3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity

Raphaël Loubère¹,*,†, Pierre-Henri Maire² and Pavel Váchal³

¹Institut de Mathématiques de Toulouse, CNRS, Université de Toulouse, Toulouse, France
²CEA-CESTA, BP 2, 33114, Le Barp, France
³Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, Prague, Czech Republic

SUMMARY

The aim of the present work is the 3D extension of a general formalism to derive a staggered discretization for Lagrangian hydrodynamics on unstructured grids. The classical compatible discretization is used; namely, momentum equation is discretized using the fundamental concept of subcell forces. Specific internal energy equation is obtained using total energy conservation. The subcell force is derived by invoking the Galilean invariance and thermodynamic consistency. A general form of the subcell force is provided so that a cell entropy inequality is satisfied. The subcell force consists of a classical pressure term plus a tensorial viscous contribution proportional to the difference between the node velocity and the cell-centered velocity. This cell-centered velocity is an extra degree of freedom solved with a cell-centered approximate Riemann solver. The second law of thermodynamics is satisfied by construction of the local positive definite subcell tensor involved in the viscous term. A particular expression of this tensor is proposed. A more accurate extension of this discretization both in time and space is also provided using a piecewise linear reconstruction of the velocity field and a predictor-corrector time discretization. Numerical tests are presented in order to assess the efficiency of this approach in 3D. Sanity checks show that the 3D extension of the 2D approach reproduces 1D and 2D results. Finally, 3D problems such as Sedov, Noh, and Saltzman are simulated. Copyright © 2012 John Wiley & Sons, Ltd.

Received 24 February 2012; Revised 18 July 2012; Accepted 20 August 2012

KEY WORDS: Lagrangian hydrodynamics; compressible flow; staggered scheme; 3D; artificial viscosity; Riemann solver; unstructured mesh

1. INTRODUCTION

The aim of the present work is twofold: to develop the 3D version of a general formalism, which was first derived in [1], and to numerically validate this formalism via a 3D staggered Lagrangian hydrodynamics code.

One classical staggered Lagrangian hydrodynamics compatible scheme, which is vastly used for designing 3D code, dates back to [2–4]. The staggered discretization of variables (kinematic variables located at nodes, thermodynamic ones at cell centers) allows the scheme to fulfill naturally the geometric conservation law (GCL) compatibility requirement and, at the same time, provides the construction of a natural discrete divergence operator. The discretizations of momentum and specific internal energy are derived from each other by use of the important concepts of subcell mass, subcell force, and total energy conservation [4]. This compatible hydrodynamics algorithm is thus designed to conserve momentum and total energy exactly in discrete form by using the adjointness
The dissipation of kinetic energy into internal energy through shock waves is ensured by means of artificial viscosity, which can be edge based [5], tensorial [6], or more advanced [7, 8]. This mechanism leads to a dissipation that is compatible with the second law of thermodynamics. The subcell pressure method is also used for the control of hourglass-type motion [9]. Finally, the time integration method is a predictor-corrector technique, which is detailed in [4]. The extension of this compatible Lagrangian hydrodynamics algorithm to unstructured grids, where each zone is a polygon with an arbitrary number of sides, has been presented in [10]. The subcell formalism greatly simplifies the 3D version of the scheme [11].

More recently, cell-centered Lagrangian schemes were developed [12–14]. Their primary variables are the cell-centered density, the velocity, and the total energy. These schemes solve the difficulty of defining a nodal velocity to move the mesh consistently with the GCL. An approximate Riemann problem is constructed at each node, the solution of which provides the nodal velocity. A 3D version of such schemes has been presented for instance in [15–17].

In [1], the authors derived a staggered discretization for 2D Lagrangian hydrodynamics on unstructured grids. This derivation uses the concept of subcell discretization of the staggered Lagrangian schemes and also a cell-centered interpretation of the Riemann solver from the cell-centered Lagrangian schemes. This leads to an artificial viscosity force obtained by invoking the Galilean invariance and thermodynamic consistency. In Godunov methods, the dissipation of kinetic energy into internal energy is provided by a solution of a Riemann problem. The same mechanism is used in the framework of a 3D staggered scheme. The solution of the cell-centered Riemann problem provides an approximation of the cell-centered velocity \( U_c \), which is used to define the viscous part of the subcell force. Let us note that, independently, a very similar and interesting work in 2D has been published in [18]. This formulation allows a straightforward more accurate extension in space by constructing linear velocity vector field approximation with frame-invariant limitation, applicable on any mesh structure. A careful and sensitive vector limitation is a key issue to effective exploitation of the improvement gained by frame-invariant higher-order extension. We extend the approach proposed in [1] to obtain a 3D frame invariant vector limitation. As for temporal integration, we achieve second order in time by using a classical predictor-corrector time discretization. The 3D derivation of the scheme is very similar to the 2D one. The resulting viscosity force can be expressed with the help of a \( 3 \times 3 \) positive definite matrix \( M_{cp} \), so that the thermodynamic consistency is satisfied automatically by the viscous term \( M_{cp}(U_c - U_p) \). The method in [1] can be viewed as an extension of the work by Christensen [19], who noticed that under certain assumptions, the staggered Lagrangian schemes with artificial viscosity can be written in the same form as Godunov’s scheme with Harten, Lax, van Leer (HLL) approximate Riemann solver and stressed the potential synergy of both approaches (e.g., higher-order extension of simple staggered scheme by techniques typically used in the Godunov community, such as Total Variation Diminishing (TVD) limiters). The relationship between staggered Lagrangian and cell-centered Godunov methods from the viewpoint of shock-capturing mechanism has been discussed already in earlier works, for example, by Wilkins [20] or Dukowicz [21].

The paper is organized as follows. First, the governing equations and notation are stated. The compatible discretization is then derived from first principles. The fourth section deals with the definition of the fundamental object named subcell force. This previous derivation shows the necessity of the introduction of a cell-centered velocity as a new degree of freedom. This velocity is then determined using a cell-centered approximate Riemann solver. A more accurate scheme is built using a piecewise linear limited reconstruction of the velocity field. This section also presents the second-order time discretization. In the fifth section, numerical results are provided. First, sanity test cases are shown to assess the validity of this approach when problems with 1D or 2D symmetries are run in 3D. In a second series of tests, genuinely 3D demanding test cases are proposed. The numerical results provided by our scheme are compared with exact solutions or solutions obtained by a classical staggered Lagrangian scheme with edge artificial viscosity possibly supplemented with several useful tools (symmetry corrections [11] or vorticity damping artificial viscosity [22]). Conclusion and perspectives are finally drawn.
2. GOVERNING EQUATIONS AND NOTATION

