Fully implicit solution of large-scale non-equilibrium radiation diffusion with high order time integration ☆

Peter N. Brown, Dana E. Shumaker, Carol S. Woodward *

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, PO Box 808, L-561, Livermore, CA 94551, USA

Abstract

We present a solution method for fully implicit radiation diffusion problems discretized on meshes having millions of spatial zones. This solution method makes use of high order in time integration techniques, inexact Newton–Krylov nonlinear solvers, and multigrid preconditioners. We explore the advantages and disadvantages of high order time integration methods for the fully implicit formulation on both two- and three-dimensional problems with tabulated opacities and highly nonlinear fusion source terms.
© 2004 Elsevier Inc. All rights reserved.

Keywords: Radiation diffusion; Implicit methods; Time integration; Parallel computing

1. Introduction

The simulation of nonlinear and coupled physical phenomena requires efficient numerical methods for solution. One application which is computationally intensive is modeling transport of neutral particles. Simulations of this application are important in calculations relevant to astrophysics, shielding, inertial confinement fusion, and atmospheric radiation. The Boltzmann transport equation is often used for modeling these problems. Due to the six degrees of freedom present in this equation, diffusion approximations are often used to give coarse estimates of solutions. Solution of the nonlinear diffusion approximation is still a demanding task, however, and the need for computational methods to efficiently solve these problems is still required.

In this paper, we present a fully implicit solution method for radiation diffusion problems. Fully implicit methods can allow for larger time steps and more accurate solves for a given amount of work than can explicit or semi-implicit methods. Our method makes use of high order in time integration techniques, inexact Newton–Krylov nonlinear solvers, and multigrid preconditioners. We target problems discretized on meshes having millions of spatial zones.

The first work on implicit solution of radiation diffusion problems was done by Axelrod et al. in 1984 when they presented a solution method using a high order GEAR ODE solver [1] for integration in time coupled with a Newton method and an approximate direct solver [2]. Their work showed a distinct advantage in both accuracy and run time of the implicit solver over an operator split approach on multigroup diffusion in one dimension. This work did not, however, follow with development of solvers for large-scale problems. In particular, the very effective Newton–Krylov methods [3–5] were not applied to implicit radiation diffusion problems until much later. One obstacle to use of these methods was the requirement of effective preconditioners for the linear Jacobian systems. In 1999, Rider and co-workers [6,7] applied these methods along with multigrid preconditioning to one- and two-dimensional equilibrium problems and showed substantial benefit in accuracy of a second order in time fully implicit solver over a first order semi-implicit solver where nonlinearities are lagged between time steps.

For semi-implicit methods, time steps are often chosen to be small enough to maintain accuracy despite explicit (or lagged) parts of the computation. When moving to an implicit method, however, the question of how to choose the time step should be revisited since an implicit method generally allows a larger step size for a given accuracy within a computation. Rider and Knoll [8] suggested a method of choosing steps for the fully implicit formulation by using a hyperbolic model of the system. Brown et al. [9] simultaneously pursued implicit methods based on the work of Axelrod et al. and employed ODE time integrator technology for three-dimensional multigroup diffusion. These methods choose time steps based on solution accuracy requirements of the user and local time truncation error estimates [10].

Mousseau et al. extended the work of Rider et al. to problems where the radiation and matter fields are not in equilibrium. They again applied Newton–Krylov methods with first and second order time stepping on two-dimensional problems [11,12]. They used an operator-splitting method and multigrid for preconditioning. Brown and Woodward [13], also looking at non-equilibrium problems, developed an effective Schur complement-based preconditioner and demonstrated parallel scalability of a fully implicit solver based on high order time integration, Newton–Krylov methods, and semicoarsening multigrid techniques on three-dimensional problems with hundreds of millions of unknowns. They found that preconditioners for the implicit system need to account for the coupling between matter and radiation effectively and also showed that the choice of preconditioner is crucial to the success of the fully implicit solve for large-scale problems.

More recently, implicit formulations of radiation diffusion have been combined with hydrodynamics problems in a coupled manner. Bates et al. [14] have developed a nonlinearly consistent solver for the coupled system. Howell and Greenough [15] have also extended implicit diffusion methods to include coupled problems with hydrodynamics within the context of adaptive mesh refinement. Recent work in the area has also included investigations into the relative performance of Newton–Krylov methods with nonlinear multigrid techniques [16]. These studies have shown that Newton–Krylov is in general faster for these problems, but that nonlinear multigrid can have advantages in early time steps [17–19]. Additionally, Lowrie [20] has compared several implicit time integration approaches for nonlinear relaxation and diffusion problems and Ropp et al. [21] have studied the accuracy of time integration methods for reaction-diffusion equations, including operator-splitting approaches.

With the exception of the original work by Axelrod et al. and the work of Brown and Woodward, the above work does not consider high order time integration and tabulated opacities. High order methods can result in significant benefits in run time reduction while maintaining high accuracy. Care must be taken, however, in how these methods are applied as poor tolerance selection can lead to instabilities. In addition, when using Newton–Krylov methods, a nonlinear residual evaluation is required at each linear iteration.

These evaluations require re-computation of opacity values from outside tables. Often packages developed to use these tables will perform interpolation or spline fits leading to possible significant computational expense for each opacity evaluation. (Note that traditional methods just time-lag these values.) Lastly, none of the above work examines performance of implicit solvers for problems with highly nonlinear fusion source terms. These sources result in solutions which exhibit fast changes. Good solvers will need to capture these changes accurately which can most efficiently be done with adaptive time stepping technology.

In this paper, we present a fully implicit solution method based on ODE time integration technology targeting large-scale radiation diffusion problems. We examine its advantages and disadvantages as compared to that of a semi-implicit method for three-dimensional problems with tabulated opacities from the LEOS equation of state library [22]. We also consider nonlinearities introduced from fusion source terms in the material energy equation. These terms are highly nonlinear and give rise to a potentially difficult nonlinear problem within the implicit formulation. Our results indicate that a fully implicit solution approach can achieve more accurate solutions than semi-implicit solution methods in many simulations involving the interaction of radiation and matter with highly nonlinear source terms. Furthermore, the fully implicit approach can be as cost effective as semi-implicit approaches in many cases despite the use of tabulated values for the opacities. Lastly, the solution approach is shown to scale well to very large problems solved on parallel machines.

Numerical results in this paper compare our fully implicit method with a semi-implicit method that represents the state-of-the-art in many simulation areas. This semi-implicit technique (detailed below) time-lags the opacity terms which have been commonly thought to be too expensive to evaluate with higher frequency. In addition, the term coupling radiation and matter is linearized. The advantages of the scheme are that a single linear system must be solved to high accuracy for each time step, that system is symmetric and positive definite, and many current solvers are particularly efficient on symmetric systems (such as conjugate gradients or multigrid). The main disadvantage is the need to take very small steps in order to maintain accuracy after the time-lagging and linearizations have been done. As stated above, our fully implicit method can take larger time steps and can easily use high order time integration methods. In addition, general purpose packages for solution of implicit systems can easily be used, and a semi-implicit implementation can be taken advantage of for use in preconditioning an implicit method thereby making use of an existing semi-implicit code. The main disadvantage of the implicit method is the need to re-evaluate all nonlinearities at every iteration. The results in this paper indicate, however, that the implicit method is still faster than the semi-implicit method even with these extra nonlinearity evaluations.

In the next section of this paper, we outline the model problem we are considering. The section following overviews both our fully implicit method and a semi-implicit solution method to which we compare, as well as the nonlinear and linear solvers we use. In the results section, we give comparisons of the two methods, examinations of some algorithmic elements of our solution strategy, and some demonstrations of our method's performance on a large, parallel machine. We conclude with some remarks about the viability of fully implicit methods on radiation diffusion problems.

2. Flux-limited radiation diffusion model

For this work, we consider the flux-limited, two-temperature formulation of radiation diffusion given by [23,24]

$$\frac{\partial E_{\mathrm{R}}}{\partial t} = \nabla \cdot \left(\frac{c}{3\rho\kappa_{\mathrm{R}}(T_{\mathrm{R}}) + \frac{\|\nabla E_{\mathrm{R}}\|}{E_{\mathrm{R}}}} \nabla E_{\mathrm{R}} \right) + c\rho\kappa_{\mathrm{P}}(T_{\mathrm{M}}) \cdot \left(aT_{\mathrm{M}}^4 - E_{\mathrm{R}} \right) + \chi(\mathbf{x})caT_{\mathrm{source}}^4, \tag{1}$$

where $E_{\mathrm{R}}(\mathbf{x},t)$ is the radiation energy density ($\mathbf{x} = (x,y,z)$), $T_{\mathrm{M}}(\mathbf{x},t)$ is the material temperature, $\rho(\mathbf{x})$ is the material density, c is the speed of light, and $a = 4\sigma/c$, where σ is the Stephan–Boltzmann constant. The

Rosseland opacity, κ_R, is a nonlinear function of the radiation temperature, T_R, which is defined by the relation $E_R = aT_R^4$. The Planck opacity, κ_P, is a nonlinear function of material temperature, T_M, which is related to the material energy through an equation of state, $E_M = \text{EOS}(T_M)$. Here, T_{source} is a given source temperature, and $\chi(\mathbf{x})$ is a function of the spatial variable $\mathbf{x}$. In the limiter, the norm $\|\cdot\|$ is taken to be the l^2 norm of the gradient vector.

This equation is coupled to an equation expressing conservation of material energy given by

$$\frac{\partial E_M}{\partial t} = -c\rho\kappa_P(T_M) \cdot (aT_M^4 - E_R) + \mu(\mathbf{x},t)T_M^5, \tag{2}$$

where μT_M^5 is a fusion source term with $\mu(\mathbf{x},t)$ a function of both space and time.

This system is highly nonlinear due to the opacity dependencies on temperatures as well as the fusion source term. Opacities typically depend on temperatures as a power law with typical expressions like $\kappa = CT^{-p}$, where C is a constant and p may be 3–5 depending on the material and physical regime [25,26]. We consider Dirichlet, Neumann, and Robin boundary conditions for the system (1) and (2), and our focus here is on the development of solution methods for this system.

