Journal of Computational Physics **167,** 99–130 (2001)
doi:10.1006/jcph.2000.6661, available online at http://www.idealibrary.com on IDEAL®

On Consistent Time-Integration Methods for Radiation Hydrodynamics in the Equilibrium Diffusion Limit: Low-Energy-Density Regime

J. W. Bates,* D. A. Knoll,† W. J. Rider,‡ R. B. Lowrie,‡ and V. A. Mousseau†

**Plasma Physics Division, Naval Research Laboratory, Washington, DC 20375; †Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545; and ‡Computational and Computer Science Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545*

Received February 14, 2000; revised October 19, 2000

We compare the accuracy of three, mixed explicit–implicit schemes for simulating nonrelativistic, radiative hydrodynamic phenomena in the equilibrium diffusion limit. Only the "low-energy-density" regime is considered, where it is possible to ignore the effects of radiation pressure and energy density in comparison to the fluid pressure and energy density. The governing equations are then those of compressible Eulerian hydrodynamics with a nonlinear, radiative heat-transfer term appearing in the energy equation. All three finite-volume methods in this study utilize an explicit Godunov method with an approximate Riemann solver to integrate the Euler equations, but differ in their iterative treatment of the radiation diffusion term, which is handled in an "operator-split" fashion. In the first method, diffusive effects are computed with a linearized implicit technique that does not converge nonlinearities within a computational time step. In the other two methods, a Jacobian-free Newton–Krylov procedure is used to converge the nonlinearities, and improved accuracy (but not always greater efficiency) is achieved over the more traditional linearized–implicit approach. The two Newton–Krylov methods differ in their order of accuracy in time; one is strictly first-order accurate, while the other attempts to achieve second-order accuracy by making use of a predictor–corrector architecture. Several examples are considered to demonstrate the convergence properties of the three schemes, but attention is limited to spherically symmetric problems such as the one-dimensional point explosion.

Key Words: finite-volume methods; radiation hydrodynamics; nonlinear iterative schemes; Newton–Krylov.

0021-9991/01 $35.00

I. INTRODUCTION

The transport of radiation and its interaction with matter play an important role in astrophysics, inertial confinement fusion, and other high-temperature hydrodynamic systems. The processes of emission, absorption, and scattering of radiation result in an exchange of momentum and energy between the radiation field and matter that can greatly affect fluid motion [1, 2]. Typically, radiative phenomena occur on time scales that differ by many orders of magnitude from those characterizing hydrodynamic activity [3]. As a result, the accurate and efficient modeling of radiation hydrodynamics with high-resolution numerical codes presents significant computational challenges.

Although high-resolution methods for simulating pure hydrodynamic flows are well established [4–7], their extension to coupled radiative regimes has only recently begun (e.g., see Dai and Woodward [8, 9], Lowrie *et al.* [10], Balsara [11, 12], and references therein). In practice, both explicit and implicit methods are currently used for radiative hydrodynamic calculations. While explicit methods are usually simple to implement, associated time-step sizes often need to be restrictively small to satisfy stability conditions [13]. Implicit schemes allow much larger time-step sizes to be taken [14], but strong nonlinearities in the radiation terms are difficult to treat; frequently, a linearized solution is sought, which limits the accuracy of the scheme, or a nonlinear iteration procedure is employed. One of the most popular nonlinear iterative schemes is Newton's method [15], which in many circumstances allows an accurate treatment of nonlinearities. Standard Newton methods have a significant disadvantage, though, in that they require the formation and inversion of a Jacobian matrix, which is expensive in terms of CPU cost when a large system of equations is involved [16].

Newton–Krylov schemes [17, 18], on the other hand, achieve Newton-like, nonlinear convergence properties without this expense. The method is a unique combination of an outer Newton-based iteration and an inner conjugate-gradient-like (Krylov) iteration. The effects of the Jacobian are probed only through approximate matrix–vector products required in the Krylov iteration. Previously, Knoll *et al.* [16] demonstrated the benefit of using Jacobian-free Newton–Krylov methods to converge nonlinearities within a time step for a class of radiation diffusion problems. Here, we explore the consequences of including hydrodynamics in the description.

Toward this end, we present an accuracy comparison in time of three, mixed explicit–implicit algorithms for modeling radiative hydrodynamic phenomena. All three methods are finite-volume based, and separate the effects of hydrodynamics and radiation transport into two stages. The first stage, which is common to all three schemes, employs an explicit Godunov method [5, 7, 19] with a approximate Riemann solver to integrate the inviscid hydrodynamic equations in an Eulerian coordinate frame. The influence of nonlinear, radiative thermal conduction is then accounted for in the second stage, which differs from scheme to scheme. In one method, linear approximations of nonlinear terms are used, with the resulting system of implicit equations being solved by a linear iterative solver. Of course, this "linearized implicit" approach does not converge nonlinearities within a computational time step, and hence lacks (nonlinear) numerical consistency [5, 14]. In the other two algorithms, a nonlinear, iterative Newton–Krylov solver is used to converge nonlinearities; it is shown that much larger time steps can be taken for the same level of accuracy when the nonlinear residual is monitored and converged.

A related approach that also combines an explicit Godunov scheme with an implicit iterative method to solve coupled sets of radiative hydrodynamic equations was proposed by

Dai and Woodward in Ref. [8]. In that study, as here, attention was limited to the equilibrium diffusion limit [2, 20], and the effects of radiative thermal conduction were temporally "split" from the hydrodynamics. We should point out, however, that the scope of the present investigation differs from that of Ref. [8] in at least two significant ways. First, we are only considering the "low-energy-density" regime, in which the contribution of radiation pressure and energy density can be neglected in comparison to the total (fluid) pressure and energy density. Dai and Woodward, on the other hand, retained these contributions by developing a specialized Riemann solver to explicitly account for radiative terms in the conservation laws. Second, in order to converge nonlinearities in the radiation diffusion equation, Dai and Woodward developed an iterative procedure based solely on *linear* solvers such as Gauss–Seidel [15] and multigrid [21]. In contrast, the focus in this paper is on examining the utility of Newton-based iterative techniques for radiative hydrodynamic calculations and, in particular, how these techniques perform in comparison to linear methodologies. (The use of linear iterative solvers is currently a common practice in many radiation hydrodynamics codes; see Baldwin *et al.* [22] for a comprehensive review of five of the more popular schemes used in this context.)

The organization of this paper is as follows. In Section II, the governing equations of radiation hydrodynamics are written down in the equilibrium diffusion limit and then specialized to the one-dimensional, low-energy-density regime. In Section III, we outline the salient aspects of the three numerical schemes considered here: (i) the linearized implicit approach; (ii) a one-pass, iterative Newton–Krylov scheme for converging the nonlinearities within the radiation diffusion equation; and (iii) a two-step Newton–Krylov procedure with a predictor–corrector architecture that attempts to achieve second-order accuracy in time. Numerical results for several spherically symmetric test problems are presented in Section IV. Finally, in Section V, the conclusions of the paper are given, as well as a prospectus of future work.

II. PHYSICAL MODEL

For "optically thick" materials, gradients of the radiation density do not change appreciably over a distance of the order of the radiation mean-free path [3]. This is known as the "diffusion approximation." In this limit, and in the case that local thermodynamic equilibrium exists between the radiation field and the fluid, the nonrelativistic equations of radiation hydrodynamics can be written as [2, 20]

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{1}$$

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) + \nabla (p + p_\nu) = 0, \tag{2}$$

$$\frac{\partial}{\partial t}(\rho\varepsilon + \rho \mathbf{v}^2/2 + E_\nu) + \nabla \cdot [(\rho\varepsilon + \rho \mathbf{v}^2/2 + p + p_\nu + E_\nu)\mathbf{v}] = \nabla \cdot (\kappa \nabla T), \tag{3}$$

where ρ, p, $\mathbf{v}$, ε, and T are the mass density, pressure, flow velocity, internal energy per unit mass, and temperature of the fluid, respectively. Here, κ is the coefficient of thermal conduction, which may characterize material and/or radiative properties, and in general is a nonlinear function of ρ and T. The above description applies to "gray" materials such that the opacity $\omega \propto (\rho\kappa)^{-1}$ is independent of the frequency of photons in the radiation

field. In addition, we have that $p_\nu = E_\nu/3 = \frac{4}{3}\sigma T^4/c$, where p_ν is the radiation pressure, E_ν is the radiant energy density, σ is the Stephan–Boltzmann constant, and c is the speed of light. In the low-energy-density regime, p_ν and E_ν are neglected in comparison to p and $\rho\varepsilon + \rho\mathbf{v}^2/2$, respectively. Equations (1)–(3) constitute a closed system if an equation of state, $p = p(\rho, T)$, is specified. Throughout this paper, we assume an ideal gas equation of state, $p = R\rho T = (\gamma - 1)\rho\varepsilon$, where R is the gas constant per unit mass and γ is the ratio of specific heats. We also make the somewhat artificial choice $\gamma = 5/4$ in order to be able to compare our results with those in Refs. [23, 24].

In the case of spherical symmetry, such that the independent variables are the radial coordinate r and the time t only, Eqs. (1)–(3) in the low-energy-density regime reduce to

$$\frac{\partial \rho}{\partial t} + \frac{1}{r^2}\frac{\partial}{\partial r}(r^2 \rho v) = 0, \tag{4}$$

$$\frac{\partial}{\partial t}\rho v + \frac{1}{r^2}\frac{\partial}{\partial r}(r^2 \rho v^2) + \frac{\partial p}{\partial r} = 0, \tag{5}$$

$$\frac{\partial E}{\partial t} + \frac{1}{r^2}\frac{\partial}{\partial r} r^2 v(E + p) = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \kappa \frac{\partial T}{\partial r}\right), \tag{6}$$

where $E = \rho\varepsilon + \rho v^2/2$ is the total energy density of the fluid. For the purposes of this study, we assume that κ can be written as a power law in density and temperature, $\kappa = \kappa_0 \rho^a T^b$, where κ_0, a, and b are constants. This choice corresponds to an opacity $\omega \propto \rho^{-(a+1)}/T^b$, which is equivalent to the functional form assumed by Marshak in Ref. [25]. The remainder of this paper is concerned with the accurate numerical solution of Eqs. (4)–(6) for different values of a and b, and for different initial conditions. Let us now outline the three numerical schemes considered for this purpose.

III. TIME-INTEGRATION METHODS

For all three methods, the solution strategy can be loosely described as "operator splitting," which is a common approach in numerical codes used to model multiple physical phenomena [1]. For our purposes here, the essential ideal is to separate hydrodynamic and radiative processes into two stages during each time step. In the first stage, the hydrodynamic (Euler) equations are solved, and in the second stage, the total energy density is updated to account for the effects of radiation transport. The three algorithms are distinguished by the way in which this second stage is carried out, and in particular, whether or not nonlinearities in the radiative conduction term are converged.

Specifically, the splitting procedure can be accomplished by writing Eqs. (4)–(6) as a system of hyperbolic conservation laws [26],

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial (A\mathbf{F})}{\partial V} + \frac{\partial \mathbf{G}}{\partial r} = 0, \tag{7}$$

plus a diffusion equation

$$\frac{\partial E}{\partial t} = \frac{\partial}{\partial V}\left(A\kappa \frac{\partial T}{\partial r}\right), \tag{8}$$

where

$$\mathbf{U} \equiv \begin{bmatrix} \rho \\ \rho v \\ E \end{bmatrix}, \quad \mathbf{F}(\mathbf{U}) \equiv \begin{bmatrix} \rho v \\ \rho v^2 \\ v(E+p) \end{bmatrix}, \quad \mathbf{G}(\mathbf{U}) \equiv \begin{bmatrix} 0 \\ p \\ 0 \end{bmatrix}. \tag{9}$$

Here, $V = 4\pi r^3/3$ is the generalized volume coordinate in one-dimensional spherical geometry, with $A = dV/dr = 4\pi r^2$ the associated cross-sectional area. Note that the flux vector $\mathbf{F}$ differs from the usual planar case in that a pressure term is absent from the second entry. The solution algorithm proceeds by consecutively solving Eqs. (7) and (8) at each time step. Initially, new values of the mass, momentum, and total energy densities are found from the solution of the Euler equations [Eq. (7)]. The total energy density obtained in this step, though, is only an intermediate result, which is then updated according to Eq. (8). Note that the value of E determined in the hydrodynamic stage serves as the initial value for integrating Eq. (8).

All three numerical approaches considered in this paper utilize a Godunov method with a modified version of Roe's approximate Riemann solver [27] to integrate the Euler equations in Eq. (7). The use of the word "modified" here relates to the fact that the solution procedure is carried out in terms of primitive variables ρ, v, and p, instead of conserved variables ρ, ρv, and E. The three algorithms differ, however, in the iterative method used to solve Eq. (8); two of them employ a nonlinear Newton–Krylov scheme [17, 18], while the other relies on a linear solution methodology. Let us first outline the Godunov method that is common to all three algorithms, and then proceed to a discussion of the different conduction modules.

