ALTERNATING-DIRECTION GALERKIN METHODS FOR PARABOLIC, HYPERBOLIC AND SOBOLEV PARTIAL DIFFERENTIAL EQUATIONS

Richard E. Ewing
Mobil Research and Development Corporation
Field Research Laboratory
P. O. Box 900
Dallas, Texas 75221

Abstract

A survey of some recent results in the use of alternating-direction finite element methods for linear and nonlinear partial differential equations of parabolic, hyperbolic, and Sobolev type is presented. These equations have applications to fluid flow in porous media, thermodynamics, wave propagation, nonlinear viscoelasticity, and hydrodynamics. The use of alternating-direction or operator-splitting methods will reduce multidimensional problems to repeated solution of one-dimensional problems. Thus optimal order work estimates can be obtained in all cases. Other new high-order and computationally efficient time-stepping procedures are also discussed and used as base schemes for the alternating-direction variants.

ALTERNATING-DIRECTION GALERKIN METHODS FOR PARABOLIC, HYPERBOLIC, AND SOBOLEV PARTIAL DIFFERENTIAL EQUATIONS

1. INTRODUCTION

In this paper, we shall present a survey of some recent results in the use of alternating-direction Galerkin methods for a variety of partial differential equations. We shall discuss methods for time-stepping partial differential equations of parabolic, hyperbolic, and Sobolev types in two and three spatial dimensions. The use of alternating-direction or operator-splitting methods will reduce multidimensional problems to repeated solution of one-dimensional problems. Thus optimal order work estimates can be obtained in all alternating-direction methods.

We shall basically consider only Galerkin or finite element alternating-direction (henceforth called AD) methods in this paper. Similar results can also be obtained for finite difference versions of our methods. Since the analysis of our methods will appear elsewhere, we shall only describe the methods in this manuscript and reference the analysis.

Alternating-direction methods were first used for time-dependent problems in the context of reservoir engineering models for fluid flow in porous media. The methods were developed in order to treat large scale multidimensional problems in a one-dimensional fashion on the small early-generation computers. Finite difference methods were developed for linear parabolic problems and analyzed thoroughly by Douglas, Peaceman, Rachford and others (see [10, 17, 18, 32]). Later Douglas and Dupont developed and analyzed a Laplace-modified Galerkin AD method for parabolic and hyperbolic equations with certain nonlinearities in [12]. These ideas were extended to stronger nonlinearities by Dendy in [8] and to unions of rectangular regions by Dendy and Fairweather in [9]. Then in [26, 27] Hayes extended these results to non-rectangular regions via patch approximations. In [28] Hayes and Percell extended these results to nonlinear capacity terms. Finally, in [11], Douglas discussed the combination of the results of [12, 28] with some of the iterative stabilization techniques presented in [14] to obtain other effective AD time-stepping procedures.

In this paper we shall discuss some recent advances in several different directions. First we discuss a tensor product projection of the solution into our computational subspaces and approximation theory results which greatly relax the smoothness assumptions required for all the earlier analysis of AD methods. Then we discuss some higher-order multistep time-stepping procedures which yield second, third, and in special cases fourth order time-truncation errors for parabolic problems. Previously, only second order methods with fairly strenuous coefficient constraints were known. We then extend the AD ideas to various partial differential equations of Sobolev type which are used in fluid flow in fractured media, thermodynamics, vibrational problems, nonlinear visocelasticity, and hydrodynamics (see [6, 7, 25, 29, 30, 31, 33, 34]). Finally we present some direct methods and iterative stabilization techniques which yield new, high-order and computationally efficient methods.

Let Ω be a bounded domain in R^d, $2 \le d \le 3$, with boundary $\partial\Omega$, and let $J = (0,T]$. We shall consider partial differential equations for $u = u(x,t)$ of the form

$$\text{a)}\quad e(x,u)\frac{\partial^2 u}{\partial t^2} + c(x,u)\frac{\partial u}{\partial t} - \nabla \cdot \left(a(x,u)\nabla u + b(x,u)\nabla \frac{\partial u}{\partial t} + g(x,u)\nabla \frac{\partial^2 u}{\partial t^2}\right) = f(x,t,u), \quad x\in\Omega,\ t\in J, \tag{1.1}$$

$$\text{b)}\quad u(x,t) = 0, \quad x\in\partial\Omega,\ t\in J,$$

$$\text{c)}\quad u(x,0) = u_0(x), \quad x\in\Omega,$$

for various choices of a, b, c, e, and g. If $e > 0$ and $g > 0$, we must also specify an additional initial condition of the form

$$u_t(x,0) = V(x), \quad x\in\Omega. \tag{1.2}$$

If $e = b = g = 0$ in (1.1) above, the equation is of parabolic type. This survey includes recent joint work by Jim Bramble and the author [3, 4] on problems of this type. If $e > 0$ and $c \ge b \ge g \ge 0$, the problems are of hyperbolic type. If $e > 0$ and either $b > 0$ or $g > 0$, the problems are of

Sobolev type. Joint work with Linda Hayes [22, 23] on problems of this type will be discussed.