2.1. Governing equations
In a Lagrangian framework, the 3D gas dynamics equations write

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

$$\rho \frac{d}{dt} \mathbf{U} + \nabla P = 0, \quad (2)$$

$$\rho \frac{d}{dt} E + \nabla \cdot (P \mathbf{U}) = 0, \quad (3)$$

where $\rho$ is the density, $\mathbf{U}$ the velocity, $E$ the specific total energy, and $\frac{d}{dt}$ the material derivative. Equation (1) expresses the volume conservation, and (2) and (3) are the momentum and total energy conservation laws. Volume conservation equation is referred to as the GCL. A thermodynamic closure (equation of state (EOS)) supplements the previous system by relation $P = P(\rho, \varepsilon)$. The specific internal energy is $\varepsilon = E - \frac{\rho U^2}{2}$. For smooth solutions, energy equation is reformulated into

$$\rho \frac{d}{dt} \varepsilon + P \nabla \cdot \mathbf{U} = 0, \quad (4)$$

and using (1), one obtains

$$\rho \frac{d}{dt} \varepsilon + P \rho \frac{d}{dt} \left( \frac{1}{\rho} \right) = 0. \quad (5)$$

The Gibbs relation for temperature $T$ and specific entropy $S$ writes $T dS = d\varepsilon + P d\left( \frac{1}{\rho} \right)$, and the second law of thermodynamics, namely, $T \frac{dS}{dt} \geq 0$, implies that for non-smooth flows, we have

$$\rho \frac{d}{dt} \varepsilon + P \rho \frac{d}{dt} \left( \frac{1}{\rho} \right) \geq 0. \quad (6)$$

The previous system (1–3) is replaced in this work by the non-conservative system composed of Equations (1), (2), and (4). The last equations are the trajectory equations

$$\frac{d\mathbf{X}}{dt} = \mathbf{U}(\mathbf{X}(t), t), \quad \mathbf{X}(0) = \mathbf{x}, \quad (7)$$

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

2.2. Notation and geometrical consideration
An unstructured grid consisting of a collection of non-overlapping cells is considered. Each cell is assigned a unique index $c$ and is denoted as $\Omega_c$. Each point/vertex of the mesh is assigned a unique index $p$, and $\mathcal{C}(p)$ is the set of cells sharing a particular vertex $p$. Each cell is subdivided into subcells, each being uniquely defined by a pair of indices $c$ and $p$ and denoted as $\Omega_{cp}$. This subcell is the volume defined by connecting the center of $\Omega_c$ to the mid-points of cell edges impinging at point $p$ and the centers of the faces meeting at point $p$ (Figure 2). In other words, $\Omega_{cp} = \Omega_p \cap \Omega_c$. The union of subcells $\Omega_{cp}$ sharing a particular point $p$ defines the dual vertex-centered cell $\Omega_p$ related to point $p$ as $\Omega_p = \bigcup_{c \in \mathcal{C}(p)} \Omega_{cp}$.

The primary grid is $\bigcup_c \Omega_c$ and the dual grid $\bigcup_p \Omega_p$. The set of faces $f$ of a cell is denoted as $\mathcal{F}(c)$, and the set of faces of a subcell is $\mathcal{F}(cp)$. A face $f$ is uniquely defined by a set of points. Note that a face may not be coplanar if more than three vertexes compose it. Consequently, a face
having more than three vertexes is split into triangles. The surface $A_f$ of face $f$ is computed as the sum of the surfaces of the triangles composing it. Moreover, $N_f$, the unit normal of face $f$, is deduced from the sum of normals of the triangles composing the face.

The method used to calculate volumes and surface vectors is to tetrahedralize the polyhedron of arbitrary order so that the volume of any domain can be formed in a general manner on an unstructured grid (Figure 1). The cell center point is defined as the average of the coordinates of the points that define the cell. A face center point is defined as the average of the coordinates of the points that define it.

For the sake of simplicity, we only describe the framework on a hexahedral mesh knowing that the extension to cells with more faces is immediate. In the case of a hexahedral cell $c$, the face composed of four points denoted in Figure 1 by $p_1, p_2, p_7, p_8$ has its center at $X_{fc} = \frac{1}{4} \sum_{k=1,2,7,8} X_{pk}$. The cell center is defined by $X_c = \frac{1}{9} \sum_{k=1}^{24} X_{pk}$. Moreover, the zone is divided into 24 tetrahedra, with two tetrahedra associated with each of the 12 edges. Three tetrahedra are shown in Figure 1. The 24 tetrahedra volumes are further summed to obtain an approximation of the cell volume that is called $V_c$. Subcell volume is computed by summing the volumes of the six tetrahedra impinging at point $p$ and dividing the sum by factor 2. Let us emphasize that the face vectors satisfy the fundamental geometric identities

$$
\sum_{f \in \mathcal{F}(c)} A_{f,c} N_{f,c} = 0,
\sum_{f \in \mathcal{F}(cp)} A_{f,cp} N_{f,cp} = 0,
$$

where $A_{f,c}, N_{f,c}$ are the surface and the unit outward normal of face $f$ in cell $c$. Similarly, $A_{f,cp}, N_{f,cp}$ are the surface and unit outward normal of face $f$ in subcell $\Omega_{cp}$. Note that the faces of the subcell may not be coplanar either. Consequently, each subcell face is split into two triangles by the diagonal line passing through the cell center for internal faces (arrows on Figure 1) or the point for the external faces. The surface of the subcell face called $A_{f,cp}^{\text{int/ext}}$ is therefore the sum of the surfaces of these two triangles. Similarly, $N_{cp}^{\text{int/ext}}$, the unit normal vector to the subcell face, is the sum of the normals to the triangles divided by its norm. Equation (8) is equivalent to the well-known result that the sum of the outward normals to a closed polyhedral surface is equal to zero.

The sets of internal and external (cell-boundary) faces with respect to cell $c$ are respectively named $\mathcal{F}_{\text{int}}(cp), \mathcal{F}_{\text{ext}}(cp)$, as depicted in Figure 2; and according to the second equation in (8), one has\(^1\)

$$
\sum_{f \in \mathcal{F}_{\text{ext}}(cp)} A_{f,cp}^{\text{ext}} N_{f,cp}^{\text{ext}} = - \sum_{f \in \mathcal{F}_{\text{int}}(cp)} A_{f,cp}^{\text{int}} N_{f,cp}^{\text{int}}.
$$

---

\(^1\)Another definition of interior faces is $\partial \Omega_p \cap \Omega_c$ and $\partial \Omega_c \cap \Omega_p$ for exterior faces.
Figure 2. Fragment of an unstructured mesh. Position and velocity are defined at grid points (●), whereas thermodynamic variables are located at cell centers. A cell, $\Omega_c$, is subdivided into hexahedral subcells $\Omega_{cp}$. Points are denoted by subscript $p$ and cells by subscript $c$. Each internal face of subcell $\Omega_{cp}$ (brown, blue, and green) has a superscript ‘int’. Similarly, one uses ‘ext’ for external faces. A unit normal to face $f$ denoted as $N_{f,cp}$ is the sum of the normals to the triangles divided by its norm. The surface of face $f$ is $A_{f,cp}$.

