ACADEMIC PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computational Physics 185 (2003) 213–251

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

A level set approach to Eulerian–Lagrangian coupling

Marco Arienti, Patrick Hung, Eric Morano, Joseph E. Shepherd *

Graduate Aeronautical Laboratories, MS 205-45, California Institute of Technology, Pasadena, CA 91125, USA

Received 14 June 2002; received in revised form 10 September 2002; accepted 25 November 2002

Abstract

We present a numerical method for coupling an Eulerian compressible flow solver with a Lagrangian solver for fast transient problems involving fluid–solid interactions. Such coupling needs arise when either specific solution methods or accuracy considerations necessitate that different and disjoint subdomains be treated with different (Eulerian or Lagrangian) schemes. The algorithm we propose employs standard integration of the Eulerian solution over a Cartesian mesh. To treat the irregular boundary cells that are generated by an arbitrary boundary on a structured grid, the Eulerian computational domain is augmented by a thin layer of Cartesian ghost cells. Boundary conditions at these cells are established by enforcing conservation of mass and continuity of the stress tensor in the direction normal to the boundary. The description and the kinematic constraints of the Eulerian boundary rely on the unstructured Lagrangian mesh. The Lagrangian mesh evolves concurrently, driven by the traction boundary conditions imposed by the Eulerian counterpart. Several numerical tests designed to measure the rate of convergence and accuracy of the coupling algorithm are presented as well. General problems in one and two dimensions are considered, including a test consisting of an isotropic elastic solid and a compressible fluid in a fully coupled setting where the exact solution is available.

Keywords: Partial differential equations; Boundary value problems; Fluid–solid interaction; Shock waves and blast waves

1. Introduction

There is a variety of numerical methods today that can be used to tackle multi-material mechanics problems involving both fluid and solid motions as well as their interactions. Traditionally, fluid dynamicists have favored Eulerian methods while solid mechanicians prefer Lagrangian methods.

It is clear that a large class of applications is neither ideally suited for a pure Lagrangian nor a pure Eulerian approach. In a Lagrangian calculation, the mesh points correspond to elements of mass in the material and their trajectories follow the particle paths of the material elements. Thus, the position of a boundary is automatically calculated. Since the initial accuracy of the approximation is generally maintained throughout the computation, Lagrangian schemes have proven to be very accurate for a constant

* Corresponding author. Tel.: 1-818-895-3283; fax: 1-818-449-2677.
E-mail address: jeshep@galcit.caltech.edu (J.E. Shepherd).

0021-9991/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
doi:10.1016/S0021-9991(02)00055-4

number of mesh points as long as the approximating mesh remains regular. However, as the computation evolves, the stretching of the computational grid may drastically reduce the stable time step (which is proportional to the minimum size of an element of a triangular grid, say). Furthermore, if the mesh becomes highly distorted, the calculation becomes increasingly inaccurate. Remeshing is then required, at the price of increased complexity and computational effort. Conversely, a pure Eulerian calculation allows the development of a complex flow, at the price of a loss in accuracy when treating an arbitrarily varying, time-dependent boundary (which is intrinsically of a Lagrangian type). Examples of the limitations of a pure Eulerian or Lagrangian approach can be found in [1].

Hybrid methods such as the arbitrary Lagrangian–Eulerian (ALE) method address multi-material applications involving fluid–solid interactions. Refs. [2–5] report on recent advances in ALE coupling. In ALE methods, the conservation equations are expressed in a control volume formulation, where the control volume is bounded by a surface $S_a(t)$ moving with arbitrary local velocity $\mathbf{u}_a$. The control volume velocity $\mathbf{u}_a$ has two important limit values. In the Eulerian limit, $\mathbf{u}_a = \mathbf{0}$ and the control volumes are fixed. In the Lagrangian limit $\mathbf{u}_a = \mathbf{u}$, the local flow field velocity, and the control volumes coincide with material volumes.

In ALE methods, a Lagrangian phase is followed by a coordinate transformation due to the mesh motion (remapping or advection phase). The advection step performs an incremental rezone in which nodes are moved only a small fraction of a typical length of the surrounding elements. Monotonic advection algorithms are used to prevent the advection step from creating new minimum or maximum values for the solution variables.

In fluid–structure interactions with ALE methods, the equations of the structural elements are usually expressed using a purely Lagrangian scheme, so that the nodes follow the motion of material particles. The interaction with a fluid can be modeled through intermediate regions in which the mesh moves with a spatially varying velocity. A grid-rezoning technique is used within the bulk of the fluid domain to respect the movement of Lagrangian interfaces by simultaneously minimizing the grid distortion. The specification of $\mathbf{u}_a$ is key to the success of ALE methods. Unfortunately, this process often requires a priori knowledge of the solution when modeling the problem. For systems where the Lagrangian domain suffers large deformation or where the Eulerian flow has high rotations, ALE methods will often fail to give a solution. Finally, the ALE method does not appear to be very suitable for loosely coupling separate Eulerian and Lagrangian software packages since this introduces a third solution algorithm and increases the complexity of the coupling process rather than simplifying it.

The scheme we propose provides an alternative to ALE methods. Unlike ALE schemes, there is no mixed Eulerian–Lagrangian region and coupling occurs only through interaction at the boundary of the Eulerian and Lagrangian regions. The two subdomains are integrated separately by two independent (synchronized) Eulerian and Lagrangian solvers. We will refer to this scheme using the acronym GEL (ghost-fluid Eulerian–Lagrangian).

Since the Eulerian region mesh is usually Cartesian and stationary, the problem of how to treat moving and irregular flow domain boundaries arises. When an arbitrary boundary is embedded in a structured grid, any cell whose interior contains this boundary will be a “cut cell”. By filling in the cut cells and a small layer of neighboring regular cells with an appropriate “ghost fluid”, cell updates can be performed in the same standard way as in the bulk of the computational domain. Additionally, the time step is not constrained by the geometry of the cut cells.

The treatment of the cut cells is crucial to any Cartesian grid method, since a straightforward approach would reduce the stable time step to an arbitrarily small value due to the reduced size of the irregular cells. Also, accuracy and conservation at the boundary must be addressed, but in such a way that extension of the coupling scheme to two and three dimensions is relatively easy. Early work by Noh [1] made use of redistribution and cell-merging, probably the most popular approach in Cartesian grid methods. An extensive review of these methods can be found in [6]. Numerical results typically present only first-order accuracy at the boundary, independent of the accuracy of the flow solver in the bulk of the computational domain.

Pember et al. [6] propose an adaptive Cartesian grid method where the boundary is reconstructed with a shock tracking approach and treated as a stationary reflecting wall. The coupling scheme is an explicit two-step method, enforcing conservation in the cut cells and in the neighboring regular cells through a redistribution algorithm. Numerical results for a Prandtl–Meyer expansion show a degradation of the accuracy of the scheme to first-order at the boundary. Recently, Falcovitz et al. [7] proposed a coupling scheme that maintains conservation across the boundary cells when both the fluid and the boundary undergo uniform motion. No indication is given on how the algorithm performs in actual dynamical problems.

The method we propose has its origin (and name) in the ghost fluid method (GFM) originated by Fedkiw et al. [8–10]. GFM originated as an algorithm for handling Eulerian multi-phase multi-fluid problems where interfaces separate regions of different fluids, e.g., air and water. The original GFM is designed to capture discontinuous interfaces with an Eulerian solver on each side. Within a prescribed distance of an interface, an Eulerian grid point is a real node to one solver and a ghost node to the other. The prescription for populating a state of a ghost node in the GFM is to replace pressure and normal velocity from the real node, while extrapolating in the normal direction a second thermodynamic variable (entropy) and the tangential velocity.

GEL differs from the original formulation in the way the solution variables in the ghost region are populated. In this respect, GEL is more akin to the local mirroring extrapolation technique presented by Forrer and Berger [11] and Forrer and Jeltsch [12], since it treats the Eulerian–Lagrangian interface as an impermeable wall at any new iteration of the Eulerian solver. The main difference with respect to [11,12] lies in the way the Eulerian–Lagrangian interface is tracked; in our case, a level set-based approach [13,14] is followed. Also, GEL does not use boundary cell averaging and the order of the mirroring extrapolation is lower than in [11,12], at least for a smooth flow. While less sophisticated, GEL is expected to be more robust when dealing with arbitrarily complex boundaries and fluid–solid shock interactions, particularly in three dimensions. We note that results presented by Forrer and Berger for moving boundaries are limited to flow interactions with rigid bodies, whereas we address fully coupled problems in the sense described by Noh [1].

In Section 1, a thorough treatment of the coupling algorithm is provided. Section 2 is an overview, stating the class of problems we intend to address and outlining the coupling algorithm. As the Eulerian and Lagrangian solvers that are used in this paper are well established, their descriptions are brief by intent, with further references provided for readers desiring more information. Section 3 focuses on the decomposition of the solution domain into Eulerian and Lagrangian subdomains, which leads to the definition of the Eulerian–Lagrangian interface. The dynamical and kinematic constraints that need to be satisfied by this interface are described there. Section 4 addresses the numerical implementation of the coupling scheme. Other pertinent issues, including the temporal evolution of the subdomains, possible enhancements, and areas of future research are also discussed.

The second part of the paper presents several numerical tests designed to measure the rate of convergence and accuracy of the algorithm. Section 5 describes several one-dimensional verification tests. The behavior of the interface under shock wave transmission is examined in the “transparency tests”. Section 6 discusses several two-dimensional tests. Verification tests proposed in [11] are adopted in order to investigate issues of mass conservation. The final example we consider is a verification test consisting of a shock load over an isotropic elastic solid (superseismic loading problem). This is a fully coupled fluid–solid interaction problem where the exact solution is known.

2. Overview

The work on Eulerian–Lagrangian coupling described in this paper is a part of the research effort at Caltech to develop a virtual test facility (VTF) that provides a problem-solving environment for three-dimensional parallel simulations of the dynamic response of materials in multi-physics problems. GEL coupling is used extensively and is the workhorse behind the VTF [15]. A typical example application of the VTF is modeling the cylinder test used for high-explosive performance studies [16]. In this test, sketched in the left panel of Fig. 1, a cylinder of high explosive is detonated inside of a metal tube. The subsequent motion of the metal is used as a measure of the performance of the explosive. The motion of the metal and the progress of the detonation wave can be strongly coupled if the chemical reaction processes in the detonation wave are sufficiently slow.

The approach we follow divides the problem into a fluid mechanics and a solid mechanics portion (see Fig. 1). This division is both theoretical and computational. On the one hand, it allows research and development on the two different subject areas to progress separately and concurrently; on the other hand, this separation let us combine the strengths of Eulerian and Lagrangian solvers.

Following standard practice in gas dynamics, shock physics, and high explosives modeling, we approximate the motion in the Eulerian region as hydrodynamic in nature and neglect viscous effects. We discuss only nonreactive flow in this paper but that is not an essential limitation and the method has been applied to detonation problems. Nonreactive, inviscid fluid dynamics is described by the Euler equations [17]

$$\begin{aligned}
&\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0,\\
&\frac{\partial}{\partial t}(\rho\mathbf{u}) + \nabla\cdot(\rho\mathbf{u}\mathbf{u} + \mathbf{I}P) = 0,\\
&\frac{\partial}{\partial t}\left[\rho\left(e + \frac{\mathbf{u}^2}{2}\right)\right] + \nabla\cdot\left[\rho\mathbf{u}\left(e + \frac{P}{\rho} + \frac{\mathbf{u}^2}{2}\right)\right] = 0.
\end{aligned} \tag{1}$$

Here ρ is the density, $\mathbf{u}$ the velocity vector, P the pressure, and e is the specific internal energy. These equations may be rewritten in conservative form

$$\mathbf{U}_t + \nabla\cdot\mathbf{F}(\mathbf{U}) = 0, \tag{2}$$

by using the vector of conserved variables $\mathbf{U} = (\rho, \rho\mathbf{u}, \rho E)^{\mathrm{T}}$, the flux vector $\mathbf{F}(\mathbf{U}) = (\rho\mathbf{u}, \rho\mathbf{u}\mathbf{u} + \mathbf{I}P,\ \rho\mathbf{u}(E + P/\rho))^{\mathrm{T}}$, and the specific total energy $E = e + 1/2\|\mathbf{u}\|^2$. The equation set (1) must be supplemented by an Equation of State (EoS) that describes the equilibrium thermodynamic state of the material. For our purposes, it is sufficient to have a relationship between pressure, internal energy, and mass density, $P = P(e, \rho)$. The examples presented in the paper all use the approximation of a perfect gas with a constant ratio of specific heats γ,

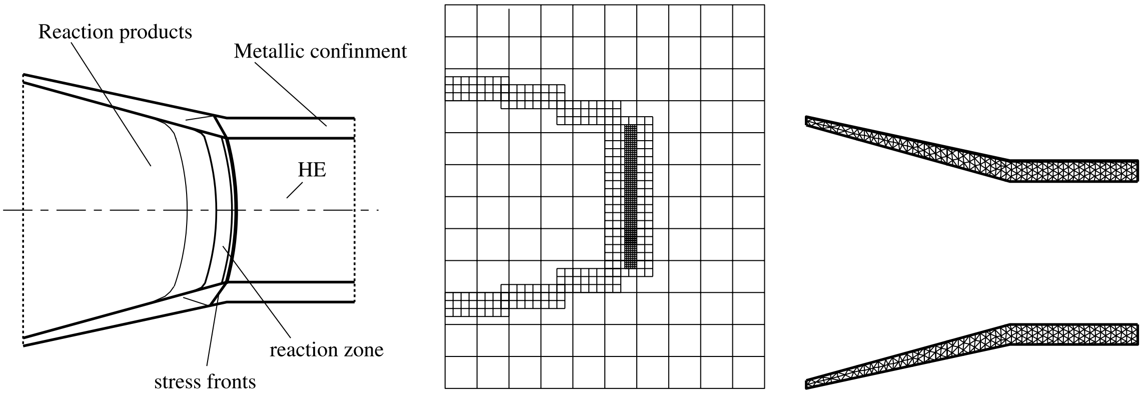

Fig. 1. Cylinder test and domain decomposition. The cylinder test problem (left) is decomposed into a Cartesian domain, here with adaptive mesh refinement, for the fluid mechanics solver (center); and a Lagrangian domain for the solid mechanics solver (right).

$$P = (\gamma - 1)\rho e. \tag{3}$$

Examples of a multi-species reactive flow with the Mie–Grüneisen EoS for the solid explosive and the Jones–Wilkins–Lee EoS for the reaction products are presented in [15].

A number of simple Lagrangian solvers were developed to verify and validate Eulerian-Lagrangian coupling. These solvers are a one- and two-dimensional integrator of motion for rigid bodies with assigned trajectory or traction boundary conditions, an Euler–Bernoulli beam module, a one-dimensional Lagrangian gas dynamics solver, and a two-dimensional explicit finite element (FE) solver. Although these Lagrangian solvers are rather simple, the methods described in this paper have also been applied [15] with a rather sophisticated [18] FE solver.

Each one of these discretizations leads to a set of ordinary differential equations in the generalized displacements $\mathbf{X}$ of the form

$$M\frac{\mathrm{d}^2\mathbf{X}}{\mathrm{d}t^2} + \mathrm{C}\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t} + K\mathbf{X} = \mathbf{f}. \tag{4}$$

We recognize in Eq. (4) the mass matrix M, the stiffness matrix K, the dissipation matrix C, and the vector $\mathbf{f}$ of generalized external forces.

In the rigid body module, $\mathbf{X}$ is the vector of coordinates of the center of mass. We have $C = 0$, $K = 0$ and $\mathbf{f}(t)$ is computed by integrating the Eulerian pressure field over the boundary. The solution is advanced in time either by a third-order Runge–Kutta or an explicit Newmark scheme. Alternatively, what is called an essential boundary condition, i.e., a specification of the trajectory of the body, can be used.

The beam module is based on the classical Euler–Bernoulli (sometimes known as Bernoulli–Euler, or Coulomb) beam theory, attributing the resistance to flexure entirely to extension and contraction of longitudinal filaments [19]. See, for example, [20] for a list of the theory's main assumptions and limitations. The derivation of the element matrices is standard and will not be reproduced here. A consistent mass matrix is used for the integration of the transient beam equation. See [21–23] for further details.

