JOURNAL OF COMPUTATIONAL PHYSICS 98, 54–63 (1992)

Adaptive Local Overlapping Grid Methods for Parabolic Systems in Two Space Dimensions*

PETER K. MOORE

Tulane University

AND

JOSEPH E. FLAHERTY

Rensselaer Polytechnic Institute

Received February 23, 1990; revised January 27, 1991

Adaptive mesh refinement techniques are described for two-dimensional systems of parabolic partial differential equations. Solutions are calculated using Galerkin's method with a piecewise bilinear basis in space and backward Euler integration in time. A posteriori estimates of the local discretization error of piecewise bilinear finite element solutions are obtained by a p-refinement technique. These error estimates are used to control a local h-refinement strategy where finer grids are recursively introduced in regions where a prescribed tolerance is exceeded. Fine grids at a given level of refinement may overlap each other and independent solutions are generated on each of them. A version of the Schwarz alternating principle is used to coordinate solutions between overlapping fine grids. Computational results demonstrating the performance of the adaptive procedure on linear and nonlinear problems and apparent convergence of the error estimate for linear heat conduction problem and uniform global refinement is presented.

1. INTRODUCTION

A variety of strategies have been developed for the adaptive solution of parabolic partial differential equations. Techniques include the method of lines (MOL) with spatial mesh (h-refinement) [1, 2, 8, 9], mesh motion (r-refinement) [6, 12, 14, 16, 24, 26], combinations of the two [3, 4], and spatial order enrichment (p-refinement) [18]. Local refinement methods (LRM), where problems are solved on local space-time meshes, have been developed using h-refinement [21, 22]. A number of overlapping grid strategies, including domain decomposition methods, have been successively applied to elliptic problems [13, 19], but have not been used widely for parabolic systems. This paper explores a local overlapping grid h-refinement solution technique with a p-refinement error estimation strategy.

Consider the solution of a two-dimensional system of m parabolic partial differential equations having the form

$$\mathbf{u}_t + \mathbf{f}(\mathbf{x}, t, \mathbf{u}, \nabla\mathbf{u}) = \nabla^T[\mathbf{D}(\mathbf{x}, t, \mathbf{u})\nabla\mathbf{u}], \quad \mathbf{x}\in\Omega, t>0, \tag{1.1a}$$

with initial and boundary conditions

$$\mathbf{u} = \mathbf{u}^0(\mathbf{x}), \quad \mathbf{x}\in\Omega\cup\partial\Omega, t>0, \tag{1.1b}$$

and, respectively,

$$\begin{aligned} \mathbf{u} &= \mathbf{g}^E, \quad \mathbf{x}\in\partial\Omega^E, \\ \mathbf{D}\mathbf{u}_\nu &= \mathbf{g}^N, \quad \mathbf{x}\in\partial\Omega^N, t>0, \end{aligned} \tag{1.1c}$$

on the rectangular region $\Omega = \{(x, y) \mid a<x<b, c<y<d\}$. Essential (Dirichlet) or natural (Neumann) boundary conditions are prescribed on portions of the boundary $\partial\Omega^E$ or $\partial\Omega^N$, respectively. Subscripts x, y, t denote partial differentiation and ν denotes the unit outer normal to the boundary $\partial\Omega = \partial\Omega^E \cup \partial\Omega^N$, of Ω.

Adaptive methods for partial differential equations are feedback systems that typically proceed by calculating a preliminary solution on a coarse discretization of Ω. "Refinement indicators" are used to identify portions of the domain in need of additional resolution. One or more of the three enrichment strategies are used to alter discretization and calculate an improved solution. As noted, most

* This research was partially supported by the U.S. Air Force Office of Scientific Research, Air Force Systems Command, USAF, under Grant AFOSR 85-0156; by the SDIO/IST under management of the U.S. Army Research Office under Contract DAAL 03-86-K-0112; and by the National Science Foundation under Grant IRI-8712838. (A portion of this work was used to satisfy the Ph.D. requirements of Peter K. Moore at the Rensselaer Polytechnic Institute.)

0021-9991/92 $3.00

adaptive strategies for parabolic problems have relied on h- and/or r-refinement.

Our procedure for solving (1.1) involves the use of piecewise bilinear finite element approximations and backward Euler integration in time (cf. Section 3). Refinement indicators are based on estimates of the local error obtained by p-refinement using piecewise second-order serendipity polynomials in space and the trapezoidal rule in time. Superconvergence properties of the finite element method reduce the computational overhead associated with this method. The resulting nonlinear systems for the solution and refinement indicators are solved using Newton's method with either a preconditioned conjugate gradient method or the incomplete orthogonalization method.

The adaptive LRM, described in Section 2, consists of initially solving (1.1) on a uniform *base* grid for one time step. An element-wise error indicator is used to identify those elements having "high error" and group them into rectangular regions called *megagrids*. As shown in Fig. 1, overlapping fine uniform grids are generated within each megagrid and (1.1) is solved again on these grids. This refinement process is repeated until a prescribed local error tolerance is satisfied. A tree is a natural data structure to manage the information associated with all of the grids. Nodes of the tree represent data at the megagrid level, with finer megagrids regarded as offspring of coarser ones. Information associated with overlapping fine grids within each megagrid is stored as records at the nodes of the tree.

A finite element problem is formulated and solved on each grid within a megagrid. This necessitates the prescription of appropriate initial and boundary conditions on each space-time grid. Since our temporal integration is implicit, prescribing boundary conditions is particularly complex in regions where meshes overlap. An iterative procedure, analogous to Schwarz alternation (cf. Dihn *et al.* [13]), is used to successively calculate solutions on fine grids within each megagrid.

Three computational examples are presented in Section 4. A linear example indicates that the a posteriori error estimate, used to control the refinement process, converges in energy to the exact local error. Convergence of a similar error estimate has been proven for one-dimensional problems [21]. Other examples demonstrate the effectiveness of our method on realistic nonlinear problems.

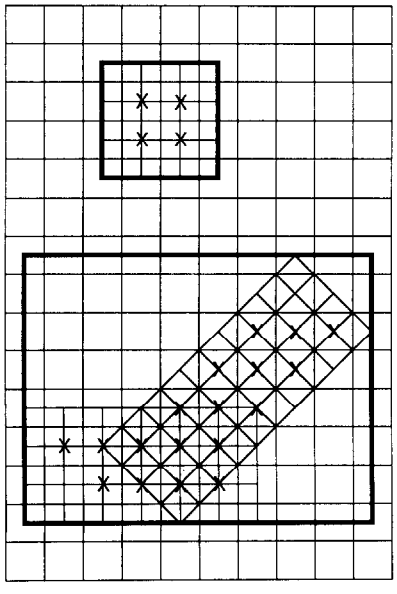

FIG. 1. Base spatial background grid with two offspring megagrids (marked with bold lines) and their local rectangles. High error elements of the base mesh are indicated by × 's.