The previous equation defines the important notion of 3D corner vector as being

$$A_{cp}N_{cp} = \sum_{f \in \mathcal{F}_{ext}(cp)} A_{f,cp}^\text{ext} N_{f,cp}^\text{ext},$$

where $N_{cp}$ is the unit vector in the direction of vector $\sum_{f \in \mathcal{F}_{ext}(cp)} A_{f,cp}^\text{ext} N_{f,cp}^\text{ext}$, and $A_{cp}$ is its norm.

We use a staggered placement of variables: Position and velocity are defined at grid points whereas thermodynamic variables are located at cell centers. In a moving framework, the volumes of the primary and dual cells are functions of time $t$. Following [4], we assume that subcells are Lagrangian entities. Consequently, subcell mass $m_{cp}$ is constant in time. Being given $V_c^0$ as the volume of cell $\Omega_c$ at time $t = 0$ and the initial density $\rho_c^0$, one defines the initial subcell mass as $m_{cp} = \rho_c^0 V_{cp}^0$. By summation of Lagrangian subcell masses, one defines cell/point masses as

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

where $\mathcal{P}(c)$ is the set of all points of cell $c$. These cell/point masses inherit the Lagrangian property of subcell masses, namely, they are constant in time.

3. COMPATIBLE DISCRETIZATION

Staggered Lagrangian schemes are constructed here using the methodology of compatible discretization [2, 4, 23]. The cornerstone of this discretization is the so-called subcell force acting from subcell $\Omega_{cp}$ onto point $p$. In this compatible approach, the discretization of the internal energy equation is deduced from total energy conservation. Closely following [1], we derive an abstract form of the subcell force in 3D so that an entropy inequality is satisfied. The subcell force consists of a classical pressure term plus a tensorial viscous contribution, which is proportional to the difference between the vertex-centered and cell-centered velocities. The cell-centered velocity is a
new degree of freedom that will be solved using the principle of the Galilean invariance. To satisfy the second law of thermodynamics, the local subcell matrix involved in the viscous part of the subcell force must be positive definite. This matrix is the fundamental object that entirely defines the artificial viscosity of the scheme required to stabilize the scheme.

3.1. Geometric conservation law

We build a discretization of the volume equation (1), which is compatible with the GCL. In other words, we are using a discrete divergence operator for the volume equation by requiring consistency of the divergence of the velocity field with the time rate of change of volume of the cell (refer to [24]). By noticing that \( m_c = \rho_c V_c \) is constant in time but \( \rho_c = \rho_c(t) \) and \( V_c = V_c(t) \), we deduce

\[
m_c \frac{d}{dt} \left( \frac{1}{\rho_c} \right) = \frac{d}{dt} V_c.
\]

Moreover, because the cell volume can be expressed as a function of the position vectors of its vertexes (see [1]), we can write the time evolution of the cell volume as

\[
\frac{d}{dt} V_c - \sum_{p \in \mathcal{P}(c)} A_{cp} N_{cp} \cdot U_p = 0,
\]

where \( A_{cp} N_{cp} \) is the corner vector defined in (10), and \( U_p \) is the node velocity.

A compatible definition of the discrete divergence operator over cell \( c \) is given by

\[
(\nabla \cdot U)_c = \frac{1}{V_c} \sum_{p \in \mathcal{P}(c)} A_{cp} N_{cp} \cdot U_p.
\]  

This kind of formalism has already been used in staggered and cell-centered (free Lagrange) discretizations [24, 25]. Note that (12) is compatible with the discrete version of the trajectory equation (7), namely,

\[
\frac{d}{dt} X_p = U_p, \quad X_p(0) = x_p.
\]

Finally, a compatible discretization of the volume equation (1) writes

\[
m_c \frac{d}{dt} \left( \frac{1}{\rho_c} \right) - \sum_{p \in \mathcal{P}(c)} A_{cp} N_{cp} \cdot U_p = 0.
\]  

3.2. Momentum equation

The semi-discrete momentum equation over the dual cell \( \Omega_p \) is postulated to be

\[
m_p \frac{d}{dt} U_p + \sum_{c \in \mathcal{C}(p)} F_{cp} = 0.
\]  

Here \( F_{cp} \) is the subcell force from cell \( c \) that acts on node \( p \), which is defined by

\[
F_{cp} = \int_{\partial \Omega_p(t) \cap \Omega_c(t)} PN \, dl.
\]  

Momentum equation (15) can be seen as the Newton law applied to a particle of mass \( m_p \) moving with velocity \( U_p \).
3.3. Specific internal energy equation

The compatible staggered Lagrangian machinery relies on the conservation of total energy to obtain a semi-discrete internal energy equation following the approach described in [4]. The total energy is then defined as \( E(t) = K(t) + \mathcal{E}(t) \), where total kinetic energy and total internal energy are given by

\[
K(t) = \sum_p \frac{1}{2} m_p U_p^2(t), \quad \mathcal{E}(t) = \sum_c m_c \epsilon_c(t),
\]

and \( \epsilon_c \) is the cell-averaged specific internal energy. The conservation of total energy neglecting boundary conditions writes \( \frac{d}{dt} E = \frac{d}{dt} K + \frac{d}{dt} \mathcal{E} = 0 \). The substitution of kinetic and internal energies yields

\[
\frac{d}{dt} E = \frac{d}{dt} \mathcal{E} + \sum_c m_c \frac{d}{dt} \epsilon_c + \sum_p m_p \frac{d}{dt} U_p \cdot U_p.
\]

Using the semi-discrete momentum equation (15) yields

\[
\sum_c m_c \frac{d}{dt} \epsilon_c - \sum_{p \in \mathcal{C}(p)} \sum_{c \in \mathcal{P}(c)} F_{cp} \cdot U_p = 0,
\]

which rewrites as

\[
\sum_c \left( m_c \frac{d}{dt} \epsilon_c - \sum_{p \in \mathcal{P}(c)} F_{cp} \cdot U_p \right) = 0. \tag{17}
\]

A sufficient condition is obtained by requiring the previous equation to hold in each cell \( c \)

\[
m_c \frac{d}{dt} \epsilon_c - \sum_{p \in \mathcal{P}(c)} F_{cp} \cdot U_p = 0. \tag{18}
\]