The basic idea in Godunov's original method [19] is to first assume a piecewise constant distribution of the conserved variables $\mathbf{U}$ in each computational cell. Then, the resulting initial-value problem for the Euler equations is solved region-by-region in the flow using these piecewise constant data. Effectively, this generates a sequence of local Riemann problems [7] with left and right state vectors centered at the boundaries between grid cells. For the Euler equations, solutions of the Riemann problem may consist of shock waves, contact discontinuities, and/or rarefactive waves. Once the Riemann solutions are determined, the updated values of the hydrodynamic variables are pieced together by interpolating back onto the computational grid.

Formally, the averaging procedure in a Godunov method is performed by replacing the original data $\mathbf{U}(r, t)$ with a discretized set of piecewise constant states:

$$\mathbf{U}_j^n \equiv \frac{1}{\Delta V_j} \int_{r_{j-1/2}}^{r_{j+1/2}} \mathbf{U}(r, t^n)\, dV. \tag{10}$$

Here, we consider a spherically symmetric numerical grid $\{r_j\}$ in the radial direction, where j is a positive integer, and $\Delta r_j \equiv r_{j+1/2} - r_{j-1/2}$ is the mesh spacing. The braces here are used to represent all j values from 1 to N, where N is the number of grid cells. In what follows, we assume a uniform mesh so that the spacing between cells is simply given by Δr. The boundary between the jth and $j+1$ cells is denoted by $r_{j+1/2}$ and the cell volume is $\Delta V_j \equiv V(r_{j+1/2}) - V(r_{j-1/2})$. The superscript n signifies the nth time level where $t^n \equiv n\Delta t$, and Δt is the time step size. Here, we consider only interior points of the computational grid; boundary conditions are addressed in our discussion of the test problems in Section IV.

With these definitions, we can apply Godunov's method to calculate the updates of the hydrodynamic variables. Integrating Eq. (7) over the cell volume ΔV_j and a time step $t^n \le t \le t^{n+1}$ yields

$$\mathbf{U}_j^{n+1} \approx \mathbf{U}_j^n - \frac{\Delta t}{\Delta V_j}\left(A_{j+1/2}\mathbf{F}_{j+1/2}^{n+1/2} - A_{j-1/2}\mathbf{F}_{j-1/2}^{n+1/2}\right) - \frac{\Delta t}{\Delta r}\left(\mathbf{G}_{j+1/2}^{n+1/2} - \mathbf{G}_{j-1/2}^{n+1/2}\right), \quad (11)$$

where $A_{j\pm1/2} \equiv A(r_{j\pm1/2})$ is the area between the jth and $j \pm 1$ cells, and we have used the approximation $dV/dr \approx \Delta V_j/\Delta r$ to obtain the last term on the right side of the equation above. The symbols $\mathbf{F}_{j\pm1/2}^{n+1/2}$ and $\mathbf{G}_{j\pm1/2}^{n+1/2}$ denote the time averages of $\mathbf{F}$ and $\mathbf{G}$ at $r_{j\pm1/2}$:

$$\mathbf{F}_{j\pm1/2}^{n+1/2} \equiv \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{F}\left[\mathbf{U}\left(r_{j\pm1/2}, t\right)\right] dt, \quad (12)$$

$$\mathbf{G}_{j\pm1/2}^{n+1/2} \equiv \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{G}\left[\mathbf{U}\left(r_{j\pm1/2}, t\right)\right] dt. \quad (13)$$

The problem then reduces to estimating these quantities from the data $\{\mathbf{U}_j^n\}$ with an appropriate Riemann solver. Once the Riemann solution is found, the hydrodynamic stage of the calculation is complete for a single time step. We employ a modified Roe's scheme to perform this stage of the calculation, the details of which are given in the Appendix. Let us now turn to a discussion of the different ways the three codes incorporate nonlinear conduction effects into the hydrodynamic solution.

A. *A Linearized Implicit Scheme for the Radiation Diffusion Equation*

The first method for integrating Eq. (8) that we consider is a linearized implicit technique. The distinguishing features of this approach to bear in mind are that it is first-order accurate by linearized analysis, and nonlinearities are not converged within a time step. By this we mean that the nonlinear coefficient of thermal conduction κ in Eq. (8) is evaluated with previous time-step solutions so that we have a *linear* implicit problem to solve at each time level. Incorporating conduction effects into the hydrodynamic solution in this way is sometimes referred to as "classical" operator splitting.

The integration scheme for our radiative hydrodynamic equations in this case proceeds as follows. At the beginning of the $n + 1$ time step, updates of mass, momentum, and total energy densities are determined from solving the Euler equations in Eq. (7) with the explicit Godunov scheme outlined above. Schematically, we can represent this process as

$$\mathbf{U}^n \to \mathbf{U}^* = \begin{Bmatrix} \rho_j^{n+1} \\ (\rho v)_j^{n+1} \\ E_j^* \end{Bmatrix}.$$

The total energy density $\{E_j^*\}$ obtained in this stage, however, is only an intermediate result; the effects of thermal conduction must now be included. This is accomplished by temporarily suspending hydrodynamic activity, holding the variables ρ_j^{n+1} and $(\rho v)_j^{n+1}$ fixed, and solving the finite-volume form of the diffusion equation in Eq. (8) for the new

temperatures T_j^{n+1}. That is, we must solve

$$\frac{c_V \rho_j^{n+1} T_j^{n+1} + (\rho v^2)_j^{n+1}/2 - E_j^*}{\Delta t} = \frac{A_{j+1/2}\,\kappa_{j+1/2}^n \left(T_{j+1}^{n+1} - T_j^{n+1}\right)/\Delta r}{\Delta V_j} - \frac{A_{j-1/2}\,\kappa_{j-1/2}^n \left(T_j^{n+1} - T_{j-1}^{n+1}\right)/\Delta r}{\Delta V_j}, \quad (14)$$

where $c_V = R/(\gamma - 1)$ is the specific heat at constant volume. Here, we have related the total energy density to the temperature according to $E = c_V \rho T + \rho v^2/2$. The coefficient of thermal conduction κ in Eq. (14) is evaluated at cell interfaces by linear interpolation of the cell-centered densities and temperatures:

$$\kappa_{j+1/2}^n = \kappa_0 \left(\frac{\rho_j^{n+1} + \rho_{j+1}^{n+1}}{2}\right)^a \left(\frac{T_j^n + T_{j+1}^n}{2}\right)^b. \quad (15)$$

Since κ is evaluated at the old time level of temperature, Eq. (14) is *linear* in the unknowns $\{T_j^{n+1}\}$, and the resulting tridiagonal system can be solved in a number of ways. The time step is complete once the new temperature profile is determined, and the solution vector is updated accordingly:

$$\mathbf{U}^* \to \mathbf{U}^{n+1} = \begin{Bmatrix} \rho_j^{n+1} \\ (\rho v)_j^{n+1} \\ E_j^{n+1} \end{Bmatrix}.$$

The process then repeats with these values serving as the initial data for the next time step.

One possibility for solving Eq. (14) is to use a direct method such as the standard tridiagonal solver [15]. This technique is extremely efficient for one-dimensional problems, but unfortunately, its efficacy does not extend to higher dimensions. For that reason, we choose not to employ a tridiagonal solver as a benchmark of standard linear-scheme performance with which to assess the accuracy and efficiency of Newton–Krylov algorithms for this class of problems. Any conclusions reached by doing so could not be expected to carry over into two or three spatial dimensions, which is undoubtedly where the true utility of Newton–Krylov schemes lies for radiative hydrodynamic calculations. Instead, the solution methodology that we adopt is a linear iterative solver. The particular solver that we use is known as GMRES [17] with symmetric Gauss–Seidel preconditioning [14]. Some of the details of the GMRES method will be explained in the next section.

Currently, linear implicit methods are widely used in radiation hydrodynamics codes [1, 22]. This is true despite the fact that usually no facility exists in the computational algorithm for monitoring the size of nonlinear residual terms or for quantifying the error introduced into the solutions as a result of a linearization procedure. Indeed, in our treatment here, nonlinearities have been completely disregarded by evaluating $\kappa(\rho, T)$ with old time-level information for the temperature. As a general practice, the extent to which this numerically inconsistent process [5, 14] degrades the fidelity of solutions is difficult to assess, but undoubtedly, significant errors can be introduced [16]. We return to this issue in Section IV when we discuss the performance of linearized implicit schemes as applied to our particular radiative hydrodynamic examples. In the next two sections, we explore consistent alternatives to the linearized implicit technique known as iterative Newton–Krylov methods, in

which nonlinear residual terms are monitored and converged within a computational time step. From previous studies without hydrodynamics, we expect these methods to permit time steps that are larger than those in the linear implicit approach for the same level of accuracy.

B. *A Newton–Krylov Scheme for the Radiation Diffusion Equation*

The computational difficulty associated with integrating Eq. (8) lies in the coefficient of thermal conduction κ, whose temperature dependence is modeled as a nonlinear power law (e.g., $\kappa \sim T^{13/2}$). In a true implicit scheme, a direct method cannot be applied to this problem; some sort of nonlinear iterative procedure is necessary. Here, we describe a first-order, iterative Newton–Krylov method and its use in evaluating the diffusion term in Eq. (8). Unlike the linear implicit method discussed in the previous section, the emphasis in this scheme is to monitor and converge nonlinearities below a certain tolerance before the completion of a time step. Compared to the standard Newton's method, the Newton–Krylov approach is advantageous in that one can avoid the computational difficulty and expense of forming and inverting the Jacobian matrix.

Since a discussion of Newton's method can be found in many textbooks on numerical methods (e.g., see Ref. [15]), we shall not be concerned with trying to give a complete description of it here. Rather, our intent is to underscore the unique features of composite Newton–Krylov schemes. In particular, we wish to elucidate the Jacobian-free aspect of coupling Newton's method with a linear Krylov solver, which we describe briefly below. For additional information on Krylov methods, the reader is referred to Refs. [28, 29].

The first step in the implementation of any Newton-based algorithm is to determine the appropriate nonlinear function(s) at each grid cell. In our case, these functions are defined from the discretized version of Eq. (8). Specifically, we have

$$f_j = \frac{c_V \rho_j^{n+1} T_j^{n+1} + (\rho v^2)_j^{n+1}/2 - E_j^*}{\Delta t} - \frac{A_{j+1/2}\kappa_{j+1/2}^{n+1}\left(T_{j+1}^{n+1} - T_j^{n+1}\right)/\Delta r}{\Delta V_j} + \frac{A_{j-1/2}\kappa_{j-1/2}^{n+1}\left(T_j^{n+1} - T_{j-1}^{n+1}\right)/\Delta r}{\Delta V_j}, \tag{16}$$

where

$$\kappa_{j+1/2}^{n+1} = \kappa_0 \left(\frac{\rho_j^{n+1} + \rho_{j+1}^{n+1}}{2}\right)^a \left(\frac{T_j^{n+1} + T_{j+1}^{n+1}}{2}\right)^b. \tag{17}$$

The goal of the Newton–Krylov scheme is to iteratively drive the function f_j toward zero at each grid cell. Let us now explain how this is accomplished.

In a standard Newton method, one seeks the solution of a nonlinear system of equations written in the form $\mathbf{f}(\mathbf{x}) = 0$, where $\mathbf{x}$ is known as the "state vector." For the problem at hand, we have $\mathbf{x} = \{T_j\}$, and $\mathbf{f}(\mathbf{x}) = \{f_j\}$, where the functions f_j are those defined in Eq. (16). To determine $\mathbf{x}$, Newton's method prescribes an iterative procedure derived from the *linear* system of equations

$$\mathbf{J}_k \delta\mathbf{x}_k = -\mathbf{f}(\mathbf{x}_k), \tag{18}$$

where $\mathbf{J}$ represents the Jacobian matrix, k is a nonlinear iterative index, and $\delta\mathbf{x}$ is the update of the state vector, which is found from multiplying both sides of Eq. (18) by the inverse of

J. New values of **x** are computed from the recurrence relation $\mathbf{x}_{k+1} = \mathbf{x}_k + \delta\mathbf{x}_k$. The process is considered complete once $\|\mathbf{f}(\mathbf{x}_k)\|_2$ is less than some nonlinear convergence tolerance. For our purposes here, this tolerance is either 10^{-8} or 10^{-7}.

