Verification Assessment of Piston Boundary Conditions for Lagrangian Simulation of Compressible Flow Similarity Solutions

Scott D. Ramsey[1]
Mem. ASME
Los Alamos National Laboratory,
P.O. Box 1663, MS T082,
Los Alamos, NM 87545
e-mail: ramsey@lanl.gov

Philip R. Ivancic
Department of Aerospace Engineering,
Mississippi State University,
330 Walker Engineering Laboratory,
Mississippi State, MS 39762
e-mail: pri4@msstate.edu

Jennifer F. Lilieholm
Department of Physics and Astronomy,
University of Maine,
2 Bennett Hall,
Orono, ME 04469
e-mail: Jennifer.Lilieholm@maine.edu

This work is concerned with the use of similarity solutions of the compressible flow equations as benchmarks or verification test problems for finite-volume compressible flow simulation software. In practice, this effort can be complicated by the infinite spatial/temporal extent of many candidate solutions or "test problems." Methods can be devised with the intention of ameliorating this inconsistency with the finite nature of computational simulation; the exact strategy will depend on the code and problem archetypes under investigation. For example, self-similar shock wave propagation can be represented in Lagrangian compressible flow simulations as rigid boundary-driven flow, even if no such "piston" is present in the counterpart mathematical similarity solution. The purpose of this work is to investigate in detail the methodology of representing self-similar shock wave propagation as a piston-driven flow in the context of various test problems featuring simple closed-form solutions of infinite spatial/temporal extent. The closed-form solutions allow for the derivation of similarly closed-form piston boundary conditions (BCs) for use in Lagrangian compressible flow solvers. The consequences of utilizing these BCs (as opposed to directly initializing the self-similar solution in a computational spatial grid) are investigated in terms of common code verification analysis metrics (e.g., shock strength/position errors and global convergence rates).
[DOI: 10.1115/1.4030929]

1 Introduction

The one-dimensional (1D) compressible flow equations for a polytropic gas admit numerous similarity solutions: these are derived using scaling or other invariances of the governing equations, as determined from dimensional analysis or Lie group techniques (as discussed by Sedov [1], Barenblatt [2], Axford [3], Zel'dovich and Raizer [4], and Ramsey et al. [5], among many others). Individual similarity solutions are characterized by distinct flow patterns that vary under space and time evolution only according to scale; as a result these flows are often called "self-similar." These solutions are also notable for their independence of dimensional units, and for typically being defined over at least semi-infinite spatial and temporal extents.

Similarity solutions often have closed-form or semi-analytical mathematical representations, making them potentially useful as, according to Coggeshall [6],

> "(i) exact solutions pertaining to certain initial/boundary conditions, (ii) benchmark solutions for numerical codes, (iii) ideal solutions with special properties, and (iv) insight into more general flow behaviors."

This work is related to item (ii) above: the use of similarity solutions as benchmarks or verification test problems for compressible flow simulation software.

Code verification analysis (reviewed, e.g., by Oberkampf et al. [7] and Roy [8]) exemplifies the process of determining that a model implementation (e.g., a software instantiation of an algorithm for approximate numerical solution of partial differential equations) accurately represents the developer's conceptual description of and solution to the model. This process complements that of validation analysis, or determining the degree to which a model is an accurate representation of reality from the perspective of the intended uses of the model (and, thus, necessarily involves the comparison of simulation results with experimental data).

A formal code verification analysis may in practice be employed in the interest of addressing the stability, computational cost, relative accuracy, or other properties of physics simulation software or a specific set of algorithms implemented within it. In the context of this work, code verification analysis will be used to assess and compare methods of initializing similarity solutions in finite-volume compressible flow simulation software. In practice, this effort can be complicated by the infinite spatial/temporal extent of many candidate solutions or test problems. Methods can be devised with the intention of reconciling this inconsistency with the finite nature of computational simulation; the exact strategy will depend on the code and problem archetypes under investigation.

For example, self-similar shock wave propagation can be represented in Lagrangian compressible flow simulations as rigid boundary-driven flow, even if no such piston is present in the counterpart mathematical similarity solution. The purpose of this work is to investigate in rigorous detail the methodology of representing self-similar shock wave propagation as a piston-driven flow in the context of various test problems featuring simple closed-form mathematical solutions of infinite spatial/temporal extent. The closed-form mathematical solutions allow for the

[1]Corresponding author.

Manuscript received January 12, 2015; final manuscript received May 20, 2015; published online December 10, 2015. Assoc. Editor: William Rider.

derivation of similarly closed-form piston BCs for use in Lagrangian compressible flow solvers. The consequences of utilizing these BCs (as opposed to explicitly initializing the self-similar solution in a computational spatial grid) are investigated in terms of common code verification analysis metrics (e.g., shock strength/position errors and global convergence rates).

In support of this goal, the structure of this article is as follows: Section 2 provides a brief review of the governing compressible flow equations, Rankine–Hugoniot conditions for discontinuous shock waves, and three closed-form similarity flows that solve these equations. Section 3 details two initialization methods for these and other arbitrary similarity solutions in Lagrangian compressible flow solvers. Comparison of compressible flow simulation results for these test problems (for both initialization methods) with exact solutions is presented in Sec. 4, including quantitative verification analysis. A discussion of results, associated conclusions, and recommendations for future study are provided in Sec. 5.

2 Governing Equations

The work that follows involves similarity solutions of the 1D symmetric compressible flow equations

$$\frac{\partial \rho}{\partial t}+\frac{\partial(\rho u)}{\partial r}+\frac{k\rho u}{r}=0 \tag{1}$$

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial r}+\frac{1}{\rho}\frac{\partial P}{\partial r}=0 \tag{2}$$

$$\frac{\partial \ln(P\rho^{-\gamma})}{\partial t}+u\frac{\partial \ln(P\rho^{-\gamma})}{\partial r}=0 \tag{3}$$

where $\rho(r,t)$, $u(r,t)$, and $P(r,t)$ denote the fluid mass density, radial flow velocity, and pressure, respectively, and k is the space index ($k=0$, 1, or 2 for 1D planar, cylindrical, and spherical symmetries, respectively). Implicit in Eqs. (1)–(3) is the assumption of a polytropic gas equation of state

$$P=(\gamma-1)\rho e \tag{4}$$

where $\gamma>1$ is the adiabatic index and $e(r,t)$ is the fluid specific internal energy.

Discontinuous shock solutions are admitted by the above system, in that they satisfy Eqs. (1)–(3) at every location except the shock itself. In this case, conservation of mass, momentum, and total energy across the discontinuity is ensured through the introduction of the Rankine–Hugoniot or "shock jump" conditions

$$(u_2-D)\rho_2=(u_1-D)\rho_1=m \tag{5}$$

$$P_2+m\,u_2=P_1+m\,u_1 \tag{6}$$

$$e_2+\frac{P_2}{\rho_2}+\frac{1}{2}(u_2-D)^2=e_1+\frac{P_1}{\rho_1}+\frac{1}{2}(u_1-D)^2 \tag{7}$$

where the subscripts 1 and 2 denote, respectively, the unshocked and shocked states immediately adjacent to the shock, m is a mass flux, and D is the shock velocity. For a strong shock, $P_2\gg P_1$ and $e_2+P_2/\rho_2\gg e_1+P_1/\rho_1$, and with Eq. (4), Eqs. (5)–(7) become to leading order

$$\rho_2=\frac{\gamma+1}{\gamma-1}\rho_1 \tag{8}$$

$$P_2=\frac{2\rho_1(D-u_1)^2}{\gamma+1} \tag{9}$$

$$u_2=\frac{2D+(\gamma-1)u_1}{\gamma+1} \tag{10}$$

Equations (8)–(10) will be exclusively used for flows with shocks throughout the remainder of this work.

