JOURNAL OF COMPUTATIONAL PHYSICS **139,** 58–78 (1998)
ARTICLE NO. CP975863

Numerical Simulations for Nonlinear Heat Transfer in a System of Multimaterials

Wenlong Dai and Paul R. Woodward

University of Minnesota, 116 Church Street S.E., Minneapolis, Minnesota 55455
E-mail: wenlong@lcse.umn.edu

Received February 24, 1997; revised October 1, 1997

An implicit scheme is developed for nonlinear heat transfer problems. The scheme possesses a number of properties. The most notable are the second-order accuracy in both space and time, the conservative feature, quick damping of numerical errors when the size of time step is large, the iterative approach and fast convergence, the accurate treatment for nonlinearities and different kinds of material, and the capability to handle a system composed of more then one kind of material, which have dramatically different thermal diffusivities. The scheme may be easily vectorized. Numerical examples are presented to show these features.

Key Words: heat conduction; heat transfer; finite difference; iterative; multigrid.

1. INTRODUCTION

An extensive amount of literature exists on numerical methods for the solution of heat transfer (see, e.g., [13, 21, 24] and references therein). Each method has its advantages depending on the nature of the physical problem to be solved. In the point of view of energy conservation, a numerical scheme may be either conservative or nonconservative. By conservative, it is meant that a numerical solution satisfies the energy conservation law for each grid cell and for any assembly of grid cells. In terms of accuracy, schemes may be divided into first order, second order, and higher order ones. If time accuracy is important, second order or higher order schemes are preferred. But, normally, a scheme accurate more than second order is very complicated to formulate and expensive in CPU time. Therefore, second-order schemes have become practical for time-dependent problems.

Schemes may also be divided into explicit and implicit methods. An explicit scheme, for example, the forward Euler scheme, is simple. But, the size of time step is limited by a stability condition which is normally much smaller than the required accuracy. Therefore, an explicit scheme may be inefficient for some of problems, especially when the thermal diffusivity in a problem varies significantly. On the other hand, in implicit schemes, the

0021-9991/98 $25.00

size of time step is not limited by any stability condition, and therefore may be changed according to the requirement of a problem itself. But implicit schemes normally involve solving a large set of algebraic equations at each time step.

There are two approaches to solve the large set of algebraic equations, i.e. direct approach and iterative approach. The direct approach for solving linear algebraic equations are presented in all traditional courses of linear algebra (see, e.g., [3]). Generally, exact solvers may not be recommended even for linear problems in two and three dimensions because normally an exact solver is expensive in CPU time and is difficult to vectorize. For the iterative approach, a significant question is whether an iterative process will actually be successful and will lead to the solution of the algebraic equations. An important related aspect is the rate of convergence. Several procedures are available to analyze convergence for some simple situations (see, e.g., [7, 8]). The nonlinearity is another headache in an implicit scheme. Newton iteration is very expensive in CPU time in an implicit scheme because there are a large set of unknown variables. Iterative methods for a class of nonlinear difference schemes have been discussed in [23].

Two typical implicit schemes are the backward Euler scheme and the Crank–Nicolson scheme [1]. The backward Euler scheme is first order accurate. Numerical errors in the backward Euler scheme undergo quick damping for large time steps, and therefore, it is very useful for steady state problems. Although the Crank–Nicolson scheme is second-order accurate, numerical errors do not damp out for large time steps.

In this paper, we will develop a numerical scheme for nonlinear heat transfer in multidimensions. The scheme will have the following features: second-order accurate in both space and time, stable for any size of time step, conservative, iterative, accurate in the treatment for nonlinearities and different kinds of material, and capable of handling a system composed of multikinds of material. The development of a numerical scheme with these features is mainly motivated for two kinds of problems. One is the numerical treatment for radiative hydrodynamics [17], and the other is the heat transfer involved in laser fusion [19]. For these kinds of problem, the temporal accuracy is important because the evolution of a physical system with time is what we have to find out. Since the size of a time step in an explicit scheme is dictated by the maximum of the thermal diffusivity in a system no matter whether or not the local phenomena is important to the dynamics, implicit schemes stable for any size of time step are preferred for the problems in which thermal diffusivities either in different regions or at different instants are significantly different. Since most problems in radiative hydrodynamics and laser fusion are nonlinear, a sufficiently accurate treatment for the nonlinearity is required. In laser fusion, the thermal diffusivity in a pellet is dramatically different from that in its surroundings. A numerical scheme should be able to handle the thermal diffusivity which varies dramatically with space coordinates.

The time discretization to be presented in this paper is adopted from the implicit–explicit hybrid schemes [20, 25] for the Euler equations in gas dynamics. Resolving shock fronts is one of major difficulties in gas dynamics, which was handled through an approximate Riemann solver. The schemes in [20, 25] are good only for hyperbolic systems of conservation laws, and only a single kind of material is considered in [20, 25]. Since the Euler equations describe nonlinear wave interactions, a multicolor relaxation, instead of the multigrid method, has been used in [25] for the fast convergence of iterations.

The plan of this paper is as follows. In the second section the difference equations are derived. An iterative solver for the difference equations is described in the third section. The implementation of a multigrid method for the scheme is presented in the fourth section.

Numerical examples are given in the fifth section to demonstrate the features mentioned above. The final section is the conclusion and a brief discussion about our approach.

2. DIFFERENCE EQUATIONS

Suppose a system of heat transfer is composed of multikinds of material, each of which occupies a part of a simulation domain. The material in different regions may be dramatically different in thermal diffusivity. In this section, we will derive the formulations for the two-dimensional situation. The extension to the three-dimensional situation is straightforward.

We will solve the heat transfer problem:

$$\frac{\partial T}{\partial t} - \nabla \cdot [\kappa(T)\nabla T] = s(T). \tag{1}$$

Here T is temperature, κ is thermal diffusivity, and $s(T)$ is a thermal source. Equation (1) should be completed by boundary and initial conditions. Initial and boundary conditions are problem-dependent. Two typical boundary conditions are those for fixed temperature and fixed heat flux at a boundary of a simulation domain. At the interfaces between two kinds of material, temperature and heat flux are continuous, but the thermal diffusivity and spatial derivatives of temperature may be discontinuous across the interfaces. In this paper, we present only the formulations for Eq. (1), but our approach to be presented in this paper may be directly applied to more general equations for heat transfer problems. We will discuss the extension at the end of this paper.

Consider a numerical grid $\{x_i, y_j\}$ in a two-dimensional domain. Integrating Eq. (1) in a grid cell (x_i, x_{i+1}) and (y_j, y_{j+1}), and over a time step $(0, \Delta t)$ yields

$$T^N = T_0 + \frac{\Delta t}{\Delta x}(\bar{q}_{xW} - \bar{q}_{xE}) + \frac{\Delta t}{\Delta y}(\bar{q}_{yS} - \bar{q}_{xN}) + \bar{s}\Delta t. \tag{2}$$

Here T_0 and T^N are cell-averaged values of T at $t = 0$ and $t = \Delta t$, respectively, $\bar{q}_{xW}$ and $\bar{q}_{xE}$ (or, $\bar{q}_{yS}$ and $\bar{q}_{yN}$) are the time-averaged values of a flux at cell-interfaces of the cell to the west and east (or, to the south and north), respectively, $\bar{s}$ is the time- and cell-averaged value of s. They are defined as

$$T^N \equiv \frac{1}{\Delta x \Delta y}\int_{y_j}^{y_{j+1}}\int_{x_i}^{x_{i+1}} T(x, y; \Delta t)\, dx\, dy, \quad \bar{s} \equiv \frac{1}{\Delta t \Delta x \Delta y}\int_0^{\Delta t}\int_{y_j}^{y_{j+1}}\int_{x_i}^{x_{i+1}} s\, dx\, dy\, dt,$$

$$\bar{q}_{xW} \equiv \frac{1}{\Delta t \Delta y}\int_0^{\Delta t}\int_{y_j}^{y_{j+1}} q_x(x_i, y; t)\, dy\, dt, \quad \bar{q}_{xE} \equiv \frac{1}{\Delta t \Delta y}\int_0^{\Delta t}\int_{y_j}^{y_{j+1}} q_x(x_{i+1}, y; t)\, dy\, dt, \tag{3}$$

$$\bar{q}_{yS} \equiv \frac{1}{\Delta t \Delta x}\int_0^{\Delta t}\int_{x_i}^{x_{i+1}} q_y(x, y_j; t)\, dx\, dt, \quad \bar{q}_{yN} \equiv \frac{1}{\Delta t \Delta x}\int_0^{\Delta t}\int_{x_i}^{x_{i+1}} q_y(x, y_{j+1}; t)\, dx\, dt, \tag{4}$$

and $\mathbf{q} \equiv -\kappa(T)\nabla T$. We should mention that Eq. (2) is exact because no approximations have been involved yet.

If the time-averaged flux in Eq. (2) is replaced by a value at $t = 0$ (or at $t = \Delta t$), the approximation results in the forward (or backward) Euler scheme which is first-order accurate

in time. If the time-averaged flux is replaced by their averaged values at $t = 0$ and $t = \Delta t$, the result is the Crank–Nicolson scheme, which is second-order accurate in time. As stated before, numerical errors in the Crank–Nicolson scheme do not undergo damping when the size of a time step is large, although the scheme is unconditionally stable. The reason for the absence of damping is that when the size of a time step is very large, the solution should be independent on the initial condition of a problem, and the solution is determined only by the boundary condition. But, in the Crank–Nicolson scheme, the flux calculation is based on the values at $t = \Delta t$, as well as initial values.