The primary disadvantage of Newton's method in its standard implementation is that it can be difficult and time consuming. To form and then invert the Jacobian matrix in Eq. (18), one must take partial derivatives of **f** with respect to the different components of the state vector **x**, and one must do this for each stage of the nonlinear iteration process. For some systems, the evaluation of the Jacobian coefficients can be carried out analytically, but more likely, some sort of numerical scheme for estimating the derivatives is required. Such an exercise can be unduly problematic and expensive, particularly when a large nonlinear system of equations is involved, or when the functional dependences of physical variables are expressed in tabular form. In the present study, we avoid this difficulty by coupling Newton's method with a Krylov solver [28] that never requires the formation of the actual Jacobian matrix. The particular Krylov method that we employ is the GMRES algorithm due to Saad and Shultz [17].

In a Krylov method, the solution update is expressed as a linear combination of so-called Krylov vectors $(\mathbf{r}_0, \mathbf{J}\mathbf{r}_0, \mathbf{J}^2\mathbf{r}_0, \ldots, \mathbf{J}^l\mathbf{r}_0)$, where the index l denotes the level of the Krylov iteration, and $\mathbf{r}_0$ is an initial linear residual defined by $\mathbf{r}_0 = -\mathbf{f}(\mathbf{x}_k) - \mathbf{J}_k\delta\mathbf{x}_{k0}$. Here, the symbol $\delta\mathbf{x}_{k0}$ represents an initial guess for the update of the state vector at a particular nonlinear iteration level; this value is typically taken to be zero. Although the GMRES iteration takes place at a fixed nonlinear index k, we find it helpful to write the update with two indices, k and l. The expansion of $\delta\mathbf{x}_{kl}$ can be written as

$$\delta\mathbf{x}_{kl} = \delta\mathbf{x}_{k0} + \sum_{m=0}^{l-1} \beta_m \mathbf{J}_k^m \mathbf{r}_0, \tag{19}$$

where the β_m's are scalar coefficients. These coefficients are determined for the lth linear GMRES iteration by minimizing the quantity $\|\mathbf{J}_k\delta\mathbf{x}_{kl} + \mathbf{f}(\mathbf{x}_k)\|_2$ with a least-squares method. This requires the use of the Krylov vectors constructed during the previous $l-1$ iterations. The condition for the termination of a linear GMRES iteration is given by

$$\|\mathbf{J}_k\delta\mathbf{x}_{kl} + \mathbf{f}(\mathbf{x}_k)\|_2 < \tau\|\mathbf{f}(\mathbf{x}_k)\|_2, \tag{20}$$

where $\|\mathbf{f}(\mathbf{x}_k)\|_2$ is the residual of the present nonlinear iteration, and τ is a linear convergence tolerance; typically, we take $\tau = 10^{-2}$. Throughout this study, we limit the maximum number of Newton and GMRES iterations to 15 and 100, respectively.

Formulated in this way, our scheme is more correctly described as an "inexact" Newton's method because the convergence criterion is adjusted during each nonlinear iteration. This is a useful feature that can do much to improve the efficiency of Newton's method for the following reasons. When the nonlinear residual is large and the solution is far from convergence, there is no point in spending much computational effort solving the linear problem well. However, when the solution is nearly converged, it makes sense to expend more effort in the linear routine since the superlinear convergence rate in the final Newton iterations is tightly coupled to the accuracy of that solution [30].

Another noteworthy quality of the Newton–Krylov scheme is the fact that the GMRES algorithm involves only the *product* of the Jacobian matrix with a vector. This fact plays a key role in the utility of the method. Because the Jacobian–vector product can be well

approximated by a first-order Taylor series as [16, 18]

$$\mathbf{J}\mathbf{y} \approx [\mathbf{f}(\mathbf{x} + \epsilon\mathbf{y}) - \mathbf{f}(\mathbf{x})]/\epsilon, \tag{21}$$

the Jacobian by itself is never required, which is highly desirable from a computational standpoint. Here, the vector $\mathbf{y}$ is an element of the Krylov basis, and the scalar ϵ is a small perturbation. In this study, the value of ϵ is given by

$$\epsilon \equiv \frac{\eta \sum_{j=1}^{N} x_j}{N \|\mathbf{y}\|_2}, \tag{22}$$

where the constant parameter η has a value of 10^{-5}.

To further enhance the performance of our Newton–Krylov scheme, we employ a "preconditioner." Preconditioning is a process that approximates the inverse of the Jacobian matrix in such a way as to efficiently cluster together the eigenvalues of the iteration matrix [16, 30]. This in turn reduces the number of GMRES iterations required to reach convergence. The inverse of the preconditioning matrix is much easier to form than the inverse of the true Jacobian of the system.

Let $\mathbf{P}$ denote the preconditioning matrix such that $\mathbf{J}\mathbf{P}^{-1} \approx \mathbf{I}$. The system of equations we wish to solve is then

$$\mathbf{J}\mathbf{P}^{-1}\,\mathbf{P}\,\delta\mathbf{x} = -\mathbf{f}(\mathbf{x}). \tag{23}$$

We choose $\mathbf{P}$ to be the tridiagonal matrix resulting from the linearized implicit scheme discussed in Section IIIA. Letting the vector $\mathbf{q}$ denote the product $\mathbf{P}\,\delta\mathbf{x}$, the left side of Eq. (23) is approximated as before using a first-order Taylor expansion

$$\mathbf{J}\mathbf{P}^{-1}\mathbf{q} \approx [\mathbf{f}(\mathbf{x} + \epsilon\mathbf{P}^{-1}\mathbf{q}) - \mathbf{f}(\mathbf{x})]/\epsilon. \tag{24}$$

The inverse of $\mathbf{P}$ is found iteratively, *but not to convergence*, using a symmetric Gauss–Seidel technique [14]. In other studies, preconditioners based on the multigrid method have also been used [30–32].

The description above highlights the important principles of Newton–Krylov algorithms as employed in this investigation. Because of their utility for accurately and efficiently integrating other nonlinear systems of equations, we have been motivated to apply Newton–Krylov methods to radiation hydrodynamics. Previous work [16, 30] has shown that Newton-like nonlinear-convergence behavior can be achieved without undue computational expense through the approximate matrix–vector multiplication procedure in the GMRES routine. This savings in computational effort is principally a result of the fact that the Jacobian matrix never has to be formed or inverted, which is the impetus for referring to the scheme as "Jacobian-free." In the previous studies cited above, significantly improved performance over traditional linearized–implicit methods was documented using Newton–Krylov techniques to converge nonlinearities in nonequilibrium diffusion calculations; our aim here is to see if this trend extends to the present class of problems.

Note that we are coupling a Newton–Krylov scheme to a Godunov-based hydrodynamic integrator in the same way as the linearized implicit method. Once again, the procedure at each time step is first to find provisional updates for the hydrodynamic variables $\mathbf{U}^* = [\rho^{n+1}, (\rho v)^{n+1}, E^*]^{\mathcal{T}}$ with the Godunov scheme. Unlike the linear implicit

method, though, the approach here is then to incorporate the effects of radiation transport by iteratively solving Eq. (18) to convergence for the temperature profile $\{T_j^{n+1}\}$ using a Newton–Krylov method. The temperature data are then used to compute the final update $\mathbf{U}^{n+1} = [\rho^{n+1}, (\rho v)^{n+1}, E^{n+1}]^{\mathcal{T}}$. One disadvantage of this version of the integration scheme is that it is strictly first-order accurate. In the next section, we describe a possible technique for extending the coupled Godunov/Newton–Krylov algorithm to higher order in time for smooth regions of the fluid flow. This is done by embedding the Newton–Krylov procedure within the predictor–corrector structure of Hancock's two-step Godunov solver [33].

C. *A Newton–Krylov Method within a Predictor–Corrector Framework*

We now consider an approach for achieving higher-order accuracy in time with our coupled Godunov/Newton–Krylov method. One likely means of improving the time accuracy is to structure the Newton–Krylov algorithm within a predictor–corrector framework [15]. Since the Godunov method used to integrate the hydrodynamic equations already possesses this architecture, it is straightforward to modify the algorithm accordingly. Our procedure for implementing this modification can be summarized as follows.

The first step is to replace the implicit diffusive fluxes in Eq. (16) with a combination of both implicit and explicit terms. To do this, it is convenient to introduce an implicit–explicit parameter θ that varies between zero and one. For $\theta = 1$ the Newton–Krylov algorithm is fully implicit, whereas for $\theta = 0$, it is completely explicit; the choice $\theta = 1/2$ corresponds to an equal mixture of explicit and implicit terms (as in the Crank–Nicolson method [34]). For simple linear diffusion equations, this choice is a common one because it provides both unconditional stability and second-order accuracy in time [6]. In the predictor–corrector formulation of our coupled algorithm, the Newton–Krylov iterative process is called twice during each time step, and a different value of θ is used for each call. During the "predictive" Newton–Krylov iterations, where there is no reason to solve the problem to high accuracy, we choose $\theta = 1$, while for the "corrector" stage, we set $\theta = 1/2$ in an effort to achieve second-order convergence.

With the parameter θ, the discrete approximation for (the negative of) the heat flux at each cell interface can be written in the mixed implicit/explicit representation as

$$d_{j+1/2} = \theta\, \kappa_{j+1/2}^{n+1}\left(T_{j+1}^{n+1} - T_j^{n+1}\right)/\Delta r + (1-\theta)\kappa_{j+1/2}^{n}\left(T_{j+1}^{n} - T_j^{n}\right)/\Delta r. \qquad (25)$$

Here, the explicit form of the coefficient of thermal conduction, $\kappa_{j+1/2}^{n}$, is to be evaluated using old time-level information for both temperature *and* density. Note that this differs from the expression in Eq. (15). The appropriate set of nonlinear functions is

$$f_j = \frac{c_V \rho_j^{n+1} T_j^{n+1} + (\rho v^2)_j^{n+1}/2 - E_j^*}{\Delta t} - \frac{A_{j+1/2} d_{j+1/2} - A_{j-1/2} d_{j-1/2}}{\Delta V_j}. \qquad (26)$$

As before, the goal of our Newton–Krylov procedure is to iteratively drive the function f_j below some specified nonlinear convergence tolerance at each grid cell.

The predictor stage of the calculation begins by first invoking the Newton–Krylov method with $\theta = 1$ in Eq. (25) to compute cell-centered temperature values at the half time level: $\{T_j^{n+1/2}\}$. These data are used to estimate the divergences of the heat flux at each cell, which are then added to the time derivatives of $\rho\varepsilon$ and p used in the primitive-variable Riemann

solver; see Eqs. (48) and (49) in the Appendix. Specifically, we add the factor

$$\frac{A_{j+1/2}\,\kappa_{j+1/2}^{n+1/2}\left(T_{j+1}^{n+1/2}-T_{j}^{n+1/2}\right)/\Delta r - A_{j-1/2}\,\kappa_{j-1/2}^{n+1/2}\left(T_{j}^{n+1/2}-T_{j-1}^{n+1/2}\right)/\Delta r}{\Delta V_j}$$

to the right-hand side of Eq. (48). The redundant expression for the time derivative of the pressure in Eq. (49) is modified by the addition of the same term, but with a multiplicative factor of $(\gamma - 1)$ in front (see Appendix). Here, the coefficients of thermal conduction $\kappa_{j+1/2}^{n+1/2}$ are evaluated with temperature data at the half time level $n + 1/2$, but with density values at the old time level n. The evaluation of time-centered expressions for the hydrodynamic variables [Eqs. (44) and (45)] in this way constitutes the "predictive" part of the algorithm. In the corrector stage, the Riemann problem is solved to give the provisional updates of the conserved variables $\mathbf{U}^* = [\rho^{n+1}, (\rho v)^{n+1}, E^*]^{\mathcal{T}}$. This is followed by a second call to the Newton–Krylov scheme with $\theta = 1/2$ to determine the temperature at the time level $n + 1$, and hence the final update $\mathbf{U}^{n+1} = [\rho^{n+1}, (\rho v)^{n+1}, E^{n+1}]^{\mathcal{T}}$.

Formulated in this way, conduction effects have an opportunity to provide "feedback" and influence hydrodynamic motion during each time step. It can be said that this method of coupling nonlinear radiation diffusion with hydrodynamics retains less of the character of operator splitting than the previous Newton–Krylov scheme, which itself differs from the traditional brand of splitting typified by the linearized implicit approach. Before comparing the performance of the linearized implicit, Newton–Krylov, and predictor–corrector Newton–Krylov schemes on several numerical examples, we first turn to a brief description of the method of time step selection used in this study.