2.1 Closed-Form Similarity Flows.

Equations (1)–(3) admit numerous closed-form and semi-analytic similarity solutions with and without shocks: these are derived using scaling or other invariances, as determined from dimensional analysis or Lie group techniques. Similarity solutions of Eqs. (1)–(3) transform under space and time evolution only through scaling relations; they are thus referred to as self-similar or "scale invariant," as they are defined under any self-consistent choice of physical units. Moreover, aside from the Rankine–Hugoniot conditions given by Eqs. (8)–(10) and a symmetry condition at $r=0$ for $k=1$ or 2, similarity solutions of Eqs. (1)–(3) often do not feature explicit BCs; they will be at least semi-infinite in both time and space (that is, they will be defined for at least $0\le t<\infty$ and $0\le r<\infty$, and in some cases, $-\infty<t<\infty$ and/or $-\infty<r<\infty$).

Of the many candidate similarity flows involving strong shocks, three are considered in this work: the Zel'dovich problem, Coggeshall problem 19, and Coggeshall problem 20.

2.1.1 Zel'dovich. The Zel'dovich problem was formulated by Zel'dovich in 1956 as the "motion of a gas under the action of an impulsive load," [4,9] and is a 1D planar configuration consisting of a half-space ($x>0$) containing a uniform density, quiescent, zero-pressure/specific internal energy polytropic gas. A surface at $x=0$ separates the half-space containing the gas from vacuum for $x<0$.

As described in detail by Zel'dovich and Raizer [4], a right-moving strong shock is created in this configuration by applying at $t=0$ an instantaneous "pressure pulse" to the gas–vacuum interface. The gas–vacuum boundary is simultaneously removed, so that the gas behind the right-moving shock expands leftward into the vacuum.

A closed-form similarity solution to this problem can be constructed for $k=0$ and $\gamma=7/5$ and is given by Eqs. (11)–(17), where $x=x_s$ is the shock location.

Unshocked region ($t>0$, $x_s<x<\infty$)

$$\rho=\rho_0 \tag{11}$$

$$u=0 \tag{12}$$

$$P=0 \tag{13}$$

Shocked region ($t>0$, $-\infty<x\le x_s$)

$$\rho=6\rho_0\left(5-\frac{4x}{at^{3/5}}\right)^{-5/2} \tag{14}$$

$$u=\frac{a}{2}t^{-2/5}\left[\frac{2x}{at^{3/5}}-1\right] \tag{15}$$

$$P=\frac{3}{10}a^2t^{-4/5}\left[5-\frac{4x}{at^{3/5}}\right]^{-3/2} \tag{16}$$

Shock speed

$$D=\frac{3}{5}at^{-2/5} \tag{17}$$

In Eqs. (11)–(17), $\rho_0>0$ and $a>0$ are otherwise arbitrary constants. The shock position x_s is determined by the shock speed.

2.1.2 Coggeshall 19. In 1991, Coggeshall [6] used Lie group methods to derive a collection of 22 closed-form similarity solutions to a variant of the compressible flow equations including an

optional heat conduction term in Eq. (3). In this set, solutions 1–7 are smooth flows without conduction, solutions 8–18 are smooth flows with conduction, and solutions 19–22 are shock flows without conduction.

Coggeshall solution 19 (Cog19) is more widely known as "Noh's constant-velocity shock problem" [3,10–12] (or more simply the "Noh problem") and describes a strong stagnation shock generated at $t=0$ by a zero-pressure/specific internal energy, constant-velocity inflow of polytropic gas against a rigid boundary. A closed-form similarity solution to this problem can be constructed for $k=0$, 1, or 2 and arbitrary $\gamma>1$ and is given by Eqs. (18)–(24), where $r=r_s$ is the shock location.

Unshocked region ($t>0$, $r_s<r<\infty$)

$$\rho=\rho_0\left(\frac{r-u_0t}{r}\right)^k \tag{18}$$

$$u=u_0 \tag{19}$$

$$P=0 \tag{20}$$

Shocked region ($t>0$, $0\le r\le r_s$)

$$\rho=\rho_0\left(\frac{\gamma+1}{\gamma-1}\right)^{k+1} \tag{21}$$

$$u=0 \tag{22}$$

$$P=\frac{\rho_0u_0^2(\gamma+1)^{k+1}}{2(\gamma-1)^k} \tag{23}$$

Shock speed

$$D=-\frac{(\gamma-1)u_0}{2} \tag{24}$$

In Eqs. (18)–(24), $\rho_0>0$ and $u_0<0$ are otherwise arbitrary constants. The shock position r_s is determined by the shock speed.

2.1.3 Coggeshall 20. Included in Coggeshall's [6] Lie group analysis of Eqs. (1)–(3) are methods for extending existing solutions of those equations using global invariance transformations. For example, Coggeshall solution 20 (Cog20) was generated from Cog19 via projective group invariance of Eqs. (1)–(3) in the r–t plane, under the necessary condition

$$\gamma=\frac{k+3}{k+1} \tag{25}$$

If Eq. (25) is satisfied for the given geometry, the Cog20 solution can be constructed from Cog19 and describes a strong shock generated by a zero-pressure/specific internal energy, spatially varying inflow of polytropic gas against a rigid boundary. A closed-form solution to this problem is given by Eqs. (26)–(32), where $r=r_s$ is the shock position.

Unshocked region ($t>0$, $r_s<r<\infty$)

$$\rho=\rho_0\left(\frac{r-u_0t}{r}\right)^k(1-at)^{-k-1} \tag{26}$$

$$u=\frac{u_0-ar}{1-at} \tag{27}$$

$$P=0 \tag{28}$$

Shocked region ($t>0$, $0\le r\le r_s$)

$$\rho=\rho_0\left(\frac{\gamma+1}{\gamma-1}\right)^{k+1}(1-at)^{-k-1} \tag{29}$$

$$u=\frac{ar}{at-1} \tag{30}$$

$$P=\frac{\rho_0u_0^2(\gamma+1)^{k+1}}{2(\gamma-1)^k}(1-at)^{-k-3} \tag{31}$$

Shock speed

$$D=-\frac{(\gamma-1)u_0}{2} \tag{32}$$

In Eqs. (26)–(32), $\rho_0>0$, $u_0<0$, and $a<0$ are otherwise arbitrary constants. The shock speed appearing in Eq. (32) is different than that originally given by Coggeshall [6]; see the Appendix for more details. The shock position r_s is determined by the shock speed.

3 Computational Initialization of Similarity Flows

The similarity solutions outlined in Sec. 2 are three examples of potential test problems that can be used for verification of software intended to produce approximate numerical solutions of Eqs. (1)–(3). As discussed in Sec. 1, this code verification process may be employed in the interest of addressing the stability, computational cost, relative accuracy, or other properties of physics simulation software or a specific subset of algorithms implemented within it. Conclusions drawn from simulation of simple test problems with known solutions can both qualitatively and quantitatively reinforce confidence in the credibility of production software within its intended scope (i.e., simulation of more complicated problems without known solutions).

Despite their obvious utility and apparent simplicity, similarity solutions, such as those discussed in Sec. 2, can prove subtle to instantiate as test problems within a finite-volume compressible flow solver due to their infinite temporal/spatial extent (e.g., Eqs. (11)–(16), (18)–(23), and (26)–(31) are all defined over $0<t<\infty$ and $0\le r<\infty$ at least). Methods can be devised with the intention of reconciling this inconsistency with the finite nature of computational simulation; the exact strategy will depend on the code and problem archetypes under investigation.