A one-dimensional Lagrangian gas dynamics module has been developed to conduct a class of verification problems referred to as transparency tests. The user can specify any combination of linear (von-Neumann) artificial viscosity [24], quadratic artificial viscosity [25,26], and artificial heat conduction [27] for solving the Euler equations with a perfect gas EoS. Nodal variables are nodal displacements and their derivatives. Cell variables are pressure, internal energy, specific volume, artificial viscosity, and heat conduction. Properties of the finite elements (or cell variables) are staggered spatially with respect to the nodal variables (in this one-dimensional setting). Cell variables and nodal variables are also staggered temporally as shown in Fig. 2. Initial conditions corresponding to $n = 0$ are displacement and velocity at $t = -0.5$ and pressure and specific volume at $t = 0$. Pressure at $t = 0$ is used to update velocity to $t = 0.5$ by the momentum equation. Velocity at $t = 0.5$ is used to update specific volume to $t = 1.0$ by the continuity equation. The energy equation and EoS are used to compute the pressure at $t = 1.0$. This completes one cycle, and n is now at 1. The finite difference numerical method follows essentially the scheme of von-Neumann and Richtmyer [24]. For a more recent treatment of artificial viscosity and a detailed description of its implementation in two and three dimensions, see [28].

The two-dimensional FE module is a displacement-based solver used for computing plane stress/plane strain problems. Linear (3 nodes) or quadratic (6 nodes) triangular plane elements, with linearized kinematics and explicit time integration of a Hookean elastic material, are implemented. The solution is advanced by evaluating the force balance at time t^n in order to compute the acceleration of the system between t^n and t^{n+1}. This is equivalent to computing the velocity and acceleration via central differences, and the method is, therefore, second-order accurate $\mathrm{O}(\Delta t^2)$ when equal time steps are used. The advantage of this scheme is low computational cost and low storage when a diagonal mass matrix is used [23].

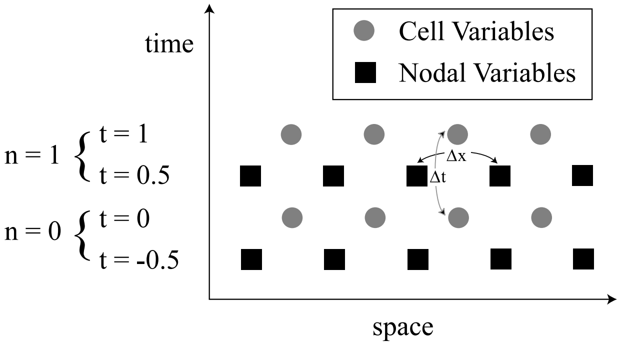

Fig. 2. Finite difference space–time discretization for one-dimensional Lagrangian solver.

In this paper, we restrict our attention to the class of problems involving the coupling of an inviscid fluid with a solid whose interface is assumed to be impermeable, nonreactive, adiabatic, and unable to support surface tension. The VTF, for which the method is designed, targets problems (e.g., the cylinder test) which fall under this category. With these assumptions, the Eulerian–Lagrangian interface (EL-interface, to be defined in the following section) can be treated as a contact discontinuity with the following properties: (1) no mass flux; (2) no jump in normal velocity; (3) free-slip boundary condition for the tangential velocity; (4) no jump in the normal stress. Jumps in entropy (or density) across the interface are admitted. These properties are enforced by the coupling scheme through the application of boundary conditions at discrete times, as will be seen in the next section.

3. Time and space discretization

In our procedure, the entire solution domain is decomposed into subdomains of two different types, Eulerian and Lagrangian. We assume that coupling exists only between subdomains that share a boundary, and only through boundary conditions. The GEL coupling scheme provides these boundary conditions to the solvers in a manner to be described next. The cases of Eulerian–Eulerian coupling (e.g., multi-fluid simulations) and Lagrangian–Lagrangian coupling (e.g., contact mechanics) will not be discussed but their importance is noted.

3.1. Spatial discretization and interface representation

Given an initial boundary value problem (IBVP) on $\Omega \times T$, the domain Ω (and the time coordinate on the interval T, treated in the next section) needs to be discretized for numerical solution. Loosely speaking, part of the domain will be covered by a Lagrangian mesh and the rest by an Eulerian mesh.

The discretization Ω_L of Ω that is associated with the Lagrangian solver is the *Lagrangian domain*. Generally, Ω_L is an unstructured grid. The discretization Ω_E that is associated with the Eulerian solver is the *Eulerian domain*. In this paper, we further specialize Ω_E to be a collection of structured grids, namely Cartesian grids. Note that, in general, $\Omega_L \subset \Omega$ and $\Omega_E \subset \Omega$. Fig. 3 is a sketch of a Lagrangian domain made of triangular elements which is partially superimposed on a Cartesian grid.

The boundary representation of Ω_L is an oriented surface denoted by $\partial\Omega_L$. We are particularly interested in the subset of $\partial\Omega_L$ whose points lie in Ω_E. We call this subset the *EL-interface* $\partial\Omega_{EL}$. All the coupling that takes place between the Eulerian domain and the Lagrangian domain is assumed to occur at this interface.

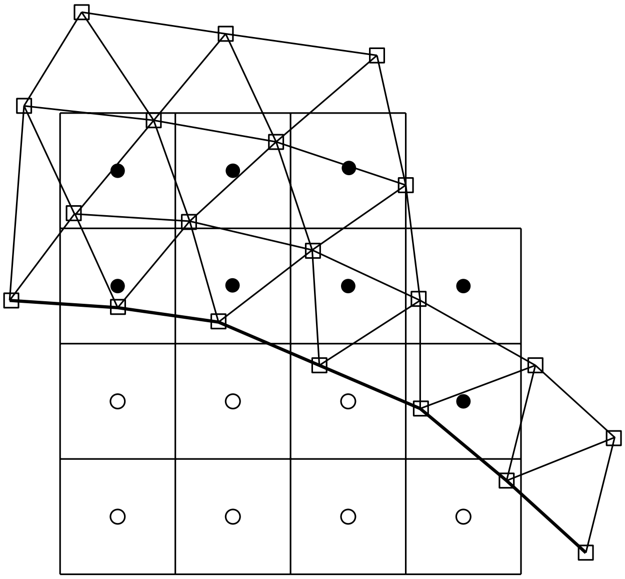

Fig. 3. Eulerian (regular grid) and Lagrangian discretizations (triangular mesh) showing overlap of domains and a layer of ghost cells (filled points). The EL-interface is indicated by the thicker line on the Lagrangian boundary.

The level set function (denoted by the scalar field φ) contains an implicit representation of $\partial\Omega_{\mathrm{EL}}$ on Ω_{E}. It is defined as the signed distance from $\partial\Omega_{\mathrm{EL}}$, evaluated at the center of a cell (in a finite volume scheme) or at a grid vertex (in a finite difference scheme). In this paper, we will refer to such a point as a *node* when the distinction between vertex and cell center is unimportant. The level set is discretized as the set of points $\varphi_{i,j} = \varphi(x_i, y_j)$, where the indices i, j span the Cartesian grid. The level set was originally applied to computations on Cartesian grids by Osher and Sethian in [29] and has been successfully employed to resolve sharp interfaces between materials with different properties or different equations of state [8].

The equation $\varphi = 0$, or the zero level of the discretized field $\varphi_{i,j}$, identifies the EL-interface $\partial\Omega_{\mathrm{EL}}$. At each time step, the level set is reconstructed on the Cartesian grid from the current Lagrangian description of the boundary. To this end, ray-intersection, a popular approach for determining whether a point lies inside or outside of a surface (a process known as point-classification), is implemented [30]. The complexity of the overall algorithm is of order $\mathcal{O}(M \cdot N)$, where M is the number of points in the Lagrangian boundary representation and N is the number of points in the Eulerian grid. Recently, an algorithm to reconstruct (in three dimensions) the closest point transform φ with optimal complexity has been developed [31].

Throughout this paper, we will use the sign convention introduced in [9],

$$\Omega_{\mathrm{R}} = \left\{\Omega_{\mathrm{E}} : \varphi_{i,j} \leqslant 0\right\}. \tag{5}$$

The set Ω_{R} designates the real (or flow) part of the Eulerian domain, where the flow field is computed, as opposed to the ghost region, where boundary conditions are set. Eq. (5) means that the level set in the real part of the Eulerian domain is negative. Since φ defines contour levels of the signed distance function, the gradient $\nabla\varphi$ at $\partial\Omega_{\mathrm{EL}}$ must be perpendicular to $\partial\Omega_{\mathrm{EL}}$ itself. A corollary of the sign convention is that the normal vector $\nabla\varphi/\|\nabla\varphi\|$ is oriented from the Eulerian to the Lagrangian domain. In practice, $\nabla\varphi/\|\nabla\varphi\|$ is numerically approximated by central differencing of $\varphi_{i,j}$.

To use regular Cartesian grid cells, the collection of cells Ω_{R} needs to be augmented so that each computational node has a complete stencil. This is done by adding to Ω_{R} a thin layer of cells $\Omega_{\mathrm{G}} \subset \Omega_{\mathrm{E}}$ at the EL-interface. We call this subset the *ghost region*. The Eulerian solver operates exclusively on

$\Omega_R \cup \Omega_G \subseteq \Omega_E$. In Fig. 3, these cells are marked by a filled circle. The ghost region Ω_G is required to be "big enough" and the required size depends on the details of the Eulerian solver. A more precise statement will be made in the next section.

3.2. Ghost cells

The concept of ghost cells as used here is an extension for arbitrary boundaries of the commonplace guard (or ghost) cells that surround a computational patch, i.e., a rectangular Cartesian grid. This practice allows for the application of boundary conditions (BC). Additional rows of guard cells are used to "complete" the stencil of the external cells of a patch so that the solver does not need to be aware of the boundary of its computational domain. The actual number of rows depends only on the stencil of the Eulerian scheme.

The introduction of a level set function allows for the extension of this idea to an arbitrary boundary. The Eulerian node (i,j) is a *ghost node* if the level set at that point satisfies

$$\Omega_G = \left\{\Omega_E : 0 < \varphi_{i,j} \leqslant \varphi_S\right\}. \tag{6}$$

The corresponding cell can be partially or fully covered by the Lagrangian discretization Ω_L.

The parameter φ_S depends exclusively on the stencil of the numerical scheme that is used to compute the fluxes. For example, a second-order accurate essentially-non-oscillatory (ENO) scheme requires four nodes, two on each side of the node that is being updated, to compute the numerical flux. The extent of the ghost region should be at least $2 \cdot \Delta x$, since we cannot expect a cell boundary to lie exactly at $\varphi = 0$. We must account for the motion of the EL-interface because a ghost cell can be "exposed" and become a real cell at the next time step. Thus, in this example, the buffer area must be increased at least to $\varphi_S = 3 \cdot \Delta x$. This one-dimensional argument can be extended to higher dimensions as long as the solver implements a dimension-by-dimension integration.

According to Eq. (6), the ghost region has to be initialized (or populated) up to a distance φ_S; the details will be discussed in the next section. Note, however, that the coupling procedure is completely independent of the patch integrator. We can think of it as setting the proper boundary conditions before advancing the solution by one time step. Indeed, given a generic patch integrator, we require only two additions to the code: (1) a test to compute the numerical flux only if $\varphi \leqslant \varphi_S$; (2) a test to update the solution only if $\varphi < 1 \cdot \Delta x$. The second point above is important because a ghost cell within one Δx from the EL-interface can become a real cell after a time step. No more than one ghost cell layer needs to be updated since the Courant–Friedrichs–Levy (CFL) condition is applied over all $\Omega_R \cup \Omega_G$ when estimating a stable time step for the fluid solver. The CFL condition prevents the contact discontinuity at the EL-interface from sweeping more than a fraction of a Cartesian cell when the solution is advanced.

3.3. Time discretization and temporal coupling

The time coordinate $T = [t_i, t_f]$ is discretized (partitioned) by $\theta = \{t^0, t^1, t^2, \ldots, t^n\}$ as follows:

$$T = \bigcup_1^N \tau^n, \tag{7}$$

where $\tau^n = (t^{n-1}, t^n]$.

It is clear that $t^0 = t_i$ (the initial time) and $t^n = t_f$ (the final time). The set θ represents instants in time when the *coupling is performed*. Note that this can be different from the temporal discretizations used for the Eulerian or the Lagrangian solvers. In fact, Eulerian solvers and Lagrangian solvers usually employ different time integrators, possibly with multiple steps.

The current implementation of GEL defines the common time increment Δt which is allowable from stability considerations for the Eulerian and Lagrangian grids, say Δt_{E} and Δt_{L}. This increment can be computed at run time from the previous cycle of computation, i.e., $\Delta t^{n+1} = \min(\Delta t_{\mathrm{E}}^{n}, \Delta t_{\mathrm{L}}^{n})$. However, Noh [1] observes that several applications typically have $\Delta t_{\mathrm{L}} \ll \Delta t_{\mathrm{E}}$, and allows Δt_{E} to be a (stable) multiple of Δt_{L}.

To simplify the discussion on temporal coupling, we will refer to the set of equations integrated by the Eulerian solver as E. Similarly, L refers to the set of equations integrated by the Lagrangian solver. The solution of E depends on boundary conditions provided by L, which depends on the state of L, and vice versa. The solution is assumed to be available at discrete times up to t^n.

One can advance the solution by integrating E using $L(t^n)$ and by integrating L using $E(t^n)$; this simple approach is called *concurrent time coupling* and it is used in the VTF as it is more suitable for solution by parallel computers than staggered methods. An example of a staggered method is as follows: E is integrated using $L(t^n)$ and L is subsequently integrated using $E(t^{n+1})$. With staggered methods, only one set of equations can be solved at any given time.

A viable alternative is called *PC-Heun time coupling* [32]. After the first integration, one can re-integrate E using a combination (average) of $L(t^n)$ and $L(t^{n+1})$ to get a new $E(t^{n+1})$ and, concurrently, re-integrate L using a combination (average) of $E(t^n)$ and $E(t^{n+1})$ to get a new $L(t^{n+1})$. Using a predictor–corrector scheme such as this is often not practical because of its high overhead in both CPU and memory requirements. Numerical experiments using this scheme are presented in Section 5.2.

4. Coupling scheme

GEL is a boundary condition coupling scheme for Eulerian and Lagrangian solvers which are sharing portions of their boundaries (their EL-interface). In this section, we will describe GEL, and variations of it, in detail.

In formulating the boundary condition exchange, we make the following assumptions: (1) the EL-interface is defined by the boundary as geometrically determined by the Lagrangian solver. It is identified by $\varphi = 0$; (2) the Lagrangian solver uses a natural (pressure) boundary condition at the EL-interface; (3) the Eulerian solver requires a no-flux boundary condition (with free-slip) at the EL-interface.

A consequence of the first assumption is that, as stated earlier, the EL-interface location is recomputed as the Lagrangian boundary moves.

As for the second assumption, either displacements or force boundary conditions could be applied to the boundary of a Lagrangian solver. The second assumption indicates that only a force boundary condition is used. In our implementation, pressure is linearly interpolated in the Eulerian domain at the location of the Lagrangian pressure control points and used to enforce the traction boundary condition. It is interesting to note that applying a velocity boundary condition by using the Eulerian velocity at the boundary does not work. This can be seen with the following one-dimensional experiment. Imagine a shock wave in Ω_{E} traveling towards an initially stationary solid Ω_{L}. Since the solid is initially stationary, it will act as a reflective boundary to the Eulerian solver at the first integration, but this implies that the Eulerian flow velocity at the boundary remains zero also after marching by one time step. This information is fed back to the solid, which, therefore, remains still. Thus, there is no shock transmission in the Lagrangian domain when we would expect one.

The last assumption implies that the Eulerian solver sees the domain computed by the Lagrangian solver only as a moving boundary (completely ignoring all states in the interior of the Lagrangian domain). The only information needed from Ω_{L} is the velocity vector evaluated at the boundary. This conclusion is similar to the one obtained in [33] for coupling compressible to incompressible flow.

4.1. Populating the ghost cells: fluid solver boundary condition

Population of a ghost cell states requires an extrapolation algorithm operating on the real flow. In the following, the state that is extrapolated from Ω_{R} is marked with the subscript E, whereas the state that is evaluated from $\partial\Omega_{\mathrm{EL}}$ is denoted by W. The populated state in Ω_{G} is denoted by G.

The level set φ introduces a vector field of normals $\mathbf{n} = \nabla\varphi/\|\nabla\varphi\|$. We define $\mathbf{t}$ to be a unit vector normal to $\mathbf{n}$. The projection of a ghost node over Ω_{EL} is

$$\mathbf{x}_{\mathrm{W}} = \mathbf{x}_{\mathrm{G}}(i,j) + \varphi(i,j)\mathbf{n} \tag{8}$$

and the corresponding boundary velocity $\mathbf{V}_{\mathrm{W}}$ can be found by interpolation of Lagrangian boundary values at $\mathbf{x}_{\mathrm{W}}$.

