$

This is an important property for flows with constant pressure *i.e.* the gradient of pressure goes to zero, ensuring that no numerical inaccuracies are introduced.

Remark 2. The method outlined in this section is generic and can be used to determine other operators. For instance, the same approach could be adopted to determine the discrete gradient operator for vectors. In this case, the chosen prime operator is $[\boldsymbol{\nabla}_{div\ \boldsymbol{w}}]_c$ to yield $[\boldsymbol{\nabla}_{grad\ scalar}]_p$. Then the adjointness property between the gradient of scalars with the gradient of vectors

$$\int_{\mathscr{V}(t)}\varphi\boldsymbol{\nabla}\boldsymbol{u}dV = \int_{\partial\mathscr{V}(t)}\varphi\boldsymbol{u}\times\boldsymbol{n}Rdl - \int_{\mathscr{V}(t)}\boldsymbol{u}\cdot\boldsymbol{\nabla}\varphi dV \tag{85}$$

can be exploited to result in the expressions given in Eqs (64)-(67).

Remark 3. It is important to point out the similarity between the expression given in Eqs (83)-(84) and what is reported in [17]. In [17], the discrete gradient operator for scalar was determined using the Green's integral identity. In contrast, the discrete expression presented above is obtained by satisfying important vector identities in the discrete model.

4.3.4. Discrete Operator for Divergence of Tensors

The task set forth in this section is to derive the discrete expression for the divergence of tensor operator. It is important to endure this analysis for the following two reasons:

- To establish the compatibility between the form of the gradient of vector operator presented in section 4.3.2 and the total energy conservation principle (Eq (29)).

- To assert the formulation for the gradient of scalar and vector operators. It is recalled here that the formulation for the gradient of vector operator formulated in section 4.3.2 has not been reported in the past and therefore it is imperative to verify the same.

In [14], several vector identities were employed to determine the the discrete divergence of stress tensor operator. This was deliberately done to circumvent from using the gradient of vector (in [14], the gradient of vector operator was derived from the divergence of stress tensor operator) as the prime operator. In this work, a more direct approach is adopted. The discrete divergence of stress tensor operator is determined by enforcing the adjointness property between the gradient and divergence operators. In order to satisfy the conservation of total energy principle, the integral form of the vector identity (Eq (11)) is invoked as the commencing point:

$$\int_{\mathscr{V}(t)} \boldsymbol{\sigma}\boldsymbol{\nabla}\cdot\boldsymbol{w}dV = \int_{\partial\mathscr{V}(t)} \boldsymbol{n}\cdot\boldsymbol{\sigma}\cdot\boldsymbol{w}Rdl - \int_{\mathscr{V}(t)} (\boldsymbol{\nabla}\cdot\boldsymbol{\sigma})\cdot\boldsymbol{w}dV \tag{86}$$

The above integral form reduces to the following in the discrete space (with vanishing boundary integral terms [14]):

$$\sum_c \boldsymbol{w}_c\cdot[\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}]_c V_c = -\sum_p [\boldsymbol{\nabla}\boldsymbol{w}]_p : \boldsymbol{\sigma}^p V^p \tag{87}$$

Setting $\boldsymbol{w}=\boldsymbol{u}$ and in combination with the analysis presented in sections 4.3.2 & 4.3.3, the gradient of vector in the dual mesh can be expressed as:

$$[\boldsymbol{\nabla}\boldsymbol{u}]_p V^p = \sum_{\mathscr{C}(p)} \begin{pmatrix} -a^p_{c_x}u_c & -0.5\left(a^p_{c_y}u_c + a^p_{c_x}v_c\right) & 0 \\ -0.5\left(a^p_{c_y}u_c + a^p_{c_x}v_c\right) & -a^p_{c_y}v_c & 0 \\ 0 & 0 & 6\beta A^p u^p \end{pmatrix} \tag{88}$$

where Eq (81) is used in deriving the above expression. Substituting this expression in Eq (87) and performing the summation over the vertices of the mesh yields:

$$\sum_p [\boldsymbol{\nabla}\boldsymbol{u}]_p : \boldsymbol{\sigma}^p V^p = -\sum_c \sum_{\mathscr{P}(c)} \left\{ \left(a^p_{c_x}\sigma^p_{xx} + a^p_{c_y}\sigma^p_{xy}\right)u_c + \left(a^p_{c_x}\sigma^p_{xy} + a^p_{c_y}\sigma^p_{yy}\right)v_c + 6\beta A_c \sigma_{zz_c} u_c \right\} \tag{89}$$

where the summation in the left hand side of the above expression is shifted to the primal mesh and the last term is a consequence of the conservation of stress volume work ($\int_{\varpi(t)} \boldsymbol{\sigma} : \boldsymbol{u}dV = \sum_c \boldsymbol{\sigma}_c : \boldsymbol{u}_c V_c = \sum_p \boldsymbol{\sigma}^p : \boldsymbol{u}^p V^p$). Comparing the coefficients of velocity components between the left and the right hand side leads to the final expression for the divergence of tensor operator:

$$[\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}]_c = \frac{1}{V_c}\sum_{\mathscr{P}(c)} \left\{ \begin{pmatrix} a^p_{c_x}\sigma^p_{xx} + a^p_{c_y}\sigma^p_{xy} \\ a^p_{c_x}\sigma^p_{xx} + a^p_{c_y}\sigma^p_{yy} \end{pmatrix} + \beta \begin{pmatrix} A^P_c \sigma_{zz_c} \\ 0 \end{pmatrix} \right\} \tag{90}$$

The consistency of the discrete expression presented above can be established by comparing it with the continuum analog. The form of the divergence operator in Eq (90) is exactly the same as the corresponding continuum expression displayed in Eq (10). It it important to point out that the discrete expression (Eq (90)) is derived without referring to Eq (10). In contrast to existing FV, FD and FE schemes, wherein the integral form of Eq (10) is utilized for discretizing the governing equations, the current formulation is constructed by infusing the fundamental vector identities and conservation principles in the discrete expression. The similarity between Eqs (90) & (10) also confirms the validity of the discrete expression for the gradient of vector operator. Furthermore, the compatibility with gradient of scalar operator can be established by setting $\boldsymbol{\sigma} = -\varphi\boldsymbol{I}$ to recover:

$$\begin{aligned} \mathscr{D}_{(x,tensor)} &= \mathscr{G}_{x,scalar} \\ \mathscr{D}_{(y,tensor)} &= \mathscr{G}_{y,scalar} \\ \mathscr{D}_{(r,tensor)} &= \mathscr{G}_{r,scalar} \end{aligned} \tag{91}$$

Remark 4. In the derivation presented above, an implicit assumption is made regarding the existence of tensor field at the quadrature points *i.e.* at the vertices of the mesh. Moreover, the degree of freedom in choosing the quadrature points is restricted to vertices because of the choice of the quadrature points for the prime operator, *i.e.* the divergence of vector operator. Alternatively, it can be concluded that the vector identities and discrete conservation laws hold true as long as the quadrature points for scalars, vectors and tensors coincide. A concrete proof for this claim can be provided by performing a rigorous analysis on the discrete curl operator. Since

$$[\nabla \times [\nabla \cdot \boldsymbol{w}]_c]_p = [\nabla \times [\nabla \cdot \boldsymbol{w}]_p]_c = 0 \text{ and} \tag{92}$$

$$[\nabla \times [\nabla \cdot \boldsymbol{\sigma}]_c]_p = \Big[\nabla \times [\nabla \cdot \boldsymbol{\sigma}]_p\Big]_c = 0, \tag{93}$$

can be shown to be true when the quadrature points for vectors and tensors coincide [19].

4.3.5. Discrete Expressions for the Plane Strain Case

The discrete expressions for the divergence and gradient operators can be recovered for the plane strain assumption by setting $\beta = 0$. With $\beta = 0$, Eq (44) reduces to

$$A_c = \frac{1}{2} \sum_{\mathscr{P}(c)} (\boldsymbol{x}^{p^-} \times \boldsymbol{x}^p) \cdot \boldsymbol{k}. \tag{94}$$

Consequently, the GCL for the plane strain assumption reduces to

$$\frac{1}{A_c}\frac{dA_c}{dt} = \boldsymbol{\nabla} A_c(\boldsymbol{x}^p) \cdot \boldsymbol{u}^p \; \forall \; p \in \mathscr{P}(c) \tag{95}$$

$$= \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_c^p \cdot \boldsymbol{u}^p \tag{96}$$

where

$$\widetilde{\boldsymbol{a}}_c^p = \frac{1}{2}\left(\widetilde{\boldsymbol{a}}_c^{(p,p^+)} + \widetilde{\boldsymbol{a}}_c^{(p,p^-)}\right) \tag{97}$$

$$\widetilde{\boldsymbol{a}}_c^{(p,p^\pm)} = l^{(p,p^\pm)} \boldsymbol{n}_c^{(p,p^\pm)} \tag{98}$$

Therefore, the corner area vector for the axisymmetric case can be expressed in terms of the half-edge area vectors corresponding to the plane strain assumption as follows:

$$\boldsymbol{a}_c^p = \frac{1}{2}\left(R_c^{(p,p^+)} \widetilde{\boldsymbol{a}}_c^{(p,p^+)} + R_c^{(p,p^-)} \widetilde{\boldsymbol{a}}_c^{(p,p^-)}\right) \tag{99}$$

The expressions for the discrete divergence and gradient operators for the plane strain assumption can be obtained by imposing $\beta = 0$ on the corresponding expressions for the axisymmetric case:

$$\widetilde{\mathscr{G}}_{x,scalar|vector} = \widetilde{\mathscr{D}}_{x,scalar|vector} = \frac{1}{2A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,x}^{(p,p^-)} + \widetilde{\boldsymbol{a}}_{c,x}^{(p,p^+)} \tag{100}$$

$$\widetilde{\mathscr{G}}_{y,scalar|vector} = \widetilde{\mathscr{D}}_{y,scalar|vector} = \frac{1}{2A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,y}^{(p,p^-)} + \widetilde{\boldsymbol{a}}_{c,y}^{(p,p^+)} \tag{101}$$

where

$$\widetilde{\boldsymbol{a}}_{c,x}^{(p,p^\pm)} = l^{(p,p^\pm)} n_{c,x}^{(p,p^\pm)} \tag{102}$$

$$\widetilde{\boldsymbol{a}}_{c,y}^{(p,p^\pm)} = l^{(p,p^\pm)} n_{c,y}^{(p,p^\pm)} \tag{103}$$

4.3.6. Discrete Operators for the Area Weighted Scheme

The discrete operators discussed thus far are pertinent to the control volume formulation. However, as it has been mentioned before, what is actually solved are the area weighted representation of the governing equations. Therefore, it is necessary to formulate the discrete version corresponding to the area weighted representation of the divergence and gradient operators. This constitutes the topic of discussion for this section.

Defining an average radius R_c as the ratio between the volume and planar area of the cell,

$$R_c = \frac{V_c}{A_c} \tag{104}$$

the GCL for the area weighted formulation can be re-written as:

$$\begin{aligned} \frac{dV_c}{dt} &= R_c \frac{dA_c}{dt} + A_c \frac{dR_c}{dt} && (105) \\ &= R_c \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_c^p \cdot \boldsymbol{u}^p + \beta A_c u_c && (106) \end{aligned}$$

where Eq (96) is used for deriving the last expression. u_c in the above equation is the volume averaged horizontal component of the velocity vector for the cell c. Analogous to Eq (56), the following identity can be established for a constant velocity field $\boldsymbol{u}^p = \boldsymbol{u}_c = C$:

$$R_c \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_c^p = -\beta A_c \boldsymbol{r} \tag{107}$$

Utilizing the expression given in Eq (106), the area weighted representation of the discrete divergence operator for (velocity) vectors can be written as

$$[\boldsymbol{\nabla} \cdot \boldsymbol{u}]_c^{AW} = \frac{1}{V_c}\frac{dV_c}{dt} = \frac{1}{A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_c^p \cdot \boldsymbol{u}^p + \beta \frac{u_c}{R_c} \tag{108}$$

Similar to the derivation presented in section 4.3.2, the expression for the area weighted formulation for the discrete gradient of vectors can be obtained by comparing the trace of the velocity gradient tensor and the divergence of velocity vector:

$$tr\left([\boldsymbol{\nabla u}]_c^{AW}\right) = [\boldsymbol{\nabla} \cdot \boldsymbol{u}]_c^{AW} \tag{109}$$

Substituting the expression from Eq (108),

$$tr\left([\boldsymbol{\nabla u}]_c^{AW}\right) = \frac{1}{A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,x}^p u^p + \frac{1}{A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,y}^p v^p + \beta \frac{u_c}{R_c} \tag{110}$$

results in the following expression for the discrete velocity gradient operator:

$$[\boldsymbol{\nabla u}]_c^{AW} = \frac{R_c}{V_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_c^p \otimes \boldsymbol{u}^p + \frac{\beta}{R_c} \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & u_c \end{pmatrix} \tag{111}$$

As a consequence of the area weighted expressions for divergence and gradient of vectors, and following the steps outlined in section 4.3.3, the discrete expression for the gradient of scalar can be expressed as:

$$[\boldsymbol{\nabla} \varphi]_c^{AW} = \frac{1}{A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,x}^p \varphi^p + \frac{1}{A_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{a}}_{c,y}^p \varphi^p \tag{112}$$

The only operator that is left to be determined is the discrete divergence of tensor operator. In contrast to the

control volume formulation for the divergence of tensor operator (section 4.3.4), the analysis for determining the discrete area weighted analog is more involved. Similar to the discussion presented in section 4.3.4, the discrete identity displayed in Eq (87) is used as the starting point. The area weighted operator for the divergence of vector formulated with respect to the dual grid reads as:

$$[\boldsymbol{\nabla}\cdot\boldsymbol{w}]_p^{AW}V^p = R^p\sum_{\mathscr{C}(p)}\widetilde{\boldsymbol{a}}_p^c\cdot\boldsymbol{w}^c+\beta w^pA^p \text{ where} \tag{113}$$

$$\widetilde{\boldsymbol{a}}_p^c = \widetilde{\boldsymbol{a}}_p^{(c,c^-)}+\widetilde{\boldsymbol{a}}_p^{(c,c^+)} \tag{114}$$

$$\widetilde{\boldsymbol{a}}_p^{(c,c^\pm)} = l^{(c,c^\pm)}\boldsymbol{n}_p^{(c,c^\pm)} \tag{115}$$

where $R^p=\frac{A^p}{V^p}$. Considering the subcell Ψ_c^p and defining a pseudo volume $\hat{V}_c^p=R^pA_c^p$, the rate of change of volume can be expressed as

$$\frac{d\hat{V}_c^p}{dt} = R^p\frac{dA_c^p}{d\boldsymbol{x}}\frac{d\boldsymbol{x}}{dt}+\frac{dR^p}{dt}A_c^p \tag{116}$$

$$= R^p(\widetilde{\boldsymbol{a}}_p^c\cdot\boldsymbol{u}^c+\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{u}^p)+\beta u^pA_c^p\boldsymbol{r} \tag{117}$$

Hence, for a constant velocity field, the identity in Eq (107) reduces to

$$R^p\widetilde{\boldsymbol{a}}_p^c = -R^p\widetilde{\boldsymbol{a}}_c^p-\beta A_c^p\boldsymbol{r} \tag{118}$$

Setting $\boldsymbol{w}=\boldsymbol{u}$ and substituting the above expression, the discrete gradient and divergence of vector operators with respect to the dual grid can be re-written as:

$$[\boldsymbol{\nabla}\cdot\boldsymbol{u}]_p^{AW}V^p = -\sum_{\mathscr{C}(p)}(R^p\widetilde{\boldsymbol{a}}_c^p+\beta A_c^p\boldsymbol{r})\cdot\boldsymbol{u}_c+\beta u^p\sum_{\mathscr{C}(p)}A_c^p \tag{119}$$

$$[\boldsymbol{\nabla u}]_p^{AW}V^p = -\sum_{\mathscr{C}(p)}\begin{pmatrix}\left(R^p\widetilde{\boldsymbol{a}}_{c_x}^p+\beta A_c^p\right)u_c & 0.5R^p\widetilde{\boldsymbol{a}}_{c_y}^pu_c+0.5\left(R^p\widetilde{\boldsymbol{a}}_{c_x}^p+\beta A_c^p\right)v_c & 0\\ 0.5R^p\widetilde{\boldsymbol{a}}_{c_y}^pu_c+0.5\left(R^p\widetilde{\boldsymbol{a}}_{c_x}^p+\beta A_c^p\right)v_c & R^p\widetilde{\boldsymbol{a}}_{c_y}^pv_c & 0\\ 0 & 0 & -A_c^pu^p\end{pmatrix} \tag{120}$$

Therefore, the scalar product between the stress and velocity gradient tensor can be expressed as

$$[\boldsymbol{\nabla w}]_p:\boldsymbol{\sigma}^pV^p = -R^p\sum_{\mathscr{C}(p)}\left\{\left(\widetilde{\boldsymbol{a}}_{c_x}^p\sigma_{xx}^p+\widetilde{\boldsymbol{a}}_{c_y}^p\sigma_{xy}^p\right)u_c+\left(\widetilde{\boldsymbol{a}}_{c_x}^p\sigma_{xy}^p+\widetilde{\boldsymbol{a}}_{c_y}^p\sigma_{yy}^p\right)v_c+\beta A_c^p\left(\sigma_{xx}^pu_c-u^p\sigma_{zz}^p+\sigma_{xy}^pv_c\right)\right\} \tag{121}$$

Using the fact that $\sum_p V^p=\sum_c V_c$, an expression for R^p in terms of R_c can be obtained as follows:

$$\sum_c V_c = \sum_c R_c\sum_{\mathscr{P}(c)}A_c^p \tag{122}$$

$$= \sum_p\sum_{\mathscr{C}(p)}R_cA_c^p \tag{123}$$

$$= \sum_p R^pA^p \tag{124}$$

$$R^p = \frac{1}{\sum_{\mathscr{C}(p)}A_c^p}\sum_{\mathscr{C}(p)}R_cA_c^p \tag{125}$$

Performing a global sum over the vertices of the mesh, the following identity can be established:

$$\begin{aligned}\sum_p R^p &= \sum_p \frac{1}{\sum_{\mathscr{C}(p)} A_c^p} \sum_{\mathscr{C}(p)} R_c A_c^p && (126)\\ &= \sum_c \frac{R_c}{\sum_{\mathscr{P}(c)} A_c^p} \sum_{\mathscr{P}(c)} A_c^p && (127)\\ &= \sum_c R_c && (128)\end{aligned}$$

Substituting for R^p in Eq (121) and performing the sum over all vertices of the mesh yields:

$$\sum_p [\boldsymbol{\nabla w}]_p : \boldsymbol{\sigma}^p V^p = \begin{array}{c} \sum_p \frac{-1}{\sum_{\mathscr{C}(p)} A_c^p} \left(\sum_{\mathscr{C}(p)} R_c A_c^p\right) \sum_{\mathscr{C}(p)} \left\{ \left(\widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xx}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{xy}^p\right) u_c + \left(\widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xy}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{yy}^p\right) v_c \right\} + \\ \beta \sum_{\mathscr{C}(p)} \left\{ A_c^p \left(\sigma_{xx}^p u_c - u^p \sigma_{zz}^p + \sigma_{xy}^p v_c\right)\right\} \end{array} \quad (129)$$

Interchanging the summation from vertices to cell and using Eq (128) results in the following expression:

$$\sum_p [\boldsymbol{\nabla w}]_p : \boldsymbol{\sigma}^p V^p = \begin{array}{c} -\sum_c R_c \sum_{\mathscr{P}(c)} \left\{ \left(\widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xx}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{xy}^p\right) u_c + \left(\widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xy}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{yy}^p\right) v_c \right\} + \\ \beta \sum_{\mathscr{P}(c)} A_c^p \left(\sigma_{xx}^p - \sigma_{zz}^p\right) u_c + \left(\sigma_{xy}^p v_c\right) \end{array} \quad (130)$$

In the above equation, the following relation due to the definition of volume averaged quantity (Φ_c) is used:

$$\begin{aligned}\sum_c \Phi_c A_c &= \sum_c \int_{A_c} \Phi dA && (131)\\ &= \sum_p \int_{A_p} \Phi dA && (132)\\ &= \sum_p \Phi^p A^p && (133)\\ &= \sum_c \sum_{\mathscr{P}(c)} \Phi^p A_c^p && (134)\\ \Phi_c &= \frac{1}{A_c} \sum_{\mathscr{P}(c)} \Phi^p A_c^p && (135)\end{aligned}$$

Therefore, employing Eq (87) and comparing the like terms, the discrete expression for the divergence of tensor is obtained:

$$[\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}]_c^{AW} = \frac{R_c}{V_c} \sum_{\mathscr{P}(c)} \begin{pmatrix} \widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xx}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{xy}^p \\ \widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xy}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{yy}^p \end{pmatrix} + \beta \frac{A_c^p}{V_c} \begin{pmatrix} \sigma_{xx}^p - \sigma_{zz}^p \\ \sigma_{xy}^p \end{pmatrix} \quad (136)$$

Alternatively, as a result of Eq (135), the above expression can be further simplified as follows:

$$[\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}]_c^{AW} = \frac{1}{A_c} \sum_{\mathscr{P}(c)} \begin{pmatrix} \widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xx}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{xy}^p \\ \widetilde{\boldsymbol{a}}_{c_x}^p \sigma_{xy}^p + \widetilde{\boldsymbol{a}}_{c_y}^p \sigma_{yy}^p \end{pmatrix} + \frac{\beta}{R_c} \begin{pmatrix} \sigma_{xx_c} - \sigma_{zz_c} \\ \sigma_{xy_c} \end{pmatrix} \quad (137)$$

As in the case of the control volume formulation, the exact representation of the functional form of the continuum expression given in Eq (9), in the discrete versions displayed in Eqs (136) & (137), is recognized. The discrete divergence operator is derived without referring to the continuum equation. Furthermore, the consistency and compatibility with the area weighted discrete gradient of scalar and vector operators is apparent.

4.4. Uniqueness of Mimetic Formulations, Quadrature Points and the Discrete Conservation Laws

Formulation	CV - Axisymmetric	AW - Axisymmetric	CV - Plane Strain
Summation	$\frac{1}{V_c}\sum_{\mathscr{P}(c)}$	$\frac{1}{V_c}\sum_{\mathscr{P}(c)}$	$\frac{1}{A_c}\sum_{\mathscr{P}(c)}$
$[\boldsymbol{\nabla}\cdot\boldsymbol{u}]_c$	$\boldsymbol{a}_c^p\cdot\boldsymbol{u}^p$	$R_c\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{u}^p+\beta A_c u_c$	$\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{u}^p$
$[\boldsymbol{\nabla}\varphi]_c$	$\boldsymbol{a}_c^p\varphi_c^p-A_c^p\varphi_c$	$R_c\widetilde{\boldsymbol{a}}_c^p\varphi_c^p$	$\widetilde{\boldsymbol{a}}_c^p\varphi_c^p$
$[\boldsymbol{\nabla}\boldsymbol{u}]_c$	$\begin{pmatrix}(a_{c_x}^p-A_c^p)u^p & a_{c_y}^p u^p & 0\\ a_{c_x}^p v^p & a_{c_y}^p v^p & 0\\ 0 & 0 & A_c^p u^p\end{pmatrix}$	$R_c\widetilde{\boldsymbol{a}}_c^p\otimes\boldsymbol{u}^p+\beta A_c u_c\boldsymbol{r}\otimes\boldsymbol{r}$	$\widetilde{\boldsymbol{a}}_c^p\otimes\boldsymbol{u}^p$
$[\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}]_c$	$\boldsymbol{a}_c^p\cdot\boldsymbol{\sigma}_c^p-A_c^p\sigma_{zz}^p$	$R_c\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{\sigma}_c^p+\beta A_c\begin{pmatrix}\sigma_{xx_c}-\sigma_{zz_c}\\ \sigma_{xy_c}\end{pmatrix}$	$\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{\sigma}_c^p$

Table 1: Summary of discrete gradient and divergence operators for the control volume and area weighted representations.