D. *Time Step Control*

Time step control in radiative hydrodynamic computations is complicated by the fact that the coefficient of thermal conduction κ is usually a (nonlinear) function of density and temperature. In general, there is no optimal method of selecting time step sizes in this case. Over the years, though, several techniques have been developed and utilized. For schemes that do not converge nonlinearities, one of the most popular approaches is based on monitoring the relative change in a dependent variable such as the total energy density E. (See Bowers and Wilson [1] for a discussion of this common practice.) The process begins by first making a conservative initial guess for the time step Δt that is likely to satisfy all stability requirements. The idea is then to increase the size of Δt as much as possible in a systematic way. This is done by first computing the maximum relative change in E over a single time step according to

$$\left(\frac{\Delta E}{E}\right)^{n+1} = \max_j \left(\frac{\left|E_j^{n+1} - E_j^n\right|}{E_j^{n+1} + E_0}\right), \tag{27}$$

where the parameter E_0 in the denominator is added for regularity and is usually an estimate for the lower bound of the energy density. The new time step is then determined from

$$\Delta t^{n+1} = \Delta t^n \sqrt{\frac{(\Delta E/E)^{n+1}}{(\Delta E/E)_{\max}}}, \tag{28}$$

where $(\Delta E/E)_{\max}$ is a specified target value for the relative change in E; typically, this value is chosen to lie between 5 and 20%. The use of the square root in Eq. (28) is an attempt to make the new time step smoothly approach the target value. The justification often cited for employing this method of time step selection is that as long as the relative change in energy is kept "small," the errors introduced by a linearization procedure are not significant.

An alternate method for selecting time step sizes in nonlinear diffusion problems was suggested by Rider and Knoll [35] and is based on a heuristic criterion for estimating nonlinear wave speeds. The essential idea is to replace the complicated system of equations involved with a simplified hyperbolic model given by

$$\frac{\partial E}{\partial t} + v_f \frac{\partial E}{\partial r} = 0, \tag{29}$$

where the "front velocity" v_f is an unknown function. The selection procedure is then to specify a desired front-CFL value from which the time step is computed according to

$$\Delta t^{n+1} = \text{front-CFL} \times \frac{\Delta r}{v_f^n}. \tag{30}$$

In this approach, the prescription for calculating the front velocity v_f is

$$v_f^n = \frac{\| |\Delta E^n|/\Delta t^n \|_1}{\| |\Delta E^n|/\Delta r \|_1}. \tag{31}$$

Coupled with the Newton–Krylov technique, the front-CFL method of time step control in nonlinear diffusion problems has proven [40] to contribute to high-quality results with reduced cost compared to the standard $\Delta E/E$ method and is used exclusively in this investigation. We should mention, though, that for more complex, radiative hydrodynamic systems, the general criterion by which the front velocity is to be selected with this approach is unclear. It is likely that the expression given in Eq. (31) will require modification when applied to problems more complicated than those considered in this study.

We should also remark that in our numerical codes, the actual time-step selection process involves a comparison with an estimate of Δt based on pure fluid motion for a hydrodynamic CFL number of about $1/2$. The procedure is always to select the most restrictive of the two estimates. Furthermore, the size of Δt is never permitted to increase by more than a factor of 2 between any two consecutive time steps. Let us now consider the application of the time-integration techniques described in this section to a series of one-dimensional test problems.

IV. RESULTS

In this section, we investigate the performance of our three numerical schemes on several one-dimensional problems in spherical geometry. Our intent is to compare the time accuracy, convergence properties, and CPU cost of the three methods for solving the same problem. As we shall see, the predictor–corrector Newton–Krylov scheme is consistently and significantly more accurate for roughly the same time step size than both the linearized implicit and first-order Newton–Krylov methods, but also more expensive in terms of CPU cost. For each of the test problems considered, the time accuracy comparisons are made at

a fixed grid size that was chosen to give a reasonably well-converged solution. We should point out, however, that the scope of the present study was limited to errors associated with time step size only; we are not concerned here with presenting results of the convergence behavior of the solutions under refinement of the spatial grid. All runs in this study were performed in double precision on a Sun Ultra 60 computer operating at 360 MHz.

As mentioned in the previous section, the front-CFL technique of time step control [35] is used throughout this investigation. While convergence trends are similar for both the $\Delta E/E$ and front-CFL methods, the latter appears to be a more advantageous means of time step selection for the types of radiative hydrodynamic problems studied here. In all cases, the time accuracy and convergence properties of the three methods are examined by plotting the absolute L_2 error in the final temperature profile versus the specified (target) front-CFL value. The absolute L_2 error is defined as

$$\left[\sum_{j=1}^{N}\left(T_j - T_j^e\right)^2\right]^{1/2},$$

where the data T_j^e constitute the "exact" solution for $\Delta t \to 0$. This is taken to be the result from a run using the predictor–corrector Newton–Krylov method with an exceedingly small front-CFL value (typically 10^{-4}–10^{-3}).

The first test case that we consider is the so-called Barenblatt problem [3], which describes nonlinear thermal conduction from a point source. Since this problem has an analytical solution in terms of a similarity variable [3, 36], it is a useful starting point in our study for purposes of code validation. For all three numerical algorithms, the Barenblatt limit is reached by simply disabling the update of fluxes in Eq. (11) so that hydrodynamic motion is suspended, and we are essentially solving a nonlinear diffusion equation for the temperature profile. The nonlinearity in the problem results from the assumed form of the coefficient of thermal conduction κ. For this class of problems, results for two different temperature dependences of κ are presented: $\kappa \sim T^{5/2}$ and $\kappa \sim T^{13/2}$. (The first choice represents an effort to mimic classical Spitzer–Härm electron thermal conduction in a plasma [41], whereas the second corresponds to the dependence in a high-temperature fully ionized plasma in which bremsstrahlung is the dominate radiative transfer mechanism [24].) The power law $T^{13/2}$ gives rise to a solution with a much steeper thermal front than $T^{5/2}$, but overall temporal convergence rates for both dependences are still second order.

In the next two test cases, hydrodynamic activity is enabled in our numerical schemes, and our first objective is to examine convergence properties for nondiscontinuous initial conditions. This is done by simulating the evolution of a smooth initial distribution of the energy density up until a point where the solution profile has an opportunity to "shock." The motivation for considering this problem is not necessarily to model a particular physical process, but to examine the accuracy and convergence behavior in the absence of nearly discontinuous spatial profiles such as shock fronts. We find, though, that even when steep gradients are not present, the convergence rate in time of the error is essentially first order for all three schemes. This result is a consequence of the nature of the two-step Hancock scheme used in our Godunov method and is discussed further in Section IVB.

For the third class of radiative hydrodynamic test problems, we follow the recent work of Shestakov [23] and simulate point explosions with nonlinear heat conduction. Two different problems, each with a different initial energy deposition, ambient density profile, and coefficient of thermal conduction, are considered. These test cases are essentially a hybrid

of the Barenblatt [3, 36] and Sedov [37, 38] blast-wave problems. Not surprisingly, the convergence rates of all three coupled schemes for this class of problems are first order—as they would be for any shock capturing method [5].

A. *Nonlinear Thermal Conduction from a Point Source (the Barenblatt Problem)*

Let us begin by comparing the accuracy of the three methods for a simplified physical model in which hydrodynamics is absent from the description ($v = 0$). That is, we seek solutions of the diffusion equation

$$\frac{\partial T}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \chi \frac{\partial T}{\partial r}\right), \tag{32}$$

where we have introduced the thermal diffusivity χ defined by $\chi = \kappa/(\rho c_V)$; the density ρ and the specific heat c_V of the material are assumed fixed. For coefficients of thermal conduction that obey a power-law dependence on temperature, $\kappa = \kappa_0 T^b$ with $b > 0$, it is well known that this equation possesses an analytical solution in terms of a similarity variable. This solution is due to Barenblatt [36].

In the statement of the Barenblatt problem, we imagine that at time $t = 0$ an energy $\mathcal{E}_0$ is released in a static material at the point $r = 0$. (In our numerical code, we simulate this situation by initially assigning the energy density in the first computational cell a value of $\mathcal{E}_0/\Delta V_1$; no "injection" of energy across a cell boundary is required.) For $t > 0$, a propagating thermal front results with the zero-flux boundary condition $\partial T/\partial r|_{r=0} = 0$. The temperature profile at any given time is related to $\mathcal{E}_0$ by a statement of energy conservation:

$$Q \equiv \frac{\mathcal{E}_0}{\rho c_V} = \int_0^\infty T 4\pi r^2 \, dr = \text{const.} \tag{33}$$

The constant Q has physical units of deg $\cdot$ cm^3 and is a convenient parameter for formulating the similarity solution.

For purpose of benchmarking our numerical results, it is helpful to write down the analytical solution of the Barenblatt problem in spherical geometry. This can be summarized as follows. Letting r_f denote the position of the thermal front, the temperature profile for time $t > 0$ and $r < r_f$ is given by [3]

$$T = T_c \left(1 - \frac{r^2}{r_f^2}\right)^{1/b}, \tag{34}$$

where

$$r_f \equiv \xi_0 (\chi_0 Q^b t)^{1/(3b+2)}. \tag{35}$$

Here, $\chi_0 = \kappa_0/(\rho c_V)$ and the constant ξ_0 is a function of b only:

$$\xi_0 \equiv \left[\frac{3b+2}{2^{b-1} b \pi^b}\right]^{1/(3b+2)} \left[\frac{\Gamma(5/2 + 1/b)}{\Gamma(1 + 1/b)\Gamma(3/2)}\right]^{b/(3b+2)}. \tag{36}$$

In Eq. (36), the symbol Γ denotes the gamma function. The temperature at $r = 0$ is

$$T_c \equiv \frac{Q\xi_0^3}{r_f^3}\left[\frac{b\xi_0^2}{2(3b+2)}\right]^{1/b}. \tag{37}$$

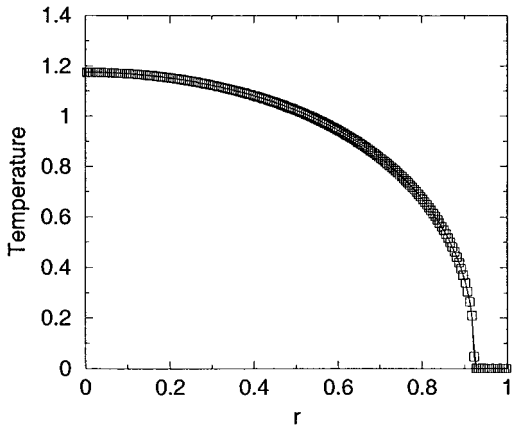

FIG. 1. Nonlinear thermal conduction from a point source (the Barenblatt problem) at $t = 0.3$ for $\mathcal{E}_0 = 10$ and $\kappa = T^{5/2}$. The solid line is the analytical solution.

For $r > r_f$, the temperature is given by its ambient value, which we take to be a positive number close to zero ($\sim 10^{-4}$). In the simulations, the temperature initially has this value everywhere except in the first cell, where it is equal to $\mathcal{E}_0/(\rho c_V \Delta V_1)$.

Figure 1 shows a comparison for $b = 5/2$ of Barenblatt's solution with the temperature profile obtained from the predictor–corrector Newton–Krylov code in which hydrodynamic motion was disabled. The agreement for this method, as well as for the other two schemes (not shown), is quite good. The initial energy in this case was $\mathcal{E}_0 = 10$, and the data are presented at $t = 0.3$ with $\kappa_0 = 1$, $c_V = (\gamma - 1)^{-1}$, and $\rho = 1$. Figure 2 shows the dependence of the L_2 solution error for the three different methods versus the target value for the front CFL. In all three cases, a computational grid with 200 cells was used, and the guess for the initial time step was very small: $\Delta t^{(0)} = 10^{-28}$. (This guess was based on a rough estimate for the characteristic diffusion time between the first and second cells: $\Delta r^2/\chi \sim 10^{-20}$.) The "exact" solution was taken as the predictor–corrector Newton–Krylov method with a front-CFL value of 10^{-3}.

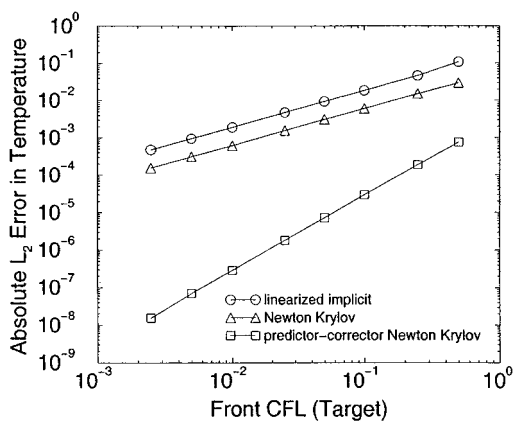

FIG. 2. Time-step convergence comparison using 200 computational cells for the Barenblatt problem shown in Fig. 1. The absolute L_2 error in the temperature profile is plotted versus the target front-CFL number. The nonlinear convergence tolerance for this series of runs was 10^{-7}. For the linearized implicit data, a linear tolerance of 10^{-9} was used.