For example, self-similar shock wave propagation (such as that modeled for various problem configurations in Sec. 2) can be represented in Lagrangian compressible flow code simulations as rigid boundary-driven flow, even if no such piston is present in the counterpart mathematical similarity solution. The purpose of this work is to investigate in rigorous detail the methodology of representing self-similar shock wave propagation as a piston-driven flow in the context of the Zel'dovich, Cog19, and Cog20 similarity solutions. This methodology will be quantitatively compared with a more direct means of initializing the same test problems in the same Lagrangian compressible flow solver.

3.1 Direct Method. The direct method of initializing similarity solutions within a finite-volume compressible flow solver is conceptually simple, as depicted in Fig. 1. An "initialization time" $t=t_i$ is arbitrarily chosen so that a space and time-dependent similarity solution of Eqs. (1)–(3) collapses to a collection of continuous space-dependent density, velocity, pressure, and specific internal energy profiles. These profiles are then sampled zone-by-zone onto a prescribed computational spatial grid via weighted averaging.

The direct method also requires that BCs be prescribed for all t, though most similarity solutions of Eqs. (1)–(3) are defined up to

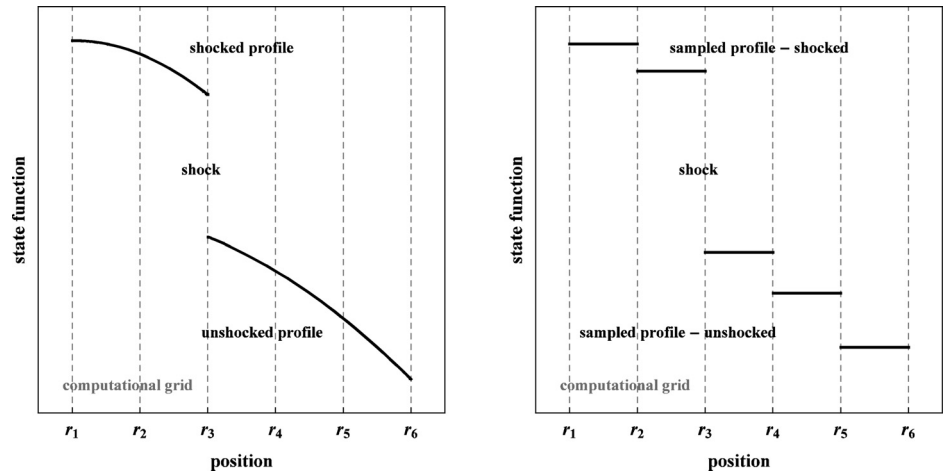

Fig. 1 Direct initialization of a similarity solution onto a computational spatial grid

$r \to \infty$, as seen in Sec. 2. In this case, the similarity solution must be artificially truncated at an arbitrary boundary point $r = r_{b,i}$ within the computational spatial grid, and a condition imposed there.[2]

For a Lagrangian compressible flow solver, the boundary at initial position $r_{b,i}$ will move as time progresses; this motion must correspond to the similarity solution being simulated. From the definition of the Lagrangian representation

$$\frac{dr_b(t)}{dt} = u[r_b(t), t] \tag{33}$$

where $u(r,t)$ is the known velocity field for the similarity solution. Equation (33) is a first-order ordinary differential equation (ODE) that can be solved for $r_b(t)$ with the initial condition

$$r_b(t = t_i) = r_{b,i} \tag{34}$$

Once $r_b(t)$ is known, its velocity and acceleration as functions of time can be found through time-differentiation, and then used as BCs for a Lagrangian compressible flow solver.

In addition to being consistent with the simulation of a similarity solution, this time-dependent boundary position approach suggests an alternative method for initializing self-similar shock problems.

3.2 Piston Method. In the Lagrangian representation of compressible flow, spatial boundaries can in principle move as time progresses. The instantaneous motion of a boundary can also create shock waves, as discussed for such rigid pistons by Landau and Lifshitz [13], Sachdev [14], and Lazarus [15] (among many others).

As in Sec. 3.1, consider at some initialization time $t = t_i$ an unshocked region similarity solution with continuous spatial profiles $\rho_1(r,t_i)$, $u_1(r,t_i)$, and $P_1(r,t_i)$. This region is artificially truncated by a rigid boundary at an arbitrary position $r = r_{b,i}$ within, as depicted in Fig. 2. The boundary immediately begins moving into the unshocked region with velocity $u_b(t)$. If $u_b(t)$ is given by

$$u_b(t) = \frac{dr_b(t)}{dt} = u_2[r_b(t), t] \tag{35}$$

then $u_b(t)$ will generate a shock consistent with the similarity solution composed of $u_1(r,t)$ and $u_2(r,t)$ if

$$r_{b,i} = r_s(t = t_i) = r_{s,i} \tag{36}$$

where $r_s(t)$ is the position of the shock separating the unshocked and shocked regions.

In this configuration, the rigid boundary/piston immediately begins moving into the unshocked region, and a shock propagates ahead of it. A shocked region with the corresponding shocked state is generated and supported by the subsequent boundary/piston motion.

If $u_2(r,t) = 0$, then $u_b(t) = 0$ by inspection of Eq. (35). No finite piston motion therefore results; accordingly, stagnation shock similarity solutions (such as Cog19) are not immediately compatible with the piston initialization method. However, any similarity solution with $u_2(r,t) > 0$ (for $t \geq t_i$) is amenable to initialization via this method, as will be demonstrated in Secs. 3.2.1–3.2.3.

3.2.1 Zel'dovich. Unshocked region ($t > 0$, $r_s < r < \infty$)

$$\rho = \rho_0 \tag{37}$$

$$u = 0 \tag{38}$$

$$P = 0 \tag{39}$$

ODE for boundary/piston location

$$\frac{dr_b(t)}{dt} = \frac{a}{2} t^{-2/5} \left[\frac{2r_b(t)}{at^{3/5}} - 1 \right], \quad r_b(t_i) = at_i^{3/5} \tag{40}$$

Boundary/piston location

$$r_b(t) = \frac{a\left(5t^{3/5} - t_i^{-2/5} t\right)}{4} \tag{41}$$

In Eqs. (37)–(41), $\rho_0 > 0$, $a > 0$, and $t_i > 0$ are otherwise arbitrary constants.

3.2.2 Coggeshall 19. Cog19 is a stagnation shock problem, so as discussed in Sec. 3.2 it is not immediately amenable to the piston initialization method.

However, under the assumption of 1D planar symmetry ($k = 0$), Eqs. (1)–(3) and their associated similarity solutions (such as 1D planar Cog19) are invariant under a Galilean velocity boost $u \to u + a$, as discussed by Coggeshall [6]. This nonstagnation version of Cog19 is compatible with the piston initialization method.

[2] A second boundary point and condition for 1D planar simulations may also be needed when a solution is defined for $r < 0$; 1D cylindrical and spherical simulations must include the necessary symmetry BC at $r = 0$.