In order to introduce quick damping for numerical errors within the framework of second-order accuracy, we introduce an additional time level $t = \Delta t/2$. Under second-order accuracy, we approximately evaluate the time-averaged flux at $t = \Delta t/2$, and Eq. (2) becomes

$$T^N = T_0 + \frac{\Delta t}{\Delta x}\left(q_{xW}^H - q_{xE}^H\right) + \frac{\Delta t}{\Delta y}\left(q_{yS}^H - q_{xN}^H\right) + s^H \Delta t. \tag{5}$$

Here $s^H \equiv s(T^H)$, T^H is the cell-average of T at $t = \Delta t/2$, and

$$q_{xW}^H \equiv \frac{1}{\Delta y}\int_{y_j}^{y_{j+1}} q_x(x_i, y; \Delta t/2)\, dy, \quad q_{xE}^H \equiv \frac{1}{\Delta y}\int_{y_j}^{y_{j+1}} q_x(x_{i+1}, y; \Delta t/2)\, dy, \tag{6}$$

$$q_{yS}^H \equiv \frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} q_y(x, y_j; \Delta t/2)\, dx, \quad q_{yN}^H \equiv \frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} q_y(x, y_{j+1}; \Delta t/2)\, dx. \tag{7}$$

For the first half-time step, through the similar procedure for Eq. (2), we have

$$T^H = T_0 + \frac{\Delta t}{2\Delta x}\left(\bar{q}_{xW}^H - \bar{q}_{xE}^H\right) + \frac{\Delta t}{2\Delta y}\left(\bar{q}_{yS}^H - \bar{q}_{xN}^H\right) + \frac{1}{2}\bar{s}^H \Delta t. \tag{8}$$

Here $\bar{s}^H$ is the time- and cell-averaged value of s over the first half-time step, and $\bar{q}_{xW}^H$ and $\bar{q}_{xE}^H$ (or, $\bar{q}_{yS}^H$ and $\bar{q}_{yN}^H$) are the time-averaged values of the flux over the first half-time step, and they are defined as

$$\bar{q}_{xW}^H \equiv \frac{2}{\Delta t \Delta y}\int_0^{\Delta t/2}\int_{y_j}^{y_{j+1}} q_x(x_i, y; t)\, dy\, dt, \quad \bar{q}_{xE}^H \equiv \frac{2}{\Delta t \Delta y}\int_0^{\Delta t/2}\int_{y_j}^{y_{j+1}} q_x(x_{i+1}, y; t)\, dy\, dt, \tag{9}$$

$$\bar{q}_{yS}^H \equiv \frac{2}{\Delta t \Delta x}\int_0^{\Delta t/2}\int_{x_i}^{x_{i+1}} q_y(x, y_j; t)\, dx\, dt, \quad \bar{q}_{yN}^H \equiv \frac{2}{\Delta t \Delta x}\int_0^{\Delta t/2}\int_{x_i}^{x_{i+1}} q_y(x, y_{j+1}; t)\, dx\, dt. \tag{10}$$

The time-averaged flux involved in Eq. (8) may be approximately calculated through an interpolation in time. As stated before, an approximate calculation for the time-averaged flux must not, even partially, depend on initial information when a time step is very large. Our interpolation for the time-averaged flux is uniquely determined by values at $t = \Delta t$

and $t = \Delta t/2$. Therefore, the time-averaged flux in Eq. (8) is approximately obtained:

$$\bar{q}^H_{xW} \approx \frac{3}{2}q^H_{xW} - \frac{1}{2}q^N_{xW}, \quad \bar{q}^H_{xE} \approx \frac{3}{2}q^H_{xE} - \frac{1}{2}q^N_{xE}, \tag{11}$$

$$\bar{q}^H_{yS} \approx \frac{3}{2}q^H_{yS} - \frac{1}{2}q^N_{yS}, \quad \bar{q}^H_{yN} \approx \frac{3}{2}q^H_{yN} - \frac{1}{2}q^N_{yN}. \tag{12}$$

Here, q^N_{xW}, q^N_{xE}, q^N_{yS}, and q^N_{yN} have the similar definitions as Eqs. (6), (7), except that they are evaluated at $t = \Delta t$ instead of $\Delta t/2$.

In order to give specific forms of the flux at interfaces, we write the flux as $\mathbf{q} = -\nabla K(T)$. If the material in the current cell is the same as that in its neighboring cells, then spatial derivatives of temperature are continuous across interfaces between grid cells, and therefore we can use a center difference to approximately calculate the flux at four interfaces. For example,

$$q^H_{xW} \approx \frac{1}{\Delta x}\left[K\left(T^H_L\right) - K(T^H)\right]. \tag{13}$$

Here, the subscript L refers to the left cell to the current cell, and T^H_L is the cell-averaged value in the left cell at $t = \Delta t/2$. Since the material in a cell may be dramatically different from that in its neighboring cells, the derivative of temperature is no longer continuous across an interface. Therefore, Eq. (13) is not true at an interface between two kinds of material. Consider the flux at an interface $x = x_i$ between two kinds of material and suppose the temperature at $x = x_i$ is T^H_*. Values of the flux calculated from two sides of the interface are approximately

$$q^H_{x_i-} \approx \frac{2}{\Delta x}\left[K_L\left(T^H_L\right) - K_L(T^H_*)\right], \quad q^H_{x_i+} \approx \frac{2}{\Delta x}[K(T^H_*) - K(T^H)]. \tag{14}$$

Here, the function $K(T)$ is for the material to the right of x_i, while the function $K_L(T)$ is for the material to the left of x_i. Two functions $K(T)$ and $K_L(T)$ may be dramatically different. Since $q^H_{x_i-} = q^H_{x_i+}$, we may find T^H_* through the equation:

$$K(T^H_*) + K_L(T^H_*) = K(T^H) + K_L\left(T^H_L\right).$$

Generally, the temperature at the interface, T^H_*, may be obtained numerically for given T^H and T^H_L, although T^H_* is a linear function of T^H and T^H_L for linear problems.

Writing $K(T)$ in the form of $\hat{\kappa}(T)T$ and writing the equation above in the form

$$T^H_* = \frac{1}{\hat{\kappa}_L(T^H_*) + \hat{\kappa}(T^H_*)}\left[K(T^H) + K_L\left(T^H_L\right)\right],$$

we may write the flux q^H_{xW} in the form

$$q^H_{xW} = \frac{1}{\Delta x}\left[\left(1 - \alpha^H_L\right)K_L\left(T^H_L\right) - \left(1 + \alpha^H_L\right)K(T^H)\right]. \tag{15}$$

Here, α^H_L is defined as

$$\alpha^H_L \equiv \frac{\hat{\kappa}_L(T^H_*) - \hat{\kappa}(T^H_*)}{\hat{\kappa}_L(T^H_*) + \hat{\kappa}(T^H_*)}.$$

Equation (15) is the general form for the flux. Similarly, we may write the flux at other three interfaces:

$$q_{yS}^H = \frac{1}{\Delta y}\left[\left(1-\alpha_B^H\right)K_B\left(T_B^H\right) - \left(1+\alpha_B^H\right)K\left(T^H\right)\right], \tag{16}$$

$$q_{xE}^H = -\frac{1}{\Delta x}\left[\left(1-\alpha_R^H\right)K_R\left(T_R^H\right) - \left(1+\alpha_R^H\right)K\left(T^H\right)\right], \tag{17}$$

$$q_{yN}^H = -\frac{1}{\Delta x}\left[\left(1-\alpha_T^H\right)K_T\left(T_T^H\right) - \left(1+\alpha_T^H\right)K\left(T^H\right)\right]. \tag{18}$$

Applying the flux above in Eq. (5), we obtain a set of nonlinear difference equations:

$$T^N = T_0 + D^H, \tag{19}$$

$$T^H = T_0 + \frac{3}{4}D^H - \frac{1}{4}D^N. \tag{20}$$

Here, D^H is defined as

$$\begin{aligned} D^H \equiv \frac{\Delta t}{(\Delta x)^2}&\left[\left(1-\alpha_L^H\right)K_L\left(T_L^H\right) - \left(1+\alpha_L^H\right)K\left(T^H\right) + \left(1-\alpha_R^H\right)K_R\left(T_R^H\right)\right. \\ &\left. - \left(1+\alpha_R^H\right)K\left(T^H\right)\right] + s^H\Delta t + \frac{\Delta t}{(\Delta y)^2}\left[\left(1-\alpha_B^H\right)K_B\left(T_B^H\right)\right. \\ &\left. - \left(1+\alpha_B^H\right)K\left(T^H\right) + \left(1-\alpha_T^H\right)K_T\left(T_T^H\right) - \left(1+\alpha_T^H\right)K\left(T^H\right)\right], \end{aligned} \tag{21}$$

and D^N has the same form as the equation above if the superscript H is replaced by N. If there is only one kind of material, then α_L^H, α_R^H, α_B^H, and α_T^H vanish. For the steady state of a single kind of material, Eqs. (19), (20) reduce to

$$2\left(\frac{1}{\Delta x^2}+\frac{1}{\Delta y^2}\right)K(T) = \frac{1}{\Delta x^2}[K(T_L)+K(T_R)] + \frac{1}{\Delta y^2}[K(T_B)+K(T_T)], \tag{22}$$

which turns into the usual Poisson solver if the system is linear.