TABLE I
The Approximate Cost in CPU Seconds of the Three Methods versus the Target Front-CFL Value for the Data in Fig. 2[a]

	Method		
Front-CFL value	LI	NK	PCNK
0.5	6	34	62
0.25	10	43	96
0.1	20	70	180
0.05	33	120	327
0.025	59	227	591
0.01	129	501	1271
0.005	243	831	2331
0.0025	466	1669	4362

[a] The abbreviations LI, NK, and PCNK stand for linearized implicit, Newton–Krylov, and predictor–corrector Newton–Krylov, respectively.

In this figure, it is clear that the predictor–corrector Newton–Krylov method, with its second-order convergence rate, is far superior to the other two schemes, which only show first-order convergence. For the same front-CFL target value, this scheme achieves errors that are approximately three to five orders of magnitude smaller than the linearized implicit approach. Furthermore, the algorithm is also significantly more efficient, as can be seen by consulting Table I, which shows the cost in CPU seconds for the data in Fig. 2. For example, at an absolute L_2 error of roughly 10^{-3}, the predictor–corrector Newton–Krylov code can be run with a target front-CFL value of 0.5 for a cost of 62 CPU seconds, whereas the linearized implicit algorithm requires a front-CFL value about 100 times smaller, which is approximately four times more expensive. This advantage, however, does not hold for the first-order Newton–Krylov code. While roughly a factor of 5 more accurate for a fixed front-CFL value, it is slightly less economical than the linearized implicit algorithm.

A second series of runs was performed using $\kappa(T) = T^{13/2}$. An exemplary solution profile is shown in Fig. 3 for $t = 1$ and $\mathcal{E}_0 = 10$. Note that the thermal front now has a much steeper

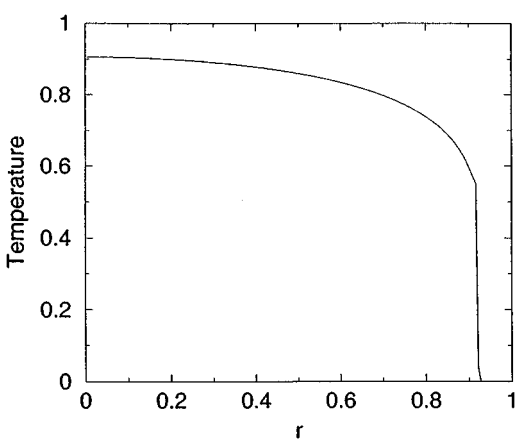

FIG. 3. Nonlinear thermal conduction from a point source at $t = 1$ for $\mathcal{E}_0 = 10$ and $\kappa = T^{13/2}$. Note the steeper thermal front in this figure compared to that in Fig. 1.

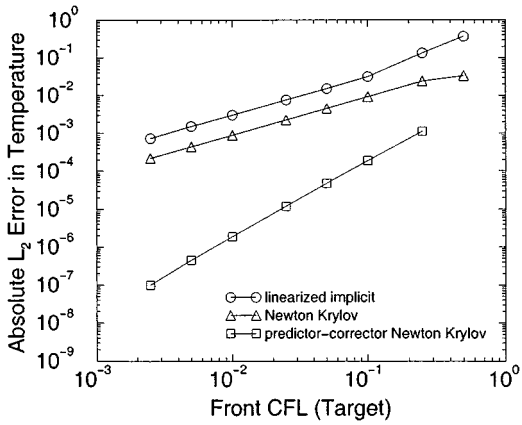

FIG. 4. Time-step convergence comparison using 200 computational cells for the Barenblatt problem shown in Fig. 3. The nonlinear convergence tolerance for this series of runs was 10^{-7}. For the linearized implicit data, a linear tolerance of 10^{-9} was used.

appearance than in Fig. 1. Here, the characteristic diffusion time was $\Delta r^2/\chi \sim 4 \times 10^{-46}$, and consequently, a rather conservative value of 10^{-54} was used for the initial time step, $\Delta t^{(0)}$. A convergence plot is shown in Fig. 4 for the three different numerical schemes, and a comparison of the CPU cost for these data is given in Table II. The "exact" solution was defined as the predictor–corrector Newton–Krylov method with a target front-CFL value of 10^{-3}. The trends here are similar to the previous Barenblatt problem. Despite the presence of a very steep solution profile, the predictor–corrector Newton–Krylov code still achieves second-order convergence. Once again, for an absolute L_2 error of roughly 10^{-3}, this scheme is about four times more efficient than the linearized implicit one.

We should point out that the reason a second-order, temporal convergence rate is seen in these problems is that both the "exact" and computed temperature profiles were determined at the same set of discrete points on the fixed numerical grid. The "exact" solutions were obtained by fine temporal—*but not spatial*—refinement and thus possess thermal fronts

TABLE II
The Approximate Cost in CPU Seconds of the Three Methods versus the Target Front-CFL Value for the Data in Fig. 4

	Method		
Front-CFL value	LI	NK	PCNK
0.5	10	51	—[a]
0.25	16	66	163
0.1	32	118	303
0.05	53	187	513
0.025	91	349	954
0.01	194	695	2068
0.005	360	1365	3620
0.0025	678	2402	7195

[a] Solution did not converge to specified tolerance within 15 Newton iterations during at least one time step.

with more or less the same "sharpness" as in the computed solutions. It is important to note, though, that these thermal fronts are not true discontinuities, but rather are composed of a sequence of steep line segments joined together and extending over several grid points. If an exact (and truly discontinuous) analytical solution to the Barenblatt problem were used to compute the error terms, the convergence rate would almost certainly not be second order; it would likely be much closer to zeroth order [5].

We should also comment here about an issue concerning the Crank–Nicolson structure of the corrector stage of the Newton–Krylov algorithm. For linear ($\kappa = \text{const}$) heat conduction problems, it is well known [6, 13] that while the Crank–Nicolson method is second-order accurate in time and mathematically stable, it gives completely wrong answers when the size of the time step is relatively large. In the present context of nonlinear thermal conduction, however, the use of a Crank–Nicolson formulation (with a time step based on the front-CFL method) does not appear to lead to any loss of fidelity in our numerical solutions, even though the time step sizes used in our calculations are often orders of magnitude larger than that required for linear stability (and accuracy). This observation is not too surprising, though, in light of the fact that in the nonlinear case, the coefficient of thermal conduction $\kappa(T)$ is large behind the thermal front (where the temperature is large, but the gradient is small), and small near its "foot" (the only place where the temperature gradient is appreciable). Such an arrangement tends to reduce the magnitude of the heat flux term, $-\kappa(\partial T/\partial r)$, in the energy equation [Eq. (6)], and thus permits high-fidelity solutions for somewhat larger time step sizes than could otherwise be used in a linear conduction problem.

B. *The "Smooth" Problem*

In this and the following section, we turn to more complicated classes of problems in which hydrodynamic motion is allowed to participate in the dynamics. Here, however, we first limit our attention to so-called "smooth" flows in which nearly discontinuous features such as steep thermal fronts and shock waves are absent from the numerical solution. The simulation of smooth flows is performed by following the evolution of the nonlinear waves that result from an initially nondiscontinuous energy density profile up until a point where the solution steepens appreciably, or becomes "shocked." The purpose of this section is to assess the convergence rate of our schemes under such circumstances. The boundary conditions for the hydrodynamic variables in this section are "reflective" and "outflow" conditions [13] at the left and right ends, respectively, of the computational domain. As in the previous section, zero-heat-flux boundary conditions are used for the temperature.

Our approach is to adopt a Gaussian profile for the initial energy-density profile. That is, we take

$$E(r)|_{t=0} = \frac{\mathcal{E}_0 \exp\left(-r^2/c_0^2\right)}{(c_0\sqrt{\pi})^3}, \tag{38}$$

where c_0 is a positive constant; for our purposes, we choose $c_0 = 1/4$. Note that the limit $c_0 \to 0$ gives a delta function at the origin, and the normalization factor is such that

$$\int_0^\infty E\, 4\pi r^2\, dr = \mathcal{E}_0. \tag{39}$$

In order to specify the energy-density profile in the numerical codes initially, though, we need to compute the discrete values of E in each cell. This is done by integrating Eq. (38)

over the jth cell from $r_{j-1/2}$ to $r_{j+1/2}$. The result is

$$E_j = \frac{\mathcal{E}_0\left[\mathrm{erf}\left(r_{j+1/2}/c_0\right) - \mathrm{erf}\left(r_{j-1/2}/c_0\right)\right] - 2\pi c_0^2\left[r_{j+1/2}E\left(r_{j+1/2}\right) - r_{j-1/2}E\left(r_{j-1/2}\right)\right]}{\Delta V_j}, \tag{40}$$

where the symbol "erf" denotes the standard error function.

We now follow the evolution of this smooth profile in time, being careful to stop the simulation at a point well before the solution becomes essentially discontinuous. We choose $\mathcal{E}_0 = 100$ and $\rho(r) = 1/r$ for the initial energy and density, respectively, and set the coefficient of thermal conduction to $\kappa(T) = T^{5/2}$. The guess for the initial time step here was $\Delta t^{(0)} = 10^{-10}$. The resulting simulated profiles of density, pressure, velocity, and temperature are shown in Fig. 5 at $t = 0.01$.

Figure 6 presents a comparison of the convergence properties of the three methods as applied to this coupled problem. The "exact" solution was defined to be the predictor–corrector Newton–Krylov result with a target front-CFL value of 10^{-4}. Note that while the convergence rate of all three schemes is nominally first order, the predictor–corrector Newton–Krylov method still has an absolute L_2 error in the temperature profile that is always two to three orders of magnitude smaller than that obtained with the linearized implicit algorithm for a fixed front-CFL value. Compared to the Barenblatt problem, though, the gain in efficiency of this method has decreased somewhat, as can be seen from the CPU costs displayed in Table III for the data in Fig. 6. For the few data points that permit comparison

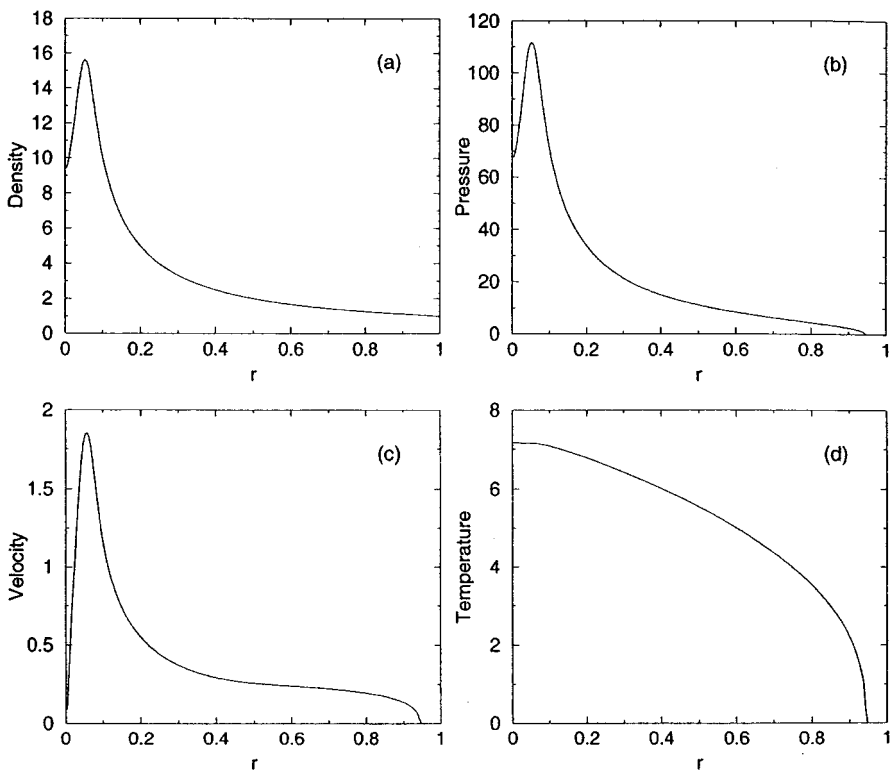

FIG. 5. The profiles of (a) density, (b) pressure, (c) velocity, and (d) temperature at $t = 0.01$ for an initially Gaussian energy-density profile with $\mathcal{E}_0 = 100$, $\kappa = T^{5/2}$, and $\rho(r) = 1/r$. The solution has not yet "shocked."

