SIAM J. Sci. Comput.
Vol. 29, No. 2, pp. 656–673

A SCHWARZ WAVEFORM MOVING MESH METHOD*

RONALD D. HAYNES[†] AND ROBERT D. RUSSELL[‡]

Abstract. An r-refinement (moving mesh) method is considered for solving time dependent partial differential equations (PDEs). The resulting coupled system, consisting of the physical PDE and a moving mesh PDE, is solved by a Schwarz waveform relaxation method. In particular, the computational space-time domain is decomposed into overlapping subdomains and the solution obtained by iteratively solving the system of PDEs on each subdomain. Dirichlet boundary conditions are used to pass solution information between neighboring regions. The efficacy of this approach is demonstrated for some model problems. For problems where the solutions evolve on disparate time scales in different regions of the spatial domain, this approach demonstrates the significant savings in computational time and effort which are possible.

Key words. Schwarz waveform relaxation, moving meshes, domain decomposition, dynamic iteration

AMS subject classifications. 65M55, 65M06, 65M50, 65M20

DOI. 10.1137/050631549

1. Introduction.

The method of lines (MOL), one of the predominant approaches for solving time dependent parabolic partial differential equations (PDEs), involves discretizing spatially and then integrating in time along either fixed or moving lines. The efficacy of the method rests on that of the subsequent initial value problem (IVP) solvers. Traditional IVP solvers integrate using time steps chosen to keep an estimate of the local error below some user specified tolerance. As a consequence, the time step, largely dictated by the behavior of the fastest components locally, is used to integrate all solution components.

This global approach introduces inefficiencies by often requiring a small time step for each unknown, particularly when there are disparate scales in solution components. This occurs when solution scales along lines differ substantially, which is the case, for example, when the solution has moving interfaces or singular behavior.

A number of ways to overcome this limitation by integrating using steps locally suited to the time dynamics have been investigated. The idea appears to have originated in a paper by Rice [42] which considers split or multirate Runge–Kutta methods applied to systems of ordinary differential equations (ODEs). In the independent work of Gander and Stuart [22] and Giladi and Keller [25], the domain decomposition method is applied for solving time dependent PDEs. Splitting the problem spatially allows the solution in different parts of the spatial domain to be evolved according to local time scale. The technique, known as Schwarz waveform relaxation, is able to overcome many of the problems which exist in applying the ODE methods mentioned above to semidiscretized PDEs.

It is well accepted that the efficient solution of complex PDEs generally requires methods which are adaptive in both space and time. In this paper we are interested in

*Received by the editors May 13, 2005; accepted for publication (in revised form) September 20, 2006; published electronically April 6, 2007.

http://www.siam.org/journals/sisc/29-2/63154.html

[†]Department of Mathematics & Statistics, Acadia University, Wolfville, NS B4P 2R6, Canada (ronald.haynes@acadiau.ca). This author was supported in part by NSERC (Canada) postdoctoral fellowships and discovery grants.

[‡]Department of Mathematics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (rdr@cs.sfu.ca). This author was supported in part by NSERC (Canada) under grant OGP-0008781.

a class of spatially adaptive moving mesh PDE methods introduced in [41, 29, 30]. Our purpose is to introduce and explore a natural coupling of domain decomposition, in the Schwarz waveform context, and the spatially adaptive moving mesh PDE methods.

In the next two sections we review the relevant literature for both the ODE and PDE contexts. In section 4 we review the so-called moving mesh PDE methods. Section 5 introduces a Schwarz waveform moving mesh method. In section 6 we provide some numerical results to justify the utility of the technique. We conclude in section 7 with a summary and discussion of ongoing and future work.

2. ODE methods.

There have been a number of approaches for solving systems of ODEs using decoupled integration. A decoupled integration method proceeds by identifying groups of components which evolve at similar time scales or for which it is advantageous to use particular integration formulas. Communication between the subsystems is typically handled through either interpolation or some iterative procedure.

Interpolation based methods. Gear [24] investigates solving a system of ODEs where the ith component is integrated using a step size function $h_i(t)$. Due to the cost of the required interpolations the method is deemed competitive only if the cost of the function evaluations is high or if f_i depends on few solution components. Hofer [28] proposes a method where the ODE is decoupled into stiff and nonstiff components. Computational efficiency is achieved by the decoupling of the integration and by restricting the number of implicit formulations to a few equations. Assuming the system has been decoupled a priori, Andrus [2] presents a method where both a different integration method and time step are used for the slow and fast subsystems. Skelboe [44, 43] investigates multirate decoupled backward differentiation formulas (BDFs) for general systems of ODEs separated into subsystems. The partitioning requires subsystems having strongly interacting components but weak couplings to the other subsystems. Deuflhard [10] suggests analyzing the Jacobian to detect groups of fast and slow components. More recently, Engstler and Lubich [11, 12] have considered a multirate technique which does dynamic decoupling using extrapolation methods.

Iterative based methods. One of the earliest techniques for decoupling the integration of systems of ODEs was waveform relaxation, where subsystems are integrated separately and communication between them handled through iteration. The theory of the method is considered in Lelarasmee, Ruehli, and Sangiovanni-Vincentelli [35] and later in Miekkala and Nevanlinna [38], which contributed to the subsequent popularization of the method.

3. PDE methods.

There have been several attempts to incorporate a multirate strategy within a solution methodology for PDEs. The MOL, for example, yields a large system of ODEs for which there may be a motivation to decouple the fast and slow components. The partitioning of components can be done dynamically and automatically to adapt to temporally and spatially evolving features in the solution, although this complicates the use of the majority of the above ODE methods.

Not surprisingly the motivation to use different space-time grids in regions of the computational domain has spawned intensive research. Flaherty and Moore [14] develop integrated space-time *hp*-refinement strategies. These methods make decisions about the number of mesh points, time step size, and the order of the time integration method locally in space-time in a unified manner. In this sense these methods may be viewed as a class of multirate methods. These methods are computationally robust; however, the resulting space-time meshes are complicated and require sophisticated data structures for efficient implementation [13].

Domain decomposition methods (e.g., see [7, 40]) attempt to solve a problem over the entire spatial domain by iteratively solving on subdomains. There have been at least three general approaches for applying domain decomposition to parabolic problems in a space-time domain:

1. Discretize in time and solve the resulting elliptic problems with classical domain decomposition [5, 37, 34].
2. Discretize in space and apply waveform relaxation to the system of ODEs [36, 33, 32].
3. Subdivide the spatial domain and iteratively solve a sequence of PDEs defined on each subdomain [15, 18, 25].

The first approach is in some sense the most natural since one utilizes the extensive literature for domain decomposition applied to elliptic problems. This technique suffers from a couple of difficulties. First, since the elliptic problems arise after discretizing in time, we are forced to use the same time step on each subdomain. Second, the computed solution information must be exchanged at the end of each time step. This necessitates that identical time steps be used throughout the spatial domain and makes a natural parallel implementation difficult.

The second approach allows different time steps in different regions of the spatial domains. Information is transmitted between subdomains after solving a subset of ODEs over a time window, not after each step. Unfortunately, for Jacobi, Gauss–Seidel, and SOR waveform relaxation methods, in the case where the ODEs result from a spatial discretization of a PDE, the constants which arise in the error estimates depend negatively on the mesh parameter Δx. Sophisticated multigrid techniques have been proposed to overcome the negative dependence on the mesh parameter for these classical waveform relaxation methods; cf. [32, 36, 46].

