On Efficient Generation of Numerically Robust Sedov Solutions

James R. Kamm 1 and F.X. Timmes 1,2

ABSTRACT

We describe the efficient generation of robust numerical solutions for a Sedov blast wave propagating through a polytropic gas characterized by a density gradient $\rho = \rho_0 r^{-\omega}$. We discuss all possible families of real solutions, in all common geometries, and address all six removable singularities. The nature of the solutions changes dramatically as ω and the adiabatic index γ vary. Implementation of Sedov solutions requires extended precision arithmetic near the origin to avoid running out of significant figures. We suggest several new problem definitions for verifying hydrodynamic codes, and offer public domain modules to generate the necessary closed-form solutions.

Subject headings: methods: analytical — methods: numerical — shock waves

1. Introduction

Over 60 years ago, von Neumann (1941), Taylor (1941), and Sedov (1946) independently derived a self-similar description of the evolution of the blast wave arising from a powerful explosion in a cold, uniform density background (also see Bethe et al. (1947)). They treated the explosion as an instantaneous release of energy at a point and assumed that the background material through which the expanding blast sweeps behaves as an ideal polytropic fluid. Later, Korobeinikov et al. (1962) extended the description to power-law initial background densities of the form $\rho = \rho_0 r^{-\omega}$. It is remarkable that these models yield closed-form expressions for the fluid quantities.

The venerable Sedov problem might appear to be an old, solved problem. However, there is a paucity of papers that discuss the typographical errors in Sedov (1959) and Korobeinikov et al. (1962), numerical implementation of all possible families of real solutions, in all common geometries, and address all the removable singularities. Book (1991) and Kamm (2000) come close, but both omit discussion of the singularities at the lower bounds of the energy integral.

¹Applied Physics Division, Los Alamos National Laboratory, Los Alamos, NM 87545

²Joint Institute for Nuclear Astrophyics, University of Notre Dame, South Bend, IN 46556

In this paper we gather together the results of these two papers (and others), extend them by discussing all the removable singularities, and offer an instantiation of the ideas in a succinct computer code.

Sedov solutions are commonly used to analyze portions of the evolution of a supernova remnant, the interaction of ambient gas with stellar material ejected by a supernova. In particular, the standard paradigm has this interaction characterized by four stages: ejectadominated, Sedov-Taylor, pressure-driven snowplow, and momentum-conserving snowplow (Chevalier 1976). The first two stages are commonly classified as nonradiative because radiative losses are dynamically insignificant, although they may be of observational interest. In the later stages, radiative evolution should be considered.

In addition, the constant density, spherically symmetric Sedov blast wave is stalwart test case in the verification of hydrodynamic codes. It is not a particularly strenuous test for a modern hydrocode, except perhaps for using Cartesian meshes to find the spherically symmetric solution (Fryxell et al. 2000; Teyssier 2002; O'Shea et al. 2004; Dursi & Timmes 2006). In this paper we define several new verification problems for hydrodynamic codes involving more challenging Sedov blast waves.

In §2 we describe the calculation of the four Sedov functions that describe the spatial variation of density, material speed, and pressure with distance at any point in time. The general behavior of these functions is analyzed as a function of ω and the adiabatic index γ for various geometries. In §3 we use these Sedov functions to compute the energy integral, taking a rigorous approach to removing the singularities at the lower integration limit. In §4 we generate physical solutions for a number of types of Sedov blast waves, and offer definitions for new code verification test problems. In the Appendix B we list key source code modules for efficient generation of numerically robust Sedov solutions.

2. Calculating the Sedov functions

An energy E_{blast} is deposited at the origin at time zero in an infinite domain characterized by a non-uniform density $\rho = \rho_0 r^{-\omega}$, material speed v_0 , pressure P_0 , specific internal energy E_0 , sound speed c_{s0} , and adiabatic index γ . Here ρ_0 and ω are constants. When the value of ω is identically zero, the initial density is uniform. A solution for the complete physical state is desired at any distance $0 < r_{\text{want}} < \infty$ at any time $0 < t_{\text{want}} < \infty$.

To solve this problem Sedov (1959) considers the set of equations governing one-dimensional,

compressible hydrodynamics:

$$\rho_t + v \,\rho_r + \frac{\rho}{r^{j-1}} \,\left(r^{j-1} \,v\right)_r = 0\,, \tag{1}$$

$$v_t + v v_r + \frac{1}{\rho} P_r = 0,$$
 (2)

$$(P/\rho^{\gamma})_t + u \ (P/\rho^{\gamma})_r = 0.$$
 (3)

The index j = 1, 2, or 3 is the dimensionality index for one-dimensional planar, cylindrical, or spherical geometry, respectively. Note that to keep the total mass finite within the domain of interest one must have $0 \le \omega < j$. The thermodynamics is assumed to be governed by the incomplete equation of state (in the sense of Menikoff & Plohr (1989)) for a polytropic gas

$$(\gamma - 1) E = P/\rho \quad . \tag{4}$$

Distinct from the analysis of Reinicke & Meyer-ter-Vehn (1991) or Shestakov (1999), heat conduction is assumed to be negligible, so that purely hydrodynamic motion occurs.

One can begin a numerical solution by determining the family type to which the initial conditions correspond (Kamm 2000). We begin by evaluating the immediate post-shock velocity in the similarity variable V_2 and the location of the singular point in this variable V_* :

$$V_2 = \frac{4}{j+2-\omega} \tag{5}$$

$$V_* = \frac{2}{j(\gamma - 1) + 2} . (6)$$

These two quantities are sufficient to determine the type of solution: standard, singular, or vacuum. In the standard case, a nonzero solution extends from the shock to the origin, where the pressure is finite. In the singular case, a nonzero solution extends from the shock to the origin, where the pressure vanishes. In the vacuum case, a nonzero solution extends from the shock to a boundary point between the origin and the shock, where the density vanishes. In particular, if

$$|V_2 - V_*| \le \epsilon \tag{7}$$

then the solution type is singular, while if

$$V_2 < V_* - \epsilon \tag{8}$$

then the solution type is standard, while if

$$V_2 > V_* + \epsilon \tag{9}$$

the solution is a vacuum type. Here ϵ is a "small" numerical parameter (~ 10⁻⁴) assigned to (i) avoid a hard zero and (ii) limit some of the exponents to be discussed below from becoming sufficiently large to create numerical overflows. It is useful to note that the singular case occurs at $\omega_1 = (3j - 2 + \gamma(2 - j))/(\gamma + 1)$.

Although the guiding differential equations for the similarity variables are analytic, the Sedov functions and the energy integral to be discussed below become singular for various combinations of ω , j and γ . These singularities, however, are only apparent: they are a consequence of the way the solution is formulated. Two of these removable singularities, using the notation of Book (1991), should be recorded at this point in the calculation. If either

$$|\omega - \omega_2| = \left|\omega - \left[\frac{2(\gamma - 1) + j}{\gamma}\right]\right| \le \epsilon \tag{10}$$

or

$$|\omega - \omega_3| = |\omega - j(2 - \gamma)| \le \epsilon \quad , \tag{11}$$

then various denominators will become zero, and appropriate limits must be taken. That is, the removable singularities occur at ω_2 and ω_3 . Nothing particularly interesting either mathematically or physically happens at these apparent singularities; at best they are inflection points for the behavior of the pressure near the origin.

The distance r_2 from the center of symmetry to the shock position is given as a function of time by

$$r_2 = \left(\frac{E_{\text{blast}}t^2}{\alpha\rho_0}\right)^{1/(j+2-\omega)} \quad , \tag{12}$$

where α is related to the energy integral discussed below and depends only on ω , j, and γ . The speed of the shock follows as

$$u_s = \frac{2}{j+2-\omega} \frac{r_2}{t} . (13)$$

An infinitely strong shock is assumed, so that the pre-shock pressure is negligible, i.e., $P_0 \rightarrow 0$. With this assumption substituted into the standard shock jump relations (Whitham 1974), the immediate post-shock values of the material speed u_2 , density ρ_2 , pressure P_2 , specific energy E_2 , and the non-relativistic sound speed c_2 are determined as

$$u_2 = \frac{2u_s}{\gamma + 1} \tag{14}$$

$$\rho_2 = \frac{\gamma + 1}{\gamma - 1} \rho_0 \tag{15}$$

$$P_2 = \frac{2\rho_0 u_s^2}{\gamma + 1} \tag{16}$$

$$E_2 = \frac{P_2}{\rho_2(\gamma - 1)}$$
(17)

$$c_2 = \sqrt{\frac{\gamma P_2}{\rho_2}} . \tag{18}$$

Now form the six exponents

$$\alpha_0 = \frac{2}{j+2-\omega} \tag{19}$$

$$\alpha_2 = -\frac{\gamma - 1}{\gamma(\omega_2 - \omega)} \tag{20}$$

$$\alpha_1 = \frac{(j+2-\omega)\gamma}{2+j(\gamma-1)} \left[\frac{2[j(2-\gamma)-\omega]}{\gamma(j+2-\omega)^2} - \alpha_2 \right]$$
(21)

$$\alpha_3 = \frac{j - \omega}{\gamma(\omega_2 - \omega)} \tag{22}$$

$$\alpha_4 = \frac{(j+2-\omega)(j-\omega)}{\omega_3 - \omega} \alpha_1 \tag{23}$$

$$\alpha_5 = \frac{\omega(1+\gamma) - 2j}{\omega_3 - \omega} \tag{24}$$

and the frequently-used combinations

$$a = \frac{(j+2-\omega)(\gamma+1)}{4}$$
 (25)

$$b = \frac{\gamma + 1}{\gamma - 1} \tag{26}$$

$$c = \frac{\gamma(j+2-\omega)}{2} \tag{27}$$

$$d = \frac{(j+2-\omega)(\gamma+1)}{(j+2-\omega)(\gamma+1) - 2[2+j(\gamma-1)]}$$
(28)

$$e = \frac{2+j(\gamma-1)}{2}.$$
 (29)