For the extrapolation algorithm, the approach of capturing the EL-interface as a contact discontinuity (proposed in [10,9]) suggests that pressure and normal velocity should be continuous across the interface. However, the choice of the extrapolation scheme is, in general, not unique. In this paper, we experimented with different algorithms for extrapolating density, pressure, and the flow field velocity. These techniques are described in the following sections, and they consist of one-sided constant extrapolation (injection), a variation on constant extrapolation (reflection), and linear extrapolation (mirroring). Results for each of these schemes can be found in the second part of this paper for one- and two-dimensional problems. The current section is closed by a few considerations on the implementation of the boundary conditions as a Riemann problem.

4.1.1. Constant extrapolation or injection

Constant extrapolation of a scalar quantity I can be achieved through advection by integrating the Eikonal equation

$$I_t + \mathbf{n}\cdot\nabla I = 0, \tag{9}$$

subject to the boundary condition

$$I = I_{\mathrm{W}} \quad \text{on } \partial\Omega_{\mathrm{EL}}.$$

The numerical discretization of Eq. (9) can be implemented as a first-order upwind space discretization with first-order accurate time integration [34]. The equation has to be solved for a number of pseudo time steps until the ghost region has been fully populated [9,10]. Our experience is that this scheme is robust even for irregular interfaces and shock interactions, since sharp variations in the advected quantities are smoothed by the first-order advection algorithm. To reduce computational cost, advection needs to be performed only on a tiny strip of the computational domain, enclosing the ghost region and the closest strip of real flow region (say $-\Delta x \leqslant \varphi \leqslant \varphi_{\mathrm{s}}$).

The prescription for populating a ghost node is

$$\begin{pmatrix} \rho_{\mathrm{G}} \\ \mathbf{V}_{\mathrm{G}} \\ P_{\mathrm{G}} \end{pmatrix} = \begin{pmatrix} \rho_{\mathrm{E}} \\ (\mathbf{V}_{\mathrm{W}}\cdot\mathbf{n})\mathbf{n} + \mathbf{V}_{\mathrm{E}} - (\mathbf{V}_{\mathrm{E}}\cdot\mathbf{n})\mathbf{n} \\ P_{\mathrm{E}} \end{pmatrix}, \tag{10}$$

which assigns the normal velocity of the Lagrangian boundary to the normal velocity of the ghost node. We call this process the normal velocity treatment by *injection*.

4.1.2. Extrapolation by reflection

The extrapolation scheme by *reflection* is a variation of the extrapolation by injection. In the standard treatment of an impermeable wall, the normal component of the velocity in a ghost cell is set to be antisymmetric with respect to the normal velocity in the reference frame moving at $(\mathbf{V}_\mathrm{W}\cdot\mathbf{n})\mathbf{n}$. Based on this argument, we propose an alternate formula for $\mathbf{V}_\mathrm{G}$

$$\begin{pmatrix}\rho_\mathrm{G}\\ \mathbf{V}_\mathrm{G}\\ P_\mathrm{G}\end{pmatrix}=\begin{pmatrix}\rho_\mathrm{E}\\ (2\mathbf{V}_\mathrm{W}\cdot\mathbf{n}-\mathbf{V}_\mathrm{E}\cdot\mathbf{n})\mathbf{n}+(\mathbf{V}_\mathrm{E}\cdot\mathbf{t})\mathbf{t}\\ P_\mathrm{E}\end{pmatrix}. \tag{11}$$

The normal velocity treatment by reflection relies on advection as the previous treatment. It is not a linear extrapolation because it does not account for the distance from the wall to the ghost node. An advantage over linear extrapolation is that this prescription avoids the overshoot of an extrapolated velocity when the interface is very close to a real node. An advantage over injection is that the antisymmetric treatment of normal velocity is a more accurate implementation of an impermeable wall.

4.1.3. Linear extrapolation or mirroring

Linear extrapolation in the ghost region can be performed by computing the mirror image $\hat{\mathbf{x}}$ in Ω_R of the (i,j) ghost node, $\mathbf{x}_\mathrm{G}$ [11,35]. This operation requires the normal distance of $\mathbf{x}_\mathrm{G}$ from $\partial\Omega_\mathrm{EL}$, which is (with the correct positive sign) the value $\varphi(i,j)$. Thus, the relation between $\hat{\mathbf{x}}$ and $\mathbf{x}_\mathrm{G}$ is provided by

$$\hat{\mathbf{x}}=\mathbf{x}_\mathrm{G}(i,j)+2\varphi(i,j)\mathbf{n}. \tag{12}$$

The prescription for populating the ghost state is formally identical to the one in Eq. (11), only now $\mathbf{V}_\mathrm{E}$, ρ_E, P_E are interpolated values (we use bilinear interpolation) at $\hat{\mathbf{x}}$ in Ω_R.

It should be noted that extrapolation by mirroring is based on the assumption that the EL-interface is moving at constant speed within a given time increment; in general, this is not true. A typical counterexample is an explicit FE solver, where all nodes assume a constant *acceleration* in any given time step. One can include acceleration effects by imposing a suitable pressure gradient in the ghost region. In one dimension, if $\ddot{x}_\mathrm{W}(t)$ is the acceleration of the wall, we would have

$$P_x(x_\mathrm{W},t)=-\rho(x_\mathrm{W},t)\ddot{x}_\mathrm{W}. \tag{13}$$

However, complications arise when the flow is not smooth (a shock, even when smeared, is associated with high acceleration) and implementation is difficult in two and three dimensions. Furthermore, one may argue that since the state on the EL-interface is only interpolated to first-order at best, it is of little advantage trying to predict the flow behavior at the interface to better than first-order.

4.1.4. Coupling as a Riemann problem

In this section, we suggest an improvement of GEL via formulation of a Riemann problem at the EL-interface. The implementation requires that the equations of state of the materials be communicated to a software module which solves the Riemann problem at the interface. In the simplest case of a linearized solver, knowledge of the stiffness (or the impedance) of the material on both sides of the EL-interface is sufficient. In any case, this approach involves a more complicated coupling algorithm.

The Riemann problem can be understood as follows. Consider an initial jump discontinuity at $x=0$ separating two constant states,

$$\mathbf{U}(x,t=0)=\begin{cases}\mathbf{U}_\mathrm{l} & \text{for } x<0,\\ \mathbf{U}_\mathrm{r} & \text{for } x>0.\end{cases} \tag{14}$$

Eq. (14) is an example of a shock-tube problem, where a hypothetical "membrane" situated at $x=0$ separates the two states at $t=0$. Without loss of generality, we take Ω_R to be on the right of the membrane and Ω_L to be on the left. If we consider only the subdomain Ω_L, this becomes a piston problem where the entire domain is initially a constant state with a boundary condition at $x=0$.

If the original shock-tube problem is to be solved as a coupled Eulerian–Lagrangian problem, then the Lagrangian scheme solves the piston problem for an interval dt subject to a natural (pressure) boundary condition P_{p}.

There is a unique driving pressure P_{p} for the piston problem that corresponds to the original shock-tube problem. This is the boundary condition that needs to be applied to the Lagrangian solver if it were to simulate the original system Eq. (14) correctly. It is natural to ask whether the pressure P_{l} corresponding to the left state $\mathbf{U}_{\mathrm{l}}$ and the pressure boundary condition P_{p} are equal. Rephrased in the context of GEL coupling, is the pressure boundary condition to the Lagrangian solver the pressure of the fluid given by the Eulerian solver? The answer is no, in general. In the case of a shock-tube filled with perfect gases having the ratio of specific heats γ, sound speed a and pressure P, with subscripts l and r denoting the left and right states, P_{l} and P_{rmp} are related by the shock-tube [17] equation

$$\frac{P_{\mathrm{l}}}{P_{\mathrm{r}}} = \frac{P_{\mathrm{p}}}{P_{\mathrm{r}}}\left[1 - \frac{(\gamma_{\mathrm{l}} - 1)(a_{\mathrm{r}}/a_{\mathrm{l}})(P_{\mathrm{p}}/P_{\mathrm{r}} - 1)}{\sqrt{2\gamma_{\mathrm{r}}[2\gamma_{\mathrm{r}} + (\gamma_{\mathrm{r}} + 1)(P_{\mathrm{p}}/P_{\mathrm{r}} - 1)]}}\right]^{-2\gamma_{\mathrm{l}}/(\gamma_{\mathrm{l}}-1)}. \tag{15}$$

Thus, when there is a discontinuity in pressure across the EL-interface, the pressure boundary condition (P_{p}) assigned to the Lagrangian solver is not the pressure given by the Eulerian solver (P_{l}). In practice, large jumps of pressure are often smeared and $P_{\mathrm{l}}/P_{\mathrm{r}} \approx 1$. In this limit, P_{l} is well approximated by P_{p} and solving the Riemann problem at the interface to calculate P_{p} is not necessary. Furthermore, these pressure discontinuities do not remain at the interface as they get convected away with the flow, and the resulting error is usually small.

5. One-dimensional tests

In this section, we will restrict ourselves to the analysis of one-dimensional test problems. Two-dimensional problems are discussed in Section 6.

The following results are obtained by using an essentially nonoscillatory local Lax–Friedrichs (ENO-LLF) flow solver [36,37] on a Cartesian grid. A local Lax–Friedrichs form [36] is used to avoid entropy violating shocks near sonic points, making the scheme very robust at the price of perhaps too much dissipation to accurately capture all the flow features. Time marching is achieved through the total variation diminishing (TVD) third-order Runge–Kutta method devised by Shu and Osher [36]. The time step Δt is selected to account for the CFL stability condition.

We first consider a prescribed boundary motion to examine issues of conservation of mass and entropy in isentropic flows. The next step is to couple the boundary motion to the flow in a simple fashion and examine the problem of free expansion of a piston. The simplicity of this test case allows us to explore different temporal couplings and coupling strategies. Then we study the oscillations of a spring-mass system in a compressible gas. The problem is reducible to a nonlinear oscillator, and we compare trajectories of the piston in the underdamped and overdamped cases with the numerical solutions of the corresponding ordinary differential equation. Finally, we examine wave interactions in a setting where the Lagrangian and the Eulerian materials have identical acoustic impedance. This test, called the transparency test, assesses how transparently waves can be transmitted between domains that differ only in their numerical modeling. The Lagrangian modules were described previously in Section 2 of this paper.

5.1. Prescribed rigid boundary motion

Conservation is not guaranteed [7] by a scheme which does not take into account cut cells and the motion of cell interfaces during a time step Δt. In this section, the convergence properties of the treatments by reflection and injection are compared. Results for mirror flow extrapolation are also computed for completeness. In all cases, we show that the losses in mass and entropy decrease at least linearly with the mesh size, i.e., with first-order convergence. For this smooth one-dimensional problem, quadratic reduction of these errors could be achieved by an appropriate treatment of how boundary cells are cut by the interface [11,12]. As noted earlier, these schemes are potentially cumbersome in higher dimensions and they tend to provide only linear convergence in multi-dimensional simulations involving shocks [11].

We will consider a perfect gas confined between two rigid walls at $x_{\mathrm{r}} = 1.0$ and at $x_{\mathrm{l}} = 0.5 + v_{\mathrm{l}}t + a_{\mathrm{l}}/2t^2$. This problem has been previously considered by Forrer and Berger [11], who demonstrated second-order convergence. Initial conditions are

$$\begin{aligned} &\rho(x,0) = 1 + 0.2\cos(2\pi(x - 0.5)), \\ &v(x,0) = 2(1 - x)v_{\mathrm{l}}, \\ &P(x,0) = \rho(x,0)^{\gamma}. \end{aligned} \tag{16}$$

If the left wall is moving leftward ($v_{\mathrm{l}} < 0$), an expansion takes place and the flow field is isentropic for all times, $s(x,t) = s(x,0) = p(x,0)\rho(x,0)^{\gamma} = 1.0$. The numerical value of the entropy s can, therefore, be monitored for error analysis.

We denote an initial value with the superscript i and a final value with the superscript f. Discrepancies in mass Δm and entropy Δs at the final time are given by

$$\begin{aligned} &\Delta m = \left(\sum_{j=1}^{n_C^{\mathrm{i}}} \rho_j^{\mathrm{i}} |C_j| - \sum_{j=1}^{n_C^{\mathrm{f}}} \rho_j^{\mathrm{f}} |C_j| \right) \Bigg/ \sum_{j=1}^{n_C^{\mathrm{i}}} \rho_j^{\mathrm{i}} |C_j|, \\ &\Delta s = \sum_{j=1}^{n_C^{\mathrm{f}}} \left| s_j^{\mathrm{f}} - 1 \right| |C_j| \Bigg/ \sum_{j=1}^{n_C^{\mathrm{i}}} |C_j|, \end{aligned} \tag{17}$$

where $|C_j|$ is the length of that part of a computational cell C_j that lies in Ω_{R}. The final time t^{f} is chosen so that the walls are located exactly at the interface between two grid nodes. Since, at that time, there are no cut cells, the two equations above are exact estimates of the errors of a piecewise constant solution. Thus, there are exactly n_C^{i} cells belonging to Ω_{R} at time t_{i} and n_C^{f} cells at time t_{f}. We also introduce a measure of the error of the entropy at the left moving wall

$$\Delta s_{\mathrm{W}} = \left| s_{\mathrm{W}}^{\mathrm{f}} - 1 \right|, \tag{18}$$

where W is the index of the left boundary cell.

Equipped with these error estimators, we now consider two cases and study the convergence of the results as the grid spacing h decreases. In all the computations, a fixed ratio $\mathrm{d}t/h = 0.32$ is used, corresponding to a CFL number approximately equal to 0.6.

Case A. We set the left wall velocity $v_{\mathrm{l}} = -0.5$ and the acceleration $a_{\mathrm{l}} = 0$. Results for different grid refinements at $t^{\mathrm{f}} = 0.5$ are shown in log–log plots in Fig. 4 (left column). Reference lines for linear and quadratic convergence are also displayed.

For all the extrapolation schemes, the convergence rate is linear for Δm and quadratic for Δs, whereas Δs_{W} displays an intermediate behavior. Mirroring gives the best performance in all the three error indicators, particularly the value Δm which is an order of magnitude smaller than the value obtained by using injection or reflection. This result is expected, as the linear extrapolation described in Section 4.1.3 is designed to implement an impermeable boundary for the case of constant velocity of the interface. However, we notice that mirroring is slower in achieving linear convergence of Δm and that the rate of decrease of Δs_{W} is only linear, and not superlinear, as for the other two extrapolation schemes.

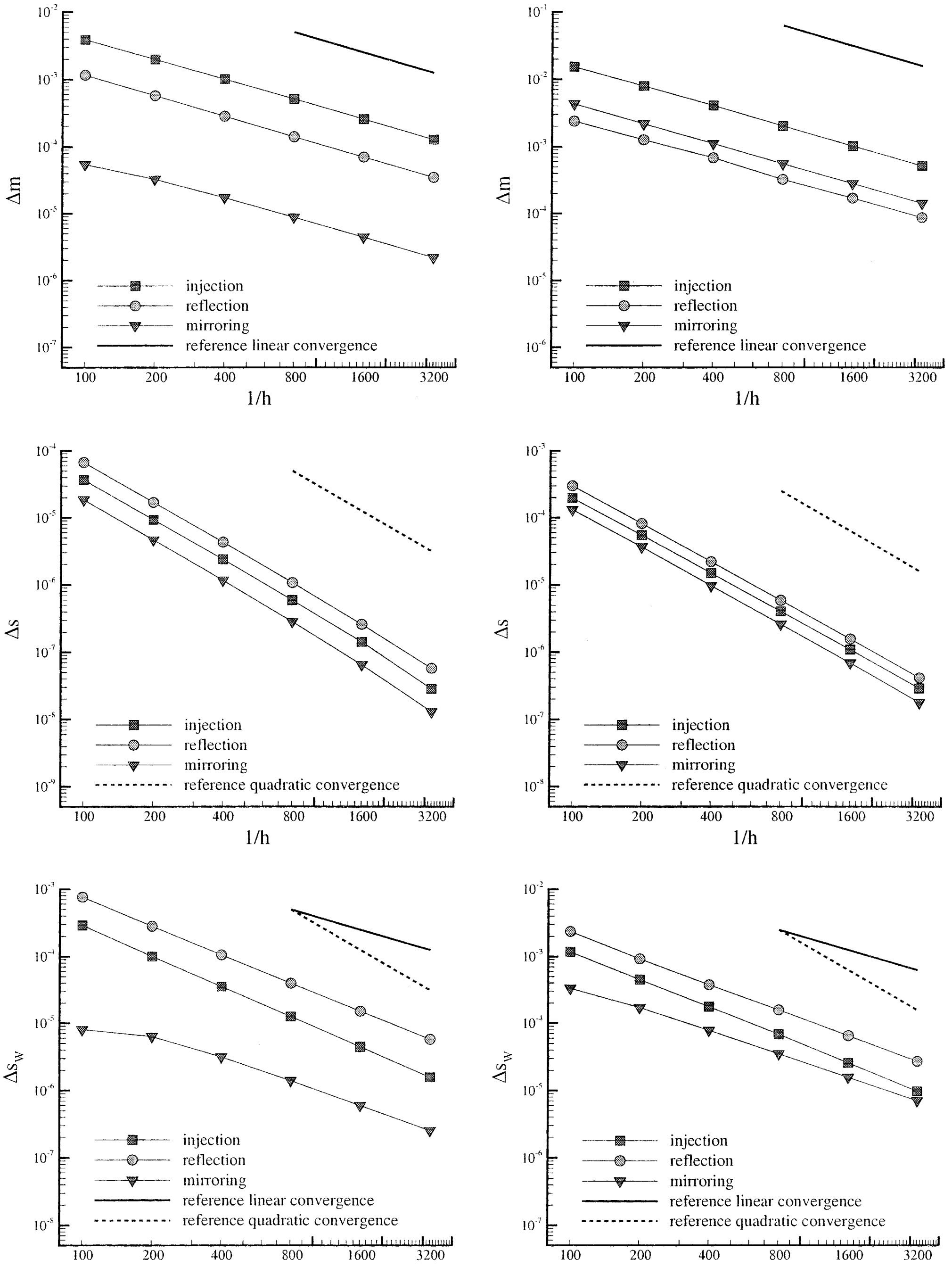

Fig. 4. Convergence study for prescribed rigid motion. Left column: wall moving with constant speed (case A). Right column: wall moving with constant acceleration (case B). Solid reference line indicates linear convergence; dashed reference line indicates quadratic convergence.