To summarize, the semi-discrete equations for the primary variables \( \left( \frac{1}{\rho_c}, U_p, \epsilon_c \right) \) are

\[
m_c \frac{d}{dt} \left( \frac{1}{\rho_c} \right) - \sum_{p \in \mathcal{P}(c)} A_{cp} N_{cp} \cdot U_p = 0,
\]

\[
m_p \frac{d}{dt} U_p + \sum_{c \in \mathcal{C}(p)} F_{cp} = 0,
\]

\[
m_c \frac{d}{dt} \epsilon_c - \sum_{p \in \mathcal{P}(c)} F_{cp} \cdot U_p = 0.
\]

The mesh motion given by the trajectory equations

\[
\frac{d}{dt} X_p = U_p(X_p(t), t), \quad X_p(0) = x_p,
\]

is compatible with the GCL. The thermodynamic closure is provided by the EOS, which writes \( P_c = P(\rho_c, \epsilon_c) \). This subcell-based compatible semi-discretization ensures total energy conservation regardless of the subcell force form.

4. SUBCELL FORCE

Here we follow the definition of the subcell force from [1], where the Galilean invariance and thermodynamic consistency are invoked. The Galilean invariance is a principle of relativity,
which states that the fundamental laws of physics are the same in all inertial frames. To fulfill the Galilean invariance, the specific internal energy equation (18) must remain unchanged under a uniform translation denoted as $C$. Equation (18) becomes

$$m_c \frac{d}{dt} \varepsilon_c - \sum_{p \in P(c)} F_{cp} \cdot (U_p + C) = 0.$$ 

Substitution of (18) into this last equation leads to

$$\sum_{p \in P(c)} F_{cp} \cdot C = 0,$$

which must hold for all vectors $C$. Therefore, specific internal energy equation remains invariant under uniform translation if and only if

$$\sum_{p \in P(c)} F_{cp} = 0. \quad (19)$$

This condition also implies total momentum conservation without taking into account boundary conditions§.

In [1], the authors derived a possible form of the subcell force as a sufficient condition to fulfill the second principle of thermodynamics. We refer the reader to [1] for the derivation and only recall the final subcell form

$$F_{cp} = -A_{cp} P_c N_{cp} + M_{cp}(U_p - U_c). \quad (20)$$

In 3D, $M_{cp}$ is a subcell-based $3 \times 3$ positive semi-definite matrix. Such form provides the following entropy inequality satisfied by the semi-discrete scheme [1]:

$$m_c T_c \frac{d}{dt} S_c = \sum_{p \in P(c)} M_{cp} (U_p - U_c) \cdot (U_p - U_c) \geq 0, \quad (21)$$

where $S_c$ and $T_c$ are the specific entropy and temperature of cell $c$. Indeed, using the Gibbs formula, one obtains

$$m_c T_c \frac{d}{dt} S_c = m_c \left[ \frac{d}{dt} \varepsilon_c + P_c \frac{d}{dt} \left( \frac{1}{\rho_c} \right) \right]. \quad (22)$$

Substituting into (22) the specific internal energy equation (18) and the volume equation (14) yields

$$m_c T_c \frac{d}{dt} S_c = \sum_{p \in P(c)} F_{cp} \cdot (U_p - U_c) + P_c \left( \sum_{p \in P(c)} A_{cp} N_{cp} \cdot U_p \right)$$

$$= \sum_{p \in P(c)} (F_{cp} + A_{cp} P_c N_{cp}) \cdot (U_p - U_c)$$

$$= \sum_{p \in P(c)} M_{cp} (U_p - U_c) \cdot (U_p - U_c).$$

Here we have used the geometric identity $\sum_{p \in P(c)} A_{cp} N_{cp} = 0$ to obtain the second line, and we have substituted the subcell force form (20) to obtain the third and final equation. As $M_{cp}$ is a positive semi-definite matrix, (21) is fulfilled.

---

§ Summing (15) over all points $p$ yields $\sum_{p} m_p \frac{d}{dt} U_p + \sum_{c \in \Omega} \sum_{p \in P(c)} F_{cp} = 0$. Interchanging the sums yields $\sum_{p} m_p \frac{d}{dt} U_p + \sum_{c} \sum_{p \in P(c)} F_{cp} = 0$, which, according to (19), leads to $\sum_{p} m_p \frac{d}{dt} U_p = 0$, which means that the total momentum is conserved.
Entropy production (21) within cell \( c \) is directly governed by the subcell matrix \( M_{cp} \) and the velocity jump between the nodal and the cell-centered velocity, which still remains to be determined. In [1], the authors provided an abstract formulation, which, by analogy with the node-centered approximate Riemann solver introduced in the context of cell-centered Lagrangian discretization [13], provides a cell-centered approximate Riemann solver. This solver allows to determine one particular form of the subcell matrix \( M_{cp} \). To this end, we first compute the unknown cell center velocity by substituting the subcell force expression (20) into the Galilean invariance condition (19), leading to the following system satisfied by the cell-centered velocity \( U_c \):

\[
M_c U_c = \sum_{p \in P(c)} M_{cp} U_p,
\]

where \( M_c = \sum_{p \in P(c)} M_{cp} \) is a symmetric positive definite matrix. Once the definition of the subcell matrix \( M_{cp} \) is known, one can solve the previous system to obtain a unique expression of the cell-centered velocity.

There is no unique definition of subcell matrix neither in 2D nor in 3D. Consequently, we rely on the proposition made in [1], which we further extend in 3D. That is to say,

\[
M_{cp} = \sum_{f \in \partial \Omega_{cp}} Z_{f, cp} A_{f, cp} (N_{f, cp} \otimes N_{f, cp})
\]

is a \( 3 \times 3 \) symmetric positive definite matrix where each matrix \( (N_{f, cp} \otimes N_{f, cp}) \) is symmetric positive definite, \( A_{f, cp} \) is the surface of internal face \( f \) of subcell \( \Omega_{cp} \), and \( N_{f, cp} \) is its unit outward-pointing normal. The last ingredient of formula (24) is the swept mass flux for a given face \( f \) of subcell \( \Omega_{cp} \) that we define following Dukowicz [21] as

\[
Z_{f, cp} = \rho_c [\sigma_c + c_Q \Gamma_c \left| (U_c - U_p) \cdot N_{f, cp} \right|].
\]

Here \( \sigma_c \) is the isentropic sound speed, \( c_Q \) a user-defined parameter (set to 1 in our simulations), and \( \Gamma_c \) a material-dependent coefficient, which, for a \( \gamma \) law gas, is defined by

\[
\Gamma_c = \begin{cases} 
\frac{\gamma + 1}{2} & \text{if } (\nabla \cdot U)_{cp} < 0, \\
0 & \text{if } (\nabla \cdot U)_{cp} \geq 0.
\end{cases}
\]

Here \( (\nabla \cdot U)_{cp} = -\frac{1}{V_{cp}} A_{cp} N_{cp} \cdot (U_c - U_p) \) is the subcell related velocity divergence. In case of a rarefaction wave, we recover the acoustic approximation, whereas in case of a shock wave, we obtain the well-known two-shock approximation.