With the above definitions, we turn to the calculation of the Sedov functions λ (scaled position), f (scaled speed), g (scaled density), and h (scaled pressure). Given a value for the similarity variable V, first compute the following four auxiliary functions and their first derivatives with respect to the similarity variable V

$$x_1 = aV \qquad \qquad \frac{dx_1}{dV} = a \tag{30}$$

$$x_2 = b (cV - 1) \qquad \qquad \frac{dx_2}{dV} = bc \tag{31}$$

$$x_3 = d (1 - eV) \qquad \qquad \frac{dx_3}{dV} = -de \tag{32}$$

$$x_4 = b \left(1 - (cV/\gamma)\right) \qquad \frac{dx_4}{dV} = -bc/\gamma \quad . \tag{33}$$

The first derivatives will be used later when computing the energy integral. If the solution type is standard or vacuum type (i.e., equation 8 or equation 9 is true), then the four Sedov functions and the first derivative of the scaled position with respect to the similarity variable V are

$$r/r_2 \equiv \lambda = x_1^{-a_0} x_2^{-a_2} x_3^{-a_1} \tag{34}$$

$$\frac{d\lambda}{dV} = -\lambda \left[\frac{\alpha_0}{x_1} \frac{dx_1}{dV} + \frac{\alpha_2}{x_2} \frac{dx_2}{dV} + \frac{\alpha_1}{x_3} \frac{dx_3}{dV} \right]$$
(35)

$$v/v_2 \equiv f = x_1\lambda \tag{36}$$

$$\rho/\rho_2 \equiv g = x_1^{\alpha_0\omega} x_2^{\alpha_3 + \alpha_2\omega} x_3^{\alpha_4 + \alpha_1\omega} x_4^{a_5}$$
(37)

$$p/p_2 \equiv h = x_1^{\alpha_0 \omega} x_3^{\alpha_4 + \alpha_1 (\omega - 2)} x_4^{1 + \alpha_5} .$$
 (38)

If the second apparent singularity asserts itself (equation 10 is true), then by taking the limit $\omega \to \omega_2$ of equations 34–38 one finds (Book 1991)

$$\lambda = x_1^{-a_0} x_2^{(\gamma-1)/2e} \exp\left[\frac{(\gamma+1)}{2e} \frac{(1-x_1)}{(x_1 - \frac{\gamma+1}{2\gamma})}\right]$$
(39)

$$\frac{d\lambda}{dV} = -\lambda \left[\frac{\alpha_0}{x_1} \frac{dx_1}{dV} + \frac{(\gamma - 1)}{2ex_2} \frac{dx_2}{dV} - \frac{(\gamma + 1)}{2e} \frac{dx_1}{dV} \left[\frac{1}{(x_1 - \frac{\gamma + 1}{2\gamma})} \right] \left[1 + \frac{(1 - x_1)}{(x_1 - \frac{\gamma + 1}{2\gamma})} \right] \right] (40)$$

$$f = x_1 \lambda$$

$$(41)$$

$$g = x_1^{\alpha_0 \omega} x_2^{4-j-2\gamma/2e} x_4^{a_5} \exp\left[\frac{(\gamma+1)}{e} \frac{(1-x_1)}{(x_1 - \frac{\gamma+1}{2\gamma})}\right]$$
(42)

$$h = x_1^{\alpha_0 \omega} x_3^{-j\gamma/2e} x_4^{1+\alpha_5} .$$
(43)

If the third removable singularity asserts itself (equation 11 is true), then by taking the limit $\omega \to \omega_3$ of equations 34–38 one finds (Book 1991)

$$\lambda = x_1^{-a_0} x_2^{-a_2} x_4^{-a_1} \tag{44}$$

$$\frac{d\lambda}{dV} = -\lambda \left[\frac{\alpha_0}{x_1} \frac{dx_1}{dV} + \frac{\alpha_2}{x_2} \frac{dx_2}{dV} + \frac{\alpha_1}{x_4} \frac{dx_4}{dV} \right]$$

$$f = x_1 \lambda$$
(45)

$$g = x_1^{\alpha_0 \omega} x_2^{\alpha_3 + \omega \alpha_2} x_4^{1-2/e} \exp\left[-\frac{j\gamma(\gamma+1)}{2e} \frac{1-x_1}{(\gamma+1)/2 - x_1}\right]$$
(46)

$$h = x_1^{\alpha_0 \omega} x_4^{(j(\gamma-1)-\gamma)/e} \exp\left[-\frac{j\gamma(\gamma+1)}{2e} \frac{1-x_1}{(\gamma+1)/2-x_1}\right] .$$
(47)

If the solution type is singular (i.e., equation 7 is true), then

$$\lambda = r_{\rm want}/r_2 \tag{48}$$

$$\frac{d\lambda}{dV} = 0 \tag{49}$$

$$f = \lambda \tag{50}$$

$$q = \lambda^{j-2} \tag{51}$$

$$h = \lambda^j . (52)$$

In this case, the (dimensional) speed is an exactly linear function of the (dimensional) position by virtue of equation 50. Finally, if the solution is of the vacuum type (i.e., equation 9 is true) and we are in the "hole" region ($r_{\text{want}} < r_{\text{vac}}$), then

$$\lambda = \frac{d\lambda}{dV} = f = g = h = 0 \quad . \tag{53}$$

There are few instances in the literature of analyzing the Sedov functions f, g, and h as functions of λ for a representative selection of the entire (j, γ, ω) parameter space (Chevalier 1976; Ryu & Vishniac 1987; Vishniac & Ryu 1987). Book (1991) filled this gap by plotting the physical profiles for $\gamma=1.1$, 1.2, 1.3, 1.4, 1.5, and 5/3 (values appropriate to ordinary gases) and $0 \leq \omega < 3$ (the values for which the total mass in the region is finite) for j=3. To keep the number of figures within reasonable bounds, Book (1991) does not carry out a similarly complete survey for j=1 and j=2, but does include a few examples in order to indicate how they differ. As part of the verification of the code listed in the Appendix B, we replicate those plots.

Figure 1 shows the Sedov functions for planar geometry j=1 and $\gamma=1.1$ and 5/3 with ω between 0 and 1. The values of ω are chosen so that there are no drastic changes in behavior from one case to the next, i.e., à *la* Book (1991), they are "roughly evenly separated in phenomenology space." As ω varies from 0 to 1 the following qualitative changes occur: the density develops an inflection point (Fig. 1c) and then becomes non-zero at $\omega=j/\gamma$ (Fig. 1d). For $\omega > j/\gamma$ the density diverges at the origin. As $\omega \to 1$ the singular case (Fig. 1h) occurs and the pressure goes to zero at the origin. That is, the vacuum case cannot occur in planar geometry. The same behavior takes place for $\gamma=5/3$ (Fig. 1i-l), but each stage occurs at a smaller ω , so that the pattern is spread out over the whole range of ω .

Figure 2a-h displays analogous results for cylindrical geometry j=2 and $\gamma=1.1$ with ω between 0 and 2. Now the vacuum type solutions occur for $\omega_1 < \omega \leq \omega_2$. Note the velocity is finite at the vacuum boundary. As ω continues to increase, the pressure, which is initially concave upward, becomes convex and then, in the limit $\omega \to \omega_2$, discontinuous at

the vacuum boundary. Figures 2i–p show the same stages occur for $\gamma=5/3$, but spread out over a broader range of ω .

In spherical geometry Figs. 3–5 exhibit three new features that occur in the vacuum region. First, when the vacuum region develops, the density becomes convex upward. Second, the density bulges increasingly upward with increasing ω , eventually developing a local maximum. Third, as $\omega \to 2j/(\gamma + 1)$ the density becomes discontinuous at the periphery of the vacuum boundary and divergent (with a negative slope) for larger values of ω . These same stages occur in the same order for all values of γ —only the values of ω marking the locations of the transitions change.

Overall, the scaled material speed, f, changes the least and the scaled mass density, g, changes the most as the parameters j, γ , and ω vary. In all cases f is nearly linear with scaled distance λ when a vacuum region is not present; f is exactly linear in λ in the singular case. There are ranges of ω where the scaled density g has a negative slope over some range of λ (Book 1991; Chevalier 1976). Since the slope of the scaled pressure h is always positive, the two have opposite signs. Hence these regions are formally unstable to convection (Book 1991; Chevalier 1976). A rigorous calculation of the growth of even small amplitude perturbations is complicated not only by the spacetime dependence of the basic state, but by the difficulty of obtaining the correct boundary conditions at the shock front, which is continually sweeping over new material (Lifshitz 1946; Bernstein & Book 1978, 1980; Ryu & Vishniac 1987; Vishniac & Ryu 1987). The problem is analogous to that of calculating the stability of a detonation front or an ablating inertial confinement fusion pellet (Oppenheim et al. 1972; Kidder 1974, 1976; Bernstein & Book 1980; Remington et al. 1999; Hansen et al. 2005).

Figure 6a and 6b display the profiles for $\omega = \omega_1$ in the cylindrical and spherical cases, respectively. Two lines might appear to be missing: for the cylindrical case the density profile equals unity and coincides with the upper axis box, while for the spherical case the density and velocity profiles overlap everywhere. The quantity hg^{γ} , which is related to the entropy, is plotted in Fig.6c and 6d for $\gamma = 1.1$ and $\gamma = 5/3$ and ten different values of ω . These two entropy plots differ substantially from those in Book (1991).