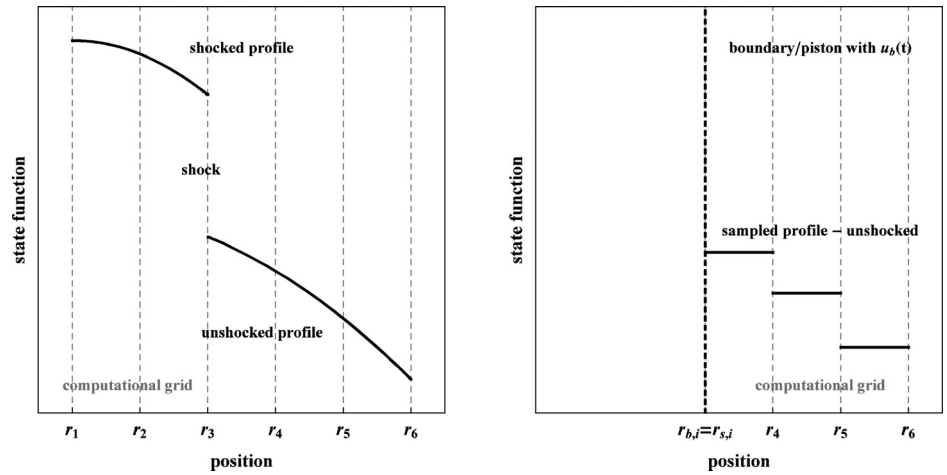

Fig. 2 Piston initialization of a similarity solution onto a computational spatial grid

Unshocked region ($t>0$, $r_s<r<\infty$)

$$\rho = \rho_0\left(\frac{r-u_0 t}{r}\right)^k \tag{42}$$

$$u = u_0 + a \tag{43}$$

$$P = 0 \tag{44}$$

ODE for boundary/piston location

$$\frac{dr_b(t)}{dt} = a, \quad r_b(t_i) = \left[a - \frac{(\gamma-1)u_0}{2}\right] t_i \tag{45}$$

Boundary/piston location

$$r_b(t) = at - \frac{(\gamma-1)u_0 t_i}{2} \tag{46}$$

In Eqs. (42)–(46), $\rho_0>0$, $a\neq 0$, $u_0<0$, $\gamma>1$, and $t_i>0$ are otherwise arbitrary constants.

3.2.3 Coggeshall 20. Since the projective transformation from Cog19 to Cog20 results in a nonzero velocity in the shocked region (given by Eq. (31)), this problem is immediately amenable to the piston initialization method.

Unshocked region ($t>0$, $r_s<r<\infty$)

$$\rho = \rho_0\left(\frac{r-u_0 t}{r}\right)^k (1-at)^{-k-1} \tag{47}$$

$$u = \frac{u_0 - ar}{1-at} \tag{48}$$

$$P = 0 \tag{49}$$

ODE for boundary/piston location

$$\frac{dr_b(t)}{dt} = \frac{ar_b(t)}{at-1}, \quad r_b(t_i) = -\frac{(\gamma-1)u_0 t_i}{2} \tag{50}$$

Boundary/piston location

$$r_b(t) = \frac{(\gamma-1)u_0 t_i(at-1)}{2(1-at_i)} \tag{51}$$

In Eqs. (47)–(51), $\gamma=(k+3)/(k+1)$, and $\rho_0>0$, $u_0<0$, $a<0$, and $t_i>0$ are otherwise arbitrary constants.

4 Simulation Results

With closed-form results for the Zel'dovich, modified Cog19, and Cog20 test problems available, counterpart numerical simulations have been performed (for both initialization methods) using the FLAG code [16]. This algorithm uses 1D, 2D, or 3D finite-volume staggered-grid hydrodynamics solvers in conjunction with various artificial viscosity and ancillary grid stability methods, as discussed by Burton [16] and Caramana et al. [17,18]. The FLAG code has previously been subject to numerous quantitative code verification studies, as detailed among others by Morgan [19] and Morgan et al. [20]. The closed-form similarity solutions for each test problem enable quantitative code verification analysis of this compressible flow algorithm under the two initial condition archetypes discussed in Sec. 3.

Quantitative code verification analysis involves the establishment of local or global error metrics for direct mathematical comparison of exact and computed solutions. Local error metrics considered in this work are differences at a given time in the position r_s and magnitude M_s (as reflected by the density ratio) of a shock wave.

A global error metric considered in this work is the weighted spatial L_1 error norm computed at a chosen time

$$\|y^E - y^C\|_1 \approx \frac{\sum_{i=1}^{N} w_{zi}|y_i^E - y_i^C|}{\sum_{i=1}^{N} w_i} \tag{52}$$

In Eq. (52), N denotes the total number of zones over which the solution is computed, y_i denotes either the exact (superscript E) or computed (superscript C) flow variable solution within zone i, and w_i is a weighting function computed over zone i (e.g., cell volume). When used in conjunction with several simulations performed at different resolutions, this global metric thus provides an example of the exchange between resolution and global accuracy.

Furthermore, taking as axiomatic the standard error ansatz [7,8] (with convergence coefficient A, convergence rate B, and characteristic grid dimension Δr)

$$\|y^E - y^C\|_1 = A(\Delta r)^B \tag{53}$$

quantitative verification analysis can also be used to establish whether or not a series of calculations convergences under mesh refinement to an exact solution—and if so, at an observed order B that can be compared with the formal order of accuracy expected for the software, methods, and test problem under consideration.

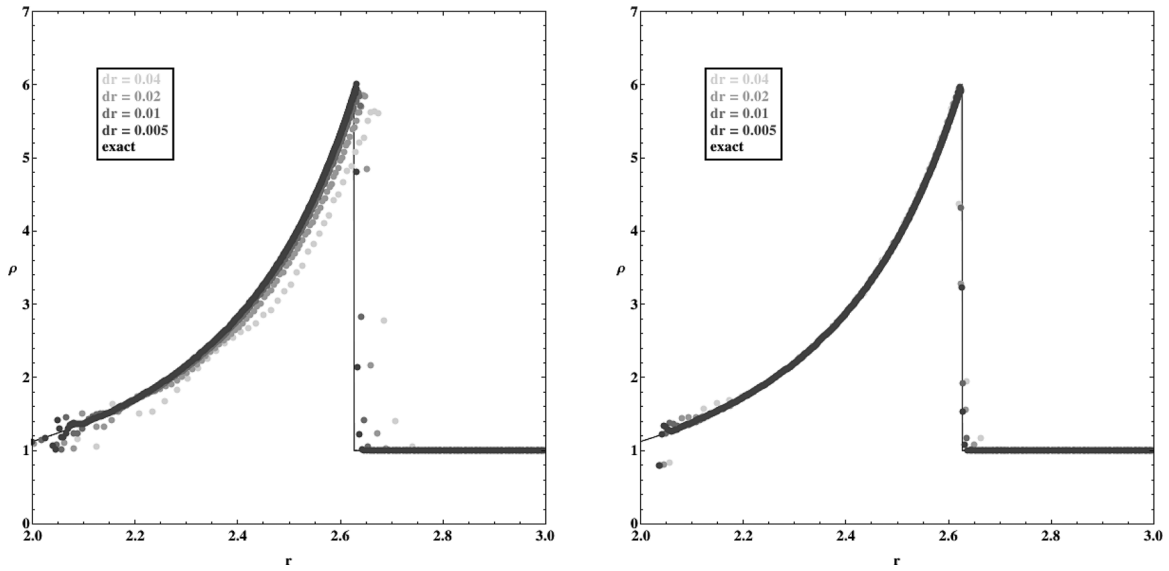

Fig. 3 One-dimensional planar Zel'dovich problem density simulation results for direct (left) and piston (right) initialization methods

Table 1 Shock location/magnitude for 1D planar Zel'dovich problem simulations: direct initialization method

	$\Delta x = 0.04$	$\Delta x = 0.02$	$\Delta x = 0.01$	$\Delta x = 0.005$	Exact
r_s	2.675	2.645	2.635	2.633	2.627
M_s	5.601	5.855	5.890	5.899	6.0

Table 2 Shock location/magnitude for 1D planar Zel'dovich problem simulations: piston initialization method

	$\Delta x = 0.04$	$\Delta x = 0.02$	$\Delta x = 0.01$	$\Delta x = 0.005$	Exact
r_s	2.613	2.620	2.623	2.626	2.627
M_s	5.783	5.890	5.964	5.930	6.0

In this work, A and B are established by four simulations at different initial spatial resolutions Δr, for a given test problem/initialization strategy. The resulting Δr–L_1 error series is fit by Eq. (53). According to Banks et al. [21], the convergence rate for an otherwise smooth, inviscid flow containing a shock wave is estimated as $B \sim p/(p+1)$, where p is the nominal convergence rate for smooth flows. In the limit $p \to \infty$, $B \to 1$. Additional details regarding convergence of flows containing shock waves are provided by Majda and Ralston [22].