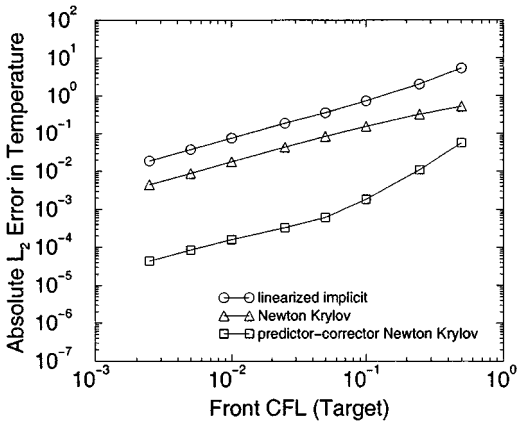

FIG. 6. Time-step convergence comparison using 200 computational cells for the "smooth" problem shown in Fig. 5. The nonlinear convergence tolerance for this series of runs was 10^{-8}. For the linearized implicit data, a linear tolerance of 10^{-8} was used.

at the same error level, we see that the predictor–corrector Newton–Krylov code provides about a factor of 2 savings over the linearized implicit algorithm. Compared to this latter method, the other Newton–Krylov scheme also offers consistently more accurate solutions, but is not as economical.

We should remark on an issue concerning the first-order convergence rate of the predictor–corrector Newton–Krylov scheme seen in this example. Since nearly discontinuous features such as steep thermal gradients and shock fronts are absent from the numerical solution, one might expect this method to achieve approximately second-order convergence, as in the Barenblatt problems. The resolution of this apparent paradox has to do with the particular Godunov scheme employed here for integrating the hydrodynamic equation; it is almost certainly not the result of any coupling issues between the hydrodynamic and conduction stages of the solution methodology. The two-step Hancock procedure that we use (see Appendix) is only second-order accurate in space and time for a *fixed* Courant number, $v\Delta t/\Delta r$ [33]. In Fig. 6, though, the grid spacing Δr is constant, and convergence in Δt alone for Hancock's method is only first order. Thus, given the way in which we have chosen to construct convergence plots, we can expect to see first-order behavior for any

TABLE III
The Approximate Cost in CPU Seconds of the Three Methods versus the Target Front-CFL Value for the Data in Fig. 6

	Method		
Front-CFL value	LI	NK	PCNK
0.5	2	10	21
0.25	2	11	23
0.1	3	13	31
0.05	4	18	44
0.025	6	26	70
0.01	13	49	143
0.005	22	90	243
0.0025	41	161	441

coupled, radiative hydrodynamic problem to which our numerical schemes are applied. This of course will include the point explosions discussed in the next section.

C. *Point Explosions*

In this section, we study the performance of the three numerical schemes for modeling one-dimensional point explosions with nonlinear heat conduction. An explosion is characterized by the release of a large amount of energy in a small region of space, and this is simulated in our dynamical codes by initializing the energy density E with a discontinuous profile. That is, at time $t = 0$, the value of E is set to a small value (10^{-3}) in all computational cells except the first one, where it is assigned a much larger value ($\sim$10–1000, typically). Note that this corresponds to simply letting the parameter c_0 in the previous section approach zero. The solution profiles are then allowed to evolve in time according to Eqs. (4)–(6). In general, both a thermal front and a shock wave appear in the numerical solution. This class of problem was recently studied by Shestakov [23] within the framework of a Lagrangian code [39]. The hydrodynamic boundary conditions used here are the same as in the previous section.

The first problem that we consider is a point explosion into a cold quiescent fluid with ambient density $\rho = 1$. The initial energy $\mathcal{E}_0 = 20$ is "released" in the first computational cell at $t = 0$. The coefficient of thermal conduction is assumed to be a function of temperature only with $\kappa(T) = T^{5/2}$. Profiles of fluid density, pressure, velocity, and temperature for this case are shown in Fig. 7 at time $t = 0.05$. In this simulation, a shock wave is clearly visible at $r \sim 0.93$. Figure 8 shows a time-step convergence comparison for the solution to this problem obtained with the three numerical schemes. For each case, a computational grid with 100 cells was used and the guess for the initial time step was $\Delta t^{(0)} = 10^{-12}$. The "exact" solution was taken as the predictor–corrector Newton–Krylov method with a target front-CFL value of 10^{-4}.

Not surprisingly, the predictor–corrector Newton–Krylov method gives results that are consistently more accurate (by a factor of approximately 10–15) for the same target front-CFL value than both the Newton–Krylov and the linearized–implicit schemes, with the latter being the least accurate. Compared to previous examples, though, the greater accuracy here has become more costly, as can be seen in Table IV. Now, the predictor–corrector Newton–Krylov method is only slightly more efficient than the other two. For example, an overall L_2 error in the temperature profile of 1.5×10^{-3} or less requires at least 58 CPU seconds with the predictor–corrector Newton–Krylov code, whereas an equally reliable answer can be obtained by expending just 20% more computational effort (69 CPU seconds), with the linearized implicit scheme. The comparisons are much less favorable for the Newton–Krylov code without the predictor–corrector architecture.

The next simulation that we discuss is that of a point explosion with a greater nonlinear temperature dependence for the coefficient of thermal conduction. In this case, the density and temperature exponents for κ are $a = -2$ and $b = 13/2$, respectively. The ambient fluid density has a spatial dependence given by $\rho(r) = r^{-2.111}$. These particular parameters were chosen in an effort to compare results with those presented previously by Shestakov [23], as well as with solutions given by Reinicke and Meyer-ter-Vehn [24]. It is interesting to note that this particular example belongs to a special class of similarity solutions that exists in radiation hydrodynamics for certain forms of the coefficient of thermal conduction and certain spatial profiles of the ambient density [24]. For this class of problems, the shock and thermal front both evolve with the same power-law dependence in time so that if the

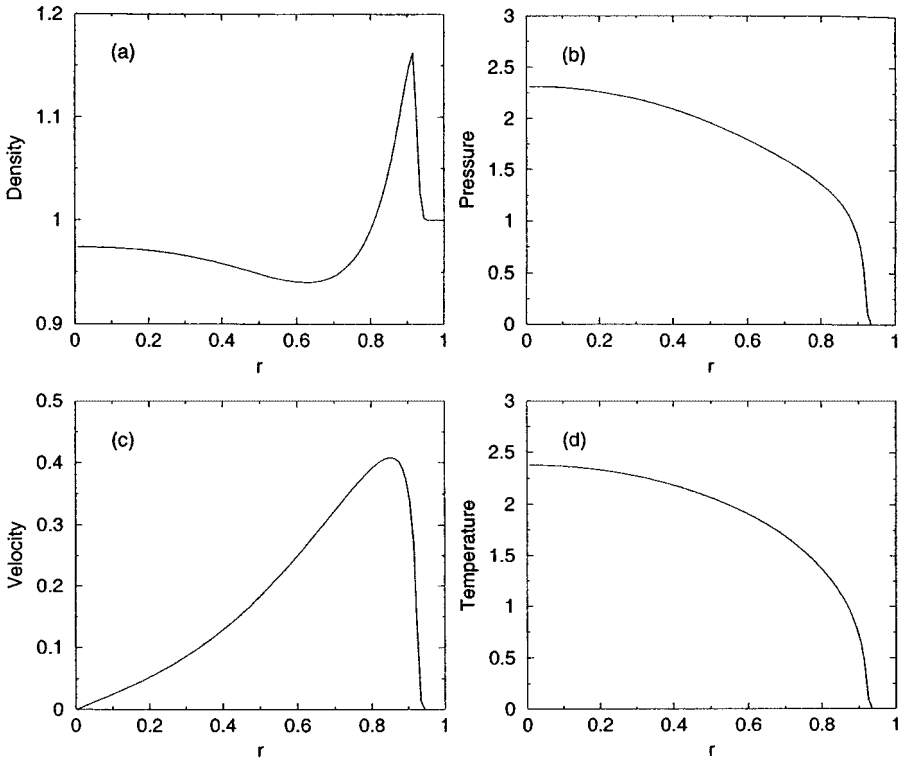

FIG. 7. Simulation of a weak point explosion with an initial energy of $\mathcal{E}_0 = 20$. Profiles of (a) density, (b) pressure, (c) velocity, and (d) temperature are shown at $t = 0.05$. The coefficient of thermal conduction here is $\kappa = T^{5/2}$, and $\rho = 1$ initially.

initial energy of the explosion is large enough, a thermal front will always precede the hydrodynamic shock wave.

An example of such a "strong" point explosion is depicted in Fig. 9 for an initial energy $\mathcal{E}_0 = 235$. Simulated profiles of fluid density, pressure, velocity, and temperature are shown

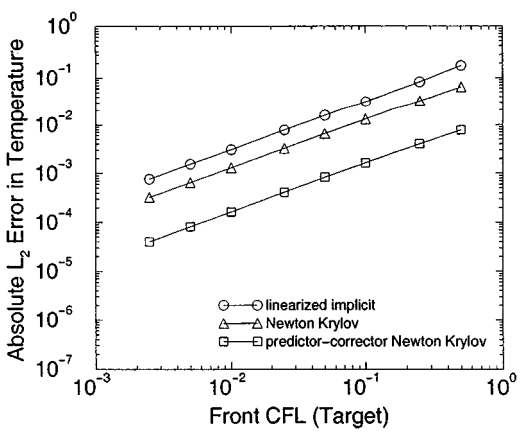

FIG. 8. Time-step convergence comparison using 100 computational cells for the weak point explosion shown in Fig. 7. The nonlinear convergence tolerance for this series of runs was 10^{-8}. For the linearized implicit data, a linear tolerance of 10^{-8} was used.

TABLE IV
The Approximate Cost in CPU Seconds of the Three Methods versus the Target Front-CFL Value for the Data in Fig. 8

	Method		
Front-CFL value	LI	NK	PCNK
0.5	2	12	27
0.25	3	14	35
0.1	6	23	58
0.05	9	34	106
0.025	16	66	162
0.01	37	138	372
0.005	69	274	625
0.0025	123	453	1281

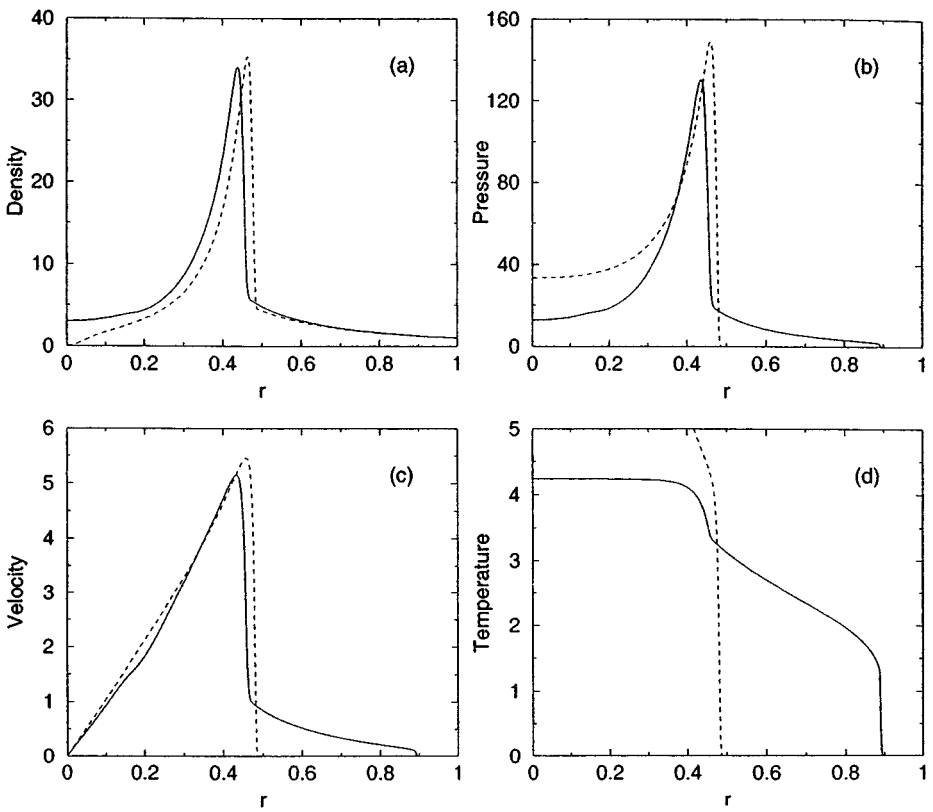

FIG. 9. Simulation of a strong point explosion with an initial energy of $\mathcal{E}_0 = 235$. Profiles of (a) density, (b) pressure, (c) velocity, and (d) temperature are shown at $t = 0.05145$. The coefficient of thermal conduction here is $\kappa = \rho^{-2}T^{13/2}$, and $\rho(r) = r^{-2.111}$ initially. A thermal front at $r \sim 0.9$ is visible ahead of a shock wave at $r \sim 0.45$. The dotted lines denote the solution in the absence of radiative conduction (i.e, the pure hydrodynamic limit known as the "Sedov" solution [37]).