3. Calculating the Energy Integral

To determine the solution corresponding to the given initial energy released E_{blast} at the origin, we must relate the solution parameters to that quantity. The energy, which is constant throughout the motion, is the sum of the kinetic and internal energies, which can be expressed in non-dimensional form (after some manipulation) as

$$\alpha = \frac{8C_0}{(\gamma^2 - 1)(j + 2 - \omega)^2} \int_0^1 (gf^2 + h) \ \lambda^{j-1} \ d\lambda$$
(54)

where $C_0 = 2$ if j = 1, 2π if j = 2, or 4π if j = 3. It is judicious to convert this single integral over scaled position into two integrals over the similarity variable V (Kamm 2000), in part, to isolate singularities that may occur at the lower integration bounds:

$$J_1 = \int_{V_{min}}^{V_2} \frac{\gamma+1}{\gamma-1} \lambda^{j+1} g V^2 \frac{d\lambda}{dV} dV$$
(55)

$$J_{2} = \int_{V_{min}}^{V_{2}} \frac{8}{(\gamma+1)(j+2-\omega)^{2}} \lambda^{j+1} h \frac{d\lambda}{dV} dV$$
(56)

so that
$$\alpha = \begin{cases} \frac{1}{2}J_1 + \frac{1}{\gamma - 1}J_2 & \text{if } j = 1\\ (j - 1)\pi \left(J_1 + \frac{2}{\gamma - 1}J_2\right) & \text{if } j = 2, 3. \end{cases}$$
 (57)

The integrands can be readily evaluated from Sedov functions given in the previous sections.

If the solution is of the singular type, the equation (54) can be integrated analytically to yield

$$J_2 = \frac{\gamma + 1}{j[(\gamma - 1)j + 2]^2}$$
(58)

$$J_1 = \frac{2}{(\gamma - 1)} J_2 \tag{59}$$

$$\alpha = 2^{j-1}\pi J_2 av{60}$$

For either the standard or vacuum cases, however, the integrals must be evaluated numerically. In either case the upper integration limit is the post-shock location V_2 of equation (5). The lower integration limit in the standard case is the post-shock origin V_0 while in the vacuum case the lower limit is the vacuum boundary V_v :

$$V_0 = \frac{2}{(j+2-\omega)\gamma} \tag{61}$$

$$V_v = \frac{2}{(j+2-\omega)} . (62)$$

These lower limits of integration can, depending on (ω, j, γ) , make the integrands singular at the lower limit. Without specifically addressing the singularities, quadrature routines complain bitterly about inaccurate answers or too many iterations. Simply pushing the lower limit of integration away from the singularity can result in inaccurate answers and does not address the inefficiency. A correct treatment notes the singularities are integrable power-law singularities and, hence, removable. Removing the apparent singularities makes quadrature evaluations accurate, repeatable, and efficient. In the standard case, the J_1 integrand may be singular from the term

$$(cV-1)^{\alpha_3 - \alpha_2(j+2-\omega) - 1} \tag{63}$$

if the exponent is negative. Similarly, the J_2 integrand may be singular from the term

$$(cV-1)^{\alpha_3 - \alpha_2(j+2-\omega)-2} \tag{64}$$

if the exponent is negative. In the vacuum case, both J_1 and J_2 integrands may be singular from the term

$$\left(1 - \frac{cV}{\gamma}\right)^{\alpha_5} \tag{65}$$

if the exponent is negative. To address an integrand with an integrable singularity at its lower limit, one makes a change of variable (Press et al. 1996). If the integrand diverges as $(x-a)^p$, $0 \le p < 1$, near x = a, then use the identity

$$\int_{a}^{b} f(x) \, \mathrm{d}x = \frac{1}{1-p} \int_{0}^{(b-a)^{1-p}} t^{\frac{p}{1-p}} f\left(t^{\frac{1}{1-p}} + a\right) \, \mathrm{d}t \quad .$$
 (66)

If the integrand diverges as $(a - x)^p$, $0 \le p < 1$, near x = a, use the identity

$$\int_{a}^{b} f(x) \, \mathrm{d}x = \frac{1}{p-1} \int_{0}^{(a-b)^{1-p}} t^{\frac{p}{1-p}} f\left(a - t^{\frac{1}{1-p}}\right) \, \mathrm{d}t \quad . \tag{67}$$

These changes of variables can be made transparent to the user by defining suitable modules that change the variable automatically.

Figure 7 shows α for various values of γ as a function of ω for spherical geometry, obtained by numerical quadrature of the integrals in equation (57). For precision verification analysis of hydrodynamic codes, taking α to be "a constant of order unity" provides insufficient accuracy.

4. Generating Physical Solutions

With the Sedov functions and energy integrals in hand, physical solutions can be constructed. To generate a solution at the dimensional location r_{want} , locate the similarity value V^* that corresponds to r_{want} by seeking the value of zero of the function $f(V^*)$ given by

$$f(V^*) = r_2 \ \lambda - r_{\text{want}} \ . \tag{68}$$

With this derived value of V^* , compute the Sedov functions discussed in §2. This root-find is a key component of the solution method. Implementations of this root-find for the similarity variable in IEEE compliant 64 bit arithmetic (16 significant figures) run out of precision near the origin in the standard case or the transition region in the vacuum case. It is important to implement the root-find, and indeed the entire Sedov solution, in IEEE compliant 128 bit arithmetic (quad precision) to achieve accurate and robust solutions in these regions.

If a solution is desired ahead of the blast wave, $r_{want} \ge r_2$, then the physical solution is just that of the ambient material into which the disturbance propagates:

$$\rho = \rho_0 r^{-\omega} \tag{69}$$

$$v = v_0 \tag{70}$$

$$E = E_0 \tag{71}$$

$$P = P_0 \tag{72}$$

$$c = c_0 . (73)$$

If a solution between the origin and the shock front, $0 < r_{want} < r_2$, is desired, then

$$\rho = \rho_2 g \tag{74}$$

$$v = u_2 f \tag{75}$$

$$P = P_2 h \tag{76}$$

$$E = \frac{P}{\rho(\gamma - 1)} \tag{77}$$

$$c = \sqrt{\frac{P}{\rho\gamma}} . \tag{78}$$

4.1. Constant Initial Density Test Cases

The constant initial density case is frequently used in hydrocode verification tests, as the initial conditions are typically straightforward to prescribe in hydrocodes. This case also admits verification by comparison of values of the similarity variables λ , f, g, and hwith the values tabulated for $\gamma=1.4$ in Sedov (1959). Those published values appear to be tabulated using λ as the independent variable. Since the method we have outlined uses Vas the independent variable, we obtain solutions at Sedov's tabulated values by a root-find procedure described above. In Tables 1, 2 and 3 we provide the numerical values from the tables in Sedov (1959) as well as the values calculated with the code provided with this paper (denoted Exact). Most of these values match "exactly," i.e., to the four-digit mantissa quoted by Sedov (1959); the majority of the remaining values match to three significant figures; a few match to only two significant figures. The reason for the disagreements include (i) the limited precision for λ provided in Sedov (1959), (ii) inexactness in the root-finding routine used to match the published value of λ ; (iii) differences in the tolerances used in the numerical quadratures; and (iv) probable precision differences in the calculations.

The uniform density initial condition has been used for hydrocode comparison (Reile & Gehren 1991; Buchler et al. 1997; Owen et al. 1998; Caramana et al. 1998; Shashkov & Wendroff 1999; Fryxell et al. 2000), primarily in spherical geometry. Although these authors set the undisturbed uniform density to unity, each uses a different initial energy source; i.e., in the literature for this problem, there does not appear to be a standard initial configuration, \hat{a} la the Sod shock tube problem Sod (1978) or the Woodward-Colella blast wave problem Colella & Woodward (1984).

Therefore, we consider the following problems, based on those proposed by Klein & Boldstad (1999). Set the undisturbed uniform initial density to $\rho_0=1$ g/cm³ and $\omega=0$ in a $\gamma=1.4$ polytropic gas with initial energy $E_0=0.0673185$, 0.311357, and 0.851072 erg in planar, cylindrical, and spherical geometries, respectively. These values are chosen so that the shock is at r = 0.5, 0.75, and 1.0 cm in the planar, cylindrical, and spherical cases, respectively, at a final time of t = 1.0 s. The solution is computed at the centers of 120 equally sized zones on the domain between the origin and 1.2 cm. Key characteristics of the solutions for these problems are presented in Tables 4 and 5. Figure 8 plots the density, velocity, specific internal energy, and pressure for these three verification problems. These figures show the peak in the density and pressure immediately behind the shock, and the peak in specific internal energy near the origin. In each case, the pressure asymptotes to a nonzero value at the origin; since the density vanishes there, the specific internal energy grows without bound.

4.2. Variable Density Test Cases

The power-law initial density case is less frequently used in hydrocode verification tests, as it is less straightforward to set up the initial conditions in codes. Indeed, we have been unable to locate any hydrocode results using a variable densty in the refereed literature. We consider, therefore, two problems, the first of which is a singular case, and the second of which is a vacuum problem, the spherically symmetric version of which has been proposed by Klein & Boldstad (1999).

In the variable density case one must ensure that the hydrocode is initialized correctly.

In a finite volume code, for example, one must assign the initial cell density as the volume integral of the power-law density divided by the cell volume. In cylindrical and spherical geometries, this value of the initial cell density will likely not equal the point value of the power-law density at the cell center.

For the singular case, set $\gamma=1.4$, $\rho_0 = 1$ g/cm³, and $\omega = 5/3$ in the cylindrical case and $\omega = 7/3$ in the spherical case. Recall that there is no physically admissible singular solution in planar geometry. Set the initial energy deposited to 2.45749 erg in the cylindrical case and 4.90875 erg in the spherical case. These values are chosen so that the shock is at r = 0.75 cm in the cylindrical case and at r = 1.0 cm in the spherical case. As in the previous problems, the solution is computed at the center of 120 equally sized zones on the domain between the origin and 1.2 cm. Tables 6 and 7 present various key values as computed for these cases. Figure 9 contains plots of the density, velocity, specific internal energy, and pressure as functions of position for these two problems.