The third class of methods, which have become known as Schwarz waveform relaxation methods, overcome the poor convergence of waveform relaxation methods. These methods were developed by Gander and colleagues [17, 22, 18, 20, 15] and independently by Giladi and Keller in [25]. The methods allow different numerical treatments (time step and integration formula) on different subdomains and have convergence independent of the mesh parameter, without the added complication of the multigrid framework. Since these methods provide the basis for the methods developed in the next section, we describe them in more detail below.

Consider a general parabolic problem

$$u_t = \mathcal{L}(u, x, t) \qquad \text{in } \Omega,$$

subject to appropriate initial and boundary conditions on $\partial\Omega$. In the case of two overlapping subdomains Ω_0 and Ω_1, the classical Schwarz waveform relaxation method can be succinctly written as follows: for $j = 0, 1$,

$$\begin{aligned} \frac{\partial u_j^{(k+1)}}{\partial t} &= \mathcal{L}(u_j^{(k+1)}, x, t), \qquad (x,t) \in \Omega_j, \\ u_j^{(k+1)}(x,t) &= \begin{cases} u_{1-j}^{(k)}(x,t), & (x,t) \in \Gamma_j = \partial\Omega_j \cap \Omega_{1-j}, \\ \text{given boundary condition} & x, t \in \partial\Omega_j - \Gamma_j. \end{cases} \end{aligned}$$

There have been many theoretical advances towards understanding the convergence of Schwarz waveform relaxation. In [22] linear and superlinear convergence results are obtained for the one-dimensional heat equation. Convergence is shown to be independent of the mesh parameter (and hence robust with respect to mesh refinement), and the convergence rate improves by increasing the size of the overlap. Giladi and Keller [25] prove superlinear convergence for constant coefficient convection diffusion equations. Gander [17, 18] extends these results to the one-dimensional, variable coefficient reaction diffusion case. Gander, Halpern, and Nataf [20] apply overlapping Schwarz waveform methods to the wave equation and a constant coefficient, linear convection reaction diffusion equation. The effect of the Dirichlet transmission conditions at the subdomain boundaries is studied and found to slow convergence of the algorithms. Optimal transmission conditions are then derived which lead to nonoverlapping Schwarz methods which converge in a finite number of steps [16]. Multidimensional extensions of many of these results are available [23, 19]. In a recent paper [15], Gander and Rohde provide results for one-dimensional convection-dominated conservation laws.

In the next section we describe the moving mesh PDE method and in section 5 we propose a new moving mesh Schwarz waveform method which inherits the favorable properties of the basic method and the spatial mesh resolution abilities of moving mesh PDEs.

4. Moving mesh methods.

Adaptive mesh methods for PDEs typically fall into one or more of the following broad categories:

- r-refinement: moving a fixed number of mesh points to difficult regions of the physical domain,
- p-refinement: varying the order of the numerical method to adapt to local solution smoothness,
- h-refinement: mesh refinement and derefinement, depending upon the local level of resolution.

These methods may be applied in either a static or dynamic fashion. Static methods involve refining/coarsening or redistributing mesh points at fixed times during a calculation. Dynamic (or moving mesh) methods solve for the solution and mesh simultaneously. The latter, some of whose advantageous features will be explained subsequently, are the ones considered here.

To be specific, consider the solution of a PDE of the form

$$u_t = \mathcal{L}(u), \qquad 0 < x < 1, \quad t > 0,$$

subject to appropriate initial and boundary conditions, where again $\mathcal{L}$ denotes a spatial differential operator in the physical coordinate x. Our goal is basically to find, for fixed t, a one-to-one coordinate transformation

$$x = x(\xi, t) : [0,1] \to [0,1] \quad \text{with } x(0,t) = 0,\ x(1,t) = 1$$

such that $u(x(\xi,t),t)$ is sufficiently smooth that a simple mesh can be used to resolve solution features in the computational domain $\xi \in [0,1]$. Typically a uniform mesh,

$$\xi_i = \frac{i}{N}, \quad i = 0, 1, \ldots, N,$$

suffices, and the mesh in the physical coordinate x is then specified from the mesh transformation by $x_i(t) = x(\xi_i, t),\ i = 0, 1, \ldots, N$.

A standard way to perform mesh adaptation in space is to use the equidistribution principle (EP) [9]: Given some measure $M(t,x,u)$ of the error in the numerical solution, equidistribution requires that the mesh points satisfy

$$\int_{x_{i-1}}^{x_i} M(t,\tilde{x},u)\, d\tilde{x} \equiv \frac{1}{N} \int_0^1 M(t,\tilde{x},u) d\tilde{x}$$

on each subinterval $[x_{i-1}(t), x_i(t)]$ or

$$\int_0^{x(\xi_i,t)} M(t,\tilde{x},u)\,d\tilde{x} \equiv \xi_i \int_0^1 M(t,\tilde{x},u)d\tilde{x} \quad \text{for } i = 1,\ldots,N.$$

The continuous generalization of this is that

$$\int_0^{x(\xi,t)} M(t,\tilde{x},u)\,d\tilde{x} = \xi\theta(t), \tag{4.1}$$

where $\theta(t) \equiv \int_0^1 M(t,\tilde{x},u)\,d\tilde{x}$ (e.g., see [29]). It follows directly from (4.1) that

$$\frac{\partial}{\partial \xi}\left\{M(t,x(\xi,t),u)\frac{\partial}{\partial \xi}x(\xi,t)\right\} = 0. \tag{4.2}$$

Note that (4.2) does not explicitly involve the node speed $\dot{x}$. This is generally introduced by relaxing the equation to require equidistribution at time $t+\tau$. In [41, 29, 30] a number of parabolic moving mesh PDEs (MMPDEs) are developed using somewhat subtle simplifying assumptions and their correspondence to various moving mesh methods (previously motivated only by heuristic arguments) is shown. One particularly useful MMPDE is

$$\frac{\partial}{\partial \xi}\left(M(t,x(\xi,t),u)\frac{\partial \dot{x}}{\partial \xi}\right) = -\frac{1}{\tau}\frac{\partial}{\partial \xi}\left(M(t,x(\xi,t),u)\frac{\partial x}{\partial \xi}\right). \tag{MMPDE4}$$

The relaxation parameter τ is chosen in practice so that the mesh evolves at a rate commensurate with that of the solution $u(x,t)$.

A simple popular choice for $M(t,x,u)$ is the arclength-like monitor function [3, 45]

$$M(x,u,t) = \sqrt{1 + \frac{1}{\alpha}|u_x|^2}. \tag{4.3}$$

This choice is based on the premise that we expect the error in the numerical solution to be largest in regions where the solution has large gradients. The choice of monitor function is often problem class dependent; e.g., monitor functions which preserve the scaling invariance of the physical PDE [4] are seen to work well for reaction diffusion problems having blow-up solutions (as opposed to the arclength monitor function, for instance, which ultimately fails to resolve the solution near the singularity in time).