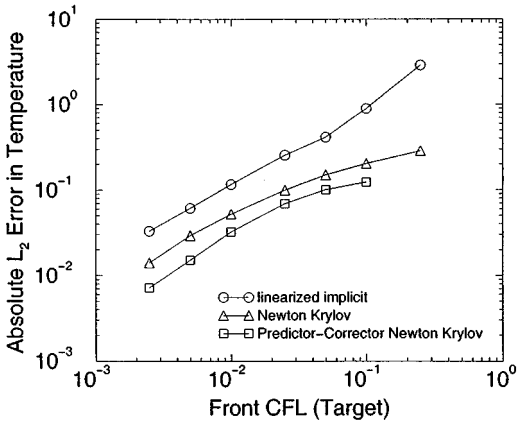

FIG. 10. Time-step convergence comparison using 400 computational cells for the strong point explosion shown in Fig. 9. The nonlinear convergence tolerance for this series of runs was 10^{-8}. For the linearized implicit data, a linear tolerance of 10^{-9} was used.

at time $t = 0.05145$. The dotted lines in this figure denote the pure hydrodynamic solution [37]. Note the presence of a thermal front in Fig. 9 at $r \sim 0.9$, which is ahead of a shock located at $r \sim 0.45$. The guess for the initial time step here was $\Delta t^{(0)} = 10^{-12}$.

In Fig. 10, we show the results of a time-step convergence comparison using the three different numerical schemes for this example. A computational grid with 400 cells was used throughout because coarser grids for this problem gave solutions that were not well resolved near the origin. Once again, we have plotted the absolute L_2 error in the temperature profile versus the target front-CFL value. Since the computations were more intensive here than in the other point explosion, we chose the predictor–corrector Newton–Krylov method with a slightly larger front-CFL target value (10^{-3}) as the "exact" solution for this problem. Generally speaking, the conclusions regarding accuracy at a fixed front-CFL value here are the same as for the previous two examples, but now the separation between convergence lines has been reduced somewhat. The predictor–corrector Newton–Krylov scheme still yields

TABLE V
The Approximate Cost in CPU Seconds of the Three Methods versus the Target Front-CFL Value for the Data in Fig. 10

	Method		
Front-CFL value	LI	NK	PCNK
0.25	19	186	—[a]
0.10	35	283	848
0.05	60	450	1359
0.025	108	726	2208
0.01	245	1477	4517
0.005	445	2556	7801
0.0025	821	4292	13649

[a] Solution did not converge to specified tolerance within 15 Newton iterations during at least one time step.

the most accurate solution, but at a significantly greater comparative cost than before, as shown in Table V. An L_2 error of approximately 3.5×10^{-2} in the temperature profile, for example, now requires 372 CPU seconds, whereas an equally accurate solution can be obtained for only 123 CPU seconds using the linearized implicit method.

V. SUMMARY AND FUTURE WORK

In this paper, we have studied the performance of three numerical methods for integrating the equations of radiation hydrodynamics in the equilibrium diffusion limit and the low-energy-density regime. Accuracy and efficiency comparisons were made by considering several, one-dimensional, spherically symmetric test problems. For each of the methods, an "operator split" approach was used in which the problem was separated into two parts: a purely hydrodynamic piece followed by a conduction module. All methods used the same technique to integrate the Euler equations in the hydrodynamic stage (a high-resolution Godunov scheme [4–7]), but differed in the way they incorporated the contributions of nonlinear thermal conduction into the problem.

In the first approach, a linearized implicit method was used that simplifies nonlinear terms by evaluating them with old time-level data. The resulting system of linear equations was then solved with a linear iterative technique. Such a method lacks (nonlinear) numerical consistency [5, 14] since it makes no attempt to monitor or converge nonlinearities within a computational time step. In the field of radiation hydrodynamics, however, this is a common practice [1, 22]. The performance of this linear scheme was compared with that of two *consistent*, nonlinear iterative solvers based on Jacobian-free Newton–Krylov techniques [17, 18], which do converge nonlinear terms within a computational time step. These techniques offer a significant advantage over the standard Newton's method [15] in that they do not require the formation and inversion of a Jacobian matrix, yet still achieve desirable nonlinear-convergence behavior. The two Newton–Krylov codes used in this investigation differed in their order of accuracy in time; one was strictly first-order accurate, while the other sought to achieve second-order accuracy with a predictor–corrector architecture.

Not surprisingly, the Newton–Krylov algorithms always gave more accurate results. This finding is consistent with studies of radiation diffusion without hydrodynamics [16], although for that class of problems, even more impressive gains in accuracy were made by using Newton–Krylov methods. In most cases studied here, the same level of accuracy was preserved by taking time steps in the predictor–corrector Newton–Krylov scheme that were at least an order of magnitude larger than in the linearized implicit approach. This is true for pure diffusion (the Barenblatt limit), and in the coupled, radiative hydrodynamic problems. Because our hydrodynamic scheme is only first-order accurate in time, though, the overall convergence rate for coupled problems was also limited to first order, even in the absence of nearly discontinuous profiles such as shocks in the solution. For the Barenblatt problems, the predictor–corrector Newton–Krylov method achieved a second-order convergence rate, even for a problem with a steep thermal gradient.

We should mention that the selection of the nonlinear and linear tolerances was an important consideration in this study. When this work was first initiated, a value of 10^{-5} was used for both the nonlinear convergence tolerance in the Newton–Krylov schemes and the tolerance of the linear iterative solver in the linearized implicit approach. For the types of problems and range of front-CFL values considered, though, this value was not always sufficient and needed to be reduced to 10^{-9}–10^{-7} to provide reasonable

convergence behavior, particularly below a target front-CFL value of 10^{-2}. Newton–Krylov runs performed for the strong point explosion at a tolerance of 10^{-5}, for example, often had an error that generally diminished as the target front-CFL value was lowered, but sometimes "stalled out" or contained local sections that were not monotonically decreasing. This behavior was witnessed for several problems in which the linearized implicit method was used as well. The explanation for this behavior is likely the accumulation of round-off error over the hundreds of thousands of time steps that were sometimes required for these calculations.

Perhaps the most salient issue of this study is that of Newton–Krylov efficiency for radiative hydrodynamic calculations. Quite frankly, the results for the coupled problems are somewhat disappointing in comparison to earlier applications of Newton–Krylov algorithms [16, 30], which consistently demonstrated the superiority and CPU-cost efficiency of these methods over standard linearized techniques. While the predictor–corrector Newton–Krylov code used here proved much more efficient for the Barenblatt problem and somewhat more efficient for the weak point explosion, this advantage waned for the strong point explosion. Our assessment of this finding is that it is likely a result of the time splitting of the hydrodynamic and diffusion integrators in our solution methodology, and the application of a nonlinear iterative scheme to only the latter.

We wish to point out that this paper represents a first attempt to examine the utility of Newton–Krylov methods for radiation hydrodynamics. In the future, our intent is to abandon the splitting of hydrodynamic and diffusion operators altogether and study the performance of the Newton–Krylov technique as applied to the full set of coupled, radiative hydrodynamic equations. In addition, it will be desirable to model the dynamics of systems in the high energy-density regime, where the effects of radiation pressure and energy density are not neglected. Multidimensions and nonideal equations of state are other aspects of more realistic simulations that we also wish to consider. It is here that we expect to see the true efficacy of Newton–Krylov methods for accurate and efficient numerical calculations of radiative hydrodynamic problems.

APPENDIX: PRIMITIVE-VARIABLE RIEMANN SOLVER

In this appendix, we outline the details of the Riemann solver used in the Godunov method described in Section III. The principal task is to approximate time-centered values of $\mathbf{F}$ and $\mathbf{G}$ in Eq. (11) at the cell interfaces of our one-dimensional spherical grid. Our approach differs somewhat from the conventional formulation of Godunov's method in that we choose to work in terms of an augmented primitive-variable representation [40]. Defining the vector $\mathbf{W} \equiv (\rho, v, \rho\varepsilon, p)^{\mathcal{T}}$, where the superscript $\mathcal{T}$ denotes the transpose, we follow a two-step Hancock procedure discussed by Huynh in Ref. [33]. For convenience, we shall sometimes drop subscripts and superscripts for ease of notation when it causes no confusion.

In terms of primitive variables, Eq. (7) can be rewritten as

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A}_p \frac{\partial \mathbf{W}}{\partial r} + \mathbf{K} = 0, \tag{41}$$

where $\mathbf{K} \equiv [\rho v, 0, v(\rho\varepsilon + p), \gamma p v]^{\mathcal{T}}$, and the matrix $\mathbf{A}_{\mathrm{p}}$ is given by

$$\mathbf{A}_{\mathrm{p}} \equiv \begin{pmatrix} v & \rho & 0 & 0 \\ 0 & v & 1/\rho & 0 \\ 0 & \rho\varepsilon + p & v & 0 \\ 0 & \gamma p & 0 & v \end{pmatrix}. \tag{42}$$

Note that the third equation is superfluous with the fouth in this system. Since the specific internal energy and the pressure are uniquely related by the relation $\rho\varepsilon = p/(\gamma - 1)$ for ideal gases, it is not necessary to have two separate equations for advancing $\rho\varepsilon$ and p in time. Nevertheless, we choose to purposely incorporate this redundancy in the governing equations so as to avoid additional evaluations of the equation of state.

The next step in Hancock's method is to determine the spatial gradients $\mathbf{S}_j$ of the primitive variables in each cell. These are approximated by

$$\mathbf{S}_j^n \approx \frac{\mathbf{W}_{j+1}^n - \mathbf{W}_{j-1}^n}{2\,\Delta r}, \tag{43}$$

which are usually limited to ensure locally monotonic profiles [42]. The spatial gradients are then used to find time-centered expressions for the primitive variables at the cell edges using a Taylor series expansion in both space and time. Keeping only first-order terms, we have

$$\mathbf{W}_{j+1/2,L}^{n+1/2} \approx \mathbf{W}_j^n + \frac{\Delta r}{2}\mathbf{S}_j^n + \frac{\Delta t}{2}\frac{\partial \mathbf{W}_j^n}{\partial t}, \tag{44}$$

$$\mathbf{W}_{j+1/2,R}^{n+1/2} \approx \mathbf{W}_{j+1}^n - \frac{\Delta r}{2}\mathbf{S}_{j+1}^n + \frac{\Delta t}{2}\frac{\partial \mathbf{W}_{j+1}^n}{\partial t}. \tag{45}$$

Subscripts L and R in Eqs. (44) and (45) denote "left" and "right" states, respectively. The time derivatives on the right hand sides of these equations are determined from the Euler equations expressed in primitive variable form. Explicitly, the individual components of $\partial\mathbf{W}/\partial t$ are

$$\frac{\partial \rho}{\partial t} = -\frac{\partial}{\partial V}(A\rho v), \tag{46}$$

$$\frac{\partial v}{\partial t} = -v\frac{\partial v}{\partial r} - \frac{1}{\rho}\frac{\partial p}{\partial r}, \tag{47}$$

$$\frac{\partial \rho\varepsilon}{\partial t} = -\frac{\partial}{\partial V}(A\rho\varepsilon v) - p\frac{\partial Av}{\partial V}, \tag{48}$$

$$\frac{\partial p}{\partial t} = -\frac{\partial}{\partial V}(Apv) - (\gamma - 1)p\frac{\partial Av}{\partial V}. \tag{49}$$

In this study, we choose to limit our attention to piecewise constant data only, such that $\mathbf{S}_j = 0$ for all j.

On either side of each cell interface, we now have time-centered expressions for the primitive variables. The resulting sequence of Riemann problems is then solved to give

$$\mathbf{W}_{j+1/2}^{n+1/2} = \mathcal{R}\big(\mathbf{W}_{j+1/2,L}^{n+1/2}, \mathbf{W}_{j+1/2,R}^{n+1/2}\big), \tag{50}$$

where $\mathcal{R}$ stands for "Riemann solver." Finally, these quantities are used to compute the time-centered vectors $\mathbf{F}^{n+1/2}_{j\pm1/2} \approx \mathbf{F}(\mathbf{W}^{n+1/2}_{j\pm1/2})$ and $\mathbf{G}^{n+1/2}_{j\pm1/2} \approx \mathbf{G}(\mathbf{W}^{n+1/2}_{j\pm1/2})$, and update the Euler equations from time level n to $n+1$ according to Eq. (11).

The one-dimensional Riemann problem, represented by the symbol $\mathcal{R}$ in Eq. (50), describes the evolution of a planar discontinuity separating two uniform fluid states; it is the key ingredient in Godunov's method. Although there is no closed-form solution of the Riemann problem for the Euler equations (even for ideal equations of state), an iterative scheme can be employed to compute the exact solution numerically to an arbitrary level of precision [43]. In a practical computation, though, it may be necessary to solve the Riemann problem billions of times [7], which can be a demanding and undesirable task. Our approach here is not to seek the *exact* solution of Eq. (50), but rather to find a good approximation to it using a less expensive Riemann solver.