For the vacuum case, set $\gamma=1.4$, $\rho_0 = 1$ g/cm³, and ω to 1.7 in the cylindrical case and 2.4 in the spherical case. Set the initial energy deposited to 2.67315 erg in the cylindrical case and 5.45670 erg in the spherical case. The spherical case has been proposed by Klein & Boldstad (1999) as a hydrocode verification problem. As in the previous problems, the solution is computed at the center of 120 equally sized zones on the domain between the origin and 1.2 cm. Tables 8 and 9 list various key computed values for these cases. Figure 10 contains plots of the density, velocity, specific internal energy, and pressure as functions of position for these two problems.

5. Summary

We have described a procedure for producing robust numerical solutions of a Sedov blast wave propagating through polytropic gas with a power-law initial density. In addition to the "standard Sedov problem," we have included the solution to the singular and vacuum problems in both cylindrical and spherical geometries. We addressed all of the removable singularities, investigated the effects of finite-precision arithmetic on the ability to generate solutions, and offered an instantiation of these ideas in a succinct computer code.

We showed many examples of these solutions, both dimensionless and physical. We suggested that these problems can be used in the verification of hydrodynamics codes. We suspect that modern Godunov-type high-resolution methods perform adequately on the standard Sedov problem, but may perform inadequately on problems for which the pressure vanishes at or near the origin. Although the magnitude of the error in the vicinity of the origin diminishes under mesh refinement, we make the heuristic albeit plausible speculation that the hydrocode solutions may not be not converging to the exact solution for these problems. We speculate that a possible fundamental shortcoming of present hydrodynamic integration methods for these highly singular flows may be uncovered by a thorough investigation of the singular and vacuum Sedov verification problems that we have proposed.

Acknowledgments This work was supported in part by Jerry Brock and Joyce Guzik. We thank ZZZ for careful reviews and discussions. Los Alamos National Laboratory is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396.

6. Appendix A: Code Verification of a Sedov Solution

Here we give an example of a code verification analysis using the Sedov problem. We focus on one of the standard cases discussed in §4.1, namely $\rho_0=1$ g/cm³, $\omega=0$, $\gamma=1.4$, $E_{\text{blast}}=0.851072$ erg in spherical geometry. The shock is at r=1.0 cm at the final time of t = 1.0 s, and the solution is computed on the domain between the origin and 1.2 cm (Klein & Boldstad 1999). We use the RAGE radiation-hydrodynamic code (Gittings et al. 2006) to generate numerical solutions. RAGE is an adaptive mesh, parallel, Eulerian frame code similiar to other modern hydrocodes such as FLASH (Fryxell et al. 2000) or ZEUS-MP (Whalen & Norman 2006).

Initialization of a Sedov problem on a mesh will generate spirited debate whose antagonists are divided between depositing all the energy into a single central zone or depositing the energy in a small fixed size region. While the one-cell case is perhaps a more authentic way of initializing the problem, it is rarely seen in the refereed literature (Reile & Gehren 1991; Buchler et al. 1997; Fryxell et al. 2000), although see Swesty & Myra (2007).

Figure 11 shows the numerical solutions for the two cases along with the analytical solution at the final time of 1.0 s. The numerical solutions were generated on a 480 cell uniform mesh with a time-step controller of cstab=0.9. The parameter cstab sets the time-step based on the local sound speed and the material velocity, $\Delta t = \text{cstab} \cdot \Delta x / (c + \max(|v_x| + |v_y| + |v_z|))$, and determines the time-step in the numerical solution of the Sedov problem. The single cell initialization ran to completion in 90000 timesteps, while the small fixed region (energy uniformly distributed within 0.02 cm) initialization took 10000 timesteps.

Figure 11 shows that for r > 0.3 the differences between the specific internal energy for the two cases is small and generally agrees with the analytic solution. For r < 0.3 the differences between the two cases become substantially worse as the origin is approached. Neither case agrees with the analytic solution, although the single cell initialization is closer. Similar comments hold for the density. In contrast, the pressure and material speed solutions for the two cases generally agree with the analytic solution as $r \to 0$.

Rates of spatial convergence may be determined be equating an error metric to a model for the error (Kamm, Rider & Brock 2002). Here we consider the error metric to be the absolute L_1 norm

$$L_1 = \frac{\sum |f_i^{\text{exact}} - f_i^{\text{rage}}| V_i}{\sum V_i}$$
(79)

where f is a field quantity of interest (e.g., the density), and V_i is the appropriate volume element weighting. Other error metrics, such as the relative L_1 norm or the absolute L_2 norm could be used. For cases where the time-step controller is held constant and the spatial resolution varied, we could model the L_1 error norm as a power-law function of the grid spacing;

$$L_1 = A \left(\Delta x\right)^q \quad , \tag{80}$$

where Δx is the cell spacing and q is the spatial convergence rate (Kamm, Rider & Brock 2002). In this case, the rate of convergence between two grids, one coarse and one fine, is given explicitly by

$$q = \log\left[\frac{L_{1,\text{fine}}}{L_{1,\text{coarse}}}\right] / \log\left[\frac{\Delta x_{\text{fine}}}{\Delta x_{\text{coarse}}}\right]$$
(81)

The error model above follows from a modified-equation analysis which is typically done in terms of length scales (not volumes). Other error models, such oscillatory convergence or a polynomial in both cell size and time step, may be more suitable for a given situation.

It should be noted that the spatial discretization errors, temporal discretization errors, or coupled space-time errors may change with simulation time during a numerical computation of the Sedov problem. Although the solution is self-similar, various physical effects are exercised in different proportions during an evolution, so the dominant contributor to the overall numerical error may not remain the same (Hemez 2005). For example, the effects of time discretization on a hydrodynamic simulation may be more pronounced early in the evolution. Likewise, inadequate spatial discretization at some instants of the simulation may be replaced as the dominant source of solution error by truncation errors at other times. These remarks imply that convergence rates in equation 80 may be functions of spacetime. To keep the present study practical, we consider only the global code verification properties at the ending time of a test problem's evolution.

Figure 12 shows the absolute value of the relative errors in the density, pressure, specific internal energy, and material speed for one-dimensional uniform grids with 120, 240, 480, 960, 1920, and 3840 cells at the final time of 1.0 s. The single cell initialization procedure was used. The time-step controller was kept at cstab=0.9. The relative cpu cost on a single processor of increasing the spatial resolution is given. Note that doubling the number of cells increases the single processor cpu by factors of three to four. The singularity at the origin means the temperature grows without bound, implying large errors in the specific energy near the origin. With the exception of the specific internal energy, there is a steady decline in the magnitude of the errors between the origin and the shock front as the spatial resolution is increased. Figure 12 and Table 10 show that the density, pressure, and material speed have roughly linear global spatial convergence rates ($q \sim 1$), while the specific internal energy has a near zero global spatial convergence rate (because of the large persistent errors at the origin). For the fixed region initialization procedure we find the same convergence rates to within two significant figures.

7. Appendix B: Computation of a Sedov Solution

This appendix lists six key routines necessary to generate robust Sedov solutions. Common functionality such as quadrature or root-find routines are not listed.

- 1. test_sedov_solver example of how to use the routines.
- 2. sed_1d drives the generation of physical solutions.
- 3. sedov_funcs generates the Sedov functions.
- 4. efun01 integrand of the first energy integral; to be called by a quadrature routine.
- 5. efun02 integrand of the second energy integral; to be called by a quadrature routine.
- 6. sed_v_find given the distances, find the similarity variable; to be called by a root-finder.

program test_sedov_solver
implicit none

```
c..exercises the sedov solver
```

c..declare

outfile, string
i,nstep,iargc,nmax
(nmax = 1000)
<pre>time,zpos(nmax),</pre>
<pre>eblast,rho0,omega,vel0,ener0,pres0,cs0,gamma,</pre>
xgeom,
<pre>den(nmax),ener(nmax),pres(nmax),vel(nmax),</pre>
cs(nmax),
zlo,zhi,zstep,value

c..popular formats

```
format(1x,t4,a,t8,a,t22,a,t36,a,t50,a,t64,a,t78,a,t92,a)
01
02
     format(1x,i4,1p8e12.4)
     format(1x, i4, 1p8e14.6)
03
c..explicitly set some parameters
c..standard cases
c...spherical constant density should reach r=1 at t=1
     nstep = 120
     eblast = 0.851072q0
     xgeom = 3.0q0
     omega = 0.0q0
     outfile = 'spherical_standard_omega0p00.dat'
c..input parameters in cgs
     time = 1.0q0
     rho0 = 1.0q0
     vel0 = 0.0q0
     ener0 = 0.0q0
```

```
cs0 = 0.0q0
gamma = 1.4q0
```

pres0 = 0.0q0

```
c..number of grid points, spatial domain, spatial step size
c..to match hydrocode output, use the mid-cell points
      zlo
            = 0.0q0
      zhi
            = 1.2q0
     zstep = (zhi - zlo)/float(nstep)
      do i=1,nstep
      zpos(i)
                 = zlo + 0.5q0*zstep + float(i-1)*zstep
      enddo
c..get the solution for all spatial points at once
       call sed_1d(time,nstep,zpos,
     1
                   eblast,omega,xgeom,
     2
                   rho0,vel0,ener0,pres0,cs0,gamma,
     3
                   den,ener,pres,vel,cs)
c..output file
      open(unit=2,file=outfile,status='unknown')
      write(2,02) nstep,time
      write(2,01) 'i','x','den','ener','pres','vel','cs'
      do i=1,nstep
      write(2,03) i,zpos(i),den(i),ener(i),pres(i),vel(i),cs(i)
      enddo
      close(unit=2)
c..close up shop
      end
```

```
subroutine sed_1d(time,nstep,xpos,
     1
                        eblast,omega_in,xgeom_in,
     2
                        rho0,vel0,ener0,pres0,cs0,gam0,
     3
                        den,ener,pres,vel,cs)
      implicit none
c..this routine produces 1d solutions for a sedov blast wave propagating
c..through a density gradient rho = rho**(-omega)
c.. in planar, cylindrical or spherical geometry
c.. for the standard, singular and vaccum cases.
c..standard case: a nonzero solution extends from the shock to the origin,
с..
                  where the pressure is finite.
c..singular case: a nonzero solution extends from the shock to the origin,
                  where the pressure vanishes.
с..
c..vacuum case : a nonzero solution extends from the shock to a boundary point,
                  where the density vanishes making the pressure meaningless.
с..
c..input:
           = temporal point where solution is desired seconds
c..time
c..xpos(i) = spatial points where solution is desired cm
c..eblast = energy of blast erg
           = ambient density g/cm**3
                                        rho = rho0 * r**(-omega_in)
c..rho0
c..omegain = density power law exponent rho = rho0 * r**(-omega_in)
c..vel0
         = ambient material speed cm/s
c..pres0
           = ambient pressure erg/cm**3
            = ambient sound speed cm/s
c..cs0
c..gamO
         = gamma law equation of state
c..xgeom_in = geometry factor, 3=spherical, 2=cylindircal, 1=planar
c.. for efficiency reasons (doing the energy integrals only once),
c..this routine returns the solution for an array of spatial points
c..at the desired time point.
c..output:
```