Case B. The left wall velocity is initially zero, while the acceleration is constant $a_l = -2.0$. Results at $t^f = 0.5$ are shown in log–log plots in Fig. 4 (right column). The convergence is again first-order for Δm and second-order for Δs. The error in entropy at the wall, Δs_W, decreases linearly for reflection and mirroring, and superlinearly for injection. In this situation of constant wall acceleration, the mass loss is slightly smaller with the reflection treatment. This result suggests that none of the extrapolation schemes considered here can be expected to minimize all error indicators for all possible tests. The role of interface acceleration is clearly important for simulations of shock interactions at an EL-interface, and it is further investigated in the following sections.

5.2. Free expansion

The free-expansion experiment is the simplest nontrivial test of GEL coupling. It is simple because an exact solution is available; it is nontrivial because the physics of the problem has the fluid and the solid tightly coupled together.

The setup of the free-expansion problem consists of a frictionless piston in a tube with a vacuum to the right of the piston and an initially constant state to the left at pressure, density, and sound speed given by P_0, ρ_0, $c = \sqrt{\gamma P_0/\rho_0}$, γ being the ratio of specific heats (see Fig. 5).

The solution to the problem can be simplified with the definition of the following time constant:

$$\tau = \frac{2mc_0}{P_0(1+\gamma)}, \tag{19}$$

where m is the mass per unit area of the frictionless piston. The speed of the piston and the pressure at the interface are determined by the method of characteristics [17]. Writing $t^* = t/\tau$, the solution is given by

$$\frac{u}{c_0} = \frac{2}{\gamma - 1}\left[1 - \left[\frac{1}{1+t^*}\right]^{(\gamma-1)/(\gamma+1)}\right] \tag{20}$$

and

$$\frac{P}{P_0} = \left[\frac{1}{1+t^*}\right]^{2\gamma/(\gamma+1)}. \tag{21}$$

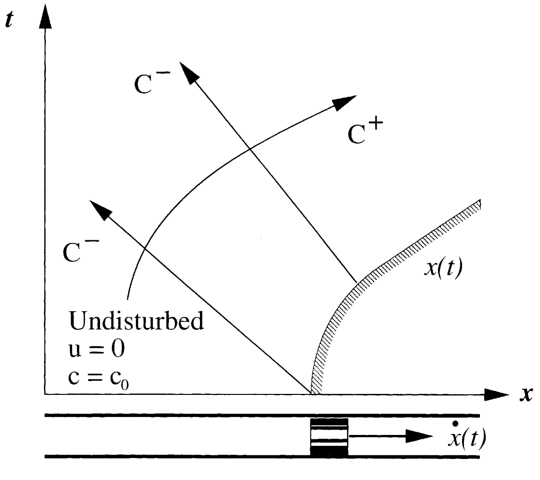

Fig. 5. Wave diagram for continuous piston withdrawal.

The exact velocity and pressure are plotted in Fig. 6.

The simplicity of this test case allows us to explore different temporal couplings and coupling strategies. Four cases are presented in this section.

1. Concurrent integration, normal velocity treatment by injection.
2. Concurrent integration, normal velocity treatment by reflection.
3. PC-Heun integration, normal velocity treatment by injection.
4. PC-Heun integration, normal velocity treatment by reflection.

In all cases, a third-order ENO solver is used for the solution of the fluid problem. For cases using concurrent integration (cases 1 and 2), a third-order TVD time integration is used for the fluid and explicit integration is used for the piston motion. For cases using PC-Heun integration (cases 3 and 4), a second-order predictor–corrector method is used for the fluid. The motion of the solid is also written as a first-order system and integrated with a second-order predictor–corrector method.

The grid size is compared against a characteristic length L,

$$L = c_0\tau = \frac{2c_0^2 m}{P_0(1+\gamma)}. \tag{22}$$

Values $\Delta x/L$ of 0.2, 0.1, 0.05, and 0.025 are used for the convergence study, corresponding to 20, 40, 80, and 160 grid cells. The total time of the simulation is $t^* = 2$. A CFL number of 0.1 is used for all simulations.

Figs. 7 and 8 compare the four cases against the analytical solution for the pressure at the interface (Eq. (21)) and the piston velocity (Eq. (20)), respectively. All cases converge to the analytical solution in pressure at times sufficiently far from $t = 0$. The L_1 metric

$$\|P - P_e\|_1 = \int_\tau |(P - P_e)|\,d\tau \tag{23}$$

is used to measure the distance between the exact solution P_e and the numerical solution P. Convergence results are shown in Table 1.

It is observed that while all cases show first-order convergence, those where predictor–corrector time coupling is used display better accuracy. Additionally, velocity treatment by reflection also improves accuracy in this test problem. Two observations can be made from examining Fig. 7. All cases studied overestimate the pressure at the beginning, and the reflection cases undershoot the exact profile (see the

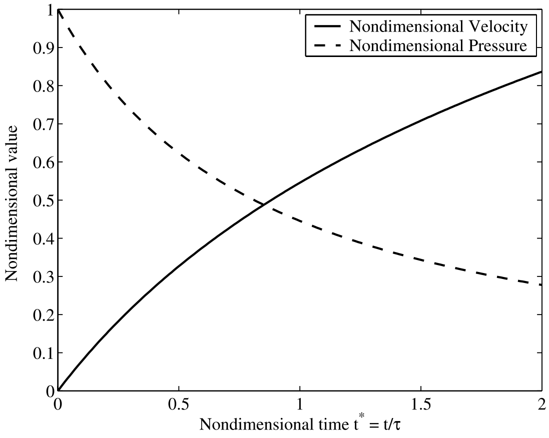

Fig. 6. Nondimensional velocity of piston of the piston–air system and the nondimensional pressure at the piston–air interface.

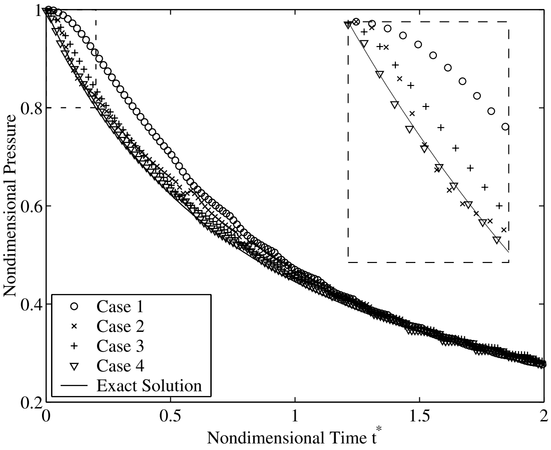

Fig. 7. Nondimensional pressure computed at the piston–air interface plotted against analytical solution ($1/\Delta x_E = 40$). The box on the right is a close-up of the first iterations.

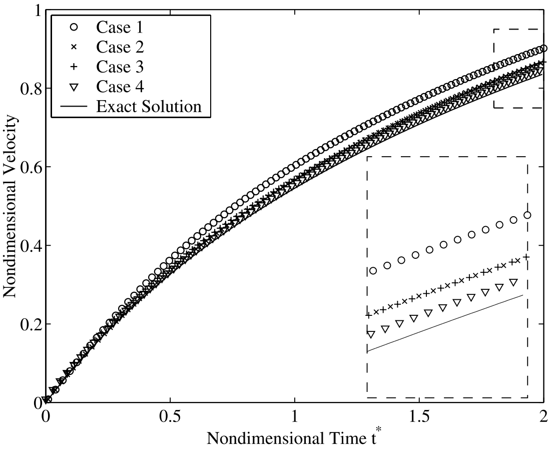

Fig. 8. Computed nondimensional piston velocity plotted against analytical solution ($1/\Delta x_E = 40$). The box on the right is a close-up of the solution for t^* close to 2.

zoom window of the figure). The initial overestimate can be understood as follows. In the first time step, the piston sees a pressure and accelerates but the fluid sees a stationary wall and no flow occurs; thus, the pressure drop that would accompany the expansion lags behind the piston motion. In the next step, since the Eulerian velocity is still zero and the piston has attained a finite velocity, reflection will assign twice the piston velocity in the ghost region while injection assigns the piston velocity in that region. In case 2, the ghost velocity is slightly too high and leads to a pressure drop, resulting in the undershoot displayed in the close-up of Fig. 7.

5.3. One-dimensional spring-mass system

A spring-mass system contains the key features of the very simplest of Lagrangian schemes for treating elastic solids.

Table 1
Results for the free-expansion problem

$1/\Delta x_E$	Case	Coupling	Integration	L_1 error
20	1	Injection	Concurrent	1.56×10^{-1}
20	2	Reflection	Concurrent	6.74×10^{-2}
20	3	Injection	PC-Heun	5.73×10^{-2}
20	4	Reflection	PC-Heun	2.07×10^{-2}
40	1	Injection	Concurrent	7.56×10^{-2}
40	2	Reflection	Concurrent	3.45×10^{-2}
40	3	Injection	PC-Heun	2.88×10^{-2}
40	4	Reflection	PC-Heun	1.01×10^{-2}
80	1	Injection	Concurrent	3.72×10^{-2}
80	2	Reflection	Concurrent	1.73×10^{-2}
80	3	Injection	PC-Heun	1.45×10^{-2}
80	4	Reflection	PC-Heun	4.96×10^{-3}
160	1	Injection	Concurrent	1.84×10^{-2}
160	2	Reflection	Concurrent	8.70×10^{-3}
160	3	Injection	PC-Heun	7.27×10^{-3}
160	4	Reflection	PC-Heun	2.47×10^{-3}

The equation of motion for a one-dimensional spring-mass system is

$$m\ddot{x} = P(t) - k(x - x_0), \tag{24}$$

where $x(t)$ is the position of the spring, x_0 is the equilibrium position, k is the stiffness, $P(t)$ is the pressure applied at the piston face, and m is the piston mass per unit area. Using the method of characteristics [17], we rewrite the equation of motion as

$$\ddot{x} + \omega_0^2(x - x_0) = \frac{P_0}{m}\left(1 - \frac{\gamma - 1}{2}\frac{\dot{x}}{c_0}\right)^{2\gamma/(\gamma-1)}. \tag{25}$$

At time $t = 0$, the piston is at rest and the pressure of the fluid is uniformly P_0. Recognizing $\omega_0^2 = k/m$ as the natural frequency of the harmonic oscillator, it is evident that Eq. (25) describes a nonlinear oscillator with an equilibrium point at $x = x_0 + P_0/k$. To formulate the problem in nondimensional form, take $L = P_0/k$ and ω_0 to be the characteristic length and frequency, respectively. Then Eq. (25) becomes

$$X'' + X = (1 - \alpha X')^{2\gamma/(\gamma-1)} \tag{26}$$

with

$$\alpha = \frac{\gamma - 1}{2}\frac{\omega_0 L}{c_0}, \tag{27}$$

where $X = (x - x_0)/L$ and the derivative is taken with respect to $\tau = t\omega_0$.

When $\alpha \ll 1$, linearization of Eq. (26) leads to an equation describing underdamped motion. With initial conditions $x(0) = 0$ and $\dot{x}(0) = 0$, the solution is

$$\frac{x}{L} = 1 - e^{(-\gamma/(\gamma-1))\alpha\omega_0 t}\left(\cos\beta\omega_0 t + \frac{\gamma}{\gamma - 1}\frac{\alpha}{\beta}\sin\beta\omega_0 t\right) \tag{28}$$

with

$$\beta = \sqrt{1 - \left(\frac{\gamma}{\gamma - 1}\alpha\right)^2}. \tag{29}$$

Although approximate, Eq. (28) suggests that for α large enough (say $\alpha > (\gamma - 1)/\gamma$), the solution becomes overdamped, i.e., the trajectory does not exhibit oscillations.

We consider next a numerical solution of Eq. (26) with zero initial displacement and velocity. The coupling algorithm uses concurrent integration and normal velocity treatment by reflection.

Case A. $P_0 = 10^6\,\mathrm{Pa}$, $\rho_0 = 4\,\mathrm{kg/m^3}$, $k = 10^7\,\mathrm{N/m}$ and $m = 3\,\mathrm{kg/m^2}$. For $\gamma = 1.4$, Eq. (27) gives $\alpha = 0.062$.

Case B. $P_0 = 10^6\,\mathrm{Pa}$, $\rho_0 = 4\,\mathrm{kg/m^3}$, $k = 2 \times 10^7\,\mathrm{N/m}$ and $m = 0.02\,\mathrm{kg/m^2}$. For $\gamma = 1.4$, Eq. (27) gives $\alpha = 0.53$.

Results are shown in Fig. 9 (case A) and Fig. 10 (case B). Fig. 9 displays two different grid resolutions ($\Delta x/L = 0.2$ and $\Delta x/L = 0.1$) both for first-order and second-order ENO. Note that in the coarsest case, only about eight nodes are swept back and forth by the interface. The trajectory is reconstructed almost exactly,

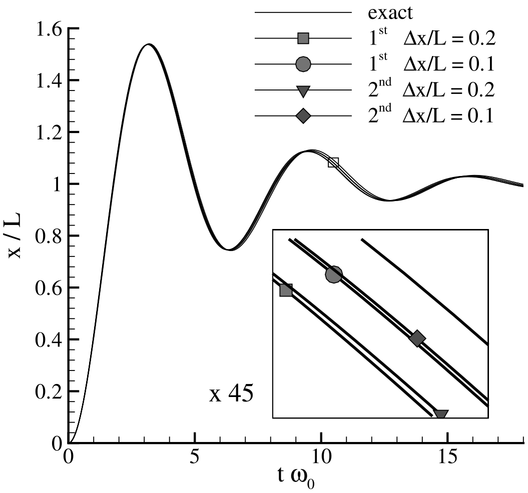

Fig. 9. Computed trajectories of the piston in the underdamped spring-mass case (case A) are compared against the numerical solution of Eq. (26) (labeled as “exact”) for a piston initially at rest. The box displays a detail of the trajectories, magnified 45×, at $\tau \cdot \omega_0 \approx 11$.