In the preceding sections, the discrete versions of the gradient and divergence operators that constitute the governing conservation laws are derived. A summary of the functional form of these operators is displayed in Table 1. As mentioned before, existing cell-centered schemes proposed in [15, 16, 17, 19, 20] are constructed using the Green's integral identity, in combination with GCL, for approximating the discrete operators. Here, the mimetic theory is invoked to demonstrate consistency and compatibility of such formulations with the continuum principles. When Green's integral identity is used for discretizing the governing conservation laws, there are no underpinning rules for defining the quadrature points for integration. For instance, the quadrature points for the formulations presented in [15, 16, 19, 20] lie at the vertices of the cell whereas in [17] the quadrature points are assumed to reside at the half-edges. It is argued in [20], that in light of the importance in preserving the angular momentum conservation principle (*i.e.* to ensure that the stress tensor remains symmetric), it is required to use the vertices as the quadrature points for integration. As it is evident from the analysis presented in the previous sections, this claim can be substantiated using the mimetic theory. In order for the compatibility and consistency conditions to remain valid it is found necessary for the quadrature points for scalars, vectors and tensors to be coincident. Furthermore, since the kinematic condition together with GCL constrains the functional form of the divergence operator, the quadrature points must be defined at the vertices of the mesh. Nevertheless, both versions of the cell-centered formulations have been applied with widespread success.

Examining Table 1, it can be inferred that the velocity field is distinctly defined at the quadrature points (vertices) whereas the scalar and stress field are approximated at the vertex for a given cell (alternatively at the subcell) *i.e.* $\boldsymbol{\sigma}_c^p$ as opposed to $\boldsymbol{\sigma}^p$. This is because, the kinematic condition demands a unique velocity field to be defined at the vertices. Due to the lack of similar constraint(s) on the scalar and stress fields, there exist an additional degree of freedom in defining these variables at the quadrature points. Nevertheless, since in the cell centered formulation the variables are stored and evolved at the element centroid, the scalar, vector and tensor fields required to evolve the governing equations must be suitably approximated at the vertices. As will be shown in the following section, the approximation at the vertices are made by introducing additional constraints on the scalar/tensor quantities.

Utilizing the expressions from Table 1 and substituting in the governing equations, the evolution equations in the discrete space are given as follows:

$$m_c\frac{d\boldsymbol{u}_c}{dt}-\frac{R_c}{2}\sum_{\mathscr{P}(c)}R_c\widetilde{\boldsymbol{a}}_c^p\cdot\boldsymbol{\sigma}_c^p+\beta A_c\begin{pmatrix}\sigma_{xx_c}-\sigma_{zz_c}\\ \sigma_{xy_c}\end{pmatrix}=0 \tag{138}$$

$$m_c\frac{dE_c}{dt}-\frac{1}{2}\sum_{\mathscr{P}(c)}\boldsymbol{a}_c^p\cdot\boldsymbol{\sigma}_c^p\cdot\boldsymbol{u}^p=0 \tag{139}$$

$$m_c\left(\frac{d\boldsymbol{S}_c}{dt}+\boldsymbol{S}_c\boldsymbol{\Omega}_c-\boldsymbol{\Omega}_c\boldsymbol{S}_c-G\sum_{\mathscr{P}(c)}\left(\widetilde{\boldsymbol{a}}_c^p\otimes\boldsymbol{u}^p+\boldsymbol{u}^p\otimes\widetilde{\boldsymbol{a}}_c^p-\frac{2}{3}\left(\boldsymbol{u}^p\cdot\widetilde{\boldsymbol{a}}_c^p+\beta A_c u_c\right)\boldsymbol{I}\right)\right)=0 \tag{140}$$

The governing equations presented above can be re-written in a compact form by introducing the corner traction

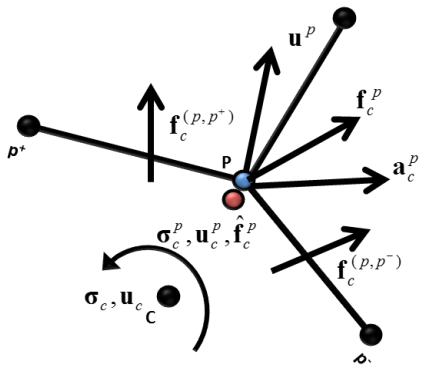


Figure 4: The corner traction force vector $\boldsymbol{f}_c^p$ is the sum of the half edge traction force vectors $\boldsymbol{f}_c^{(p,p^{\pm})}$. Also displayed in the figure are the corner area vector $\boldsymbol{a}_c^p$ and the traction force vector $\widehat{\boldsymbol{f}_c^p} = \boldsymbol{\sigma}_c^p \cdot \boldsymbol{a}_c^p$ such that $\widehat{\boldsymbol{f}_c^p} \stackrel{?}{=} \boldsymbol{f}_c^p; \boldsymbol{\sigma}_c^p = \boldsymbol{\sigma}_c(\boldsymbol{x}_c)$ for first-order and $\boldsymbol{\sigma}_c^p = \boldsymbol{\sigma}_c(\boldsymbol{x}^p)$ for second-order accuracy in space.

force vectors $\boldsymbol{f}_c^p$ & $\widetilde{\boldsymbol{f}}_c^p$ defined as:

$$\boldsymbol{f}_c^p = \boldsymbol{a}_c^p \cdot \boldsymbol{\sigma}_c^p \tag{141}$$

$$\widetilde{\boldsymbol{f}}_c^p = R_c \widetilde{\boldsymbol{a}}_c^p \cdot \boldsymbol{\sigma}_c^p \tag{142}$$

With the above definition for $\boldsymbol{f}_c^p$ & $\widetilde{\boldsymbol{f}}_c^p$, the momentum and total energy equations can be expressed in the following concise form:

$$m_c \frac{d\boldsymbol{u}_c}{dt} - \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{f}}_c^p + \boldsymbol{f}_{HOOP_c} = 0 \tag{143}$$

$$m_c \frac{dE_c}{dt} - \sum_{\mathscr{P}(c)} \boldsymbol{f}_c^p \cdot \boldsymbol{u}^p = 0 \tag{144}$$

In the above equations, the contribution due to the hoop stress term is included in the momentum equation:

$$\boldsymbol{f}_{HOOP_c} = \beta A_c \begin{pmatrix} \sigma_{xx_c} - \sigma_{zz_c} \\ \sigma_{xy_c} \end{pmatrix} \tag{145}$$

$$= \beta \sum_{\mathscr{P}(c)} A_c^p \begin{pmatrix} \sigma_{xx_c} - \sigma_{zz_c} \\ \sigma_{xy_c} \end{pmatrix} \tag{146}$$

$$= \sum_{\mathscr{P}(c)} \boldsymbol{f}_{HOOP_c}^p \tag{147}$$

where Eq (146) is a consequence of the definition of the volume averaged quantities. $\boldsymbol{f}_{HOOP_c}^p$ in Eq (147) is the subcell contribution of the hoop stress term. The representation of the subcell contribution of the hoop stresses will be useful when the conservation properties are analyzed for the system of governing equations. The significance of the formulation presented above is that the structure of the underlying gas dynamic solver is retained and the material strength terms (corresponding to stress tensors) are infused in to the corner traction force vectors. Examining Eqs (143), (144) & (140), the only unknowns that are yet to be determined are the corner traction force vectors $\widetilde{\boldsymbol{f}}_c^p$ & $\boldsymbol{f}_c^p$ (or the half edge traction vectors $\widetilde{\boldsymbol{f}}^{(p,p^{\pm})}$ in the case of the formulation due to [17]) and the vertex velocity vectors $\boldsymbol{u}^p$. The unknowns are determined by invoking the total energy conservation along with the increase in entropy principle

as elucidated in the following sections.

Remark 5. The governing equations corresponding to the formulation given in [17] take the form given below:

$$m_c \frac{d\boldsymbol{u}_c}{dt} - \frac{R_c}{2} \sum_{\mathscr{P}(c)} \left(\widetilde{\boldsymbol{a}}_c^{(p,p^-)} \cdot \boldsymbol{\sigma}_c^{(p,p^-)} + \widetilde{\boldsymbol{a}}_c^{(p,p^+)} \cdot \boldsymbol{\sigma}_c^{(p,p^+)} \right) + \beta A_c \begin{pmatrix} \sigma_{xx_c} - \sigma_{zz_c} \\ \sigma_{xy_c} \end{pmatrix} = 0 \tag{148}$$

$$m_c \frac{dE_c}{dt} - \frac{1}{2} \sum_{\mathscr{P}(c)} \left(\boldsymbol{a}_c^{(p,p^-)} \cdot \boldsymbol{\sigma}_c^{(p,p^-)} + \boldsymbol{a}_c^{(p,p^+)} \cdot \boldsymbol{\sigma}_c^{(p,p^+)} \right) \cdot \boldsymbol{u}^p = 0 \tag{149}$$

$$m_c \left(\frac{d\boldsymbol{S}_c}{dt} + \boldsymbol{S}_c \boldsymbol{\Omega}_c - \boldsymbol{\Omega}_c \boldsymbol{S}_c - G \sum_{\mathscr{P}(c)} \left(\widetilde{\boldsymbol{a}}_c^p \otimes \boldsymbol{u}^p + \boldsymbol{u}^p \otimes \widetilde{\boldsymbol{a}}_c^p - \frac{2}{3} \left(\boldsymbol{u}^p \cdot \widetilde{\boldsymbol{a}}_c^p + \beta A_c u_c \right) \boldsymbol{I} \right) \right) = 0 \tag{150}$$

In this case, the corner traction force vectors $\boldsymbol{f}_c^p$ & $\widetilde{\boldsymbol{f}}_c^p$ are defined as:

$$\boldsymbol{f}_c^p = R^{(p,p^+)} \widetilde{\boldsymbol{f}}_c^{(p,p^+)} + R^{(p,p^-)} \widetilde{\boldsymbol{f}}_c^{(p,p^-)} \tag{151}$$

$$\widetilde{\boldsymbol{f}}_c^p = \frac{R_c}{2} \left(\widetilde{\boldsymbol{f}}_c^{(p,p^+)} + \widetilde{\boldsymbol{f}}_c^{(p,p^-)} \right) \text{ where} \tag{152}$$

$$\widetilde{\boldsymbol{f}}_c^{(p,p^\pm)} = \frac{1}{2} \widetilde{\boldsymbol{a}}_c^{(p,p^\pm)} \cdot \boldsymbol{\sigma}_c^{(p,p^\pm)} \tag{153}$$

Therefore, with the above definitions the momentum and total energy equations can be expressed in the same form as in Eqs (143) & (144). The unknowns in this case are the corner traction force vectors $\widetilde{\boldsymbol{f}}_c^p$ and the half edge traction vectors $\widetilde{\boldsymbol{f}}^{(p,p^\pm)}$, and the vertex velocity vectors $\boldsymbol{u}^p$. Thus, the ability of the current formulation in reproducing existing cell-centered schemes is evident.

4.4.1. Consequence of Conservation Laws

As mentioned before, the unknown corner traction force and vertex velocity vectors required to evolve the governing equations in time are determined by invoking the statements of conservation of momentum and total energy in the discrete space. Following the arguments presented in [11, 12, 45] and adding the contribution from all cells constituting the discrete space results in the following relations:

$$\sum_c m_c \frac{d\boldsymbol{u}_c}{dt} = \sum_c \left(\sum_{\mathscr{P}(c)} \left(\widetilde{\boldsymbol{f}}_c^p - \boldsymbol{f}_{HOOP_c} \right) \right) = \sum_p \left(\sum_{\mathscr{C}(p)} \left(\widetilde{\boldsymbol{f}}_c^p - \boldsymbol{f}_{HOOP}^p \right) \right) = \sum_{p \in \partial\varpi} \boldsymbol{f}_{bdry}^p \tag{154}$$

$$\sum_c m_c \frac{dE_c}{dt} = \sum_c \sum_{\mathscr{P}(c)} \boldsymbol{f}_c^p \cdot \boldsymbol{u}^p = \sum_p \boldsymbol{u}^p \cdot \left(\sum_{\mathscr{C}(p)} \boldsymbol{f}_c^p \right) = \sum_{p \in \partial\varpi} \boldsymbol{f}_{bdry}^p \cdot \boldsymbol{u}^p \tag{155}$$

where $\boldsymbol{f}_{HOOP}^p$ is an approximation of the hoop stress contribution at the vertices and $\boldsymbol{f}_{bdry}^p$ is the external boundary force that is defined on the vertices of the domain boundaries. It can be recognized that the final expressions in the above equations are obtained by interchanging the summation between cells and vertices. The outer summation in the last expressions is performed over all vertices $p \in \varpi(t)$. Recalling the statements of conservation of momentum and total energy [11, 12, 17], the following can be inferred for the vertices belonging to the interior of the domain:

$$\sum_{\mathscr{C}(p)} \widetilde{\boldsymbol{f}}_c^p = \boldsymbol{f}_{HOOP}^p \tag{156}$$

$$\sum_{\mathscr{C}(p)} \boldsymbol{f}_c^p = 0 \tag{157}$$

where Eq (156) is a consequence of momentum conservation and Eq (157) is the consequence of energy conservation. Eq (157) is the statement of compatibility with total energy conservation, analogous to the compatible staggered grid formulations [11, 12]. In other words, Eq (157) states that the sum of the corner traction force vectors in the interior vertices is null. Eqs (157) & (156) serve as constraints that must be satisfied by the not yet determined corner traction force vectors $\widetilde{\boldsymbol{f}}_c^p$ & $\boldsymbol{f}_c^p$. For the plane strain condition, $\boldsymbol{f}_{HOOP}^p = 0$ and therefore Eq (156) reduces to Eq (157). However unlike in the plane strain case, as it has been pointed out in [11, 12, 17], enforcing both constraints simultaneously is not feasible for axisymmetric geometries. Therefore, similar to the compatible formulation [11, 12], the constraint due to the total energy conservation is utilized in determining the corner traction force vectors $\widetilde{\boldsymbol{f}}_c^p$ & $\boldsymbol{f}_c^p$. The condition in Eq (157) can be viewed as a sufficient condition for determining the corner traction force vectors. At this juncture, it is worth pointing out that neither such constraints are derived for the vertex velocity vectors $\boldsymbol{u}^p$ nor condition(s) relating $\boldsymbol{f}_c^p$ and $\boldsymbol{u}^p$ are established. The kinematic condition is the only other constraint for $\boldsymbol{u}^p$. In essence, Eq (157) indicates that the conservation of total energy is trivially satisfied for any arbitrary $\boldsymbol{u}^p$. Therefore, in order to determine a physically meaningful and geometrically (GCL) consistent velocity field $\boldsymbol{u}^p$, an additional constraint in the form of increase in entropy principle (and consequently dissipation of internal energy to kinetic energy condition) is invoked. This constitutes the topic of discussion for the remaining part of this section.

4.4.2. Riemann Solver for Determining the Corner Traction Vectors

In the preceding section, constraints that must be enforced on the corner traction force vectors were derived. However, such conditions do not exist for the vertex velocity vectors. Therefore, it is unclear how the unknown quantities can be evaluated while simultaneously imposing the aforementioned constraints on the corner traction force vectors. The task set forth in determining the unknown corner traction force and vertex velocity vectors can be immensely simplified if a relation between the two quantities can be established. Traditionally, this task is accomplished by relating the two unknown vectors via viscosity/dissipation models. The purpose of this section is to formulate a viscosity model which in combination with the constraints introduced in the preceding section yields a closed form solution for the unknown vectors.

It is recognized that a traction force vector $\widehat{\boldsymbol{f}_c^p}$ can be computed at each vertex $p \in \mathscr{P}(c)$ such that

$$\widehat{\boldsymbol{f}_c^p} = \boldsymbol{\sigma}_c \cdot \boldsymbol{a}_c^p \tag{158}$$

$\widehat{\boldsymbol{f}_c^p}$ is not always equal to the corner traction force vector $\boldsymbol{f}_c^p$, particularly in the presence of shocks and other discontinuities (Figure 4). In the absence of discontinuities, the equality of the two traction forces can be safely assumed. Typically, Riemann problems are solved to automatically determine the jump in the force vectors. Therefore, following the conventional approach, a Riemann problem is erected at each vertex to determine the jump:

$$\delta\boldsymbol{f}_c^p = \boldsymbol{f}_c^p - \widehat{\boldsymbol{f}_c^p} \tag{159}$$

Solution to the Riemann problem are generally of the form [46]:

$$\delta\boldsymbol{f}_c^p = \boldsymbol{\Theta}_c^p \delta\boldsymbol{u}_c^p \tag{160}$$

where

$$\delta\boldsymbol{u}_c^p = \boldsymbol{u}^p - \boldsymbol{u}_c \tag{161}$$

is the jump in velocity across the discontinuity and $\boldsymbol{\Theta}_c^p$ is the viscosity tensor, that ensures proper dissipation of energy within the discontinuity. The matrix $\boldsymbol{\Theta}_c^p$ can also be viewed as a transformation matrix that rotates $\delta\boldsymbol{f}_c^p$ to be aligned with $\delta\boldsymbol{u}_c^p$ and vice-versa. As a result, the viscosity tensor must be non-singular in order to permit the inverse transformation. As will be shown later, the transformation expressed in Eq (160) will satisfy the requirement imposed by the second law of thermodynamics. At this point, the existence of $\boldsymbol{\Theta}_c^p$ is merely assumed and the conditions for its definition will be discussed by revisiting the total energy equation and increase in entropy principle.

4.5. Consequence of Second Law of Thermodynamics - The Statement of Increase in Entropy Principle

Following the analysis presented in [20] and inserting the expressions for $\boldsymbol{f}_c^p$ & $\boldsymbol{u}^p$ (Eqs (159) & (161)) in the total energy equation (Eq (144)) yields:

$$\begin{aligned} m_c \frac{dE_c}{dt} &= \sum_{\mathscr{P}(c)} (\widehat{\boldsymbol{f}_c^p} + \delta \boldsymbol{f}_c^p) \cdot (\boldsymbol{u}_c + \delta \boldsymbol{u}_c^p) \\ &= \left(\sum_{\mathscr{P}(c)} \boldsymbol{f}_c^p \cdot \boldsymbol{u}_c \right) + \left(\sum_{\mathscr{P}(c)} \widehat{\boldsymbol{f}_c^p} \cdot \delta \boldsymbol{u}_c^p \right) + \left(\sum_{\mathscr{P}(c)} \delta \boldsymbol{f}_c^p \cdot \delta \boldsymbol{u}_c^p \right) \end{aligned} \tag{162}$$

where the term in the first parenthesis, in the Right Hand Side (RHS) of Eq (162), corresponds to the evolution equation for kinetic energy. This term, in turn, is obtained by the dot product between $\boldsymbol{u}_c$ and the momentum equation:

$$m_c \boldsymbol{u}_c \cdot \frac{d\boldsymbol{u}_c}{dt} = \sum_{\mathscr{C}(p)} \boldsymbol{f}_c^p \cdot \boldsymbol{u}_c - \beta A_c \sigma_{zz_c} u_c. \tag{163}$$

It is important to point out that the divergence operator utilized in the above equation corresponds to that of the control volume representation of the divergence of stress tensor. Subtracting the evolution equation for kinetic energy from the total energy equation, an evolution equation for internal energy can be readily obtained:

$$m_c \frac{de_c}{dt} = \left(\sum_{\mathscr{P}(c)} \widehat{\boldsymbol{f}_c^p} \cdot \delta \boldsymbol{u}_c^p \right) + \left(\sum_{\mathscr{P}(c)} \delta \boldsymbol{f}_c^p \cdot \delta \boldsymbol{u}_c^p \right) + \beta A_c \sigma_{zz_c} u_c \tag{164}$$

As noted in [20], the advantage of computing internal energy through an evolution equation of the form shown above is that the round off numerical errors that often results in negative internal energy for kinetic energy dominated flows can be entirely averted. Nevertheless, for the problems attempted in this work, it was found sufficient to compute internal energy from the total energy update even for flows that are primarily driven by initial kinetic energy of the system.

According to Gibbs relation, the statement of thermodynamic consistency can be written as [25]:

$$m_c T_c \frac{ds_c}{dt} = m_c \left(\frac{de_c}{dt} - \boldsymbol{\sigma}_c : \boldsymbol{D}_c \right) \tag{165}$$

where s_c & T_c denote the volume averaged entropy and temperature of the cell c. The representation of stress power given above can be deduced using the vector identity expressed in Eq (11) as follows:

$$\boldsymbol{\sigma}_c : \boldsymbol{D}_c = \boldsymbol{\nabla} \cdot (\boldsymbol{\sigma} \boldsymbol{u}) - \boldsymbol{u} \cdot (\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}) \tag{166}$$

$$= \frac{1}{V_c} \left(\sum_{\mathscr{P}(c)} \widehat{\boldsymbol{f}}_c^p \cdot \boldsymbol{u}^p - \boldsymbol{u}_c \cdot \sum_{\mathscr{P}(c)} \widehat{\boldsymbol{f}}_c^p + \beta A_c \sigma_{zz_c} u_c \right) \tag{167}$$

$$= \frac{1}{V_c} \left(\sum_{\mathscr{P}(c)} \widehat{\boldsymbol{f}}_c^p \cdot \delta \boldsymbol{u}_c^p + \beta A_c \sigma_{zz_c} u_c \right) \tag{168}$$

Substituting the expressions for internal energy and stress power in Eq (165), the rate of change of entropy of the system can be written as

$$m_c T_c \frac{ds_c}{dt} = \delta \boldsymbol{f}_c^p \cdot \delta \boldsymbol{u}_c^p \tag{169}$$

This equation can be simplified by incorporating the solution from Riemann problem (Eq (160)) to yield:

$$m_c T_c \frac{ds_c}{dt} = \sum_{\mathscr{P}(c)} \delta \boldsymbol{f}_c^p \cdot \delta \boldsymbol{u}_c^p = \sum_{\mathscr{P}(c)} (\boldsymbol{\Theta}_c^p \delta \boldsymbol{u}_c^p) \cdot \delta \boldsymbol{u}_c^p \tag{170}$$

Therefore, for any positive semi-definite matrix $\boldsymbol{\Theta}_c^p$ (for a detailed proof the reader is referred to [47]), the following inequality holds true:

$$m_c T_c \frac{ds_c}{dt} = \sum_{\mathscr{P}(c)} (\boldsymbol{\Theta}_c^p \delta \boldsymbol{u}_c^p) \cdot \delta \boldsymbol{u}_c^p \geq 0 \tag{171}$$

Thus the statement of thermodynamic consistency condition serves as an additional constraint for determining the viscosity tensor $\boldsymbol{\Theta}_c^p$. The expressions for determining the vertex velocities can be obtained by combining Eqs (157) & (160) to yield:

$$\boldsymbol{\Theta}^p \boldsymbol{u}^p \quad = \quad - \left\{ \sum_{\mathscr{C}(p) \ \forall \ p \ni \partial \varpi} \left(\widehat{\boldsymbol{f}_c^p} - \boldsymbol{\Theta}_c^p \boldsymbol{u}_c \right) \right\} \tag{172}$$

where $\boldsymbol{\Theta}^p = \sum_{\mathscr{C}(p)} \boldsymbol{\Theta}_c^p$. Once the vertex velocities $\boldsymbol{u}^p$ are computed, Eq (160) can be utilized to determine the corner traction force vector $\boldsymbol{f}_c^p$. Therefore, the unknowns that are yet to be determined are the corner traction force vector $\widetilde{\boldsymbol{f}}_c^p$ required to evolve the governing equation and the viscosity matrix $\boldsymbol{\Theta}_c^p$

Remark 6. It is worth mentioning that, in the analysis presented above, the control volume formulation of the momentum equation (Eq (163)) is employed instead of the area-weighted counterpart. This is a well known source of error for finite-volume Lagrangian schemes formulated in cylindrical coordinate systems. A detailed discussion on the consequence of the above formulation and the importance of preserving symmetry can be found in [10, 48, 17], in the context of gas dynamics schemes. It is recognized here that the analysis presented in [10, 48, 17] can be naturally extended to the formulation considered in this work.