2. LOCAL REFINEMENT STRATEGY

The LRM finite element procedure for solving (1.1) is most conveniently described at the megagrid level. Thus, let $R(\omega, p, q, F, S, L)$ denote an arbitrary space-time hexahedral megagrid, where $\omega := \{(x, y) \mid \alpha < x < \beta, \gamma < y < \delta\}$ is the spatial domain; p and q are times at the beginning and end of the time step, respectively; F and S point to parent and offspring megagrids of R, respectively; and L is a record of information associated with fine grids contained within R. For each *base* time step $[t_k, t_{k+1}]$, $k = 1, 2, \ldots$, initial solutions are obtained on a *base megagrid* $R(\Omega, t_k, t_{k+1}, 0, S, L)$ which contains one local grid L occupying the whole of Ω. Spacing of the base megagrid is prescribed and is the coarsest spacing permitted by the LRM procedure.

A top level description of a recursive LRM algorithm for solving (1.1) on $R(\omega, p, q, F, S, L)$ is presented in Fig. 2 using a pseudo-PASCAL language. A solution and error indicators are generated on R using **procedure** solve (cf. Fig. 4). Refinement indicators are used to isolate those elements having local errors that exceed a prescribed tolerance and group them into other rectangular megagrids which are regarded as the offspring of R in a tree structure.

```
procedure lrm (R, tol);
  begin
    solve (R(ω,p,q,F,S,L));
    if any error indicator > tol then
      begin
        Form offspring megagrids;
        for j := 1 to number of offspring do
          begin
            Create local rectangular grids;
            Calculate the temporal refinement factor tref[j];
            for i := 1 to tref[j] do
              begin
                p[i] := p+(i-1)*(q-p)/tref[j];
                q[i] := p[i] +(q-p)/tref[j];
                lrm (R(ω[j],p[i],q[i],R(ω,p,q,F,S,L),S[j],L[j]),tol)
              end
          end
      end
  end { lrm };
```

FIG. 2. Recursive local refinement algorithm for the solution of (1.1) on megagrid $R(\omega, p, q, F, S, L)$ to an error tolerance *tol*.

Pointers F and S simplify traversal of the tree and are used to pass information between megagrids. Sibling megagrids are isolated from each other and have edges that are parallel to the global coordinate axes (cf. Fig. 1). Overlapping fine uniform *local grids* are created within each megagrid, temporal refinement factors are selected, and finite element solutions are generated on each local grid. The recursive procedure continues to generate megagrids having finer structures until either the prescribed tolerance is satisfied or resource limitations are exceeded. In order to solve (1.1), **procedure** lrm is invoked on the base grids $R(\Omega, t_k, t_{k+1}, 0, S, L)$, $k = 0, 1, \ldots$.

Megagrids are constructed using the nearest neighbor clustering algorithm. For each megagrid, a second *local rectangle* containing the clustered high-error elements is created by envoking Berger and Oliger's [7] procedure. This local rectangle may possibly be skewed with respect to the coordinate axes. If the rectangle intersects the parent megagrid boundaries, it is replaced by a rectangle whose sides are parallel to the domain boundaries (and hence the megagrid boundaries). If the rectangle contains too few high-error elements it is bisected along its major axis, partitioning the elements into two sets (cf. Fig. 3). Two new local rectangles are created, one for each set. If they are each more efficient than the first rectangle they replace it, otherwise the initial larger rectangle is used. When two local rectangles are produced the bisection algorithm is applied to both of them, with the exception that any new local rectangles created by splitting are aligned with the domain boundaries. Thus, a maximum of four rectangles are created for each megagrid.

Uniform grids, enlarged on each edge by a λ-element buffer, are placed on each local rectangle to form a local grid. Local grids within a megagrid can overlap each other, but sibling megagrids are independent. The independence of megagrids at the same tree level further allows us to use different spatial and temporal refinement factors on different megagrids. This strategy reduces communication, which simplifies the computation of initial conditions on offspring megagrids, and should improve performance on parallel computers.

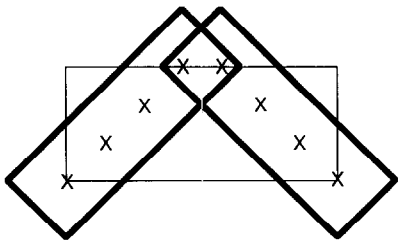

FIG. 3. A larger (thin-lined) rectangle containing too small a percentage of high-error elements (identified by $\times$s) is bisected along its major axis creating two smaller local rectangles (thick-lined) having a higher percentage of high-error elements.

Initial and boundary conditions are needed by the finite element solution procedure solve (cf. Fig. 4) for each local grid T_i within R. An initial guess for the boundary conditions on ∂T_i is obtained by using either the prescribed data on $\partial\Omega$ or by creating Dirichlet conditions using interpolation in time from a parent megagrid. This prescription is not appropriate in regions where local grids overlap; and instead, we use Schwarz alternation commonly used for elliptic problems [13]. Thus, a solution obtained on one local grid is used to update boundary data on all other intersecting local grids. This strategy is repeated on each local grid in turn until satisfactory accuracy is attained. Generally, three Schwarz iterations have been found to produce satisfactory convergence to the piecewise bilinear finite element solution. Oliger *et al.* [23] have provided a convergence analysis and acceleration methods for the Schwarz alternation for elliptic problems. Lions [20] proved the convergence of the Schwarz iteration in L^2 for linear parabolic problems, and our experimental indications imply robust convergence for nonlinear problems.

In order to prevent a loss of resolution, initial conditions on local grid T_i at $t = p^+$ are calculated from the finest scale information available $t = p^-$. The searching process is greatly simplified by the use of megagrids which limit the search to local grids within those megagrids that intersect R at $t = p^-$. The first step of the search consists of creating a list of those megagrids at $t = p^-$ which overlap R. This list of megagrids is traversed to find those local grids that intersect T_i. These grids are ordered by level into a set of lists. Finally, initial data for each element of T_i is obtained by piecewise bilinear interpolation using data from those elements in the set of lists at the highest available levels (i.e., from the finest grids at $t = p^-$). Search efficiency is improved by moving the grid in which the element is found to the head of its list, thus taking advantage of the convexity of grids and locality. Similar procedures are also used in finding internal boundary data.