4.1 Zel'dovich—1D Planar. One-dimensional planar simulation results for the Zel'dovich problem appear in Fig. 3 and Tables 1–4 for the parameterization $k = 0$, $\gamma = 7/5$, $\rho_0 = 1$, $a = 1$, and $t_i = 1$. In each case, the problem is simulated with a constant time step $\Delta t = 1.0 \times 10^{-5}$ to a final time $t_f = 5$ (boundary/piston location $x_b = 2.033$, where applicable), where analysis is performed.

With the direct method of initialization, artificial boundaries are included at $x_{L,i} = 0$ and $x_{R,i} = 3$. Following from Eqs. (15), (33), and (34), the time-dependent positions of these boundaries are

$$x_L(t) = \frac{5\left(t - t^{3/5}\right)}{4} \tag{54}$$

Table 3 Volume-weighted spatial L_1 error and associated convergence data for 1D planar Zel'dovich problem simulations: direct initialization method

	$\Delta x = 0.04$	$\Delta x = 0.02$	$\Delta x = 0.01$	$\Delta x = 0.005$	A	B
ρ	1.12	6.25×10^{-1}	3.33×10^{-1}	1.71×10^{-1}	20.9	0.903
u	5.18×10^{-2}	2.73×10^{-2}	1.40×10^{-2}	7.08×10^{-3}	1.15	0.959
P	1.47×10^{-2}	7.99×10^{-3}	4.17×10^{-3}	2.11×10^{-3}	0.305	0.936
e	9.10×10^{-3}	4.70×10^{-3}	2.43×10^{-3}	1.22×10^{-3}	0.205	0.965

Table 4 Volume-weighted spatial L_1 error and associated convergence data for 1D planar Zel'dovich problem simulations: piston initialization method

	$\Delta x = 0.04$	$\Delta x = 0.02$	$\Delta x = 0.01$	$\Delta x = 0.005$	A	B
ρ	8.29×10^{-2}	3.73×10^{-2}	2.31×10^{-2}	9.86×10^{-3}	2.00	0.994
u	7.30×10^{-3}	3.66×10^{-3}	1.78×10^{-3}	9.19×10^{-4}	0.183	1.00
P	1.16×10^{-3}	5.37×10^{-4}	3.04×10^{-4}	2.76×10^{-4}	0.030	1.02
e	1.66×10^{-3}	7.85×10^{-4}	4.62×10^{-4}	2.20×10^{-4}	0.035	0.953

$$x_R(t) = \frac{5t^{3/5} + 7t}{4} \tag{55}$$

With the piston method of initialization, the boundary initially at $x_{R,i}$ is retained, as is its subsequent motion given by Eq. (55). In this case, however, the boundary initially at $x_{L,i}$ is replaced with the boundary/piston with time-dependent position given by

$$x_b(t) = \frac{5t^{3/5} - t}{4} \tag{56}$$

In Tables 1–4, the Δx values indicate the initially uniform spatial grid zone size corresponding to a given simulation.

4.2 Modified Coggeshall 19—Quasi-2D Planar. As an alternative to an additional 1D planar simulation, the Cog19 similarity solution (modified to include the Galilean velocity boost

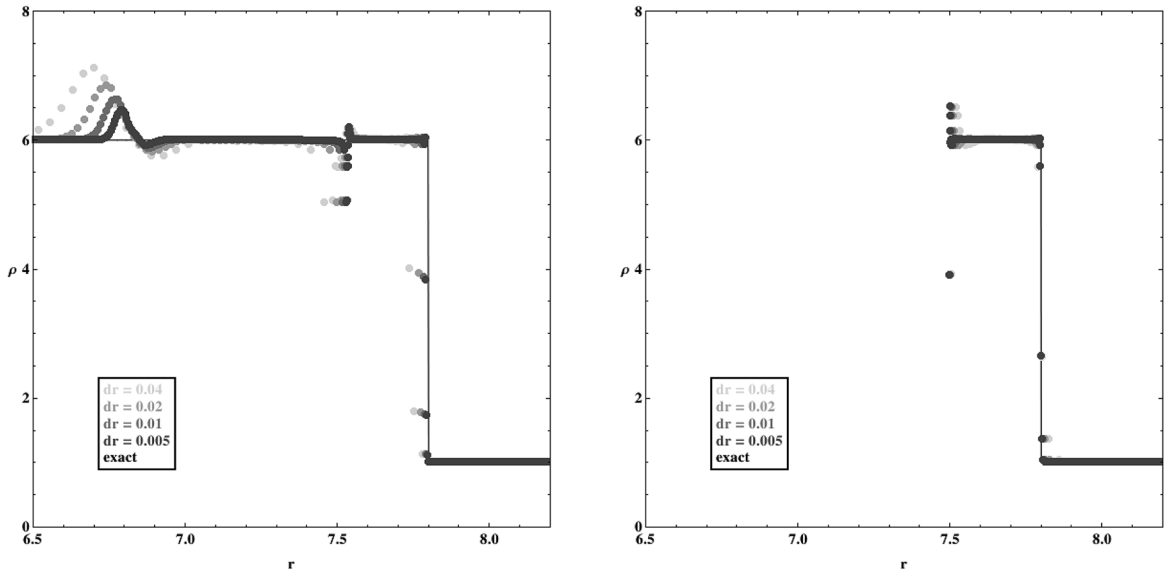

Fig. 4 Two-dimensional planar modified Cog19 problem density simulation results for direct (left) and piston (right) initialization methods

Table 5 Shock location/magnitude for 2D planar modified Cog19 problem simulations: direct initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	Exact
r_s	7.693	7.746	7.773	7.786	7.8
M_s	5.946	5.951	5.960	5.968	6.0

Table 6 Shock location/magnitude for 2D planar modified Cog19 problem simulations: piston initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	Exact
r_s	7.804	7.802	7.801	7.801	7.8
M_s	6.019	6.018	6.019	6.018	6.0

$u \to u + a$) is modeled as a quasi-2D planar test problem on a uniform, square grid bounded "vertically" between $y = y_{\min} = 0$ and $y = y_{\max} = 1$. The y-boundaries are constrained to be motionless in the y-direction, but are unconstrained in the r-direction. For each initialization method, spatial profiles and r-boundary/piston motion vary in r as given in Secs. 2.1.2 or 3.2.2, but do not vary in y.

Quasi-2D planar simulation results for the modified Cog19 problem appear in Fig. 4 and Tables 5–8 for the parameterization $k = 0$, $\gamma = 7/5$, $\rho_0 = 1$, $a = 5$, and $t_i = 0$. In each case, the problem is simulated with a constant time step $\Delta t = 1.0 \times 10^{-5}$ to a final time $t_f = 1.5$ (boundary/piston location $r_b = 7.5$, where applicable), where analysis is performed.