We note that \( M_c \) in (23) is symmetric positive definite, which ensures its invertibility. We remark that this system is non-linear because of the dependency of the swept mass flux on the cell-centered velocity. This non-linear system can be solved using an iterative procedure such as fixed point or Newton algorithms. In practice, few iterations are needed to reach convergence. Once the cell-centered velocity is known, the subcell force is deduced from Equation (20). The present cell-centered approximate Riemann solver can be viewed as a 3D extension of the work initiated by Christensen in 1D framework [19] and extended to 2D in [1].

5. IMPROVEMENTS

Anti-hourglass subpressure force. Subcell pressure forces have been designed in [9] in classical staggered discretization to counteract some parasitic hourglass grid motion that may otherwise appear. Following [9] or in the same fashion as in [1], subpressure force formalism is extended to 3D without further modification. If the technique from [1] is considered, then the cell pressure \( P_c \) must be replaced by the subcell pressure \( P_{cp} \) obtained from the subcell density \( \rho_{cp} \) and the
cell-based specific internal energy $\varepsilon_c$ via the EOS. In other words, the subcell force for the first-order scheme is

$$F_{cp} = -A_{cp} P_{cp} N_{cp} + M_{cp} (U_p - U_c).$$

(27)

We skip the whole description and refer the reader to [9] and [1].

Reconstruction and limitation of 3D velocity field. The previous solver utilizes piecewise constant nodal velocities defined over the subcells. In order to obtain a more accurate approximation in a monotone upstream-centered schemes for conservation laws sense, we construct a piecewise linear representation of the nodal velocity field. The extrapolated velocity at cell centers is further used in the solver.

We introduce a piecewise linear reconstruction of the velocity field over the dual grid by setting

$$U_p(X) = U_p + \nabla U_p \cdot (X - X_p),$$

(28)

where $\nabla U_p$ is the velocity tensor gradient constant over the dual cell $\Omega_p$ obtained with a least-squares approach. This approach is valid for any type of unstructured grid and preserves linear velocity field. Monotonicity is achieved thanks to a modification of the classical Barth–Jespersen slope limiter [26] later popularized in [27][ as described in [1] and further extended in 3D.

Limiting for vectors is usually applied separately to each component. However, such a procedure is frame dependent. Consequently, component limiters may not preserve symmetry because a rotation of the coordinate axes may produce different results. This is critical in Lagrangian hydrodynamics especially in 3D because we are dealing with moving mesh discretizations, which are particularly sensitive to symmetry discrepancy. The limiter in [1] is a frame-invariant tensorial limitation for vector field. Because its original form did not rely on dimensionality argument, its 3D extension is immediate. More precisely in 2D, this limiter determines two subcell-based orthonormal directions, which are the ones given by the node velocity and its associated perpendicular direction. In 3D, we define the local basis $(\xi^p, \eta^p, \zeta^p)$ according to the direction of the flow at and around node $p$. There is no unique way of defining these vectors. For example, we define the orthonormal basis as shown in Figure 3: The node velocity is chosen as the main direction, and the first basis vector is therefore $\xi^p = \frac{U_p}{\|U_p\|}$. One possible choice is to define $\eta^p$ as the second most representative direction of the flow, which is the average flow in the surrounding ($\bar{U}_p$ in Figure 3) further projected onto the plane $\mathcal{P}_p$ perpendicular to $\xi^p$. Now $\eta^p$ is perpendicular to $\xi^p$. The basis vectors

---

*We use the ‘Barth–Jespersen’ name, which is usual in the community, although such limiter was first introduced in [26] by Desideri and Dervieux.*
\( \eta_p \) and \( \zeta_p \) form the orthonormal basis of plane \( \mathcal{P}_p \). In the case of a uniform flow, we define \( \eta_p \) and \( \zeta_p \) as any two mutually perpendicular vectors in plane \( \mathcal{P}_p \).

The transformation matrix from the canonical basis to the local basis \((\xi^p, \eta^p, \zeta^p)\) is denoted as \( A_p \). Consequently, the transformation of the nodal velocity \( U_p \) to the local coordinates is given by

\[
W_p = \begin{pmatrix} W_p^\xi \\ W_p^\eta \\ W_p^\zeta \end{pmatrix} = A_p U_p = \begin{pmatrix} \xi^p \cdot U_p \\ \eta^p \cdot U_p \\ \zeta^p \cdot U_p \end{pmatrix}.
\] (29)

Then we find the minimum and maximum values from projections of neighboring nodes’ velocities into these new directions:

\[
W_{p,\text{min}}^\xi = \min_{k \in \mathcal{N}(p)} (\xi^p \cdot U_k), \quad W_{p,\text{min}}^\eta = \min_{k \in \mathcal{N}(p)} (\eta^p \cdot U_k), \quad W_{p,\text{min}}^\zeta = \min_{k \in \mathcal{N}(p)} (\zeta^p \cdot U_k),
\]

\[
W_{p,\text{max}}^\xi = \max_{k \in \mathcal{N}(p)} (\xi^p \cdot U_k), \quad W_{p,\text{max}}^\eta = \max_{k \in \mathcal{N}(p)} (\eta^p \cdot U_k), \quad W_{p,\text{max}}^\zeta = \max_{k \in \mathcal{N}(p)} (\zeta^p \cdot U_k),
\] (30)

where \( \mathcal{N}(p) \) is the set of neighbor points of current point \( p \). Now consider cell \( c \in \mathcal{C}(p) \) centered at \( X_c \). Using the unlimited piecewise linear representation of the velocity field, the extrapolated values of the velocity at point \( X_c \) are given by

\[
U_{p,c} = U_p(X_c) = U_p + \nabla U_p \cdot (X_c - X_p),
\] (32)

and their transformation into the local basis \((\xi^p, \eta^p, \zeta^p)\) is

\[
W_{p,c} = \begin{pmatrix} W_{p,c}^\xi \\ W_{p,c}^\eta \\ W_{p,c}^\zeta \end{pmatrix} = A_p U_{p,c}.
\] (33)