As outlined in the previous section, using a moving mesh method to solve a PDE requires solving a coupled system of PDEs for the unknown mesh transformation and solution. Using the mesh transformation $x = x(\xi,t)$ to rewrite the physical PDE in quasi-Lagrangian form, we get

$$\dot{u} - u_x\dot{x} = \mathcal{L}u, \tag{4.4}$$

where the total time derivative $\dot{u}$ is given as

$$\dot{u} = u_t + u_x\dot{x}.$$

Equation (4.4) and a moving mesh equation are solved simultaneously for the mesh $x(\xi,t)$ and corresponding solution $u(x(\xi,t),t)$. Specifically, we solve the coupled PDE system

$$\begin{aligned} \dot{u} - u_x\dot{x} &= \mathcal{L}(u), \\ \left(M(x,u,t)\dot{x}_\xi\right)_\xi &= -\frac{1}{\tau}\left(M(x,u,t)x_\xi\right)_\xi \end{aligned} \tag{4.5}$$

using the moving MOL approach. Discretizing spatially with centered differences we obtain the semidiscrete approximation

$$\tag{4.6} \begin{gathered} \dot{u}_i - \left(\frac{u_{i+1}-u_{i-1}}{x_{i+1}-x_{i-1}}\right)\dot{x}_i = f_i, \\ \frac{M_{i+1}+M_i}{2(1/N)^2}\left(\dot{x}_{i+1}-\dot{x}_i\right) - \frac{M_i+M_{i-1}}{2(1/N)^2}\left(\dot{x}_i-\dot{x}_{i-1}\right) = \\ -\frac{1}{\tau}\left\{\frac{M_{i+1}+M_i}{2(1/N)^2}\left(x_{i+1}-x_i\right) - \frac{M_i+M_{i-1}}{2(1/N)^2}\left(x_i-x_{i-1}\right)\right\} \end{gathered}$$

for $i = 1, \ldots, N-1$. The right-hand side of the semidiscrete physical PDE, f_i, denotes the discrete approximation to the spatial operator $\mathcal{L}$ at $\xi = \xi_i$. Centered differences are used to discretize $\mathcal{L}$. The quantity M_i denotes a centered difference approximation to the monitor function at x_i. In actual calculations, M_i is replaced by a smoothed value

$$\tilde{M}_i = \sqrt{\sum_{k=i-p}^{i+p} (M_k)^2 \left(\frac{\gamma}{\gamma+1}\right)^{|k-i|} \Bigg/ \sum_{k=i-p}^{i+p} \left(\frac{\gamma}{\gamma+1}\right)^{|k-i|}},$$

where $\gamma = 2$ and $p = 2$ [29].

Initial and boundary conditions for the physical PDE come from the problem description. On a fixed interval we specify $\dot{x}_0 = \dot{x}_N = 0$ as boundary conditions for the mesh. If the initial solution $u(x,0)$ is smooth, then an initial uniform mesh for $x(\xi,0)$ is normally sufficient. If the initial solution has features which are not resolved on a uniform mesh for the chosen value of N, then an equidistributed initial mesh is required. One way used in [31] to compute an initial equidistributed mesh is to solve the PDE

$$u_t = u_0(x), \quad u(x,0) = 0,$$

coupled with a moving mesh equation. Solving this system over the time interval $0 \le t \le 1$ with a uniform initial mesh will yield the solution $u(x,1) = u_0(x)$, and the resulting mesh will approximately equidistribute the initial solution. To avoid propagating any errors from this calculation, the initial solution is reevaluated at the new initial mesh.

The semidiscrete approximation (4.6) describes a system of linearly implicit ODEs which may be written in the form

$$\tag{4.7} V(y)y' = g(y).$$

The vector y denotes the unknowns ordered as $(u_0, x_0, u_1, x_1, \ldots, u_N, x_N)^T$, and $V(y)$ is a banded matrix which depends on the unknown solution and mesh.

A typical implementation (cf. [31]) for solving (4.5) involves discretizing spatially and solving (4.7) with a stiff ODE solver such as DASSL [39]. In the next section we propose a Schwarz waveform moving mesh method to solve (4.5) and describe its implementation.

5. Schwarz waveform moving mesh method. In this section we propose a new Schwarz waveform moving mesh (or simply "moving Schwarz") method. The basic idea is to apply Schwarz waveform relaxation, as described in section 3, to the coupled system of physical and moving mesh PDEs (4.5). Specifically, we first decompose $\Omega = [0, 1]$ into D nonoverlapping fixed subdomains $\tilde{\Omega}_j$, $j = 1, \ldots, D$. Each subdomain $\tilde{\Omega}_j$ is enlarged by an overlap region consisting of R mesh points, giving overlapping domains $\Omega_1, \Omega_2, \ldots, \Omega_D$. The physical PDE is then discretized along with a moving mesh equation on each subdomain. We fix the mesh points on the boundary of $\tilde{\Omega}_j$, $j = 1, \ldots, D$, but allow the mesh points to move within and on the outer boundary of the overlap regions. Algorithmically, there is no reason to keep the boundaries fixed. However, there is a tendency for moving mesh methods to emphasize the most pathological regions of the solution. Allowing the subdomain boundaries to move could potentially negate the effort to isolate difficult regions within individual subdomains. We will return to this point in the next section.

The convergence of overlapping classical Schwarz waveform relaxation is known to depend on the width of the overlap region. In this implementation we have found that having the width of the overlap in the computational domain of moderate size works well, even for somewhat pathological solution behavior in the physical domain where the width of the overlap in the physical variables varies (but the number of mesh points in this region is fixed). A priori determining an optimal width of the overlap region in the computational domain would be difficult. As in the fixed mesh case, the rate of convergence of the Schwarz iteration is improved as the size of the overlap is increased, with the faster convergence being offset by the increased computational cost. Things are further complicated by the desire to isolate difficult regions of the solution from regions where there is little activity. As the overlap is increased more subdomains become "active."

Let x_j and ξ_j denote the physical and computational meshes on each subdomain Ω_j, and denote the solution on each subdomain by u_j. The moving Schwarz method requires solving

$$\begin{gathered} \dot{u}_j^k - \frac{\partial u_j^k}{\partial x}\dot{x}_j^k = \mathcal{L}(u_j^k), \\ \left(M(x_j^k, u_j^k, t)\frac{\partial \dot{x}_j^k}{\partial \xi} \right)_\xi = -\frac{1}{\tau}\left(M(x_j^k, u_j^k, t)\frac{\partial x_j^k}{\partial \xi} \right)_\xi , \end{gathered}$$

on Ω_j with $j = 1, \ldots, D$, for $k = 1, 2, \ldots$.

The boundary values for u_j^k and x_j^k come from the values $u_{j-1}^{k-1}, x_{j-1}^{k-1}$ and $u_{j+1}^{k-1}, x_{j+1}^{k-1}$ on the left and right boundaries of Ω_j, respectively, from the previous iteration. Each Schwarz waveform iteration requires the solution of D problems with (known) moving boundaries. This is illustrated in Figure 5.1 for $D = 3$.

On each subdomain the coupled system of PDEs is discretized in space as described in section 4 and the resulting ODEs are integrated over successive time windows. The time dependent boundary conditions for u_j and x_j on these time windows are obtained by cubic spline interpolation of the data from the previous iteration. Interpolation is required since in general the data from iteration $k - 1$ is not computed on the same sequence of time steps as iteration k.