With the direct method of initialization, artificial r-boundaries are included at $r_{L,i} = 0$ and $r_{R,i} = 3$. Following from Eqs. (22) (with the Galilean velocity boost $u \to u + a$ included), (33), and (34), the time-dependent positions of these boundaries are

$$r_L(t) = 5t \tag{57}$$

$$r_R(t) = 5t + 3 \tag{58}$$

With the piston method of initialization, the boundary initially at $r_{R,i}$ is retained, as is its subsequent motion given by Eq. (58). However, in this case, the boundary initially at $r_{L,i}$ is replaced with the boundary/piston with time-dependent position given by

Table 7 Unweighted spatial L_1 error and associated convergence data for 2D planar modified Cog19 problem simulations: direct initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	A	B
ρ	2.99×10^{-1}	1.52×10^{-1}	7.71×10^{-2}	3.90×10^{-2}	7.06	0.981
u	1.94×10^{-2}	1.02×10^{-2}	5.20×10^{-3}	2.60×10^{-3}	0.430	0.959
P	6.35×10^{-2}	3.18×10^{-2}	1.65×10^{-2}	7.90×10^{-3}	1.57	0.994
e	1.30×10^{-2}	6.70×10^{-3}	4.50×10^{-3}	1.70×10^{-3}	0.272	0.935

Table 8 Unweighted spatial L_1 error and associated convergence data for 2D planar modified Cog19 problem simulations: piston initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	A	B
ρ	8.52×10^{-2}	4.30×10^{-2}	2.15×10^{-2}	1.08×10^{-2}	2.10	0.995
u	1.64×10^{-2}	8.50×10^{-3}	4.30×10^{-3}	2.20×10^{-3}	0.378	0.973
P	6.70×10^{-3}	3.40×10^{-3}	1.70×10^{-3}	8.00×10^{-4}	0.172	1.01
e	5.80×10^{-3}	2.90×10^{-3}	1.50×10^{-3}	7.00×10^{-4}	0.144	0.996

Table 9 Shock location/magnitude for 1D spherical Cog20 problem simulations: direct initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	Exact
r_s	2.254	2.304	2.314	2.326	2.333
M_s	4.483	4.304	4.237	4.138	4.0

$$r_b(t) = 5t \tag{59}$$

In Tables 5–9, the Δr values indicate the initially uniform spatial grid zone size corresponding to a given simulation (with $\Delta r = \Delta y$ in all cases).

4.3 Coggeshall 20—1D Spherical. One-dimensional spherical simulation results for Cog20 appear in Fig. 5 and Tables 9–12 for the parameterization $k = 2$, $\gamma = 5/3$, $\rho_0 = 1$, $u_0 = -1$, $a = -1$,

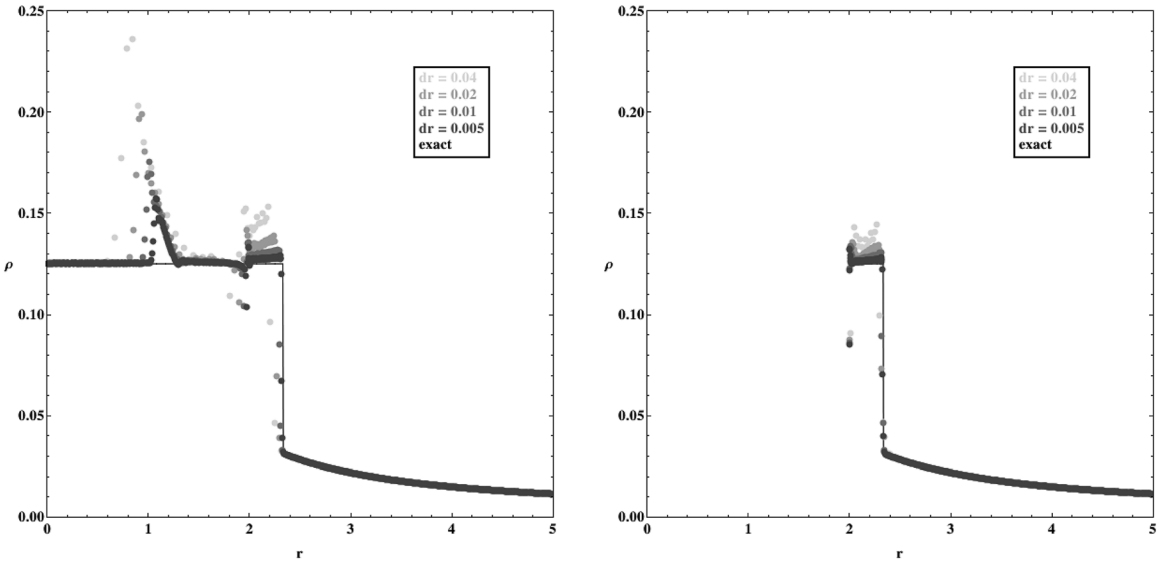

Fig. 5 One-dimensional spherical Cog20 problem density simulation results for direct (left) and piston (right) initialization methods

Table 10 Shock location/magnitude for 1D spherical Cog20 problem simulations: piston initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	Exact
r_s	2.337	2.335	2.334	2.334	2.333
M_s	3.752	3.410	3.987	4.135	4.0

Table 11 Unweighted spatial L_1 error and associated convergence data for 1D spherical Cog20 problem simulations: direct initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	A	B
ρ	1.71×10^{-2}	9.02×10^{-3}	4.16×10^{-3}	2.05×10^{-3}	0.483	1.03
u	7.36×10^{-3}	3.58×10^{-3}	1.98×10^{-3}	7.57×10^{-4}	0.233	1.07
P	1.68×10^{-4}	8.30×10^{-5}	3.66×10^{-5}	1.75×10^{-5}	0.006	1.10
e	6.24×10^{-5}	2.97×10^{-5}	1.61×10^{-5}	7.88×10^{-6}	0.002	0.98

Table 12 Unweighted spatial L_1 error and associated convergence data for 1D spherical Cog20 problem simulations: piston initialization method

	$\Delta r = 0.04$	$\Delta r = 0.02$	$\Delta r = 0.01$	$\Delta r = 0.005$	A	B
ρ	6.65×10^{-3}	3.30×10^{-3}	1.87×10^{-3}	9.26×10^{-4}	0.133	0.935
u	1.84×10^{-3}	9.09×10^{-4}	5.19×10^{-4}	2.69×10^{-4}	0.034	0.914
P	3.46×10^{-5}	1.57×10^{-5}	1.05×10^{-5}	4.77×10^{-6}	0.001	0.921
e	4.05×10^{-5}	2.14×10^{-5}	1.21×10^{-5}	6.09×10^{-6}	0.001	0.902

and $t_i = 3$. In each case, the problem is simulated with a constant time step $\Delta t = 1.0 \times 10^{-5}$ to a final time $t_f = 7$ (boundary/piston location $r_b = 2$, where applicable), where analysis is performed.

With the direct method of initialization, an artificial boundary is included at $r_{R,i} = 3$ (the right boundary is the 1D spherical origin). Following from Eqs. (30), (33), and (34), the time-dependent position of this boundary is

$$r_R(t) = \frac{3(t+1)}{4} \tag{60}$$

With the piston method of initialization, the boundary initially at $r_{R,i}$ is retained, as is its subsequent motion given by Eq. (60). In this case, however, the boundary at the origin is replaced with the boundary/piston with time-dependent position given by

$$r_b(t) = \frac{t+1}{4} \tag{61}$$

In Tables 9–12, the Δr values indicate the initially uniform spatial grid zone size corresponding to a given simulation.