In Section 2 we shall present some preliminaries and notation. We then illustrate the basic ideas of AD methods for various cases with constant coefficients in Section 3. In Section 4 we shall discuss higher-order direct methods which use the ideas of [8, 12, 26, 27, 28]. In Section 5 we discuss iterative stabilization ideas which use the ideas of [13, 14, 19, 20, 24]. We also discuss certain computational aspects of these methods.

II. PRELIMINARIES AND NOTATION

Let $(u,v) = \int_\Omega uv\,dx$ and $\|u\|^2 = (u,u)$. Let the norm on the Sobolev space $W^{k,p}(\Omega)$ be denoted by $\|u\|_{k,p}$, with the second index being suppressed if $p = 2$. Assume that $\partial\Omega$ is Lipschitz continuous. Assume that the coefficients and solutions are smooth; we refer to the various papers referenced for more precisely defined constraints.

For h from a sequence of small positive numbers, let $\{\overset{o}{M}_h[0,1]\}$ be a family of finite-dimensional subspaces of $W^{1,\infty}([0,1])$ which vanish at $x = 0$ and $x = 1$ and which satisfy:

For some integer $r \geq 2$ and some constant K_o and any

$$\phi \in W^{q,2}([0,1]) \cap W^{1,\infty}([0,1]),$$

$$\inf_{\chi \in \overset{o}{M}_h[0,1]} \left[\|\phi-\chi\|_o + h\|\phi - \chi\|_1 + h^{d/2}\left(\|\phi - \chi\|_{o,\infty} + h\|\phi - \chi\|_{1,\infty}\right)\right]$$

(2.1)

$$\leq K_o \|\phi\|_q h^q$$

for $1 \leq q \leq r + 1$.

An example of a family of subspaces satisfying (2.1) is the continuous subspace of piecewise polynomials of degree at most r on each subinterval of length h of a uniform partition of $[0,1]$.

We next define one-dimensional projection operators P_x, P_y, and P_z:

$W^{1,2} \to \overset{o}{M}_h[0,1]$ by

$$\text{a)} \quad \int_o^1 \frac{\partial}{\partial x}(P_x u - u)\frac{\partial}{\partial x}\chi\,dx = 0 \qquad , \chi \in \overset{o}{M}_h[0,1],$$

$$\text{(2.2)} \quad \text{b)} \quad \int_o^1 \frac{\partial}{\partial y}(P_y u - u)\frac{\partial}{\partial y}\chi\,dy = 0 \qquad , \chi \in \overset{o}{M}_h[0,1],$$

$$\text{c)} \quad \int_o^1 \frac{\partial}{\partial z}(P_z u - u)\frac{\partial}{\partial z}\chi\,dz = 0 \qquad , \chi \in \overset{o}{M}_h[0,1].$$

Next, let I_d denote the unit cube in R^d and define a sequence of subspaces on I_3 by

$$M_h \equiv M_h[I_3] \equiv \overset{\circ}{M}_h[0,1] \times \overset{\circ}{M}_h[0,1] \times \overset{\circ}{M}_h[0,1]. \tag{2.3}$$

We henceforth assume that $\Omega = I_3$ (or I_2 in R^2). See [9, 27] for techniques to extend these results to more general regions. We then define the three-dimensional tensor product projection $Z = P_x P_y P_z u$ in M_h. Note that the one-dimensional operators commute and thus can be taken in any order. Using (1.1.b), we can then obtain a very important orthogonality result.

Lemma 2.1: If $d = 2$ or $d = 3$, respectively,

$$\text{a)} \quad \left(\frac{\partial^2}{\partial x \partial y}(P_x P_y u - u), \frac{\partial^2}{\partial x \partial y}\chi\right) = 0 \qquad , \chi \in M_h[I_2],$$

$$\text{b)} \quad \left(\frac{\partial^3}{\partial x \partial y \partial z}(P_x P_y P_z u - u), \frac{\partial^3}{\partial x \partial y \partial z}\chi\right) = 0, \qquad , \chi \in M_h[I_3]. \tag{2.4}$$