c..den(i) = density g/cm**3

```
c..ener(i) = specific internal energy erg/g
c..pres(i) = presssure erg/cm**3
c..vel(i) = velocity cm/s
c..cs(i)
           = sound speed cm/s
c..although the ordinary differential equations are analytic,
c..the sedov expressions appear to become singular for various
c..combinations of parameters and at the lower limits of the integration
c..range. all these singularies are removable and done so by this routine.
c..these routines are written in real*16 precision because the
c..real*8 implementations simply run out of precision "near" the origin
c..in the standard case or the transition region in the vacuum case.
c..declare the pass
      integer
                       nstep
      real*16
                       time,xpos(*),
     1
                       eblast, rho0, omega_in, vel0, ener0, pres0, cs0,
                       gam0,xgeom_in,den(*),ener(*),pres(*),
     1
     3
                       vel(*),cs(*)
c..local variables
      external
                       midpnt,midpowl,midpowl2,sed_v_find,sed_r_find,
     1
                       efun01,efun02
      integer
                       i
                       efun01,efun02,eval1,eval2
      real*16
      real*16
                       v0,v2,vstar,vmin,midpnt,midpowl,midpowl2,
     1
                       alpha, vstep, us, u2, rho2, p2, e2, cs2,
     2
                       zeroin,sed_v_find,sed_r_find,
     3
                       vat,l_fun,dlamdv,f_fun,g_fun,h_fun,
     4
                       denom2,denom3,rho1
c..eps controls the integration accuracy, don't get too greedy or the number
c.. of function evaluations required kills.
c..eps2 controls the root find accuracy
```

c..osmall controls the size of transition regions

real*16 iprint,eps,eps2,osmall,pi

parameter	(iprint = 1,
1	eps = 1.0q-10,
2	eps2 = 1.0q-30,
3	osmall = 1.0q-4,
4	pi = 3.1415926535897932384626433832795029q0)

```
c..common block communication
```

```
logical
                  lsingular,lstandard,lvacuum,lomega2,lomega3
real*16
                  gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
1
                  a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
2
                  omega,vv,xlam_want,vwant,rvv
common /slap/
                  gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
                  a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
1
2
                  omega,vv,xlam_want,vwant,rvv,
3
                  lsingular,lstandard,lvacuum,lomega2,lomega3
```

```
c..common block communication with the integration stepper
    real*16         gam_int
    common /cmidp/ gam_int
```

```
c..popular formats
```

```
87 format(1x,1p10e14.6)
```

```
88 format(1x,8(a7,1pe14.6,' '))
```

```
c..initialize the solution
do i=1,nstep
den(i) = 0.0q0
vel(i) = 0.0q0
pres(i) = 0.0q0
ener(i) = 0.0q0
cs(i) = 0.0q0
end do
```

c..transfer the pass to common block and create some frequent combinations

```
gamma = gam0
      gamm1 = gamma - 1.0q0
      gamp1 = gamma + 1.0q0
      gpogm = gamp1 / gamm1
      xgeom = xgeom_in
      omega = omega_in
      xg2
            = xgeom + 2.0q0 - omega
      denom2 = 2.0q0*gamm1 + xgeom - gamma*omega
      denom3 = xgeom * (2.0q0 - gamma) - omega
c..post shock location v2 and location of singular point vstar
c..kamm & timmes equations 5-6
      v2
           = 4.0q0 / (xg2 * gamp1)
      vstar = 2.0q0 / (gamm1*xgeom + 2.0q0)
c..set two logicals that determines the type of solution
      lstandard = .false.
      lsingular = .false.
      lvacuum = .false.
      if (abs(v2 - vstar) .le. osmall) then
      lsingular = .true.
       if (iprint .eq. 1) write(6,*) 'singular'
      else if (v2 .lt. vstar - osmall) then
       lstandard = .true.
       if (iprint .eq. 1) write(6,*) 'standard'
      else if (v2 .gt. vstar + osmall) then
       lvacuum = .true.
       if (iprint .eq. 1) write(6,*) 'vacuum'
      end if
c..two apparent singularies, book's notation for omega2 and omega3
```

```
lomega2 = .false.
lomega3 = .false.
if (abs(denom2) .le. osmall) then
lomega2 = .true.
```

```
denom2 = 1.0q-8
       if (iprint .eq. 1) write(6,*) 'omega2 case'
      else if (abs(denom3) .le. osmall) then
      lomega3 = .true.
      denom3 = 1.0q-8
      if (iprint .eq. 1) write(6,*) 'omega3 case'
      end if
c..various exponents, kamm & timmes equations 19-24
     a0 = 2.0q0/xg2
     a2 = -gamm1/denom2
     a1 = xg2*gamma/(2.0q0 + xgeom*gamm1) *
     1
           (((2.0q0*(xgeom*(2.0q0-gamma) - omega))/(gamma*xg2*xg2))-a2)
     a3 = (xgeom - omega) / denom2
     a4 = xg2 * (xgeom - omega) * a1 / denom3
     a5 = (omega*gamp1 - 2.0q0*xgeom)/denom3
c..frequent combinations, kamm & timmes equations 25-29
     a_val = 0.25q0 * xg2 * gamp1
     b_val = gpogm
     c_val = 0.5q0 * xg2 * gamma
     d_val = (xg2 * gamp1)/(xg2*gamp1 - 2.0q0*(2.0q0 + xgeom*gamm1))
     e_val = 0.5q0 * (2.0q0 + xgeom * gamm1)
c..evaluate the energy integrals
c..the singular case can be done by hand; save some cpu cycles
c..kamm & timmes equations 58-60
     if (lsingular) then
      eval2 = gamp1/(xgeom*(gamm1*xgeom + 2.0q0)**2)
      eval1 = 2.0q0/gamm1 * eval2
      alpha = gpogm * 2**(xgeom)/(xgeom*(gamm1*xgeom + 2.0q0)**2)
       if (int(xgeom) .ne. 1) alpha = pi * alpha
```

```
c..for the standard or vacuum cases
```

```
c..v0 = post-shock origin v0 and vv = vacuum boundary vv
c..set the radius coresponding to vv to zero for now
c..kamm & timmes equations 61-62
      else
       v0 = 2.0q0 / (xg2 * gamma)
      vv = 2.0q0 / xg2
       rvv = 0.0d0
       if (lstandard) vmin = v0
       if (lvacuum) vmin = vv
c..the first energy integral
c..in the standard case the term (c_val*v - 1) might be singular at v=vmin
       if (lstandard) then
        gam_int = a3 - a2*xg2 - 1.0q0
        if (gam_int .ge. 0) then
         call qromo(efun01,vmin,v2,eps,eval1,midpnt)
        else
         gam_int = abs(gam_int)
         call qromo(efun01,vmin,v2,eps,eval1,midpowl)
        end if
c..in the vacuum case the term (1 - c_val/gamma*v) might be singular at v=vmin
       else if (lvacuum) then
        gam_int = a5
        if (gam_int .ge. 0) then
         call qromo(efun01,vmin,v2,eps,eval1,midpnt)
        else
         gam_int = abs(gam_int)
         call qromo(efun01,vmin,v2,eps,eval1,midpowl2)
        end if
       end if
c..the second energy integral
c..in the standard case the term (c_val*v - 1) might be singular at v=vmin
```

```
if (lstandard) then
        gam_int = a3 - a2*xg2 - 2.0q0
        if (gam_int .ge. 0) then
         call qromo(efun02,vmin,v2,eps,eval2,midpnt)
        else
         gam_int = abs(gam_int)
         call qromo(efun02,vmin,v2,eps,eval2,midpowl)
        end if
c..in the vacuum case the term (1 - c_val/gamma*v) might be singular at v=vmin
       else if (lvacuum) then
        gam_int = a5
        if (gam_int .ge. 0) then
         call qromo(efun02,vmin,v2,eps,eval2,midpnt)
        else
         gam_int = abs(gam_int)
         call qromo(efun02,vmin,v2,eps,eval2,midpowl2)
        end if
       end if
c..kamm & timmes equation 57 for alpha
       if (int(xgeom) .eq. 1) then
        alpha = 0.5q0*eval1 + eval2/gamm1
       else
        alpha = (xgeom - 1.0q0) * pi * (eval1 + 2.0q0 * eval2/gamm1)
       end if
      end if
c..write what we have for the energy integrals
      if (iprint .eq. 1)
     1
          write(6,88) 'xgeom =',xgeom,'eblast=',eblast,
     2
                      'omega =',omega,'alpha =',alpha,
     3
                      'j1 =',eval1,'j2 =',eval2
```

```
c..immediate post-shock values
```

```
c..kamm & timmes equations 12-18
c..r2 = shock position, u2 = shock speed, rho1 = pre-shock density,
c..u2 = post-shock material speed, rho2 = post-shock density,
c..p2 = post-shock pressure, e2 = post-shoock specific internal energy,
c..and cs2 = post-shock sound speed
          = (eblast/(alpha*rho0))**(1.0q0/xg2) * time**(2.0q0/xg2)
     r2
          = (2.0q0/xg2) * r2 / time
     us
     rho1 = rho0 * r2**(-omega)
     u2
          = 2.0q0 * us / gamp1
     rho2 = gpogm * rho1
     p2
          = 2.0q0 * rho1 * us**2 / gamp1
     e2
          = p2/(gamm1*rho2)
     cs2 = sqrt(gamma*p2/rho2)
c..find the radius corresponding to vv
      if (lvacuum)
                     then
       vwant = vv
       rvv = zeroin(0.0q0,r2,sed_r_find,eps2)
      end if
     if (lstandard .and. iprint .eq. 1)
          write(6,88) 'r2 =',r2,'rho2 =',rho2,
    1
    2
                      'u2 =',u2,'e2 =',e2,
                      'p2 =',p2,'cs2 =',cs2
     3
     if (lvacuum .and. iprint .eq. 1)
    1
          write(6,88)
    2
                      'rv =',rvv,
    3
                      'r2 =',r2,'rho2 =',rho2,
                      'u2 =',u2,'e2 =',e2,
    4
    5
                      'p2 =',p2,'cs2 =',cs2
```