5 Discussion

5.1 Start-Up and Wall-Heating Errors. Figures 3–5 show that the simulation results manifest two error modes; these appear to be the primary contributors to the L_1 error norm data appearing in Tables 3, 4, 7, 8, 11, and 12.

Each of the simulations corresponding to the direct initialization method shows evidence of an error archetype called a "start-up error," by LeVeque [23]. As discussed in further detail by LeVeque [23] and Arora and Roe [24], start-up errors are associated with the use of exact discontinuities as initial data in compressible flow solvers (as exemplified by the direct initialization method).

Figures 3–5 indicate that the start-up errors are most prominent in the Cog19 and Cog20 problems, where they appear as high-amplitude wave structures to the left of the shock position; they are comparatively larger in spatial grid-dependent extent and magnitude as opposed to the more diffuse errors observed for the corresponding Zel'dovich problem. This difference suggests that the relative strength of the start-up errors is affected by the precise nature of the discontinuous initial data; Cog19 and Cog20 as implemented feature nonzero flow velocities in both initial regions, while the Zel'dovich problem features one region with zero velocity. Explicitly quantifying the impact of this key difference on the resulting manifestation of these errors is outside the scope of this work, but is a potential avenue for future study.

As observed by LeVeque [23] and Ramsey et al. [5], start-up errors are acoustic waves that move at either the local shocked fluid velocity $u_2(r)$ or a local characteristic speed $u_2(r) \pm c_2(r)$, where $c_2(r)$ is the local sound speed of the shocked fluid. Start-up errors are affected by grid refinement: in some cases, both their

extent and magnitude are reduced under refinement, while in other cases, only the extent is reduced. In any event, the start-up error contribution to the spatial L_1 error norm is reduced as the spatial grid is refined.

In contrast to the direct initialization method simulation results, each of the simulations corresponding to the piston initialization method shows evidence of the "wall-heating" error archetype. Wall-heating errors arise in simulations involving compressible fluid flow normal to rigid boundaries and have been discussed in detail by Noh [10] and Rider [11] (among many others). Figures 3–5 show distinct evidence of wall-heating errors for all test problems involving the piston initialization method.

Unlike start-up errors, the wall-heating errors shown in Figs. 3–5 remain confined to the vicinity of the boundary/piston (the left problem boundary in each case). Like the start-up errors, their extent and magnitude depend on the problem under consideration. Their extent (but not magnitude) also depends on spatial grid resolution and diminishes under refinement.

Both start-up errors and wall-heating effects are inherent features of finite-volume shock-capturing algorithms (of which the FLAG compressible flow solver is an example). While it is reasonable to assume the details of these phenomena vary with different test problems and compressible flow solver algorithms, to the authors' knowledge there is no evidence in the literature that these phenomena can be altogether eliminated in finite-volume codes.

5.2 Accuracy and Convergence. Shock position r_s, shock magnitude M_s, spatial L_1 error norm, and associated convergence data are summarized for each test problem and initialization strategy in Tables 1–12.

Tables 1, 2, 5, 6, 9, and 10 show that for each problem and for a given spatial grid resolution Δr, the piston initialization method subsequently results in more accurate shock positions r_s. For the Zel'dovich problem, the direct method of initialization results in an overpredicted shock location, while the piston method of initialization results in an underpredicted shock location. This trend is reversed for both the Cog19 and Cog20 test problems. In all cases, the computed shock location converges to the exact shock location under spatial grid refinement.

In addition, Tables 1, 2, 5, 6, 9, and 10 show that for each problem and for a given spatial grid resolution Δr, the piston initialization method subsequently results in more accurate shock magnitudes M_s (as reflected by the density ratio across the shock). Distinct trends associated with this metric—including convergence to the exact solution under spatial grid refinement—are not readily discernible due to the presence of various errors in the shocked fluid regions (as discussed in Sec. 5.1).

Tables 3, 4, 7, 8, 10, and 12 show that for each problem and for a given spatial grid resolution Δr, the piston initialization method subsequently results in smaller spatial L_1 error norms, as also reflected by universally smaller values of the convergence coefficient A. This trend is expected because shock locations and magnitudes are captured more accurately using the piston initialization method. Moreover, in each case, the wall-heating errors associated with the piston initialization method appear to be less extensive than the start-up errors associated with the direct initialization method.

Figures 1–3 show that in general, the piston initialization method subsequently results in a spatially less extensive shocked region than that corresponding to the direct initialization method. This phenomenon likely explains the discrepancy between the spatial L_1 error norm and convergence coefficient A results between the two initialization methods: the postshock error is compressed over a smaller region in the piston initialization method simulation results.

Finally, the values of the convergence rate B appearing in Tables 3, 4, 7, 8, 10, and 12 show that the spatial L_1 error norm calculated for all flow variables converges to the corresponding exact solutions at approximately first-order, regardless of test problem and initialization method. Degradation of the computed convergence rate to $B \sim 0.9$ appears for some flow variables in simulations of the Zel'dovich (direct initialization method) and Cog20 (piston initialization method) problems. In view of the approximate nature of the fitting procedure used to generate the convergence rate B (see Sec. 4), it is unlikely that these isolated trends indicate a significant difference between the two initialization methods for these test problems.

In any event, the approximately first-order convergence observed for each test problem and initialization method is consistent with the discussion provided in Sec. 5 for simulation of discontinuous compressible flows (e.g., shock flows).

5.3 Conclusions and Future Study. The simulation results discussed in Secs. 5.1 and 5.2 indicate that the piston method of initialization—where applicable—appears to be a viable alternative to the direct initialization method for simulation of compressible flow similarity solution test problems. For the compressible flow solver, test problems, grid spacing series, and other parameter choices investigated in this work, the piston initialization method generates more accurate numerical solutions for shock positions, shock magnitudes, and spatial L_1 error norms. The associated convergence of the computed solutions to the corresponding exact solutions appears to be insensitive to initialization method.

Despite its apparent benefits, to the knowledge of the authors, the piston method of initialization is not universally applicable. As discussed in Sec. 3.2, this method is nontrivial to implement for problems containing stagnation shocks (e.g., the Noh problem [3,10–12]). The computational implementation of the piston initialization method is also likely to be nontrivial within the Eulerian or arbitrary Lagrangian–Eulerian frameworks, which do not naturally lend themselves to the required moving boundary mechanism. The reconciliation of this inconsistency represents a potential avenue for further research.

In addition to the resolution of these outstanding issues, there appear to be numerous potential avenues for the extension of this work. For completeness, additional parameterizations or instantiations of the Zel'dovich, modified Cog19, and Cog20 test problems should be considered (e.g., using different choices of various free parameters, or extended into 2D or 3D computational representations). The large computational parameter space can also be further investigated, including various time-step control methods, artificial viscosity methods, or additional Lagrangian compressible flow codes. Temporal and combined spatial–temporal convergence (as discussed by Kamm et al. [25]) can be investigated.

Additional similarity solutions of Eqs. (1)–(3) containing shocks can also be investigated, including the Taylor–von Neumann–Sedov blast wave [1,26,27] and Guderley converging shock [5,28] test problems. In the former context [14], the piston initialization method may prove useful in addressing issues surrounding nonsymmetrical initialization of the required initial condition [29]. Moreover, in the latter case [14,15], the piston method of initialization may prove a viable alternative to the more complicated current state of initialization strategy [5,30].

Acknowledgment

This work was performed under the auspices of the United States Department of Energy by Los Alamos National Security, LLC, at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396. The authors acknowledge the support of the U.S. Department of Energy Advanced Strategic Computing Program Physics Verification Project under project leader S. Doebling, and the U.S. Department of Energy Advanced Strategic Computing Program Computational Physics Student Summer Workshop under program director S. Runnels. The authors thank T. Carney, M. Clover, R. Hendon, J. Kamm, M. Kenamond, L. Margolin, N. Morgan, D. Shirk, and P. Whalen for valuable insights on these topics.