Since we have used the linear interpolations in Eqs. (11), (12), (14) in both space and time, Eqs. (19) and (20) are second-order accurate in both space and time. In Eqs. (19), (20), we have not introduced any approximation for the nonlinearity of $K(T)$ and our interpolation equations (11), (12), (14) do not cross any interface between two kinds of material. Therefore, our treatment for the nonlinearity and different kinds of material is accurate. The numerical solution we are looking for is the solution of the difference equations (19), (20).

We should mention that the extra half-time step and the interpolation based on $\Delta t/2$ and Δt were first introduced in [20] for gas dynamics and are iteratively implemented in [26]. Compared to two typical implicit schemes, the backward Euler and the Crank–Nicolson schemes, the number of unknown variables has been doubled in Eqs. (19), (20). But, Eqs. (19), (20) have advantages we need. As we mentioned before, the backward Euler scheme is only first-order accurate in time, and the numerical error in the Crank–Nicolson scheme do not damp for large time steps. The difference scheme, the set of Eqs. (19), (20), provides quick damping for numerical errors for large time steps, as demonstrated in [25].

Actually, three (or more) time-level implicit schemes have been developed in order to reach high-order accuracy. For example, the three time-level implicit Dupont scheme

[10, 16] may produce excellent results for nonlinear problems. The three-level scheme of Lees [6] does not require iteration over a time step to handle nonlinearities. Since the flux calculation in the previous three (or more) time-level implicit schemes depends on initial data, numerical errors do not undergo damping when the size of the time step is very large.

3. ITERATIVE SOLVER

Equations (19), (20) may be iteratively solved. A straightforward procedure is to evaluate the right-hand sides (RHSs) of Eqs. (19), (20) using an initial guess for cell-averaged values of temperature at $t = \Delta t/2$ and $t = \Delta t$. Thus, we obtain improved temperature at $t = \Delta t/2$ and $t = \Delta t$ through Eqs. (19), (20). Unfortunately, this iterative procedure dose not converge when $\kappa \Delta t/(\Delta x)^2$ is larger than unity, because through each iteration numerical errors in T^N are increased by a factor larger than unity when $\kappa \Delta t/(\Delta x)^2$ is larger than unity.

The nonlinearity in Eqs. (19), (20) is another headache. One of typical approaches for nonlinearities is to use Newton iteration. But the calculation for Jacobi coefficients is extremely time consuming if the number of unknown variables is large. Another typical approach is to linearize Eqs. (19), (20) around the initial temperature. If the linearization were used, numerical errors should not undergo a damping for large time steps, because calculation of the flux is based on the initial temperature in the linearization.

Our approach is as follows: We write Eq. (15) in the form

$$\begin{aligned} q_{xW}^H &= \frac{1}{\Delta x}\left\{\left(1-\alpha_L^H\right)\left[K_L\left(T_L^H\right) - K_L(T^H)\right] + \left(1-\alpha_L^H\right)K_L(T^H) - \left(1+\alpha_L^H\right)K(T^H)\right\} \\ &= \frac{1}{\Delta x}\left[\left(1-\alpha_L^H\right)\tilde{\kappa}_L^H\left(T_L^H - T^H\right) - g_L^H T^H\right]. \end{aligned} \tag{23}$$

Here, $\tilde{\kappa}_L^H$ is the Taylor expansion of $[K_L(T) - K_L(T^H)]$ at $T = T_L^H$ divided by $(T_L^H - T^H)$, and

$$g_L^H \equiv \left(1+\alpha_L^H\right)\hat{\kappa}(T^H) - \left(1-\alpha_L^H\right)\hat{\kappa}_L(T^H).$$

Generally, $\tilde{\kappa}_L^H$ and g_L^H are functions of both T_L^H and T^H. The form of the function g_L^H depends on the properties of the material on both the current and left cells. Equations (16)–(18) may be similarly rewritten. We should point out that we have not introduced any approximation in Eq. (23), and therefore, our treatment for the nonlinearity and for different kinds of material is completely nonlinear.

Using these expressions for the flux, from Eqs. (19), (20) we obtain

$$T^N + \beta^H T^H = T_0 + Q^H, \tag{24}$$

$$-\frac{1}{4}\beta^N T^N + \left(1 + \frac{3}{4}\beta^H\right)T^H = T_0 + \frac{3}{4}Q^H - \frac{1}{4}Q^N. \tag{25}$$

Here, β^H and Q^H are defined as

$$\begin{aligned} \beta^H \equiv\ & \frac{\Delta t}{(\Delta x)^2}\left[\left(1-\alpha_L^H\right)\tilde{\kappa}_L^H + g_L^H + \left(1-\alpha_R^H\right)\tilde{\kappa}_R^H + g_R^H\right] \\ & + \frac{\Delta t}{(\Delta y)^2}\left[\left(1-\alpha_B^H\right)\tilde{\kappa}_B^H + g_B^H + \left(1-\alpha_T^H\right)\tilde{\kappa}_T^H + g_T^H\right] \end{aligned} \tag{26}$$

$$Q^H \equiv \frac{\Delta t}{(\Delta x)^2}\left[\left(1-\alpha_L^H\right)\tilde{\kappa}_L^H T_L^H + \left(1-\alpha_R^H\right)\tilde{\kappa}_R^H T_R^H\right] + \frac{\Delta t}{(\Delta y)^2}\left[\left(1-\alpha_B^H\right)\tilde{\kappa}_B^H T_B^H + \left(1-\alpha_T^H\right)\tilde{\kappa}_T^H T_T^H\right] + s^H \Delta t, \tag{27}$$

and β^N and Q^N are exactly the same as Eqs. (26), (27), except for the superscript H which should be replaced by N for β^N and Q^N. Equations (24), (25) may be solved for T^N and T^H:

$$T^N = \frac{1}{A}\left[\left(1-\frac{1}{4}\beta^H\right)T_0 + Q^H + \frac{1}{4}\beta^H Q^N\right], \tag{28}$$

$$T^H = \frac{1}{A}\left[\left(1+\frac{1}{4}\beta^N\right)T_0 + \frac{1}{4}(3+\beta^N)Q^H - \frac{1}{4}Q^N\right]. \tag{29}$$

Here A is defined as

$$A \equiv 1 + \frac{1}{4}\beta^H(3+\beta^N).$$

Our iterative procedure is as follows: Initially we guess the cell-averaged values of the temperature at $t = \Delta t$ and $t = \Delta t/2$ and evaluate the RHSs of Eqs. (28), (29). The improved solution is obtained through Eqs. (28), (29). If the improved solution does not satisfy the accuracy requirement, we may consider the improved solution as an initial guess to continue the iteration. This primitive iterative approach is called the Gauss–Seidel approach. Numerical experiments show that this iterative procedure converges. We will show the convergence rate in the section of numerical examples.

4. MULTIGRID METHOD

In the Gauss–Seidel approach, information is carried over only one grid cell through each iteration, and therefore the convergence is very slow. In order to speed up the convergence, we divide all grid cells into two sets which are staggered with each other, called red and black sets. If we implement Eqs. (24), (25) for the red set first, then for the black set in each iteration, the number of iterations may be reduced to half for a given required accuracy because the information is carried over two cells through each iteration.

From numerical analysis and experiments for iteratively solving Eq. (1) for a constant thermal diffusivity, it is known that numerical errors with high frequencies are efficiently killed in the first few iterations, and errors with low frequencies remain even after many iterations (see, e.g., [21]). These phenomena indicate that we may use a coarse grid to kill low frequency errors and a fine grid to kill high frequency errors; i.e., we may use the multigrid method to speed up the convergence.

The multigrid method have been developed for many years. Fedorenko [4] and Bachvalov [5] formulated multigrid algorithms for the standard five point finite difference discretization for the Poisson equation and the general linear elliptic partial differential equations. The paper by Brandt [9] is one of the earliest in which practical results were reported. At first there was much debate and scepticism about the true merits of the multigrid method. This led researchers to the development of more transparent convergence proofs (see, e.g., [11, 16, 18] for a survey of theoretical development). Although rate of convergence proofs of the

multigrid method are complicated, their structure has now become more or less standard and transparent. The multigrid method for discontinuous coefficients in some situations have also been studied (see, e.g., [14]).

A typical algorithm of the multigrid method starts from a coarse grid. After a few iterations, the grid points in each direction are doubled, resulting in a fine grid. Then another set of a few iterations are implemented in the fine grid with the initial guess obtained from the solution on the coarse grid. Therefore, low frequency errors are significantly killed in the coarse grid, and high frequency errors are significantly killed in the fine grid. We should point out that, even if we start with a very accurate solution in the coarse grid, low frequency errors still remain after many iterations in the fine grid. One efficient approach for killing these remaining low frequency errors is to go back to the coarse grid and implement another set of a few iterations on the coarse grid. Since the coarse grid is more efficient for a smoother solution, we may work on the residue of the solution, instead of the solution itself on the coarse grid.