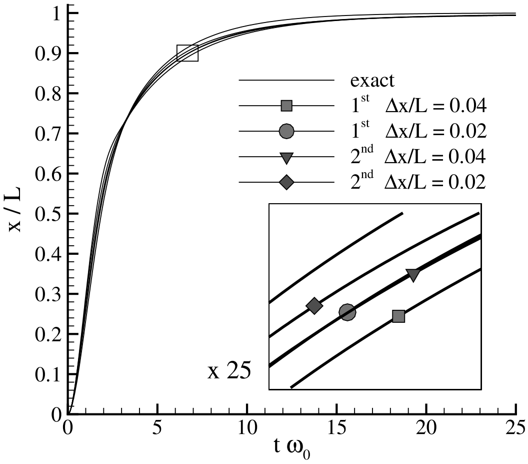

Fig. 10. Computed trajectories of the piston in the overdamped spring-mass case (case B) are compared against the numerical solution of Eq. (26) (labeled as “exact”) for a piston initially at rest. The box displays a detail of the trajectories, magnified 25×, at $\tau \cdot \omega_0 \approx 7$.

with a small phase error at later times. The box inside the plot magnifies the trajectory by a factor of 45. Since the accuracy of the scheme at the boundary is expected to be of order Δx, it comes as no surprise that the coupling with the second-order solver performs only slightly better than the one with the first-order solver.

For case B, the two grid resolutions are $\Delta x/L = 0.04$ and $\Delta x/L = 0.02$. In Fig. 10, the convergence is again linear and we note a small phase error at later times. The close-up box magnifies the trajectory by a factor of 25. This time, second-order ENO shows improved accuracy with respect to first-order ENO.

5.4. Transparency test

In this section, we propose a test problem for evaluating the accuracies of different coupling techniques. A transparency test consists of a domain of a single material with a fictitiously introduced interface that separates it into two abutting subdomains. Each subdomain is separately solved and updated so that coupling between domains occurs only via boundary condition exchanges.

Clearly, the fictitious interface will be transparent to waves in the case of perfect coupling (hence the name). Although the test is easily generalizable to higher dimensions, only one-dimensional tests are performed here. We will refer to them as (1) the EL transparency test, where a shock wave travels from Ω_E to Ω_L, and (2) the LE transparency test, where a shock wave travels from Ω_L to Ω_E.

5.4.1. The one-dimensional EL transparency test

In the EL case, we consider a shock wave of Mach numbers 1.2 and 1.5 initially propagating on the Eulerian mesh. A fixed Eulerian grid size of $\Delta x_E = 0.05$ and two different Lagrangian grid sizes, $\Delta x_L = 0.05$ and $\Delta x_L = 0.025$, are considered. Results of coupling using normal velocity treatment by injection and by reflection are compared.

The problem has no natural length scale. Discretizing the solution introduces the Eulerian grid spacing Δx_E and the two Lagrangian grid spacings (before and after compression) Δx_{L1} and Δx_{L2}. From mass conservation, the two Lagrangian length scales are related by $\Delta x_{L1}/\Delta x_{L2} = \rho_2/\rho_1$, where ρ_1 and ρ_2 are the pre- and post-shock densities, respectively. Other relevant length scales include the width of the shock in the Lagrangian domain, which depends on the artificial viscosity and artificial heat conduction models, and the width of the shock in the Eulerian domain, which depends on the details of the Eulerian solver and is typically three to five mesh points.

Representative results are shown in Fig. 11 for a shock Mach number 1.5, $\Delta x_E = 0.05$ and $\Delta x_{L1} = 0.025$. The plots show the spatial density profiles at selected times. The dotted line in the middle represents the one-dimensional EL-interface. The Eulerian domain is to the left of the interface; the Lagrangian domain to the right. The exact shock profile is shown as the solid line through the data. Every second data point is plotted in the Eulerian domain, and every fourth data point is plotted in the Lagrangian domain to avoid cluttering the figure. The coupling is performed using the injection method of Eq. (10).

We remark that the state in Ω_E is quite uniform after the transmission, and that the EL-interface exhibits "good" transparency (a quantitative measure of error will be introduced later). Additionally, the shock wave that forms in Ω_L has the correct strength and position. The density profile in Ω_L in the post-shocked state is uniform except near the boundary, which exhibits the well-known effect of wall-heating [27] due to artificial viscosity. Such wall-heating can be reduced or removed by adding an artificial heat flux.

The pressure computed by the Eulerian solver is nondimensionalized by a constant equal to the exact post-shock pressure P_s. To assess the transparency of the EL test case, the following metric is used:

$$\left\| \frac{P}{P_s} - 1 \right\|_1 = \int_{\Omega_E} \left| \left(\frac{P}{P_s} - 1 \right) \right| d\Omega. \tag{30}$$

Eq. (30) gives a measure of the distance between the exact Eulerian solution and the computed solution. All errors quoted in this section use Eq. (30) evaluated at $t = 0.001$ s.

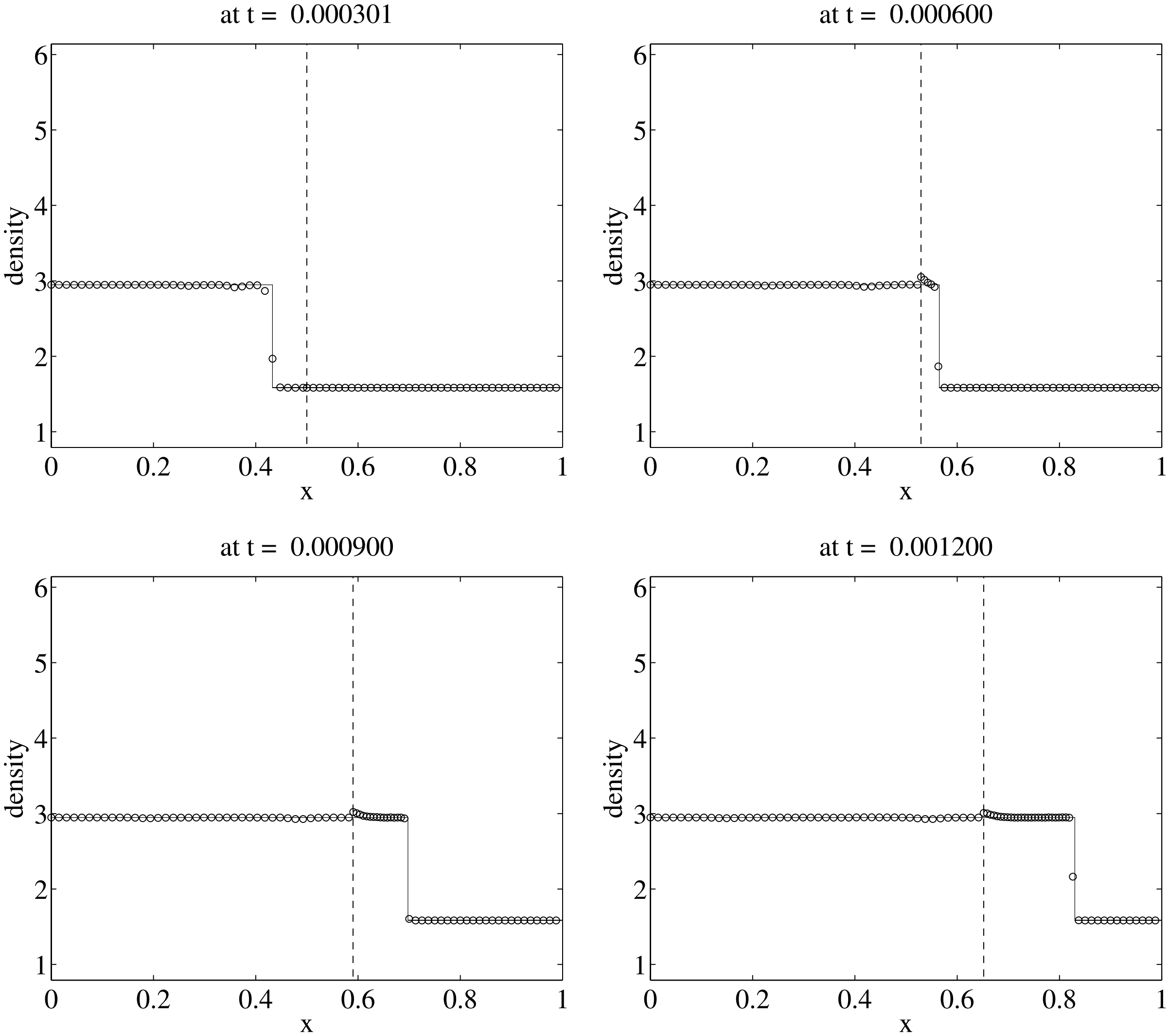

Fig. 11. Spatial profiles of density plotted at selected times for the EL transparency test. Eulerian solution is to the left of the dotted line and Lagrangian solution to the right.

The errors of a subset of cases studied are given in Table 2. To give an idea of the size of the errors in the table, the nondimensional pressure profile for the first case (corresponding to an error of 4.23×10^{-4}) is plotted in Fig. 12. It can be seen that most of the contribution to the error comes from the initial startup error, a consequence of the prescribed sharp shock profile smearing itself out across a few computational cells. Within the range of shock strengths and grid sizes tested, we observe that the interface behaved transparently with a small error in the solution, quite independent of the coupling scheme used.

5.4.2. The one-dimensional LE transparency test

In the one-dimensional LE transparency test, a shock is formed in the Lagrangian domain. We will see that shock waves no longer transmit across the EL-interface transparently. Accuracy is measured by

Table 2
Error analysis of the EL test

$1/\Delta x_{\mathrm{E}}$	$1/\Delta x_{\mathrm{L1}}$	Coupling	Mach No.	L_1 error
400	200	Injection	1.5	4.23×10^{-4}
400	200	Injection	1.2	1.87×10^{-4}
400	400	Injection	1.2	1.31×10^{-4}
400	400	Reflection	1.2	1.35×10^{-4}
400	200	Reflection	1.2	1.81×10^{-4}
400	200	Reflection	1.5	4.69×10^{-4}

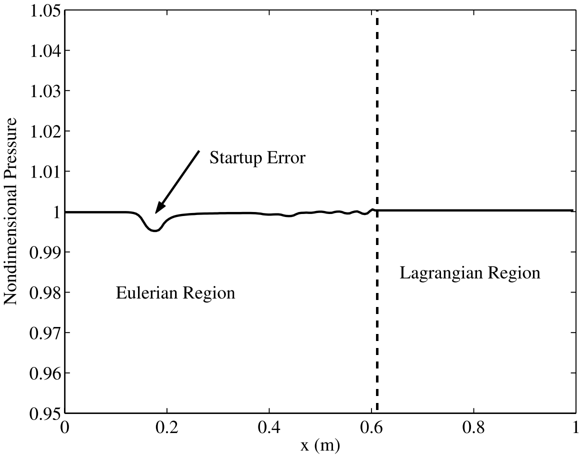

Fig. 12. A plot of the nondimensional pressure profile for the EL transparency test after shock transmission.

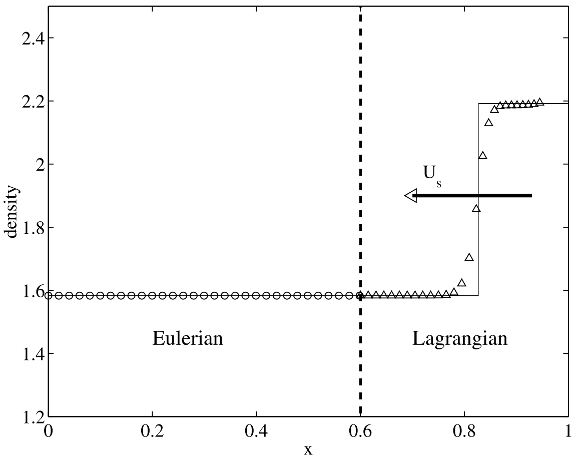

Fig. 13. One-dimensional LE (Lagrangian–Eulerian) transparency test setup (with mesh points shown).

computing the error in pressure in the Lagrangian domain after the shock has passed into the Eulerian domain. The L_1 norm of the error in pressure is given by

$$\left\| \frac{P}{P_s} - 1 \right\|_1 = \int_{\Omega_L} \left| \left(\frac{P}{P_s} - 1 \right) \right| \mathrm{d}\Omega, \tag{31}$$

where the integral is approximated over the Lagrangian domain, Ω_L. As in the previous error estimate, P_s is a constant equal to the exact post-shock pressure.

Following the conventions in continuum mechanics, we will use X to represent the original (reference, Lagrangian) coordinate. The setup is a one-dimensional "column of gas" of unit length divided into two regions. The region $0.6 < X < 1$ of the gas in the reference configuration is modeled by a Lagrangian solver. The region $0 < X < 0.6$ is modeled by an Eulerian solver (see Fig. 13). A shock is created in the Lagrangian region by instantaneously increasing the right hand boundary (piston) velocity at time $t = 0$.

At $X = 1.0$, we have $\dot{X}(t) = -U_p H(t)$, where $H(t)$ is the Heaviside step function, and U_p is the piston velocity. For a perfect gas, the Mach number M_s of a shock moving in an undisturbed medium (with sound speed c_1) is related to U_p by [17]

$$M_s(U_p) = \frac{U_p}{c_1} + \sqrt{\frac{U_p}{c_1}^2 + 4\left(\frac{2}{\gamma+1}\right)^2} \Big/ 2\left(\frac{2}{\gamma+1}\right). \tag{32}$$

For the LE test, the following parameters are considered: $1/\Delta x_E = 200$ and 400, $1/\Delta x_{L1} = 100$, 200, and 400. A perfect gas equation of state with $\gamma = 1.4$ is used. Piston velocities of 100, 200, and 400 m/s are specified. The corresponding Mach numbers from Eq. (32) are $M_s = 1.22$ and 1.48. Results of coupling using normal velocity treatment by injection and by reflection are compared.

The density, velocity, and pressure profiles of a representative case with $1/\Delta x_E = 200$ and $1/\Delta x_{L1} = 200$ at $M_s = 1.22$ are presented. Fig. 14 shows the results obtained with coupling by reflection and Fig. 15 the results obtained with coupling by injection.

It is clear from Figs. 14 and 15 that the EL-interface causes spurious reflections as the shock wave passes from the Lagrangian domain into the Eulerian domain. The L_1 error in pressure as computed by Eq. (31) measures the absolute area of the "blip" in pressure that appeared after the shock transmission through the EL-interface. It is clear from the figures that the blip is smaller in the injection case.

The errors as a function of time are shown in Fig. 16 for the two cases plotted in Figs. 14 and 15. The vertical dotted line depicts the time of arrival of a perfectly sharp Mach 1.22 shock to the EL-interface.

The error before the shock reaches the interface is due to the smearing of the shock in the Lagrangian domain. It grows as the shock widens to its natural width (which is a function of the artificial viscosity parameters) and then levels off. After transmission, the error in pressure (in the Lagrangian domain) reaches a final value quickly. If the interface was truly transparent, this error would be zero. The error analysis (Table 3) gives the error of various test cases at the final simulation time of 2 ms.

The explanations for these spurious reflections are as follows. As the shock travels across the EL-interface, all the gradients in the ghost region are set to zero through the advection algorithm. In other words, the smeared shock wave is truncated at the EL-interface. As a result, the Eulerian solver does not see the full extent of the incoming wave. Most notably, the removal of the pressure gradient in the ghost region reduces the acceleration effect that accompanies a shock wave. Therefore, the pressure as seen by the Eulerian solver does not build up as quickly as it should and this gets fed back as a traction boundary condition to the Lagrangian solver. This explains why there is a rarefaction-like reflection back into the Lagrangian domain as the shock wave passes through the interface.