4.5.1. Determining the Unknowns

As shown in the previous section, the viscosity matrix is related to the corner traction force vector $\boldsymbol{f}_c^p$ and the vertex velocity vector $\boldsymbol{u}^p$. Determination of the viscosity matrix will lead to the deduction of unique nodal force and velocity vectors. However, there is another quantity, namely the corner traction force vector $\widetilde{\boldsymbol{f}}_c^p$, that needs to determined as demanded by the momentum evolution equation. Thus in order to complete the the analysis, a relationship between $\widetilde{\boldsymbol{f}}_c^p$ and $\boldsymbol{\Theta}_c^p$ has to be established. Recalling the definition of $\boldsymbol{f}_c^p$ from Eq (141) and substituting the same in the definition of the Riemann problem displayed in Eq (160), the following relation can be established:

$$\boldsymbol{f}_c^p = \frac{1}{2} \left(R_c^{(p,p^+)} \widetilde{\boldsymbol{a}}_c^{(p,p^+)} + R_c^{(p,p^-)} \widetilde{\boldsymbol{a}}_c^{(p,p^-)} \right) \cdot \boldsymbol{\sigma}_c^p = \frac{1}{2R_c} \left(R_c^{(p,p^+)} \widetilde{\boldsymbol{f}}_c^{(p,p^+)} + R_c^{(p,p^-)} \widetilde{\boldsymbol{f}}_c^{(p,p^-)} \right) \tag{173}$$

In contrast to the formulation presented in [17], it is important to recognize that the half-edge traction vectors ($\widetilde{\boldsymbol{f}}_c^{(p,p^\pm)}$) are projections of a single stress-tensor $\boldsymbol{\sigma}_c^p$ defined at each subcell. Therefore, the following can be inferred by solving a Riemann problem for the half-edge traction vectors $\widetilde{\boldsymbol{f}}_c^{(p,p^\pm)}$:

$$\widetilde{\boldsymbol{f}}_c^{(p,p^\pm)} = R_c \widetilde{\boldsymbol{a}}_c^{(p,p^\pm)} \cdot \boldsymbol{\sigma}_c + R_c \boldsymbol{\Theta}_c^{(p,p^\pm)} \delta \boldsymbol{u}_c^p \tag{174}$$

Thus, once the expression for the viscosity matrix $\boldsymbol{\Theta}_c^{(p,p^\pm)}$ is determined, the unknown quantities (including the corner traction force vectors $\widetilde{\boldsymbol{f}}_c^{(p,p^\pm)}$) can be easily computed with the following assumption implicitly implied:

$$\boldsymbol{\Theta}_c^p = \boldsymbol{\Theta}_c^{(p,p^+)} + \boldsymbol{\Theta}_c^{(p,p^-)} \tag{175}$$

As pointed in [47], several choices exist that could be used for determining the viscosity matrix. Potentially, any non-singular square matrix (2×2 in two-dimensions and 3×3 in three-dimensions) that is positive semi-definite can be used to enforce the transformation expressed in Eq (160). Existing viscosity models can be classified into vertex or edge viscosity models, based on the direction which the viscosity matrix is aligned. These viscosity models are summarized in Table 2. Different flavors of cell-centered formulations can be recovered by varying the viscosity

Viscosity model	Citation	Expression
Edge viscosity	[17]	$\boldsymbol{\Theta}_c^{(p,p^{\pm})} = Z_c^{(p,p^{\pm})} \widetilde{\boldsymbol{a}}_c^{(p,p^{\pm})} \otimes \boldsymbol{n}_c^{(p,p^{\pm})}$
Vertex viscosity	[15, 16]	$\boldsymbol{\Theta}_c^{(p,p^{\pm})} = \frac{1}{2} Z_c^{(p,p^{\pm})} \widetilde{\boldsymbol{a}}_c^{(p,p^{\pm})} \otimes \left(\boldsymbol{n}_c^{(p,p^{+})} + \boldsymbol{n}_c^{(p,p^{-})} \right)$
Vertex viscosity	[20]	$\boldsymbol{\Theta}_c^{(p,p^{\pm})} = Z_c^{(p,p^{\pm})} \delta \boldsymbol{u}_c^p \cdot \widetilde{\boldsymbol{a}}_c^{(p,p^{\pm})} \boldsymbol{I}$

Table 2: Summary of existing viscosity matrix for cell-centered formulations. The expression $\delta \boldsymbol{u}_c^p = \boldsymbol{u}^p - \boldsymbol{u}_c$ is velocity difference vector and $Z_c^{(p,p^{\pm})} = \rho_c c_c$ is the acoustic impedance factor. When $Z_c^{(p,p^{\pm})} = \rho_c (c_c + \Gamma_c |(\boldsymbol{u}^p - \boldsymbol{u}_c) \cdot \boldsymbol{n}_c^{(p,p^{\pm})}|)$, the impedance model due to the Riemann solver proposed in [46] can be obtained.

model. Examining the table, the issue with the viscosity model proposed in [15, 16] can be identified. The viscosity matrices $\boldsymbol{\Theta}_c^{(p,p^{\pm})}$, in this case, is not symmetric. However, the $\boldsymbol{\Theta}_c^p$, which is the sum of these matrices, is a symmetric and positive semi-definite matrix. Hence, this model is not suitable for the applications of interest to this work. The edge viscosity model proposed in [17] requires the definition of half edge stress tensors and therefore shifts quadrature points to half edges. Consequently, the compatibility properties may not be readily extended. Therefore, in this work, a physically consistent viscosity tensor that is proposed in [20] is employed. The viscosity tensor is constructed from the vector product between the velocity difference vector $\delta \boldsymbol{u}_c^p$ and corner area vector $\boldsymbol{a}_c^p$. As a result, the viscosity tensor reduces to a diagonal matrix and hence requires no matrix inversion. Once the half edge traction vectors $\widetilde{\boldsymbol{f}}_c^{(p,p^{\pm})}$ are computed, the corner traction force vectors can be evaluated.

5. High-Order Extension

5.1. Spatial Accuracy

The solution strategy outlined in the preceding sections made no recourse to the accuracy of the scheme. To be more specific, Eq (172) can be re-written as

$$\boldsymbol{\Theta}^p \boldsymbol{u}^p \quad = \quad - \left\{ \sum_{\mathscr{C}(p) \ \forall \ p \ni \partial \varpi} \left(\widehat{\boldsymbol{f}_c^p}(\boldsymbol{x}) - \boldsymbol{\Theta}_c^p \boldsymbol{u}_c(\boldsymbol{x}) \right) \right\} \tag{176}$$

If $\widehat{\boldsymbol{f}_c^p}(\boldsymbol{x}) = \widehat{\boldsymbol{f}_c^p}(\boldsymbol{x_c})$ & $\boldsymbol{u}_c(\boldsymbol{x}) = \boldsymbol{u}_c(\boldsymbol{x}_c)$, then the resulting solution corresponds to first-order accurate solution in space. Instead, if these quantities were evaluated at $\boldsymbol{x} = \boldsymbol{x}^p$, then the accuracy of the solution corresponds to the order of accuracy with which the quantities are reconstructed within the cell. Since linear elements are employed in this work, the accuracy of the reconstruction procedure is limited to second-order. Following [49], high-order accuracy is achieved by performing a Taylor series reconstruction within the cell, *i.e.* for any quantity $\boldsymbol{\Phi} \in [\rho, \boldsymbol{u}, e, P, \boldsymbol{S}]$, the following statement for reconstruction is implied:

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{\Phi}_c + (\boldsymbol{x} - \boldsymbol{x}_c) \cdot (\psi_c \boldsymbol{\nabla \Phi}) \tag{177}$$

where $\boldsymbol{\nabla \Phi}$ is the gradient of the quantity $\boldsymbol{\Phi}$ and ψ_c is the slope limiter [31]. Traditionally, the gradient of scalar, vector and tensor quantities, required for the reconstruction procedure, are evaluated using standard techniques such as least squares [50, 51] or Green's integral theorem [25, 21]. However, in this work, a new gradient estimation technique based on mimetic formulation is proposed. Furthermore, a new physics-based limiter schemes for vectors and tensors is also proposed.

5.1.1. Mimetic Gradient Estimation Technique

In line with the central of theme of this work, a technique based on mimetic formulation for computing gradients of scalar, vector and tensor quantities is proposed. Since mimetic formulation preserves the fundamental identities of vector/tensor calculus in the discrete space, the gradients estimated using mimetic formulation may not introduce spurious modes/vortices that are otherwise not filtered by the limiter schemes. Furthermore, the numerical study conducted in this work indicate that existing techniques such as the least squares, do not perform adequately when the evolving mesh becomes skewed. For problems that are driven by the initial kinetic energy of the system, it may not be entirely possible (without mesh optimization) to not permit meshes with poor quality. Therefore, it is important to formulate a gradient estimation technique, preferably based on mimetic formulation, that is robust and independent of the quality of the underlying evolving mesh.

Recalling from section 4.3, the quadrature points for discrete divergence and gradient operators reside at the mesh vertices. Thus, in order to employ the mimetic theory for evaluating gradients, it is first necessary to design a linearity and bound preserving "lifting" procedure, for interpolating variables from the cell centers to the vertices of the mesh. At any arbitrary vertex p in the mesh, such a procedure can be defined by considering the cells $c \in \mathscr{C}(p)$. Therefore, to evaluate the quantity $\mathbf{\Phi}^p$ at the vertex p, a convex hull for the points $c \in \mathscr{C}(p)$ is defined as follows:

$$\mathbf{\Phi}^p = \alpha_c \mathbf{\Phi}_c \, \forall \, c \in \mathscr{C}(p) \mathscr{W}(p) \tag{178}$$

where

$$\alpha_c \in \mathbb{R}, \alpha_c > 0, \sum_{c \in \mathscr{C}(p)} \alpha_c = 1 \tag{179}$$

The weights α_c, employed in Eq (178), can be chosen in several different ways. Some of the options that were explored include the following:

$$\alpha_c \in \left[m_c, m_c^p, V_c, V_c^p, A_c, A_c^P, \rho_c, \rho_c^p, \sqrt{\rho_c} \,\&\, \sqrt{\rho_c^p} \right] \tag{180}$$

where m_c, V_c, A_c, ρ_c denote the mass, volume, area and density of the cell and $m_c^p, V_c^p, A_c^P, \rho_c^p$ denote the mass, volume, area and density of the subcell. For instance, an obvious choice for the weighting function is the mass m_c (m_c^p) of the cell (subcell). When $\alpha_c = m_c | m_c^p$, the remapping approach used in ALE methods [52] and the interpolation strategy employed in mesh coarsening algorithms [53] are recovered. However, such an option was found to be ineffective because the mass of the cell/subcell is a constant quantity and therefore does not reflect the dynamics of the mesh. Alternatively, setting the weights $\alpha_c = V_c | V_c^p$, accommodated for mesh quality effects but was found to be inaccurate near the axis of symmetry. Even the intermediate options, $\alpha_c = \rho_c | \rho_c^p$ or $\alpha_c = A_c | A_c^p$ was found to be inadequate. The two options that were found to be robust for several problems of interest to this work include $\alpha_c = \sqrt{\rho_c} | \sqrt{\rho_c^p}$. When $\alpha_c = \sqrt{\rho_c} | \sqrt{\rho_c^p}$, Eq (178) resembles the multi-dimensional version of the Riemann solver proposed in [54]. Thus, with $\alpha_c = \sqrt{\rho_c} | \sqrt{\rho_c^p}$, the pressure, velocity and shear stress tensor are interpolated at the vertices of the mesh using Eq (178). Once the flow quantities are projected to the vertices, the expression for discrete gradient or divergence operator can be employed to determined the gradient $\mathbf{\nabla \Phi}$.

5.1.2. Physics-Based Frame Invariant Limiter Schemes

Slope limiters are enforced to ensure monotonicity of the reconstructed field. Discretization schemes are particularly sensitive to slope limiters and therefore special care must be exercised in devising appropriate limiter schemes. In [31], frame invariant and spatial symmetry preserving limiter schemes for stress tensors were proposed and the same is used in this work. In what follows, the limiter schemes for scalar, vectors and tensor variables are briefly described.

Slope limiter for scalar variables is determined by enforcing the following monotonicity condition:

$$\Phi_{\min} \leq \Phi^p \leq \Phi_{\max} \tag{181}$$

where

$$\Phi_{\min} = \min_{n \in \mathscr{N}(c)} (\Phi_n, \Phi_c) \text{ and } \Phi_{\max} = \max_{n \in \mathscr{N}(c)} (\phi_n, \phi_c) \tag{182}$$

and Φ^p denotes the reconstructed value at the vertex $p \in \mathscr{P}(c)$. The monotonicity criterion (Eq (181)) is applied to yield:

$$\psi_c^p = \mathscr{L}\left(\frac{\phi_{\max} - \phi_c}{(\boldsymbol{x}^p - \boldsymbol{x}_c) \cdot \boldsymbol{\nabla}\phi}, 1\right) \text{ if } (\boldsymbol{x}^p - \boldsymbol{x}_c) \cdot \boldsymbol{\nabla}\phi > 0 \tag{183}$$

$$\psi_c^p = \mathscr{L}\left(\frac{\phi_{\min} - \phi_c}{(\boldsymbol{x}^p - \boldsymbol{x}_c) \cdot \boldsymbol{\nabla}\phi}, 1\right) \text{ if } (\boldsymbol{x}^p - \boldsymbol{x}_c) \cdot \boldsymbol{\nabla}\phi < 0 \tag{184}$$

where the function $\mathscr{L}(x)$ is either the Barth-Jespersen [50, 55, 56]

$$\mathscr{L}(x) = \min(x, 1.0) \tag{185}$$

or the Venkatakrishnan [57]

$$\mathscr{L}(x) = \min\left(\frac{x^2 + 2x}{x^2 + x + 1}, 1\right) \tag{186}$$

limiter function [58]. Consequently, the slope limiter ψ_c is defined as:

$$\psi_c = \min_{n \in \mathscr{P}(c)} (\psi_c^p, 1) \tag{187}$$

The slope limiter defined for scalar variables cannot be applied to individual components of vector and tensor fields. This is because, such component-wise limiters are frame dependent and do not preserve fundamental features like rotational or planar symmetry [59, 31]. Furthermore, component-wise limiter schemes produce different results for rotational and translation effects on the tensor field [58]. Lagrangian hydrodynamics are particularly sensitive to these issues as the underlying moving mesh discretization schemes are already prone to symmetry violation. Hence, the following approaches are adopted for limiting velocity vectors and shear stress tensors.

First, the slope limiter scheme for constraining shear stress tensor is presented. Following the analysis presented in [31], an inequality for a linear function $h = \sqrt{J_2}$ is used as the commencing point:

$$h_{\min} - h_c \leq h_c^p - h_c \leq h_{\max} - h_c \tag{188}$$

where J_2 is the second invariant of the shear stress tensor. Noting the definition of second invariant $J_2^p = \frac{1}{2}\boldsymbol{S}^p : \boldsymbol{S}^p$ and combining with the expression for $\boldsymbol{S}^p$ (Eq (177)), the following is obtained:

$$J_2^p = \frac{1}{2}(\boldsymbol{S}_c : \boldsymbol{S}_c + 2\boldsymbol{S}_c : \delta\boldsymbol{S}^{pc} + \delta\boldsymbol{S}^{pc} : \delta\boldsymbol{S}^{pc}) \tag{189}$$

where

$$\delta\boldsymbol{S}^{pc} = (\boldsymbol{x}^p - \boldsymbol{x}_c) \cdot \boldsymbol{\nabla}\boldsymbol{S} \tag{190}$$

Therefore,

$$h_c^p - h_c = (\alpha_c^p)^{-1}(\boldsymbol{S}^p + \boldsymbol{S}_c) : \delta\boldsymbol{S}^{pc} \tag{191}$$

where $\alpha_c^p = h_c^p + h_c$ is a positive quantity and consequently, the inequality in Eq (188) can be cast in the following form:

$$\alpha_c^p (h_c^p - h_c) \leq (\boldsymbol{S}^p + \boldsymbol{S}_c) : \delta\boldsymbol{S}^{pc} \leq \alpha_c^p (h_{\max} - h_c) \tag{192}$$

It is now possible to define a unique scalar slope limiter $\psi^p_{c_{J_2}}$ based on the invariant bounds as follows:

$$\psi^p_{c_{J_2}} = \mathscr{L}\left(\frac{h_{\max} - h_c}{h^p_c - h_c}\right) \text{ if } h^p_c - h_c > 0 \tag{193}$$

$$\psi^p_{c_{J_2}} = \mathscr{L}\left(\frac{h_{\min} - h_c}{h^p_c - h_c}\right) \text{ if } h^p_c - h_c < 0 \tag{194}$$

$$\psi^p_{c_{J_2}} = 1 \text{ otherwise} \tag{195}$$

and consequently the slope limiter for the cell is determined as

$$\psi_{c_{J_2}} = \min_{p \in \mathscr{P}(c)} \psi^p_{c_{J_2}} \tag{196}$$

As opposed to deducing a slope limiter matrix, a single scalar variable is determined for limiting the gradient of the stress tensor. The slope limiter thus defined is guaranteed to ensure monotonicity, accuracy and preserve symmetry. Since the slope limiter obtained above is a scalar variable, it is intrinsically frame invariant.

The slope limiter scheme formulated for velocity vector is a direct extension of the J_2 based slope limiter designed for stress tensor. Defining the linear field in Eq (188) as $h = \sqrt{2k}$, where $k = \frac{1}{2}\boldsymbol{u} \cdot \boldsymbol{u}$ is the kinetic energy, the slope limiter for velocity vector can be determined as follows:

$$\psi^p_{c_k} = \mathscr{L}\left(\frac{h_{\max} - h_c}{h^p_c - h_c}\right) \text{ if } h^p_c - h_c > 0 \tag{197}$$

$$\psi^p_{c_k} = \mathscr{L}\left(\frac{h_{\min} - h_c}{h^p_c - h_c}\right) \text{ if } h^p_c - h_c < 0 \tag{198}$$

$$\psi^p_{c_k} = 1 \text{ otherwise} \tag{199}$$

and consequently the slope limiter for the cell is determined as

$$\psi_{c_k} = \min_{p \in \mathscr{P}(c)} \psi^p_{c_k} \tag{200}$$

As it is evident, the slope limiter schemes defined above has close relevance to the underlying physics that is being resolved. The slope limiter based on J_2 scheme ensures that the reconstructed stress tensor satisfies the consistency condition whereas the slope limiter for velocity vector preserves the extrema in the kinetic energy field. The Galilean invariance of the limiting process follows directly from the construction procedure.

5.2. Temporal Accuracy

The evolution equations are advanced from time level n to $(n + \frac{1}{2})$ using a forward Euler scheme followed by a corrector step centered in time to determine the values at time level $n + 1$. The predictor and corrector steps are computed in two sweeps as follows:

$$(\ )^{(n+\frac{iter}{2})} = c_1(\ )^n + c_2(\ )^{(n+\frac{1}{2})} + c_3\frac{\Delta t}{2}\mathscr{F}^{(n+\frac{iter-1}{2})} \tag{201}$$

where the coefficients c_1, c_2 & c_3 take the values $1, 0$ & 1 for the predictor sweep $(iter = 1)$ and $\frac{1}{2}, \frac{1}{2},$ & $\frac{1}{2}$ for the corrector sweep $(iter = 2)$. $\mathscr{F}^{(n+\frac{iter-1}{2})}$ denotes the fluxes evaluated at the cell faces/edges corresponding to the preceding sweep.

6. Second-Order Predictor-Corrector Algorithm

To reduce the burden of algorithm development and facilitate the ease of code implementation, the summary of steps to retain second-order accuracy in both space and time is given below. For the sake of clarity of the presentation, the superscripts for source terms (RHS) associated with the preceding time steps $((\ )^n$ for $iter = 1$ and $(\ )^{(n+\frac{1}{2})}$ for

$iter = 2$) are dropped wherever possible.
DO $iter = 1, 2$

1. Gradients of the solution variables $\boldsymbol{\Phi} \in [\boldsymbol{u}, P, \boldsymbol{S}]$ are computed and piecewise monotonic linear reconstruction of $\boldsymbol{\Phi}$ is performed:

$$\boldsymbol{\Phi}(\boldsymbol{x}) = \boldsymbol{\Phi}_c + (\boldsymbol{x} - \boldsymbol{x}_c) \cdot (\psi_c \boldsymbol{\nabla \Phi}) \tag{202}$$

2. Utilizing reconstructed $\boldsymbol{\Phi}(\boldsymbol{x})$, the vertex velocity vector $\boldsymbol{u}^{p^{(n+\frac{iter-1}{2})}}$ is evaluated employing either an acoustic or Dukowicz Riemann solver:

$$\boldsymbol{u}^{p^{(n+\frac{iter-1}{2})}} = -\boldsymbol{\Theta}^{p^{-1}} \left\{ \sum_{\mathscr{C}(p) \ \forall \ p \ni \partial \varpi} \left(\widehat{\boldsymbol{f}_c^p}(\boldsymbol{x}^p) - \boldsymbol{\Theta}_c^p \boldsymbol{u}_c(\boldsymbol{x}^p) \right) \right\} \tag{203}$$

where

$$\widehat{\boldsymbol{f}_c^p}(\boldsymbol{x}^p) = \boldsymbol{a}_c^p \cdot \boldsymbol{\sigma}_c(\boldsymbol{x}^p) \tag{204}$$
$$\boldsymbol{\sigma}_c(\boldsymbol{x}^p) = \boldsymbol{S}_c(\boldsymbol{x}^p) - P(\boldsymbol{x}^p)\boldsymbol{I} \tag{205}$$

3. Once the nodal velocities are computed, the corner traction force vectors ($\widetilde{\boldsymbol{f}}_c^{p^{(n+\frac{iter-1}{2})}}$, $\boldsymbol{f}_c^{p^{(n+\frac{iter-1}{2})}}$) are evaluated as follows:

$$\widetilde{\boldsymbol{f}}_c^{(p,p^{\pm})^{(n+\frac{iter-1}{2})}} = \widehat{\widetilde{\boldsymbol{f}}}_c^{(p,p^{\pm})}(\boldsymbol{x}^p) + \frac{1}{R^{(p,p^{\pm})}} \boldsymbol{\Theta}_c^{(p,p^{\pm})} \left(\boldsymbol{u}^p - \boldsymbol{u}_c(\boldsymbol{x}^p) \right) \tag{206}$$

$$\boldsymbol{f}_c^{p^{(n+\frac{iter-1}{2})}} = R^{(p,p^-)} \widetilde{\boldsymbol{f}}_c^{(p,p^-)^{(n+\frac{iter-1}{2})}} + R^{(p,p^+)} \widetilde{\boldsymbol{f}}_c^{(p,p^+)^{(n+\frac{iter-1}{2})}} \tag{207}$$

4. The position vectors of the vertices and consequently the Lagrangian mesh are updated by solving the kinematic equation:

$$\boldsymbol{x}^{p^{(n+\frac{iter}{2})}} = c_1 \boldsymbol{x}^{p^n} + c_2 \boldsymbol{x}^{p^{(n+\frac{1}{2})}} + c_3 \frac{\Delta t}{2} \boldsymbol{u}^{p^{(n+\frac{iter-1}{2})}} \tag{208}$$

5. With vertex velocity and corner traction force vectors determined above, the governing equations are solved:

$$\boldsymbol{u}_c^{(n+\frac{iter}{2})} = c_1 \boldsymbol{u}_c^n + c_2 \boldsymbol{u}_c^{(n+\frac{1}{2})} + c_3 \frac{\Delta t}{2m_c} \sum_{\mathscr{P}(c)} \widetilde{\boldsymbol{f}}_c^{p^{(n+\frac{iter-1}{2})}} \tag{209}$$

$$E_c^{(n+\frac{iter}{2})} = c_1 E_c^n + c_2 E_c^{(n+\frac{1}{2})} + c_3 \frac{\Delta t}{2m_c} \sum_{\mathscr{P}(c)} \boldsymbol{f}_c^{p^{(n+\frac{iter-1}{2})}} \cdot \boldsymbol{u}^{p^{(n+\frac{iter-1}{2})}} \tag{210}$$

$$e_c^{(n+\frac{iter}{2})} = E_c^{(n+\frac{iter}{2})} - \frac{1}{2} \boldsymbol{u}_c^{(n+\frac{iter}{2})} \cdot \boldsymbol{u}_c^{(n+\frac{iter}{2})} \tag{211}$$

6. Volume and density of the cell are updated and subsequently pressure is computed using the equation of state:

$$V_c^{(n+\frac{iter}{2})} = V(\boldsymbol{x}^{p^{(n+\frac{iter}{2})}}) \tag{212}$$
$$\rho_c^{(n+\frac{iter}{2})} = \frac{m_c}{V_c^{(n+\frac{iter}{2})}} \tag{213}$$
$$P_c^{(n+\frac{iter}{2})} = EOS(\rho_c^{(n+\frac{iter}{2})}, e_c^{(n+\frac{iter}{2})}) \tag{214}$$

7. The stress evolution equation is solved to determine the predicted trial elastic state ($\boldsymbol{S}_{tr,c}^{(n+\frac{iter}{2})}$):

$$\boldsymbol{S}_{tr,c}^{(n+\frac{iter}{2})} = c_1\boldsymbol{S}_c^n + c_2\boldsymbol{S}_c^{(n+\frac{1}{2})} - c_3\frac{\Delta t}{2}\left(\boldsymbol{\Omega}_c^{(n+\frac{iter-1}{2})}\boldsymbol{S}_c^{(n+\frac{iter-1}{2})} - \boldsymbol{S}_c^{(n+\frac{iter-1}{2})}\boldsymbol{\Omega}_c^{(n+\frac{iter-1}{2})} - 2G\bar{\boldsymbol{D}}_c^{(n+\frac{iter-1}{2})}\right) \quad (215)$$

8. J_2 based von Mises plastic flow rule is enforced by radially re-scaling the stress components to lie on or within the yield surface:

END DO

7. Numerical Examples

In this section, a set of benchmark examples are presented to demonstrate the performance and effectiveness of the cell-centered scheme in accurately capturing the elasto-plastic response of solids. In essence, a hypo-elastic model with realistic plastic flow material models (Eqs (C.1),(C.2) & (C.3)) are used and an iterative radial return algorithm (section Appendix B) is performed to enforce the consistency condition. An equation of state of the Mie-Grüneisen form is used for computing the pressure [39, 38]. The gradients for stress tensor, velocity and pressure are estimated using the mimetic gradient evaluation technique presented in section 5.1.1. Unless otherwise stated, limiter schemes presented in section 5.1.2 in combination with Venkatakrishnan limiter function [57] is used. For the unlimited case, no limiter is applied for stress and velocity gradients. The results from current calculations are compared and contrasted with the solutions computed using the following hydrocodes:

- FLAG - a well established staggered grid Lagrangian finite volume hydrocode developed at the Los Alamos National Laboratory [11].
- ELAFINT-3D - a robust level set based multi-material Eulerian hydrocode developed in [39, 38]
- PAGOSA - an Eulerian hydrocode (Lagrangian hydrocode in combination with a remap to the initial grid) developed at the Los Alamos National Laboratory [60].