Suppose that C^N (or C^H) and R^N (or R^H) are the error and residue on the fine grid, i.e.,

$$C = T - T^{(e)}, \tag{30}$$

$$R^N \equiv T^N - T_0 - D^H, \tag{31}$$

$$R^H \equiv T^H - T_0 - \frac{3}{4}D^H + \frac{1}{4}D^N. \tag{32}$$

Here, $T^{(e)}$ is the exact solution of Eqs. (24), (25) on the fine grid. Then, we solve the following two equations for C^H and C^N on the coarse grid:

$$C^N + \beta_C^H C^H = Q_C^H + R^N, \tag{33}$$

$$-\frac{1}{4}\beta_C^N C^N + \left(1 + \frac{3}{4}\beta_C^H\right)C^H = \frac{3}{4}Q_C^H - \frac{1}{4}Q_C^N + R^H. \tag{34}$$

If the thermal diffusivity for each kind of material is a constant, then β_C^H, β_C^N, Q_C^H, and Q_C^N in Eqs. (33), (34) are exactly the same as β^H, β^N, Q^H, and Q^N in Eqs. (24), (25), except for T_L^H, T_R^H, T_B^H, and T_T^H in Eqs. (24), (25), which are replaced by C_L^H, C_R^H, C_B^H, and C_T^H in Eqs. (33), (34). A few iterations of Eqs. (33), (34) on the coarse grid will give very accurate solutions for C^N and C^H, since C^N and C^H are smooth and the vanishing C^N (or C^H) is a reasonable initial guess. After a few iterations for C^N and C^H on the coarse grid, we may correct the solution on the fine grid:

$$T^N = T^N - C^N, \tag{35}$$

$$T^H = T^H - C^H. \tag{36}$$

This procedure is often called a "coarse grid correction." We should mention that the solution after a coarse grid correction always contains large high frequency errors for which a few iterations on the fine grid are needed.

For nonlinear problems, α_L^H, $\tilde{\kappa}_L^H$, and g_L^H in β^H and Q^H depend on temperature T_L^H and T^H. Strictly speaking, Eqs. (33), (34) are no longer true for nonlinear problems. But, we should realize that a coarse grid is used only for killing low frequency errors, but not for the solution itself. After the coarse grid correction, we still have to go back to the fine grid for

the solution. Therefore, equations we use in the coarse grid for the coarse grid correction do not necessarily have to be exact, as long as the coarse grid correction may speed up the convergence. Therefore, even for nonlinear problems, we still use Eqs. (33), (34) for the coarse grid correction, in which α_L^H, $\tilde{\kappa}_L^H$, and g_L^H are all evaluated at the solution on the fine grid. Numerical experiments show that the coarse grid correction based on Eqs. (33), (34) works for nonlinear problems even for systems composed of the more than one kind of material. We will show the convergence rate in the next section.

In the multigrid method described above, we have to transfer cell-averages between fine and coarse grids. For the transformation from a fine grid to a coarse grid, by definition we only have to add appropriate cell-averages defined on the fine grid together in order to find cell-averages on a coarse grid. For the transformation of C^N and C^H from a coarse grid to a fine grid, the simplest interpolation is to assume no internal structures in the coarse grid; i.e., C^N and C^H are piecewise constants on the coarse grid. From this assumption, we may easily find the values of C^N and C^H on the fine grid, which are used in Eqs. (35), (36), from their values on the coarse grid. We may also assume a linear (or parabolic) internal structure of C^N and C^H on a coarse grid, and the internal structure may be determined by local information of C^N and C^H. The internal structure will result in more accurate estimates of C^N and C^H on the fine grid. But, the difference in the values of C^H and C^N on the fine grid between the two approaches, i.e. piecewise constant and piecewise linear (or parabolic), is dominated by high frequencies, and the high frequency difference will be immediately killed in the first few iterations on the fine grid. Therefore, there will be no difference in the convergence rate between the two approaches. We have tested both interpolations and found no difference in the convergence rate.

Now we would like to discuss the conservative feature. After finding a sufficiently accurate solution for T^H through the iterations described above, we substitute the solution into Eqs. (15), (16) to find the flux across each interface, q_{xW}^H and q_{yS}^H, through Eqs. (15), (16). Finally, we follow the conservation law, Eq. (5), to update the temperature.

5. NUMERICAL EXAMPLES

The scheme developed in the previous sections has been tested for some heat transfer problems, a few of which will be presented here to illustrate the features of the scheme. For the multigrid method, one iteration in a fine grid may need many iterations in coarse grids both for a better initial guess and for the coarse grid correction. Therefore, it is necessary to show the convergence rate in terms of both the number of iterations used in the fine grid and the actual CPU time used. In all numerical examples below, except the last one, the initial condition is a constant temperature $T(x, y) = 0.25$, simulations are performed on a 1×1 domain, 128×128 grid cells are used, and $T = 1$, $T = 2$, $T = 3$, and $T = 4$ are assigned at the left, top, right, and bottom boundaries, respectively, as boundary conditions, unless specified otherwise.

In order to demonstrate the correctness of our algorithm through a comparison with an exact solution, we choose a plate with temperature-dependent diffusivity $\kappa(T) = 1 + T$. The simulation domain is $0 < x < 1$, and the temperature is 0.1 and 1.0 at $x = 0$ and $x = 1$, respectively. The solid line in Fig. 1 is the exact solution of the steady state [2]. The points marked with "$\circ$," "$\times$," and "$+$" in Fig. 1 are our numerical solutions after one time step $\Delta t = 100$. The "$\circ$" points are obtained when eight grid points are used in the

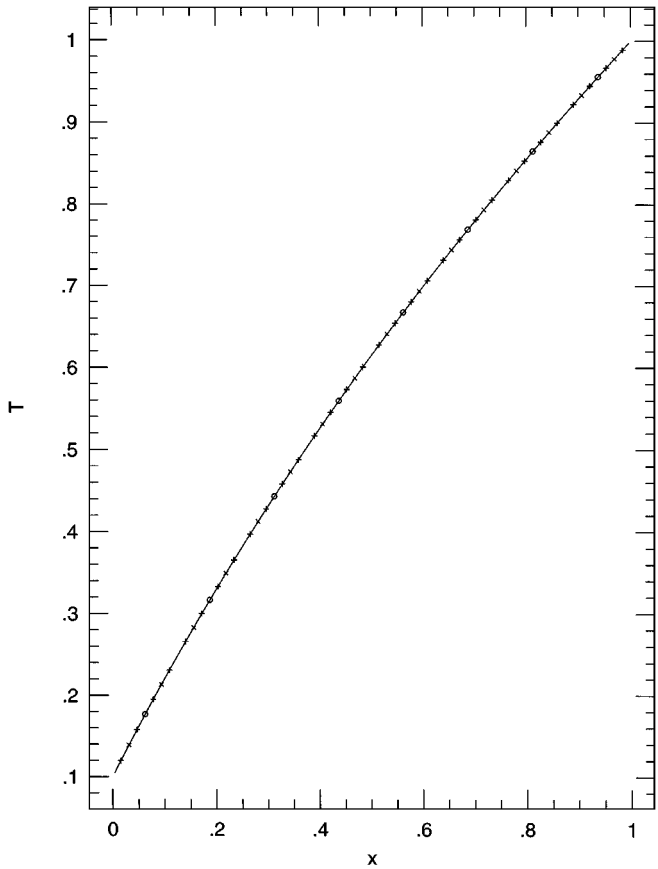

FIG. 1. A comparison between numerical solutions and the exact solution. The solid line is the exact solution. The "$\circ$" points are obtained when eight grid points are used in the range $0 < x < 1$, the "$\times$" points are obtained when 16 grid points are used, and the "$+$" points are obtained when 32 grid points are used.

range $0 < x < 1$, the "$\times$" points are obtained when 16 grid points are used, and the "$+$" points are obtained when 32 grid points are used. Our numerical solution has an excellent agreement with the exact solution.

The second example, in which the thermal diffusivity is a constant $\kappa = 1$, is to test the accuracy of the scheme. The dashed lines in Fig. 2 show four solutions of the temperature along the line $y = 0.5$ at $t = 0.1$ obtained for four different simulations. The four simulations are different only in the sizes of time step used, which are $\Delta t = 0.1, 0.05, 0.025, 0.0125$. The corresponding parameter, $2\kappa\Delta t/(\Delta x)^2$, in the four simulations is 3276.8, 1638.4, 819.2, and 409.6, respectively. The solid line in the figure is the solution obtained when the size of time step, $\Delta t = 0.001$, is used, which is considered as a reference. The dashed line obtained from a larger size time step is further away from the reference. Actually, the solutions obtained through $\Delta t = 0.025, 0.0125$ coincide with the reference.

As an example, we would like to show the convergence of numerical solutions for a nonlinear time-dependent problem, $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$. Figure 3 shows the profiles along $y = 0.5$ of the solution at $t = 0.05$ when different time steps are used. The solid line in Fig. 3 is obtained when the time step $\Delta t = 10^{-4}$, which is considered as an "exact solution,"

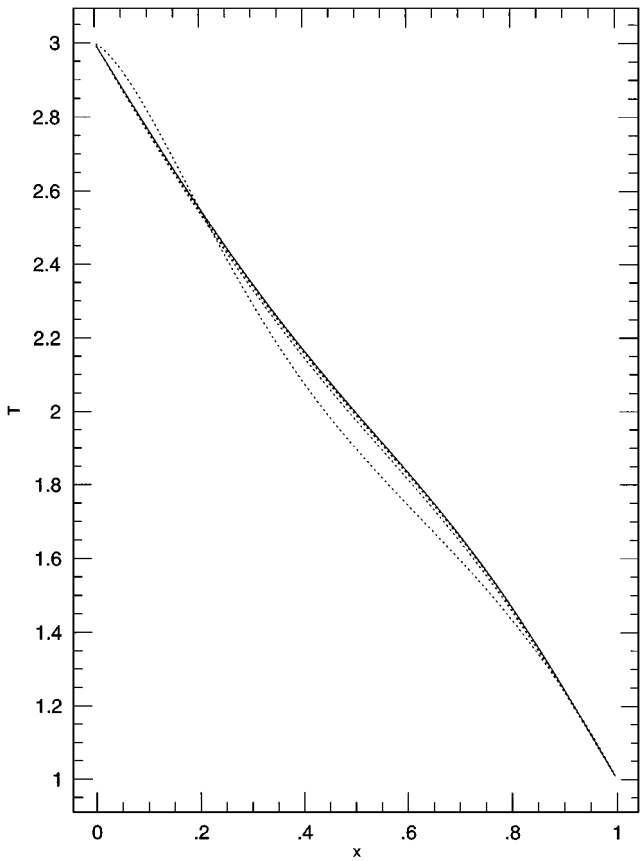

FIG. 2. The solution at $t = 0.1$ obtained from four simulations. The four simulations are different only in the size of time steps used, which are $\Delta t = 0.1, 0.05, 0.025$, and 0.0125. The solid line is considered an "exact" solution. The dashed line obtained from a larger time step is further away from the exact solution. The solutions obtained through $\Delta t = 0.025, 0.0125$ already coincide with the exact solution. The parameter, $2\kappa\Delta t/(\Delta x)^2$, in the four simulations is 3276.8, 1638.4, 819.2, and 409.6, respectively.

and two dashed lines in Fig. 3 are obtained when $\Delta t = 0.05, 0.025$ are used. The solution obtained when $\Delta t = 0.025$ almost coincides with the exact solution.