3. Solution methods

For both the fully implicit and semi-implicit formulations, we employ a cell-centered finite difference approach for the spatial discretization. We use a tensor product grid with N_x, N_y, and N_z cells in the x, y, and z directions, respectively. Defining $E_{R,i,j,k}(t) \approx E_R(\mathbf{x}_{i,j,k},t)$ and $E_{M,i,j,k}(t) \approx E_M(\mathbf{x}_{i,j,k},t)$, with $\mathbf{x}_{i,j,k} = (x_i,y_j,z_k)$, and

$$\mathbf{E}_R \equiv \begin{pmatrix} E_{R,1,1,1} \\ \vdots \\ E_{R,N_x,N_y,N_z} \end{pmatrix} \quad \text{and} \quad \mathbf{E}_M \equiv \begin{pmatrix} E_{M,1,1,1} \\ \vdots \\ E_{M,N_x,N_y,N_z} \end{pmatrix},$$

we can write our discrete equations in terms of a discrete diffusion operator given by

$$\mathbf{L}(\mathbf{E}_R) \equiv \left(L_{1,1,1}(\mathbf{E}_R), \ldots, L_{N_x,N_y,N_z}(\mathbf{E}_R)\right)^T, \tag{3}$$

a local coupling operator given by

$$\mathbf{S}(\mathbf{E}_R,\mathbf{E}_M) \equiv \left(S_{1,1,1}(\mathbf{E}_R,\mathbf{E}_M), \ldots, S_{N_x,N_y,N_z}(\mathbf{E}_R,\mathbf{E}_M)\right)^T, \tag{4}$$

and a material source term

$$\mathbf{R}(\mathbf{E}_M) \equiv \left(R_{1,1,1}(\mathbf{E}_M), \ldots, R_{N_x,N_y,N_z}(\mathbf{E}_M)\right)^T, \tag{5}$$

where

$$\begin{aligned}
L_{i,j,k}(\mathbf{E}_R) \equiv &\left(D_{i+1/2,j,k}\frac{E_{R,i+1,j,k} - E_{R,i,j,k}}{\Delta x_{i+1/2,j,k}} - D_{i-1/2,j,k}\frac{E_{R,i,j,k} - E_{R,i-1,j,k}}{\Delta x_{i-1/2,j,k}}\right)\bigg/\Delta x_i \\
&+ \left(D_{i,j+1/2,k}\frac{E_{R,i,j+1,k} - E_{R,i,j,k}}{\Delta y_{i,j+1/2,k}} - D_{i,j-1/2,k}\frac{E_{R,i,j,k} - E_{R,i,j-1,k}}{\Delta y_{i,j-1/2,k}}\right)\bigg/\Delta y_j \\
&+ \left(D_{i,j,k+1/2}\frac{E_{R,i,j,k+1} - E_{R,i,j,k}}{\Delta z_{i,j,k+1/2}} - D_{i,j,k-1/2}\frac{E_{R,i,j,k} - E_{R,i,j,k-1}}{\Delta z_{i,j,k-1/2}}\right)\bigg/\Delta z_k
\end{aligned} \tag{6}$$

with

$$D_{i+1/2,j,k} \equiv \frac{c}{3\rho_{i+1/2,j,k}\kappa_{R,i+1/2,j,k} + \|\nabla E_R\|_{i+1/2,j,k}/E_{R,i+1/2,j,k}},$$

$$D_{i-1/2,j,k} \equiv \frac{c}{3\rho_{i-1/2,j,k}\kappa_{R,i-1/2,j,k} + \|\nabla E_R\|_{i-1/2,j,k}/E_{R,i-1/2,j,k}},$$

$$D_{i,j+1/2,k} \equiv \frac{c}{3\rho_{i,j+1/2,k}\kappa_{R,i,j+1/2,k} + \|\nabla E_R\|_{i,j+1/2,k}/E_{R,i,j+1/2,k}},$$

$$D_{i,j-1/2,k} \equiv \frac{c}{3\rho_{i,j-1/2,k}\kappa_{R,i,j-1/2,k} + \|\nabla E_R\|_{i,j-1/2,k}/E_{R,i,j-1/2,k}},$$

$$D_{i,j,k+1/2} \equiv \frac{c}{3\rho_{i,j,k+1/2}\kappa_{R,i,j,k+1/2} + \|\nabla E_R\|_{i,j,k+1/2}/E_{R,i,j,k+1/2}},$$

$$D_{i,j,k-1/2} \equiv \frac{c}{3\rho_{i,j,k-1/2}\kappa_{R,i,j,k-1/2} + \|\nabla E_R\|_{i,j,k-1/2}/E_{R,i,j,k-1/2}},$$

and

$$S_{i,j,k}(E_{R,i,j,k},E_{M,i,j,k}) = c\rho_{i,j,k}\kappa_{P,i,j,k}\left(aT_{M,i,j,k}^4 - E_{R,i,j,k}\right), \tag{7}$$

and

$$R_{i,j,k}(E_{M,i,j,k}) = \mu_{i,j,k}T_{M,i,j,k}^5. \tag{8}$$

Thus, our discrete scheme is to find $\mathbf{E}_R(t)$ and $\mathbf{E}_M(t)$ such that,

$$\frac{d\mathbf{E}_R}{dt} = \mathbf{L}(\mathbf{E}_R) + \mathbf{S}(\mathbf{E}_R,\mathbf{E}_M) + \mathbf{Q}, \tag{9}$$

$$\frac{d\mathbf{E}_M}{dt} = -\mathbf{S}(\mathbf{E}_R,\mathbf{E}_M) + \mathbf{R}(\mathbf{E}_M), \tag{10}$$

where $\mathbf{Q}$ includes the source term along with terms from the discretized boundary conditions. For more details, see [13].

3.1. Fully implicit

For the fully implicit formulation, we use an ODE time integrator to handle the implicit time step selection for the system (9) and (10). In particular, we employ the parallel ODE solver, CVODE [10], developed at Lawrence Livermore National Laboratory and based on the VODPK package [27]. CVODE employs the fixed leading coefficient variant of the backward differentiation formula (BDF) method [28,29] and allows for variation in the order of the time discretization as well as in the time step size.

The methods in CVODE are predictor–corrector in nature, and so each time step begins with the calculation of an explicit predictor. An implicit corrector is then employed to solve for the time step solution. This time integration technique leads to a coupled, nonlinear system of equations that must be solved at each time step. For example, solving the ODE system

$$\dot{y} = f(t,y), \tag{11}$$

with the backward Euler method (i.e., the BDF method of order 1), leads to the following nonlinear system:

$$0 = F(y) \equiv y - \Delta t f(t_n,y) - y_{n-1} \left(\text{i.e., } \frac{y_n - y_{n-1}}{\Delta t} = f(t_n,y_n)\right) \tag{12}$$

that must be solved for $y = y_n$ at each time step. For the solution of this system, CVODE uses an inexact Newton–Krylov method with Jacobian-vector products approximated by finite differences of the form

$$F'(y)v \approx \frac{F(y + \theta v) - F(y)}{\theta},$$ (13)

where θ is a scalar. Within the Newton–Krylov paradigm, only the implementation of the nonlinear function is necessary, and Jacobian matrix entries need never be formed or stored. The explicit predictor, $y_{n(0)}$, is used as an initial guess to the nonlinear system (12). The details of how tolerances are selected for the nonlinear and linear iterations in CVODE are discussed in detail in the overview paper [30]. Finally, when the initial nonlinear residual, $F(y_{n(0)})$, is small enough (i.e., meets the nonlinear system stopping tolerance), CVODE sets the linear solver answer to $s_0 = -F(y_n(0))$, which then gives $y_{n(1)} = y_n(0) + s_0 = \Delta t f(t_n, y_{n(0)}) + y_{n-1}$. This amounts to one step of functional iteration for solving (12), and so the number of linear iterations can be 0 for some Newton iterations.

In the methods discussed above, we use the scaling technique incorporated into CVODE. Thus, we include an absolute tolerance (ATOL) for each unknown and a relative tolerance (RTOL) applied to all unknowns. These tolerances are then used to form a weight that is applied to each solution component during the time step from t_{n-1} to t_n. This weight is given as

$$w_i = \text{RTOL} \cdot |y_{n-1}^i| + \text{ATOL}_i,$$ (14)

and then the weighted root mean square norm

$$\|y\|_{\text{WRMS}} = \left[N^{-1} \sum_{i=1}^{N} (y_i/w_i)^2 \right]^{1/2}$$ (15)

is applied on all error-like vectors within the solution process. This scaling gives each solution component equal weight when measuring the size of errors in y. For our application, we supply two absolute tolerances, one to be used with the radiation energy unknowns and one to be used with the material energy unknowns.

3.1.1. Time step selection

Time step sizes are chosen in an attempt to maximize step sizes while controlling the local truncation error, and thus give a solution that obeys a user-specified accuracy bound. For a typical linear multistep method (under mild assumptions on the step sizes), the local truncation error (LTE), at order q and step size Δt, satisfies an asymptotic relation of the form

$$\text{LTE} = C(\Delta t)^{q+1} y^{(q+1)} + \text{O}((\Delta t)^{q+2}),$$

where $y^{(q+1)}$ is the $q+1$st derivative of y, and C is a constant. A similar relation holds for the predictor $y_{n(0)}$. These are combined to get the following order $q + 2$ estimate used in CVODE, namely

$$\text{LTE}(\Delta t^n) \equiv C_q(y_n - y_{n(0)}),$$ (16)

where y_n is the final iterate in the Newton iteration and C_q is a constant that depends on the BDF method order q but is independent of the solution. If $\|\text{LTE}(\Delta t^n)\|_{\text{WRMS}} < 1$, then the time step is accepted. If this condition is violated, the step size is cut, and the solution is recomputed. New steps are chosen by estimating the local truncation error at the new step, $\Delta t'$, as

$$\|\text{LTE}(\Delta t')\|_{\text{WRMS}} \approx \left(\frac{\Delta t'}{\Delta t^n} \right)^{q+1} \|\text{LTE}(\Delta t)\|_{\text{WRMS}},$$ (17)

where q is the current method order. The new step is chosen to give the largest time step still satisfying $\|\text{LTE}(\Delta t')\|_{\text{WRMS}} < 1$. CVODE also changes the BDF method order by comparing the local truncation errors for the BDF methods of order $q - 1$ and $q + 1$ when using order q, and then taking the order that allows the largest time step.

We use the GMRES Krylov iterative solver for solution of the linear Jacobian system at each Newton iteration [31]. The tolerance for the Newton iteration is taken to guarantee that iteration error introduced from the nonlinear solver is smaller than the local truncation error. The default linear system tolerance in CVODE is taken to be the factor $\alpha = 0.05$ times the nonlinear system tolerance. This factor can be optionally set in the CVODE solver, and for some of the problems discussed below we use a smaller value of α, as the default of 0.05 did not work for the larger RTOL values. The default maximum subspace dimension for GMRES in CVODE is 5, and we use this default in all of our tests. For more details regarding the step size and order selection strategies in CVODE, as well as acceptance of a step and nonlinear convergence, we refer the reader to the review article [30].