Another observation is that the normal velocity treatment by injection results in a lower error than the treatment by reflection. A likely explanation is that the reflection strategy, which is designed to approximate the effect of a nonaccelerating boundary, filters (or averages) out acceleration effects more than the injection

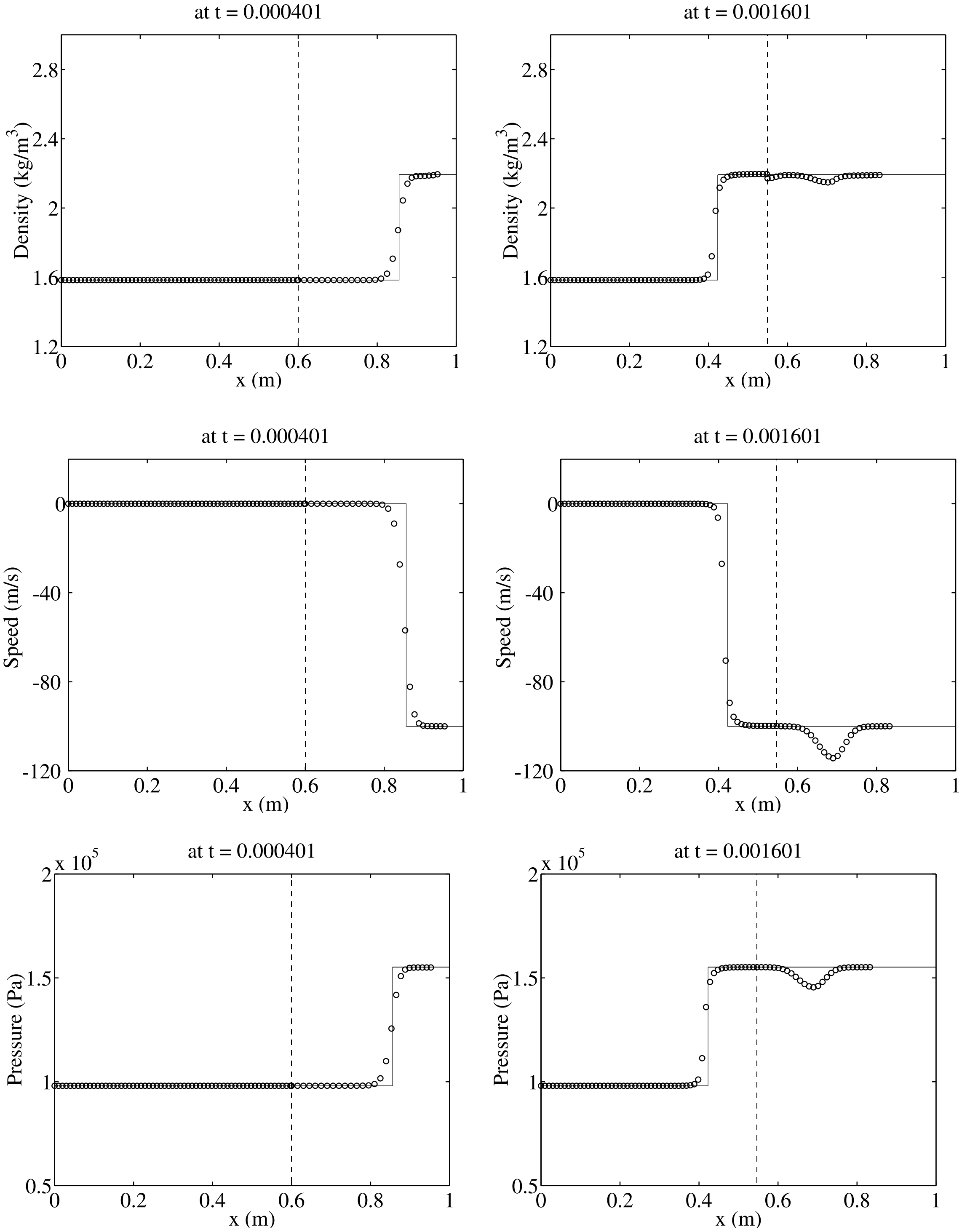

Fig. 14. Transparency test, shock wave generated in Lagrangian region and transmitted to Eulerian region. Normal velocity treatment by reflection is used.

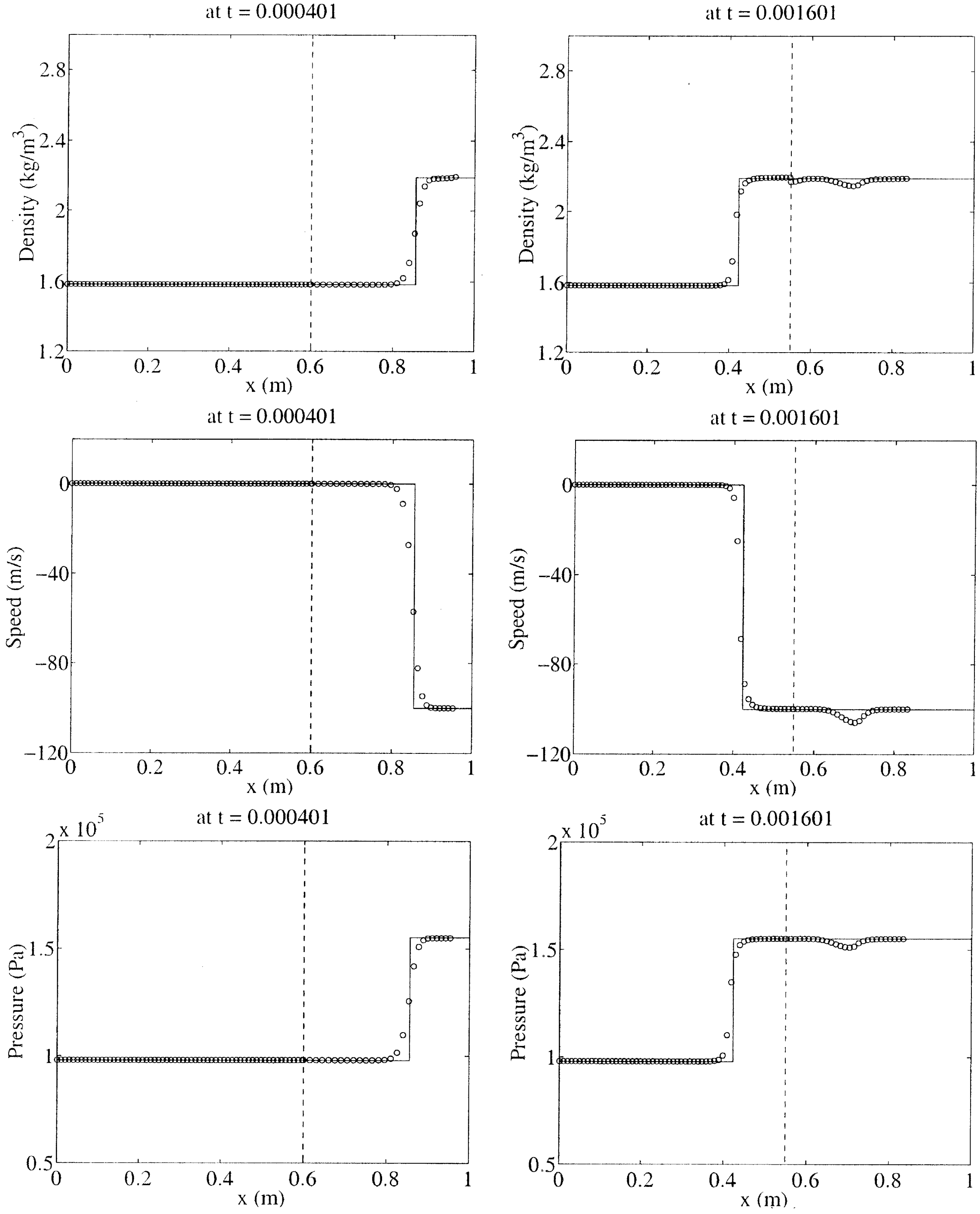

Fig. 15. Transparency test, shock wave generated in Lagrangian region and transmitted to Eulerian region. Velocity BC by injection used.

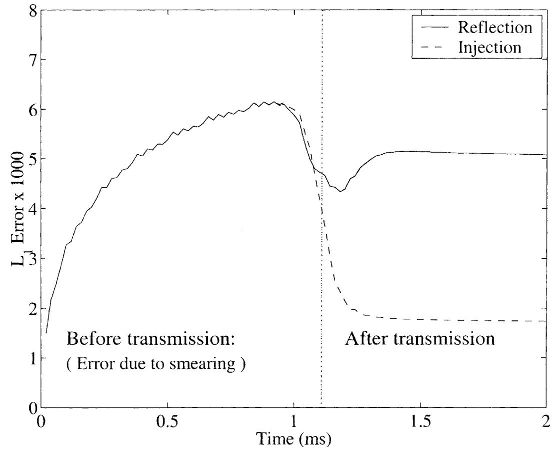

Fig. 16. Error in pressure: reflection vs injection

Table 3
Error analysis for the LE test

$1/\Delta x_E$	$1/\Delta x_{L1}$	Coupling	Mach No.	L_1 error
200	200	Reflection	1.22	5.07×10^{-3}
200	200	Injection	1.22	1.73×10^{-3}
200	100	Reflection	1.22	4.34×10^{-3}
200	400	Reflection	1.22	5.74×10^{-3}
400	400	Reflection	1.22	2.52×10^{-3}
400	200	Reflection	1.22	2.17×10^{-3}
400	400	Injection	1.22	0.87×10^{-3}
200	200	Injection	1.48	2.97×10^{-3}

method does. This filtering is believed to play a significant role because in this transparency test the EL boundary is subjected to very high acceleration upon arrival of the shock (directly proportional to the steepness of the smeared shock wave).

Numerical results are shown in Table 3. Note that these errors are about an order of magnitude larger than those shown in Table 2 for the EL transparency test.

5.4.3. Transparency test conclusions

A series of experiments was done to test grid effects by refining the Lagrangian and the Eulerian grids. The effect of shock width and normal velocity treatment was also examined in separate tests. It is found that by simultaneously refining the grids, little gain in accuracy (as measured by the L_1 norm pressure) can be obtained. The most dramatic improvements (in terms of reducing the "blip") occur through refining the Eulerian grid and *coarsening* the Lagrangian grid. This can be explained by the argument in the previous section about the sharpness of the shock wave in the Lagrangian region.

A shock in the Eulerian solver has a natural width depending on the type and the order of the scheme. A shock in the Lagrangian solver also has a natural width associated with the amount of artificial viscosity and artificial heat conduction. A natural question arises whether matching these two widths is important. The answer appears to be negative. The reflection seems to depend most strongly on the amount of acceleration undergone by the EL-interface. This acceleration can be decreased by:

1. Coarsening the Lagrangian grid. Since for given values of the parameters used in the artificial viscosity, the shock has a constant mesh width (e.g., 5 mesh cells); a grid coarsening leads to a thicker shock (e.g., $5\Delta x$) which reduces the pressure gradient in the smeared profile.
2. Increasing artificial viscosity, which smears the shock over a larger number of mesh cells.

In addition, as the Eulerian grid is refined, a more accurate interpolation of pressure on the Eulerian grid onto the EL-interface is obtained. As a result, refining the Eulerian mesh improves the transmission of shock waves through the EL-interface.

6. Two-dimensional tests

The following results are obtained by using the same ENO-LLF Cartesian solver described in the previous section. The one-dimensional scheme is extended to higher dimensions via a dimension-by-dimension technique [38]. We first consider the interaction of a shock with walls modeled by Euler–Bernoulli beams. Again, we verify that mass is conserved to first-order in the grid resolution. This result is confirmed when we examine one of the few two-dimensional tests for moving boundaries available in the literature, the cylinder lift-off problem. Finally, the supersonic (superseismic) problem is described. We prove that a steady self-similar solution exists in the reference frame of the shock generating the load. Also, we show that numerical solutions of a simple channel flow evolve to this steady solution and that the shock deflection angles converge to the values predicted by the theory.

6.1. *The inflatable bladder*

In this experiment, a shock enters a two-dimensional cavity with the boundaries modeled by a system of Euler–Bernoulli beams. Although the axial symmetry of the problem is evident, no effort is made to enforce it in order to provide a basic test of the correctness of the implementation.

The test uses the following setup. The cavity, or deformable channel, has an upper and lower wall with identical properties. Each wall is composed of two very rigid shells located on the left and on the right to form the inlet and the outlet of the channel. A flexible shell is mounted between them at the center. The shells are modeled by 10 and 50 beam elements, respectively. The boundary conditions on the left are supersonic inflow and on the right, a reflecting wall.

At time $t = 0$, the cavity is undeformed, with the walls aligned in the horizontal direction and the shock positioned in the inlet. The pre-shock load acting on the inner side of the shells is balanced by an equal external pressure on the outside, so that the beams are in equilibrium. Fig. 17 displays the cavity at a later time. The rigid inlet and outlet shells are undeformed, but the upper and lower central shells have been deflected outward substantially by the post-shock pressure.

We define $|C_{i,j}|$ as a measure of the volume of each computational cell (possibly cut) where $\varphi < 0$. The quantity

$$\Delta m = \left(\sum_{i,j} \rho_{i,j} |C_{i,j}| - \sum_{i,j} \left(\rho_{i,j} |C_{i,j}| \right)_0 - tA_i (\rho u)_i \right) \Bigg/ \sum_{i,j} \left(\rho_{i,j} |C_{i,j}| \right)_0 \tag{33}$$

gives an indication of the amount of mass that is lost (or gained) in the process. The estimate of $|C_{i,j}|$ introduces only a higher order error in Eq. (33) at cut cells when evaluating the polygon intersection area.

The next series of figures illustrates the results in the context of a grid refinement study. A shock propagates from left to right through a perfect gas ($\gamma = 1.4$) at $M = 3$, with pre-shock pressure $P_0 = 10^7\,\mathrm{Pa}$ and density $\rho_0 = 100\,\mathrm{kg/m^3}$. The shock is initially located at $x_s = 0.03\,\mathrm{m}$.

The two central shells have a cross-sectional area $A = 6.35 \times 10^{-3}\,\mathrm{m^2}$, length $l = 0.6\,\mathrm{m}$, moment of inertia $I = 2.13 \times 10^{-8}\,\mathrm{m^3}$, and cross-sectional mass $m = 49.53\,\mathrm{kg/m}$. This set of parameters is chosen to

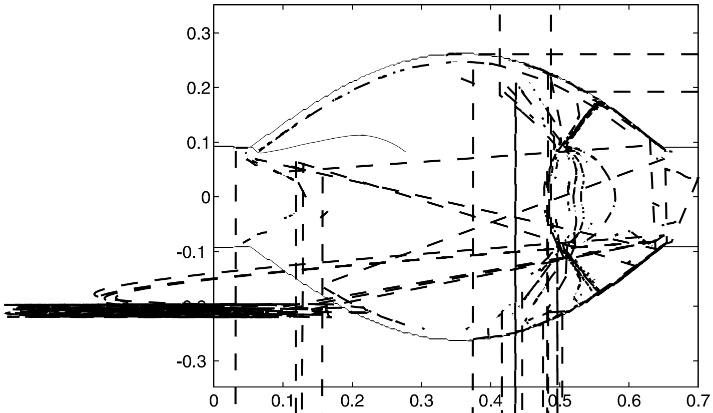

Fig. 17. Density contours of the inflatable bladder problem solution are displayed at time $\tau = 0.5345$.

generate a large deformation of the shells in a characteristic time $\tau = tc_0/L$ ($L = 0.7\,\mathrm{m}$ is the total length of the channel) of order unity.

A contour plot of density at $\tau = 0.5345$ is shown in Fig. 17 for a 201×201 grid. The shock has already been reflected from the closed end and is now moving leftward; at the left inlet, an expansion fan is observable.

The contour plot has 17 levels equally spaced, from 50 to $700\,\mathrm{kg/m^3}$. In Fig. 17, the contour plot also shows the ghost region, extending for a few cells beyond the line marked as *level set* = 0. Note that this line defines the boundary, as reconstructed on the Cartesian grid. The outermost line in Fig. 17 is simply the somewhat fuzzy division between the ghost region and the empty computational domain.

The test is executed with three different grid resolutions: 51×51, 101×101, and 201×201. The history of $\Delta m(t)$ is depicted in Fig. 18. High frequency oscillations are probably due to errors in estimating the area of the cut cells. Low frequency oscillations are related to actual mass loss or production, and their amplitude decreases with Δx decreasing.

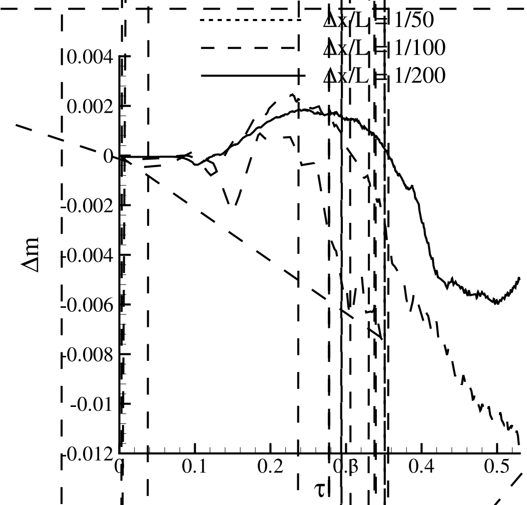

Fig. 18. The normalized mass loss is displayed as a function of time τ and parameterized by the grid refinement.