In the nest set of three examples, we would like to examine the convergence rate for the different approaches mentioned in the last section. We start with a linear problem $\kappa = 1$. The time step $\Delta t = 1.0$. The parameter $2\kappa\Delta t/(\Delta x)^2$ is 3.2768×10^4 in the simulation, and $2\kappa\Delta t/(\Delta x)^2 < 1$ is the stability requirement in the forward Euler scheme. The broken lines in Fig. 4 show the maximum residue, R^N and R^H, as a function of the number of iterations and the CPU time for the Gauss–Seidel method. It is clear that only the first few iterations are efficient in killing numerical errors, and the remaining errors need an extremely large amount of iterations. The dashed lines in Fig. 4 give the result obtained through the red–black method. As stated before, error information is carried over two cells in the red–black method, while it is carried over only one cell in the Gauss–Seidel method. Therefore, for a given maximum of residue, the number of iterations required in the red–black method is about a half of that required in Gauss–Seidel method. The solid lines in Fig. 4 are obtained through the multigrid method. Compared to the red–black and Gauss–Seidel methods, only

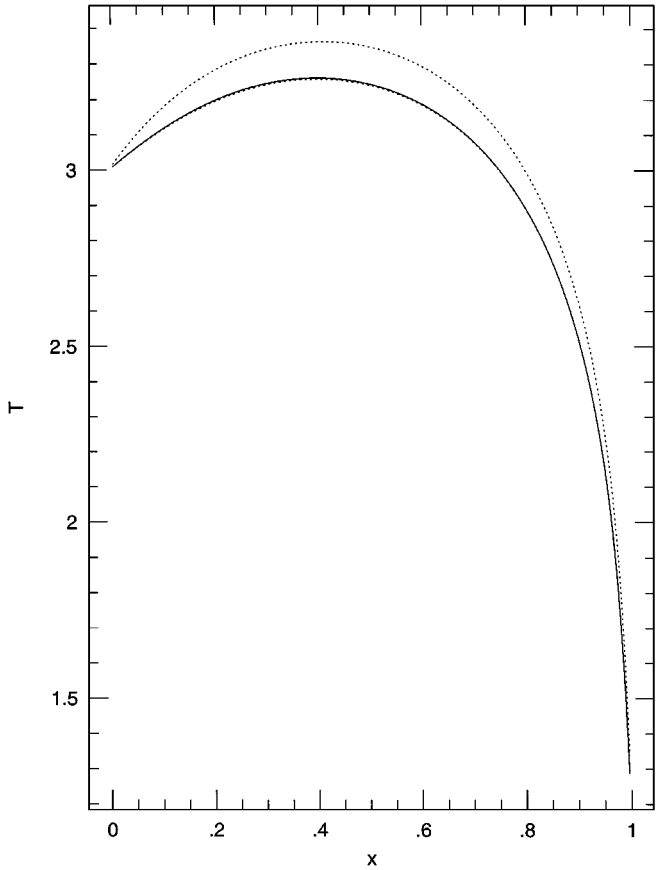

FIG. 3. A convergence study for $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$. The solid line is obtained when the time step $\Delta t = 10^{-4}$, which is considered as an "exact solution," and two dashed lines are obtained when $\Delta t = 0.05, 0.025$ are used.

a small number of iterations, or a small amount of CPU time is needed in the multigrid method. The unit used for the CPU time in Fig.4 is the CPU time used for one iteration in the red–black method.

The third example is a problem in which there are two kinds of material. In the region $0.25 \le x \le 0.75$ and $0.25 \le y \le 0.75$, $\kappa = 10^4$, while $\kappa = 1$ in the other part of the simulation domain. The time step $\Delta t = 0.02$. The maximum value of the parameter $2\kappa\Delta t/(\Delta x)^2$ is 6.5536×10^6 in the simulation. The dashed lines in Fig. 5 show the maximum of residue as a function of the number of iterations and CPU time used in the red–black method, and the solid lines in Fig. 5 are obtained through the multigrid method. Figure 6 is the temperature after one time step obtained from our scheme.

The fourth simulation is for a nonlinear problem $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$ for a single kind of material. The time step $\Delta t = 0.015$. The maximum value of the parameter $2\kappa(T^H)\Delta t/(\Delta x)^2$ is 3.28206×10^4 in the simulation. The lines in Fig. 7 show the maximum of residue as either the number of iterations or the CPU time used to reach the residue. The dashed lines are obtained from the red–black iteration, and the solid lines are obtained from the multigrid method. Figure 8 gives the solution after one time step.

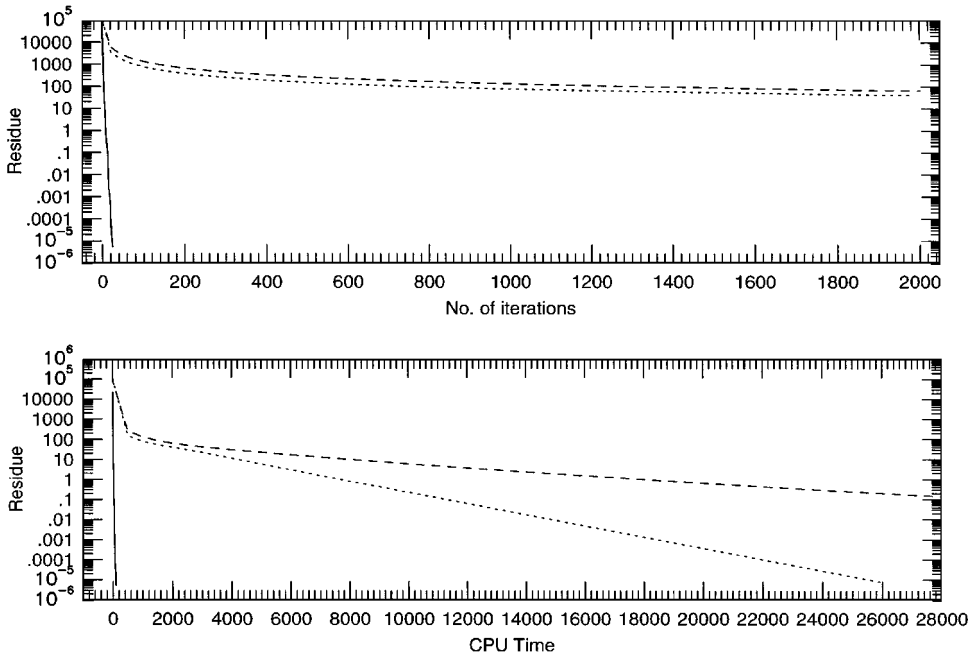

FIG. 4. The convergence rate for a linear problem. $\Delta t = 1.0$. $2\kappa\Delta t/(\Delta x)^2$ is 3.2768×10^4. The broken, dashed, and solid lines are respectively obtained from Gauss–Seidel, the red–black, and the multigrid methods. The CPU time is measured in terms of the CPU time used for one iteration in the red–black method.

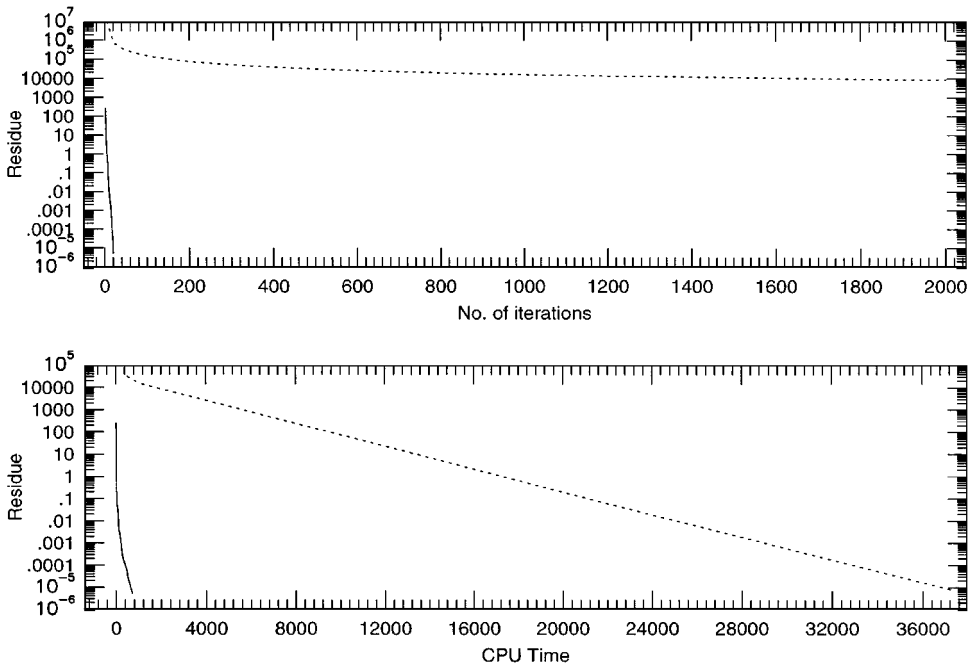

FIG. 5. The convergence rate for a system composed of two kinds of material. $\kappa = 10000$ in the region $0.25 \le x \le 0.75$ and $0.25 \le y \le 0.75$, and $\kappa = 1$ in the other region. $\Delta t = 0.02$. The maximum value of $2\kappa\Delta t/(\Delta x)^2$ is 6.5536×10^6. The dashed and solid lines are respectively obtained from the red–black and multigrid methods.