For efficiency, the algorithm is designed with an adaptive time windowing strategy. The algorithm begins by solving the sequence of moving boundary problems on a time window $[0, T]$. If u_j or x_j fails to reach $t = T$ in the maximum allowable time

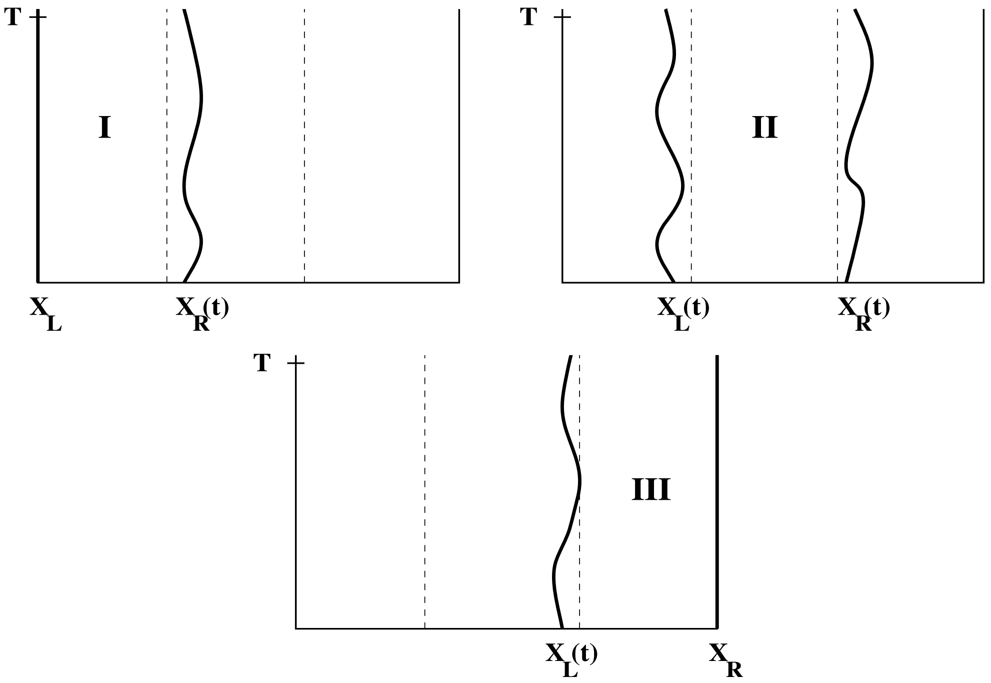

FIG. 5.1. *Sequence of moving boundary problems solved during one iteration of the Schwarz waveform moving mesh method over a time window* $[0, T]$.

steps, the iteration is restarted with a smaller time window. The window is also reduced if the Schwarz iteration fails to converge in 6 iterations and enlarged if the iteration converges in fewer than 4 iterations. These values, chosen on the basis of significant experimentation [27], provide a balance between efficiency and storage. The latter can be significant even in the one-dimensional case because in the absence of a guarantee that the maximum principle holds, we conservatively require successive approximations over subdomains to agree to within some user specified tolerance δ before the waveform iteration is terminated. A tolerance of $\delta = 1e - 4$ is used in the experiments with local error control tolerance (for the time integration) of $1e - 6$.

As a final note, the code provides each subdomain with an initial equidistributed mesh. That is, we solve $u_t = u_0(x)$ for $x \in \tilde{\Omega}_j$ and a moving mesh PDE subject to $u(x, 0) = 0$. This provides a grid which equidistributes the initial solution to the PDE over each subdomain.

6. Numerical results.

To provide some initial testing of the moving Schwarz method we first solve a straightforward test problem.

Example 1: *Burgers' equation* I. Our first example is a standard test problem for adaptive methods, Burgers' equation. Specifically, we solve the one-dimensional viscous Burgers' equation

$$u_t = \epsilon u_{xx} - \frac{1}{2}(u^2)_x,$$

$$u(0, t) = 1, \quad u(1, t) = 0,$$

$$u(x, 0) = c - \frac{1}{2}\tanh\left(\frac{(x - x_0)}{4\epsilon}\right),$$

where $c = 1/2$, $x_0 = 1/4$ or $1/10$, and $\epsilon \ll 1$. The solution is a traveling front of thickness $O(\epsilon)$ which moves to the right from x_0 at speed c. The initial and boundary conditions are not compatible at $t = 0$ and $x = 0, 1$, but the discrepancy is small for small ϵ. This does not pose any difficulties with the numerics.

We take $\epsilon = 1e - 4$ and use 40 mesh points in each of the three equal sized subdomains. The top row of plots in Figure 6.1 illustrates the computed solution at times $t = 0.25, 0.45$, and 1.7. The bottom plots show the corresponding pointwise error $|\tilde{u}(x_i, t^*) - u(x_i, t^*)|$, where $\tilde{u}$ is the numerical approximation to the exact solution u obtained using the Schwarz waveform moving mesh method. As expected we see a sharp front moving from left to right. The mesh trajectories as functions of time are displayed in Figure 6.2.

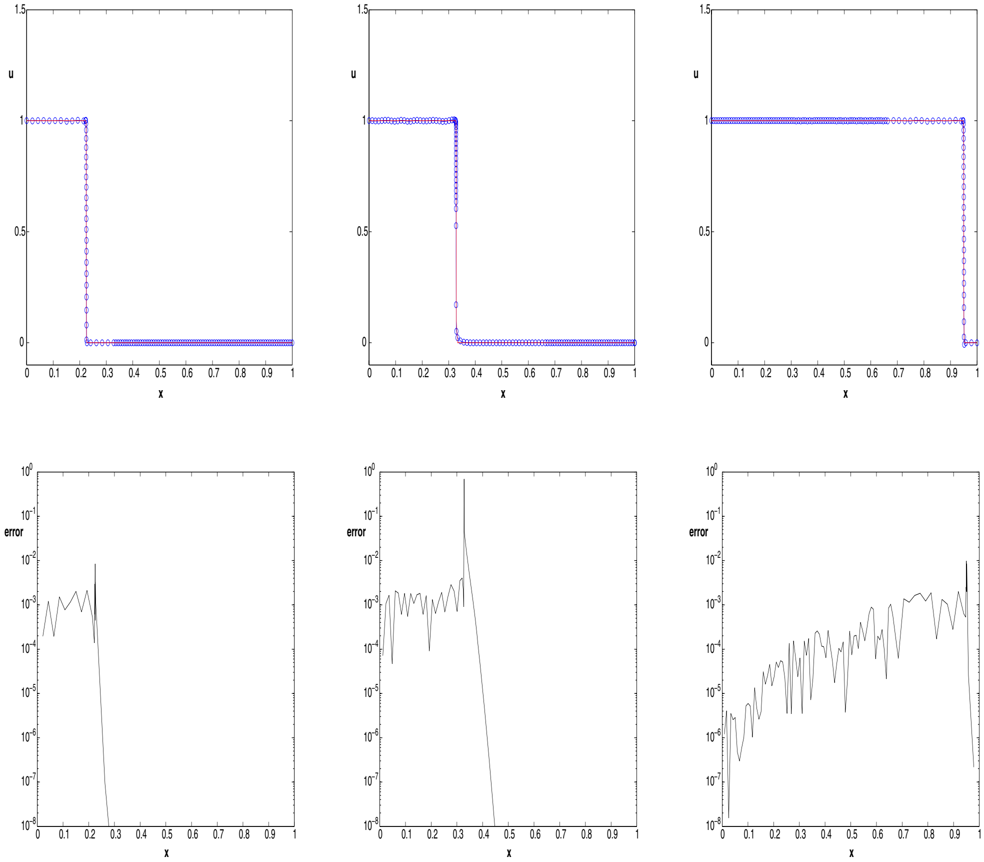

FIG. 6.1. *Solutions and pointwise errors for Burgers' equation with the Schwarz waveform moving mesh method at $t = 0.25$ (left), $t = 0.45$ (middle), and $t = 1.7$ (right).*

At $t = 0.25$ and $t = 1.7$, the actual errors in the solution are on the order of the tolerance controlling the convergence of the Schwarz waveform iteration. At $t = 0.45$, however, the front encounters the boundary between the first and second subdomains, and the level of communication between the solutions in these adjacent subdomains is inadequate to preserve the accuracy. A possible fix may be provided by some of the more sophisticated Schwarz waveform methods which use "higher-order" transmission conditions at the boundary [20, 21].