We next define some other projections into M_h. If $a(x,u)$, $b(x,u)$, and $g(x,u)$ are bounded below by positive constants, let W_a, W_b, and W_g be the weighted elliptic projections satisfying:

$$\text{a)} \quad \left(a(x,u)\nabla(W_a - u), \nabla\chi\right) = 0 \qquad , \chi \in M_h,$$

$$\text{b)} \quad \left(b(x,u)\nabla(W_b - u), \nabla\chi\right) = 0 \qquad , \chi \in M_h, \tag{2.5}$$

$$\text{c)} \quad \left(g(x,u)\nabla(W_g - u), \nabla\chi\right) = 0 \qquad , \chi \in M_h.$$

Then, using the super-close approximation properties of the Galerkin solution in $W^{1,2}$ and Lemma 3.1 of [16], we obtain the following important result:

Lemma 2.2: For $Z = P_x P_y P_z u$ and W_a, W_b, and W_g defined in (2.5), we have for some $K_o > 0$,

$$(2.6) \qquad \|W_a - Z\|_1 + \|W_b - Z\|_1 + \|W_g - Z\|_1 \leq K_o \|u\|_{r+1} h^{r+1} .$$

Proof: (see [3]).

For $k > 0$, let $N = T/k \in \mathbb{Z}$ and $t^\sigma = \sigma k$, $\sigma \in \mathbb{R}$. Also let $\phi^n \equiv \phi^n(x) \equiv \phi(x,t^n)$. Define the following backward difference operators:

$$(2.7) \qquad \begin{aligned} &a) \quad \delta\phi^n = \phi^n - \phi^{n-1} \\ &b) \quad \delta^2\phi^n = \phi^n - 2\phi^{n-1} + \phi^{n-2} \\ &c) \quad \delta^3\phi^n = \phi^n - 3\phi^{n-1} + 3\phi^{n-2} - \phi^{n-3} \\ &d) \quad \delta^4\phi^n = \phi^n - 4\phi^{n-1} + 6\phi^{n-2} - 4\phi^{n-3} + \phi^{n-4} . \end{aligned}$$

III. DESCRIPTION OF THE METHODS - CONSTANT COEFFICIENTS

In this section we shall describe various methods for efficiently time-stepping the Galerkin spatial procedures for various forms of (1.1) with constant coefficients. We first consider the parabolic case of (1.1) where $e \equiv b \equiv g \equiv 0$ and c and a are positive constants:

$$c \frac{\partial u}{\partial t} - a \Delta u = f(x,t,u).$$

For this case, we first present several multistep methods which will form our base schemes. Next, we shall introduce terms which allow us to use AD ideas in space.

For various special choices of parameters, we define the following class of backward differentiation, multistep, discrete time methods. Let $U:\{t_o, \cdots, t_N\} \to M_h$ be an approximate solution of (1.1). Assume that U^k are known for $k \leq n$. Given a desired global time-truncation error of order k^μ, $\mu = 1, 2, 3, 4$, we choose parameters $\alpha_i(\mu)$, $i = 1, 2, 3$, and $\beta(\mu)$ and an extrapolation operator $E(\mu)$ for $f(x,t,u)$ to define a method for determining U^{n+1} which satisfies

$$\begin{aligned} & k^{-1}\left(c\delta U^{n+1}, \chi\right) + \beta\left(a\nabla U^{n+1}, \nabla\chi\right) \\ & \quad = k^{-1}\left(c\left[\alpha_1 \delta U^n + \alpha_2 \delta U^{n-1} + \alpha_3 \delta U^{n-2}\right], \chi\right) \\ & \quad + \beta\left(f\left(t^{n+1}, E(\mu) U^{n+1}\right), \chi\right) \qquad , \chi \in M_h . \end{aligned} \tag{3.1}$$

Choices of the parameters and extrapolation operator for $\mu = 1, \cdots, 4$ are given in Table 1. By extrapolating the values of U^k in the nonlinear term f, we have produced a linear operator equation for U^{n+1} in terms of previous known values of U^k, $k \leq n$. See [5, 21] for a detailed analysis of the stability and accuracy of these methods. We note that the case for $\mu = 2$ is not the second-order Crank-Nicolson method which has a characteristic bounce. Instead, all the methods presented here are dissipative and strongly stable.