Unless stated otherwise, similar mesh resolutions are employed for both the current and reference calculations.

7.1. Elastic Vibration of Beryllium Plate

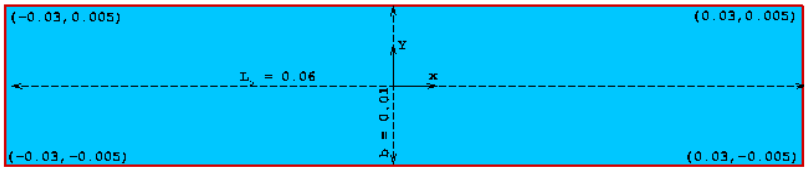


Figure 5: Initial configuration of the two-dimensional beryllium plate.

The problem has been analyzed in [61]. This is a two dimensional problem comprising of a beryllium rectangular plate of infinite extent, with no supports or constraints. The plate extends from -3.0 cm to +3.0 cm along the X-axis and from -0.5 cm to +0.5 cm along the Y-axis (Figure 5). As shown in Figure 5, the centerline of the plate coincides with the x-axis. The plate is prescribed with an initial velocity distribution as given below:

$$v(x,t) = A\omega\left\{c_1\left[\sinh\Omega\left(x+3\right)+\sin\Omega\left(x+3\right)\right]+c_2\left[\cosh\Omega\left(x+3\right)+\cos\Omega\left(x+3\right)\right]\right\}\sin(\omega t) \quad (216)$$

with initial velocity computed at t = 0. The plate, when subject to this initial velocity distribution, begins to oscillate elastically. Following the analysis in [61], Ω, the first non-zero root of the frequency equation is determined as:

$$\cosh 6\Omega\cos 6\Omega = 1 \quad (217)$$

Solving for the above equation, the value for Ω is determined as:

$$\Omega = 0.7883401241 \text{ cm}^{-1} \tag{218}$$

Similarly, for the given dimensions and material parameters

$$\rho = 1845 \text{ kgm}^{-3}, Y_o = 1 \text{ Mbar}, k = 1.18896 \text{ Mbar}, E = 3.182656 \text{ Mbar}, \nu = 0.053896 \tag{219}$$

of the bar, the constants c_1, c_2 and the angular frequency ω are determined as:

$$c_1 = 56.63685154, c_2 = 57.64552048, \omega = 0.2359739922 \ \mu\text{s}^{-1}, A = 0.004336850425 \text{ cm} \tag{220}$$

where $k, E \ \& \ \nu$ are the bulk modulus, elastic modulus and Poisson's ratio respectively and the parameter A is related to the amplitude of the oscillation as:

$$A = \frac{1}{2}\left(\frac{Y_{max}}{c_2}\right) \tag{221}$$

with Y_{max} denoting the maximum displacement in the vertical direction. The angular frequency is related to the material parameters and Ω through the following equation:

$$\omega = \Omega^2 \sqrt{\frac{E(\Delta y)^2}{12\rho(1-\nu^2)}} \tag{222}$$

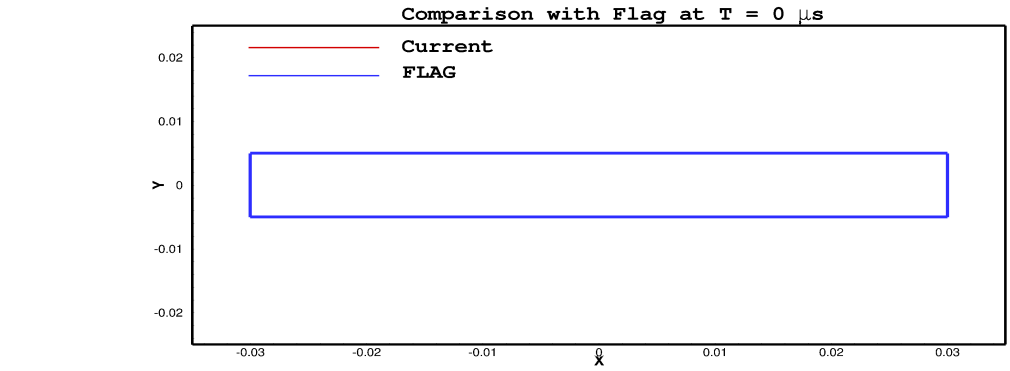
where $\Delta y = 1$ cm is the thickness of the plate. The yield stress for the material is set at a very high value so that the plate oscillates elastically. The choice of yield model and the radial return algorithm for this problem is inconsequential as the yield stress conditions are never reached and hence no plastic strain is incurred.

7.1.1. Comparison with FLAG - Structured Mesh Calculation

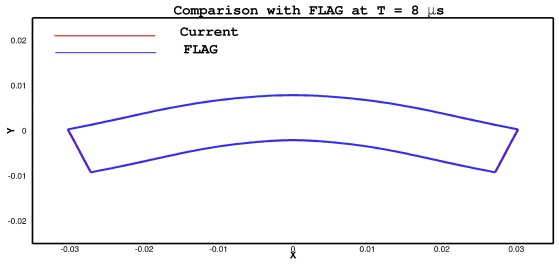
In Figure 6, results from the current calculations are displayed. The calculations presented in the figure are obtained by computing on a Cartesian mesh with 100×25 cells along the length and width of the plate. The shape of the plate at different instants in time are compared with FLAG computations. Excellent agreement is noted between the two calculations. Both FLAG and current calculations were performed on a structured Cartesian mesh with same mesh resolution. The figure indicates that the plate oscillates elastically with no plastic deformation. The snapshots shown in the figure correspond to one complete cycle of oscillation of the plate. The plate continues to oscillate with insignificant amount of dissipation of energy. Mesh convergence study for this problem is presented in Figure 7. The figure indicates that the solution is independent of the mesh resolution. The maximum and minimum amplitude of oscillation is preserved throughout the computation. The plot for vertical velocity magnitude shown in Figure 7(b) shows that the over-/under-shoots present in FLAG calculations are not present in the current calculations.

7.1.2. Comparison with FLAG and PAGOSA - Unstructured Mesh Calculations

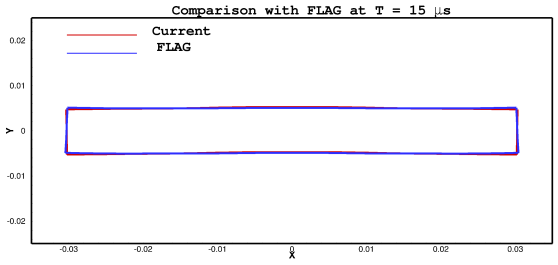
In Figure 8, the computations are repeated on an unstructured, initially Voronoi mesh. In Figure 8(a), the mesh at0 μs is shown. The configuration of the bar and the mesh along with the contours of velocity magnitude are shown at different instants in time for one complete cycle. The figure also shows the configuration of the bar corresponding to FLAG calculations computed on a structured Cartesian mesh. Except for minor discrepancy at 23 μs no visible difference can be seen. The figure indicates that the effect of the underlying mesh is not negligible.In Figure 9, the locus of the center of the bar and the history of vertical velocity are compared with FLAG and PAGOSA calculations. The plots indicate that the unstructured mesh calculation is more dissipative than the corresponding structured mesh computations. Moreover the trend in this case follows closely PAGOSA's results. Although the extremum for the velocity magnitude is well preserved (Figure 9(b)), the maximum and minimum amplitudes for vertical displacement (Figure 9(a)) are smaller than FLAG's predictions. History of energy conversion plotted in Figure 9(c) shows that the total energy is well preserved as the kinetic energy is converted to internal energy back and forth as the plate oscillates through its cycle.



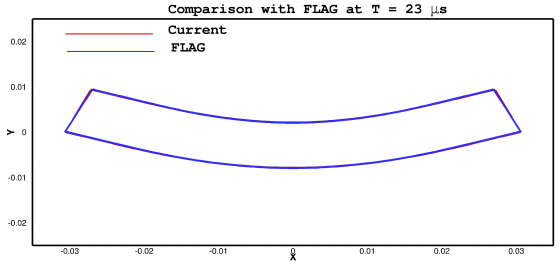
(a) 0 μs



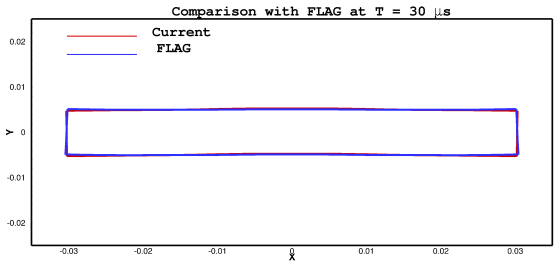
(b) 8 μs



(c) 15 μs

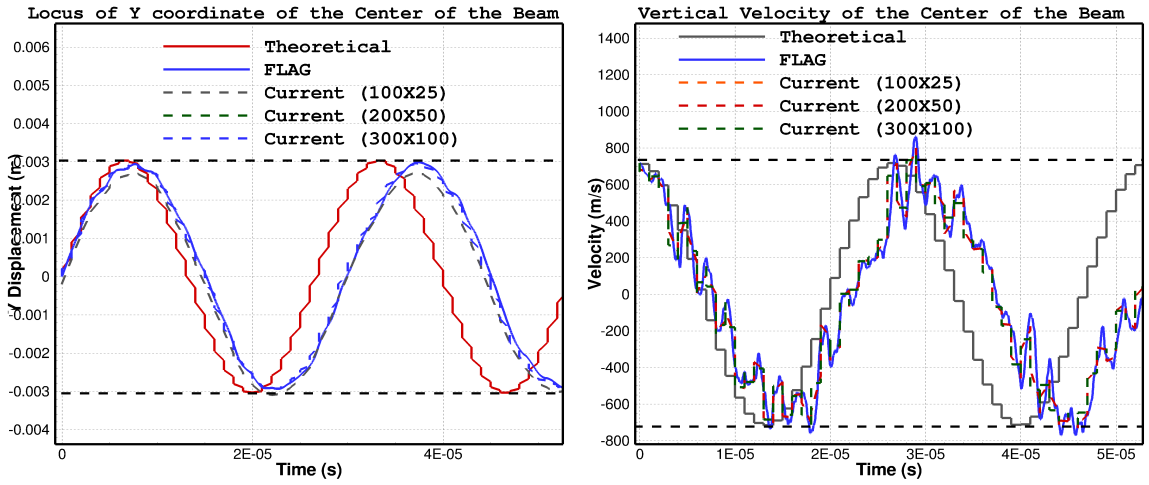


(d) 23 μs



(e) 30 μs

Figure 6: Shape of the plate and contours of velocity magnitude at different instants in time for the elastic vibration of beryllium plate. Also shown in the Figures is the shape of the bar from FLAG computations. Both current and FLAG computations were computed on a Cartesian structured grid with same mesh resolution.



(a) Vertical Displacement

(b) Vertical Velocity Magnitude

Figure 7: Mesh convergence study and comparison with FLAG for the elastic vibration of beryllium plate - (a) Locus of vertical displacement of the origin (center of the plate) (b) Magnitude of vertical velocity of the center of the plate for three different mesh resolutions

7.2. Incompressible Plastic Collapse of Spherical Beryllium Shell

Cylindrical and spherical collapse of Beryllium shell is discussed in this section. The problem has been discussed at length in [62]. The initial configuration of the shell is displayed in Figure 10. The problem consists of a 2 cm thick beryllium shell undergoing an incompressible plastic collapse. The inner and outer radius of the shell are set at 8 cm and 10 cm respectively. For the cylindrical shell, a plane strain assumption is made. The initial velocity profile that drives the shell is given below:

$$U(r) = -U_0(r/R_i)^{\alpha} \tag{223}$$

where $\alpha = 1.0$ for plane geometry (cylindrical case) and $\alpha = 2.0$ for RZ geometry (spherical shell). The final stopping radius and time are determined by the initial kinetic energy of the shell. These are summarized in Table 3 for both the cylindrical and spherical shell collapse. The computations for both cylindrical and spherical shell are

configuration	U_0	Final Stopping Radius (cm)	Final Stopping Time (μ s)
Cylindrical	490.2	4	137
Spherical	673.4	3	100

Table 3: Analytical stopping radius and time for spherical and cylindrical collapse of beryllium shell

performed with the following material parameters:

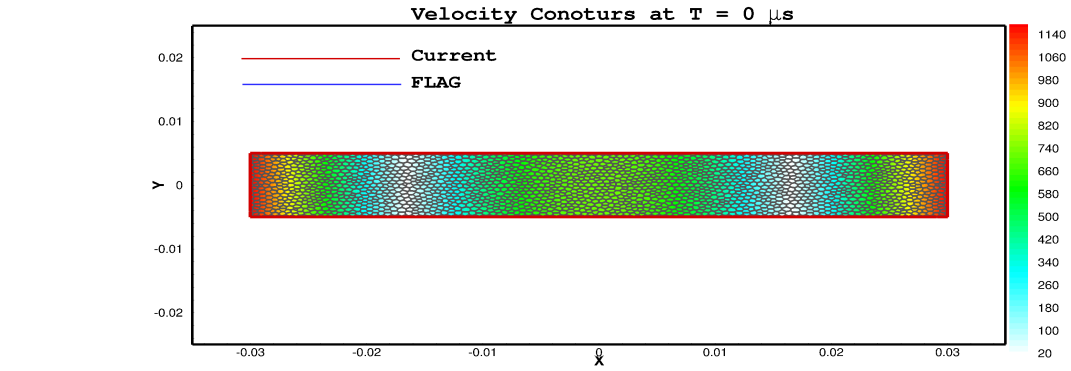
$$\rho_o = 1845.0 \text{ kgm}^{-3}, Y_o = 0.33Mbar, G_o = 151Mbar$$

$$C_o = 7998.0 \text{ ms}^{-1}, S = 1.124, \gamma_o = 1.16$$

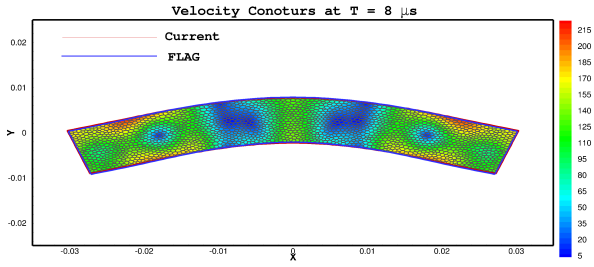
The simulations are run upto a final time T = $200\mu s$ and the results from these computations are reported below.

7.2.1. Collapse of Cylindrical Beryllium Shell in an Arbitrary Polygonal Mesh

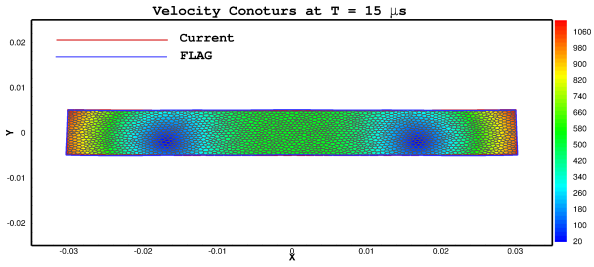
For the cylindrical collapse, $U_0 = 490.2$ ms^{-1} and the calculations are performed with plane strain assumption. Furthermore, the computations are performed a polygonal grid constructed from the dual of a grid with triangular elements (Figure 11). The total number of polygonal elements are 10500 and these elements are randomly distributed in the computational domain. The results from the current calculations are displayed in Figure 12. In Figure 12(a),



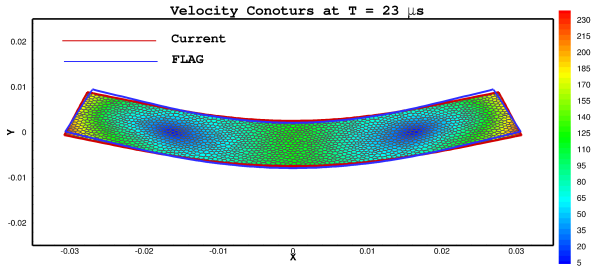
(a) 0 μs



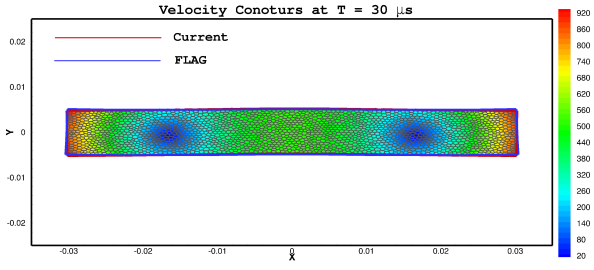
(b) 8 μs



(c) 15 μs

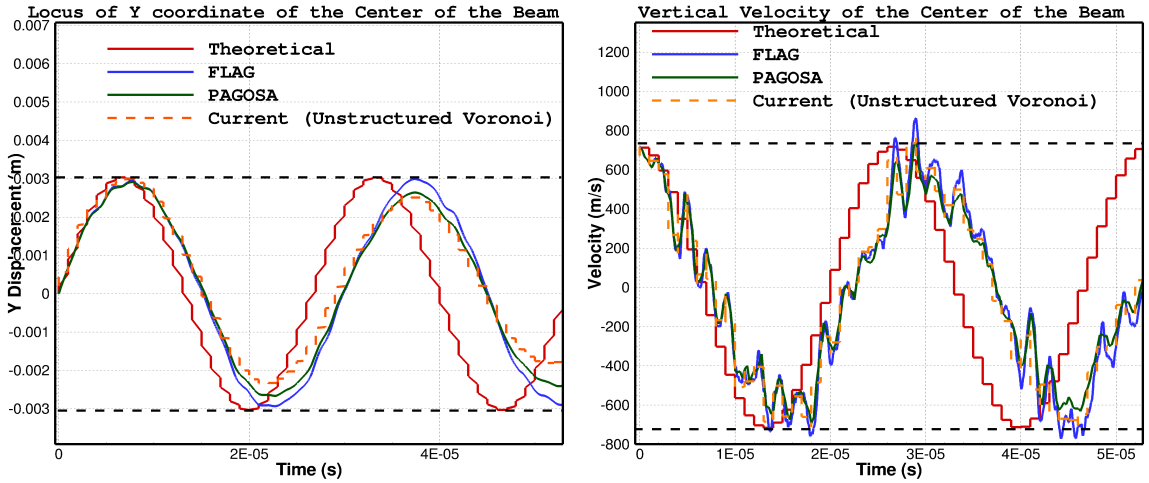


(d) 23 μs



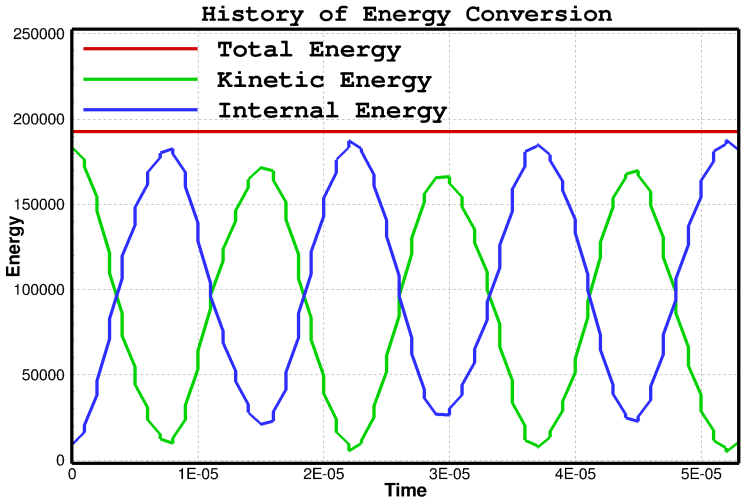
(e) 30 μs

Figure 8: Shape of the plate and contours of velocity magnitude at different instants in time for the elastic vibration of beryllium plate computed using an initially Voronoi mesh. Also shown in the Figures is the shape of the bar from FLAG computations. FLAG computations for this case were computed on a Cartesian structured grid with same mesh resolution as the unstructured case.