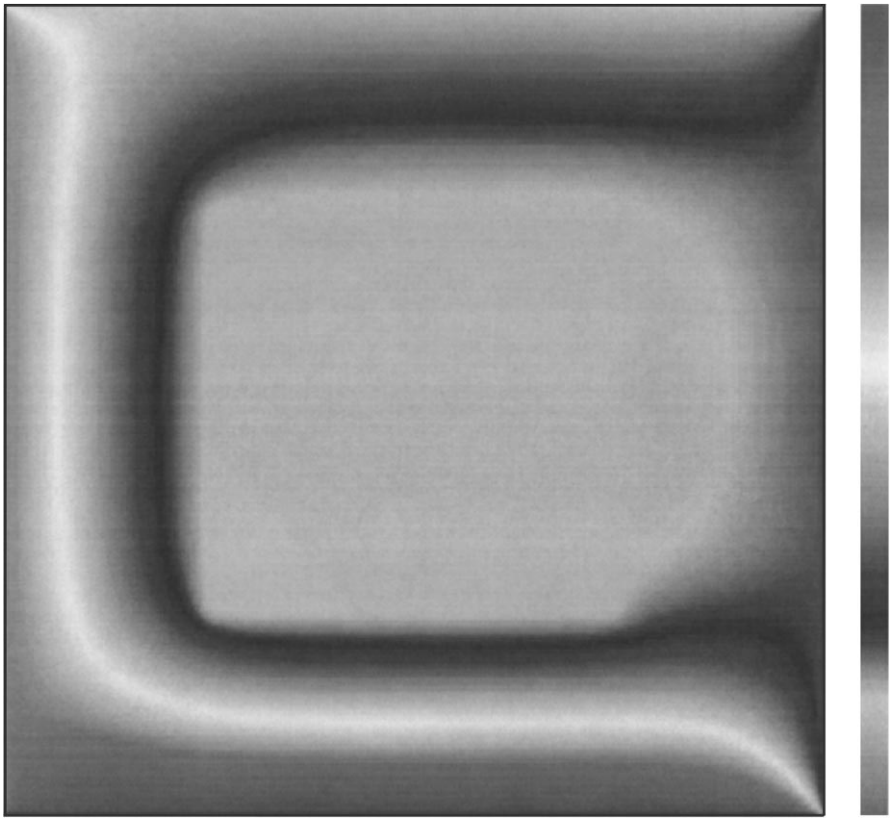

FIG. 6. The solution for temperature in a system composed of two kinds of material with $\kappa = 1$, and $\kappa = 10{,}000$ after one time step $\Delta t = 0.02$.

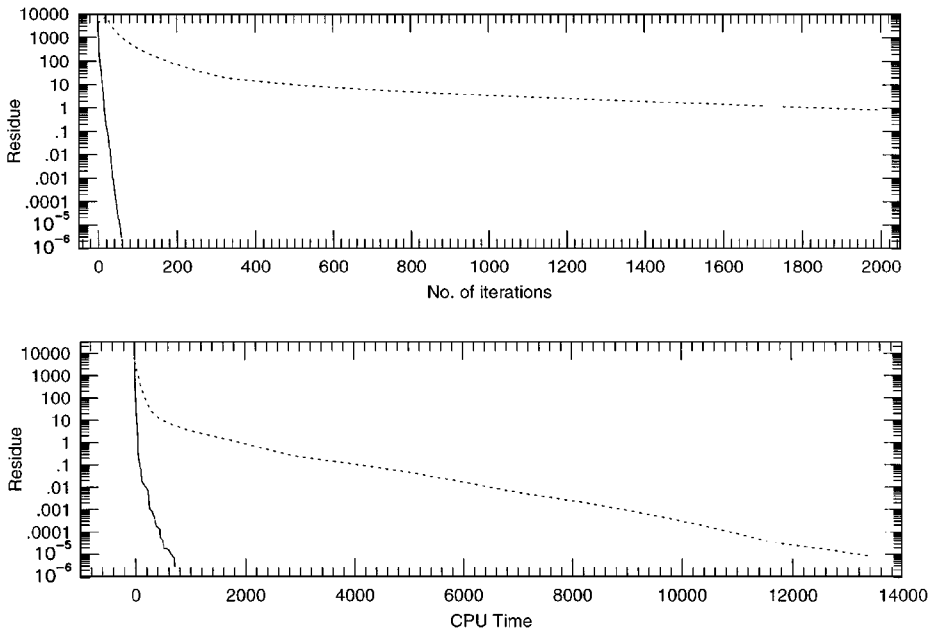

FIG. 7. The convergence rate for a nonlinear problem: $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$, $\Delta t = 0.015$. The maximum value of $2\kappa\Delta t/(\Delta x)^2$ is 3.28206×10^4. The dashed and solid lines are respectively obtained from the red–black and multigrid methods.

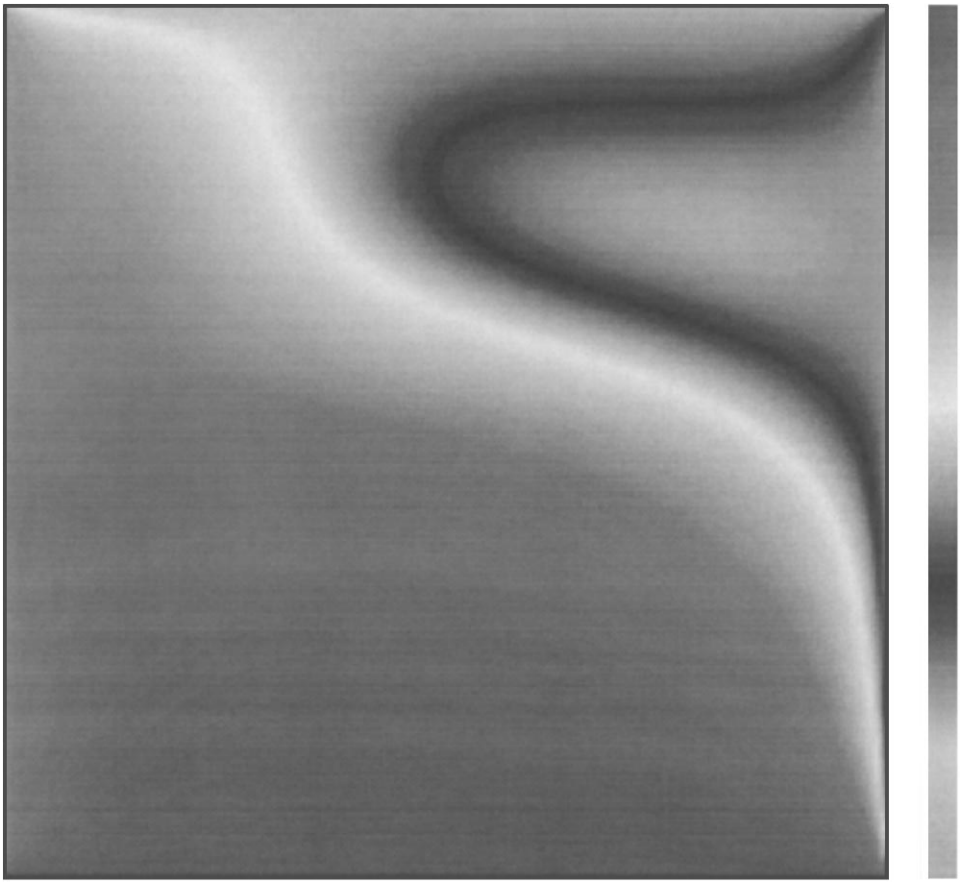

FIG. 8. The solution for temperature for a nonlinear problem: $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$ after one time step $\Delta t = 0.015$.

The fifth simulation is for a nonlinear problem in which there are two kinds of material. In the inner part of the simulation domain, $0.25 < x < 0.75$ and $0.25 < y < 0.75$, $\kappa(T) = 10^4 + 100T^2$, and in the outer part $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$. The time step is $\Delta t = 0.01$. The maximum value of the parameter $2\kappa(T^H)\Delta t/(\Delta x)^2$ is 3.37728×10^6 in the simulation. The lines in Fig. 9 show the maximum residue as a function of either the number of iterations or the CPU time used to reach the residue. The dashed lines are obtained from the red–black method, and the solid lines are obtained from the multigrid method. Figure 10 gives the solution after one time step.