We next consider AD variants of (3.1). Let U^{n+1} satisfy

$$
\begin{aligned}
& k^{-1}\left(c\delta U^{n+1}, \chi\right) + \beta\left(a\nabla U^{n+1}, \nabla\chi\right) + \frac{k\beta^2a^2}{c}\left[\left(\frac{\partial^2}{\partial x\partial y} D(\mu)\, U^{n+1}, \frac{\partial^2}{\partial x\partial y}\chi\right)\right. \\
& \quad + \left(\frac{\partial^2}{\partial x\partial z} D(\mu)\, U^{n+1}, \frac{\partial^2}{\partial x\partial z}\chi\right) + \left.\left(\frac{\partial^2}{\partial y\partial z} D(\mu)\, U^{n+1}, \frac{\partial^2}{\partial y\partial z}\chi\right)\right] \\
& \quad + \frac{k^2\beta^3a^3}{c^2}\left(\frac{\partial^3}{\partial x\partial y\partial z} D(\mu)\, U^{n+1}, \frac{\partial^3}{\partial x\partial y\partial z}\chi\right) \qquad (3.2) \\
& \quad = k^{-1}\left(c\left[\alpha_1\,\delta U^{n} + \alpha_2\,\delta U^{n-1} + \alpha_3\,\delta U^{n-2}\right], \chi\right) \\
& \quad + \beta\left(f\left(t^{n+1}, E(\mu)\, U^{n+1}\right), \chi\right) \qquad , \chi\in M_h ,
\end{aligned}
$$

where the operator $D(\mu)U^{n+1}$ makes the additional terms "small" enough so as not to increase the order of the errors already present in the approximations. For example, for $\mu = 1$ or $\mu = 2$, the choice $D(\mu)U^{n+1} = \delta U^{n+1}$ will yield convergent schemes. For $\mu = 3$, we shall use $D(3)U^{n+1} = \delta^2 U^{n+1}$. For the case $\mu = 4$, the choice $D(4)U^{n+1} = \delta^3 U^{n+1}$ would make the perturbation terms small enough for proper truncation error analysis, but will cause the method to be unstable. Instead, we shall choose

$$
D(4) = \delta^2\, U^{n+1} - cS_\gamma^{-1}\,\delta^2 U^{n} \qquad (3.3)
$$

with

$$
\begin{aligned}
S_\gamma &= \left(1 + k\gamma\left[\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}\right]\right. \\
&\quad + k^2\gamma^2\left[\frac{\partial^2}{\partial x\partial y} + \frac{\partial^2}{\partial x\partial z} + \frac{\partial^2}{\partial y\partial z}\right] + \left. k^3\gamma^3\,\frac{\partial^3}{\partial x\partial y\partial z}\right) \qquad (3.4) \\
&= \left(1 + k\gamma\,\frac{\partial}{\partial x}\right)\left(1 + k\gamma\,\frac{\partial}{\partial y}\right)\left(1 + k\gamma\,\frac{\partial}{\partial z}\right) .
\end{aligned}
$$

Since cS_γ^{-1} is comparable to the identity operator, this choice of D(4) acts like $\delta^3 U^{n+1}$, and γ is chosen sufficiently large to make the method stable.

The additional terms in (3.2) allow the operator to factor in a manner exactly as in (3.4) into a sequence of one-dimensional operators. Since the methods presented in (3.2) involve up to five time levels, special start-up procedures must be discussed. Higher-order start-up procedures for the methods described in (3.1) have been presented and analyzed in [4]; however, the procedures have not been shown to be effective for AD methods. Start-up procedures for cases $\mu = 1, 2, 3$ will appear in [3], but no procedure has been analyzed for the case $\mu = 4$ at this time. The AD methods of (3.2) yield the same order convergence rates as the multistep methods of (3.1) but yield optimal order work estimates as well.

Next, we consider other partial differential equations by making different choices of coefficients in (1.1). If $a > 0$, $e > 0$, and $c \equiv b \equiv g \equiv 0$, we have an equation of hyperbolic type:

$$e \frac{\partial^2 u}{\partial t^2} - \nabla \cdot (a(x,u) \nabla u) = f(x,t,u) .$$

AD methods of the form with $d = 2$

$$k^{-2} (e\delta^2 U^{n+1}, \chi) + (a \nabla U^n, \nabla \chi) + \lambda (\nabla \delta^2 U^{n+1}, \nabla \chi)$$

$$+ \frac{\lambda^2 k^2}{e} \left(\frac{\partial^2}{\partial x \partial y} \delta^2 U^{n+1}, \frac{\partial^2}{\partial x \partial y} \chi\right) = (f(t^n, U^n), \chi), \quad \chi \in M_h, \tag{3.5}$$

have been presented and analyzed in [8, 12]. The Laplace-modified ideas were presented and analyzed for both parabolic and hyperbolic equations in [12] and yield second order time-truncation estimates. Extensions to higher dimensions are straightforward as pointed out in [8]. However, since only the weighted elliptic projection (2.5.a) was used in the analysis, more smoothness on u was required than if $Z = P_x P_y P_z u$ and Lemmas 2.1 and 2.2 had been used.