3.1.2. Preconditioning

Preconditioning is generally essential when using Krylov linear solvers. To describe our preconditioning strategy, we begin by considering the content and structure of the Jacobian matrix. In (11), set $y = (\mathbf{E}_R^T, \mathbf{E}_M^T)^T$, and then form f using the right-hand sides of (9) and (10). The Jacobian matrices used in the Newton method are of the general form $F'(y) = (I - \gamma J)$, where $J = \partial f/\partial y$ is the Jacobian of the nonlinear function f, and the parameter $\gamma \equiv \Delta t \beta$ with Δt the current time step value and β a coefficient depending on the order of the BDF method. Recalling the definitions of the discrete divergence and source operators, the block form of the Jacobian of f is

$$J = \begin{pmatrix} \partial\mathbf{L}/\partial\mathbf{E}_R + \partial\mathbf{S}/\partial\mathbf{E}_R & \partial\mathbf{S}/\partial\mathbf{E}_M \\ -\partial\mathbf{S}/\partial\mathbf{E}_R & -\partial\mathbf{S}/\partial\mathbf{E}_M + \partial\mathbf{R}/\partial\mathbf{E}_M \end{pmatrix} = \begin{pmatrix} A + G & B \\ -G & -B + C \end{pmatrix},$$

where $A = \partial\mathbf{L}/\partial\mathbf{E}_R$, $G = \partial\mathbf{S}/\partial\mathbf{E}_R$, $B = \partial\mathbf{S}/\partial\mathbf{E}_M$, and $C = \partial\mathbf{R}/\partial\mathbf{E}_M$. We note that G, B and C are diagonal matrices.

On close inspection of the nonlinear diffusion operator $\mathbf{L}(\mathbf{E}_R)$, we can write

$$\mathbf{L}(\mathbf{E}_R) = \hat{\mathbf{L}}(\mathbf{E}_R)\mathbf{E}_R, \tag{18}$$

where $\hat{\mathbf{L}}$ is a nonlinear matrix-valued function of $\mathbf{E}_R$. In all of our preconditioning strategies, we neglect the nonlinearity in the diffusion term and use the approximation

$$A = \partial\mathbf{L}(\hat{\mathbf{E}}_R)/\partial\mathbf{E}_R \approx \hat{\mathbf{L}}(\hat{\mathbf{E}}_R) \equiv \tilde{A},$$

where $\partial\mathbf{L}(\hat{\mathbf{E}}_R)/\partial\mathbf{E}_R$ is the Jacobian of $\mathbf{L}$ evaluated at a radiation energy, $\hat{\mathbf{E}}_R$. The size of the neglected term is related to the derivatives of the Rosseland opacity and the flux-limiter. Our motivation for neglecting this term arises from the fact that $-\tilde{A}$ is symmetric and positive definite, whereas $-A$ is not. Making this approximation leads to a symmetric preconditioning matrix. Symmetry matters within the preconditioner since we will be applying multigrid methods which work most efficiently on symmetric and positive definite systems. In addition, symmetric preconditioning matrices require half the storage of non-symmetric systems. Lastly, we note that the derivative of the flux-limiter may lead to numerical errors if $\nabla\mathbf{E}_R$ approaches 0.

Our preconditioning strategy is to factor the matrix, $I - \gamma J$, as

$$\begin{pmatrix} P & Q \\ U & T \end{pmatrix} \equiv \begin{pmatrix} I - \gamma(\tilde{A} + G) & -\gamma B \\ \gamma G & I - \gamma(C - B) \end{pmatrix} = M$$

into the following:

$$M_{\text{Schur}} = \begin{pmatrix} I & QT^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} P - QT^{-1}U & 0 \\ 0 & T \end{pmatrix} \begin{pmatrix} I & 0 \\ T^{-1}U & I \end{pmatrix}.$$

Letting $S = P - QT^{-1}U$, we write the solution to $M_{\text{Schur}}x = b$ as

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} S^{-1}(b_1 - QT^{-1}b_2) \\ T^{-1}(-Ux_1 + b_2) \end{pmatrix}.$$

If the Schur complement, S, is exactly inverted, there will be no error associated with this preconditioner for the non-flux-limited, constant opacity case. In addition, because B, C and hence T are diagonal, there is no penalty associated with inverting T for every iteration of a method that inverts S, as there would be if a material energy diffusion term were added to the equations. Also note that S is formed by modifying the diagonal of P and thus is composed of a symmetric diffusion-like matrix with a modified diagonal. Hence, we can employ multigrid methods to invert this Schur complement.

The Rosseland opacity will exhibit large changes where material interfaces exist in the domain. The temperature dependence gives rise to large value changes as well. These changes imply that the problem can be very heterogeneous. As a result, to invert the Schur complement matrix (S), we use a multigrid method designed to handle large changes in problem coefficients. In particular, we use one V-cycle of a semi-coarsening multigrid algorithm, such as the ParFlow semi-coarsening multigrid (PFMG) [32] or the semi-coarsening multigrid (SMG) developed by Schaffer [33,34] as our multigrid solver. Semi-coarsening multigrid methods have been found to be quite effective on highly heterogeneous problems. A comparison of PFMG and SMG can be found in [35]. Details of both these methods can be found in the cited references, and more information about multigrid methods in general can be found in [36].

Since Jacobian approximations can be expensive to compute, in CVODE the preconditioner is not updated with every Newton iteration. Preconditioner updates occur only when the Newton iteration fails to converge, 20 time steps pass without an update, or when there is a significant change in the time step size and order of the ODE method.

In summary, the main advantage of the fully implicit method is that we have accurate error control in the time step selection process allowing step sizes to automatically adjust to the problem physics while maintaining accuracy. The main disadvantage of the method is that opacities must be calculated for every linear iteration, as a nonlinear function evaluation is required in the matrix–vector product approximation (13). In general, fully implicit methods require more sophisticated solvers than semi-implicit methods. The solution method presented above has been tested on very large, three-dimensional problems and has been shown to be parallel scalable [13].

3.2. Semi-implicit

In the semi-implicit method we compare against, a backward Euler time stepping technique is applied, wherein opacities, flux-limiters, and material sources are evaluated at the start of a new time step using the solution from the previous step, and the term coupling radiation to matter, $c\rho\kappa_P(\mathbf{T}_M^n)(a(\mathbf{T}_M^{n+1})^4 - \mathbf{E}_R^{n+1})$, is linearized about the solution from the previous step. The problem is put in a residual formulation so that the single linear solve required at each time step gives the increment to the solution values from the previous step's solution.

3.2.1. Linearization
Beginning with the discrete system (9) and (10) and using (18), we can write

$$\frac{\mathbf{E}_R^{n+1} - \mathbf{E}_R^n}{\Delta t} = \hat{\mathbf{L}}(\mathbf{E}_R^n)\mathbf{E}_R^{n+1} + \mathbf{K}(\mathbf{T}_M^n)(a(\mathbf{T}_M^{n+1})^4 - \mathbf{E}_R^{n+1}) + \mathbf{Q}^{n+1}, \tag{19}$$

$$\frac{\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n}}{\Delta t} = -\mathbf{K}(\mathbf{T}_{\mathrm{M}}^{n})(a(\mathbf{T}_{\mathrm{M}}^{n+1})^{4} - \mathbf{E}_{\mathrm{R}}^{n+1}) + \mathbf{R}(\mathbf{T}_{\mathrm{M}}^{n+1}),$$ (20)

where $\mathbf{K}(\mathbf{T}_{\mathrm{M}}^{n})$ is a diagonal matrix with entries given by $K_{i,j,k} \equiv c\rho\kappa_{\mathrm{P}}(T_{\mathrm{M},i,j,k}^{n})$ and $E_{\mathrm{M},i,j,k}^{n+1} = \mathrm{EOS}(T_{\mathrm{M},i,j,k}^{n+1})$. Next, letting $\mathbf{T}_{\mathrm{M}}^{n+1} = \mathbf{T}_{\mathrm{M}}^{n} + \Delta\mathbf{T}_{\mathrm{M}}^{n}$ we linearize to obtain

$$(\mathbf{T}_{\mathrm{M}}^{n+1})^{4} = (\mathbf{T}_{\mathrm{M}}^{n} + \Delta\mathbf{T}_{\mathrm{M}}^{n})^{4} \approx (\mathbf{T}_{\mathrm{M}}^{n})^{4} + 4(\mathbf{T}_{\mathrm{M}}^{n})^{3}\Delta\mathbf{T}_{\mathrm{M}}^{n}.$$

Similarly, we linearize $E_{\mathrm{M}} = \mathrm{EOS}(T_{\mathrm{M}})$ to obtain

$$\mathbf{E}_{\mathrm{M}}^{n+1} = \mathbf{EOS}(\mathbf{T}_{\mathrm{M}}^{n} + \Delta\mathbf{T}_{\mathrm{M}}^{n}) \approx \mathbf{EOS}(\mathbf{T}_{\mathrm{M}}^{n}) + \frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\Delta\mathbf{T}_{\mathrm{M}}^{n},$$

or

$$\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n} \approx \frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\Delta\mathbf{T}_{\mathrm{M}}^{n}.$$

Thus,

$$(\mathbf{T}_{\mathrm{M}}^{n+1})^{4} \approx (\mathbf{T}_{\mathrm{M}}^{n})^{4} + 4(\mathbf{T}_{\mathrm{M}}^{n})^{3}\left[\frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\right]^{-1}(\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n}).$$

We apply a similar linearization to the fusion source term, $\mu(T^{n+1})^{5}$. Substituting this last relationship into (19) and (20), we have

$$\mathbf{E}_{\mathrm{R}}^{n+1} - \mathbf{E}_{\mathrm{R}}^{n} = \Delta t\hat{\mathbf{L}}(\mathbf{E}_{\mathrm{R}}^{n})\mathbf{E}_{\mathrm{R}}^{n+1} + \Delta t\mathbf{Q}^{n+1} + \Delta t\mathbf{K}(\mathbf{T}_{\mathrm{M}}^{n})$$
$$\cdot\left(a\left[(\mathbf{T}_{\mathrm{M}}^{n})^{4} + 4(\mathbf{T}_{\mathrm{M}}^{n})^{3}\left[\frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\right]^{-1}(\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n})\right] - \mathbf{E}_{\mathrm{R}}^{n+1}\right),$$ (21)

and

$$\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n} = \Delta t\mu\left[(\mathbf{T}_{\mathrm{M}}^{n})^{5} + 5(\mathbf{T}_{\mathrm{M}}^{n})^{4} \cdot \left[\frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\right]^{-1}(\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n})\right] - \Delta t\mathbf{K}(\mathbf{T}_{\mathrm{M}}^{n})$$
$$\cdot\left(a\left[(\mathbf{T}_{\mathrm{M}}^{n})^{4} + 4(\mathbf{T}_{\mathrm{M}}^{n})^{3}\left[\frac{\partial\mathbf{EOS}}{\partial\mathbf{T}_{\mathrm{M}}}(\mathbf{T}_{\mathrm{M}}^{n})\right]^{-1}(\mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n})\right] - \mathbf{E}_{\mathrm{R}}^{n+1}\right),$$ (22)

where we solve for the changes, $\Delta\mathbf{E}_{\mathrm{R}}^{n} \equiv \mathbf{E}_{\mathrm{R}}^{n+1} - \mathbf{E}_{\mathrm{R}}^{n}$ and $\Delta\mathbf{E}_{\mathrm{M}}^{n} \equiv \mathbf{E}_{\mathrm{M}}^{n+1} - \mathbf{E}_{\mathrm{M}}^{n}$, given the previous values of $\mathbf{E}_{\mathrm{R}}^{n}$ and $\mathbf{E}_{\mathrm{M}}^{n}$.