The megagrid data structure greatly simplifies restarting the integration at any base time step and production of data

```
procedure solve (R);
  begin
    for i := 1 to number of local grids do
      begin
        Compute initial conditions for local grid T_i;
        Compute boundary conditions for T_i
      end
    repeat
      for i := 1 to number of local grids do
        begin
          Solve the finite element problem (3.4) for (1.1) on T_i;
          Update appropriate boundary conditions
        end
    until Schwarz iteration converges;
    for i = 1 to number of local grids do Compute error on T_i
  end { solve };
```

FIG. 4. Solution algorithm on megagrid $R(\omega, p, q, F, S, L)$.

for graphical display. Interpolation inherent in the finite element solution can be used to generate data for any uniform grid which can be used either for graphics or for generating a different base mesh.

3. DISCRETIZATION AND ERROR ESTIMATION

The adaptive LRM described in Section 2 introduces uniform overlapping rectangular grids having various orientations; however, we describe the finite element procedure for the time step $[p, q]$ on a uniform local grid Δ_T that is aligned with the global (x, y) coordinate axes, i.e., on

$$T = \{(x, y) \mid \xi < x < \eta, \sigma < y < \tau\}. \tag{3.1}$$

Rotated rectangular grids are obtained using well-known coordinate transformations.

The Galerkin form of (1.1) consists of determining $\mathbf{u}(\mathbf{x}, t) \in H_E^1(T)$ on $p < t < q$ such that

$$(\mathbf{v}, \mathbf{u}_t) + (\mathbf{v}, \mathbf{f}) + A(\mathbf{v}, \mathbf{u}) = \langle \mathbf{v}, \mathbf{D}\mathbf{u}_\nu \rangle, \quad \text{for all} \quad \mathbf{v} \in H_0^1, \tag{3.2a}$$

where

$$(\mathbf{v}, \mathbf{u}) = \int_T \mathbf{v}^T \mathbf{u} \, d\,|\mathbf{x}|, \tag{3.2b}$$

$$A(\mathbf{v}, \mathbf{u}) = \int_T \nabla \mathbf{v}^T \mathbf{D} \nabla \mathbf{u} \, d\,|\mathbf{x}| \tag{3.2c}$$

and

$$\langle \mathbf{v}, \mathbf{u} \rangle = \int_{\partial T} \mathbf{v}^T \mathbf{u} \, ds. \tag{3.2d}$$

The symbol $d\,|\mathbf{x}|$ denotes an area element and H^1 denotes the usual Sobolev space, with the subscript E restricting functions to satisfy any imposed Dirichlet conditions on ∂T and the subscript 0 restricting functions to satisfy trivial versions of any applied Dirichlet conditions on ∂T. Initial conditions at $p = 0$ are obtained by, e.g., strain energy projection

$$A(\mathbf{v}, \mathbf{u}(\cdot, 0)) = A(\mathbf{v}, \mathbf{u}^0). \tag{3.2e}$$

Finite element solutions $\mathbf{U}(\mathbf{x}, t)$ of (3.2) are obtained by introducing a uniform grid Δ_T of rectangular elements on T and approximating H^1 by a finite dimensional subspace $S_{\Delta_T}^{1,k}$ consisting of a tensor product of kth degree piecewise one-dimensional polynomials with respect to Δ_T. Specifically we determine $\mathbf{U} \in S_{\Delta_T}^{1,1}$ on $p < t < q$ by

$$(\mathbf{V}, \mathbf{U}_t) + (\mathbf{V}, \mathbf{f}) + A(\mathbf{V}, \mathbf{U}) \quad \langle \mathbf{V}, \mathbf{D}\mathbf{U}_\nu \rangle, \quad \text{for all} \quad \mathbf{V} \in S_{\Delta_T}^{1,1}. \tag{3.3}$$

As noted, initial conditions are obtained by interpolation using either the initial data at $p = 0$ or the best available solution at $t = p^- > 0$.

The system of ordinary differential equations (3.3) is discretized in time using the backward Euler method which yields an approximation $\mathbf{U}^q$ of $\mathbf{U}(\mathbf{x}, q)$ as

$$(\mathbf{V}, \mathbf{U}^q - \mathbf{U}^p) + \Delta t[(\mathbf{V}, \mathbf{f}(\mathbf{U}^q)) + A(\mathbf{V}, \mathbf{U}^q)] = \Delta t \langle V, \mathbf{D}\mathbf{U}_\nu^q \rangle, \quad \text{for all} \quad \mathbf{V} \in S_{\Delta_T}^1, \tag{3.4}$$

where $\Delta t = q - p$.

The strategy employed to obtain error estimates for the piecewise bilinear solution $\mathbf{U}^q(\mathbf{x})$ uses continuous spaces $S_{\Delta_T}^{1,2}$ of piecewise second-order serendipity functions. Consider a second solution $\mathbf{U}_Q^q(\mathbf{x}) \in S_{\Delta_T}^{1,2}$ of (3.3) using trapezoidal rule integration in time. This solution is higher order in both space and time than $\mathbf{U}^q(\mathbf{x})$ and the difference $\mathbf{U}^q - \mathbf{U}_Q^q$ furnishes an estimate of the discretizon error of $\mathbf{U}^q$. The computational efficiency of this procedure can be improved by using the nodal superconvergence property of finite element methods for parabolic problems [1]. Nodal superconvergence implies that spatial convergence of finite element solutions is of higher order at nodes than it is globally. This further implies that $\mathbf{U}_Q^q$ can be approximated as

$$\mathbf{U}_Q^q(\mathbf{x}) \approx \hat{\mathbf{U}}_Q^q(\mathbf{x}) = \hat{\mathbf{U}}^q(\mathbf{x}) + \mathbf{E}^q(\mathbf{x}), \tag{3.5a}$$

where $\hat{\mathbf{U}}^q(\mathbf{x})$ is the piecewise bilinear solution of (3.3) obtained using trapezoidal rule integration, i.e.,

$$(\mathbf{V}, \hat{\mathbf{U}}^q) + 1/2\, \Delta t\, [(\mathbf{V}, \mathbf{f}(\hat{\mathbf{U}}^q)) + A(\mathbf{V}, \hat{\mathbf{U}}^q)] = (\mathbf{V}, \mathbf{U}^p) - 1/2\, \Delta t\, [(\mathbf{V}, \mathbf{f}^p) + A(\mathbf{V}, \mathbf{U}^p)], \quad \text{for all} \quad \mathbf{V} \in S_{\Delta_T}^1. \tag{3.5b}$$