Over the past two decades, many clever and sophisticated solution methodologies have been devised to find approximate solutions to the Riemann problem. One such approach is Roe's method [27], which we use exclusively in this study. Roe's scheme is probably the best known of all approximate Riemann solvers and works well even for relatively strong shocks [7]. Since it was first introduced in 1981, various refinements have been made, but here we limit our attention to a modified version of Roe's original conserved-variable method. The principal difference in our implementation is that we formulate the predictor stage of our solver in terms of primitive variables instead of conserved ones. Because our approach may be somewhat unfamiliar to the reader, details of the scheme are provided below.

In discussing the modified version of Roe's solver used here, it is convenient to first introduce shorthand notation for the left and right states that appear in Eq. (50). We temporarily drop the subscript $j+1/2$ and the superscript $n+1/2$ and simply write

$$\mathbf{W}_L \equiv \begin{bmatrix} \rho_L \\ v_L \\ (\rho\varepsilon)_L \\ p_L \end{bmatrix}, \quad \mathbf{W}_R \equiv \begin{bmatrix} \rho_R \\ v_R \\ (\rho\varepsilon)_R \\ p_R \end{bmatrix}, \tag{51}$$

for the time-centered vectors of primitive variables on either side of each cell interface. The so-called "Roe-average" of these states [27], which we denote by a tilde, is defined by

$$\tilde{\mathbf{W}} \equiv \frac{\sqrt{\rho_L}\,\mathbf{W}_L + \sqrt{\rho_R}\,\mathbf{W}_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}. \tag{52}$$

This averaging procedure is a central aspect of Roe's Riemann solver. Essentially, the primitive-variable formulation of Roe's method requires that we replace the variable matrix $\mathbf{A}_\mathrm{p}$ in Eq. (42) with a constant matrix, *to wit* the Roe-average of $\mathbf{A}_\mathrm{p}$. Specifically, the Roe-averaged matrix $\tilde{\mathbf{A}}_\mathrm{p}$ is given by

$$\tilde{\mathbf{A}}_\mathrm{p} \equiv \begin{pmatrix} \tilde{v} & \tilde{p} & 0 & 0 \\ 0 & \tilde{v} & \tilde{\rho}^{-1} & 0 \\ 0 & \tilde{\rho}\tilde{\varepsilon} + \tilde{p} & \tilde{v} & 0 \\ 0 & \gamma\,\tilde{p} & 0 & \tilde{v} \end{pmatrix}, \tag{53}$$

which has eigenvalues

$$\tilde{\lambda}_1 \equiv \tilde{v} - \tilde{c}, \quad \tilde{\lambda}_2 = \tilde{\lambda}_3 \equiv \tilde{v}, \quad \tilde{\lambda}_4 \equiv \tilde{v} + \tilde{c}, \tag{54}$$

and right eigenvectors

$$\mathbf{R}^{(1)} \equiv \begin{bmatrix} 1 \\ -\tilde{c}/\tilde{\rho} \\ (\tilde{\rho}\varepsilon + \tilde{p})/\tilde{\rho} \\ \tilde{c}^2 \end{bmatrix}, \quad \mathbf{R}^{(2)} \equiv \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{R}^{(3)} \equiv \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{R}^{(4)} \equiv \begin{bmatrix} 1 \\ \tilde{c}/\tilde{\rho} \\ (\tilde{\rho}\varepsilon + \tilde{p})/\tilde{\rho} \\ \tilde{c}^2 \end{bmatrix}. \tag{55}$$

Here, the symbol $\tilde{c}$ represents the sound speed for the Roe-averaged state: $\tilde{c} \equiv \sqrt{\gamma \tilde{p}/\tilde{\rho}}$.

The final step is to express the difference between the left and right states of primitive variables as a superposition of right eigenvectors,

$$\Delta\mathbf{W} \equiv \mathbf{W}_R - \mathbf{W}_L = \sum_{i=1}^{4} \alpha_i \, \mathbf{R}^{(i)}, \tag{56}$$

where the coefficients (right eigenvector projections) α_i are given by

$$\alpha_1 \equiv \frac{1}{2\,\tilde{c}^2}(\Delta p - \tilde{\rho}\,\tilde{c}\,\Delta v), \tag{57}$$

$$\alpha_2 \equiv \Delta\rho - \Delta p/\tilde{c}^2, \tag{58}$$

$$\alpha_3 \equiv \Delta\rho\varepsilon - \frac{1}{\tilde{\rho}\,\tilde{c}^2}[\Delta p(\tilde{\rho}\varepsilon + \tilde{p})], \tag{59}$$

$$\alpha_4 \equiv \frac{1}{2\,\tilde{c}^2}(\Delta p + \tilde{\rho}\,\tilde{c}\,\Delta v). \tag{60}$$

With these data, the time-centered vectors of primitive variables at each cell interface can be written as

$$\mathbf{W}_{j+1/2}^{n+1/2} = \frac{1}{2}(\mathbf{W}_L + \mathbf{W}_R) - \frac{1}{2}\sum_{i=1}^{4} \mathrm{sign}(\tilde{\lambda}_i)\,\alpha_i\,\mathbf{R}^{(i)}, \tag{61}$$

which completes the Riemann solution. In practice, the wave speeds in the solution are bounded by comparing $\tilde{\lambda}_1$ with $v_L - c_L$, $\tilde{\lambda}_2$ with v_L and v_R, and $\tilde{\lambda}_4$ with $v_R + c_R$, and always selecting the maximum. The symbols c_L and c_R denote the sound speeds in the left and right states, respectively: $c_L \equiv \sqrt{\gamma p_L/\rho_L}$ and $c_R \equiv \sqrt{\gamma p_R/\rho_R}$. We should also remark that the procedure outlined above can be described as an "all-shock" method because it allows for rarefactive shocks with a concomitant violation of the second law of thermodynamics. Various "entropy fixes" have been devised for Roe's Riemann solver, but we shall not attempt to describe them here since they are not required for the class of test problems considered in this investigation. For further reading on this subject, the reader is referred to Refs. [5, 44].

ACKNOWLEDGMENT

This work was performed under the auspices of the U.S. Department of Energy through Contract W-7405-ENG-36.

REFERENCES

1. R. L. Bowers and J. R. Wilson, *Numerical Modeling in Applied Physics and Astrophysics* (Jones & Bartlett, Boston, 1991).
2. D. Milhalas and B. W. Milhalas, *Foundations of Radiation Hydrodynamics* (Oxford Univ. Press, New York, 1984).
3. Y. B. Zel'dovich and Y. P. Raizer, *Physics of Shock Waves and High-Temperature Hydrodynamic Phenomena, Vols. I and II* (Academic Press, New York, 1966, 1967).
4. C. Hirsch, *Numerical Computation of Internal and External Flows* (Wiley, Chichester, 1990).
5. R. J. LeVeque, *Numerical Methods for Conservation Laws* (Birkhäuser, Basel, 1992).
6. C. A. Fletcher, *Computational Techniques for Fluid Dynamics* (Springer-Verlag, Berlin/New York, 1997).
7. E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics* (Springer-Verlag, Berlin/New York, 1997).
8. W. Dai and P. R. Woodward, Numerical simulations for radiation hydrodynamics. I. Diffusion limit, *J. Comput. Phys.* **142**, 182 (1998).
9. W. Dai and P. R. Woodward, Numerical simulations for radiation hydrodynamics. II. Transport limit, *J. Comput. Phys.* **157**, 199 (2000).
10. R. B. Lowrie, J. E. Morel, and J. A. Hittinger, The coupling of radiation and hydrodynamics, *Astrophys. J.* **521**, 432 (1999).
11. D. S. Balsara, An analysis of the hyperbolic nature of the equations of radiation hydrodynamics, *J. Quant. Spectrosc. Radiat. Transfer* **61**, 617 (1999).
12. D. S. Balsara, Linearized formulation of the Riemann problem for radiation hydrodynamics, *J. Quant. Spectrosc. Radiat. Transfer* **61**, 629 (1999).
13. P. J. Roache, *Computational Fluid Dynamics* (Hermosa, Albuquerque, NM 1998).
14. J. H. Ferziger and M. Perić, *Computational Methods for Fluid Dynamics* (Springer-Verlag, Berlin/New York, 1996).
15. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran*, 2nd ed. (Cambridge Univ. Press, Cambridge, UK 1992).
16. D. A. Knoll, W. J. Rider, and G. L. Olsen, An efficient nonlinear solution method for non-equilibrium radiation diffusion, *J. Quant. Spectrosc. Radiat. Transfer* **63**, 15 (1999).
17. Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems, *SIAM J. Sci. Stat. Comput.* **7**, 856 (1986).
18. P. N. Brown and Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, *SIAM J. Sci. Stat. Comput.* **11**, 450 (1990).
19. S. K. Godunov, Difference methods for the numerical calculation of the equations of fluid dynamics, *Math. Sb.* **47**, 271 (1959).
20. G. C. Pomraning, *The Equations of Radiation Hydrodynamics* (Pergamon, Oxford, 1973).
21. W. Hackbush, *Muti-grid Methods and Applications* (Springer-Verlag, Berlin/New York, 1985).
22. C. Baldwin, P. N. Brown, R. Falgout, F. Graziani, and J. Jones, Iterative linear solvers in a 2D radiation-hydrodynamics code: methods and performance, *J. Comput. Phys.* **154**, 1 (1999).
23. A. I. Shestakov, Time-dependent simulations of point explosions with heat conduction, *Phys. Fluids* **11**, 1091 (1999).
24. P. Reinicke and J. Meyer-ter-Vehn, The point explosion with heat conduction, *Phys. Fluids A* **3**, 1807 (1991).
25. R. E. Marshak, Effect of radiation on shock wave behavior, *Phys. Fluids* **1**, 24 (1958).
26. P. Colella, A direct Eulerian MUSCL scheme for gas dynamics, *SIAM J. Sci. Stat. Comput.* **6**, 104 (1985).
27. P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* **43**, 357 (1981).
28. C. T. Kelly, *Iterative Methods for Linear and Nonlinear Equations* (Soc. for Indus. & Appl. Math., Philadelphia, 1995).
29. Y. Saad, *Iterative Methods for Sparse Linear Systems* (PWS–Kent, Boston, 1996).

30. V. A. Mousseau, D. A. Knoll, and W. J. Rider, Physics-based preconditioning and the Newton–Krylov method for non-equilibrium radiation diffusion, *J. Comput. Phys.* **160**, 743 (2000).
31. D. A. Knoll and W. J. Rider, A multigrid preconditioned Newton–Krylov method, SIAM *J. Sci. Comput.* **21**, 691 (1999).
32. W. J. Rider, D. A. Knoll, and G. L. Olson, A multigrid Newton–Krylov method for multimaterial equilibrium radiation diffusion, *J. Comput. Phys.* **152**, 164 (1999).
33. H. T. Huynh, Accurate upwind methods for the Euler equations, *SIAM. J. Numer. Anal.* **32**, 1565 (1995).
34. J. Crank and P. Nicolson, A practical method for numerical evaluation of solution of partial differential equations of the heat conduction type, *Proc. Camb. Philos. Soc.* **43**, 50 (1947).
35. W. J. Rider and D. A. Knoll, Time step size selection for radiation diffusion calculations, *J. Comput. Phys.* **152**, 790 (1999).
36. G. I. Barenblatt, *Similarity, Self-Similarity and Intermediate Asymptotics* (Consultants Bureau, New York, 1979).
37. L. I. Sedov, *Similarity and Dimensional Methods in Mechanics* (Academic Press, New York, 1959).
38. L. D. Landau and E. M. Lifshitz, *Fluid Mechanics*, 2nd ed. (Pergamon, Oxford, 1987).
39. A. I. Shestakov, M. K. Prasad, J. L. Milovich, N. A. Gentile, J. F. Painter, and G. Furnish, *The Radiation Hydrodynamic ICF3D Code*, Report No. UCRL-JC-124448, Lawrence Livermore National Laboratory, 1997.
40. W. J. Rider, An adaptive Riemann solver using a two-shock approximation, *Comput. Fluids* **28**, 741 (1999).
41. L. Spitzer, *Physics of Fully Ionized Gases* (Interscience, New York, 1962).
42. B. Van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method, *J. Comput. Phys.* **32**, 101 (1979).
43. E. S. Oran and J. P. Boris, *Numerical Simulation of Reactive Flow* (Elsevier, Amsterdam/New York, 1987).
44. A. Harten, High resolution schemes for hyperbolic conservation laws, *J. Comput. Phys.* **49**, 357 (1983).