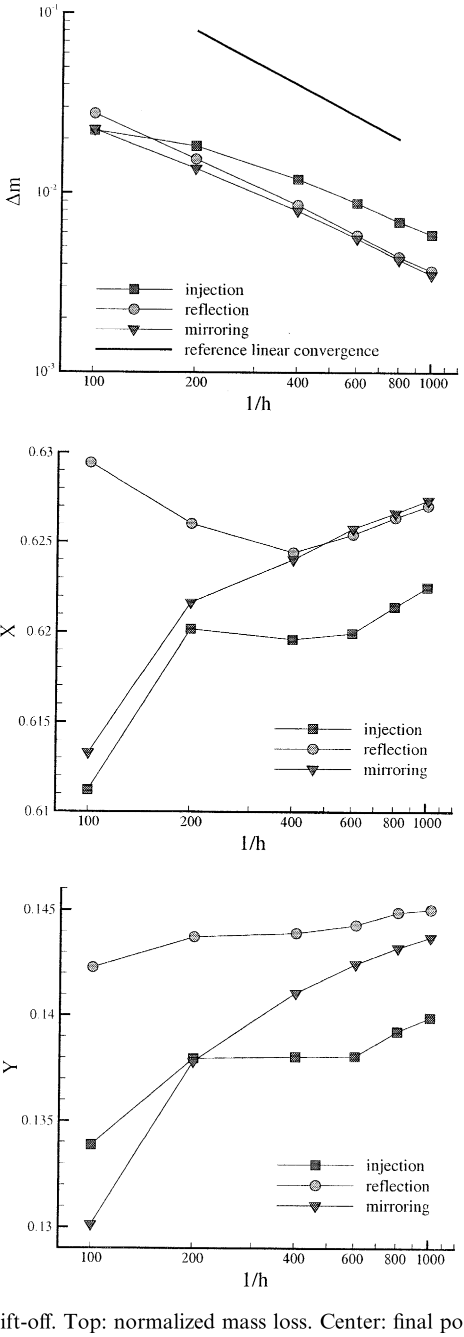

Fig. 20. Convergence study for cylinder lift-off. Top: normalized mass loss. Center: final position of cylinder's center (x-coordinate). Bottom: final position of cylinder's center (y-coordinate).

In the range of grid refinement between $1/h = 100$ and $1/h = 1000$, extrapolation by injection results in the slowest approach to a linear convergence rate. The other two extrapolating schemes perform slightly better. The slope between the two last points ($1/h = 800$ and $1/h = 1000$) in the Δm diagram is 0.96 for reflection, 0.97 for mirroring, and only 0.94 for injection. The values of Δm listed in [11] are comparable with the ones reported here. The difference in the convergence behavior between injection on one side and reflection and mirroring on the other is also visible in the plots of the x- and y-trajectories of Fig. 20.

6.3. Supersonic (superseismic) step load moving over a half-space

In this section, a verification test consisting of a compressible fluid interacting with an elastic solid is described. We consider the effect of a step load of amplitude P traveling with speed U_s across an elastic half-space (Fig. 21). The load is produced by a shock propagating in the fluid occupying the adjacent half-space below the elastic material. The value of this loading situation is that the deformation of the elastic solid affects the fluid flow by changing the shock angle so that the full fluid–solid coupling model is tested.

It is convenient to treat the two half-spaces separately, looking for a relation connecting P to the plane deflection, θ. In Section 6.3.1, results from the theory of propagation of disturbances in elastic materials are specialized to a case where steady-state solutions exist. In Section 6.3.2, the standard shock deflection relations for a perfect gas are used. It is assumed that no reflected shock exists in the gas.

Under the conditions described above, a time-independent similarity solution exists in the frame of the traveling shock. The solution is obtained by imposing continuity and mechanical equilibrium at the interface. In Section 6.3.3, we validate the coupling algorithm against the exact Mach number-shock deflection relation that is derived from this theory.

6.3.1. Elastic solid half-space

The load P in Fig. 21 is called superseismic when U_s is larger than the speed of propagation of disturbances in the elastic material, a situation that is encountered in studies on the effect of very strong blast waves propagating along the surface of the Earth. This test also has a direct relevance to high explosive applications, since it models the interaction of the leading shock front with the metal casing in a cylinder test experiment.

Bleich [39] investigated wave systems due to superseismic loads over elastic–plastic and granular materials, and we will follow his approach to analyzing the wave system and boundary deflection in the elastic region shown in Fig. 21. In the present study, we will restrict our attention to a linear elastic material.

The present analysis considers the case of plane strain ($\varepsilon_z = 0$) in an isotropic, homogeneous, linearly elastic medium subject to compression ($-P$). The load is applied in the undeformed configuration to be consistent with the FE solver.

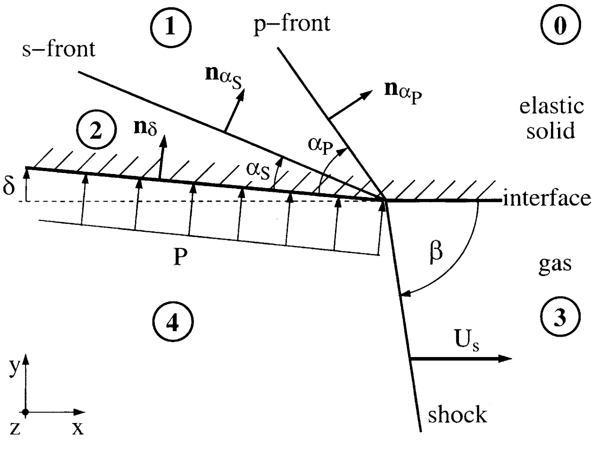

Fig. 21. Superseismic step load: p and s wave systems (above) and oblique shock (below).

In an elastic solid, stress waves propagate at two different speeds, corresponding to the propagation of dilatational (s) and distortional (p) disturbances [40]: $c_P = \sqrt{(\lambda + 2\mu)/\rho}$, $c_S = \sqrt{\mu/\rho}$. The parameters λ and μ are the Lamé constants, and ρ is the density of the medium. Following convention, we will refer to these waves as p and s waves, respectively.

Since the superseismic load travels faster than both the p and s disturbances in the elastic material, two oblique waves radiate out from the front of the step load. They form characteristic angles with the boundary of the half-space

$$\tan \alpha_P = \sqrt{U_s^2/c_P^2 - 1} \tag{34}$$

and

$$\tan \alpha_S = \sqrt{U_s^2/c_S^2 - 1}. \tag{35}$$

Fig. 21 displays the elastic solid half-plane (regions 0, 1 and 2) in the upper part, and the compressible flow half-plane (regions 3 and 4) in the lower part. The undisturbed region in the solid is labeled with 0; 1 is the region between the p and s fronts and 2 is the region behind the s front. The pre-shocked state in the fluid is 3 and the shocked state is 4.

At the p front, we expect discontinuities in the normal and tangential stresses. Call the normal discontinuity at the p front $\Delta\sigma$. The normal to the p front $\mathbf{n}_{\alpha_P}$ indicates one principal direction of the stress tensor $\overline{\sigma}_{\mathrm{I}} = \Delta\sigma$. The additional conditions of plane strain and no distortion determine a state of uniaxial strain (note that indices I, II, III here designate the principal stresses and directions)

$$\overline{\sigma}_{\mathrm{II}} = \overline{\sigma}_{\mathrm{III}} = \frac{v}{1 - v}\Delta\sigma. \tag{36}$$

In region 2, the load is applied in the direction perpendicular to the outer surface, $\mathbf{n}_\delta$. This assumption is consistent, since we plan to model the step load by an inviscid shock. Thus, $\mathbf{n}_\delta$ is also a principal axis of the stress tensor. By imposing equilibrium over an element at the loaded surface, we find $\sigma_{\mathrm{I}} = -P$. The second principal stress must be proportional to P also; we define R so that $\sigma_{\mathrm{II}} = R\sigma_{\mathrm{I}} = -RP$.

To determine the unknowns $\Delta\sigma$ and R, Bleich uses the continuity of normal and tangential stresses at the shear front (s front) and the equality $\sigma_{\mathrm{III}} = \overline{\sigma}_{\mathrm{III}}$. For a load P applied in the undeformed configuration (i.e. in the direction of the y axis),

$$\Delta\sigma = -\frac{P(1 - v)}{N} \cos 2\alpha_S \tag{37}$$

and

$$R = -1 + \frac{\cos 2\alpha_S}{N} \tag{38}$$

with

$$N = \cos^2 \alpha_S + (1 - 2v)\cos^2(\alpha_S - \alpha_P) - 1 + v. \tag{39}$$

The derivation of these quantities is given in Appendix A.

With the state of strain known, one can calculate the deflection δ as a function of the load applied, P (see Appendix A for details). We find that the velocities in the elastic solid at the interface are

$$\begin{pmatrix} u_{\mathrm{p}} \\ v_{\mathrm{p}} \end{pmatrix} = U_{\mathrm{s}}(1 - 2v)\frac{P}{4\mu N}\begin{pmatrix} \cos 2\alpha_S - \cos 2(\alpha_P - \alpha_S) \\ \sin 2\alpha_P \end{pmatrix}. \tag{40}$$

The deflection is given by

$$\tan\delta = \frac{v_\mathrm{p}}{U_\mathrm{s}+u_\mathrm{p}}. \tag{41}$$

6.3.2. Compressible fluid half-space

The flow Mach number is defined as $M_\mathrm{s} = U_\mathrm{s}/c_\mathrm{gas}$, c_gas being the speed of sound in the undisturbed medium, region 3 in Fig. 21. The angle β formed by the shock can be computed from geometrical considerations as a function of the deflection of the flow ϑ. For a perfect gas with specific heat ratio γ, we have (cf. for instance [17]) $c_\mathrm{gas} = \sqrt{\gamma P_\mathrm{gas}/\rho_\mathrm{gas}}$ and

$$\tan\vartheta = \frac{2\cot\beta(M_\mathrm{s}^2\sin^2\beta - 1)}{(\gamma+1)M_\mathrm{s}^2 - 2(M_\mathrm{s}^2\sin^2\beta - 1)}. \tag{42}$$

The corresponding jump in pressure ΔP_s is

$$\frac{\Delta P_\mathrm{s}}{P_\mathrm{gas}} = \frac{2\gamma}{\gamma+1}(M_\mathrm{s}^2\sin^2\beta - 1). \tag{43}$$

Continuity conditions and mechanical equilibrium at the interface couple the two media by imposing $\theta \equiv \delta$ and $P \equiv \Delta P_\mathrm{s}$. For a given U_s, Eqs. (40)–(43) form a system of nonlinear equations in the unknowns δ, β, and P.

The exact solution of Eqs. (40)–(43) can be represented as the $M_\mathrm{s} - \beta$ diagram (solid line) in Fig. 24. Along this locus, for a given incident Mach number M_s, there exists a unique shock angle β such that no reflected shock and the set of two transmitted waves (p and s) exist. The broken line on the left of the diagram is the boundary $M_\mathrm{s} = c_P/c_\mathrm{gas}$; superseismic solutions are possible only to the right of this line. In the following section, we test the coupling algorithm (with extrapolation by reflection) linking our ENO and FE solvers to find this locus of single deflections.

6.3.3. Validation

A physically meaningful superseismic coupling occurs when the acoustic impendances of the two materials are comparable, such as in the loading of metal plates with solid explosives.

For the elastic solid, we choose properties similar to that of copper, with Poisson's ratio $\nu = 0.33$, density $\rho = 8970\,\mathrm{kg/m^3}$, and Young's modulus $E = 110\times10^9\,\mathrm{Pa}$. For the fluid, we find that a high explosive model would greatly complicate this study and obscure the original purpose of the superseismic problem. We then use a perfect gas ($\gamma = 1.4$) with an artificially high density ($\rho_\mathrm{gas} = 1000\,\mathrm{kg/m^3}$) and pressure ($P_\mathrm{gas} = 2.584\times10^9\,\mathrm{Pa}$).

We consider the entire assembly, gas and "copper", to be under hydrostatic pressure initially. Since we are solving the linear elastic model of the solid, it is possible to superimpose a uniform static load without altering the dynamic response. The traction at the interface, ΔP_s, can then be computed by subtracting P_gas from the interpolated Eulerian pressure field (i.e., the undisturbed flow field applies zero pressure to the solid). At the mesh boundaries that are not in contact with the gas, the solid is unconstrained. The boundary conditions for the Eulerian solver are supersonic inflow on the left side and supersonic outflow on the right side.

The Eulerian domain is a rectangle large enough to contain the EL-interface portion of the deforming FE boundary for the duration of the simulation. The shock (moving from left to right) is initially parallel to the interface and the undeformed FE domain is a second rectangle, subdivided in quadratic triangles.

When the interface is a rigid wall, the shock maintains the given speed and orthogonality to the interface, and the solution is a simple channel flow. In this problem, the interface is not rigid, and the deformation behind the shock generates a transverse expansion wave. A transition occurs, with the front lagging at the interface, until the shock tilts by an angle $\beta - \pi/2$.

In the vicinity of the shock foot, no length scales are present and the solution is self-similar and steady in a reference frame traveling with U_s. The external boundary conditions of the two solvers and the initial location of the shock affect the solution, but their effect is small in the supersonic flow region behind the shock. The final shock inclination depends only on the deflection δ of the solid and, therefore, it is only a function of the strength of the shock and of the properties of the elastic material.

An approximately steady configuration is displayed in Fig. 22. The plot shows contour levels of shear stress in the upper Lagrangian mesh and contours of pressure in the lower Eulerian grid. The velocity vectors on the Cartesian grid are plotted in a frame traveling with the shock. Disturbances due to the initial transient appear on the left of the domain, but they do not affect the supersonic flow ahead. The initial Mach number for this case is $M_s = 4.5$; the steady value (as derived from post-shock conditions) is $M_s = 4.40 \pm 0.01$. The shock is tilted by an angle $2.77° \pm 0.07°$.

A vertical section of the Lagrangian data of the previous plot is displayed in Fig. 23 in the undeformed configuration. The components of the stress tensor are compared against the predicted profile. Eqs. (34) and (35) give angles of the p and s fronts equal to 30.62° and 14.87°, respectively.

This case is computed on a Cartesian $[-0.38, 0.38] \times [-0.35, 0.41]$ domain, with grid spacing $\Delta x_E = 0.0038\,\mathrm{m}$ (E1 mesh). The undeformed FE domain is $[-0.40, 0.40] \times [0.0, 0.08]$ with average node distance $\Delta x_L = 0.002\,\mathrm{m}$ (L1 mesh). In the main plot of Fig. 24, we compare data points from similar simulations against the M_s–β curve (solid line). The labels E2 and L2 indicate two coarser discretizations (by a factor of 2) of the Eulerian and Lagrangian domains.

Uncertainties in our results are due to the fact that δ, M_s, and β are not directly available from the numerical simulation. They are found first by evaluating the post-shock conditions from slices of the computational domain, and then by numerically solving Eqs. (41)–(43). Since the interface deflection is quite small, the main source of error is introduced when measuring the vertical velocity v_p. To assess the magnitude of the uncertainties, we refer to the standard deviation (Δ) of the post-shock grid values with respect to the sample average (indicated by an overbar). A simple error analysis derived from Eqs. (41)–(43) shows that, to order δ,

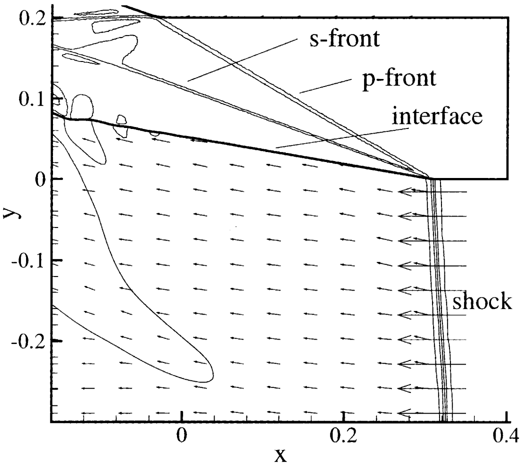

Fig. 22. Detail of the flow field behind the shock for initial Mach number 4.5. Pressure (Eulerian, bottom half) has 10 levels from $3. \times 10^9$ to $120. \times 10^9$ Pa; shear (Lagrangian, upper half) has 15 levels from $-50. \times 10^9$ to $40. \times 10^9$ Pa. The velocity vectors on the Cartesian grid are plotted in a frame traveling with the shock.

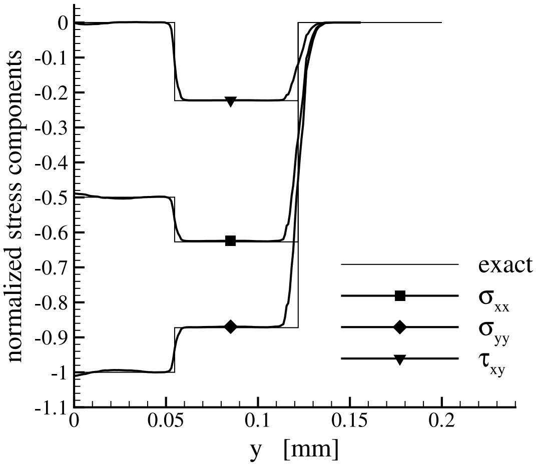

Fig. 23. Stress profiles at $x = 0.10$ m. The location of the leading edge of the deflected portion of the interface is estimated to be $x = 0.306$ m in Fig. 22. The components of the stress tensor are normalized by the magnitude of the effective applied load.