We solve the linear system (21) and (22) using the same linear solver as described above: the GMRES Krylov iterative solver with a Schur complement factorization preconditioner. The same multigrid method is used to invert the Schur complement matrix as in the fully implicit case. The linear iteration is performed until the relative residual is bounded by an input tolerance times the norm of the right-hand side

$$\|\mathbf{r}\|_{\mathrm{WRMS}} \leqslant \epsilon\|\mathbf{b}\|_{\mathrm{WRMS}},$$ (23)

where $\mathbf{r}$ is the linear system residual, $\mathbf{b}$ is the linear system right-hand side, and ϵ is an input parameter. The WRMS norm is calculated in the same way as that for CVODE, using RTOL and ATOL values chosen as in the CVODE case.

We note here that due to the lagging of opacity values and flux limiters, the semi-implicit method is only first order in time. Without changing how the lagging is handled (and thus adding new opacity evaluation requirements), this method cannot be changed to higher order.

3.2.2. Time step selection

Time steps are chosen to try to restrict changes in radiation energy and material temperature within a step. For specified minimum values, E_{min} and T_{min}, and specified fractional variations allowed in a step, E_{frac} and T_{frac}, the new step is computed by first calculating a maximum variation for each variable

$$v_R = \max_{i,j,k} \left(\frac{\Delta E_R^n}{0.5(E_R^{n-1} + E_R^n) + E_{min}} \right),$$

and

$$v_M = \max_{i,j,k} \left(\frac{\Delta T_M^n}{0.5(T_M^{n-1} + T_M^n) + T_{min}} \right).$$

Then, the new step is chosen as

$$\Delta t^{new} = \Delta t^{old} \cdot \min(E_{frac}/(v_R + \delta), T_{frac}/(v_M + \delta)), \tag{24}$$

where $\delta = 10^{-7}$ limits the maximum change in the step size. If the linear iteration fails to converge, that is, the convergence criterion, (23), is not satisfied in the allotted maximum number of iterations, then the step is repeated with $\Delta t^{new} = 0.5 \cdot \Delta t^{old}$.

3.2.3. Comparison with fully implicit

Note that this selection process is similar to the error control for the fully implicit case. However, while the semi-implicit approach bounds the maximum change in solution components over a time step, the fully implicit approach is bounding the maximum local truncation error made on a step with no direct control on the solution components. To see the similarity, consider the radiation energy case. The semi-implicit method selects the step so that

$$v_R/E_{frac} = \frac{\Delta E_R^n}{0.5(E_R^{n-1} + E_R^n)E_{frac} + E_{min}E_{frac}} < 1. \tag{25}$$

Taking $ATOL_R = E_{min}E_{frac}$, $RTOL = E_{frac}$, and noting that $0.5(E_R^{n-1} + E_R^n)$ is an approximation to the current radiation energy, we see that the ith component of the variation that is being bounded in the semi-implicit case is just,

$$\frac{\Delta E_R^n}{RTOL \cdot E_R^n + ATOL}. \tag{26}$$

The variation in local truncation error that is bounded in the fully implicit case is just this expression. But instead of looking at the variation over a time step in ΔE_R^n, we look at the variation between the predictor and the corrector. Since the error in each of these approximations can be bounded, the variation in the predictor and corrector gives us a concrete estimate on the local truncation error. In [37,38], the authors describe and analyze time step selection strategies for ODEs that control the relative change in the numerical solution, similar to that described above for our semi-implicit method. As remarked in [37], the great applicability of controlling the relative change is also its greatest weakness. That is, it treats all methods like a first order method. Thus, it can be very inefficient if used to control the time step size for higher order methods.

4. Numerical results

In this section, we present numerical results of solving nonlinear diffusion problems with the high order fully implicit method. We compare the method with a semi-implicit scheme for both accuracy and computational speed. We also investigate some of the advantages of the high order integration method and look at

which orders give the highest benefit. Lastly, we show results of solving very large-scale nonlinear diffusion problems in parallel.

In the following subsections, the numerical statistical counters and parameters are

- RTOL the relative tolerance,
- MO the maximum order allowed, implicit method only,
- NST the number of time steps,
- NNI the number of nonlinear iterations,
- NLI the number of linear iterations,
- RT the run time in seconds,
- FAC the specified fractional variation allowed in both energies within a step, semi-implicit only.

In the following examples we will use step and bi-cubic functions to define the spatial and temporal extent of the source functions $\chi(\mathbf{x})$ and $\mu(\mathbf{x},t)$ in (1) and (2), respectively, given by

$$H(x,\epsilon) \equiv \begin{cases} 1 & \text{if } |x| \leqslant \epsilon, \\ 0 & \text{otherwise.} \end{cases} \tag{27}$$

$$B(x,\epsilon) \equiv \begin{cases} 2\left(\frac{\epsilon+x}{2}\right)^2\left(6 - 8\frac{\epsilon+x}{2\epsilon}\right) & \text{if } -\epsilon < x \leqslant 0, \\ 2\left(\frac{\epsilon-x}{2}\right)^2\left(6 - 8\frac{\epsilon-x}{2\epsilon}\right) & \text{if } 0 \leqslant x < \epsilon \\ 0 & \text{otherwise.} \end{cases} \tag{28}$$

The simulations in this paper use the LEOS equation-of-state data base [22]. This data base allows for the extraction of total energy and Planck and Rosseland opacities using various interpolation methods. We use bi-cubic interpolation for the material energy $EOS(T_M)$ and bi-linear for the Planck and Rosseland opacities. Although this data base provides methods of returning out-of-bound values, we have designed the test problems presented here so that there were no out-of-bounds data requests.

Due to its robustness, the SMG algorithm was used for the multigrid portion of the preconditioner for all runs in Sections 4.1, 4.2 and 4.3. Due to its superior scalability features, we used the PFMG method for multigrid solves, unless otherwise specified, for the studies done in Section 4.4.

4.1. Demonstration of accuracy

In this section, we present a test case which demonstrates accuracy of the fully implicit and semi-implicit codes compared to analytic solutions. All runs in this and the next subsection were done on a single processor in a Compaq cluster of 1 GHz EV68 Alpha processors.

Our first numerical test case is the Su–Olson problem, a one-dimensional Marshak problem that has a published analytic solution [39]. The problem starts with a homogeneous initial condition for the radiation and material temperature. A Robin boundary condition of the form

$$E(\mathbf{0},t) - \left(\frac{2}{3\kappa_R}\right)\frac{\partial E(\mathbf{0},t)}{\partial \mathbf{x}} = 1 \tag{29}$$

is applied at $\mathbf{x} = 0$, and a homogeneous Dirichlet condition is applied at $\mathbf{x} = \infty$. In practice, this right-hand boundary condition is applied at $\mathbf{x} = 20$.

The material specific heat is given by $c_v = \frac{a}{\epsilon}T_M^3$ and the equation of state is $E_M(T_M) = \rho c_v T_M$. The flux limiter is not used in this simulation. The Planck and Rosseland opacities are both set to a constant, $\kappa_P = \kappa_R = 1.0$ cm²/g, and $\epsilon = 0.1$. Heat is applied to the left-hand boundary as a result of the above

Table 1
Statistics for both fully implicit and semi-implicit solutions of the Su–Olson problem

Method	RTOL	FAC	NST	NNI	NLI	Max. err.	RT
200 Grid points							
Implicit	10^{-5}	NA	1609	1654	1604	4.43×10^{-2}	7.49
Implicit	10^{-6}	NA	2341	2405	2338	4.43×10^{-2}	9.54
Semi-imp.	NA	10^{-2}	3919	NA	3919	4.45×10^{-2}	15.5
Semi-imp.	NA	10^{-3}	39,053	NA	39,053	4.43×10^{-2}	72.9
1000 Grid points							
Implicit	10^{-5}	NA	2561	2619	2557	9.69×10^{-3}	84.6
Implicit	10^{-6}	NA	3759	3832	3755	9.69×10^{-3}	83.1
Semi-imp.	NA	10^{-2}	5573	NA	5573	9.91×10^{-3}	86.2
Semi-imp.	NA	10^{-3}	55,782	NA	55,782	9.64×10^{-3}	2281.8
10,000 Grid points							
Implicit	10^{-5}	NA	3788	3853	3781	1.05×10^{-3}	809.1
Implicit	10^{-6}	NA	5600	5685	5595	1.05×10^{-3}	1332.1
Semi-imp.	NA	10^{-2}	7621	NA	7621	1.34×10^{-3}	1190.7
Semi-imp.	NA	10^{-3}	76,540	NA	76,540	1.04×10^{-3}	17935.2

Maximum relative error given is for radiation temperature.

boundary condition. As the temperature of the radiation field increases, energy is transfered to the material. Simulations were run to a time of 3.34×10^{-5} μs. At this time the wave front is still far enough away from the right boundary that the boundary condition does not affect the solution.

Table 1 gives a comparison of implicit and semi-implicit method statistics for this problem. The maximum order, MO, was set to 5 for all implicit runs, and the relative errors reported are the maximum over the spatial grid computed at the end of the simulation. Relative error is with respect to analytic evaluations as described in [39]. Note also in the table that the number of time steps is larger than the number of linear iterations for the implicit method. As noted earlier, it is possible for there to be no linear iterations on a Newton iteration if the prediction is good enough. For the semi-implicit runs, the preconditioner is very effective, and hence only 1 linear iteration per time step is needed.

The table shows that for each spatial grid, both methods produce approximately the same errors and that these errors are converging with the same rate as the grid spacing is refined. We also see that the discretization errors are independent of the time integration tolerances, indicating that the integration error is not polluting the spatial discretization error. Thus, the two codes have similar spatial discretization accuracies and we can consider that differences in solutions between the codes are related only to handling of time discretizations and nonlinear couplings.