Our last example is for the heating of a nonlinear system which is composed of three kinds of material. The simulation is performed in a 1×1 domain containing 256×256 grid cells. In the inner part, $0.375 < x < 0.625$ and $0.375 < y < 0.625$, $\kappa(T) = 10^8 + 100\,T^2$; in the middle part, $0.25 < x < 0.75$ and $0.25 < y < 0.75$ outside the inner part, $\kappa(T) = 10 + 0.1T^8$; and in the remaining outer part, $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$. The initial temperature is 0.1. A constant heating $dT/dl = 5$ is imposed on four boundaries of the simulation domain. Here l is a space coordinate normal to a boundary. The size of time steps is $\Delta t = 10^{-4}$, and the maximum value of the parameter $2\kappa\Delta t/(\Delta x)^2$ in each time step is about 1.31×10^9. Figure 11 gives the temperature at four instants. Since the thermal

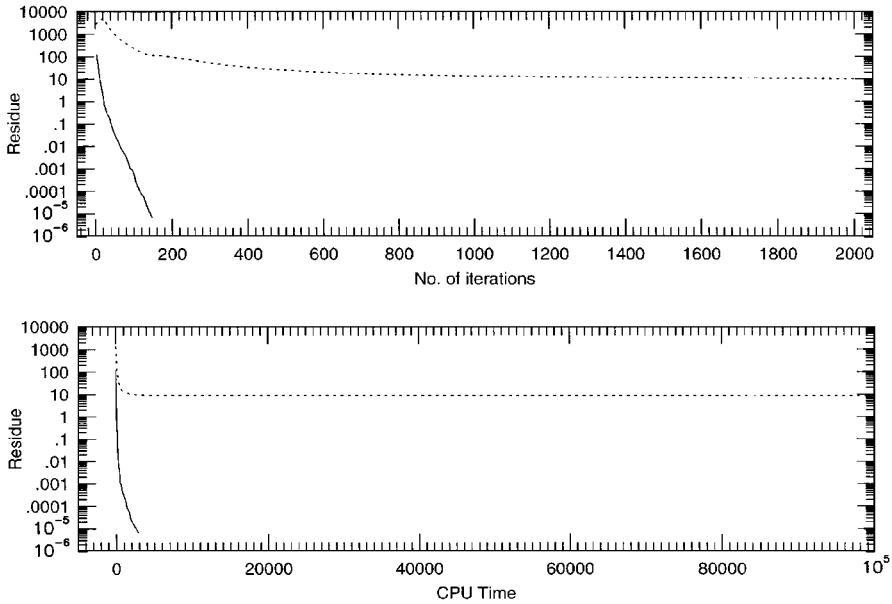

FIG. 9. The convergence rate for a system composed of two kinds of material, each of which has a temperature-dependent thermal diffusivity. $\kappa(T) = 10^4(1 + 0.01T^2)$ in the center $0.25 < x < 0.75$ and $0.25 < y < 0.75$, $\kappa(T) = 1, +0.1T^4, +0.01T^6$ in the remaining part, $\Delta t = 0.01$. The maximum value of $2\kappa\Delta t/(\Delta x)^2$ is 3.37728×10^6. The dashed and solid lines are respectively obtained from the red–black and multigrid methods.

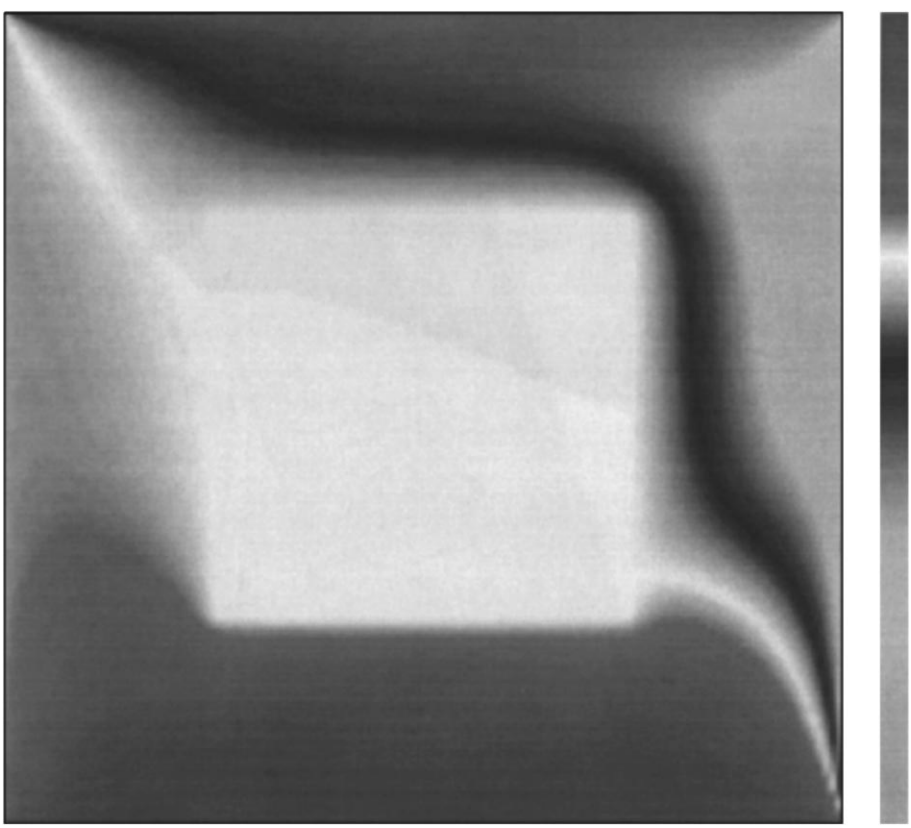

FIG. 10. The solution for temperature for a system composed of two kinds of material after one time step $\Delta t = 0.01$; $\kappa(T) = 10^4(1+0.01T^2)$ in the center $0.25 < x < 0.75$ and $0.25 < y < 0.75$, and $\kappa(T) = 1+0.1T^4+0.01T^6$ in the remaining part.

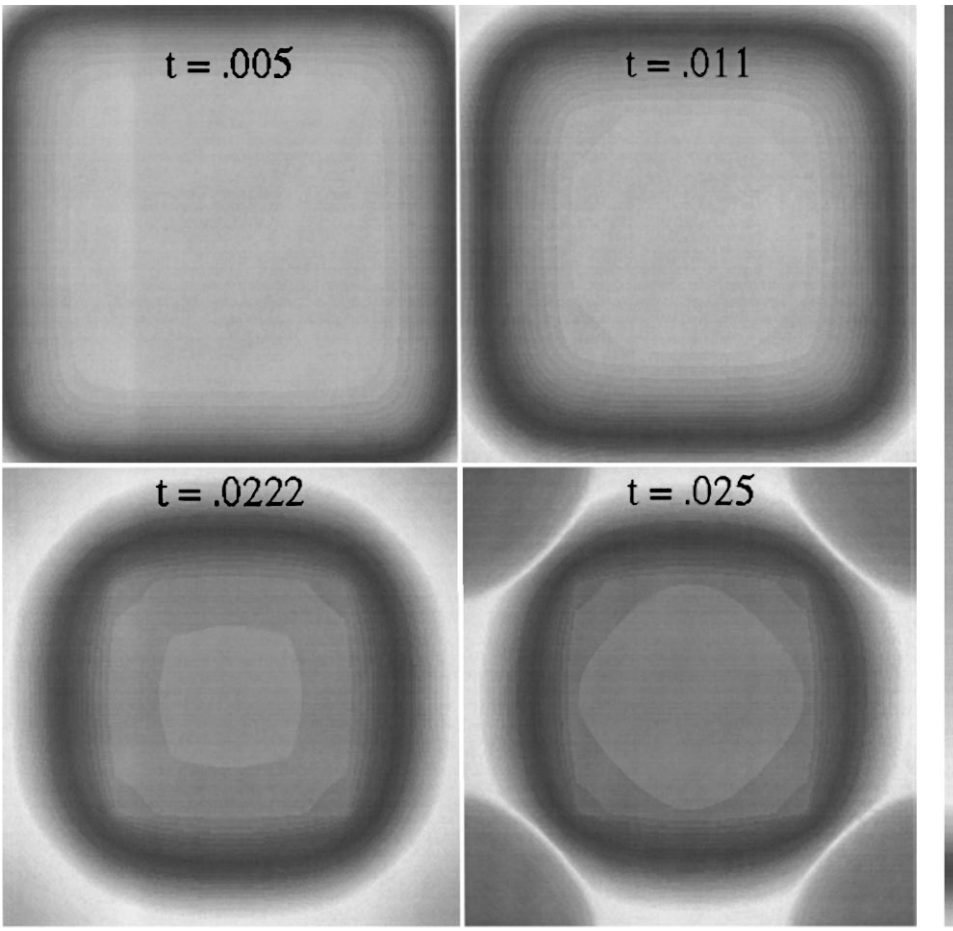

FIG. 11. The temperature at four instants under a constant heating from four boundaries. The thermal diffusivities in the inner, middle, and outer regions in the simulation domain are, respectively, $\kappa(T) = 10^8 + 100T^2$, $\kappa(T) = 10 + 0.1T^8$, and $\kappa(T) = 1 + 0.1T^4 + 0.01T^6$. The maximum value of the parameter $2\kappa\Delta t/(\Delta x)^2$ in each time step is about 1.31×10^9.

diffusivity in the inner part is very high, the time step ($=10^{-4}$) is so large that the temperature is almost uniform in the inner region after each time step. In order to display the structure near interfaces between two kinds of material, we give Fig. 12, which shows the function $\tanh(\xi)$. Here $\xi \equiv 10(T - 0.2538)$. We would like to point out that the size of the time step in any explicit scheme is limited by the largest value of $\kappa(T)$ in the simulation domain, no matter whether or not the region with largest $\kappa(T)$ has been influenced yet.