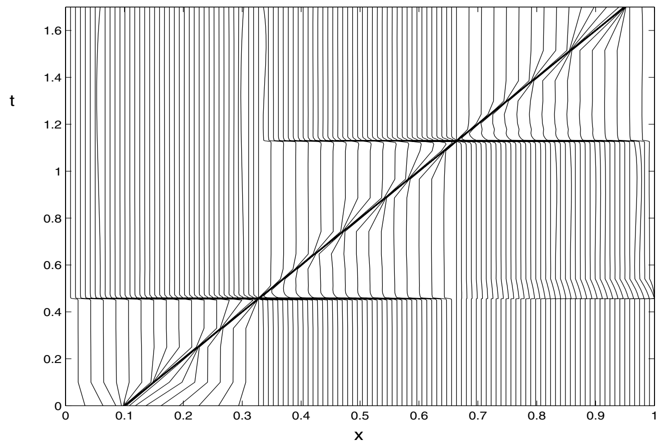

FIG. 6.2. *Mesh trajectories for Burgers' equation with* $\epsilon = 1e - 4$ *and* 40 *points per domain using the Schwarz waveform moving mesh method.*

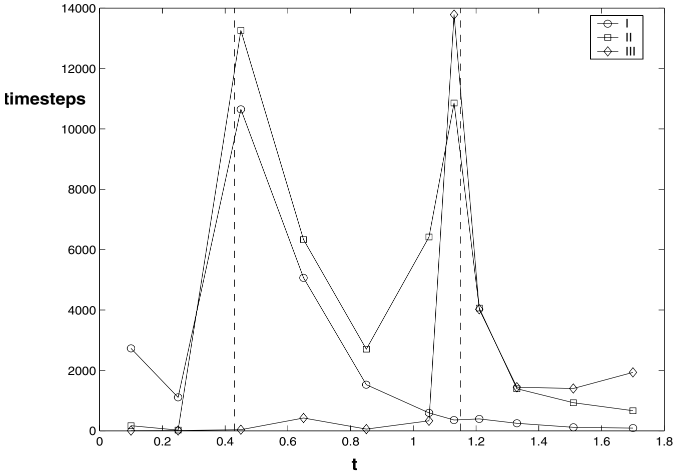

FIG. 6.3. *Number of time steps (per time window) in each subdomain.*

Both the tremendous potential and current difficulties with our basic Schwarz waveform moving mesh method are highlighted in Figure 6.3. In this plot we display the number of time steps per time window taken by implicit Euler in each subdomain, labeled I, II, and III in the legend of the plot. The dotted vertical lines specify the time at which the layer crosses the subdomain boundaries. The data shows that while the layer is completely contained in subdomain I ($0 \leq t \leq 0.4$), the work involved to integrate over subdomains II and III is negligible. As the front approaches the subdomain boundaries, the number of time steps in the adjoining domains increases dramatically, indicating the standard difficulty experienced by the classical Schwarz waveform iteration.

Another indicator of the performance of the classical Schwarz method is illustrated in Figure 6.4. The maximum number of time steps per time window is fixed, so the

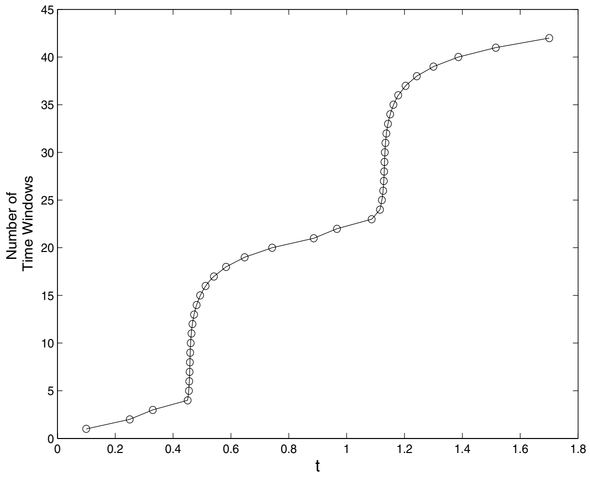

FIG. 6.4. *Number of time windows used up to time t for the Schwarz waveform moving mesh method applied to Burgers' equation with* $\epsilon = 1e - 4$.

size of the time window varies over the duration of the computation. There is a dramatic decrease in the size of the time window as the moving front moves from one subregion to the next. The time window is increased or decreased based on the convergence of the Schwarz iteration. If the iteration has not converged within a user specified number of iterations, the length of the time window is decreased. We see in Figure 6.4 that the length of the time window is shortest when the front moves from one subdomain to another.

Example 2: *Burgers' equation* II. We now solve Burgers' equation subject to the initial condition

$$u(x,0) = \sin(2\pi x) + \frac{1}{2}\sin(\pi x)$$

and boundary conditions $u(0,t) = u(1,t) = 0$, using three overlapping subdomains. The solution develops a sharp front and moves to the right with diminishing amplitude. Mesh trajectories for this initial condition are shown in Figure 6.5. In this case, the solution evolves to a front in subdomain II. This results in a nearly uniform mesh in subdomain I for the entire computation. The grids in subdomains II and III, however, react to the evolving solution and carry the front to the boundary at $x = 1$. As with the previous example, the convergence of the Schwarz iteration deteriorates as the front moves from subdomain II to subdomain III.

Example 3: *Two-spike problem.* The next problem is

$$u_t = \epsilon u_{xx} + f(x,t),$$

where $f(x,t)$ is defined such that the solution has two spikes and is given by

$$u(x,t) = \frac{1}{\epsilon}\left(\max(0, \tanh(t - t_0))e^{-(x-x_0)^2/\epsilon} + \max(0, \tanh(t - t_1))e^{-(x-x_1)^2/\epsilon}\right).$$

These two spikes are centered at fixed locations x_0 and x_1 and begin to evolve at different times t_0 and t_1. We choose x_0 and x_1 to keep the spikes well separated in the spatial domain and choose $t_1 > t_0$. This allows the first spike to grow to its

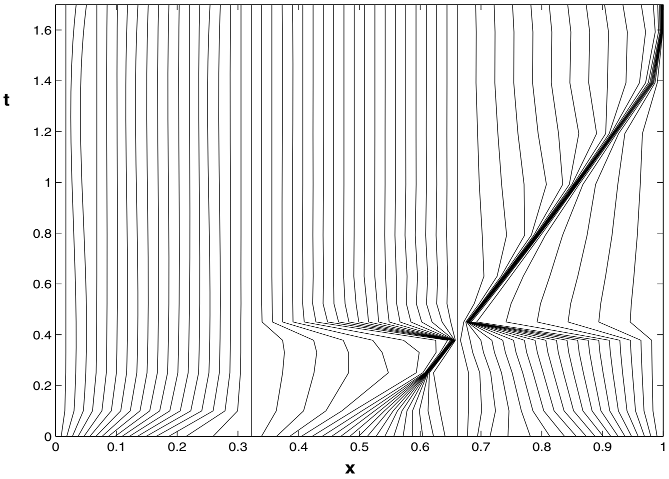

FIG. 6.5. *Mesh trajectories for Burgers' equation with* $\epsilon = 1e - 3$ *and* 20 *points per domain using the Schwarz waveform moving mesh method.*

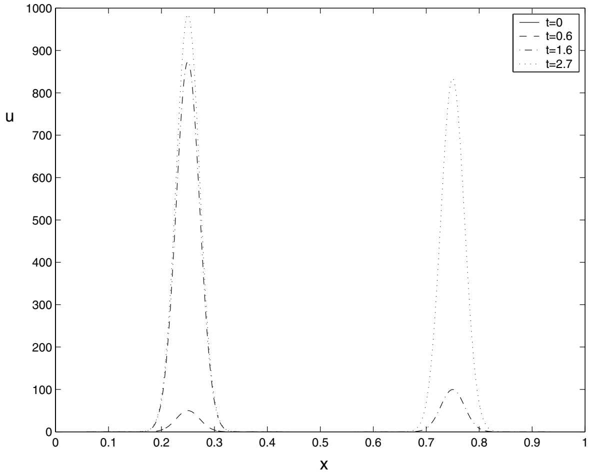

FIG. 6.6. *Exact solutions of the two-spike problem at* $t = 0, 0.6, 1.6,$ *and* 2.7.