(a) Vertical Displacement

(b) Vertical Velocity Magnitude



(c) Energy Conversion History

Figure 9: Comparison with FLAG and PAGOSA for the elastic vibration of beryllium plate - (a) Locus of vertical displacement of the origin (center of the plate) (b) Magnitude of vertical velocity of the center of the plate (c) History of energy conversion

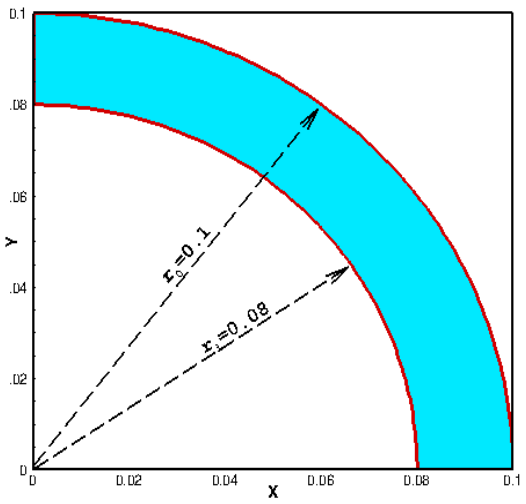


Figure 10: Initial configuration of the cylindrical beryllium shell. Dimensions shown in the figure are in meters.

contours of velocity magnitude is plotted at different instants in time. The figure indicates that the symmetry of the solution is well preserved. Furthermore, the velocity magnitude in the shell becomes negligible at about 140 μs, consistent with the analytical results. Figure 12(b), shows comparison for the effective plastic strain contours with FLAG calculations at 140 μs. The plots indicate excellent agreement between the two calculations. The results from FLAG calculations were computed on a structured annular mesh with quadrilateral elements (Figure 12(b)). The enlarged view of the mesh shown in the figure confirms that the current solution is independent of the mesh topology. The locus of inner radius (Figure 12(c)) and the history of velocity magnitude (Figure 12(d) plotted at different positions along the inner radius indicate remarkable agreement between the two calculations. Furthermore, the inherent symmetry in the solution is also well preserved.

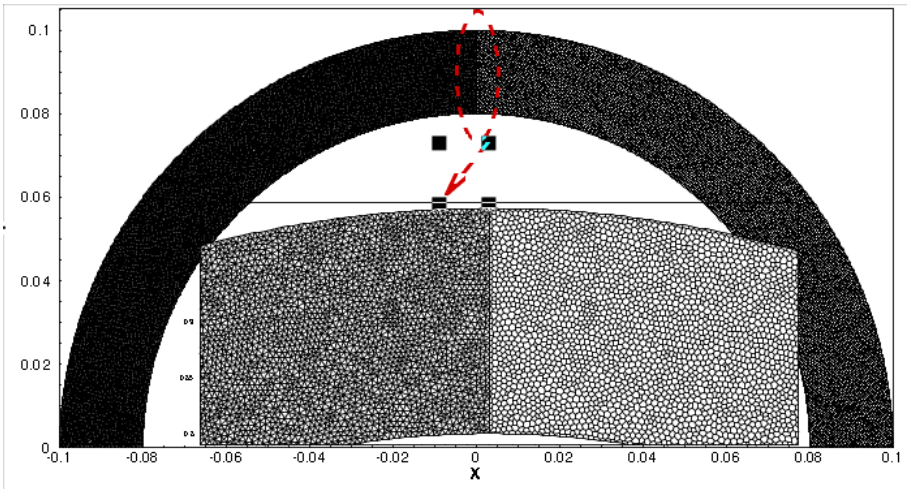
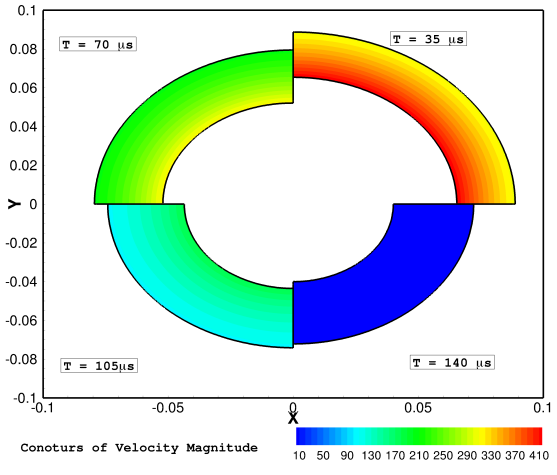


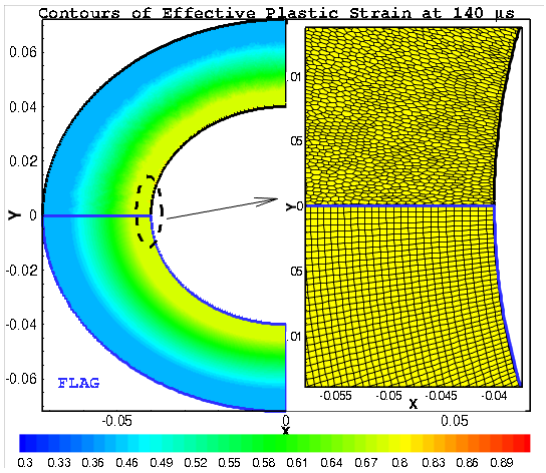
Figure 11: Topology of the polygonal mesh used for the collapse of cylindrical beryllium shell calculations. The polygonal mesh (right hand side) is constructed from the dual of the triangular mesh displayed on the left hand side.

7.2.2. Collapse of Spherical Beryllium Shell in a Structured Annular Mesh

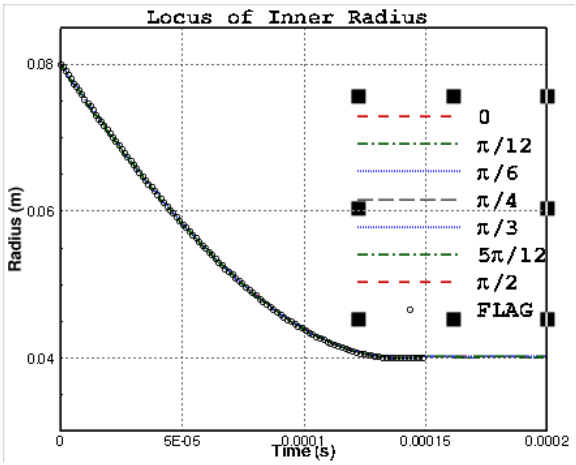
For the spherical collapse, $U_0 = 673.4$ ms^{-1} with $\alpha = 2.0$. The spherical shell coasts until the inner radius reaches 3 cm at 100 μs. At this instant in time, all the kinetic energy is converted to internal energy bringing the shell



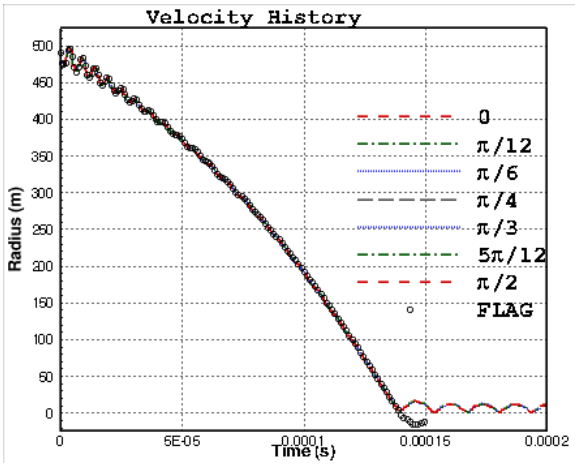
(a) Contours of Magnitude of Velocity



(b) Contours of Effective Plastic Strain and Mesh at 140 μs



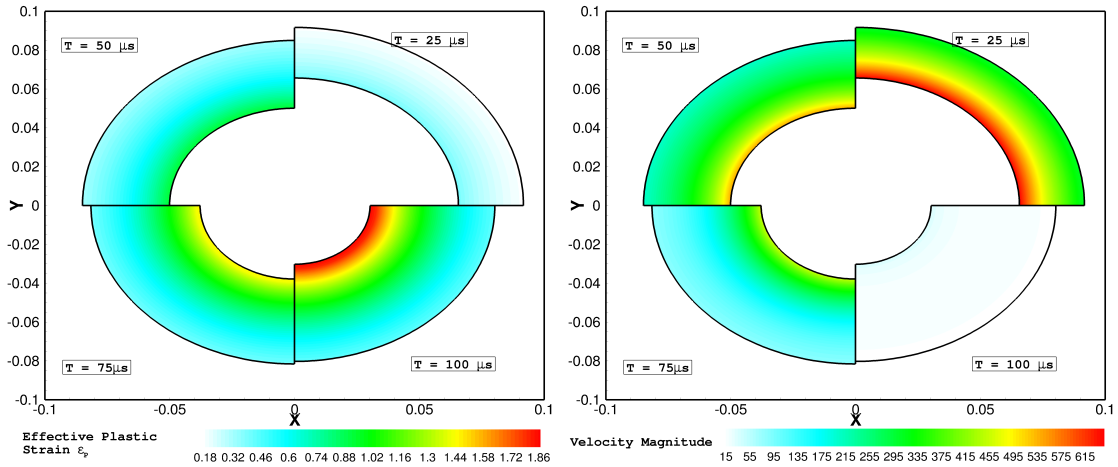
(c) Locus of Inner Radius



(d) Velocity History

Figure 12: Plots for the collapse of cylindrical beryllium shell (a) Contours of magnitude of velocity at different instants in time (b) Contours of effective plastic strain ϵ_p at 140 μs (c) Locus of inner radius and (d) Time history of velocity magnitude at different positions along the inner radius.

to a complete rest. The shell will continue to collapse if it is treated purely hydrodynamic, with no strength model. A detailed analysis for the analytical solution can be found in [63].



(a) Contours of Effective Plastic Strain

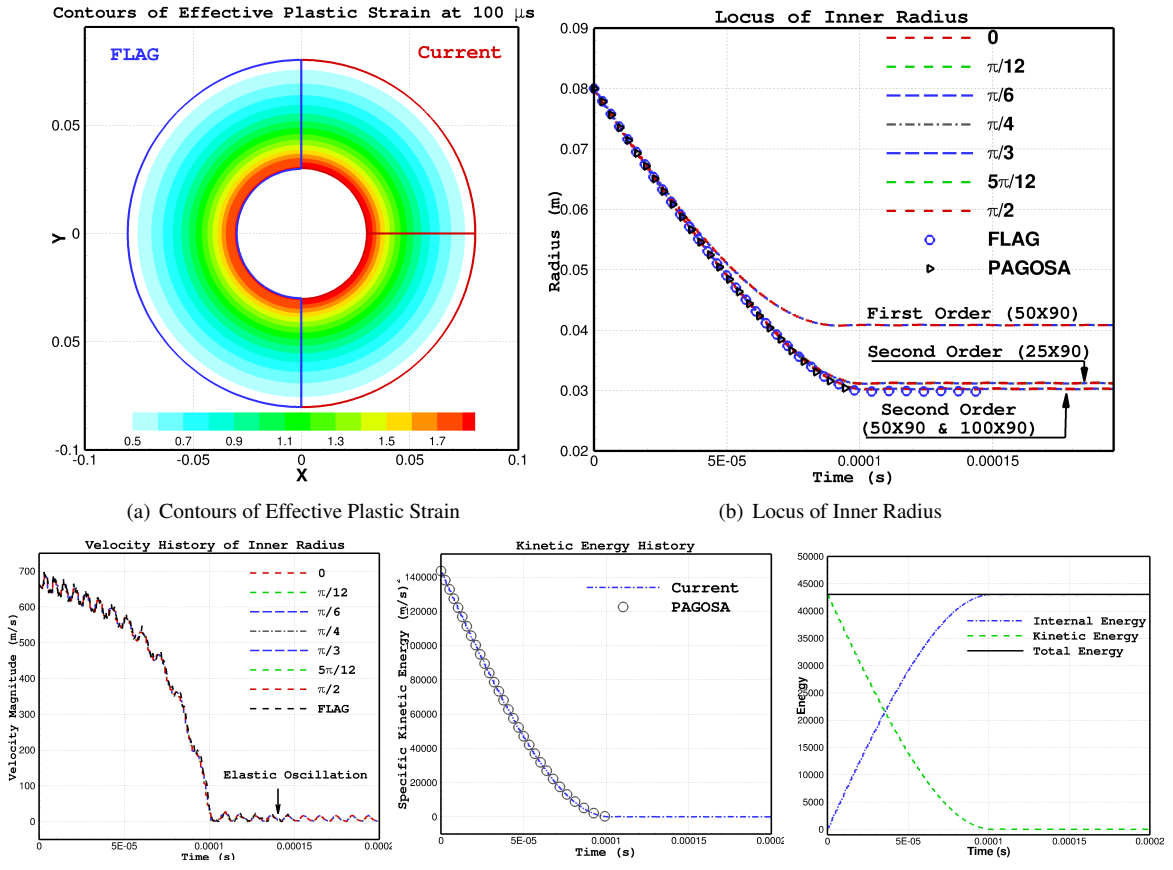
(b) Contours of Velocity Magnitude

Figure 13: Contours of effective plastic strain ϵ_p and velocity magnitude at different instants in time for the incompressible plastic collapse of spherical shell.

The calculations presented in this section are performed in RZ geometry. An annular grid with 50×90 cells in r, θ directions is used. In Figure 13, contours of effective plastic strain and velocity magnitude are plotted. The contour plots indicate that the inherent symmetry in the problem is well preserved. Figure 13(b) shows that the magnitude of the velocity of the shell is almost zero at 100 μs. The predicted final inner radius of the bar at this instant is 3 cm.

Comparison with FLAG and PAGOSA: Comparison between FLAG and current calculations for the contours of effective plastic strain is shown in Figure 14(a). No discernible difference can be seen between the two plots. The locus of inner radius plotted in Figure 14(b) confirms that the shell comes to rest at 100 μs with inner radius measuring 3 cm. Different points along the inner radius of the shell follow exactly the same trend ensuring perfect symmetry of the shell. The figure also shows that convergence of the solution is reached for the chosen mesh resolution. PAGOSA, FLAG and current predictions lie on top of each other. As expected the second-order solution is much superior to the first order case. Neither the order of accuracy of the scheme nor the mesh resolution have any effect on the symmetry of the resolution. This is to say that symmetry preservation is not a consequence of any of these factors but an intrinsic feature of the scheme. The history of velocity magnitude plotted at different points along the inner radius of the shell compares well with FLAG calculations 14(c). The shell oscillates elastically beyond 100 μs with no dissipation of energy and this behavior can be clearly identified in the figure. Conversion of kinetic to internal energy of the shell matches well with PAGOSA's prediction (Figure 14(d)). No significant dissipation of energy is noted beyond 100 μs and the same is reflected in the energy conversion history plotted in Figure 14(e). In addition, the figure also shows that the total energy of the shell is preserved throughout the computation.

Comparison between Least Squares and Mimetic Gradient Estimate Techniques : Comparisons between least squares and mimetic gradient evaluation technique are made for the spherical shell collapse. In this case, the calculations are carried on an extremely coarse mesh with 10×90 cells. The plots for these calculations are displayed in Figures 15(a) & 15(b). The plots indicate that the least squares approach fails to preserve symmetry particularly during the elastic oscillation period. It is during this period, the cells along the inner surface of the shell are stretched and compressed alternately. The zoomed view of the mesh gives an indication of the aspect ratio of the cells encountered near the inner surface of the shell. The least squares approach appears to be sensitive to the aspect ratio of the cells whereas the mimetic gradient evaluation technique appears to be robust and independent of the quality of the mesh. The symmetry of the solution is well preserved with the mimetic formulation.



(a) Contours of Effective Plastic Strain

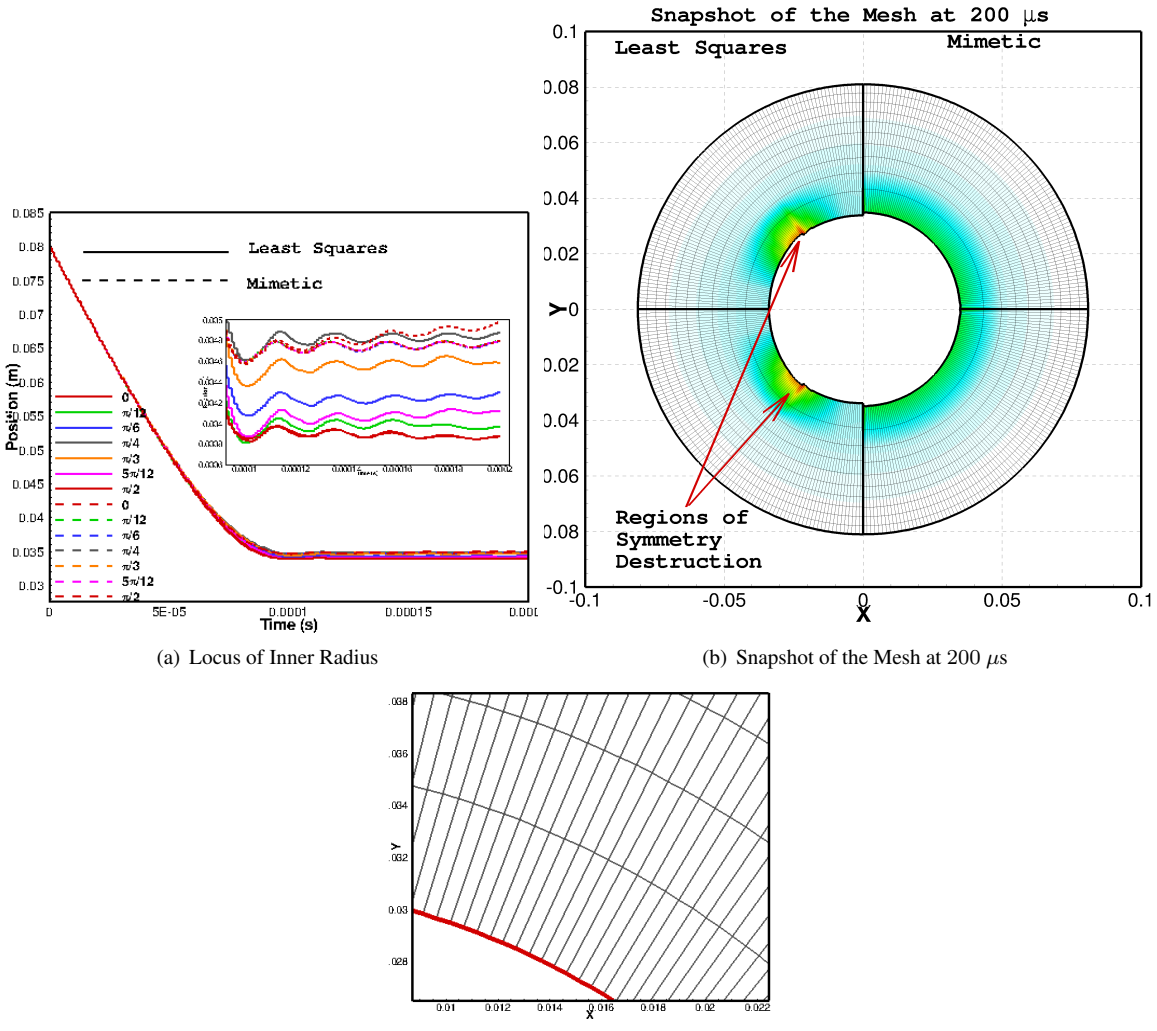
(b) Locus of Inner Radius

(c) History of Velocity Magnitude

(d) History of Kinetic Energy

(e) History of Energy Conversion

Figure 14: Comparison with FLAG and PAGOSA for the incompressible plastic collapse of spherical shell - (a) Contours of effective plastic strain at 100 μs (b) Locus of inner radius for three different mesh resolution (c) History of magnitude of velocity for points along the inner radius (d) History of Kinetic energy of the shell (e) Conversion of energy from kinetic to internal energy



(a) Locus of Inner Radius

(b) Snapshot of the Mesh at 200 μs

(c) Enlarged View of the Mesh at 200 μs

Figure 15: Comparison between least squares and mimetic gradient estimate technique for the incompressible plastic collapse of spherical shell on mesh with 10×90 cells in r, θ directions respectively - (a) Locus of inner radius for three different mesh resolution (b) Snapshot of the Mesh at 200 μs (c) Enlarged view of the mesh near the inner surface of the shell at 200 μs

7.3. Axisymmetric Shock Tube Problem in Aluminum

In this example, a slightly different version of the two-dimensional shock tube problem presented in [64] is considered. Material parameters corresponding to aluminum are given below:

$$\rho = 2710 \text{ kgm}^{-3}, G = 0.286 \text{ GPa}, Y_0 = 0.0026 \text{ GPa}, C_o = 5330 \text{ ms}^{-1}, s = 1.34, \Gamma = 2.0$$

The initial conditions for the shocked and un-shocked regions are summarized in the Table 4. Shocked conditions are

Initial Conditions	Un-shocked Conditions	Shocked Conditions:
ρ	2710.0 kg m^{-3}	2764.2 kg m^{-3}
P	0	1.60 GPa
$\boldsymbol{u}$	0	0
σ_{xx}	0	-1.60 GPa
σ_{yy}	0	-1.60 GPa
σ_{xy}	0	0

Table 4: Initial conditions for the shock tube problem in aluminum - shocked conditions are enforced when $||\boldsymbol{x}_c - \boldsymbol{x}_o|| < 1.5$ m with $\boldsymbol{x}_o = \vec{0}$

enforced when $||\boldsymbol{x}_c - \boldsymbol{x}_o|| < 1.5$ m with $\boldsymbol{x}_o = \vec{0}$.

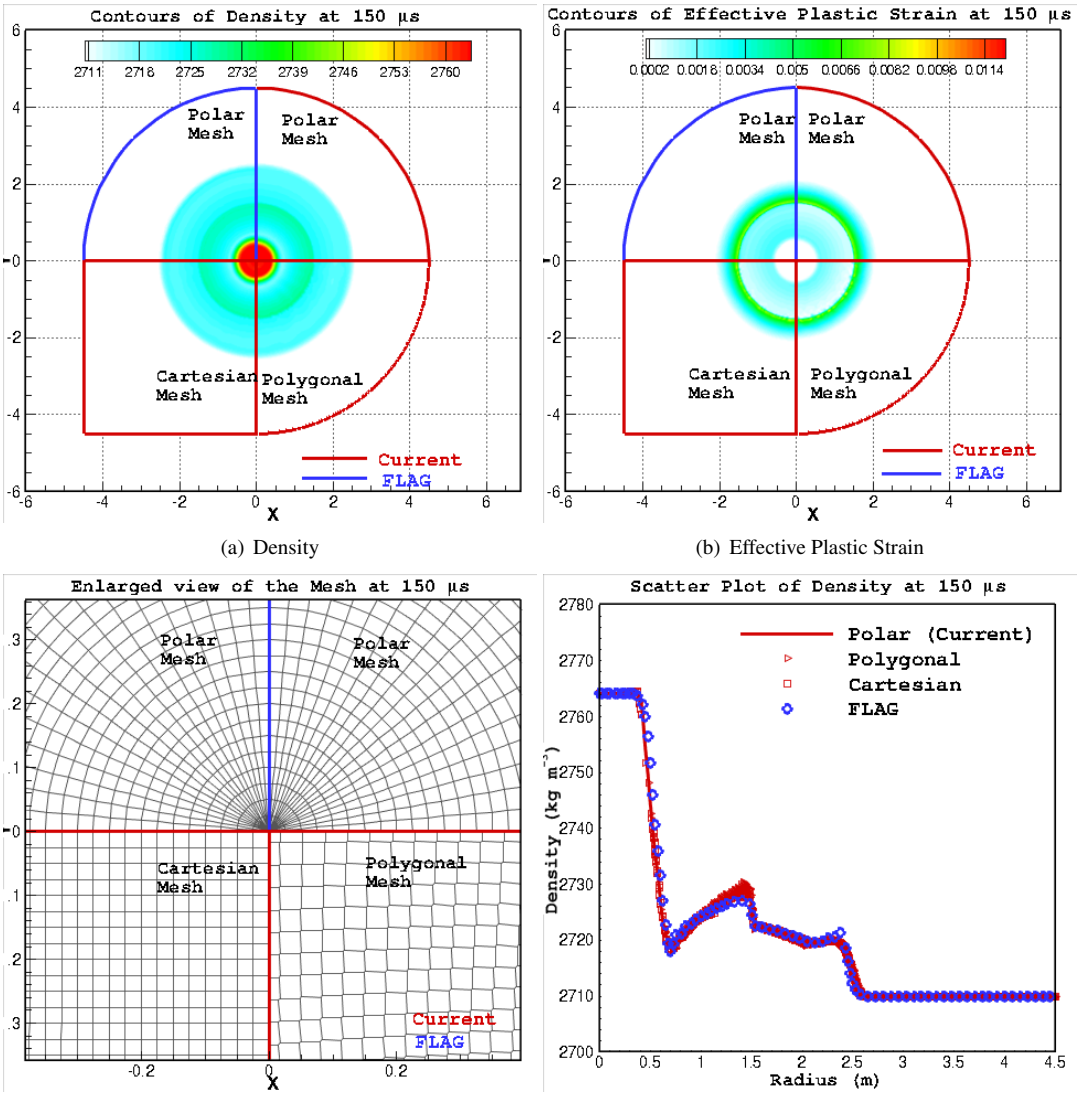
The purpose of this test case is to isolate the effects of pressure from material strength (shear stress) and benchmark the results in comparison with FLAG calculations. In this case, the calculations are performed on three-different mesh typologies:

- a polar mesh with 180×18 cells in r, θ directions in a computational domain of dimensions $4.5m \times 90^o$,
- a Cartesian mesh with 180×180 cells in a computational domain of dimensions $4.5m \times 4.5m$,
- a polygonal (Voronoi) mesh constructed from a dual of a triangular mesh with 16641 cells in a computational domain of dimensions $4.5m \times 90^o$

The simulations are run in cylindrical axisymmetric coordinates upto a final time of 150 μs. The results from the calculations are compared contrasted against the solution obtained from FLAG calculations. The FLAG calculations were carried out on a structured polar mesh with 180×18 in a computational domain of dimensions $4.5m \times 90^o$. Results from the current and FLAG calculations are displayed in Figures 16. The contour plots displayed in Figures 16(a) & 16(b) show excellent agreement between the two calculations in predicting the location of the discontinuities. Symmetry is well preserved in both calculations. The enlarged view of the mesh is shown in Figure 16(c). The scatter plot displayed in Figure 16(d) indicates that the solutions are independent of mesh topology. The inherent symmetry of the problem is well preserved in addition to being in good agreement with the reference FLAG calculations. Furthermore, the location of shock and contact discontinuity are in good agreement with FLAG predictions.