6. CONCLUSIONS AND DISCUSSIONS

In this paper we have developed a finite difference scheme for nonlinear heat transfer. We have demonstrated that the scheme provides goods results in a wide variety of situations. Accurate solutions can be obtained for time-dependent problems as well as steady states. The scheme possesses a number of features: second-order accurate in both space and time, iterative and fast in convergence, conservative, accurate in the treatment of nonlinearities

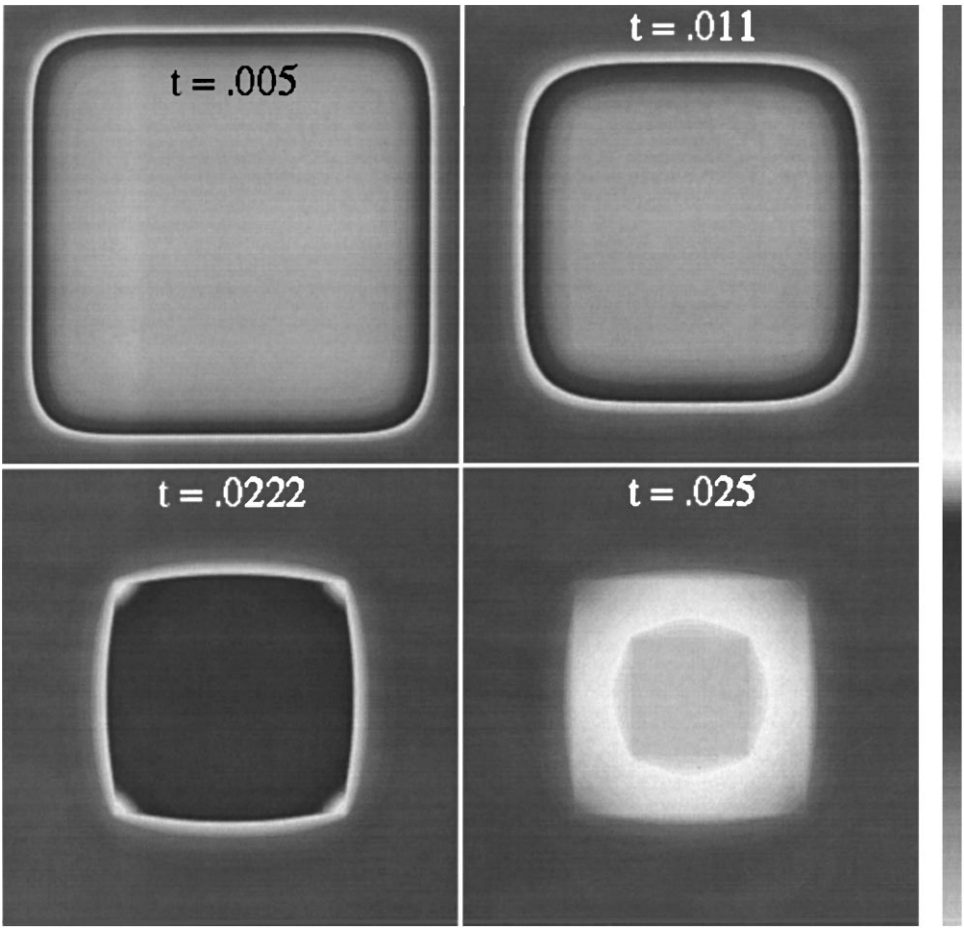

FIG. 12. The function of $\tanh[10(T - 0.2538)]$ at the four instants.

and multikinds of material, able to quickly damp numerical errors for large time steps, and capable of solving a system composed of more than one kind of material. The algorithm of the scheme may be easily vectorized. The scheme may be used to study the heat transfer problems involved in laser fusion, and the approach developed in this paper may be applied to the radiative heat transfer in radiative hydrodynamics.

A more general equation for heat transfer, i.e.,

$$\rho c_p(T)\frac{\partial T}{\partial t} - \nabla \cdot [\lambda(T)\nabla T] = \hat{s}(T), \tag{37}$$

may be changed into the form

$$\frac{\partial E(T)}{\partial t} - \nabla \cdot [\lambda(T)\nabla T] = \hat{s}(T). \tag{38}$$

In Eq. (37), ρ is a mass density of a material, c_p is a specific heat and λ is a thermal conductivity. In Eq. (38) $E(T)$ is the integral of $\rho c_p(T)$. If we write $E(T)$ as $E(T) =$

$\sigma(T)T$, from Eq. (38), we have

$$\sigma^N T^N + \beta^H T^H = T_0 + Q^H, \tag{39}$$

$$-\frac{1}{4}\beta^N T^N + \left(\sigma^H + \frac{3}{4}\beta^H\right) T^H = T_0 + \frac{3}{4}Q^H - \frac{1}{4}Q^N. \tag{40}$$

Here $\sigma^N \equiv \sigma(T^N)$ and $\sigma^H \equiv \sigma(T^H)$. Equations (39), (40) are exactly the same as Eqs. (24), (25), except for σ^N and σ^H which are replaced by a unity in Eqs. (24), (25). Equations (39), (40) may be similarly treated as Eqs. (24), (25), and the multigrid method may be implemented for Eqs. (39), (40), too.

ACKNOWLEDGMENTS

The work presented here has been supported by the Department of Energy through Grants DE-FG02-87ER25035 and DE-FG02-94ER25207, and by the University of Minnesota through its Minnesota Supercomputer Institute.

REFERENCES

1. J. Crank and P. Nicolson, A practical method for numerical evaluation of solution of partial differential equations of the heat conduction type, *Proc. Camb. Phil. Soc.* **43**, 50 (1947).
2. M. Jakob, *Heat Transfer*, Vol. I (Wiley, New York, 1949).
3. W. Wasow and G. Forsythe, *Finite-difference Methods for Partial Differential Equations* (Wiley, New York, 1960).
4. R. P. Fedorenko, The speed of convergence of one iterative process, *USSR Comput. Math. and Math. Phys.* **4**, 227 (1964).
5. N. S. Bachvalov, On the convergence of a relaxation method with natural constraints on the elliptic operator, *USSR Compt. Math. and Math. Phys.* **6**, 101 (1966).
6. M. Lees, A linear three-level difference scheme for quasi-linear parabolic equations, *Math. Comput.* **20**, 516 (1966).
7. J. Ortega and W. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (McGraw-Hill, New York, 1970).
8. D. M. Young, *Iterative Solution of Large Linear Systems* (Academic Press, New York, 1971).
9. A. Brandt, Multi-level adaptive solution technique (MLAT) for fast numerical solution to boundary value problems, in *Lecture Notes in Physics*, Vol. 18 (Springer-Verlag, Berlin, 1973), p. 82.
10. T. Dupont, G. Fairweather, and J. P. Johnson, Three-level Galerkin methods for parabolic equations, *SIAM J. Numer. Anal.* **11**, 392 (1974).
11. R. A. Nicolaides, On multiple grid and related techniques for solving discrete elliptic systems, *J. Comput. Phys.* **19**, 418 (1975).
12. A. Brandt, Multi-level adaptive solution to boundary value problems, *Math. Comput.* **31**, 333 (1977).
13. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, (Hemisphere, New York, 1980).
14. R. E. Alcouffe, A. Brandt, J. J. E. Dendy and J. W. Painter, The multigrid method for the diffusion equation with strongly discontinuous coefficients, *SIAM J. Sci. Stat. Comput.* **2**, 430 (1981).
15. M. A. Hogge, A comparison of two- and three-level integration schemes for non-linear heat conduction, in *Numerical Methods in Heat Transfer* Eds. R. W. Lewis, K. Morgan, and O. C. Zienkiewicz (John Wiley, New York, 1981), p. 75.
16. P. Wesseling, Theoretical and practical aspects of a multigrid method, *SIAM J. Sci. Comput.* **3**, 387 (1982).
17. D. Mihalas and B. W. Mihalas, *Foundations of radiation hydrodynamics*, (Oxford Univ. Press, New York, 1984).
18. W. Hackbusch, *Mult-grid methods and applications* (Springer, Berlin, 1985).

19. J. Lindl, Development of the indirect-drive approach to inertial confinement fusion and the target physics basis for ignition and Gain, *Phys. Plasmas* **2**, 2933 (1995).
20. B. A. Fryxell, P. R. Woodward, P. Colella, and K.-H. Winkler, An implicit-explicit hybrid method for Lagrangian hydrodynamics, *J. Comput. Phys.* **63**, 283 (1986).
21. Y. Jaluria and K. E. Torrance, *Computational Heat Transfer* (Spring-Verlag, Berlin, 1986).
22. S. F. McCormick, *Multigrid methods (Frontiers in Applied Mathematics)*, Vol. 3 (SIAM, Philadelphia, 1989).
23. A. A. Samarskii and E. S. Nikolaev, *Numerical Methods for Grid Equations* (Birkhauser-Verlag, Basel, 1989).
24. M. N. Özisik, *Finite Difference Methods in Heat Transfer* (CRC Press, Boca Raton, FL, 1994).
25. W. Dai and P. R. Woodward, Iterative implementation of an implicit-explicit hybrid scheme for hydrodynamics, *J. Comput. Phys.* **124**, 217 (1996).