The piecewise second-order serendipity function $\mathbf{E}^q(\mathbf{x}) \in S_{\Delta_T}^{1,2}$ that vanishes at the vertices of T satisfies

$$(\mathbf{V}, \hat{\mathbf{U}}^q + \mathbf{E}^q) + 1/2\, \Delta t\, [(\mathbf{V}, \mathbf{f}(\hat{\mathbf{U}}^q + \mathbf{E}^q)) + A(\mathbf{V}, \hat{\mathbf{U}}^q + \mathbf{E}^q)] = (\mathbf{V}, \hat{\mathbf{U}}^p + \mathbf{E}^p) - 1/2\, \Delta t\, [(\mathbf{V}, \mathbf{f}(\hat{\mathbf{U}}^p + \mathbf{E}^p)) + A(\mathbf{V}, \hat{\mathbf{U}}^p + \mathbf{E}^p)], \quad \text{for all} \quad \mathbf{V} \in S_{\Delta_T}^{1,2}. \tag{3.5c}$$

The discretization error is approximated as $\mathbf{U}^q - \hat{\mathbf{U}}^q - \mathbf{E}^q$. Separate estimates of the spatial and temporal discretization error can be obtained as $\mathbf{E}^q$ and $\mathbf{U}^q - \hat{\mathbf{U}}^q$,

respectively. Such estimates are useful when assigning different refinement strategies to different megagrids. In our procedure, all error estimates are computed subsequent to convergence of the Schwarz iteration. A refinement indicator for each element is obtained from the local value of $\mathbf{U}^q - \hat{\mathbf{U}}^q - \mathbf{E}^q$ in a local H^1 norm.

The nonlinear ordinary differential systems (3.4), (3.5) are solved by Newton's method. The Jacobians of (3.4) and (3.5b) only differ by a factor of $1/2\,\Delta t$; thus, separate assemblies can easily be avoided. In a prior effort [22], we used the Jacobian of (3.4) for both the solution of (3.4) and (3.5b). The present alternative requires the same effort to assemble the systems, but provides improved convergence for the solution of (3.5b). Poor initial guesses for $\mathbf{U}^q$ can cause divergence of the Newton iteration. Divergence is detected using a criterion suggested by Ascher *et al.* [5], whereupon the time step is halved and the Newton iteration retried. When a Newton iteration fails on a base time step, the base time step is halved until the temporal error estimate is small enough to enlarge it again.

Resulting linear algebraic systems are solved using either a preconditioned conjugate gradient iteration or a preconditioned incomplete orthogonalization method (IOM) [25]. Iterative methods are becoming increasingly popular for the solution of linear systems arising from stiff ordinary differential systems [11]. The IOM, which is an approximation of the full orthogonalization method, has been shown to work well for asymmetric systems. Briefly, the full orthogonalization method for the solution of the $n \times n$ linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{3.6a}$$

constructs an orthonormal basis

$$\mathbf{W}_n = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_n] \tag{3.6b}$$

for the Krylov subspace

$$K_n = \text{span}\{\mathbf{r}, \mathbf{A}\mathbf{r}, ..., \mathbf{A}^{n-1}\mathbf{r}\}, \tag{3.6c}$$

where the residual

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \tag{3.6d}$$

with $\mathbf{x}_0$ being an initial guess. The IOM(k) method requires that $\mathbf{A}\mathbf{v}_j$ only be orthogonal to the previous k vectors, i.e., $\mathbf{v}_{j-k+1}, ..., \mathbf{v}_j$. The preconditioner for both the conjugate gradient and the IOM(k) methods is the incomplete Cholesky factorization of $1/2(\mathbf{A} + \mathbf{A}^T)$; hence, sparsity is preserved and the preconditioner can be used even in the case that $\mathbf{A}$ is nonsymmetric. The same preconditioner is used in calculating both $\mathbf{U}^q$ and $\hat{\mathbf{U}}^q$. The adaptive LRM algorithm automatically selects either the conjugate gradient or IOM(k) method. Preconditioned conjugate gradient iteration is used only if $\mathbf{A}$ is symmetric and the incomplete Cholesky factorization is successful. IOM(2) with Gram–Schmidt orthogonalization is used if $\mathbf{A}$ is not symmetric and incomplete Cholesky factorization succeeds; otherwise, IOM(4) with orthogonalization using Householder transformations [27] is used.

4. EXPERIMENTAL RESULTS

Computational results of three examples are used to demonstrate the performance of the adaptive local refinement procedure, quantify some of its advantages relative to fixed-mesh calculations, and provide some evidence that our techniques can be used to solve practical problems arising in engineering. Calculations were performed in double precision arithmetic on either SUN 3 or SUN 4 workstations or an IBM 3090S/200 computer at the Rensselaer Polytechnic Institute and Tulane University. A one-element buffer ($\lambda = 1$) was added to each local grid.

EXAMPLE 1. Consider the linear heat conduction equation

$$u_t + f(x, y, t) = \Delta u, \qquad 0 < x, y < 1, \; t > 0. \tag{4.1}$$

The Dirichlet boundary conditions, source $f(x, y, t)$, and initial conditions are selected so that the exact solution of (4.1) is

$$u(x, y, t) = 0.8e^{-80[(x - r(t))^2 + (y - s(t))^2]}, \tag{4.2a}$$

where

$$r(t) = 0.5 + 0.25 \sin \pi t, \qquad s(t) = 0.5 + 0.25 \cos \pi t. \tag{4.2b}$$

This solution is a "cone" initially centered at (0.5, 0.75) that rotates clockwise in a circle centered at (0.5, 0.5).

In order to demonstrate the accuracy of our local error estimate, we solved (4.1) on uniform grids for one time step with $\Delta x = \Delta y = \Delta t = 0.2$, 0.1, 0.05, and 0.025. The effectivity index is a common measure of the accuracy of an error estimate. Herein, we define it as the ratio of the local error estimate to the local error in the H^1 norm, i.e.,

$$\theta_j = \frac{\| U^{\Delta t} - \hat{U}^{\Delta t} - E^{\Delta t} \|_{1,j}}{\| U^{\Delta t} - u(\cdot, \cdot, \Delta t) \|_{1,j}}, \tag{4.3a}$$

where

$$\| u \|^2_{1,j} = \iint_{\Delta_{T_j}} (u_x^2 + u_y^2 + u^2)\, dx\, dy, \tag{4.3b}$$

Δ_{T_j} being an element of the local grid Δ_T (cf. Section 3).