We note that the effect of the weighted root mean square norm is evident in the fully implicit results for this system. The fully implicit method chooses time steps in order to bound the estimated local truncation error measured in the weighted root mean square norm (15) in terms of the relative (RTOL) and absolute (ATOL) tolerances provided by the user. If CVODE used just a Euclidean 2-norm to choose time steps, then as the problem size got larger (the number of unknowns increased) and if each component of the estimated truncation error stayed about the same as in the smaller problem, the estimated error could get very large. The weighted root mean square norm thus scales the estimated error by the number of unknowns to reduce this effect. To see the motivation for this scaling in the PDE context, consider the following Banach L^2-norm for a spatially dependent function, z, over a domain Ω, $(\int_{\Omega}(z(x))^2 \, dx)^{1/2}$. A discrete approximation of this norm over a domain of length, L, decomposed into a grid of N uniform cells of size, Δx, is

$$\left(\sum_{i=1}^{N}(z_i)^2 \Delta x\right)^{1/2} = \left(\frac{L}{N}\right)^{1/2}\left(\sum_{i=1}^{N}(z_i)^2\right)^{1/2}, \tag{30}$$

where we used the relationship $\Delta x = L/N$. This last expression is just the root mean square norm multiplied by a constant related to the domain size. The result of using the root mean square norm and its induced scaling by the number of unknowns in our case is that time steps do not decrease as fast as one might expect for larger runs.

4.2. Comparisons of fully and semi-implicit

In this section, we present results of two 3D simulations with radiation sources, one in hydrogen and the other in carbon, and results of a 2D problem in hydrogen which includes a time-dependent material energy source. The 3D hydrogen problem is characterized by a rapid diffusion of radiation energy which will be limited by the flux limiter. Diffusion in the 3D carbon problem is slower, and the flux limiter is of less importance. The 2D problem is characterized by a very fast heating rate due to the nonlinear source term. These test problems demonstrate the benefit and accuracy of fully implicit over semi-implicit.

4.2.1. Radiation source problem

In this 3D simulation, energy is supplied to the radiation field by a source with a specified black body temperature. The radiation energy is involved in four physical processes, heating from the source, diffusion of energy out of the heated region, transfer of energy to the material, and interaction with boundaries. The material energy is involved in only one process, heating via transfer of energy from the radiation field.

In these simulations we use the LEOS equation of state data base as described above and a $20 \times 20 \times 20$ grid with 0.01 cm on each side. Homogeneous Neumann conditions are used on all boundaries, and the initial radiation and material temperatures are 15 eV.

The source is spherical positioned in a corner with a sharp boundary of radius 0.004 cm,

$$\chi(\mathbf{x}) = H\left(\sqrt{(x - 0.01)^2 + (y - 0.01)^2 + (z - 0.01)^2}, 0.004 \right),$$

where H is given by (27). The source temperature, T_{source}, is 300 eV.

These simulations were run for a short time interval of 10^{-6} μs. This final time was kept short as these tests were designed to compare the two methods on transient problems. Although the grid used in these examples is coarse, the previous test case shows that errors due to the spatial approximation are the same in both codes and hence cancel when the solutions are subtracted from each other.

Fig. 1 shows the Planck and Rosseland opacities over the temperature ranges included in these runs. We see that the material energy for hydrogen varies more readily with temperature than it does with carbon. Similarly, we see the opacity values decrease faster with temperature for hydrogen than for carbon. These differences result in a more difficult problem for hydrogen than for carbon.

Tables 2 and 3 summarize simulations using hydrogen and carbon, respectively. We see in all cases that allowing the implicit method to go to higher orders (above 2) results in a solution requiring fewer time steps than the second order scheme. Fewer steps are required because the integration method can take larger steps and lower the resulting error by using a higher order method. Comparing the implicit and semi-implicit methods in terms of computation run time and number of steps, the implicit method is faster than the semi-implicit for higher levels of requested accuracy. For the highly resolved solutions, for example, implicit with RTOL = 10^{-8} and semi-implicit with FAC = 10^{-3}, implicit can be several times faster.

For the hydrogen case, we see that the fully implicit method has trouble converging in a reasonable number of time steps for large values of RTOL. We believe this difficulty results from the method becoming numerically unstable. As noted in [40], BDF methods coupled with iterative linear solvers such as GMRES are essentially explicit time integration methods. The iterative solver must do a good job of solving in the subspace generated by the dominant eigenvectors (i.e., the stiff modes). If this does not happen, then these modes can cause the combined BDF-iterative method to become unstable. Because the larger RTOL
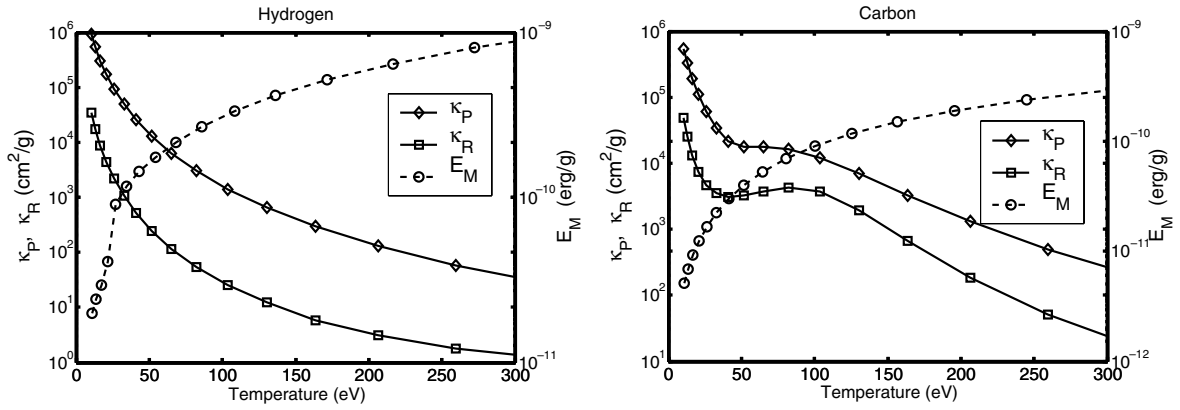
Fig. 1. Opacities over relevant temperature ranges in the 3D hydrogen (left) and carbon (right) simulations for $\rho = 1.0$ g/cc.

Table 2
Statistics for 3D Hydrogen problem

Method	RTOL	MO	FAC	NST	NNI	NLI	RT
Implicit	10^{-5}	2	NA	DNF			
Implicit	10^{-5}	5	NA	DNF			
Implicit	10^{-6}	2	NA	1673	1728	4798	370
Implicit	10^{-6}	5	NA	708	796	2993	266
Implicit	10^{-7}	2	NA	3681	3784	8900	726
Implicit	10^{-7}	5	NA	1037	1153	3674	333
Implicit	10^{-8}	2	NA	7924	8166	17,085	2089
Implicit	10^{-8}	5	NA	2133	2346	6313	524
Semi-imp.	NA	NA	10^{-1}	181	NA	1013	100
Semi-imp.	NA	NA	10^{-2}	1807	NA	9343	667
Semi-imp.	NA	NA	10^{-3}	18,089	NA	78,166	6595

(DNF = Did not finish).

Table 3
Statistics for 3D Carbon problem

| Method	RTOL	MO	FAC	NST	NNI	NLI	RT
Implicit	10^{-5}	2	NA	648	670	1693	156
Implicit	10^{-5}	5	NA	401	412	1099	88
Implicit	10^{-6}	2	NA	1434	1483	3329	314
Implicit	10^{-6}	5	NA	807	852	2024	192
Implicit	10^{-7}	2	NA	3150	3284	6583	581
Implicit	10^{-7}	5	NA	1658	1758	3969	377
Implicit	10^{-8}	2	NA	6894	7285	12,955	1323
Implicit	10^{-8}	5	NA	3462	3832	8112	985
Semi-imp.	NA	NA	10^{-1}	149	NA	654	61
Semi-imp.	NA	NA	10^{-2}	1468	NA	4986	561
Semi-imp.	NA	NA	10^{-3}	14,653	NA	29,302	4662
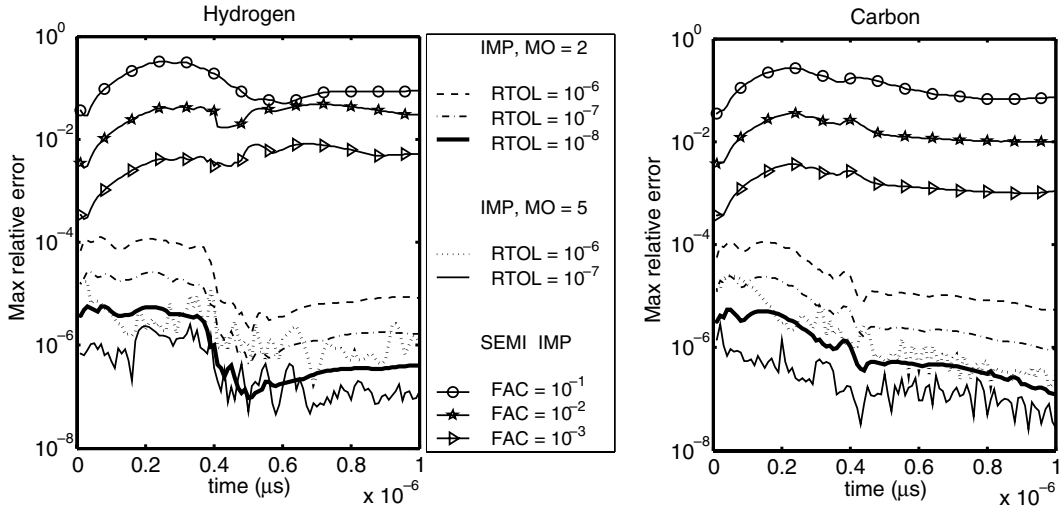
Fig. 2. Evolution of relative errors for radiation temperature in solution of 3D hydrogen (left) and carbon (right) problems. Relative error is with respect to the implicit simulation with RTOL = 10^{-8} and MO = 5.

tolerances allow more error in the solution, we believe unstable modes are growing and causing numerical instabilities. For smaller RTOL values, these modes are damped and do not grow. Also, as RTOL is reduced, we see significant benefits to using the high order in time integration both in fewer numbers of time steps and also in decreased run time as compared to the second order method. For more details on the stability of BDF methods with iterative linear solvers, see [40,41].