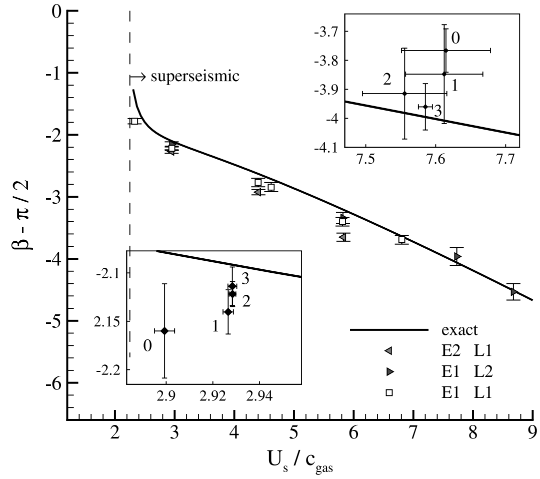

Fig. 24. Shock velocity–boundary deflection diagram for the case of superseismic load ($U_s > c_P$). The self-similar solution is plotted against the estimated values of M_s and β from different discretizations (E1, E2, L1, L2). The insets display the results of two convergence studies (from the coarsest level 0) for initial Mach numbers 3 and 8. Error bars are computed from Eqs. (44)–(46).

$$\Delta_\delta \cong \frac{\Delta_{v_p}}{\overline{v}_p}\overline{\delta}, \tag{44}$$

$$\Delta_\beta \cong \frac{(\gamma - 1)}{2}\Delta_\delta. \tag{45}$$

The uncertainty in δ can be partially corrected by measuring the slope of contour 0 of the distance function, but it cannot be neglected. The error in the estimate of M_s tends to be smaller,

$$\Delta_{M_s} \cong \frac{1}{\overline{M}_s}\frac{\gamma + 1}{4}\frac{\Delta_P}{P_{gas}}. \tag{46}$$

We conclude this section with the results of two refinement studies, for initial Mach numbers 3 and 8, also displayed in Fig. 24. In these two cases, the Cartesian domain is $[-0.38, 0.78] \times [-0.35, 0.21]$ and the undeformed FE domain is $[-0.40, 0.80] \times [0.0, 0.08]$. At level 0, $\Delta x_{\mathrm{E}} = 0.015\,\mathrm{m}$ and $\Delta x_{\mathrm{L}} = 0.016\,\mathrm{m}$; levels 1, 2, and 3 correspond to successive refinements of both meshes by a factor of 2.

The scale in both sub-plots is greatly magnified, showing the need of resolving the shock inclination within small fractions of a degree. Increasing the resolution enables us to have a greater precision by averaging over larger samples. This results in decreasing the uncertainties in our parameters, but, since the problem is self-similar, the solution at a fixed mesh point number is independent of the resolution. From Fig. 24, no conclusions can be drawn regarding the rate of convergence of the mean values to the known analytical solution. However, we verify that, within the errors estimated above, the data points are incrementally closer to the reference solution as the resolution is increased.

7. Conclusion

In certain multi-physics simulations (e.g., the cylinder test problem), it is very difficult and often impossible to treat the entire domain with either an Eulerian solver or a Lagrangian solver. In this paper, we have presented an Eulerian–Lagrangian coupling scheme for coupling these solvers and demonstrate its accuracy with a variety of test cases. This scheme is shown to have accuracy comparable to more sophisticated schemes incorporating cut cells. It is robust and a large class of problems may be solved without any one-off modifications or ad hoc tunable parameters. However, the scheme's flexibility is perhaps its greatest strength. It allows stand-alone Eulerian and Lagrangian solvers to perform fully coupled simulations with minimal addition and virtually no modifications. This flexibility is extremely important as it facilitates the development of a modular problem solving environment for multi-physics problems.

A set of benchmark exercises has also been produced. Tests that are available in the current literature for coupling algorithms have been reproduced showing that the accuracy at the boundary is at worst first-order in the grid resolution, in one and two dimensions. Also, we introduced novel verification tests to assess the performances of the coupling scheme in dynamical problems. Particularly, we proposed the transparency test to assess whether spurious oscillations exist in wave interactions when the Lagrangian and the Eulerian material have identical acoustic impedence. We also presented the superseismic loading problem as a two-dimensional test involving full coupling of a Lagrangian solid (modeled as linearly elastic and isotropic) and an Eulerian compressible gas (modeled as a perfect gas). We proved that a steady self-similar solution exists in the reference frame of a shock when the shock speed is greater than the speed of propagation of dilatational disturbances in the bulk of the solid. We also showed that numerical tests starting from a supersonic (superseismic) channel flow evolve to this steady solution and that the shock deflection angles extracted from these simulations converge to the values predicted by the theory.

Acknowledgements

We thank Dan Meiron, Julian Cummings, Ravi Samtaney, Raul Radovitzky and Michael Aivazis (all at Caltech) for their contributions to software development, numerical algorithms, and many constructive suggestions. In particular, Ravi Samtaney suggested and first demonstrated the use of reflection-type boundary conditions as an alternative to injection. We acknowledge the very substantial contributions by Ron Fedkiw (Stanford) in the initial stages of this work. He was extremely helpful in sharing his knowledge and played a key role in introducing us to the level set and ghost fluid methods through his tutorials and software. This work was carried out at Caltech ASCI ASAP Center of Excellence "Center for Simulation of Dynamic Response of Materials" and funded by Contract B341492 under DOE Contract W-7405-ENG-48.

Appendix A

In this appendix, we derive R, $\Delta\sigma$, and the Lagrangian velocity vector $\mathbf{u}$. The principal stresses in the uniform region $\alpha_S < \alpha < \alpha_P$ are

$$\overline{\sigma}_{\mathrm{I}} = \Delta\sigma$$

and

$$\overline{\sigma}_{\mathrm{II}} = \overline{\sigma}_{\mathrm{III}} = \frac{v}{1-v}\Delta\sigma.$$

The normal stress σ_n and the tangential stress σ_t with respect to the s front can be obtained from $\overline{\sigma}_{\mathrm{I}}$ and $\overline{\sigma}_{\mathrm{II}}$ through rotation of an angle $(\alpha_S - \alpha_P)$ as

$$\sigma_n = \Delta\sigma\left[\cos^2(\alpha_S - \alpha_P) + \frac{v}{1-v}\sin^2(\alpha_S - \alpha_P)\right]$$

and

$$\sigma_t = \Delta\sigma\left[\sin^2(\alpha_S - \alpha_P) + \frac{v}{1-v}\cos^2(\alpha_S - \alpha_P)\right].$$

Similarly, the principal stresses in the uniform region $\alpha < \alpha_S$ are

$$\sigma_{\mathrm{I}} = -P$$

and

$$\sigma_{\mathrm{II}} = R\sigma_{\mathrm{I}}$$

so that σ_n and σ_t can also be written as

$$\sigma_n = -P\left[\cos^2\alpha_S + R\sin^2\alpha_S\right]$$

and

$$\sigma_t = -P\left[\sin^2\alpha_S + R\cos^2\alpha_S\right].$$

Since there is no discontinuity in the normal and tangential stresses at a shear front, the above expressions for σ_t and σ_n can be equated, giving the result

$$\Delta\sigma = -\frac{P(1-v)}{N}\cos 2\alpha_S,$$

$$R = -1 + \frac{\cos 2\alpha_S}{N},$$

and

$$N = \cos^2\alpha_S + (1-2v)\cos^2(\alpha_S - \alpha_P) - 1 + v.$$

In the following, we derive the expression for the Lagrangian velocity vector $\mathbf{u} = (u_{\mathrm{p}}\ v_{\mathrm{p}})^{\mathrm{T}}$ in region 2 (refer to Fig. 21 for notation). The continuity relation for an isotropic medium can be formulated as

$$c[[\mathrm{F}\ \mathbf{n}]] + [[\mathbf{u}]] = 0.$$

The double brackets $[[\cdot]]$ denote the jump across a discontinuity of normal $\mathbf{n}$; c is the speed of the front and F the deformation gradient tensor.

For the p wave, the uniaxial strain is

$$\bar{\epsilon}_{\mathrm{I}} = \Delta\sigma/k$$

where

$$k = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)} = \frac{2(1-\nu)\mu}{(1-2\nu)}.$$

The normal to the p front is $\mathbf{n}_{\alpha_P} = (\sin\alpha_P \ \cos\alpha_P)^{\mathrm{T}}$ and the jump in the x–y coordinate system is

$$[[\mathbf{u}]]_P = -c_P \frac{\Delta\sigma}{k}\begin{pmatrix}\sin\alpha_P\\ \cos\alpha_P\end{pmatrix}.$$

At the s front, there is only a discontinuity in shear. Referring to the normal $\mathbf{n}_{\alpha_S} = (-\sin\alpha_S \ \cos\alpha_S)^{\mathrm{T}}$, the shear deformation in region 1 is

$$\bar{\gamma}_{nt} = -\frac{\Delta\sigma}{k\cos 2\alpha_S}\cos 2(\alpha_P - \alpha_S)\sin 2\alpha_{\mathrm{s}}$$

and the shear deformation in region 2 is

$$\gamma_{nt} = \frac{\Delta\sigma}{k}\sin 2(\alpha_P - \alpha_S).$$

The jump in shear deformation at the s front is

$$[[\gamma_{nt}]] = \frac{\Delta\sigma}{k\cos 2\alpha_S}\sin 2\alpha_P$$

and

$$[[F\mathbf{n}]] = [[\gamma_{nt}]]\begin{pmatrix}0\\ 1\end{pmatrix}.$$

In the x–y reference frame, this is

$$[[\mathbf{u}]]_S = c_S\frac{\Delta\sigma}{k}\frac{\sin 2\alpha_P}{\cos 2\alpha_S}\begin{pmatrix}\cos\alpha_S\\ -\sin\alpha_S\end{pmatrix}.$$

Now recall that $c_P = U_{\mathrm{s}}\sin\alpha_P$, $c_S = U_{\mathrm{s}}\sin\alpha_S$, and that region 0 is undisturbed. In linearized elasticity, we can always sum the two contributions at the s and p fronts, so that the vector $\mathbf{u}$ in the x–y reference in region 2 is

$$\mathbf{u} = U_{\mathrm{s}}(1-2\nu)\frac{P}{4\mu N}\begin{pmatrix}\cos 2\alpha_S - \cos 2(\alpha_P - \alpha_S)\\ \sin 2\alpha_P\end{pmatrix}.$$

References

[1] W.F. Noh, CEL: a time dependent two space-dimensional, coupled Eulerian Lagrangian code, in: Methods of Computational Physics, vol. 3, Academic Press, New York, 1964, p. 117.

[2] D. Kershaw, M. Prasad, M. Shaw, J. Milovich, 3D unstructured mesh ALE hydrodynamics with the upwind discontinuous finite element method, Comput. Meth. Appl. Mech. Eng. 158 (1998) 81.

[3] A. Soria, F. Casadei, Arbitrary Lagrangian–Eulerian multicomponent compressible flow with fluid–structure interaction, Int. J. Numer. Meth. Fluids 25 (1997) 1263.

[4] H.H. Hu, N.A. Patankar, M.Y. Zhu, Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian–Eulerian techique, J. Comput. Phys. 169 (2001) 427.

[5] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35.

[6] R.B. Pember, J.B. Bell, P. Colella, W.Y. Crutchfield, M.L. Welcome, An adaptive Cartesian method for unsteady compressible flow in irregular regions, J. Comput. Phys. 120 (1995) 278.

[7] J. Falcovitz, G. Alfandary, G. Hanoch, A two-dimensional conservation laws scheme for compressible flows with moving boundaries, J. Comput. Phys. 138 (1997) 83.

[8] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows, J. Comput. Phys. 152 (1999) 457.

[9] R.P. Fedkiw, Coupling an Eulerian fluid calculation to a Lagrangian solid calculation with the ghost fluid method, J. Comput. Phys. 175 (2002) 200.

[10] R.P. Fedkiw, T. Aslam, S. Xu, The ghost fluid method for deflagration and detonation discontinuities, J. Comput. Phys. 154 (1999) 393.

[11] H. Forrer, M. Berger, Flow simulations on Cartesian grids involving complex moving geometries flows, Int. Ser. Numer. Math., 129, Birkhäuser, Basel 1 (1998) 315.

[12] H. Forrer, R. Jeltsch, A higher-order boundary treatment for Cartesian-grid methods, J. Comput. Phys. 140 (1998) 259.

[13] W. Mulder, S. Osher, J.A. Sethian, Computing interface motion in compressible gas dynamics, J. Comput. Phys. 100 (1992) 209.

[14] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146.

[15] M. Aivazis, B. Goddard, D. Meiron, J.C.T. Pool, J.E. Shepherd, A virtual test facility for simulating the dynamic response of materials, Comput. Sci. Eng. 2 (2000) 42.

[16] L. Hill, Detonation product equation-of-state directly from the cylinder test, in: 21st International Symposium on Shock Waves, Great Keppel Island, Australia, July 20–25, 1997.

[17] P. Thompson, Compressible Fluid Dynamics, McGraw-Hill, New York, 1972.

[18] R. Radovitzky, M. Ortiz, Error estimation and adaptive meshing in strongly nonlinear dynamic problems, Comput. Meth. Appl. Mech. Eng. 172 (1999) 203.

[19] A.E.H. Love, A Treatise on the Mathematical Theory of Elasticity, Dover Publications, New York, 1944.

[20] W. Young, Roark's Formulas for Stress and Strain, McGraw-Hill, New York, 1989.

[21] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method, vol. 1, McGraw-Hill, New York, 1989.

[22] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method, vol. 2, McGraw-Hill, New York, 1991.

[23] K. Bathe, Finite Element Procedures, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[24] J. von Neumann, R.D. Richtmyer, A method for the numerical calculation of hydrodynamic shocks, J. Appl. Phys. 21 (1950) 232.

[25] D. Benson, Computational methods in Lagrangian and Eulerian hydrocodes, Comput. Meth. Appl. Mech. Eng. 99 (1992) 235.

[26] M.L. Wilkins, Use of artificial viscosity in multidimensional fluid dynamic calculations, J. Comput. Phys. 36 (1980) 281.

[27] W.F. Noh, Errors for calculations of strong shocks using an artificial viscosity and an artificial heat flux, J. Comput. Phys. 72 (1978) 78.

[28] E.J. Caramana, M.J. Shashkov, P.P. Whalen, Formulations of artificial viscosity for multi-dimensional shock wave computations, J. Comput. Phys. 144 (1988) 70.

[29] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed – algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12.

[30] C.M. Hoffmann, Geometric and Solid Modeling, An Introduction, Morgan Kaufmann, Los Altos, CA, 1989.

[31] S. Mauch, A fast algorithm for computing the closest point and distance transform, SIAM J. Scientific Comput. (in review) (2000) http://www.cacr.caltech.edu/ASAP/onlineresources/publications/cit-asci-tr/c = it-asci-tr077.pdf.

[32] S.C. Chapra, R.P. Canale, Numerical Methods for Engineers, McGraw-Hill, New York, 1990.

[33] R. Caiden, R.P. Fedkiw, C. Anderson, A numerical method for two-phase flow consisting of separate compressible and incompressible regions, J. Comput. Phys. 166 (2001) 1.

[34] D. Adalsteinsson, J.A. Sethian, The fast construction of extension velocities in level set methods, J. Comput. Phys. 148 (1999) 2.

[35] T.D. Aslam, J.B. Bdzil, D.S. Stewart, Level set methods applied to modeling detonation shock dynamics, J. Comput. Phys. 126 (1996) 390.

[36] S. Osher, C.W. Shu, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, J. Comput. Phys. 83 (1989) 32.

[37] C.W. Shu, S. Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, J. Comput. Phys. 77 (1988) 439.

[38] B. van Leer, Flux-vector splitting for the Euler equations, Lecture Notes in Physics 170 (1982) 507.

[39] H.H. Bleich, A. Matthews, J.P. Wright, Step load moving with superseismic velocity on the surface of a half-space of granular material, Technical Report AFWL-TR-65-59, Air Force Weapons Laboratory, 1965.

[40] H. Kolsky, Stress Waves in Solids, Dover, New York, 1963.