TABLE I

Maximum Local Error and Effectivity Index $|\theta_j|_\infty$ as Function of the Uniform Mesh Spacing $\Delta x = \Delta y = \Delta t$ for Example 1

Δt	$\|U^{\Delta t} - u(\cdot, \cdot, \Delta t)\|_{1,\infty}$	$\|\theta_j\|_\infty$
0.2	0.591	0.448
0.1	0.356	0.814
0.05	0.116	0.921
0.025	0.032	0.977

Also, let $|\theta_j|_\infty = \max_{\{j \mid \Delta T_j \in \Delta T\}} |\theta_j|$. In Table I, we present the maximum local error and $|\theta_j|_\infty$ for each Δt.

The results indicate that the local error is approaching a quadratic rate as predicted by theory. The effectivity index is approaching unity at faster than a linear rate, indicating that the error estimate is converging to the exact local error in H^1. Furthermore, the error estimate is more than 80% of the exact local error for all but the largest mesh spacing, which implies that it is a robust measure of the local error.

Adaptive solutions of (4.1) were obtained on $0 \leqslant t \leqslant 1$ using a 10×10 base grid, a time step of 0.1, and local H^1 error tolerances of 0.1, 0.05, and 0.025. Results for the global errors at $t = 0.5$, 1 and the total number of space-time elements to integrate from $t = 0$ to 0.5 and 1 are presented as functions of the local error tolerance in Table II. As the tolerance is halved, the exact error at $t = 0.5$ and 1 decreases. This indicates that controlling the local error allows us, in some sense, to control the global error as well. Local grids at $t = 0.3$, 0.5, 0.8, and 1.0 are presented in Fig. 5 for solutions obtained with a tolerance of 0.05.

For comparison, a solution computed on a uniform 40×40 grid with a time step of 0.025 had an error of 0.158 and used 32,000 space-time elements to integrate to $t = 0.5$, and had an error of 0.158 and used 64,000 space-time elements to integrate to 1.0. Assuming that the number of space-time elements is a reasonable measure of computational complexity, the adaptive LRM solution with a tolerance of 0.05 required 70% fewer elements to integrate to $t = 0.5$, and 69% fewer elements to integrate to 1.0 for approximately the same global accuracy.

TABLE II

Global Error at $t = 0.5$, 1.0 and the Total Number of Space-Time Elements N to Integrate from $t = 0$ to 0.5 and 1.0 as Functions of the Local Error Tolerance *tol*.

	$t = 0.5$		$t = 1.0$	
Tol	$\|U^{0.5} - u(\cdot, \cdot, 0.5)\|_1$	N	$\|U^{1.0} - u(\cdot, \cdot, 1.0)\|_1$	N
0.1	0.224	3840	0.224	7140
0.05	0.149	9556	0.149	19,768
0.025	0.119	19,774	0.119	40,204

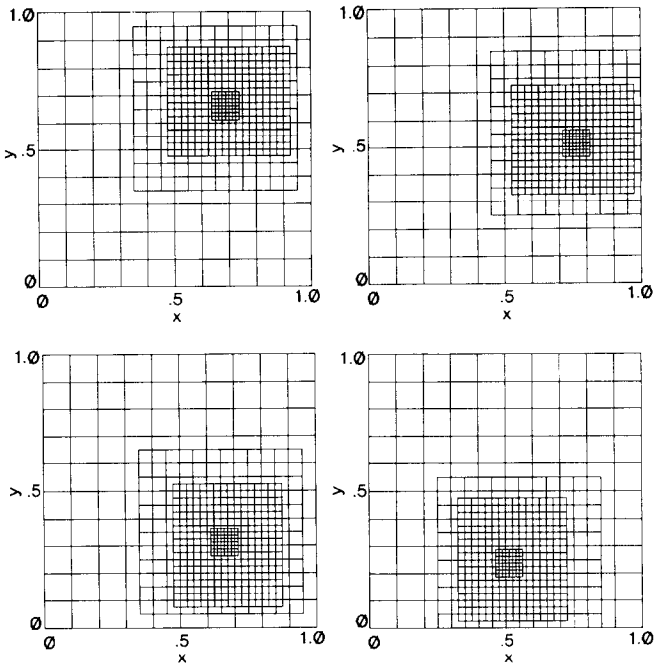

FIG. 5. Local grids generated in solving Example 1 by the LRM at $t = 0.3$ (upper left), $t = 0.5$ (upper right), $t = 0.8$ (lower left), and $t = 1.0$ (lower right) with a tolerance $tol = 0.05$, a 10×10 base spatial mesh, and a base time step of 0.1.

Identical errors at $t = 0.5$ and 1.0 indicate that the conical solution has retained its shape while rotating and that the local meshes are accurately tracking its position.

EXAMPLE 2. Consider the combustion problem

$$u_t = \Delta u - Due^{-\delta/T}, \tag{4.4a}$$

$$LT_t = \Delta T + \alpha Due^{-\delta/T}, \qquad 0 < x, y < 1, \ t > 0, \tag{4.4b}$$

$$u(x, y, 0) = T(x, y, 0) = 1, \qquad 0 \leqslant x, y \leqslant 1, \tag{4.4c}$$

$$u_x(0, y, t) = T_x(0, y, t) = 0, \quad u(1, y, t) = T(1, y, t) = 1, \qquad 0 < y < 1, \ t > 0, \tag{4.4d}$$

$$u_y(x, 0, t) = T_y(x, 0, t) = 0, \quad u(x, 1, t) = T(x, 1, t) = 1, \qquad 0 < x < 1, \ t > 0. \tag{4.4e}$$

The variables u and T denote the concentration and temperature of a chemical that is undergoing a one-step reaction in a domain $-1 < x, y < 1$. Symmetry considerations permit solving the problem in one quadrant of this domain. A one-dimensional version of this problem was investigated by Kapila [17]. The parameter L is the Lewis number, δ is the activation energy, α is the heat release, and

$$D = \mathrm{Re}^{\delta}/\alpha\delta. \tag{4.4f}$$

is the Damkohler number with R being the reaction rate.