c..now start the loop over spatial positions
 do i=1,nstep

```
rwant = xpos(i)
c.. if we are upstream from the shock front
       if (rwant .gt. r2) then
       den(i) = rho0 * rwant**(-omega)
       vel(i) = vel0
       pres(i) = pres0
       ener(i) = ener0
       cs(i) = cs0
c..if we are between the origin and the shock front
c..find the correct similarity value for this radius in the standard or vacuum cases
      else
        if (lstandard) then
        vat = zeroin(0.90q0*v0,v2,sed_v_find,eps2)
       else if (lvacuum) then
        vat = zeroin(v2,1.2q0*vv,sed_v_find,eps2)
       end if
c..the physical solution
       call sedov_funcs(vat,l_fun,dlamdv,f_fun,g_fun,h_fun)
       den(i) = rho2 * g_fun
       vel(i)
               = u2
                      * f_fun
       pres(i) = p2 * h_fun
       ener(i) = 0.0q0
       cs(i)
                = 0.0q0
       if (den(i) .ne. 0.0) then
        ener(i) = pres(i) / (gamm1 * den(i))
                 = sqrt(gamma * pres(i)/den(i))
        cs(i)
       end if
      end if
c..end of loop over positions
      enddo
      return
      end
```

```
subroutine sedov_funcs(v,l_fun,dlamdv,f_fun,g_fun,h_fun)
      implicit none
      save
c..given the similarity variable v, returns functions
c..lambda, f, g, and h and the derivative of lambda with v dlamdv
c..although the ordinary differential equations are analytic,
c..the sedov expressions appear to become singular for various
c..combinations of parameters and at the lower limits of the integration
c..range. all these singularies are removable and done so by this routine.
c..declare the pass
      real*16
                       v,l_fun,dlamdv,f_fun,g_fun,h_fun
c..common block communication
      logical
                       lsingular,lstandard,lvacuum,lomega2,lomega3
      real*16
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv
      common /slap/
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv,
     3
                       lsingular,lstandard,lvacuum,lomega2,lomega3
c..local variables
      real*16
                       x1, x2, x3, x4, dx1dv, dx2dv, dx3dv, dx4dv,
     1
                       cbag,ebag,beta0,pp1,pp2,pp3,pp4,c2,c6,y,z,
     2
                       dpp2dv,eps
                       (eps = 1.0q-30)
      parameter
С
                        (eps = 0.0q0)
       parameter
c..frequent combinations and their derivative with v
c..kamm & timmes equation 30-33
       x1 = a_val * v
```

 $dx1dv = a_val$

```
cbag = max(eps, c_val * v - 1.0q0)
       x2 = b_val * cbag
       dx2dv = b_val * c_val
       ebag = 1.0q0 - e_val * v
       x3 = d_val * ebag
       dx3dv = -d_val * e_val
       x4 = b_val * (1.0q0 - 0.5q0 * xg2 *v)
       dx4dv = -b_val * 0.5q0 * xg2
c..transition region between standard and vacuum cases
c..kamm & timmes equations 48-52
      if (lsingular) then
      l_fun = rwant/r2
      dlamdv = 0.0q0
      f_fun = l_fun
      g_fun = 1_fun **(xgeom - 2.0q0)
      h_fun = l_fun**xgeom
c.. for the vacuum case in the hole
c..kamm & timmes equation 53
      else if (lvacuum .and. rwant .lt. rvv) then
      1_{fun} = 0.0q0
       dlamdv = 0.0q0
       f_fun = 0.0q0
       g_fun = 0.0q0
      h_fun = 0.0q0
```

c..omega = omega2 = (2*(gamma -1) + xgeom)/gamma case, denom2 = 0

```
c..kamm & timmes equations 39-43
     else if (lomega2) then
      beta0 = 1.0q0/(2.0q0 * e_val)
      pp1 = gamm1 * beta0
      c6
           = 0.5q0 * gamp1
      c2
           = c6/gamma
           = 1.0q0/(x1 - c2)
      у
      z = (1.0q0 - x1)*y
      pp2 = gamp1 * beta0 * z
      dpp2dv = -gamp1 * beta0 * dx1dv * y * (1.0q0 + z)
      pp3
           = (4.0q0 - xgeom - 2.0q0*gamma) * beta0
      pp4 = -xgeom * gamma * beta0
      l_fun = x1**(-a0) * x2**(pp1) * exp(pp2)
      dlamdv = (-a0*dx1dv/x1 + pp1*dx2dv/x2 + dpp2dv) * 1_fun
      f_fun = x1 * l_fun
      g_fun = x1**(a0*omega) * x2**pp3 * x4**a5 * exp(-2.0q0*pp2)
      h_fun = x1**(a0*xgeom) * x2**pp4 * x4**(1.0q0 + a5)
c..omega = omega3 = xgeom*(2 - gamma) case, denom3 = 0
c..kamm & timmes equations 44-47
     else if (lomega3) then
      beta0 = 1.0q0/(2.0q0 * e_val)
      pp1 = a3 + omega * a2
      pp2 = 1.0q0 - 4.0q0 * beta0
      c6
           = 0.5q0 * gamp1
      pp3 = -xgeom * gamma * gamp1 * beta0 * (1.0q0 - x1)/(c6 - x1)
            = 2.0q0 * (xgeom * gamm1 - gamma) * beta0
      pp4
      l_fun = x1**(-a0) * x2**(-a2) * x4**(-a1)
      dlamdv = -(a0*dx1dv/x1 + a2*dx2dv/x2 + a1*dx4dv/x4) * l_fun
      f_fun = x1 * l_fun
```

```
– 31 –
```

```
g_fun = x1**(a0*omega) * x2**pp1 * x4**pp2 * exp(pp3)
h_fun = x1**(a0*xgeom) * x4**pp4 * exp(pp3)
c..for the standard or vacuum case not in the hole
c..kamm & timmes equations 34-38
else
    l_fun = x1**(-a0) * x2**(-a2) * x3**(-a1)
    dlamdv = -(a0*dx1dv/x1 + a2*dx2dv/x2 + a1*dx3dv/x3) * 1_fun
    f_fun = x1 * 1_fun
    g_fun = x1**(a0*omega)*x2**(a3+a2*omega)*x3**(a4+a1*omega)*x4**a5
    h_fun = x1**(a0*xgeom)*x3**(a4+a1*(omega-2.0q0))*x4**(1.0q0 + a5)
    end if
    return
    end
```

– 32 –

```
real*16 function efun01(v)
      implicit none
      save
c..evaluates the first energy integrand, kamm & timmes equation 55.
c..the (c_val*v - 1) term might be singular at v=vmin in the standard case.
c..the (1 - c_val/gamma * v) term might be singular at v=vmin in the vacuum case.
c..due care should be taken for these removable singularities by the integrator.
c..declare the pass
      real*16 v
c..common block communication
                       lsingular,lstandard,lvacuum,lomega2,lomega3
      logical
     real*16
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv
     common /slap/
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv,
     3
                       lsingular,lstandard,lvacuum,lomega2,lomega3
```

```
c..local variables
```

real*16 l_fun,dlamdv,f_fun,g_fun,h_fun

```
c..go
```

```
call sedov_funcs(v,l_fun,dlamdv,f_fun,g_fun,h_fun)
efun01 = dlamdv * l_fun**(xgeom + 1.0q0) * gpogm * g_fun * v**2
```

return end

```
real*16 function efun02(v)
      implicit none
      save
c..evaluates the second energy integrand, kamm & timmes equation 56.
c..the (c_val*v - 1) term might be singular at v=vmin in the standard case.
c..the (1 - c_val/gamma * v) term might be singular at v=vmin in the vacuum case.
c..due care should be taken for these removable singularities by the integrator.
c..declare the pass
      real*16 v
c..common block communication
      logical
                       lsingular,lstandard,lvacuum,lomega2,lomega3
     real*16
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv
     common /slap/
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv,
     3
                       lsingular,lstandard,lvacuum,lomega2,lomega3
c..local variables
      real*16 l_fun,dlamdv,f_fun,g_fun,h_fun,z
c..go
      call sedov_funcs(v,l_fun,dlamdv,f_fun,g_fun,h_fun)
      z = 8.0q0/((xgeom + 2.0q0 - omega)**2 * gamp1)
      efun02 = dlamdv * l_fun**(xgeom - 1.0q0 ) * h_fun * z
      return
      end
```

```
real*16 function sed_v_find(v)
      implicit none
      save
c..given corresponding physical distances, find the similarity variable v
c..kamm & timmes equation 68
c..declare the pass
      real*16 v
c..common block communication
      logical
                       lsingular,lstandard,lvacuum,lomega2,lomega3
     real*16
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
     1
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     2
                       omega,vv,xlam_want,vwant,rvv
     common /slap/
                       gamma,gamm1,gamp1,gpogm,xgeom,xg2,rwant,r2,
                       a0,a1,a2,a3,a4,a5,a_val,b_val,c_val,d_val,e_val,
     1
     2
                       omega,vv,xlam_want,vwant,rvv,
     3
                       lsingular,lstandard,lvacuum,lomega2,lomega3
c..local variables
      real*16 l_fun,dlamdv,f_fun,g_fun,h_fun
      call sedov_funcs(v,l_fun,dlamdv,f_fun,g_fun,h_fun)
      sed_v_find = r2*l_fun - rwant
      return
      end
```