From these values, we define \( \phi^\kappa \) stands for \( \xi, \eta, \) or \( \zeta \)

\[
\phi^\kappa_{p,c} = \begin{cases} \mathcal{L} \left( \frac{W_{p,c}^{\kappa_{\text{max}}} - W_p^\kappa}{W_p^\kappa - W_{p,c}^\kappa} \right) & \text{if } (W_{p,c}^\kappa - W_p^\kappa) > 0 \\ \mathcal{L} \left( \frac{W_p^\kappa - W_{p,c}^{\kappa_{\text{min}}} - W_{p,c}^\kappa}{W_p^\kappa - W_{p,c}^\kappa} \right) & \text{if } (W_{p,c}^\kappa - W_p^\kappa) < 0 \\ 1 & \text{if } (W_{p,c}^\kappa - W_p^\kappa) = 0 \end{cases}
\]

where \( \mathcal{L}(\alpha) \) is a limiting functional such as \( \mathcal{L}(\alpha) = \min(\alpha, 1) \) or \( \mathcal{L}(\alpha) = \frac{\alpha^2 + 2\alpha}{\alpha^2 + \alpha + 2} \). The slope limiters for node \( p \) are finally defined by \( \phi^\xi_p, \phi^\eta_p, \phi^\zeta_p \) as

\[
\phi^\xi_p = \min_{c \in \mathcal{C}(p)} \phi^\xi_{p,c}, \quad \phi^\eta_p = \min_{c \in \mathcal{C}(p)} \phi^\eta_{p,c}, \quad \phi^\zeta_p = \min_{c \in \mathcal{C}(p)} \phi^\zeta_{p,c}.
\]

This triplet of limiters is transformed back into the Cartesian coordinates with

\[
\Phi_p = A_p^{-1} \begin{pmatrix} \phi^\xi_p & 0 & 0 \\ 0 & \phi^\eta_p & 0 \\ 0 & 0 & \phi^\zeta_p \end{pmatrix} A_p^{-1}.
\] (34)
The limited tensor gradient is finally given by formula

$$\nabla U_p^{lim} = \Phi_p \nabla U_p,$$

and thus the limited velocity field reconstruction in the vicinity of node $p$ is

$$U_p(X) = U_p + \nabla U_p^{lim}(X - X_p). \tag{35}$$

At this point, let us remark that (34) is valid for any choice of non-coplanar vector set $(\xi^p, \eta^p, \zeta^p)$. In case we choose orthonormal basis vectors, such as those described earlier and shown in Figure 3, the transformation matrix $A_p$ is orthogonal, meaning that $A_p^{-1} = A_p^T$, and thus, we do not need to solve any linear system to limit the tensor gradient of velocity.

Finally, the extension of the cell-centered approximate Riemann solver consists in replacing the point velocity by its extrapolated value at cell center by using the piecewise linear monotonic reconstruction as

$$M_c U_c = \sum_{p \in N(c)} M_{cp} U_p(X_c),$$

where the swept mass fluxes are

$$Z_{f,cp} = \rho c [\sigma_c + c Q \Gamma_c |(U_c - U_p(X_c)) \cdot N_{f,cp}|].$$

The subcell force is modified accordingly to

$$F_{cp} = -A_{cp} P_{c} N_{cp} + M_{cp} (U_p(X_c) - U_c). \tag{36}$$

Time discretization. The time discretization is performed with a predictor-corrector scheme to gain second-order accuracy in time. Being given the geometric quantities and physical variables at time $t^n$, one predicts the pressures that are later used in the corrector step to update physical and geometric variables. In the corrector step, internal energy is discretized using the time-centered nodal velocity $U_p^{n+1/2}$. This choice is required to ensure total energy conservation up to machine precision. We refer the reader to [1] for more details about the algorithm.

6. NUMERICAL RESULTS

The purpose of this section is to provide numerical evidence to assess the validity of the proposed approach in 3D. We will compare the results obtained by the original scheme (referred to as ‘original’) and/or the scheme using piecewise linear velocity field reconstruction (referred to as ‘with velocity reconstruction’) against analytical solutions when available. In a first series of sanity checks, we will prove that the implementation of the 3D scheme reproduces 1D or 2D known results; these sanity tests are the 1D Sod problem and the 2D Sedov problem run on 3D hexahedral grids. In a second series of tests, we will run the 3D Sedov, 3D Noh, and 3D Saltzman problems, the exact solutions of which are known.

6.1. Sanity checks

1D Sod problem We run the 1D Sod problem with the 3D code. The Sod problem is a 1D Riemann shock tube whose solution consists of a left-moving rarefaction fan, a right-moving contact discontinuity, and a right-moving shock wave. The domain is filled with an ideal gas at rest with $\gamma = 1.4$. The discontinuity is located at $x = 0.5$ at $t = 0$. The density/pressure values on the left side of the discontinuity are $1.0/1.0$. Those on the right side are $0.125/0.1$. The final time is $t_{final} = 0.2$. In our numerical experiments, the computational domain is $\Omega = [0:1] \times [0:0.03] \times [0:0.03]$ paved with a regular hexahedral mesh. We are enforcing zero normal velocity as boundary conditions for $N_x = 100$ and $N_y = N_z = 3$. 

Copyright © 2012 John Wiley & Sons, Ltd.  
DOI: 10.1002/fld
In the 3D version of the Sedov problem, the total energy of the explosion is concentrated at the origin and has magnitude $E_{\text{total}} = 0.244816$. An ideal gas with $\gamma = 1.4$ initially at rest is considered. The initial density is equal to 1.0. At final time $t_{\text{final}} = 1.0$, the exact solution is a cylindrically symmetric diverging shock whose front radius is $r = \sqrt{x^2 + y^2 + z^2} = 1$. It has a density peak $\rho = 6.0$. In the case of a hexahedral mesh in $(x, y, z)$ coordinate system on domain $\Omega = [0.0, 1.2] \times [0.0, 0.12] \times [0.0, 1.2]$, the total energy $E_{\text{total}}$ is concentrated in cells such that $x_c = z_c = 0$ (see the red cells on the top-left panel of Figure 5).

The mesh is made of $30 \times 3 \times 30$ hexahedra in $x$, $y$, $z$ directions, respectively. The results displayed in Figure 5 show the density color map and the mesh at final time on a 3D view (top) obtained by the second-order scheme. The bottom-left panel shows the mesh and the density on the 2D $x$–$z$ plane at $y = 0$, and the bottom-right panel displays the 1D cut at $z = x$ in green and 1D cuts at $z = 3x$ and $z = x/3$ from the 2D $x$–$z$ plane at $y = 0$. The results proposed by the 3D version of the method are very similar to the 2D results obtained in [1, Figure 4(a, b)]. This proves that the 3D code can reproduce 2D results without alteration.