The temperature and concentration are related by $T + \alpha u = 1 + \alpha$ when $L = 1$ and, thus, (4.4) can be reduced to a scalar problem. This case has been studied by both analytical and numerical methods [17, 2, 21]. Its solution features a temperature that initially increases slowly from unity forming a "hot spot" at the origin. An ignition occurs at a finite time and the temperature at (0, 0) jumps from near unity to approximately $1 + \alpha$. A sharp reaction front subsequently forms and propagates rapidly towards the boundaries $x = 1$ and $y = 1$ where a boundary layer develops. Problems where L differs slightly from unity are expected to have a similar behavior.

We solved (4.4) with $L = 0.9$, $\delta = 20$, $\alpha = 1$, and $R = 5$ using a local H^1 error tolerance of 0.05 on a 20×20 base mesh having a time step of 0.05. Surface renditions of the concentration and temperature are shown in Fig. 6 as functions of position at times $t = 0.225$, 0.2375, and 0.25, which correspond to times while the front is rapidly crossing the domain. The meshes at the three times are shown in Fig. 7. The results presented in Fig. 6 and 7 demonstrate that the LRM can be used to solve complex nonlinear problems having strong dynamic structures without having a detailed knowledge of the solution behavior, without needing to tune numerical parameters and with no user intervention. Although the grids pictured in Fig. 7 do not overlap, the grids did overlap at other times during execution, thereby requiring the use of the Schwarz alternating method as outlined in Section 2.

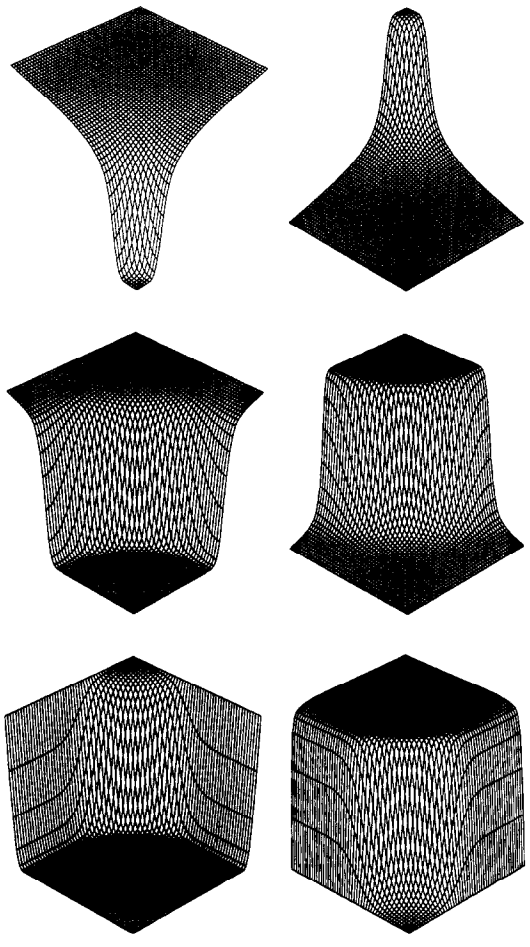

FIG. 6. Concentrations (left) and meshes (right) at $t = 0.225$, 0.2375, and 0.25 (top to bottom) for E origins for concentrations and temperatures are at the front and respectively.

EXAMPLE 3. Consider the dimensionless Fokker–Planck equation

$$P_t = -vP_x + \beta P + \left[\beta v + \frac{dU(x)}{dx}\right] P_v + \frac{\beta KT}{m\omega^2} P_{vv}, \quad (4.5a)$$

for the probability density function $P(x, v, t)$ corresponding to a particle of mass m moving in a bistable potential, $U(x)\, m\omega^2$, coupled with a heat bath [10]. Cartling [10] considers the double-well potential

$$U(x) = \begin{cases} 1/2(x+2)^2, & x < -1, \\ 1 - 1/2x^2, & -1 \leqslant x \leqslant 1, \\ 1/2(x-2)^2, & x > 1, \end{cases} \quad (4.5b)$$

formed by joining three harmonic functions where ω is the angular frequency of deterministic oscillations in either well, $\beta\omega$ is the damping constant describing the strength of coupling to the heat bath, κ is the Boltzmann constant, and

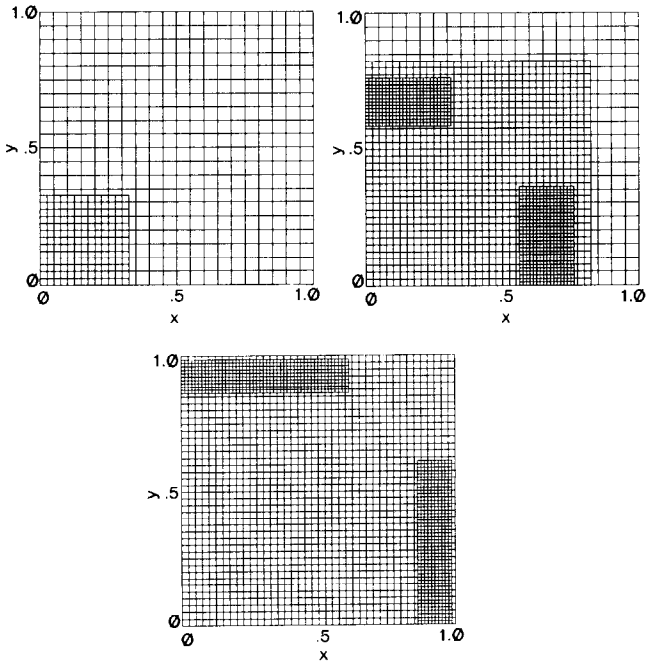

FIG. 7. Sequence of overlapping meshes at $t = 0.225$ (upper left), 0.2375 (upper right), and 0.25 (lower center) for Example 2.