REFERENCES

- Bernstein, I.B., & Book, D.L., ApJ, 225, 633, 1978
- Bernstein, I.B., & Book, D.L., ApJ, 240, 223, 1980
- Bethe, H.A., Fuchs, K., Hirschfelder, J.O., Magee, J.L., Peierls, R.E., & von Neumann, J., Blast Wave, Los Alamos Scientific Laboratory Report LA-2000, August 1947.
- Book, D.L., Shock Waves, 4, 1, 1994
- Buchler, J.R., Kolláth, Z., & Marom, A., Astrophys. Space Sci., 253, 139, 1997
- Caramana, E.J., Shashkov, M.J., & Whalen, P.P., J. Comp. Phys, 144, 70, 1998
- Chevalier, R.A., ApJ, 207, 872, 1976.
- Dursi, L.J., & Timmes, F.X., ApJ, 641, 1071, 2006
- Fryxell, B., Olson, K., Ricker, P. and Timmes, F.X., Zingale, M., Lamb, D.Q., MacNeice, P., Rosner, R., Truran, J.W., & Tufo, H., ApJS, 131, 273, 2000
- Gittings, M., Weaver, R., Clover, M., Betlach, T., Byrne, N., Coker, R., Dendy, E., Hueckstaedt, R., New, K., Oakes, W.R., Ranta, R., & Stefan, R., Los Alamos Unclassified Report 06-0027, 2006
- Hansen, J.F., Edwards, M.J., Froula, D., Gregori, G., Edens, A., & Ditmire, T., Astro. Space Sci. 298, 61, 2005
- Hemez, F.M., Los Alamos Unclassified Report 05-8228, 2005
- Kamm, J.R., Los Alamos Unclassified Report 00-6055, 2000
- Kamm, J.R., Rider, W.J., Brock, J.S., Los Alamos Unclassified Report 02-3794, 2002
- Kidder, R.E., Nucl. Fusion 14, 53, 1974
- Kidder, R.E., Nucl. Fusion 16, 3, 1976
- Klein, R., & Bolstad, J., Lawrence Livermore National Laboratory, private communication
- Korobeilnikov, V.P., Mel'nikova, N.S., & Ryazanov, Ye.V., Teoriya Tochechnogo Vzryva, FizMatLit, Leningrad, 1961. [Transl.: The theory of point explosion, JPRS 14, 334, U.S. Dept. of Commerce, Washington, DC, 1962], Chap. 2

- Lifshitz, E., Journal of Physics (USSR), 10, 116, 1946
- Menikoff, R. & Plohr, B.J., Rev. Mod. Phys., 61, 75, 1989
- Oppenheim, A.K., Knhi, A.L., Lundstrom, E.A., & Kamel, M.M., J. Fluid Mech., 52, 657, 1972
- O'Shea, B.W., Bryan, G., Bordner, J., Norman, M.L., Abel, T., Harkness, R., & Kritsuk, A., Adaptive Mesh Refinement - Theory and Applications, Eds. Plewa, T., Linde, T. & Weirs, V.G., Springer Lecture Notes in Computational Science and Engineering, 2004
- Owen, J.M., Villumsen, J.V., Shapiro, P.R., & Martel, H., ApJS, 116, 155, 1998
- Press, W.H., Teukolsky, S.A., Vetterling, W.T., & Flannery, B.P., Numerical Recipes in Fortran 90, Cambridge, Cambridge Univ. Press, 1996
- Reile, C., & Gehren, T., A&A, 242, 142, 1991
- Reinicke, R., & Meyer-ter-Vehn, J., Phys. Fluids A, 3, 7, 1991
- Remington, B.A., Arnett, D., Drake, R.P., & Takabe, H., Science, 284, 1488, 1999
- Ryu, D., & Vishniac, E.T., ApJ, 313, 820, 1987
- Sedov, L.I., Dokl. Akad. Nauk. SSSR, 52, 17, 1946
- Sedov, L.I., Similarity and Dimensional Methods in Mechanics, Academic Press, New York, NY, 1959
- Shashkov, M., & Wendroff, B., J. Comp. Phys, 150, 502, 1999
- Shestakov, A.I., Phys. Fluids, 11, 1091, 1999
- Sod, G., J. Comp. Phys, 27, 1, 1978
- Swesty, F.D., & Myra, E.S., ApJ, submitted, 2007
- Taylor, G.I., Brit Rept RC-210. Reprinted in Proc. Roy. Soc. A, 186, 159, 1950
- Teyssier, R., A&A, 385, 337, 2002
- Vishniac, E.T., & Ryu, D., ApJ, 337, 917, 1989
- von Neumann, J., NDRC Division B Rept AM-9. Reprinted in *Collected Works of John von Neumann* (Ed. Taub, A.H.) Pergamon, New York, 1963

Whalen, D. & Norman, M. L. ApJS, 162, 281, 2006

Whitham, G.B., Linear and Nonlinear Waves, John Wiley & Sons, New York, NY, 1974

Colella, P., & Woodward, P.R., J. Comp. Phys., 54, 174, 1984

This preprint was prepared with the AAS ${\rm IAT}_{\rm E}{\rm X}$ macros v5.2.

Table 1. Sedov Functions for $\gamma = 1.4$, planar geometry case^{*a*}

λ	V	Sedov f	Exact f	Sedov g	Exact g	Sedov h	Exact h
0.9797	0.5500	0.9699	0.9699	0.8625	0.8620	0.9162	0.9159
0.9420	0.5400	0.9156	0.9157	0.6659	0.6662	0.7915	0.7917
0.9013	0.5300	0.8599	0.8598	0.5160	0.5159	0.6923	0.6922
0.8565	0.5200	0.8017	0.8017	0.3982	0.3981	0.6120	0.6119
0.8050	0.5100	0.7390	0.7390	0.3019	0.3020	0.5457	0.5458
0.7419	0.5000	0.6678	0.6677	0.2200	0.2201	0.4904	0.4905
0.7029	0.4950	0.6263	0.6263	0.1823	0.1823	0.4661	0.4661
0.6553	0.4900	0.5780	0.5780	0.1453	0.1453	0.4437	0.4437
0.5925	0.4850	0.5172	0.5173	0.1074	0.1075	0.4229	0.4230
0.5396	0.4820	0.4682	0.4682	0.0826	0.0826	0.4116	0.4112
0.4912	0.4800	0.4244	0.4244	0.0641	0.0641	0.4038	0.4037
0.4589	0.4790	0.3957	0.3957	0.0536	0.0535	0.4001	0.4001
0.4161	0.4780	0.3580	0.3580	0.0415	0.0415	0.3964	0.3964
0.3480	0.4770	0.2988	0.2988	0.0263	0.0263	0.3929	0.3929
0.2810	0.4765	0.2410	0.2410	0.0153	0.0153	0.3911	0.3911
0.2320	0.4763	0.1989	0.1989	0.0095	0.0095	0.3905	0.3905
0.1680	0.4762	0.1441	0.1440	0.0042	0.0042	0.3901	0.3901
0.1040	0.4762	0.0891	0.0891	0.0013	0.0013	0.3900	0.3900

^aThe Sedov values are those published in Sedov (1959). The exact values are those calculated with the present solution method. Values that differ are highlighted.

λ	V	Sedov f	Exact f	Sedov g	Exact g	Sedov h	Exact h
0.9998	0.4166	0.9996	0.9996	0.9973	0.9972	0.9985	0.9984
0.9802	0.4100	0.9645	0.9645	0.7653	0.7651	0.8659	0.8658
0.9644	0.4050	0.9374	0.9374	0.6285	0.6281	0.7832	0.7829
0.9476	0.4000	0.9097	0.9097	0.5164	0.5161	0.7124	0.7122
0.9295	0.3950	0.8812	0.8812	0.4234	0.4233	0.6514	0.6513
0.9096	0.3900	0.8514	0.8514	0.3451	0.3450	0.5983	0.5982
0.8725	0.3820	0.7998	0.7999	0.2427	0.2427	0.5266	0.5266
0.8442	0.3770	0.7638	0.7638	0.1892	0.1892	0.4884	0.4884
0.8094	0.3720	0.7226	0.7226	0.1414	0.1415	0.4545	0.4545
0.7629	0.3670	0.6720	0.6720	0.0975	0.0974	0.4242	0.4241
0.7242	0.3640	0.6327	0.6327	0.0718	0.0718	0.4074	0.4074
0.6894	0.3620	0.5989	0.5990	0.0545	0.0545	0.3969	0.3969
0.6390	0.3600	0.5521	0.5521	0.0362	0.0362	0.3867	0.3867
0.5745	0.3585	0.4943	0.4943	0.0208	0.0208	0.3794	0.3794
0.5180	0.3578	0.4448	0.4448	0.0123	0.0123	0.3760	0.3760
0.4748	0.3575	0.4073	0.4074	0.0079	0.0079	0.3746	0.3746
0.4222	0.3573	0.3621	0.3620	0.0044	0.0044	0.3737	0.3737
0.3654	0.3572	0.3133	0.3133	0.0021	0.0021	0.3733	0.3732
0.3000	0.3572	0.2571	0.2572	0.0008	0.0008	0.3730	0.3730
0.2500	0.3571	0.2143	0.2143	0.0003	0.0003	0.3729	0.3729
0.2000	0.3571	0.1714	0.1714	0.0001	0.0001	0.3729	0.3729
0.1500	0.3571	0.1286	0.1286	0.0000	0.0000	0.3729	0.3729
0.1000	0.3571	0.0857	0.0857	0.0000	0.0000	0.3729	0.3729

Table 2. Sedov Functions for $\gamma = 1.4$, cylindrical geometry case^a

^aThe Sedov values are those published in Sedov (1959). The exact values are those calculated with the present solution method. Values that differ are high-lighted.