### 6.2. 3D Sedov problem

In the 3D version of the Sedov problem, the total energy of the explosion is concentrated at the origin and has magnitude $E_{\text{total}} = 0.851072$ for $\gamma = 1.4$ similarly to [29]. At final time $t_{\text{final}} = 1.0$, the exact solution is a spherically symmetric diverging shock whose front is at radius $r = \sqrt{x^2 + y^2 + z^2} = 1$ and has a density peak $\rho = 6.0$. 

---

Figure 4. Sod problem at $t_{\text{final}} = 0.2$ for $N_x$ cells in $x$ direction, and 3 in $y$ and $z$ directions. Cell-centered density as a function of $x$ for all cells versus exact solution (black thick line). Top line: original scheme results; bottom line: scheme with velocity reconstruction results. Left panels: $N_x = 100$ results; middle panels: comparison of $N_x = 100, 200, 400$ results; right panels: enlarged image of the shock wave for $N_x = 50, 100, 200, 400$, and 800.
Figure 5. Sedov problem at $t_{\text{final}} = 1$. Scheme with velocity reconstruction. Top: density map and mesh on a $30 \times 3 \times 30$ Cartesian grid at initial and final time (the red highlighted cells are the ones with initial high energy). Bottom-left: 2D $x-z$ plane at $y = 0$, density map and mesh. Bottom-right: density as a function of cell radius (all cells are displayed).

In the case of a hexahedral mesh in $(x, y, z)$ coordinate system on domain $\Omega = [-1.2, 1.2]^3$, the total energy $E_{\text{total}}$ is concentrated in cells $c$ in contact with the origin (that is, containing the vertex $O = (0, 0, 0)$). The specific internal energy of these cells is defined as $\varepsilon_c = E_{\text{total}}/(8 V_c)$, where $V_c$ is the volume of one cell (initially the same for all cells). Therefore, the initial pressure for this cell is $p = (\gamma - 1)\rho\varepsilon = 0.4 \frac{E_{\text{total}}}{8 V_c}$.

Figure 6 displays the color map of density and mesh at final time $t = 1.0$ when a $20 \times 20 \times 20$ hexahedral mesh is used on one-eighth of the cube. The results of the scheme with velocity reconstruction have been mirrored two times in order to see the cylindrical symmetry of the shock wave in this figure. The 3D Sedov problem on the full cube has also been simulated, and the symmetry is also perfectly reproduced (not displayed in this paper). In the next set of results (Figure 7), we ran the same problem for the original scheme, the scheme with velocity reconstruction, and a classical staggered compatible Lagrangian scheme [11] with anti-hourglass subpressure forces [9] (merit factor 1) with edge artificial viscosity [5]. The mesh is a $20 \times 20 \times 20$ hexahedral mesh. The density as a function of cell radius is plotted for these three runs versus the exact solution. We can observe the lack of cylindrical symmetry on the results for the original method with edge artificial viscosity, and this can be seen especially along the axes, whereas the two schemes proposed in this paper almost exactly reproduce the cylindrical symmetry of the shock wave. As expected, the
scheme with velocity reconstruction produces more accurate results. The panels of Figure 8 present the density as a function of radius for all cells in the domain at final time versus the exact solution for $10 \times 10 \times 10$ (column (a)), $20 \times 20 \times 20$ (column (b)), and $40 \times 40 \times 40$ (column (c)) meshes. The top line presents the original scheme whereas the bottom line presents the scheme with velocity reconstruction. These plots show that the 3D code results converge to the exact solution when the grid is refined.

In Figure 9, we compare the results without anti-hourglass force obtained by the classical compatible staggered scheme with edge viscosity and the schemes without or with velocity reconstruction (see Section 5). The mesh is made of $20 \times 20 \times 20$ hexahedra. We present the 2D view of the $y-z$ plane of the density and mesh at final time $t = 1.0$ for the classical edge viscosity at the top-left corner, original scheme at the top-right corner, and the scheme with velocity reconstruction at the bottom-left corner. For comparison purposes, we also present the scheme with velocity reconstruction supplemented by the anti-hourglass force (merit factor 1) on the bottom-right corner. For all calculations without anti-hourglass force, the mesh is tangled. For the schemes presented in this paper, the central cell and the first layer of neighbor cells are tangled, whereas inappropriate jets are observed along axes when classical edge viscosity is used. In fact, these jets were already visible in Figure 7 but with a little amplitude as the anti-hourglass force tends to damp them. Because our implementation only relies on cell volume, all these calculations are completed without code failure, neither negative cell energy nor negative density is encountered. The density as a function of cell radius is also compared with the exact solution in Figure 7(a–c). Notice that this first cell tangling effect is not cured by mesh refinement, and some kind of anti-hourglass force or stiff artificial viscous force must be employed. In our implementation, the smallest merit factor that ensures that the mesh remains valid is 0.06.

### 6.3 3D Saltzman problem

This is the extended version of the Saltzman piston as previously defined in [11] and rephrased in [22]. Let us note that the $100 \times 10 \times 10$ mesh is completely 3D in its setup because the Saltzman skewing of the grid is made to change parity uniformly in the third dimension (Figure 10). The equations to obtain the skewed grid are recalled in [17]. In 2D or 3D, this problem is used to test...
the robustness of Lagrangian schemes. For this problem, we use the polytropic index $\gamma = 5/3$. The initial state is $\rho_0 = 1$, $\epsilon_0 = 10^{-6}$. The plane originally at $x = 0$ is driven by a piston with a unit normal velocity. We set reflective boundary conditions elsewhere. The exact solution is a $y-z$ planar shock wave moving at speed $4/3$ in the $x$ direction. The final time is $t = 0.6$. In Figure 10, we present the results obtained by the original scheme and the scheme with velocity reconstruction. More precisely, the density map and mesh are displayed in panels (a) and (b). Moreover, the density
Figure 8. 3D Sedov problem. Density as a function of radius for all cells in the domain at final time $t = 1.0$. Top line: original scheme. Bottom line: scheme with velocity reconstruction. Column (a): $10 \times 10 \times 10$ mesh. Column (b): $20 \times 20 \times 20$ mesh. Column (c): $40 \times 40 \times 40$ mesh. We show the exact solution by a red thick line.