Fig. 2 shows the relative error in the radiation temperatures for both the 3D hydrogen and carbon problems. In most cases, we see that going to higher order gives better agreement with the fully resolved run than lower order for the fully implicit method. This difference results from the lower order method requiring smaller time steps in order to maintain accuracy and thus taking more time steps. We know that asymptotically, the local truncation error is smaller for higher order methods. Because we control this local truncation error per step, the global error behaves like a sum of local errors and is consequently dependent on the number of steps. Thus, we expect that the lower order run would have a greater error than the higher order run for a given tolerance. In addition, since we are using iterative solvers for both nonlinear and linear solution methods, extra time steps can result in more chances for one solution to deviate (within tolerances) from another solution. Lastly, round off error can accumulate. The results of these sources of error is the difference that we see between the lower and higher order runs. We also see that for all tolerances considered, the fully implicit method is more accurate than the semi-implicit method. Given that the semi-implicit run times are generally longer than that for the implicit method, significant speed benefits can be delivered with the implicit method.

4.2.2. Fusion source problem

Our next example is a 2D fusion source problem. In this problem we have added a material energy source which has a temperature dependence of T_M^5. This is a good fit to a tritium–deuterium reaction rate at low temperature (less than a few keV) such as in a tokamak fusion experiment [42, p. 29]. The source function, $\mu(\mathbf{x},t)$, is a product of a step function in cylindrical radius and a bi-cubic in time. The source, given in units of $\frac{\text{erg}}{\text{cm}^3 s}\frac{1}{\text{eV}^5}$, which is positioned in the upper right corner of the domain with a sharp boundary in space is given by,
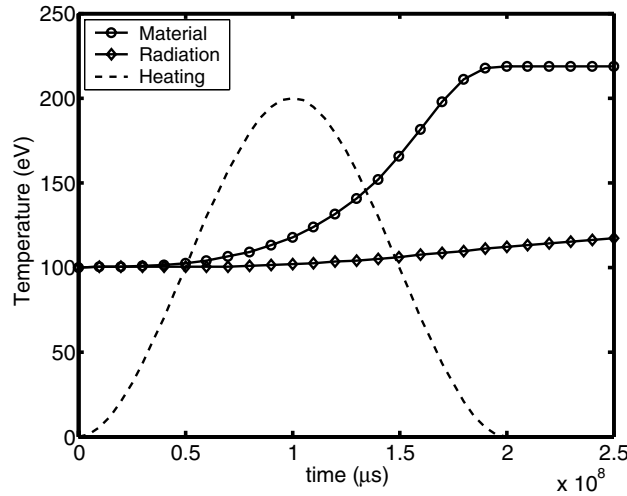
Fig. 3. Evolution of radiation and material temperatures as well as material source function (heating curve) profile in time for the 2D source problem.

$$\mu(\mathbf{x},t) = (2.31 \times 10^{-11}) H\left(\sqrt{(x-0.01)^2 + (y-0.01)^2}, 0.005\right) B(t - 10^{-8}, 10^{-8}),$$

where H and B are given by (27) and (28), respectively.

These simulations in hydrogen use a 20×20 grid with 0.01 cm on each side. The initial temperatures for both radiation and material are 100 eV, and the density was taken to be 1 gm/cm^3. All boundary conditions are Neumann, and flux limiting is used for all runs. The LEOS equation of state package is used for opacity values. The simulations are run to a final time of 2.5×10^{-8} µs.

Fig. 3 shows the history of the radiation and material source function at a point interior to the source region for this problem. For this problem energy is supplied to the material via the source term, transferred to the radiation and lost via diffusion. The strong nonlinear dependence of the source term on material temperature, T_M^5, can lead to very rapid increases in temperature. In these problems the source is turned off by the time dependence in $\mu(\mathbf{x},t)$.

The large heating rates can also lead to some problems with respect to automatic time step control in the semi-implicit simulations. In the initial time period, before the source turns on, the automatic time step control for the semi-implicit method will advance to the maximum allowed time step, HMAX. This advancement results in the system missing the turn-on of the source. In the simulations presented here, for the semi-implicit methods, we bypass this problem by using a small maximum time step. It should be noted that for all the semi-implicit simulations shown in this section, this maximum step size only limits the step selection during the initial stages of the source. No limitations due to this parameter were observed for other times in the simulation. (A more consistent way of performing this simulation would be to have a third equation in the system which models the depletion of a fusion fuel density as its energy is added to the material. Implicit solutions of this three equation system will be the subject of a following paper.)

Tables 4 and 5 show results of our implicit and semi-implicit simulations for the material source problem, respectively. The values of RTOL, HMAX, and FAC have been chosen by trial-and-error to yield a set of runs with similar relative error. In the implicit runs we see that using higher order can lead to a reduction in run time by a factor of two or three for the more accurate small RTOL runs. For similar accuracy the semi-implicit method is much slower. The maximum time step limit, HMAX, is used in these simulations to

Table 4
Statistics for implicit solution of 2D matter source problem

RTOL	MO	NST	NNI	NLI	RT
10^{-4}	2	44	63	136	0.99
10^{-4}	5	36	60	130	1.57
10^{-5}	2	91	126	220	1.66
10^{-5}	5	66	100	175	1.57
10^{-6}	2	175	218	356	3.41
10^{-6}	5	108	167	256	2.08
10^{-7}	2	369	425	632	6.32
10^{-7}	5	201	262	367	3.52
10^{-8}	2	755	849	1212	16.40
10^{-8}	5	300	403	538	4.49
10^{-9}	5	503	695	848	16.90

Table 5
Statistics for semi-implicit solution of 2D matter source problem

FAC	HMAX	NST	NLI	RT
10^{-3}	10^{-9}	904	1635	17
10^{-4}	10^{-10}	9754	17,388	191
10^{-5}	10^{-11}	96,884	130,481	1539
10^{-6}	10^{-12}	751,153	751,152	11,744

restrict the accumulation of error in early times. Without this restriction, the semi-implicit method initially takes a larger time step and is not able to reduce the time step fast enough to resolve the dynamics. The time step is controlled by HMAX up to about 0.1×10^{-8} μs in these simulations. After this initial short period of time the automatic time step control reduces the time step below HMAX.

Fig. 4 shows relative errors of the material temperatures for the 2D source problem. All methods show the same behavior, a significant increase in error once the source turns on, and a leveling off of the error once the source turns off. All methods show convergence, with tolerance, to the highly resolved solution. Although not shown, relative errors in radiation temperature show similar results, but over a smaller scale.

4.3. Order studies for the high order method

In this section, we explore the benefits of higher order and address the issue of which orders give the most gain in computational speed to solution for a given accuracy.

The first test is on a 3D fusion source problem. The source function, μ(**x**,t), is a product of a bi-cubic function in spherical radius and a bi-cubic in time. The source is positioned in the center of the domain and given in units of $\frac{erg}{cm^3}\frac{1}{s}\frac{1}{eV^5}$ as

$$\mu(\mathbf{x},t) = \left(4.75 \times 10^{-11}\right)B\left(t,1.02 \times 10^{-8}\right)B\left(\sqrt{(x-0.005)^2+(y-0.005)^2+(z-0.005)^2},0.0025\right),$$

where B is given by (28). These simulations in hydrogen use a $100 \times 100 \times 100$ grid with 0.01 cm on each side. The initial temperatures for both radiation and material were 100 eV, and the density was taken to be 1 gm/cm^3. All boundary conditions were Dirichlet with a 100 eV temperature, and flux limiting was used for all runs. The LEOS equation of state package was applied for opacity values, and the simulations were run to a final time of 2.5×10^{-7} μs. All runs were done on 8 processors of ASCI Frost which is an IBM SP
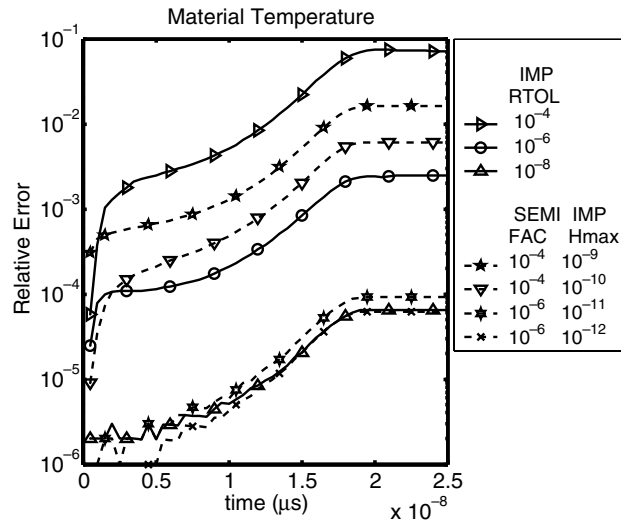
Fig. 4. Evolution of relative errors for material temperatures in the solution of the 2D source problem. Relative error is with respect to the implicit simulation with RTOL = 10^{-9} and MO = 5 at a point interior to the source region.

parallel machine at Lawrence Livermore National Laboratory running the AIX operating system with 1088 375 MHz processors grouped into 16-processor nodes.

Table 6 shows solver statistics for this problem with maximum allowed order of the time integration set at 1, 2, 3, and 5. For the simulations in this table the maximum order actually attained was 4. Thus, the runs with maximum allowed order 4 and 5 are the same. We see a significant gain in going from first to second order and in going from second to third order. Little benefit is seen in going to higher than third order. For this problem, third order is high enough to give the required accuracy and going higher introduces overhead in testing for changes to higher orders which will not give benefit.

Fig. 5 shows the solutions, heating time, and histories of time steps and order choices for a similar test problem with a density of 2 gm/cm^3 and a final time of 2.5×10^{-4} μs. We used a higher density in this case to result in more transfer of energy from the matter to the radiation field. Also in this case we increased the strength of the material heating source. We use the μ(**x**,t) of the above simulation with the leading constant factor increased to 9.5×10^{-11}. This problem was run with MO = 5 and RTOL = 10^{-7}. A longer time was used to give more information on order and step selection. In this problem, we clearly see the heating stage where the matter temperature undergoes significant increases then levels off while some of the energy is transferred to the radiation. Both fields decrease in energy toward the end of the run as energy leaves the domain due to the Dirichlet boundary condition.