7.4. Axisymmetric Taylor Bar Experiment: Impact of a Copper Rod Over a Rigid Substrate at 227 *ms*$^{-1}$

Taylor bar test [65] on a copper rod is considered. The problem set up has been discussed in [38, 39]. The Taylor bar impact test is a standard test problem to verify and validate numerical and experimental observations. In the axisymmetric setting, a cylindrical rod made of copper with an initial radius of 3.2 mm and a length of 32.4 mm impacts a rigid flat substrate at 227 m/s (Figure 17). The rod has an initial density of 8930 kgm^{-3}, Young's modulus $E = 117$ GPa, Poisson's ratio $\nu = 0.35$, and yield stress $\sigma_Y = 400$ MPa. The material is assumed to harden linearly with a plastic modulus of 100 MPa. The calculations are carried up to a time of $80\mu s$ at which point nearly all the initial kinetic energy has been dissipated as plastic work. The results from the current calculations are compared with Eulerian [38, 39] and Lagrangian (FLAG) calculations.



(a) Density

(b) Effective Plastic Strain

(c) Enlarged View of the Mesh

(d) Scatter Plot

Figure 16: Plots for the axisymmetric shock tube problem in aluminum at 150 μs - (a) Contours of density (b) Contours of effective plastic strain (c) Enlarged view of the mesh (d) Scatter plot of density.

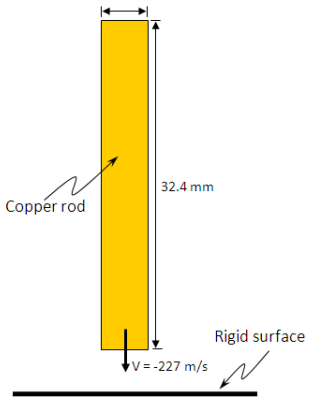


Figure 17: Initial configuration for the axisymmetric Taylor test on a Copper rod.

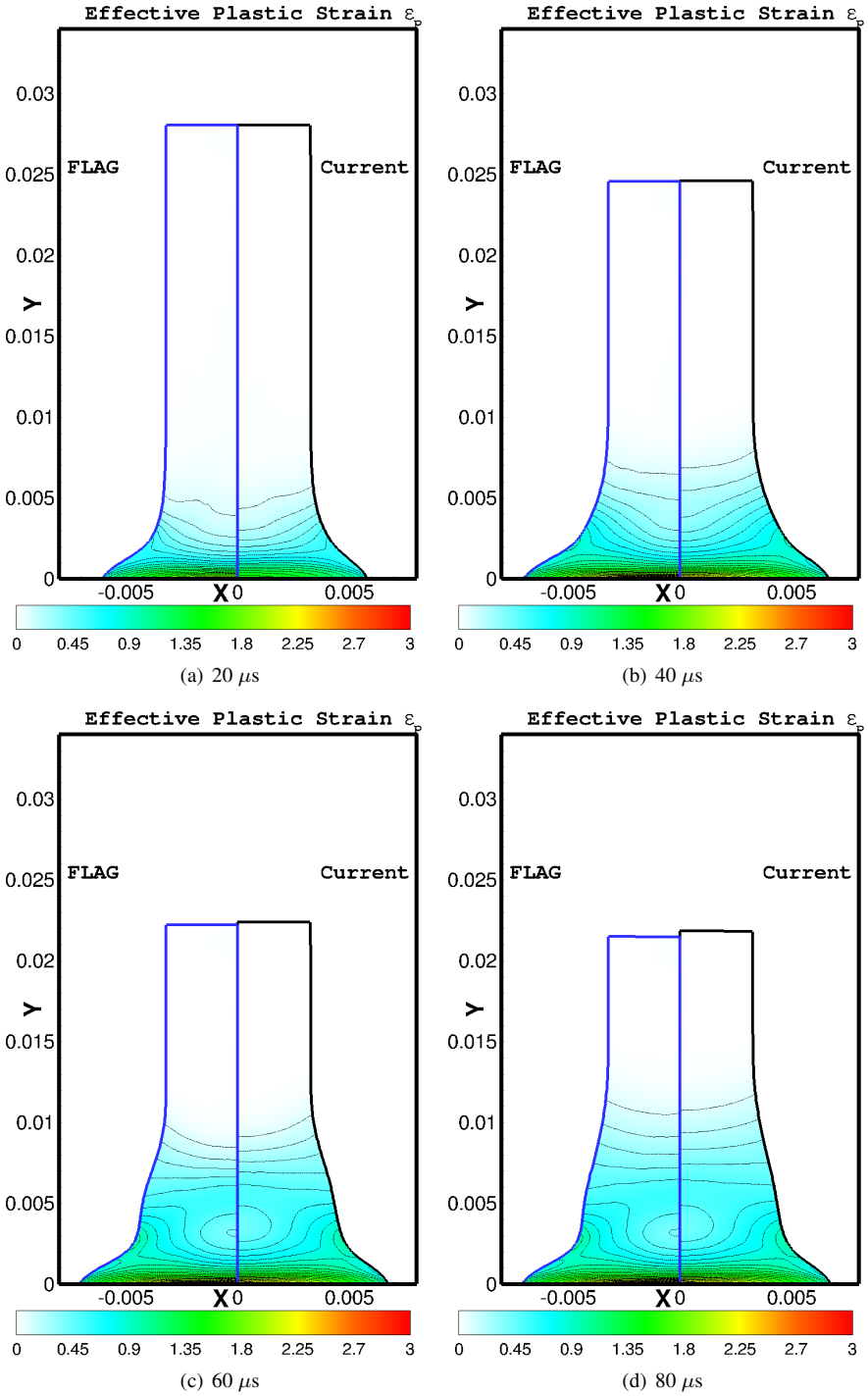
7.5. Structured Mesh Computations for the Taylor impact test case at 227 ms^{-1}: Comparison with Lagrangian Hydrocode for Two Different Gradient Evaluation Techniques

In Figures 18 & 19 comparison between FLAG predictions and current calculations computed using least squares (Figure 18) and mimetic (Figure 19) gradient evaluation techniques are presented. During the early stages of the evolution of the rod, the solution corresponding to least squares gradient estimate technique matches well with FLAG calculations. However, Figure 18(b) indicates that least squares technique based solution begins to deviate from the reference FLAG calculations. These deviations are clearly discernible in Figure 18(c) & 18(d). These deviations are also apparent in the plots of position and velocity histories displayed in Figures 20(a) & 20(b). Further evidence for this trend can be seen in the final configuration of the bar displayed in Figure 21. These figures indicate that the least squares based solution continuously under-predicts the dimensions and the velocity magnitude of the bar. The final configuration of the bar predicted by least squares implementation also deviates distinctly from FLAG calculations. The reason for this departure from the reference solution can be attributed to the large aspect ratio cells that begins to develop as the bar begins to stretch and deform. In such situations, the least squares gradient estimate approach fails to provide satisfactory results.

On the contrary, with reference to Figure 19, the contour plots corresponding to mimetic gradient estimate technique shows excellent agreement with FLAG calculations. The locus plots for the position and velocity magnitude of the head and toe of the rod displayed in Figures 20(c) & 20(d) follow closely the FLAG calculations trend. Furthermore, the snapshot shown at 80 μs (Figure 21(b)) is a conclusive evidence of the superiority of the mimetic gradient estimate technique.

7.6. Unstructured Polygonal Mesh Computations for the Taylor impact test case at 227 ms^{-1}: Comparison with Eulerian Hydrocode

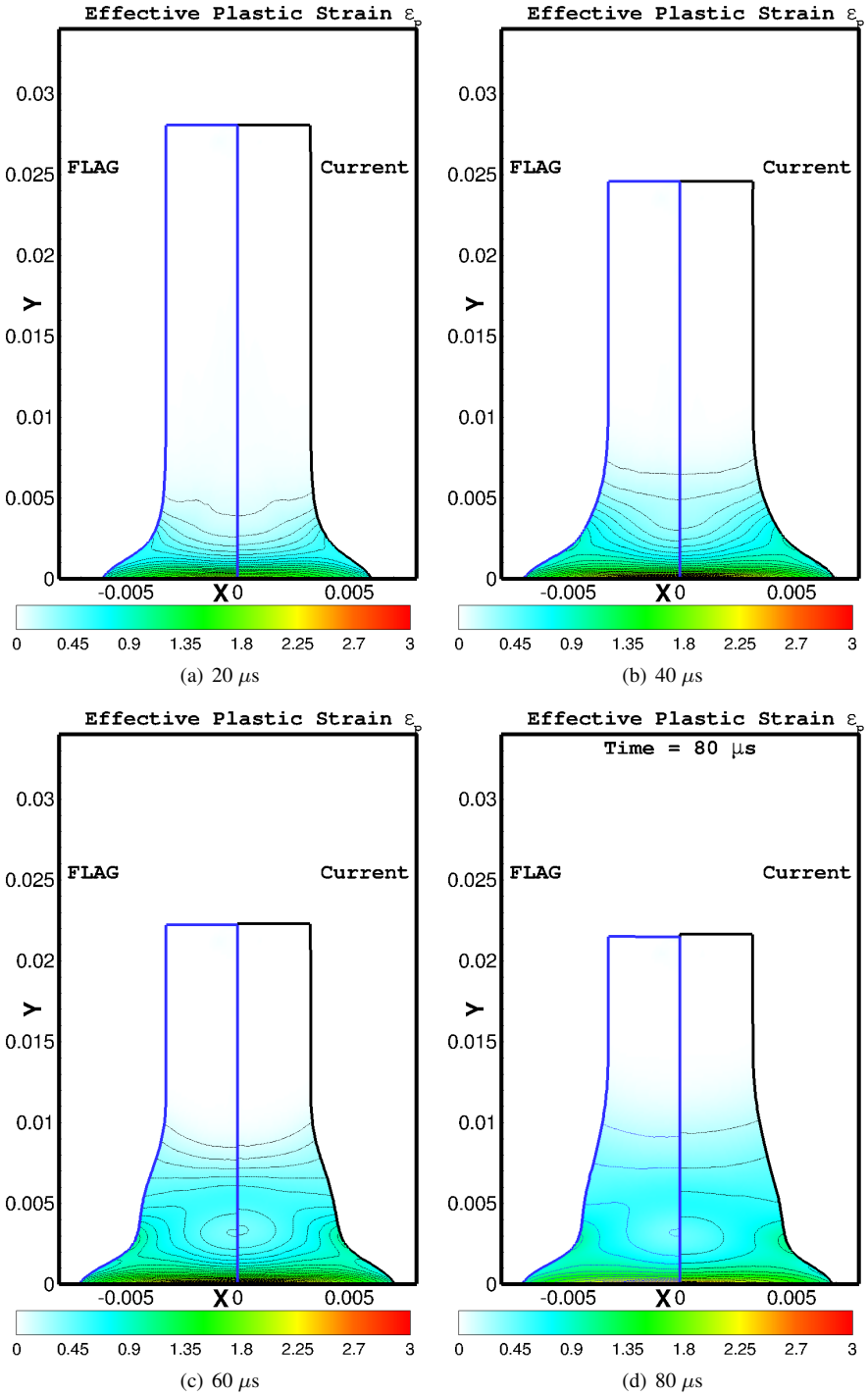
In this section, computations on mesh with arbitrary polygonal elements are reported. The initial topology of the mesh is shown in Figure 22(a) with the enlarged view of the elements comprising the mesh shown in Figure 22(b). The results from the polygonal mesh computations are compared against Eulerian calculations reported in [38]. The Eulerian calculations were performed on a mesh with 14901 cells whereas the current Lagrangian calculations were computed on a mesh with 14651 randomly distributed elements. The comparisons are displayed in Figure 23. Figures 23(a) through 23(c) show excellent agreement between the two with slight discrepancy in the final length of the bar at 75 μs as indicated in Figure 23(d). Contours effective plastic strain and the enlarged view of the polygonal mesh for the current Lagrangian computations are displayed in Figure 24. Despite of the disparities in the mesh topology, the two fundamentally different approaches for the same problem yield exactly the same outcome. The length of the bar, radius of the foot and the contours of effective plastic strain at different instants in time demonstrate striking similarities between the two calculations. The locus of head and tail of the bar plotted in Figure 25(a) and the final configuration of the bar displayed in Figure 25(b) at 75 μs further substantiates this claim. A summary of the



(a) 20 μs (b) 40 μs

(c) 60 μs (d) 80 μs

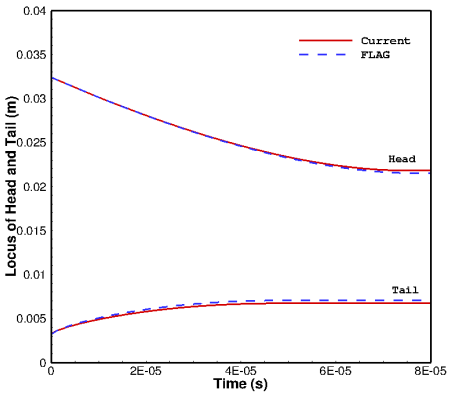
Figure 18: Structured mesh computations for the Taylor impact test case at 227 ms^{-1} compared with FLAG calculations for least squares gradient estimation technique - Contours of effective plastic strain ϵ_p at different instants in time



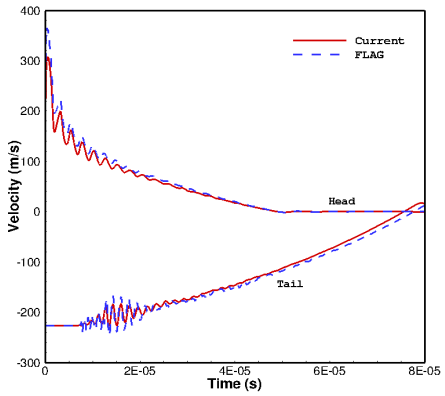
(a) 20 μs (b) 40 μs

(c) 60 μs (d) 80 μs

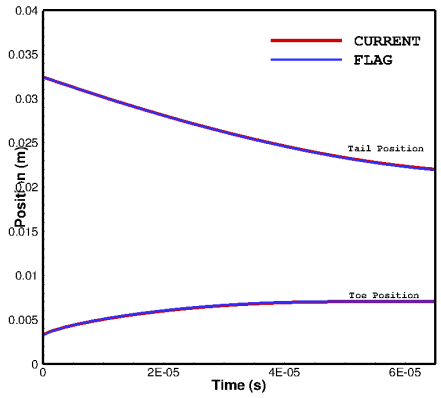
Figure 19: Structured mesh computations for the Taylor impact test case at 227 ms^{-1} compared with FLAG calculations for mimetic gradient estimation technique - Contours of effective plastic strain ϵ_p at different instants in time



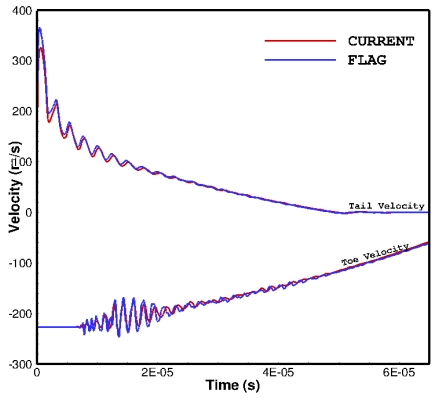
(a) Position History for Least Squares Implementation



(b) Velocity History for Least Squares Implementation

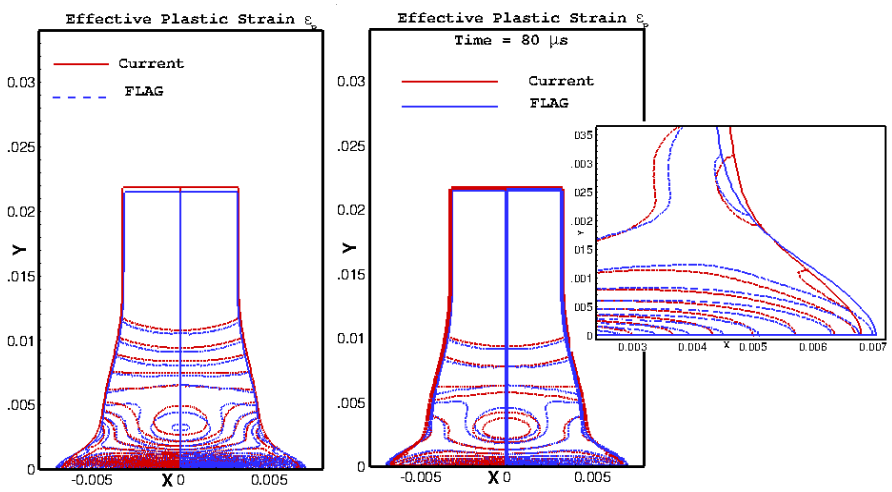


(c) Position History for Mimetic Implementation



(d) Velocity History for Mimetic Implementation

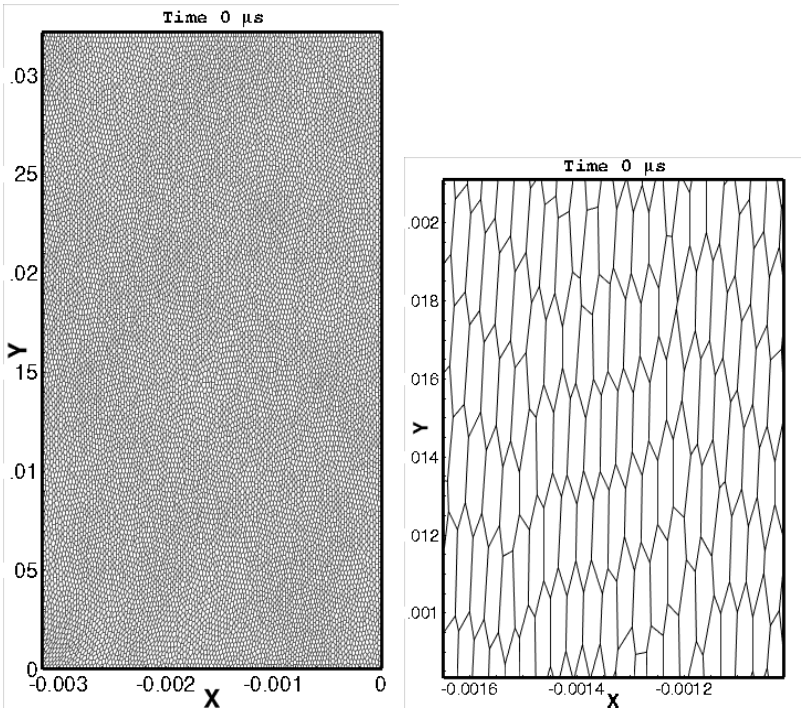
Figure 20: Structured mesh computations for the Taylor impact test case at 227 ms^{-1} compared with FLAG calculations for mimetic and least squares gradient implementation - (a) Locus of head and tail (b) Velocity history of head and tail



(a) Least Squares Implementation

(b) Mimetic Implementation

Figure 21: Comparison with FLAG calculations for mimetic and least squares gradient estimate implementation - Contours of effective plastic strain ϵ_p and configuration of the bar at T = 80 μs for the impact of copper rod at 227 ms^{-1}



(a) Initial Mesh at T = 0 ms^{-1} (b) Enlarged View of the Mesh at T = 0 ms^{-1}

Figure 22: (a) Initial topology of the mesh for the Taylor impact test case at 227 ms^{-1} (b) Enlarged view of the mesh at time T = 0 ms^{-1}

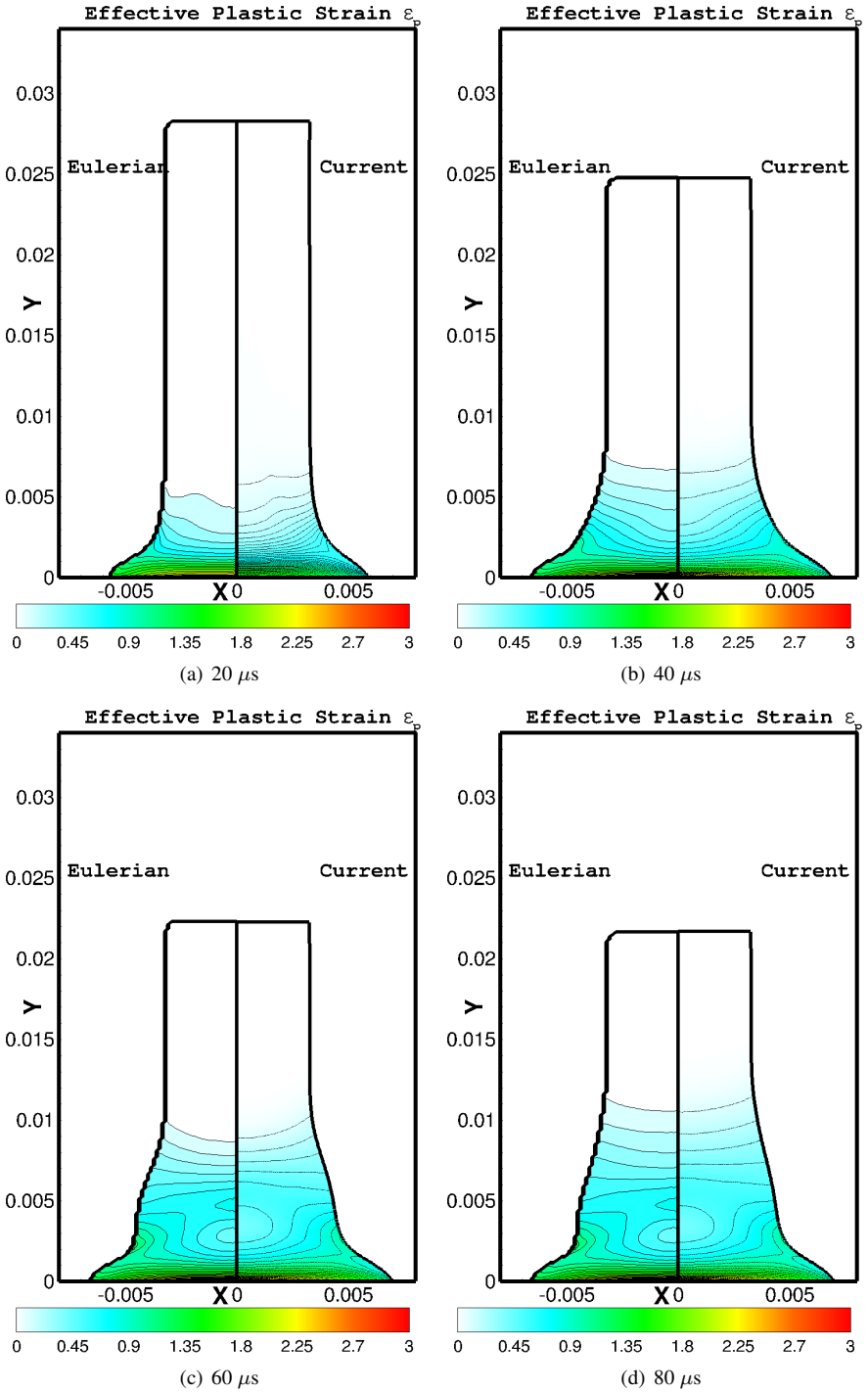
evolution of the bar at different instants in time is presented in Figure 25(d). Compatibility of the scheme with total energy conservation is evident from the energy conversion history plotted in Figure 25(d).