Next we discuss results for equations of Sobolev type which will appear in [23]. We first consider the case with $a > 0$, $b > 0$ and $c > 0$ with $e \equiv g \equiv 0$ in (1.1):

$$c \frac{\partial u}{\partial t} - \nabla \cdot \left(a \nabla u + b \nabla \frac{\partial u}{\partial t}\right) = f(x,t,u) .$$

Equations of this form are studied in [19, 34]. Since equations of Sobolev type have a time derivative in the highest-order terms, they are in general inherently more stable than corresponding parabolic equations. However, the time derivatives in the highest-order terms also make the perturbation terms needed for AD variants much larger. Therefore three time levels will be required for $O(k)$ accuracy and four levels for $O(k^2)$ accuracy in this case. One method which has time-truncation errors of order k is:

$$
\begin{aligned}
& k^{-1}\left(c\left(U^{n+1}-U^{n-1}\right), \chi\right)+\left(a \nabla U^{n}, \nabla \chi\right) \\
& \quad+k^{-1}\left(b \nabla\left(U^{n+1}-U^{n-1}\right), \nabla \chi\right) \\
& \quad+\frac{b^{2}}{kc}\left[\left(\frac{\partial^{2}}{\partial x \partial y} \delta^{2} U^{n+1}, \frac{\partial^{2}}{\partial x \partial y} \chi\right)+\left(\frac{\partial^{2}}{\partial x \partial z} \delta^{2} U^{n+1}, \frac{\partial^{2}}{\partial x \partial z} \chi\right)\right. \\
& \quad\left.+\left(\frac{\partial^{2}}{\partial y \partial z} \delta^{2} U^{n+1}, \frac{\partial^{2}}{\partial y \partial z} \chi\right)\right]+\frac{b^{3}}{kc^{2}}\left(\frac{\partial^{3}}{\partial x \partial y \partial z} \delta^{2} U^{n+1}, \frac{\partial^{3}}{\partial x \partial y \partial z} \chi\right) \\
& \quad=\left(f\left(t^{n}, U^{n}\right), \chi\right) \qquad\qquad , \chi \in M_{h} .
\end{aligned} \tag{3.6}
$$

By replacing $\delta^2 U^{n+1}$ by $\delta^3 U^{n+1}$ everywhere in the above equation, we obtain a method which yields error estimates of the form

$$
\max_{t^{n}}\left\|U^{n}\right\| \leq K_{1}\left\{k^{2}+h^{r+1}\right\} \tag{3.7}
$$

for some positive constant K_1, using spaces with approximation properties given by (2.1). See [23] for analysis and computational discussion.

Finally we consider second-order Sobolev equations obtained by choosing $e > 0$, $c \equiv 0$, $a > 0$, $b > 0$, and $g > 0$ in (1.1):

$$
e \frac{\partial^{2} u}{\partial t^{2}}-\nabla \cdot\left(a \nabla u+b \nabla \frac{\partial u}{\partial t}+g \nabla \frac{\partial^{2} u}{\partial t^{2}}\right)=f(x, t, u) .
$$

Equations of this type arise in hydrodynamics and applications of viscoelasticity [6, 7, 25, 29, 30, 31, 33, 34] and numerical approximations have been

studied analytically in [20]. If $g > 0$, a method with four time levels is needed to obtain time-truncation errors of $O(k)$. This method is given by

$$
\begin{aligned}
&k^{-2}\left(e\delta^2 U^{n+1}, \chi\right) + \left(a \nabla U^n, \nabla\chi\right) + k^{-1}\left(b \nabla \left(U^{n+1} - U^{n-1}\right), \nabla \chi\right) \\
&\quad + k^{-2}\left(g \nabla \delta^2 U^{n+1}, \nabla \chi\right) + \frac{(kb + g)^2}{k^2 e}\left[\left(\frac{\partial^2}{\partial x \partial y} \delta^3 U^{n+1}, \frac{\partial^2}{\partial x \partial y} \chi\right)\right. \\
&\quad \left. + \left(\frac{\partial^2}{\partial x \partial z} \delta^3 U^{n+1}, \frac{\partial^2}{\partial x \partial z} \chi\right) + \left(\frac{\partial^2}{\partial y \partial z} \delta^3 U^{n+1}, \frac{\partial^2}{\partial y \partial z} \chi\right)\right] \\
&\quad + \frac{(kb + g)^3}{k^2 e^2}\left(\frac{\partial^3}{\partial x \partial y \partial z} \delta^3 U^{n+1}, \frac{\partial^3}{\partial x \partial y \partial z} \chi\right) \\
&\quad = \left(f\left(t^n, U^n\right), \chi\right) \qquad , \chi \in M_h \, .
\end{aligned} \tag{3.8}
$$

Note that if $g \equiv 0$ and $b > 0$, the $\delta^3 U^{n+1}$ terms in (3.8) can be replaced by $\delta^2 U^{n+1}$ terms to obtain a three level method which yields error estimates of the form

$$
\max_{t^n} \|U^n\| \leq K_1 \left\{k^2 + h^{r+1}\right\} \tag{3.9}
$$

for some constant K_1. For details and analysis, see [23].