maximum height before the second spike appears. All computations are done with $\epsilon = 1e-3$. Figure 6.6 illustrates the exact solution of the two-spike problem for times $t = 0.6$, 1.6, and 2.7. (At $t = 0$, $u(x, 0) \equiv 0$.) At $t = 0.6$ the spike centered at $x_0 = 1/4$ has appeared and continues to grow. The second spike centered at $x_1 = 3/4$ emerges at $t = 3/2$. At $t = 2.7$, both spikes have nearly reached their maximum height. This problem is chosen to show the difficulties which can occur for moving mesh methods when the solution to the PDE behaves badly in several regions, and how the Schwarz waveform moving mesh method can be used to ameliorate the situation. Solving the problem with 80 mesh points on one domain, the final solution at $t = 2.7$ and the mesh trajectories on $[0, 1]$ are shown in Figure 6.7. The computed solution is illustrated by a line with open circles indicating the location of the mesh points, and the exact solution is depicted with a thick solid line. The moving mesh method does indeed

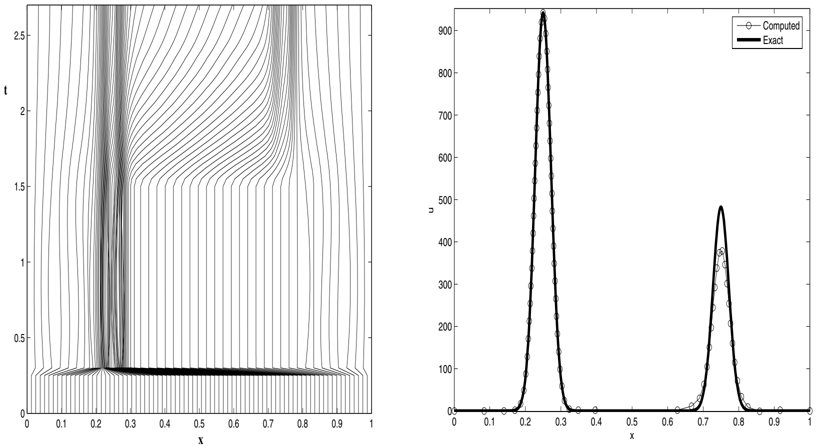

FIG. 6.7. *Mesh trajectories (left) and solutions (right) of the two-spike problem with one domain using the arclength monitor function.*

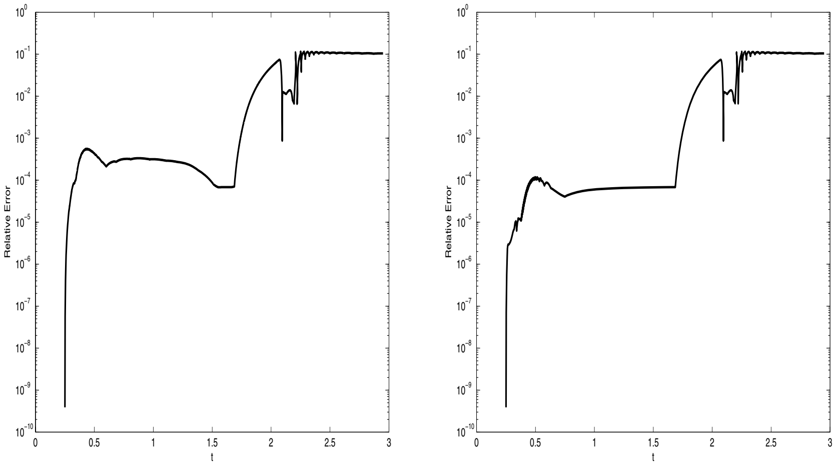

FIG. 6.8. *Relative error in the computed solution on one domain using the arclength monitor function and backward Euler (left) and variable order BDF formulation (right).*

capture both spikes; however, a loss of accuracy in the second spike is evident by the end of the computation. In fact, substantial errors arise in the computed solution for $t \geq 3/2$, corresponding to the birth of the second spike. Indeed, a moving mesh method on one domain has great difficulty achieving sufficient resolution of the second spike. The mesh trajectories indicate a relatively quick movement of mesh points from the region of the left spike towards the emerging right spike for $t > 3/2$.

Figure 6.8 illustrates the relative error in the computed solution for the two-spike problem on one domain. The plot on the left was obtained using implicit Euler to integrate forward in time, while the plot on the right was obtained using a variable order BDF implementation. The results are nearly identical. The loss of accuracy in the computed solutions arises with the birth of the second spike at $t = 1.5$. In fact, a relatively large number of mesh points is vital to allow resolution of the second spike.

The loss of accuracy which occurs as the second spike grows is due, in part, to the time required for the mesh to adapt to new features in the solution. This may be controlled by the moving mesh parameter τ in MMPDE4. Decreasing the value of τ facilitates a quicker mesh movement and hence a quicker return to an equidistributed grid. However, this improves the accuracy of the computed solution at the expense of necessitating much smaller steps in the time integration. The situation may be improved by controlling the proportion of points outside the initial region of difficulty by an appropriate choice of α in (4.3).

This problem is ideal for a two-domain simulation with the Schwarz waveform moving mesh method. The computed and exact solutions and resulting mesh trajectories are shown in Figure 6.9.

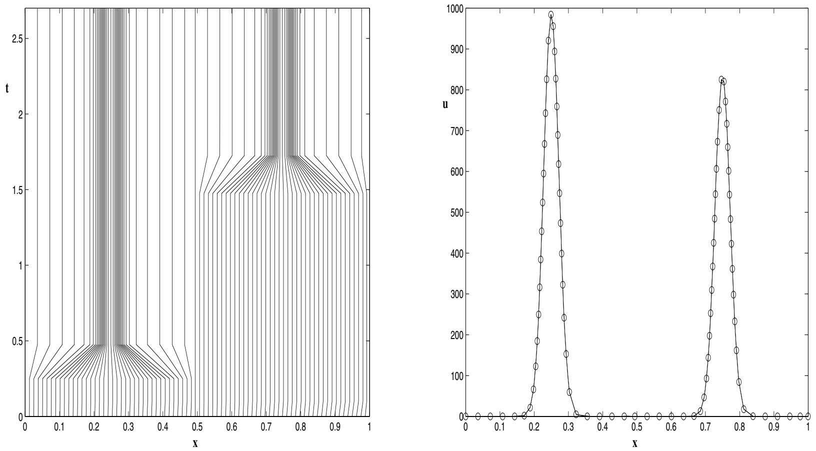

FIG. 6.9. *Mesh trajectories (left), computed and exact solutions (right) for the two-spike problem on two subdomains.*

Compared to the one-domain calculation, we see improved accuracy between the spikes and in regions of high curvature. As expected, the mesh remains uniform in each subdomain until the spike in that region is activated. At that point the mesh in the subdomain adapts well to the evolving features of the solution using a simple arclength monitor function and moderate values of τ. In fact, comparable accuracy to that for the one-domain calculation is possible with many fewer mesh points.