7.7. Axisymmetric Taylor Bar Experiment: Impact of a Copper Rod Over a Rigid Substrate at $190\ ms^{-1}$

In this section, the impact of a copper rod at 190 ms^{-1} is considered. The problem has been solved in [61] and is considered here to demonstrate the effects of nonlinear yield models. Moreover, the availability of experimental results (also presented in [61]) makes this an attractive test case to benchmark the current implementation. For this test case, the length and the radius of the bar are set at 25.4 mm and 3.81 mm in order to match the experimental results. Both Johnson-Cook (Eq (C.2)) and Steinberg-Guinan (Eq (C.4)) yield models, that includes the thermal softening effects, are employed for this calculation. The calculations are performed on a mesh comprising of polygonal elements. The mesh was constructed from a centroidal Voronoi algorithm with 150×50 generators distributed in the axial and radial directions leading to 7291 elements in the mesh. The calculations are carried upto 80 μs in RZ geometry. The plots from the current calculations are displayed in Figure 27. The enlarged view of the mesh at two different instants in time are presented in Figures 28(a) and 28(b).

7.7.1. Cartesian Mesh Calculation - Comparison with PAGOSA and Experimental Results

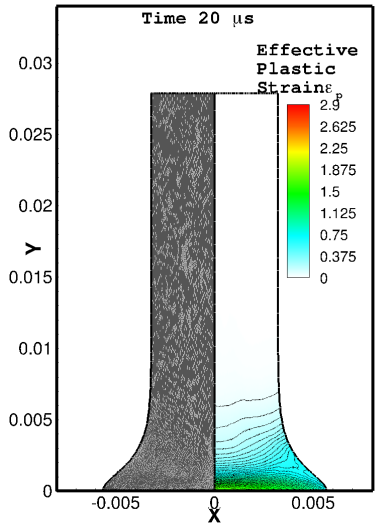
Similar to the 227 ms^{-1} impact velocity test case, significant features such as the final mushroom shape of the bar with the bulging and jetting of the foot are captured very well. However, the predictions made by the two yield models differ significantly from each other. In Figure 28(c), the results from the current calculations are compared with the experimental and PAGOSA results. PAGOSA calculations were computed on a three-dimensional mesh with $40 \times 40 \times 110$ cells in x,y and z directions respectively. As can be inferred from the figure, results obtained using the Steinberg-Guinan yield model show excellent agreement between PAGOSA and current calculations. Similarly, the plots corresponding to Johnson-Cook yield model show very good agreement between the two calculations. For the Steinberg-Guinan yield model, the overall shape of the bar and the final radius of the foot agree well with the experimental results. Both calculations fail to predict the final length of the bar that matches will with the experimental



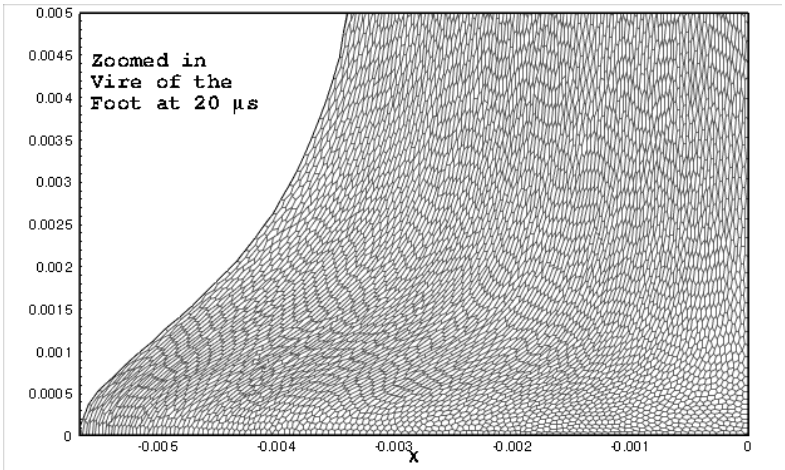
(a) 20 μs (b) 40 μs

(c) 60 μs (d) 80 μs

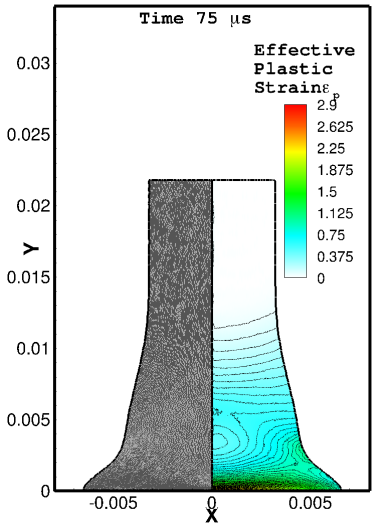
Figure 23: Unstructured polygonal mesh computations compared with Eulerian calculations from [38] for the Taylor impact test case at 227 ms^{-1} - Contours of effective plastic strain ϵ_p at different instants in time for the impact of a copper rod at 227 ms^{-1}.



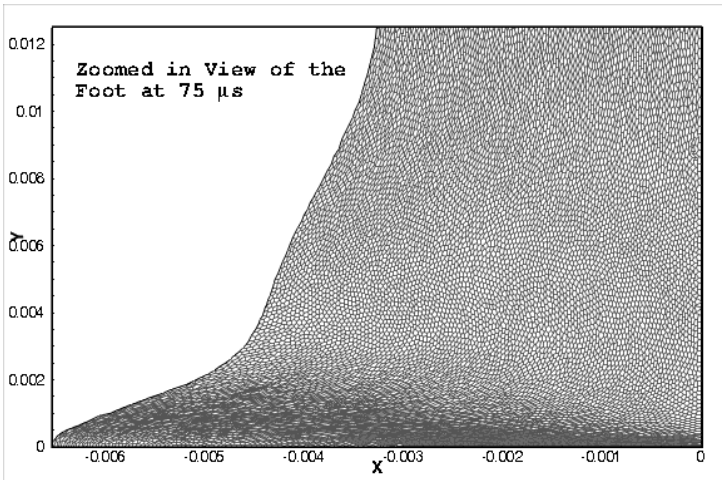
(a) Mesh at 20 μs



(b) Enlarged View at 20 μs

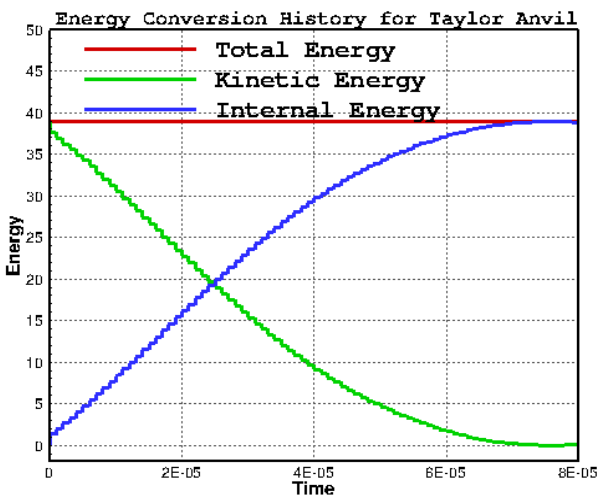


(c) Mesh at 75 μs

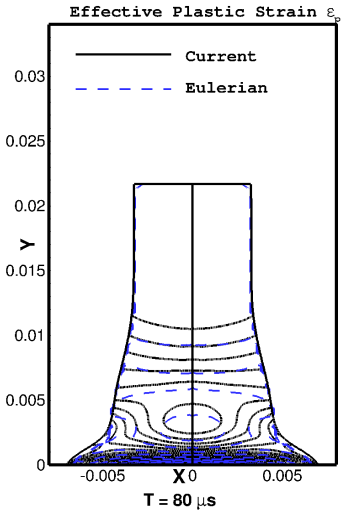


(d) Enlarged View at 75 μs

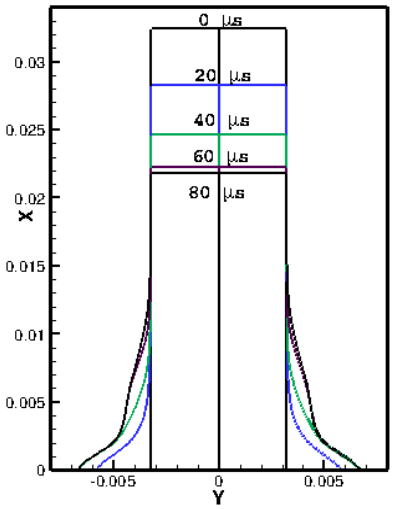
Figure 24: Unstructured polygonal mesh computations for the Taylor impact test case at 227 ms^{-1}: Contours of effective plastic strain ϵ_p and the topology of the mesh at 20 μs and 75 μs are displayed in (a) and (c) respectively and the corresponding enlarged view of the mesh are shown in (b) and (d)



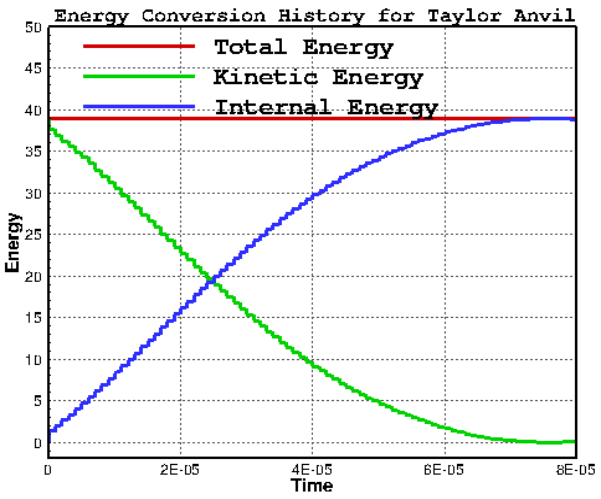
(a) Locus of Head and Tail



(b) Final Configuration of the Rod at 80 μs



(c) Evolution of the Rod



(d) Energy Conversion History

Figure 25: Plots for the unstructured polygonal mesh computations for the Taylor impact test case at 227 ms^{-1}: (a) Locus of head and tail compared with Eulerian computations (b) Contours of effective plastic strain ϵ_p and configuration of the rod at T = 75 μs compared with Eulerian computations and (c) Evolution and configuration of the rod at different instants in time (d) History of conversion of energy

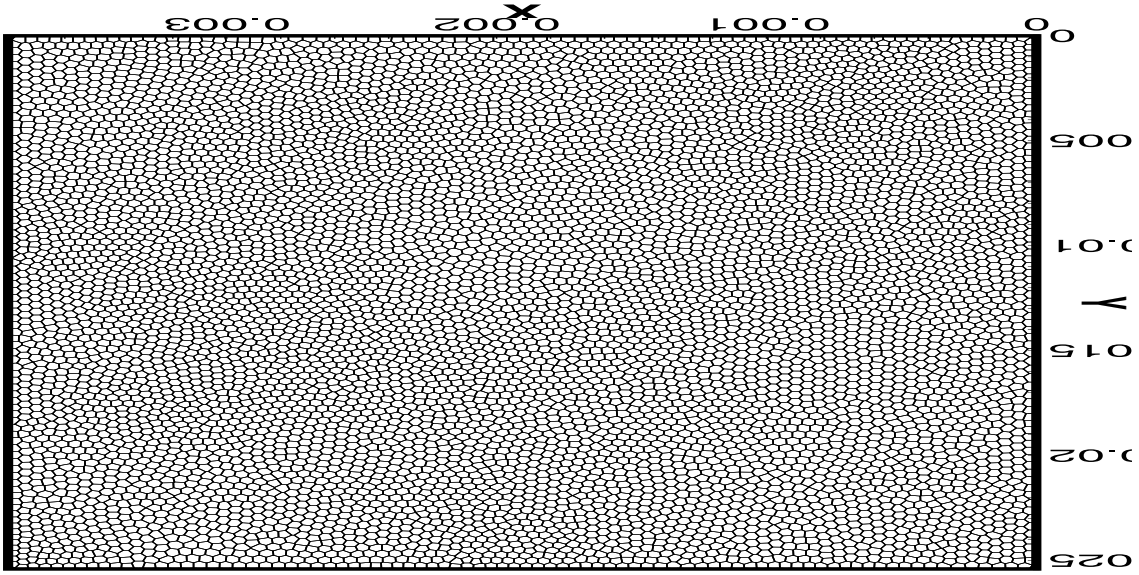


Figure 26: Snapshot of the initial unstructured mesh comprising of polygonal elements for the impact of a copper rod at 190 ms^{-1}

results. For the Johnson-Cook yield model, the final radius of the foot predicted by the two calculations agrees well with the experimental results. Nevertheless, the overall predictions from the current implementation of the two yield models follow closely the contours of PAGOSA calculation.

8. Conclusions

Appendix A. Equation of State

The incomplete Mie-Grüneisen formulation based equation of state is used. Pressure, specific internal energy and specific volume ($v = \frac{1}{\rho}$) are coupled through a relation of the form:

$$P(e, v) \approx \Gamma(v)\frac{(e - e_{ref}(v))}{v} + P_{ref}(v) = \Gamma\frac{e}{v} + f(v) \tag{A.1}$$

where e_{ref} and P_{ref} denote the reference internal energy and pressure at 0 K. $\Gamma(v)$ is the Grüneisen parameter defined as

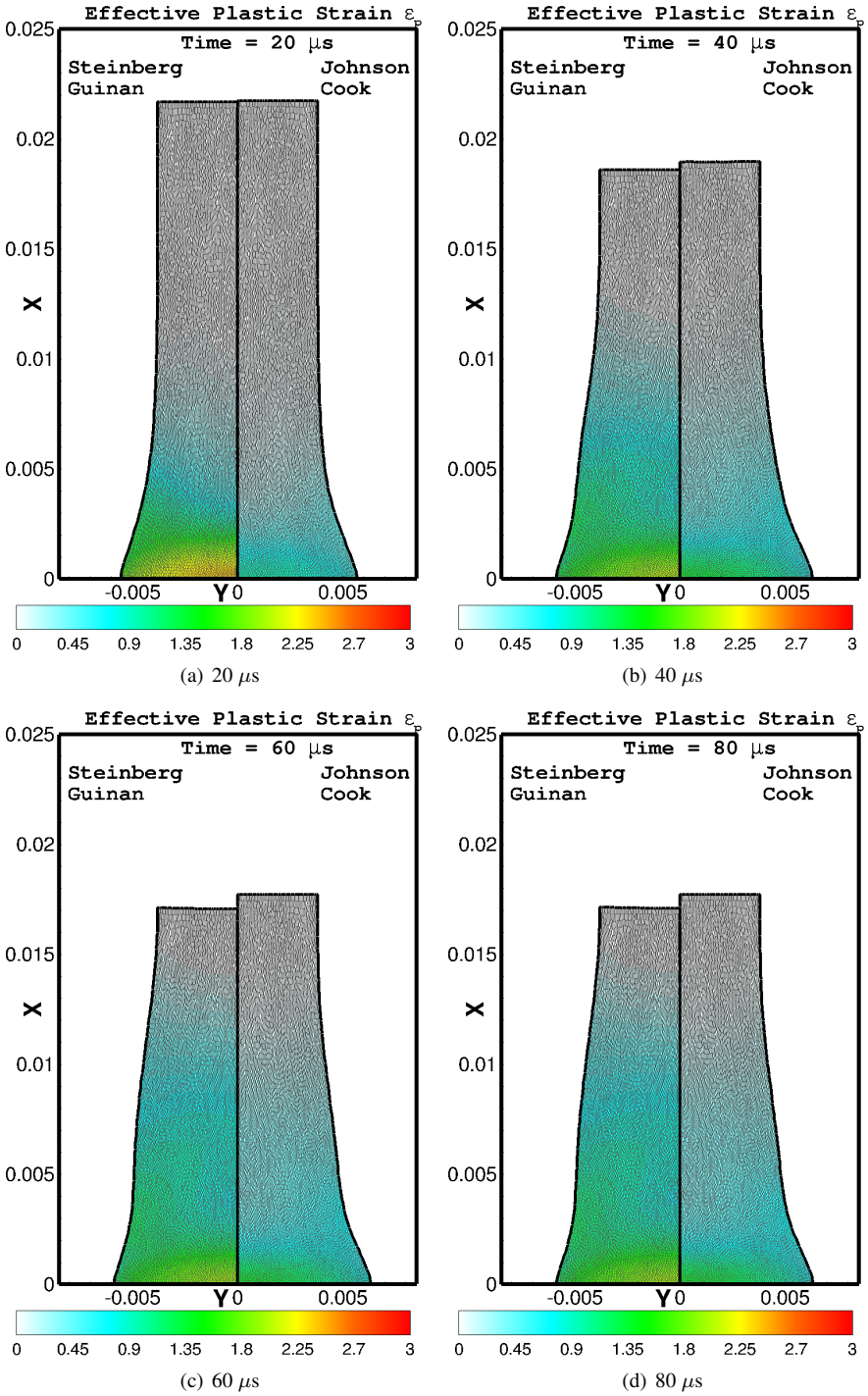
$$\Gamma(v) = v\left(\frac{\partial P}{\partial e}\right)_v = \frac{\Gamma_0 v}{v_0} \tag{A.2}$$

where $v_0 = \frac{1}{\rho_0}$ is the specific volume of the unstressed material. Accommodating for negative pressure (tension) and preserving the positivity of sound speed-squared, the function $f(v)$ in Eq (A.1) is written as

$$f(v) = \begin{cases} \frac{\rho_0 c_0^2 \Phi}{(1-s\Phi^2)^2}[1 - \frac{\Gamma}{2v}(v_0 - v)] \text{ if } v \leq v_0 \\ c_0^2(\frac{1}{v} - \frac{1}{v_0}) \text{ if } v > v_0 \end{cases} \tag{A.3}$$

In the above expression, $\Phi = 1 - \frac{v}{v_0}$, c_0 is the bulk sound speed and s is related to the isentropic pressure derivative of the isentropic bulk modulus [36]. The c_0 & s coefficients relate the shock speed U_s and the particle velocity U_p. Experiments on solids provide a relation between U_s and U_p. A first approximation consists of a linear relation given as

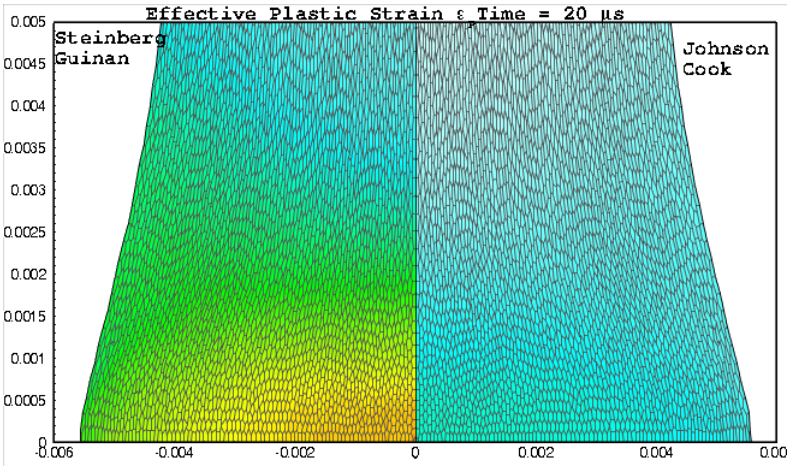
$$U_s = c_o + sU_p \tag{A.4}$$



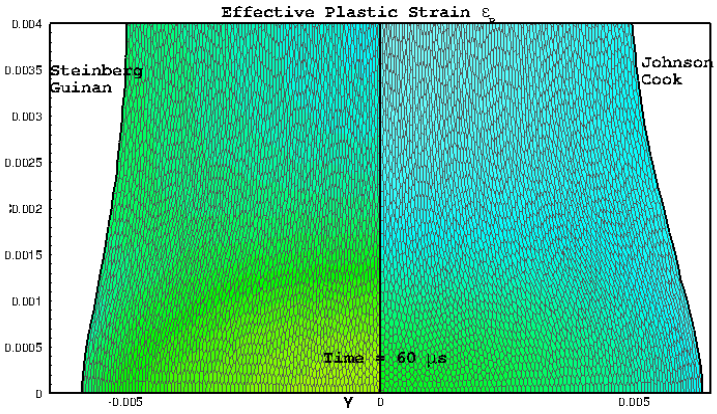
(a) 20 μs (b) 40 μs

(c) 60 μs (d) 80 μs

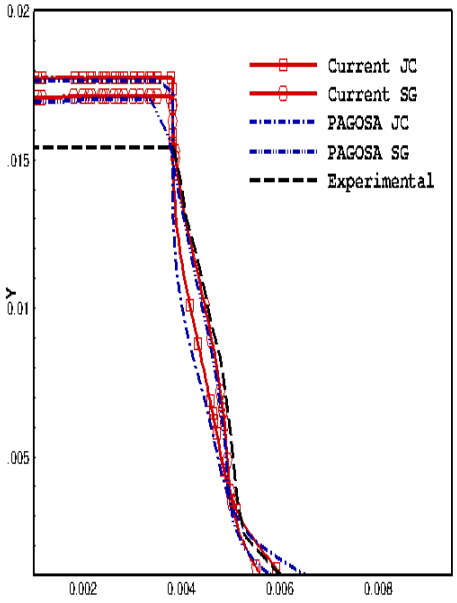
Figure 27: Unstructured polygonal mesh computations for the impact of a copper rod at 190 ms^{-1} with Johnson-Cook and Steinberg-Guinan material models - Contours of effective plastic strain ϵ_p at different instants in time



(a) Enlarged View of the Foot at 20 μs



(b) Enlarged View of the Foot at 60 μs



(c) Comparison with PAGOSA and Experimental Results

Figure 28: Unstructured polygonal mesh computations for the impact of a copper rod at 190 ms^{-1} with Johnson-Cook and Steinberg-Guinan material models - (a)Enlarged view of the mesh and foot of the bar (b)Comparison with experimental and PAGOSA Results - Final configuration of the rod for the Johnson-Cook and Steinberg-Guinan yield models for the impact of a copper rod at 190 ms^{-1}.