IV. DIRECT METHODS

Now that the basic AD ideas have been presented in the constant coefficient case in R^3 we shall discuss methods for treating the nonlinear coefficients in (1 1) in R^2. Extensions to R^3 should be obvious. We shall first consider methods which we term direct methods which have been derived from the Laplace-modified ideas presented in [12] and used extensively in [8, 11, 12, 15, 27, 28].

Again, we first consider parabolic equations with $e \equiv b \equiv g \equiv 0$ in (1.1):

$$(4.0) \qquad c(x,u)\frac{\partial u}{\partial t} - \nabla \cdot (a(x,u)\nabla u) = f(x,t,u) .$$

The basic idea of direct methods is to replace the variable coefficients at the top time levels by a constant, or sequence of constants, which is "close" to the true coefficient. Then the error made by this replacement is multiplied by a "small" term obtained by extrapolations from previous time levels. Once constant coefficient values are obtained at the advanced time levels the AD procedures described in Section 3 can be applied.

Since many important problems have different-sized diffusion components in different directions, we shall not use only Laplace-modified methods but shall allow a direction-oriented modification. We then modify (3.1) as follows. Let c_o, a_1, and a_2 be fixed, let

$$(4.1) \qquad \begin{aligned} &a) \quad \tilde{c}^{n+1} = c\left(\vec{x}, E(\mu)U^{n+1}\right) - c_0 \\ &b) \quad \tilde{a}_1^{n+1} = a_x\left(\vec{x}, E(\mu)U^{n+1}\right) - a_1 \\ &c) \quad \tilde{a}_2^{n+1} = a_y\left(\vec{x}, E(\mu)U^{n+1}\right) - a_2 \end{aligned}$$

where a_x and a_y are the components of the vector a and let U^{n+1} satisfy

$$
\begin{aligned}
&k^{-1}\left(c\left(E(\mu)\,U^{n+1}\right)\delta U^{n+1},\chi\right)+\beta\{\left(a_x\left(E(\mu)\,U^{n+1}\right)\frac{\partial}{\partial x}U^{n+1},\frac{\partial}{\partial x}\chi\right)\\
&\quad+\left(a_y\left(E(\mu)\,U^{n+1}\right)\frac{\partial}{\partial y}U^{n+1},\frac{\partial}{\partial y}\chi\right)\\
&\quad+\frac{k\beta^2a_1a_2}{c_o}\left(\frac{\partial^2}{\partial x\partial y}D(\mu)\,U^{n+1},\frac{\partial^2}{\partial x\partial y}\chi\right)\\
&\quad=k^{-1}\{\left[\tilde{c}^{n+1}F(\mu)\,U^{n+1}+c_o\left\{\alpha_1\delta U^n+\alpha_2\delta U^{n-1}\right\}\right],\chi\right)\\
&\quad+\beta\left(\tilde{a}_1^{n+1}\frac{\partial}{\partial x}G(\mu)\,U^{n+1},\frac{\partial}{\partial x}\chi\right)+\beta\left(\tilde{a}_2^{n+1}\frac{\partial}{\partial y}G(\mu)\,U^{n+1},\frac{\partial}{\partial y}\chi\right)\\
&\quad+\beta\left(f\left(t^{n+1},E(\mu)\,U^{n+1}\right),\chi\right)\qquad,\ \chi\in M_h\ .
\end{aligned}
\tag{4.2}
$$

The choices of $\alpha_i(\mu)$, $i = 1, 2, 3$, $\beta(\mu)$ and $E(\mu)$ are given in Table 1 for $\mu = 1, 2, 3$. Choices of $D(\mu)$, $F(\mu)$ and $G(\mu)$ are given in Table 2 for methods with time-truncation errors of order k^{μ} for $\mu = 1$, 2, and 3. As an example, the case $\mu = 1$, can be written in the form

$$
\begin{aligned}
&k^{-1}\left(c_o\,\delta U^{n+1},\chi\right)+\left(a_1\frac{\partial}{\partial x}U^{n+1},\frac{\partial}{\partial x}\chi\right)+\left(a_2\frac{\partial}{\partial y}U^{n+1},\frac{\partial}{\partial y}\chi\right)\\
&\quad+\frac{ka_1a_2}{c_o}\left(\frac{\partial^2}{\partial x\partial y}\delta U^{n+1},\frac{\partial^2}{\partial x\partial y}\chi\right)\\
&\quad=k^{-1}\left(\left[c\left(U^n\right)-c_o\right]\delta U^n,\chi\right)-\left(\left[a_x\left(U^n\right)-a_1\right]\frac{\partial}{\partial x}U^n,\frac{\partial}{\partial x}\chi\right)\\
&\quad-\left(\left[a_y\left(U^n\right)-a_2\right]\frac{\partial}{\partial y}U^n,\frac{\partial}{\partial y}\chi\right)+\left(f\left(t^{n+1},U^n\right),\chi\right)\qquad,\ \chi\in M_h\ .
\end{aligned}
\tag{4.3}
$$