Figure 6.10 compares the time steps taken by the ODE solver (backward Euler) for the one-domain (left) and two-subdomain (right) simulations. In the two-subdomain case we illustrate the time steps for subdomain I (solid) and subdomain II (dashed) corresponding to the first waveform iteration. The time steps chosen for the one-domain calculation are quite large until $t = 1/4$, when the left spike begins to grow. The time steps remain relatively steady at 10^{-4} until the second spike emerges at $t = 3/2$. Immediately the time steps are reduced by an order of magnitude as the mesh points race to adjust to the emerging features in the solution.

The time steps chosen in each subdomain by the Schwarz waveform moving mesh method are controlled primarily by the local features of the solution. During the solution on subdomain I the time steps reduce to approximately 10^{-4} at $t = 1/4$. This time corresponds to the emergence of the first spike. The time steps persist at

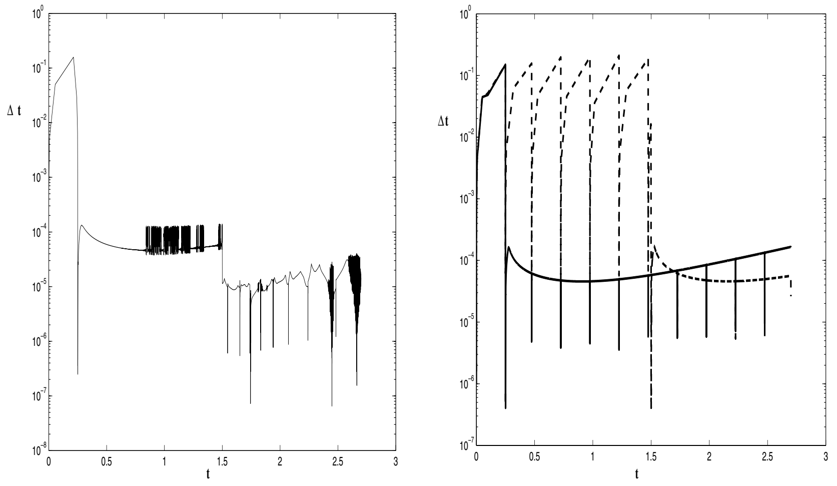

FIG. 6.10. *Time steps for the one-domain (left) and two-subdomain (right) solution of the two-spike problem.*

this level for the duration of the simulation. Little work is required to integrate the solution and mesh components in subdomain II until $t = 3/2$. At that time, the size of the acceptable time steps decreases to a level commensurate with that on subdomain I. It is important to note that the time steps in subdomain I are not affected by the development of the second spike. No mesh points move from one subdomain to another. This keeps the time steps an order of magnitude larger than for the one-domain calculation. This more than compensates for the cost of the iteration required in the Schwarz waveform moving mesh method, resulting in approximately half the run time.

7. Conclusions.

The moving method of lines has proved to be an effective tool for solving a number of time dependent PDEs [31]. In this paper we propose a modification of the basic procedure—a Schwarz waveform moving mesh method. The moving mesh PDE and physical PDE are solved by a domain decomposition algorithm. The computational space-time domain is decomposed into a number of overlapping subdomains. The solution is computed iteratively by solving the system of PDEs on each subdomain with information exchange achieved through Dirichlet conditions at the subdomain boundaries. Problems with localized fast-changing solution components are shown to benefit from this approach.

Traditionally, the solution of the resulting system of ODEs in time has been obtained using standard implementations of stiff ODE system solvers with a fully implicit time stepping package, such as in [31] where DASSL [39] is used.

In the simple implementation of the Schwarz waveform method investigated here, we solve (4.7) using a backward Euler method,

$$V(y_n)(y_n - y_{n-1}) - h_n g(y_n) = 0,$$

and a crude time-stepping procedure where the local error is controlled by step doubling/halving. The nonlinear algebraic systems are solved with a modified Newton method, using the strategies of Gustafsson and Söderlind [26] and Alexander [1] to control factorizations and evaluations of the Jacobian matrix. The Jacobian is evaluated using finite differences adapted to sparse matrices [8].

There are many important aspects to consider when investigating the feasibility of the Schwarz waveform method. Classical Schwarz iteration is known to suffer from slow convergence, particularly when the solution components have substantially different time scales. The moving mesh method is specifically designed so that solution components are closer to having the same time scale than for the standard MOL approach. It remains an issue, although we have seen marked improvement in numerical results. The slow convergence can be alleviated through the use of higher-order transmission conditions at the subdomain boundaries [20]. Furthermore, to avoid fast-changing solution components crossing subdomain boundaries, a coarse grain adaptivity may be employed as well, where one dynamically determines (spatially) the number and placement of subdomains. As a front approaches a subdomain boundary, for example, domains may be collapsed or merged until the front has passed. In our implementation, the Schwarz iteration proceeds until the solutions on all subdomains agree to within some specified tolerance. This requires storage of the solution at all points in the computational space. When a maximum principle holds, it is possible to save substantially on storage by keeping boundary data only to test for convergence. Further study is certainly required to understand the role of overlap for Schwarz waveform moving mesh methods.

The issues of efficiency are even more important in higher spatial dimensions, where fast solution behavior locally comes at a much higher cost if adaptivity is not done properly or system solutions not done efficiently; cf. [47]. One major reason that moving mesh methods are ideally suited to handle certain problems with singularities is that, by choosing a suitable monitor function to incorporate scaling as proposed by Beckett and Mackenzie [3], a proper balance between adaptivity done in the inner and outer solution regions is assured. Unfortunately, for problems with multiple scales or multiple regions with fast scale behavior, using moving mesh methods alone for adaptivity can be problematic in determining the relative importance of the regions of fast solution activity. While a simple implementation of the Schwarz waveform method is not a panacea in and of itself, it can be the ideal tool for overcoming many of these difficulties. A simple Schwarz waveform relaxation has been applied directly for the mesh equation for a two-dimensional problem in [6]. We intend to investigate a more sophisticated implementation of the Schwarz waveform method, incorporating the features discussed here, in future work. An ultimate test of the method would entail a careful comparison with the other approaches of section 3.

REFERENCES

[1] R. Alexander, *The modified Newton method in the solution of stiff ordinary differential equations*, Math. Comp., 57 (1991), pp. 673–701.

[2] J. F. Andrus, *Numerical solution of systems of ordinary differential equations separated into subsystems*, SIAM J. Numer. Anal., 16 (1979), pp. 605–611.

[3] G. Beckett and J. A. Mackenzie, *Uniformly convergent high order finite element solutions of a singularly perturbed reaction-diffusion equation using mesh equidistribution*, Appl. Numer. Math., 39 (2001), pp. 31–45.

[4] C. J. Budd, W. Huang, and R. D. Russell, *Moving mesh methods for problems with blow-up*, SIAM J. Sci. Comput., 17 (1996), pp. 305–327.

[5] X. Cai, *Additive Schwarz algorithms for parabolic convection-diffusion equations*, Numer. Math., 60 (1991), pp. 41–61.

[6] W. Cao, W. Huang, and R. D. Russell, *A moving mesh method in multiblock domains with application to a combustion problem*, Numer. Methods Partial Differential Equations, 15 (1999), pp. 449–467.

[7] T. F. Chan and T. P. Mathew, *Domain decomposition algorithms*, in Acta Numerica, 1994, Acta Numer., Cambridge University Press, Cambridge, UK, 1994, pp. 61–143.

[8] A. Curtis, M. Powell, and J. Reid, *On the estimation of sparse Jacobian matrices*, IMA J. Appl. Math., 13 (1974), pp. 117–119.

[9] C. de Boor, *Good approximation by splines with variable knots.* II, in Conference on the Numerical Solution of Differential Equations (Dundee, 1973), Lecture Notes in Math. 363, Springer, Berlin, 1974, pp. 12–20.