Appendix: Shock Position/Speed for Coggeshall Solution 20

The Cog20 similarity solution to Eqs. (1)–(3) is given by Eqs. (26)–(32). Since $P=0$ in the unshocked region, the shock is strong, and so Eqs. (8)–(10) must be satisfied at the shock position $r=r_s(t)$. For example, the strong density jump condition is

$$\rho_2 = \frac{\gamma+1}{\gamma-1}\rho_1 \tag{A1}$$

Moreover, if the unshocked and shocked densities immediately adjacent to the shock are given by the Cog20 solution evaluated at $r=r_s$

$$\rho_1 = \left[\rho_0\left(\frac{r-u_0t}{r}\right)^k(1-at)^{-k-1}\right]_{r=r_s} \tag{A2}$$

$$\rho_2 = \left[\rho_0\left(\frac{\gamma+1}{\gamma-1}\right)^{k+1}(1-at)^{-k-1}\right]_{r=r_s} \tag{A3}$$

then Eq. (A1) becomes

$$\frac{u_0t}{r_s} = -\frac{2}{\gamma-1} \tag{A4}$$

For a self-similar strong shock, Eq. (A4) must be satisfied for all t, so

$$r_s = -\frac{(\gamma-1)u_0\,t}{2} \tag{A5}$$

and

$$D = \frac{dr_s}{dt} = -\frac{(\gamma-1)u_0}{2} \tag{A6}$$

as given by Eq. (32). This is in contrast to the shock position given by Coggeshall [6] for solution 20

$$r_s = \left[\frac{(\gamma-1)u_0}{4a}\right]\left[\frac{t(1-2at)}{1-at}\right] \tag{A7}$$

which does not satisfy Eq. (A4) for all t and thus does not correspond to a strong shock, as required by the condition $P=0$ in the unshocked region for the Cog20 similarity solution.

References

[1] Sedov, L., 1959, *Similarity and Dimensional Methods in Mechanics*, Academic Press, New York.
[2] Barenblatt, G., 1996, *Scaling, Self-Similarity, and Intermediate Asymptotics: Dimensional Analysis and Intermediate Asymptotics*, Cambridge University Press, Cambridge, UK.
[3] Axford, R., 2000, "Solutions of the Noh Problem for Various Equations of State Using Lie Groups," Laser Part. Beams, **18**(1), pp. 93–100.
[4] Zel'dovich, Y., and Raizer, Y., 2002, *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena*, Dover Publications, Mineola, New York.
[5] Ramsey, S., Kamm, J., and Bolstad, J., 2012, "The Guderley Problem Revisited," Int. J. Comput. Fluid Dyn., **26**(2), pp. 79–99.
[6] Coggeshall, S., 1991, "Analytic Solutions of Hydrodynamics Equations," Phys. Fluids A, **3**(5), pp. 757–769.
[7] Oberkampf, W., Trucano, T., and Hirsh, C., 2004, "Verification, Validation, and Predictive Capability in Computational Engineering and Physics," ASME Appl. Mech. Rev., **57**(5), pp. 345–384.
[8] Roy, C., 2005, "Review of Code and Solution Verification Procedures for Computational Simulation," J. Comput. Phys., **205**(1), pp. 131–156.
[9] Zel'dovich, Y., 1956, "Motion of a Gas Under the Action of an Impulsive Pressure (Load)," Akust. Zh., **2**, pp. 28–38.
[10] Noh, W., 1987, "Errors for Calculations of Strong Shocks Using an Artificial Viscosity and an Artificial Heat Flux," J. Comput. Phys., **72**(1), pp. 78–120.
[11] Rider, W., 2000, "Revisiting Wall Heating," J. Comput. Phys., **162**(2), pp. 395–410.
[12] Gehmyer, M., Cheng, B., and Mihalas, D., 1997, "Noh's Constant-Velocity Shock Problem Revisited," Shock Waves, **7**(5), pp. 255–274.
[13] Landau, L., and Lifshitz, E., 1987, *Fluid Mechanics*, 2nd ed., Pergamon Press, Oxford, UK.
[14] Sachdev, P., 2004, *Shock Waves and Explosions*, Chapman & Hall/CRC, Boca Raton, FL.
[15] Lazarus, R., 1981, "Self-Similar Solutions for Converging Shocks and Collapsing Cavities," SIAM J. Numer. Anal., **18**(2), pp. 316–371.
[16] Burton, D., 1990, "Conservation of Energy, Momentum, and Angular Momentum in Lagrangian Staggered-Grid Hydrodynamics," Lawrence Livermore National Laboratory, Livermore, CA, Technical Report No. UCRL-JC-195926.
[17] Caramana, E., Burton, D., Shashkov, M., and Whalen, P., 1998, "The Construction of Compatible Hydrodynamics Algorithms Utilizing Conservation of Total Energy," J. Comput. Phys., **146**(1), pp. 227–262.
[18] Caramana, E., Shashkov, M., and Whalen, P., 1998, "Formulations of Artificial Viscosity for Multi-Dimensional Shock Wave Computations," J. Comput. Phys., **144**(1), pp. 70–97.
[19] Morgan, N., 2013, "A Dissipation Model for Staggered Grid Lagrangian Hydrodynamics," Comput. Fluids, **83**, pp. 48–57.
[20] Morgan, N., Lipnikov, K., Burton, D., and Kenamond, M., 2014, "A Lagrangian Staggered Grid Godunov-Like Approach for Hydrodynamics," J. Comput. Phys., **259**, pp. 568–597.
[21] Banks, J., Aslam, T., and Rider, W., 2008, "On Sub-Linear Convergence for Linearly Degenerate Waves in Capturing Schemes," J. Comput. Phys., **227**(14), pp. 6985–7002.
[22] Majda, A., and Ralston, J., 1979, "Discrete Shock Profiles for Systems of Conservation Laws," Comm. Pure Appl. Math., **32**(4), pp. 445–482.
[23] LeVeque, R., 2002, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, Cambridge, UK.
[24] Arora, M., and Roe, P., 1997, "On Postshock Oscillations Due to Shock Capturing Schemes in Unsteady Flows," J. Comput. Phys., **130**(1), pp. 25–40.
[25] Kamm, J., Rider, W., and Brock, J., 2003, "Combined Space and Time Convergence Analysis of a Compressible Flow Algorithm," AIAA Paper No. 2003-4241.
[26] Taylor, G., 1950, "The Formation of a Blast Wave by a Very Intense Explosion. I. Theoretical Discussion," Proc. R. Soc. London, Ser. A, **201**(1065), pp. 159–174.
[27] Korobeĭnikov, V., 1991, *Problems of Point Blast Theory*, Springer, New York.
[28] Guderley, G., 1942, "Starke kugelige und zylindrische Verdichtungsstöβe in der Nähe des Kugelmittelpunktes bzw. der Zylinderachse," Luftfahrtforschung, **19**, pp. 302–312.
[29] Doebling, S., and Ramsey, S., 2013, "Impact of Artificial Viscosity Models on Verification Assessment of a Lagrangian Hydrodynamics Code Using the Sedov Problem," Los Alamos National Laboratory, Los Alamos, NM, Technical Report No. LA-UR-13-23559.
[30] Hornung, H., Pullin, D., and Ponchaut, N., 2008, "On the Question of Universality of Imploding Shock Waves," Acta Mech., **201**(1–4), pp. 31–35.