Table 6
Statistics for implicit solution of 3D matter source problem

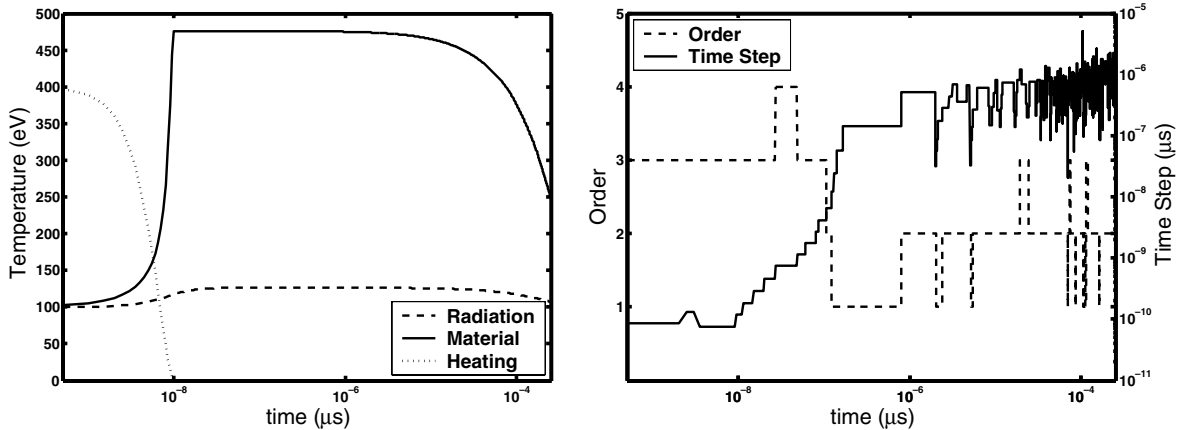| RTOL	MO	NST	NNI	NLI	RT
10^{-7}	1	NA	NA	NA	DNF
10^{-7}	2	316	336	381	2037
10^{-7}	3	168	187	259	1310
10^{-7}	4 or 5	169	188	275	1367
10^{-8}	1	5193	5408	5395	29,218
10^{-8}	2	676	720	820	4268
10^{-8}	3	360	408	549	2736
10^{-8}	4 or 5	342	395	581	2750

Fig. 5. Left: Heating time and temperature solutions for the 3D matter source problem. Temperatures are recorded at the center of the source. Right: Time step and order history for implicit high order solution method.

The order history shows the method initially using third order as the heating region is traversed. After heating, the method reduces to second order for the majority of the rest of the run. The step size stays constant initially, then increases before the exponential portion of the heating begins. When the heating becomes significant, the step size is reduced, then increases again after the heating phase. The step continues to increase with a brief pause coinciding with a move to fourth order. These changes are a result of the method adjusting to the solution "settling down" after the large changes from heating and transfer of energy from matter to radiation. As the energy leaves the system, we see the order and step sizes change simultaneously. In general, a larger time step creates a larger error and a larger order reduces the error. Thus, we see that the method will raise the step and reduce the order, then adjust back. Toward the end of the run, we see these adjustments happen frequently as the solution shows energy decreases. The statistics for this simulation are NST = 744, NNI = 1132, NLI = 1870 and RT = 4,584 s.

These results show the ability of the variable order, variable step method to adjust to solution changes while maintaining a given requirement on the size of the local time integration truncation error. This type of adaptivity results in fewer time steps.

4.4. Results for large-scale computations

In this section, we present results of parallel scalability studies for the fully implicit high order solution method. These studies give a measure of how well the solution method makes use of additional resources to solve larger problems. Although use of tabulated opacities and nonlinear sources does not directly affect parallel efficiency of the implicit method, if these terms are not handled well in the preconditioner, we would see large iteration counts and unacceptably large solution times, making large-scale runs impossible. Thus, to show the implicit method is effective for large-scale problems with these physical attributes, we conduct scalability studies for the three situations: a diffusion problem with constant opacities and a constant radiation source, a diffusion problem with tabulated opacities and a constant radiation source, and a diffusion problem with tabulated opacities and a nonlinear matter source. The first study was done with a constant opacity problem running on ASCI Red at Sandia National Laboratory. This system consist of 4536 nodes each containing two 200 MHz Intel Pentium Pro processors. The next two studies were done using the LEOS tabulated opacity data base on radiation source and fusion source problems running on ASCI Frost.

Table 7
Statistics for scalability study on ASCI Red with constant opacity problem

Processor topology	NST	NNI	NLI	RT	Avg. cost per step	Step scaled efficiency
$1 \times 1 \times 1$	123	140	186	2485	20.2	100%
$2 \times 2 \times 2$	113	127	160	2518	22.3	91%
$4 \times 4 \times 4$	105	119	154	2424	23.1	88%
$8 \times 8 \times 8$	119	136	191	2761	23.2	87%
$16 \times 16 \times 16$	116	129	212	2970	25.6	79%
$18 \times 18 \times 18$	112	130	214	3001	26.8	75%

In all studies in this section except the last, we used the PFMG multigrid method [32]. We applied this algorithm because the PFMG method scales to very large numbers of processors. Our last study was done with the SMG method and compares scaling results using this preconditioner with the PFMG method.

The first study ran on 5832 processors of ASCI Red using one processor per node running MPI for parallel communications. The system (1) and (2) was solved on the box $D \equiv \{(x,y,z):0 \leqslant x,y,z \leqslant 1 \text{ cm}\}$ with no matter source and a constant radiation source with $T_{source} = 300$ eV at the center of the domain. Constant Dirichlet conditions of $T = 300$ K were applied on all boundaries. Initial conditions for both the radiation and matter were given by $T_{R,0} = T_{M,0} = 300$ K. The equation of state was given simply as $T_M = E_M$, and the matter density was taken as $\rho = 1.0$ g/cc. The Planck and Rosseland opacities were set constant and equal as $\kappa_P = \kappa_R = 10^5$ cm^2/g. Flux-limiting was turned on for all runs, and the simulation was run to 0.01 s.

For this study, we added both unknowns and processors as we scaled up the problem keeping a spatial grid of $N_x = N_y = N_z = 40$ on each processor. Thus, problem size and computational resources were simultaneously increased.

Table 7 contains the results of the study. The reported scaled efficiency for a run on N processors was calculated by dividing the cost per step for the single processor run by the cost per step for the N processor case. As can be seen, all the statistics scaled extremely well. As this problem is dominated by the local coupling of the two fields (and not the diffusion operator), a high scaled efficiency is expected. In fact, we see 75% scaled efficiency for the largest test case with $373.2M$ grid cells.

Our next scalability study used tabulated opacities from the LEOS equation of state data base and was run on ASCI Frost. The system (1) and (2) was solved on the box $D \equiv \{(x,y,z):0 \leqslant x,y,z \leqslant 0.01 \text{ cm}\}$ filled with carbon with no matter source and a constant radiation source with $T_{source} = 300$ eV. The source which is positioned in the center of the domain with a sharp boundary and radius of 0.002 cm is given by

$$\chi(\mathbf{x}) = H\left(\sqrt{(x - 0.005)^2 + (y - 0.005)^2 + (z - 0.005)^2}, 0.002 \right),$$

where H is given by (27). Neumann conditions were applied on all boundaries, and the initial temperature for radiation and material was 15 eV. No flux-limiting was applied. The problem was run to a final time of 2.5×10^{-8} μs with a relative tolerance of 10^{-6}.

There were $40 \times 40 \times 40$ grid cells per processor, and we scaled up the number of processors from 1 to 448 giving a total of $28.67M$ grid cells. Since ASCI Frost has 16 processors per node and communication can be faster within a node than without, we used 8 processors per node for all but the two largest runs where we used 12 and 14 processors per node, respectively.

Table 8 shows the solver statistics and scaled efficiencies for this study. We see that as the problem size gets larger, the number of steps and solver iterations go up then decrease, but do not change dramatically. These results indicate that the solution method is able to solve these refined problems effectively. In addition, the run time does not significantly increase with scaling up the processors and unknowns simultaneously. We see a leveling off of the scaled efficiency for the total simulation run time at about 82%.

Table 8
Statistics for scalability study on ASCI Frost with tabulated opacity, radiation heating source problem

Processor topology	NST	NNI	NLI	RT	Scaled efficiency simulation	Scaled efficiency per step
$1 \times 1 \times 1$	489	521	901	1401	100%	100%
$2 \times 1 \times 1$	563	590	1098	1673	84%	96%
$2 \times 2 \times 2$	588	611	1085	1718	82%	98%
$4 \times 2 \times 2$	559	579	1177	1836	76%	87%
$4 \times 4 \times 4$	529	548	1082	1716	82%	88%
$6 \times 6 \times 6$	479	498	983	1735	81%	79%
$8 \times 8 \times 7$	463	482	958	1704	82%	78%

Our last scalability study used a nonlinear matter source and was run on ASCI Frost with the LEOS equation-of-state data base. The system (1) and (2) was solved on the box $D \equiv \{(x,y,z):0 \leqslant x,y,z \leqslant 0.01$ cm$\}$ with no radiation source and a nonlinear material heating source. The box was filled with hydrogen, and flux-limiting was applied. The material heating source, given in units of $\frac{erg}{cm^3}\frac{1}{s}\frac{1}{eV^5}$, is positioned in the center of the domain with a smooth boundary at a radius of 0.0025 cm and is given by

$$\mu(\mathbf{x},t) = \left(4.75 \times 10^{-11}\right)B\left(t,10^{-8}\right)B\left(\sqrt{(x-0.005)^2 + (y-0.005)^2 + (z-0.005)^2},0.0025\right),$$

where B is given by (28). In this simulation, the source is turned off by the time dependence in $\mu(x,t)$. The time dependence of this function is a maximum at $t = 0$ then turned off with a half width in time of 1.0×10^{-8} μs. Neumann conditions were applied on all boundaries. The initial temperature for radiation and material was 100 eV. The problem was run to a final time of 2.5×10^{-8} μs with a relative tolerance of 10^{-7}.

Similar to the previous example, we used $40 \times 40 \times 40$ grid cells per processor and scaled up the number of processors from 1 to 448 with the same numbers of processors per node in use.

We performed two scalability studies for this test problem. In the first, we applied the PFMG multigrid method to solve the Schur complement system (Section 3.1.2). Results for this study are found in Table 9 where we see a scaled efficiency of the simulation of about 67% for 448 processors. Table 10 contains the results of the same study but with the SMG method applied to solve the Schur complement system. Here the scaled efficiencies are still decreasing and are at 46% for 448 processors. The difference between these two studies is due to the fact that the SMG solver includes more coupling and thus requires more parallel communication. The increased couplings result in better algorithmic scaling, as can be seen from the nearly level numbers of time steps, nonlinear iterations, and linear iterations with the SMG preconditioner, but results in less parallel efficiency. These two scaling studies show a classic tradeoff between algorithm and implementation scalabilities. We further note that both the PFMG and SMG preconditioners used in these

Table 9
Statistics for scalability study on ASCI Frost with tabulated opacity, material heating source problem, PFMG multigrid method

Processor topology	NST	NNI	NLI	RT	Scaled efficiency simulation	Scaled efficiency per step
$1 \times 1 \times 1$	124	145	258	439	100%	100%
$2 \times 1 \times 1$	118	136	278	454	97%	92%
$2 \times 2 \times 2$	112	129	255	431	102%	92%
$4 \times 2 \times 2$	133	142	333	529	93%	89%
$4 \times 4 \times 4$	120	134	259	459	96%	93%
$6 \times 6 \times 6$	124	138	269	500	88%	88%
$8 \times 8 \times 7$	144	156	338	654	67%	78%

Table 10
Statistics for scalability study on ASCI Frost with tabulated opacity, material heating source problem, SMG multigrid method

Processor topology	NST	NNI	NLI	RT	Scaled efficiency simulation	Scaled efficiency per step
$1 \times 1 \times 1$	117	141	164	393	100%	100%
$2 \times 1 \times 1$	110	127	171	419	94%	88%
$2 \times 2 \times 2$	101	115	150	414	95%	82%
$4 \times 2 \times 2$	99	110	151	448	88%	74%
$4 \times 4 \times 4$	97	109	156	531	74%	61%
$6 \times 6 \times 6$	106	119	165	688	57%	52%
$8 \times 8 \times 7$	101	113	163	858	46%	40%

studies were taken from the hypre library [43,44], and the developers of this library indicated that they have seen similar differences between the two methods.