[10] P. Deuflhard and J. Heroth, *Dynamic dimension reduction in ODE models*, in Scientific Computing in Chemical Engineering, F. Keil, W. Mackens, H. Voß, and J. Werther, eds., Springer-Verlag, Berlin, 1996, pp. 29–43.

[11] C. Engstler and C. Lubich, *Multirate extrapolation methods for differential equations with different time scales*, Computing, 58 (1997), pp. 173–185.

[12] C. Engstler and C. Lubich, *MUR8: A multirate extension of the eighth-order Dormand-Prince method*, Appl. Numer. Math., 25 (1997), pp. 185–192.

[13] J. Flaherty, R. Loy, M. Shephard, B. Szymanski, J. Teresco, and L. Ziantz, *Adaptive local refinement with octree load-balancing for the parallel solution of three-dimensional conservation laws*, J. Parallel Distrib. Comput., 47 (1997), pp. 139–152.

[14] J. E. Flaherty and P. K. Moore, *Integrated space-time adaptive hp-refinement methods for parabolic systems*, Appl. Numer. Math., 16 (1995), pp. 317–341.

[15] M. J. Gander and C. Rohde, *Overlapping Schwarz waveform relaxation for convection-dominated nonlinear conservation laws*, SIAM J. Sci. Comput., 27 (2005), pp. 415–439.

[16] M. J. Gander, *Optimized Schwarz methods*, SIAM J. Numer. Anal., 44 (2006), pp. 699–731.

[17] M. J. Gander, *Overlapping Schwarz waveform relaxation for parabolic problems*, in Tenth International Conference on Domain Decomposition Methods, Contemp. Math. 218, J. Mandel, C. Farhat, and X.-C. Cai, eds., AMS, Providence, RI, 1998, pp. 425–431.

[18] M. J. Gander, *A waveform relaxation algorithm with overlapping splitting for reaction diffusion equations*, Numer. Linear Algebra Appl., (1998), pp. 125–145.

[19] M. J. Gander and D. Daoud, *Overlapping Schwarz waveform relaxation for convection reaction diffusion problems*, in Thirteenth International Conference on Domain Decomposition Methods, N. Debit, M. Garbey, R. Hoppe, J. Périaux, D. Keyes, and Y. Kuznetsov, eds., Domain Decomposition Press, Bergen, Norway, 2002, pp. 227–233.

[20] M. J. Gander, L. Halpern, and F. Nataf, *Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation*, in Eleventh International Conference on Domain Decomposition Methods, C.-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, eds., ddm.org, Augsburg, Germany, 1999, pp. 27–36.

[21] M. J. Gander, L. Halpern, and F. Nataf, *Optimized Schwarz methods*, in Twelfth International Conference on Domain Decomposition Methods (Chiba, Japan), T. Chan, T. Kako, H. Kawarada, and O. Pironneau, eds., Domain Decomposition Press, Bergen, Norway, 2001, pp. 15–28.

[22] M. J. Gander and A. M. Stuart, *Space-time continuous analysis of waveform relaxation for the heat equation*, SIAM J. Sci. Comput., 19 (1998), pp. 2014–2031.

[23] M. J. Gander and H. Zhao, *Overlapping Schwarz waveform relaxation for the heat equation in n dimensions*, BIT, 42 (2002), pp. 779–795.

[24] C. W. Gear and D. R. Wells, *Multirate linear multistep methods*, BIT, 24 (1984), pp. 484–502.

[25] E. Giladi and H. B. Keller, *Space-time domain decomposition for parabolic problems*, Numer. Math., 93 (2002), pp. 279–313.

[26] K. Gustafsson and G. Söderlind, *Control strategies for the iterative solution of nonlinear equations in ODE solvers*, SIAM J. Sci. Comput., 18 (1997), pp. 23–40.

[27] R. D. Haynes, *The Numerical Solution of Differential Equations: Grid Selection for Boundary Value Problems and Adaptive Time Integration Strategies*, Ph.D. thesis, Simon Fraser University, Burnaby, BC, 2003.

[28] E. Hofer, *A partially implicit method for large stiff systems of ODEs with only few equations introducing small time-constants*, SIAM J. Numer. Anal., 13 (1976), pp. 645–663.

[29] W. Huang, Y. Ren, and R. D. Russell, *Moving mesh methods based on moving mesh partial differential equations*, J. Comput. Phys., 113 (1994), pp. 279–290.

[30] W. Huang, Y. Ren, and R. D. Russell, *Moving mesh partial differential equations (MMPDEs) based on the equidistribution principle*, SIAM J. Numer. Anal., 31 (1994), pp. 709–730.

[31] W. Huang and R. D. Russell, *A moving collocation method for solving time dependent partial differential equations*, Appl. Numer. Math., 20 (1996), pp. 101–116.

[32] J. Janssen and S. Vandewalle, *Multigrid waveform relaxation on spatial finite element meshes: The continuous-time case*, SIAM J. Numer. Anal., 33 (1996), pp. 456–474.

[33] R. Jeltsch and B. Pohl, *Waveform relaxation with overlapping splittings*, SIAM J. Sci. Comput., 16 (1995), pp. 40–49.

[34] Y. Kuznetsov, *Domain decomposition methods for unsteady convection-diffusion problems*, in Computing Methods in Applied Sciences and Engineering (Paris, 1990), SIAM, Philadelphia, 1990, pp. 211–227.

[35] E. Lelarasmee, A. Ruehli, and A. Sangiovanni-Vincentelli, *The waveform relaxation method for time-domain analysis of large scale integrated circuits*, IEEE Trans. CAD of Integrated Circuits Syst., 1 (1982), pp. 131–145.

[36] C. Lubich and A. Ostermann, *Multigrid dynamic iteration for parabolic equations*, BIT, 27 (1987), pp. 216–234.

[37] G. A. Meurant, *A domain decomposition method for parabolic problems*, Appl. Numer. Math., 8 (1991), pp. 427–441.

[38] U. Miekkala and O. Nevanlinna, *Convergence of dynamic iteration methods for initial value problems*, SIAM J. Sci. Statist. Comput., 8 (1987), pp. 459–482.

[39] L. R. Petzold, *A description of DASSL: A differential/algebraic system solver*, in Scientific Computing (Montreal, 1982), IMACS Trans. Sci. Comput. I, IMACS, New Brunswick, NJ, 1983, pp. 65–68.

[40] A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Numer. Math. Sci. Comput., Oxford Science Publications, The Clarendon Press, Oxford University Press, New York, 1999.

[41] Y. Ren and R. D. Russell, *Moving mesh techniques based upon equidistribution, and their stability*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1265–1286.

[42] J. R. Rice, *Split Runge-Kutta method for simultaneous equations*, J. Res. Nat. Bur. Standards Sect. B, 64B (1960), pp. 151–170.

[43] S. Skelboe, *Stability properties of backward differentiation multirate formulas. Recent theoretical results in numerical ordinary differential equations*, Appl. Numer. Math., 5 (1989), pp. 151–160.

[44] S. Skelboe and P. U. Andersen, *Stability properties of backward Euler multirate formulas*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 1000–1009.

[45] J. M. Stockie, J. A. Mackenzie, and R. D. Russell, *A moving mesh method for one-dimensional hyperbolic conservation laws*, SIAM J. Sci. Comput., 22 (2001), pp. 1791–1813.

[46] S. Ta'asan and H. Zhang, *On the multigrid waveform relaxation method*, SIAM J. Sci. Comput., 16 (1995), pp. 1092–1104.

[47] A. Toselli and O. Widlund, *Domain Decomposition Methods—Algorithms and Theory*, Springer Ser. Comput. Math. 34, Springer-Verlag, Berlin, 2005.