The expression for the speed of sound in the material is then given by

$$c_{EOS}^2 = \left(\frac{\partial P}{\partial \rho}\right)_e + \frac{P}{\rho^2}\left(\frac{\partial P}{\partial e}\right)_\rho = \Gamma e + f'(v) + \Gamma\frac{P}{\rho} \tag{A.5}$$

The final expression for the speed of sound is given as

$$c^2 = c_{EOS}^2 + c_{SHEAR}^2 \tag{A.6}$$

where

$$c_{SHEAR}^2 = \frac{4}{3}\frac{G}{\rho} \tag{A.7}$$

Appendix B. Radial Return Mapping Algorithm

The radial return algorithm presened here is due to Ponthot et al [33, 34]. The consistency condition,

$$f(\boldsymbol{S}, \sigma_Y) = S_e - \sigma_Y = 0 \tag{B.1}$$

with linear hardening law given by:

$$\dot{\sigma}_Y = \sqrt{\frac{2}{3}} h \dot{\Lambda} \tag{B.2}$$

where σ_Y is the current yield stress and h (also called plastic modulus) is the slope of the effective stress versus effective plastic strain curve under uniaxial loading. Using Eq (24), the yield stress can be written as

$$\dot{\sigma}_Y = h\bar{\dot{\epsilon}}^P \tag{B.3}$$

When elastic deformation occurs, $f < 0$ and $\dot{\Lambda} = 0$. Plastic deformation is said to occur when the consistency condition holds true, $\dot{f}\left(\boldsymbol{S}, \sigma_Y\right) = 0$. In conjunction with the operator splitting algorithm, the predicted trial elastic state "tr" is obtained by freezing the plastic flow ($\bar{\boldsymbol{D}}^{PL} = 0$),

$$\dot{\boldsymbol{S}}_{tr} + \boldsymbol{S}_{tr}\boldsymbol{\Omega} - \boldsymbol{\Omega}\boldsymbol{S}_{tr} = 2G\bar{\boldsymbol{D}} \tag{B.4}$$

where $\boldsymbol{S}_{tr}$ is the trial elastic stress tensor. The trial elastic state is determined by numerically integrating the integral form of the stress evolution equation (Eq (B.4)). The plastic corrector step is enforced to bring the computed trial stress back to the yield surface:

$$\dot{\boldsymbol{S}}_{cor} = -2G\boldsymbol{D}^{PL} = -2G\dot{\Lambda}\boldsymbol{N} \tag{B.5}$$

where $\boldsymbol{S}_{cor}$ is the corrected stress update and $\boldsymbol{N}$ is the normal direction in which the return mapping is effected:

$$\boldsymbol{N}_{tr} = \frac{\boldsymbol{S}_{tr}}{\sqrt{\boldsymbol{S}_{tr} : \boldsymbol{S}_{tr}}} \tag{B.6}$$

In discrete form, the plastic corrector step can be obtained by integrating Eq (B.5) to obtain

$$\boldsymbol{S}_{cor} = \boldsymbol{S}_{tr} - 2G\boldsymbol{N}_{tr}\zeta \tag{B.7}$$

where

$$\zeta = \int_{t_0}^{t_1} \dot{\Lambda} dt \tag{B.8}$$

with t_0 and t_1 denoting the beginning and end of the time interval of integration. The parameter ζ is determined by enforcing the generalized consistency condition, $f = 0$, at time t = t_1.

$$f = \sqrt{\frac{3}{2}\left[(\boldsymbol{S}_{tr} - 2G\boldsymbol{N}_{tr}\zeta) : (\boldsymbol{S}_{tr} - 2G\boldsymbol{N}_{tr}\zeta)\right]} - \sigma_Y = 0 \tag{B.9}$$

Integrating Eqs (24) & (B.3) in time, we get

$$\bar{\epsilon}_1^P = \bar{\epsilon}_0^P + \sqrt{\frac{2}{3}}\zeta \tag{B.10}$$

$$\sigma_Y^1 = \sigma_Y^0 + \sqrt{\frac{2}{3}}h\zeta \tag{B.11}$$

where "0" and "1" denote the values at t_0 and t_1, respectively. Substituting for σ_Y^1, Eq (B.9) is simplified

$$\left(4G^2 - \frac{4}{9}\right)\zeta^2 - \left(4G\sqrt{\boldsymbol{S}_{tr} : \boldsymbol{S}_{tr}} + \frac{4}{3}\sqrt{\frac{2}{3}}h\right)\zeta + \left(\boldsymbol{S}_{tr} : \boldsymbol{S}_{tr} - \frac{2}{3}\sigma_Y^{0^2}\right) = 0 \tag{B.12}$$

to obtain

$$\zeta = \frac{\sqrt{\boldsymbol{S}_{tr} : \boldsymbol{S}_{tr}} - \frac{2}{3}\sigma_Y^0}{2G\left(1 + \frac{h}{3G}\right)} \tag{B.13}$$

Thus, once ζ is obtained, the correction for the predicted deviatoric stresses is performed using Eq (B.5) and the consistency condition is enforced iteratively.

Appendix C. Material Models

Material models are required to determine the flow (yield) stress to enforce the consistency conditions in the return mapping algorithm. Three material models that are used in this work include the Prandtl-Ruess [32], Johnson-Cook [66] and Steinberg-Guinan material model [67]. The expressions for flow stress corresponding to these models are given below:.

$$\text{Prandtl Reuss: } \sigma_Y = A + B(\dot{\bar{\epsilon}}^P)^n \tag{C.1}$$

$$\text{Johnson-Cook: } \sigma_Y = \left(A + B(\dot{\bar{\epsilon}}^P)^n\right)\left(1 + Cln\left(\frac{\dot{\bar{\epsilon}}^P}{\dot{\bar{\epsilon}}_0^P}\right)\right)(1 - \theta^m) \tag{C.2}$$

$$\text{Steinberg-Guinan: } \sigma_Y = Y_0[1 + \varkappa\dot{\bar{\epsilon}}^P]^n[1 + bP\nu^{\frac{1}{3}} - \delta(T - T_0)] \tag{C.3}$$

where $\nu = \frac{\rho_0}{\rho}$, $A = Y_0$ is the initial yield stress for the material with B, C, n, m, $\dot{\bar{\epsilon}}_0^P$ denoting the model constants and $\theta = \frac{T - T_0}{T_m - T_0}$, where T_m and T_0 are material melting and the reference room temperatures respectively. The additional conditions on the temperature, shear modulus and yield stress for the Steinberg-Guinan models are:

$$Y_0[1 + \varkappa\dot{\bar{\epsilon}}^P]^n \leq Y_{max} \tag{C.4}$$

$$Y = 0 \text{ for } T > T_m \text{ where} \tag{C.5}$$

$$T_m = T_{m0}\nu^{\frac{2}{3}}e^{[2\Gamma_0(1-\nu)]} \tag{C.6}$$

$$G = G_0[1 + bP\nu^{\frac{1}{3}} - (T - T_0)] \tag{C.7}$$

[1] J. Hyman, M. Shashkov, Natural discretizations for the divergence, gradient, and curl on logically rectangular grids, Computers & Mathematics With Applications 33 (4) (1997) 81–104.
URL http://gateway.isiknowledge.com/gateway/Gateway.cgi?GWVersion=2&SrcAuth=Alerting&SrcApp=Alerting&DestApp=WOS&DestLinkType=FullRecord;KeyUT=A1997WM41200009

[2] J. Hyman, M. Shashkov, Adjoint operators for the natural discretizations of the divergence, gradient and curl on logically rectangular grids, Applied Numerical Mathematics 25 (4) (1997) 413–442.

URL http://gateway.isiknowledge.com/gateway/Gateway.cgi?GWVersion=2&SrcAuth=Alerting&SrcApp=Alerting&DestApp=WOS&DestLinkType=FullRecord;KeyUT=A1997YJ72500006

[3] V. Subramanian, J. Perot, Higher-order mimetic methods for unstructured meshes, Journal of Computational Physics 219 (1) (2006) 68 – 85. doi:10.1016/j.jcp.2006.03.028.
URL http://www.sciencedirect.com/science/article/pii/S0021999106001409

[4] M. L. Wilkins, Computer Simulation of Dynamic Phenomena, Springer Series in Scientific Computing, Springer-Verlag, 1999.

[5] M. L. Wilkins, Calculation of elastic-plastic flow, Methods Computational Physics 3.

[6] J. VonNeumann, R. D. Richtmyer, A method for the numerical calculation of hydrodynamic shocks 21 (3) (1950) 232–237. doi:DOI:10.1063/1.1699639.
URL http://dx.doi.org/doi/10.1063/1.1699639

[7] E. J. Caramana, M. J. Shashkov, P. P. Whalen, Formulations of artificial viscosity for multi-dimensional shock wave computations, Journal of Computational Physics 144 (1998) 70–97. doi:10.1006/jcph.1998.5989.
URL http://dl.acm.org/citation.cfm?id=295586.295592

[8] E. J. Caramana, R. Loubère, "curl-q": a vorticity damping artificial viscosity for essentially irrotational Lagrangian hydrodynamics calculations, Journal of Computational Physics 215 (2006) 385–391. doi:10.1016/j.jcp.2005.11.018.
URL http://dl.acm.org/citation.cfm?id=1148052.1148053

[9] E. Caramana, M. Shashkov, Elimination of artificial grid distortion and hourglass-type motions by means of lagrangian subzonal masses and pressures, Journal of Computational Physics 142 (2) (1998) 521 – 561. doi:10.1006/jcph.1998.5952.
URL http://www.sciencedirect.com/science/article/pii/S0021999198959526

[10] E. J. Caramana, P. P. Whalen, Numerical preservation of symmetry properties of continuum problems, Journal of Computational Physics 141 (1998) 174–198. doi:10.1006/jcph.1998.5912.
URL http://portal.acm.org/citation.cfm?id=287244.287253

[11] D. E. Burton, Multidimensional discretization of conservation laws for unstructured polyhedral grids, Tech. rep., Lawrence Livermore National Laboratory, CA, presented at the SAMGOP-94: 2nd international workshop on analytical methods and process optimization in fluid and gas mechanics, Arzamas (Russian Federation), 10-16 Sep 1994 (Aug 22 1994).

[12] E. J. Caramana, D. E. Burton, M. J. Shashkov, P. P. Whalen, The construction of compatible hydrodynamics algorithms utilizing conservation of total energy, Journal of Computational Physics 146 (1998) 227–262. doi:10.1006/jcph.1998.6029.
URL http://portal.acm.org/citation.cfm?id=302638.302658

[13] J. C. Campbell, M. J. Shashkov, A tensor artificial viscosity using a mimetic finite difference algorithm, Journal of Computational Physics 172 (2001) 739–765. doi:http://dx.doi.org/10.1006/jcph.2001.6856.
URL http://dx.doi.org/10.1006/jcph.2001.6856

[14] L. G. Margolin, M. Shashkov, P. K. Smolarkiewicz, A discrete operator calculus for finite fifference approximations, Computer Methods in Applied Mechanics and Engineering 187 (3-4) (2000) 365 – 383. doi:DOI: 10.1016/S0045-7825(00)80001-8.
URL http://www.sciencedirect.com/science/article/pii/S0045782500800018

[15] B. Despres, C. Mazeran, Lagrangian gas dynamics in two dimensions and Lagrangian systems, Archive for Rational Mechanics and Analysis 178 (2005) 327–372, 10.1007/s00205-005-0375-4.
URL http://dx.doi.org/10.1007/s00205-005-0375-4

[16] G. Carré, S. Del Pino, B. Després, E. Labourasse, A cell-centered Lagrangian hydrodynamics scheme on general unstructured meshes in arbitrary dimension, Journal of Computational Physics 228 (2009) 5160–5183. doi:10.1016/j.jcp.2009.04.015.
URL http://portal.acm.org/citation.cfm?id=1552584.1553086

[17] P.-H. Maire, A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry, Journal of Computational Physics 228 (2009) 6882–6915. doi:10.1016/j.jcp.2009.06.018.
URL http://portal.acm.org/citation.cfm?id=1618877.1619060

[18] A. Barlow, P. Roe, A cell centred Lagrangian godunov scheme for shock hydrodynamics, Computers and Fluids 46 (1) (2011) 133 – 136, 10th ICFD Conference Series on Numerical Methods for Fluid Dynamics (ICFD 2010). doi:DOI: 10.1016/j.compfluid.2010.07.017.
URL http://www.sciencedirect.com/science/article/pii/S0045793010001921

[19] D. Burton, T. Carney, N. Morgan, S. Runnels, M. Shashkov, Exploration of a cell-centered Lagrangian hydrodynamics method, Conference proceedings, Los Alamos National Laboratory, presented at the SIAM Conference on Computational Science and Engineering Reno, Nevada (February 28-March 4, 2011 2011).

[20] D. E. Burton, T. C. Carney, N. R. Morgan, S. R. Runnels, S. Sambasivan, M. J. Shashkov, A cell-centered Lagrangian godunov like method for solid dynamicsSubmitted to Computers & Fluids.

[21] F. Addessio, J. Baumgardner, J. Dukowicz, N. Johnson, B. Kashiwa, R. Rauenzahn, C. Zemach, Caveat: A computer code for fluid dynamics problems with large distortion and internal slip, Tech. rep., Los Alamos National Laboratory (1986).

[22] J. G. Trulio, K. R. Trigger, Numerical solution of the one-dimensional hydrodynamic equations in an arbitrary time-dependent coordinate system, Tech. Rep. UCRL-6522, Lawrence Livermore National Laboratory (1961).

[23] P. P. Whalen, Algebraic limitations on two-dimensional hydrodynamics simulations, Journal of Computational Physics 124 (1996) 46–54. doi:http://dx.doi.org/10.1006/jcph.1996.0043.
URL http://dx.doi.org/10.1006/jcph.1996.0043

[24] L. Malvern, Introduction to the Mechanics of a Continuous Medium, Prentice Hall, 1969.

[25] M. E. Gurtin, E. Fried, L. Anand, The Mechanics and Thermodynamics of Continua, Cambridge University Press, 2009.

[26] Compatible Spatial Discretizations, Vol. 142 of The IMA Volumes in Mathematics and its Applications, Springer - Verlag, Berlin, 2006, Ch. Principles of Mimetic Discretizations of Differential Operators, pp. 89–119.

[27] M. J. Shashkov, Conservative finite-difference methods on general grids, CRC Press, 1996.

[28] S. K. Sambasivan, M. J. Shashkov, D. E. Burton, A finite volume cell-centered lagrangian hydrodynamics approach for solids in general unstructured grids.

[29] S. K. Sambasivan, M. J. Shashkov, D. E. Burton, A cell-centered lagrangian finite volume approach for computing elasto-plastic response of solids in cylindrical axisymmetric geometries, submitted to Journal of Computational Physics.
[30] T. J. Barth, H. Deconinck, High order methods for computational physics, Vol. 9, Springer-Verlag Inc., New York, NY, USA, 1999.
[31] S. K. Sambasivan, M. J. Shashkov, D. E. Burton, Exploration of new limiter schemes for stress tensors in Lagrangian and ALE hydrocodesIn press, Computers & Fluids.
[32] A. S. Khan, S. Huang, Continuum Theory of Plasticity, Wiley-Interscience, 1995.
[33] J.-P. Ponthot, An extension of the radial return algorithm to account for rate-dependent effects in frictional contact and visco-plasticity, Journal of Materials Processing Technology 80-81 (1998) 628 – 634. doi:DOI: 10.1016/S0924-0136(98)00125-3.
URL http://www.sciencedirect.com/science/article/B6TGJ-3VY0CX0-3N/2/1d4539634207f4ecded362ddb329c771
[34] J.-P. Ponthot, Unified stress update algorithms for the numerical simulation of large deformation elasto-plastic and elasto-viscoplastic processes, International Journal of Plasticity 18 (1) (2002) 91 – 126. doi:DOI: 10.1016/S0749-6419(00)00097-8.
URL http://www.sciencedirect.com/science/article/B6TWX-44PKCX7-4/2/1f181580c85f63f185f3204ceb8b1c43
[35] R. Loubère, M. Shashkov, B. Wendroff, Short note: Volume consistency in a staggered grid Lagrangian hydrodynamics scheme, Journal of Computational Physics 227 (2008) 3731–3737. doi:10.1016/j.jcp.2008.01.006.
URL http://portal.acm.org/citation.cfm?id=1349879.1349927
[36] L. B. Tran, H. UdayKumar, A particle levelset based sharp interface Cartesian grid method for impact, penetration, and void collapse, Journal of Computational Physics 193 (2004) 469–510.
[37] H.-C. Wu, Continuum Mechanics and Plasticity, Chapman & Hall, CRC Press, 2004.
[38] S. K. Sambasivan, Sharp interface cartesian grid hydrocode, Ph.D. thesis, The University of Iowa (2010).
[39] S. K. Sambasivan, H. Udaykumar, A sharp interface method for high-speed multi-material flows: strong shocks and arbitrary material pairs, International Journal of Computational Fluid Dynamics 25 (3) (2011) 139–162. arXiv:http://www.tandfonline.com/doi/pdf/10.1080/10618562.2011.558011, doi:10.1080/10618562.2011.558011.
URL http://www.tandfonline.com/doi/abs/10.1080/10618562.2011.558011
[40] S. K. Sambasivan, A. Kapahi, H. UdayKumar, Simulation of high speed impact, penetration and fragmentation problems on locally refined cartesian grids, submitted to Journal of Computational Physics.
[41] P.-H. Maire, A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes, Journal of Computational Physics 228 (7) (2009) 2391 – 2425. doi:DOI: 10.1016/j.jcp.2008.12.007.
URL http://www.sciencedirect.com/science/article/pii/S0021999108006359
[42] N. M. Bessonov, D. J. Song, Application of vector calculus to numerical simulation of continuum mechanics problems, Journal of Computational Physics 167 (2001) 22–38. doi:10.1006/jcph.2000.6653.
URL http://portal.acm.org/citation.cfm?id=371642.371650
[43] J. Morel, R. M. Roberts, M. J. Shashkov, A local support-operators diffusion discretization scheme for quadrilateralr-zmeshes, Journal of Computational Physics 144 (1) (1998) 17 – 51. doi:10.1006/jcph.1998.5981.
URL http://www.sciencedirect.com/science/article/pii/S0021999198959812
[44] M. J. Shashkov, Conservative Finite Difference Methods on General Grids, CRC Press, 1996.
[45] P.-H. Maire, A unified sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids, International Journal for Numerical Methods in Fluids 65 (11-12) (2011) 1281–1294. doi:10.1002/fld.2328.
URL http://dx.doi.org/10.1002/fld.2328
[46] J. K. Dukowicz, A general, non-iterative riemann solver for Godunov's method, Journal of Computational Physics 61 (1) (1985) 119 – 137. doi:10.1016/0021-9991(85)90064-6.
URL http://www.sciencedirect.com/science/article/pii/0021999185900646
[47] G. Kluth, B. Després, Discretization of hyperelasticity on unstructured mesh with a cell-centered Lagrangian scheme, Journal of Computational Physics 229 (2010) 9092–9118. doi:http://dx.doi.org/10.1016/j.jcp.2010.08.024.
URL http://dx.doi.org/10.1016/j.jcp.2010.08.024
[48] E. J. Caramana, C. L. Rousculp, D. E. Burton, A compatible, energy and symmetry preserving Lagrangian hydrodynamics algorithm in three-dimensional cartesian geometry, Journal of Computational Physics 157 (2000) 89–119. doi:10.1006/jcph.1999.6368.
URL http://portal.acm.org/citation.cfm?id=342135.342138
[49] B. van Leer, Towards the ultimate conservative difference scheme v. a second-order sequel to godunov's method, Journal of Computational Physics 135 (1997) 229–248. doi:10.1006/jcph.1997.5704.
URL http://dl.acm.org/citation.cfm?id=260709.260736
[50] T. J. Barth, P. O. Fredrickson, Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction, in: 28th AIAA Aerospace Sciences Meeting, Reno, NV, 1990.
[51] M. Delanaye, Y. Liu, Quadratic reconstruction finite volume schemes on 3d arbitrary unstructured polyhedral grids, in: 14th Computational Fluid Dynamics Conference, Norfolk, VA, 1999.
[52] L. G. Margolin, M. Shashkov, Second-order sign-preserving conservative interpolation (remapping) on general grids, Journal of Computational Physics 184 (2003) 266–298. doi:10.1016/S0021-9991(02)00033-5.
URL http://dl.acm.org/citation.cfm?id=775673.775686
[53] S. Sambasivan, H. UdayKumar, Sharp interface simulations with adaptive mesh refinement for multi-material dynamics in strongly shocked flows, Computers & Fluids In Press, Accepted for Publication.
[54] P. Roe, Approximate riemann ssolvers, parameter vectors, and difference-schemes, Journal of Computational Physics 43 (2) (1981) 357–372.
URL http://gateway.isiknowledge.com/gateway/Gateway.cgi?GWVersion=2&SrcAuth=Alerting&SrcApp=Alerting&DestApp=WOS&DestLinkType=FullRecord;KeyUT=A1981MV92300009
[55] T. J. Barth, Von Karman Lecture Series 1994-05, Von Karman Institute for Fluid Dynamics,Rhode-Saint-Genese, Belgium, 1994, Ch. Aspects of Unstructured Grids and Finite-Volume Solvers for the Euler and Navier-Stokes Equations.
[56] T. J. Barth, An Introduction to Recent Developments in Theory and Numerics for Conservation Laws,Proceedings of the International School

on Theory and Numerics for Conservation Laws, Lecture Notes in Computational Science and Engineering, Springer - Verlag, Berlin, 1997, Ch. Numerical Methods for Gasdynamic Systems on Unstructured Meshes, pp. 195 – 284.

[57] V. Venkatakrishnan, Convergence to steady state solutions of the Euler equations on unstructured grids with limiters, Journal of Computational Physics 118 (1) (1995) 120 – 130. doi:10.1006/jcph.1995.1084.
URL http://www.sciencedirect.com/science/article/pii/S0021999185710844

[58] P.-H. Maire, R. Loubère, P. Váchal, Staggered Lagrangian discretization based on cell-centered riemann solver and associated hydrodynamics scheme, Communication in Computational Physics 10 (4) (2011) 940–978.

[59] G. Luttwak, J. Falcovitz, Slope limiting for vectors: A novel vector limiting algorithm, International Journal for Numerical Methods in Fluids 65 (11-12). doi:10.1002/fld.2367, pages = 1365?1375.
URL http://dx.doi.org/10.1002/fld.2367

[60] W. N. Weseloh, S. P. Clancy, J. W. Painter, PAGOSA physics manual, Tech. rep., Group XTD-1 Silverton Code Project, Los Alamos National Laboratory (2010).

[61] W. N. Weseloh, PAGOSA sample problems, Tech. rep., Los Alamos National Laboratory (2011).

[62] B. P. Howell, G. J. Ball, A free-Lagrange augmented godunov method for the simulation of elastic-plastic solids, Journal of Computational Physics 175 (1) (2002) 128 – 167. doi:10.1006/jcph.2001.6931.
URL http://www.sciencedirect.com/science/article/pii/S0021999101969311

[63] S. C. Team, PAGOSA sample problems, Tech. rep., Los Alamos National Laboratory (2005).

[64] M. B. Tyndall, Numerical modelling of shocks in solids with elastic-plastic conditions, Shock Waves 3 (1993) 55–66, 10.1007/BF01414748.
URL http://dx.doi.org/10.1007/BF01414748

[65] G. I. Taylor, The use of flat-ended projectiles for determining dynamic yield stress. i. theoretical considerations, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences 194 (1038) (1948) 289–299. arXiv:http://rspa.royalsocietypublishing.org/content/194/1038/289.full.pdf+html, doi:10.1098/rspa.1948.0081.
URL http://rspa.royalsocietypublishing.org/content/194/1038/289.abstract

[66] G. R. Johnson, W. H. Cook, Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures, Engineering Fracture Mechanics 21 (1) (1985) 31 – 48.

[67] D. J. Steinberg, S. G. Cochran, M. W. Guinan, A constitutive model for metals applicable at high strain rate 51 (3) (1980) 1498–1504. doi:DOI:10.1063/1.327799.
URL http://dx.doi.org/doi/10.1063/1.327799