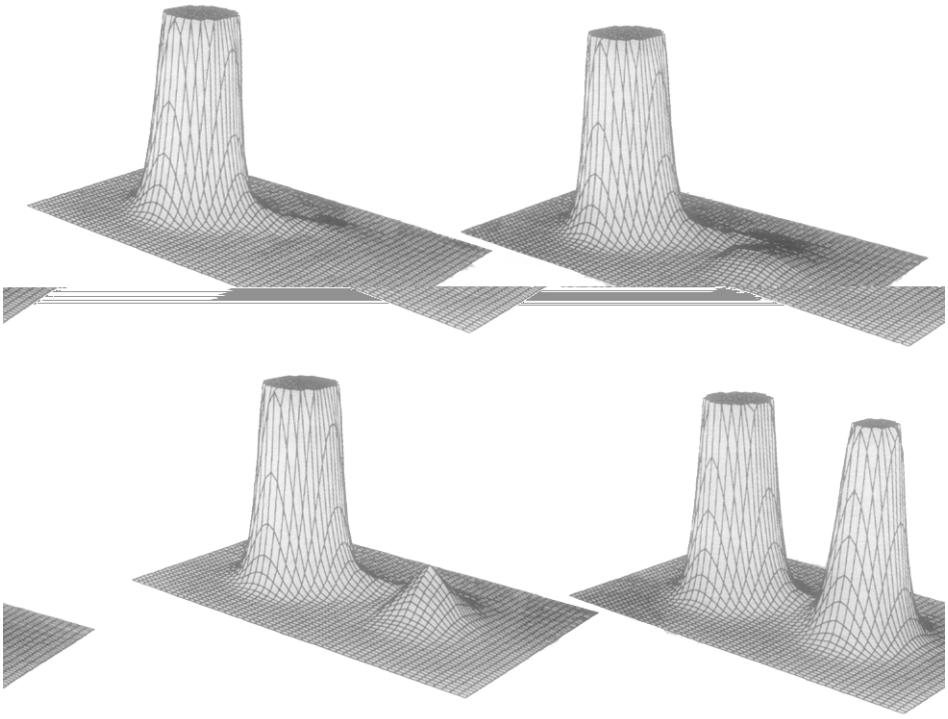

FIG. 8. Surface rendition of $P(x, v, t)$ at $t = 4$ (upper left), 10 (upper right), 20 (lower left), and 100 (lower right) for Example 3.

T is the absolute temperature. Following Cartling we take the initial state to be

$$P(x, v, 0) = \frac{e^{-1/2(m\omega^2/\kappa T)[(x+2)^2+v^2]}}{2\pi\kappa T/m\omega^2} \tag{4.5c}$$

which is the Maxwell–Boltzmann distribution corresponding to the equilibrium state where only the left harmonic part of the potential (4.5b) is present. The numerical domain is chosen so that homogeneous Dirichlet boundary conditions may be applied at the domain boundaries. For (4.5a)–(4.5c) we solve on $-5 \leqslant x \leqslant 5$, $-3 \leqslant v \leqslant 3$.

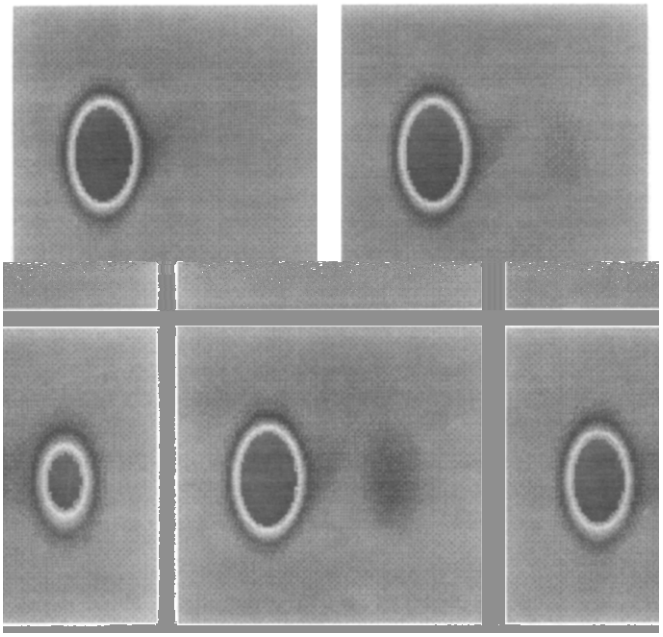

FIG. 9. Dithered renditions of center contour plots of $P(x, v, t)$ at $t = 4$ (upper left), 10 (upper right), 20 (lower left), and 100 (lower right) for Example 3.

We present results for $\beta = 0.4$ which represents moderate damping, and $m\omega^2/\kappa T = 4$ which corresponds to the maximal reaction rate. We solved (4.5a)–(4.5c) with the parameter values listed above on a 20×12 base mesh having a time step 0.5 and a local H^1 error tolerance of 0.005. Surface renditions and dithered renditions of color contour plots of our finite element solution at $t = 4$, 10, 20, and 100 are shown in Figs. 8 and 9, respectively. Corresponding meshes are shown in Fig. 10. Probability densities greater than 0.1 are not shown in order to emphasize the fine-scale structure that is present in the solution.

Our results are qualitatively the same as those of [10]

TABLE III

The Number of Space-Time Elements, N_{ADI} and N_{LRM}, for the Method of Cartling [10] and the LRM, respectively, at $t = 4$, 10, 20, 100

t	$N_{\text{ADI}} \times 10^{-6}$	$N_{\text{LRM}} \times 10^{-6}$
4.0	48	0.21
10.0	120	0.53
20.0	240	1.0
100.0	480	5.5

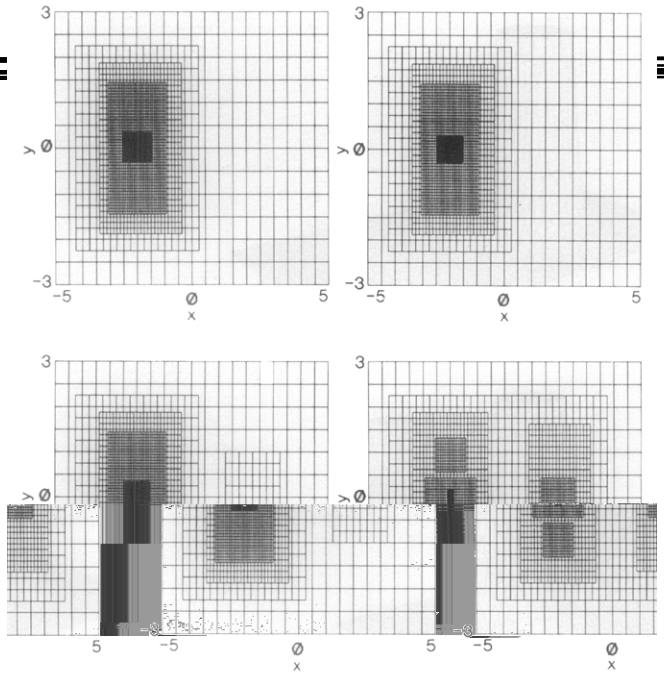

FIG. 10. Sequence of overlapping meshes at $t=4$ (upper left), 10 (upper right), 20 (lower left), and 100 (lower right) for Example 3.

for $t=4$, 10, and 20. However, by $t=100$ the probability density over the second well ($x=2$, $v=0$) is significantly larger in our results. Cartling computed the solution of (4.5a)–(4.5c) using central differencing in space and the trapezoidal rule in time. The resulting implicit system is solved using an ADI method. For this case, a mesh spacing of 0.02 in x and v and time steps of 0.025 and 0.1 were used for $0 \leqslant t \leqslant 20$ and $t > 20$, respectively. The smallest time (and space) step used in obtaining our solution was 0.03125. This time step was needed throughout the calculation on the finest spatial grid.