This equation has only constant coefficients at the advanced time level The operator for the advanced time level can thus be factored easily into a product of two one-dimensional operators. We note that the first-order method is similar to that discussed in [11, 26, 27]. The first second-order method from Table 2 is similar to the direct method discussed in [28], which has a Crank-Nicolson base scheme, but this method is strongly stable. Both of the aforementioned methods required constraints of the form

$$\text{a)}\quad \frac{3}{4}\, c\left(x,\, E(2)\, U^{n+1}\right) < c_o < \frac{5}{4}\, c\left(x,\, E(2)\, U^{n+1}\right)$$

(4.4)

$$\text{b)}\quad a\left(x,\, E(2)\, U^{n+1}\right) \leq a_o\ .$$

Although this is a very mild constraint on a_o it is a fairly restrictive two-sided constraint on c_o and is noted in Table 2. Certain patch approximation techniques presented in [26, 27, 28] help to make this constraint localized and thus less restrictive. Another second-order method which has only one-sided constraints but requires an extra matrix inversion at each time step is also presented in Table 2 and has been analyzed by Bramble and the author. If c is a positive constant, we have presented two third-order direct methods. The first has two-sided constraints on a_1 and a_2 while the second obtains one-sided constraints at greater computational expense as before. Analysis and details will appear elsewhere. Note that the operator S_Y appearing in Table 2 is given in (3.4).

In the analysis of all the methods presented by (4.2) and Table 2, the use of backward differentiation multistep base methods and the projection $Z = P_x P_y P_z u$ instead of the usual weighted elliptic projection allows very weak mesh-ratio conditions of the form:

$$\text{a)}\quad h^r \leq k \qquad\qquad \text{, for } d = 2\ ,$$

(4.5)

$$\text{b)}\quad c_1\, h^r \leq k \leq c h^{\frac{d}{2\mu}} \qquad \text{, for } d = 3\ .$$

The use of this projection also requires only the same smoothness for the AD variants as for the base schemes. Use of only the elliptic projection requires more smoothness in time than the results presented here (see [31]).

Using the ideas described above, we can also define AD methods for nonlinear Sobolev equations and wave equations. For example let $e \equiv g \equiv 0$ and a, b, and c be uniformly bounded from below by positive constants in (1.1):

$$c\,(x,u)\,\frac{\partial u}{\partial t} - \nabla \cdot \left(a\,(x,u)\,\nabla u + b\,(x,u)\,\nabla \frac{\partial u}{\partial t}\right) = f\,(x,t,u)\ .$$

We can then consider, for $\mu = 1,2$,

$$
\begin{aligned}
& k^{-1}\left(c\left(E(\mu)U^{n+1}\right)\delta U^{n+1}, \chi\right) + \beta\left\{\left(a_x\left(E(\mu)U^{n+1}\right)\frac{\partial}{\partial x}U^{n+1}, \frac{\partial}{\partial x}\chi\right)\right. \\
& \quad \left. + \left(a_y\left(E(\mu)U^{n+1}\right)\frac{\partial}{\partial y}U^{n+1}, \frac{\partial}{\partial y}\chi\right)\right\} \\
& \quad + k^{-1}\left\{\left(b_x\left(E(\mu)U^{n+1}\right)\frac{\partial}{\partial x}\delta U^{n+1}, \frac{\partial}{\partial x}\chi\right)\right. \\
& \quad \left. + \left(b_y\left(E(\mu)U^{n+1}\right)\frac{\partial}{\partial y}\delta U^{n+1}, \frac{\partial}{\partial y}\chi\right)\right\} \\
& \quad + \frac{(b_1 + k\beta a_1)(b_2 + k\beta a_2)}{kc_o}\left(\frac{\partial^2}{\partial x\partial y}D(\mu)U^{n+1}, \frac{\partial^2}{\partial x\partial y}\chi\right) \\
& = k^{-1}\left(\left[\tilde{c}^{n+1}F(\mu)U^{n+1} + c_o\alpha_1\delta U^n\right], \chi\right) \\
& \quad + k^{-1}\left(\left[\tilde{b}_1^{n+1}\frac{\partial}{\partial x}F(\mu)U^{n+1} + b_1\alpha_1\frac{\partial}{\partial x}\delta U^n\right], \frac{\partial}{\partial x}\chi\right) \\
& \quad + k^{-1}\left(\left[\tilde{b}_2^{n+1}\frac{\partial}{\partial y}F(\mu)U^{n+1} + b_2\alpha_1\frac{\partial}{\partial y}\delta U^n\right], \frac{\partial}{\partial y}\chi\right) \\
& \quad + \beta\left\{\left(\tilde{a}_1^{n+1}\frac{\partial}{\partial x}G(\mu)U^{n+1}, \frac{\partial}{\partial x}\chi\right) + \left(\tilde{a}_2^{n+1}\frac{\partial}{\partial y}G(\mu)U^{n+1}, \frac{\partial}{\partial y}\chi\right)\right\} \\
& \quad + \beta\left(f\left(t^{n+1}, E(\mu)U^{n+1}\right), \chi\right) \qquad\qquad , \chi\in M_h ,
\end{aligned}
\tag{4.6}
$$