Table 3. Sedov Functions for $\gamma = 1.4$, spherical geometry case^a

λ	V	Sedov f	Exact f	Sedov g	Exact g	Sedov h	Exact h
0.9913	0.3300	0.9814	0.9814	0.8379	0.8388	0.9109	0.9116
0.9773	0.3250	0.9529	0.9529	0.6457	0.6454	0.7993	0.7992
0.9622	0.3200	0.9237	0.9238	0.4978	0.4984	0.7078	0.7082
0.9342	0.3120	0.8744	0.8745	0.3241	0.3248	0.5923	0.5929
0.9080	0.3060	0.8335	0.8335	0.2279	0.2275	0.5241	0.5238
0.8747	0.3000	0.7872	0.7872	0.1509	0.1508	0.4674	0.4674
0.8359	0.2950	0.7397	0.7398	0.0967	0.0968	0.4272	0.4273
0.7950	0.2915	0.6952	0.6952	0.0621	0.0620	0.4021	0.4021
0.7493	0.2890	0.6496	0.6497	0.0379	0.0379	0.3856	0.3857
0.6788	0.2870	0.5844	0.5844	0.0174	0.0174	0.3732	0.3732
0.5794	0.2860	0.4971	0.4971	0.0052	0.0052	0.3672	0.3672
0.4560	0.2857	0.3909	0.3909	0.0009	0.0009	0.3656	0.3656
0.3600	0.2857	0.3086	0.3086	0.0002	0.0001	0.3655	0.3655
0.2960	0.2857	0.2538	0.2537	0.0000	0.0000	0.3655	0.3655
0.2000	0.2857	0.1714	0.1714	0.0000	0.0000	0.3655	0.3655
0.1040	0.2857	0.0892	0.0891	0.0000	0.0000	0.3655	0.3655

 $^{\rm a}{\rm The}$ Sedov values are those published in Sedov (1959). The exact values are those calculated with the present solution method. Values that differ are highlighted.

Table 4. Values of key variables for the $\gamma = 1.4$ uniform density test cases at t=1 s^a

Geometry	$E_{\rm blast}$	α	J_1	J_2
Planar	0.0673185	0.538548	0.197928	0.175834
Cylindrical	0.311357	0.984041	0.0654053	0.0495650
Spherical	0.851072	0.851060	0.0296269	0.0211647

^aThe table contains the total energy E_{blast} , the nondimensional energy α (Eq. 54), and the integrals J_1 (Eq. 55) and J_2 (Eq. 56).

Table 5. Values of key results for the $\gamma = 1.4$ uniform density test cases at t=1 s^a

Geometry	r_2	ρ_2	u_2	E_2	P_2
Planar Cylindrical	0.500000 0.750000	6.00000	0.277778 0.312500	0.0385802 0.0488281	0.0925926 0.117188
Spherical	1.000000	6.00000	0.312500 0.333334	0.0488281 0.0555559	0.117100 0.133334

^aTable gives the shock position r_2 , and the post-shock density ρ_2 , material velocity u_2 , specific internal energy E_2 , and pressure P_2 .

Table 6. Values of key variables for the $\gamma = 1.4$ singular test cases at t=1 s^a

Geometry	$E_{\rm blast}$	ω	α
Cylindrical	2.45749	1.66667	$\begin{array}{c} 4.80856 \\ 4.90875 \end{array}$
Spherical	4.90875	2.33333	

^aTable contains the total energy E_{blast} , the initial density exponent ω , and the nondimensional energy α (Eq. 54).

Table 7. Values of key results for the $\gamma = 1.4$ singular test cases at t=1 s^a

Geometry	r_2	ρ_2	u_2	E_2	P_2
Cylindrical Spherical	$\begin{array}{c} 0.75 \\ 1.00 \end{array}$	9.69131 6.00000	$0.535714 \\ 0.625000$	$0.143495 \\ 0.195313$	$0.556261 \\ 0.468750$

^aTable gives the shock position r_2 , and the post-shock density ρ_2 , material velocity u_2 , specific internal energy E_2 , and pressure P_2 .

Table 8. Values of key variables for the $\gamma = 1.4$ vacuum test cases at t=1 s^a

Geometry	$E_{\rm blast}$	ω	α	J_1	J_2
Cylindrical Spherical	$2.67315 \\ 5.45670$	$1.70 \\ 2.40$	5.18062 5.45670	$\begin{array}{c} 0.856238 \\ 0.454265 \end{array}$	$0.158561 \\ 0.0828391$

^aThe table contains the total energy $E_{\rm blast}$, the density exponent ω , the nondimensional energy α (Eq. 54), and the integrals J_1 (Eq. 55) and J_2 (Eq. 56).

Table 9. Values of key results for the $\gamma = 1.4$ uniform vacuum cases at t=1 s^a

Geometry	r_v	r_2	ρ_2	u_2	E_2	P_2
Cylindrical Spherical	$0.154090 \\ 0.272644$	0.750000 1.00000	9.78469 6.00000	$0.543478 \\ 0.641026$	$0.147684 \\ 0.205457$	$0.578018 \\ 0.493097$

^aTable gives the vacuum boundary position r_v , shock position r_2 , and the post-shock density ρ_2 , material velocity u_2 , specific internal energy E_2 , and pressure P_2 .

Table 10. Global spatial convergence rates between pairs of $\operatorname{grid} s^a$

11	Density		٨	Pressure		٨	Speed		٨
cens	\mathbf{L}_1	q	А	\mathbf{L}_1	q	А	L_1	q	А
120	1.59E-01			3.57E-03			8.70E-03		
240	1.04E-01	6.13E-01	$2.68E{+}00$	2.35E-03	5.99E-01	5.65E-02	5.58E-03	6.39E-01	1.65E-01
480	6.06E-02	7.80E-01	$6.52E{+}00$	1.37E-03	7.82E-01	1.48E-01	3.10E-03	8.47E-01	4.98E-01
960	3.29E-02	8.79E-01	$1.18E{+}01$	7.44E-04	8.81E-01	2.70E-01	1.63E-03	9.27E-01	8.02E-01
1920	1.72E-02	9.36E-01	$1.71E{+}01$	3.88E-04	9.37E-01	3.92E-01	8.59E-04	9.25E-01	7.95E-01
3840	8.83E-03	9.62E-01	$2.09E{+}01$	1.98E-04	9.71E-01	5.02E-01	4.44E-04	9.51E-01	9.63E-01

^aSee equation 80.



Fig. 1.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled distance λ for planar geometry with γ and ω as labeled. After Figure 1 of Book (1991).



Fig. 2.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled radius λ for cylindrical geometry with γ and ω as labeled. After Figure 2 of Book (1991).



Fig. 3.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled radius λ for spherical geometry with γ and ω as labeled. After Figure 3 of Book (1991).



Fig. 4.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled radius λ for spherical geometry with γ and ω as labeled. After Figure 4 of Book (1991).



Fig. 5.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled radius λ for spherical geometry with γ and ω as labeled. After Figure 5 of Book (1991).



Fig. 6.— Plots of the Sedov functions f (red; scaled material velocity), g (blue, scaled mass density), and h (green, scaled pressure) versus scaled radius λ at $\omega = \omega_1$ for (a) j=2, $\gamma=5/3$, (b) j=3, $\gamma=5/3$. Plots of the entropy $gh^{-\gamma}$ (purple) with j = and (c) $\gamma=1.1$ and $\omega=2.5$, 2.65. 2.69, 2.71, 2.72727, 2.75, 2.77, 2.79, 2.83 (top to bottom) and (d) $\gamma=5/3$ and $\omega=1.0$, 1.6. 1.72, 1.76, 1.79, 1.8, 1.84, 1.88, 1.92, and 1.96 (top to bottom). After Figure 6 of Book (1991).



Fig. 7.— Plots of α as a function of ω for $\gamma=1.1$, 1.2, 1.3, 1.4, 1.5, and 5/3 in spherical geometry. After Figure 7 of Book (1991).



Fig. 8.— Sedov solutions for $\gamma=1.4$, standard cases, uniform density gas in for planar (purple), cylindrical (green), and spherical (red) geometries. Clockwise from the upper left are the density, material velocity, pressure, and specific internal energy, computed with 120 cells on the domain [0,1.2].



Fig. 9.— Sedov solutions for $\gamma=1.4$, singular cases for cylindrical (green) and spherical (red) geometries. Clockwise from the upper left are the density, material velocity, pressure, and specific internal energy, computed with 120 cells on the domain [0,1.2].



Fig. 10.— Sedov solutions for $\gamma=1.4$, vacuum cases for cylindrical (green) and spherical (red) geometries. Clockwise from the upper left are the density, material velocity, pressure, and specific internal energy, computed with 120 cells on the domain [0,1.2].



Fig. 11.— Comparison of the solutions to the Sedov problem for the density (upper left), pressure (upper right), specific internal energy (lower left), and material speed (lower right) when the initial energy is deposited in an exact delta-function (red dashed curve), in a single cell (purple), and in a small fixed size region (blue) at the final time of 1.0 s.



Fig. 12.— Absolute value of the relative error in the density (upper left), pressure (upper right), specific internal energy (lower left), and material speed (lower right) for a variety of uniform grids at a fixed time-step control value. Cusps indicate a change of sign in the relative error and the cpu cost of the solution on each grid is given.