A comparison of the number of space-time elements used in calculating the solution at times $t=4$, 10, 20, and 100 by the finite difference ADI method of Cartling [10] and the LRM is shown in Table III. The adaptive LRM uses significantly fewer elements than the ADI method in obtaining the same solution.

The Fokker–Planck equation is not strictly of the form (1.1) because no P_{xx} term is present in (4.5a). Nevertheless, the refinement indicator has produced finer meshes in the correct areas of the problem domain. The local discretization error-based refinement indicator may not, however, be providing an accurate estimate of the error in this problem. Grid overlap was observed at times other than those of Fig. 10 during the execution of the problem.

5. DISCUSSION

We have developed an adaptive local refinement procedure for parabolic systems in two space dimensions. This differs from the more typical method-of-lines approach which requires the solution of problem over the entire domain for each time step. Thus different time steps can be used in different regions of the domain which was observed in Example 3. A tree data structure of megagrids is used to manage a nest of local overlapping grids. An implicit finite-element solution strategy using piecewise bilinear elements in space and the backward Euler method in time is formulated. We obtain an estimate of the local discretization error of the solution using a p-refinement approach with piecewise serendipity approximations in space and trapezoidal rule integration in time. A version of the Schwarz alternating principle is used to coordinate solutions between overlapping grids because of the local nature of the solution procedure.

The computational examples of Section 4 indicate that the error estimate converges for linear problems. Examples 2 and 3 demonstrate that the LRM can be used to solve some difficult nonlinear problems. Example 3 also shows the advantages that can be gained from using adaptive methods rather than fixed grids. Iterative methods have proved effective in solving the linear systems resulting from the modified Newton method even for nonsymmetric Jacobians.

Although three Schwarz iterations were used to obtain convergence, the results in Lions [20] indicate that only one may be necessary. Preliminary results using one iteration have been promising. This modification significantly reduces the amount of work since Jacobians need be assembled only once. Problems on non-rectangular domains can presumably be handled by combining our LRM with a composite grid mesh generating scheme of, e.g., Henshaw [15].

REFERENCES

1. S. Adjerid and J. E. Flaherty, *SIAM J. Sci. Statist. Comput.* **9**, 792 (1988).
2. S. Adjerid and J. E. Flaherty, *Numer. Math.* **53**, 183 (1988).
3. S. Adjerid and J. E. Flaherty, *SIAM J. Numer. Anal.* **23**, 778 (1986).
4. S. Adjerid and J. E. Flaherty, *Comput. Methods Appl. Mech. Eng.* **55**, 3 (1986).
5. U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations* (Prentice Hall, Englewood Cliffs, NJ, 1988).
6. M. J. Baines and A. J. Wathen, *J. Comput. Phys.* **79**, 245 (1988).
7. M. Berger and J. Oliger, *J. Comput. Phys.* **53**, 484 (1984).
8. M. Bieterman and I. Babuska, *Numer. Math.* **40**, 339 (1982).
9. M. Bieterman and I. Babuska, *Numer. Math.* **40**, 373 (1982).
10. B. Cartling, *J. Chem. Phys.* **87**, 2638 (1987).
11. T. F. Chan and K. R. Jackson, *SIAM J. Sci. Statist. Comput.* **7**, 378 (1986).
12. S. F. Davis and J. E. Flaherty, *SIAM J. Sci. Stat. Comput.* **3**, 6 (1982).
13. Q. V. Dihn, R. Glowinski, and J. Periaux, in *Elliptic Problem Solvers II*, edited by G. Birkhoff and A. Schoenstadt (Academic Press, Orlando, FL, 1984), p. 395.

14. R. J. Gelinas, S. K. Doss, and K. Miller, *J. Comput. Phys.* **40**, 202 (1981).
15. W. D. Henshaw and G. Chesshire, *SIAM J. Sci. Statist. Comput.* **8**, 914 (1987).
16. I. W. Johnson, M. J. Baines, and A. J. Wathen, *J. Comput. Phys.* **79**, 270 (1988).
17. A. K. Kapila, *Asymptotic Treatment of Chemically Reacting Systems* (Pitman Advanced Publishing Program, Boston, 1983).
18. S. L. Keeling, *Math. Comput.* **52**, 561 (1989).
19. D. E. Keyes and W. D. Gropp, *SIAM J. Sci. Statist. Comput.* **8**, s166 (1987).
20. P. L. Lions, in *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, edited by R. Glowinski, G. H. Golub, G. A. Meurant, and J. Périaux (SIAM, Philadelphia, 1988), p. 1.
21. P. K. Moore and J. E. Flaherty, *SIAM J. Numer. Anal.* **27**, 6 (1990).
22. P. K. Moore and J. E. Flaherty, in *Trans. Fifth Army Conf. on Appl. Math. and Comput. ARO Rep. 88-1* (U.S. Army Research Office, Research Triangle Park, 1988), p. 459 (unpublished).
23. J. Oliger, W. Skamarock, and W. Tang, Stanford technical report (Stanford University, Stanford, 1986) (unpublished).
24. L. R. Petzold, in *Adaptive Methods for Partial Differential Equations*, edited by J. E. Flaherty, P. J. Paslow, M. S. Shephard, and J. D. Vasilakis (SIAM, Philadelphia, 1989), p. 253.
25. Y. Saad, *SIAM J. Sci. Statist. Comput.* **5**, 203 (1984).
26. J. G. Verwer, J. G. Blom, R. M. Furzeland, and P. A. Zegeling, in *Adaptive Methods for Partial Differential Equations*, edited by J. E. Flaherty, P. J. Paslow, M. S. Shephard, and J. D. Vasilakis (SIAM, Philadelphia, 1989), p. 160.
27. H. F. Walker, *SIAM J. Sci. Statist. Comput.* **9**, 152 (1988).