Figure 9. 3D Sedov problem. 2D view in the $y-z$ plane of the density and mesh at final time $t = 1.0$ on a hexahedral $20 \times 20 \times 20$ mesh without anti-hourglass forces. Top panel: classical edge viscosity at the top-left corner, original scheme at the top-right corner, scheme with velocity reconstruction at the bottom-left corner, and scheme with velocity reconstruction supplemented with anti-hourglass force (merit factor 1) at the bottom-right corners. Bottom panels: density as a function of radius for all cells in the domain for classical edge viscosity (a), original schemes (b), and scheme with velocity reconstruction (c). See Figure 8 bottom-line panel (b) for the results of the scheme with velocity reconstruction supplemented with anti-hourglass force.
as a function of radius for all cells is shown in panels (c) and (d). The expected behavior for the two schemes is observed, namely, the original scheme is dissipative, hence it damps the numerical vorticity that effectively leads to a more regular mesh. On the other hand, the scheme with velocity reconstruction, being more accurate and less dissipative, does not damp enough the parasitic instabilities. This leads to a less regular numerical solution.

We remark also that some modification of the $\mathcal{M}_{c_p}$ matrix permits to improve the numerical results for instance by neglecting the off-diagonal terms. It has indeed the tendency to add more dissipation to the solution and, as a consequence, leads to a more robust scheme.

### 6.4. 3D Noh problem

In an eighth of the unit domain, a gas ($\gamma = 5/3$) is initiated with $\rho_0 = 1$, $\varepsilon_0 = 10^{-6}$, and $\mathbf{U}(x, y) = \left(\frac{-x}{\sqrt{x^2+y^2+z^2}}, \frac{-y}{\sqrt{x^2+y^2+z^2}}, \frac{-z}{\sqrt{x^2+y^2+z^2}}\right)$ (see [30]). A spherical shock wave is
generated at the origin and further propagates. The final time is chosen at \( t_{\text{final}} = 0.6 \). At this time, the exact solution is given by the position of the shock at \( r_s = 0.2 \), and the post-shock density plateau is 64.

The initial mesh is made of 30 × 30 × 30 regular hexahedral cells on the unit cube (i.e., one-eighth of the domain).

The results obtained with the classical 3D compatible staggered Lagrangian scheme with edge artificial viscosity are comparable with the ones displayed in [22, Figure 4(a)]: Jets along the axes dramatically alter the symmetry of the solution. (Such non-symmetry along the axes was already triggered in the 3D Sedov problem in Figures 7(e) and 9.) These results are already well documented; the edge artificial viscosity in 2D or 3D may produce excessive numerical vorticity that sometimes leads to a degradation of the solution. Please note that this is not a boundary condition issue as such behavior is also observed when the problem is run on the full \([-1, 1]^3\) domain. Using some vorticity damping artificial viscosity helps somehow to improve the general behavior [22]; however, this extra damping has lethal effect when physical instability is to be simulated.

In Figure 11, we present the density map and mesh for the original scheme and the scheme with velocity reconstruction. The latter scheme develops some numerical instabilities, which damage the numerical solution (see panel (b)). Let us remark that the numerical instabilities seem to develop along the lines \( \pm x = \pm y = \pm z \) contrary to the ones from classical staggered Lagrangian schemes that develop along the axes \( x, y, z \). Nevertheless, these instabilities have a lethal effect on the computation when the time grows. Moreover, these instabilities are still present when a refined grid is

![Figure 11](image1.png)

(a) (b)

Figure 11. Noh problem on a 30 × 30 × 30 hexahedral grid. Density map and mesh: (a) original scheme, (b) scheme with velocity reconstruction. The color scale is the same as that of the original scheme. The maximal density in a pinched cell along the axis is about 415.

![Figure 12](image2.png)

Figure 12. Noh problem on 30 × 30 × 30 and 60 × 60 × 60 hexahedral grids. Density map and mesh for the original scheme. Left: 30 × 30 × 30 resolution (with top-bottom mirroring). Right: 60 × 60 × 60 resolution (with top-bottom mirroring). The same color resolution is used for both simulations. The maximal density for the 60 × 60 × 60 simulation is about 100.
used. The original scheme is able to produce a correct numerical solution because its formulation leads to a more dissipative behavior. In Figure 12, we present the results of the original scheme for $30 \times 30 \times 30$ and $60 \times 60 \times 60$ hexahedral grids. The goal is to show the convergence of the numerical results towards the exact spherical shock wave whose radius is $r_s = 0.2$ at final time 0.6.

7. CONCLUSION

This work has proposed a 3D version of the previous work from [1]. It consists in a staggered discretization for Lagrangian hydrodynamics on general unstructured meshes. The framework uses fundamental objects of compatible discretizations such as Lagrangian subcell mass and subcell forces. An artificial viscosity form is formulated invoking the Galilean invariance and thermodynamic consistency as in [1]. The satisfaction of entropy inequality is ensured by using a subcell-based positive definite tensor, which is the cornerstone of the numerical schemes in 2D and 3D. Formally, the extension of the 2D scheme with piecewise constant velocity field in 3D is immediate, and only few specific points required clarification. A more accurate version of the scheme is further presented using a piecewise linear velocity field reconstruction and a vector limitation procedure that is frame independent. This limitation requires the choice of two directions compared with only one in 2D. A 3D code has been constructed on this framework in order to numerically study the behavior of such Lagrangian schemes.

The approach has been validated on sanity checks; namely, the 1D Sod problem and the 2D Sedov problems run with the 3D code. Then genuine 3D test cases have been run on hexahedral grids. The 3D Sedov problem showed that spherical symmetry of a shock wave can be preserved. Moreover, it shows the improvement obtained by the scheme with velocity reconstruction on regular grids. For more demanding test cases (3D Noh on hexahedral grid) or on less regular grids (3D Saltzman), the original scheme reveals its robustness. In fact, the extra dissipation of this scheme reduces its sensitivity to numerical instabilities, which are usually not sufficiently damped by the scheme with velocity reconstruction.

In the near future, we plan to investigate the behavior of different subcell-based matrices, which lead to different numerical schemes. We have observed that some modifications of the matrix presented in this work enhance the dissipative behavior of the resulting scheme. We plan to adapt the form of the subcell-based matrix within a cell depending for instance on a local measure of mesh regularity.

Finally, a last word on 3D implementation: A 3D code on moving mesh, no matter how much effort is dedicated to a proper implementation, is never a direct extension of its 2D counterpart.

ACKNOWLEDGEMENTS

The first author would like to acknowledge the fruitful and enlightening discussions that he shared with Ed Caramana. He also would like to acknowledge the financial support of the ANR (L’Agence Nationale de la Recherche in France) under the program JCJC ‘ALE INCubator 3D’. The last author has been partly supported by the Czech Science Foundation grant P201/12/P554 and by RVO, 68407700. The authors would like to acknowledge the support of the French embassy in Prague, Czech Republic, under the Vltava program; it provided the conditions for this work to reach maturity.

REFERENCES


Copyright © 2012 John Wiley & Sons, Ltd.  
DOI: 10.1002/fld