From these three scaling studies we observe that the fully implicit method is able to make use of additional computational resources to solve increasingly larger problems with both highly nonlinear sources and with tabulated opacities. The good scaling results mainly from effective preconditioning.

5. Conclusions

We have presented a fully implicit solution method for radiation diffusion problems with highly nonlinear sources. Our method makes use of high order in time integration techniques, inexact Newton–Krylov nonlinear solvers, and multigrid preconditioning. We have incorporated the use of tabular opacities in our model in an effort to enhance the accuracy of our test problems as well as to evaluate the added costs of additional function evaluations in the fully implicit approach. Our results indicate that a fully implicit solution approach can achieve more accurate solutions than semi-implicit solution methods in many simulations involving the interaction of radiation and matter with highly nonlinear source terms. Furthermore, the fully implicit approach can be as cost effective as semi-implicit approaches in many cases despite the use of tabulated values for the opacities. We did see, however, that when only low accuracy results are required, the semi-implicit method may be the preferred method due to speed and (in the case of very low accuracy) robustness. Lastly, the solution approach is shown to scale well to very large problems solved on parallel machines.

References

[1] A.C. Hindmarsh, Preliminary Documentation of GEARBI: Solution of ODE systems with block-iterative treatment of the Jacobian, Technical Report UCID-30149, Lawrence Livermore National Laboratory, December 1976.

[2] T.S. Axelrod, P.F. Dubois, C.E. Rhoades, An implicit scheme for calculating time- and frequency-dependent flux limited radiation diffusion in one dimension, J. Comput. Phys. 54 (1984) 205–220.

[3] P.N. Brown, A.C. Hindmarsh, Reduced storage matrix methods in stiff ODE systems, J. Appl. Math. Comput. 31 (1989) 40–91.

[4] P.N. Brown, Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, SIAM J. Sci. Statist. Comput. 11 (1990) 450–481.

[5] D.A. Knoll, D.E. Keyes, Jacobian-free Newton-Krylov methods: a survey of approaches and applications, J. Comput. Phys. 193 (2) (2004) 357–397.

[6] D.A. Knoll, W.J. Rider, G.L. Olson, An efficient nonlinear solution method for nonequilibrium radiation diffusion, J. Quantum Spectrosc. Radiat. Transfer 63 (1999) 15–29.

[7] W.J. Rider, D.A. Knoll, G.L. Olson, A multigrid Newton–Krylov method for multimaterial equilibrium radiation diffusion, J. Comput. Phys. 152 (1999) 164–191.

[8] W.J. Rider, D.A. Knoll, Time step selection for radiation diffusion calculations, J. Comput. Phys. 152 (1999) 790–795.

[9] P.N. Brown, B. Chang, F. Graziani, C.S. Woodward, Implicit solution of large-scale radiation-material energy transfer problems, in: D.R. Kincaid, A.C. Elster (Eds.), Iterative Methods in Scientific Computation IV, International Association for Mathematics and Computers in Simulations, New Brunswick, NJ, 1999, pp. 343–356.

[10] G.D. Byrne, A.C. Hindmarsh, PVODE, an ODE solver for parallel computers, Int. J. High Perf. Comput. Appl. 13 (1999) 354–365.

[11] V.A. Mousseau, D.A. Knoll, W.J. Rider, Physics-based preconditioning and the Newton–Krylov method for non-equilibrium radiation diffusion, J. Comput. Phys. 160 (2000) 743–765.

[12] D.A. Knoll, W.J. Rider, G.L. Olson, Nonlinear convergence, accuracy, and time step control in nonequilibrium radiation diffusion, J. Quantum Spectrosc. Radiat. Transfer 70 (1) (2001) 25–36.

[13] P.N. Brown, C.S. Woodward, Preconditioning strategies for fully implicit radiation diffusion with material-energy transfer, SIAM J. Sci. Comput. 23 (2) (2001) 499–516.

[14] J.W. Bates, D.A. Knoll, W.J. Rider, R.B. Lowrie, V.A. Mousseau, On consistent time-integration methods for radiation hydrodynamics in the equilibrium diffusion limit: low-energy-density regime, J. Comput. Phys. 167 (2001) 99–130.

[15] L.H. Howell, J.A. Greenough, Radiation diffusion for multi-fluid eulerian hydrodynamics with adaptive mesh refinement, J. Comput. Phys. 184 (2003) 53–78.

[16] A. Brandt, Multigrid techniques: 1984 guide with applications to fluid dynamics, Tech. Rep. monograph, Weizmann Institute of Science, available as GMD-Studie No. 85, from GMD-FIT, Postfach 1240, D-5205, St. Augustin 1, Germany, February 1984.

[17] D.J. Mavriplis, Multigrid approaches to non-linear diffusion problems on unstructured meshes, Numer. Linear Alg. Appl. 8 (8) (2001) 499–512.

[18] D.J. Mavriplis, An assessment of linear versus nonlinear multigrid methods for unstructured mesh solvers, J. Comput. Phys. 175 (2002) 302–325.

[19] L. Stals, Comparison of non-linear solvers for the solution of radiation transport equations, Elec. Trans. Numer. Anal. 15 (2003) 78–93.

[20] R.B. Lowrie, A comparison of implicit time integration methods for nonlinear relaxation and diffusion, J. Comput. Phys. 196 (2004) 566–590.

[21] D.L. Ropp, J.N. Shadid, C.C. Ober, Studies of the accuracy of time integration methods for reaction-diffusion equations, J. Comput. Phys. 194 (2004) 544–574.

[22] E.M. Corey, D.A. Young, A new prototype equation of state data library, Technical Report UCRL-JC-127698, Lawrence Livermore National Laboratory, Livermore, CA.

[23] G.C. Pomraning, The Equations of Radiation Hydrodynamics, Pergamon Press, New York, 1973.

[24] R.L. Bowers, J.R. Wilson, Numerical Modeling in Applied Physics and Astrophysics, Jones and Bartlett, Boston, 1991.

[25] M. Basko, A model for the conversion of ion-beam energy into thermal radiation, Phys. Fluids B 4 (11) (1992) 3753–3763.

[26] E. Minguez, P. Martel, J. Gil, J. Rubiano, R. Rodriguez, Analytical opacity formulas for ICF elements, Fusion Eng. Des. 60 (2002) 17–25.

[27] G.D. Byrne, Pragmatic experiments with Krylov methods in the stiff ODE setting, in: J.R. Cash, I. Gladwell (Eds.), Computational Ordinary Differential Equations, Oxford University Press, Oxford, 1992, pp. 323–356.

[28] P.N. Brown, G.D. Byrne, A.C. Hindmarsh, VODE: a variable-coefficient ODE solver, SIAM J. Sci. Stat. Comput. 10 (5) (1989) 1038–1051.

[29] K.R. Jackson, R. Sacks-Davis, An alternative implementation of variable step-size multistep formulas for stiff ODEs, ACM Trans. Math. Software 6 (1980) 295–318.

[30] A.C. Hindmarsh, P.N. Brown, K.E. Grant, S.L. Lee, R. Serban, D.E. Shumaker, C.S. Woodward, SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers, ACM Trans. Math. Soft. Also available as Lawrence Livermore National Laboratory Technical Report UCRL-JP-200037 (to be published).

[31] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.

[32] S.F. Ashby, R.D. Falgout, A parallel multigrid preconditioned conjugate gradient algorithm for groundwater flow simulations, Nucl. Sci. Eng. 124 (1) (1996) 145–159.

[33] S. Schaffer, A semi-coarsening multigrid method for elliptic partial differential equations with highly discontinuous and anisotropic coefficients, SIAM J. Sci. Comput. 20 (1) (1998) 228–242.

[34] P.N. Brown, R.D. Falgout, J.E. Jones, Semicoarsening multigrid on distributed memory machines, SIAM J. Sci. Stat. Comput. 21 (5) (2000) 1823–1834.

[35] J.E. Jones, C.S. Woodward, Newton–Krylov-multigrid solvers for large-scale, highly heterogeneous, variably saturated flow problems, Adv. Water Resour. 24 (2001) 763–774.

[36] W.F. Briggs, V.E. Henson, S.F. McCormick, A Multigrid Tutorial, second ed., SIAM, Philadelphia, PA, 2000.

[37] L.F. Shampine, Error estimation and control for ODEs, SIAM J. Sci. Comput. Also available at http://faculty.smu.edu/lshampin/errest.pdf (to be published).

[38] L.F. Shampine, A. Witt, A simple step size selection algorithm for ODE codes, J. Comput. Appl. Math. 58 (1995) 345–354.

[39] B. Su, G.L. Olson, Benchmark results for the non-equilibrium Marshak diffusion problem, J. Quantum Spectrosc. Radiat. Transfer 56 (3) (1996) 337–351.

[40] M. Botchev, G. Sleijpen, H. van der Vorst, Stability control for approximate time stepping schemes with minimal residual iterations, Appl. Numer. Math. 31 (3) (1999) 239–253.

[41] C.W. Gear, Y. Saad, Iterative solution of linear equations in ODE codes, SISC 4 (4) (1983) 583–601.

[42] T.J. Dolan, Fusion researchPrinciples, vol. 1, Pergamon Press, Oxford, 1980, p. 29.

[43] R.D. Falgout, U.M. Yang, *hypre*: a library of high performance preconditioners, in: P. Sloot, C. Tan, J. Dongarra, A. Hoekstra (Eds.), Computational Science – ICCS 2002 Part III, Lecture Notes in Computer Science, vol. 2331, Springer, Verlag, 2002, pp. 632–641.

[44] *hypre*: high performance preconditioners. Available from: <http://www.llnl.gov/CASC/hypre/>.