where b_x, b_y, b_1, b_2, $\tilde{b}_1$, and $\tilde{b}_2$ are analogous to the corresponding coefficients for a (see (4.1)) and F, D, G, and E are from Table 2 as before. We note that the base scheme used for time-stepping the Sobolev equation here is a backward differentiation multistep method and is different from that used for similar equations in Section 3. Corresponding direct methods could be defined from the methods of Section 3. Analysis of (4.6) will appear in [22].

In a similar manner, direct methods could be used to obtain efficient AD methods for hyperbolic and second-order Sobolev equations where $e(x,u)$ is nonlinear in (1.1). Techniques like those used in [20] are required. Detailed descriptions and analysis of these methods will appear elsewhere.

V. ITERATIVE METHODS

In this section we discuss iterative stabilization methods for treating the nonlinearities in the coefficients as an alternative to direct methods. We shall use the ideas developed in [14, 19] and later used for multistep methods in [5, 21]. The basic idea for the base scheme is to factor the matrix arising from the linear algebra problem at one time-step, say the initial time-step. We then use this factored matrix as a preconditioner in a preconditioned conjugate gradient iterative procedure to keep from factoring a new matrix at each time step. This factored matrix is comparable to the matrix which should be inverted at each time level. Thus we can extrapolate from past values to obtain the proper accuracy and only iterate sufficiently often to stabilize the process. For many problems this requires only two to four iterations per time step. If the coefficients begin to change considerably, one should refactor to obtain a more comparable preconditioner periodically. For discussion of these computational complexities and work estimates, see [11, 14, 19, 20, 24].

The use of iterative stabilization in conjunction with AD methods was first presented in [11]. The factored operator S_Y from (3.4) was used as a preconditioner in a first-order time method. However, since the base method did not include AD perturbation terms as in (3.2), a mesh-ratio restriction of the form

$$k \leq K h^2 \quad , \text{ for } d = 2 \, , \tag{5.1}$$

is required in [11] in order that the preconditioner be comparable to the linear operator which should be solved at each time step. Since we include an AD perturbation term in our base scheme, we obtain comparability with the preconditioner with no mesh-ratio restrictions. The only mesh-ratio restrictions required by the methods presented here are the weak conditions given by (4.5).

The base scheme for the methods to be presented in this section for parabolic problems from (4.0) is

$$
\begin{aligned}
&k^{-1}\left(c\left(x, E(\mu)\, U^{n+1}\right) \delta U^{n+1}, \chi\right) \\
&\quad + \beta\left(a_x\left(x, E(\mu)\, U^{n+1}\right) \frac{\partial}{\partial x} U^{n+1}, \frac{\partial}{\partial x} \chi\right) \\
&\quad + \beta\left(a_y\left(x, E(\mu)\, U^{n+1}\right) \frac{\partial}{\partial y} U^{n+1}, \frac{\partial}{\partial y} \chi\right) \\
&\quad + \frac{k \beta^2 a_1 a_2}{c_o}\left(\frac{\partial^2}{\partial x \partial y} D(\mu)\, U^{n+1}, \frac{\partial^2}{\partial x \partial y} \chi\right) \\
&\quad = k^{-1}\left(c\left(x, E(\mu)\, U^{n+1}\right)\left[\alpha_1 \delta U^n + \alpha_2 \delta U^{n-1}\right], \chi\right) \\
&\quad + \beta\left(f\left(x, t^{n+1}, E(\mu)\, U^{n+1}\right), \chi\right) \qquad , \chi \in M_h^o ,
\end{aligned}
\tag{5.2}
$$

where a_1, a_2, and c_o are as in (4.1) and α_1, α_2, β_1, $E(\mu)$ and $D(\mu)$ are as in Table 1. We shall next define our iterative stabilization schemes.

We first present the linear equations arising from (5.2) for the case $\mu = 3$ and note that there is no direct AD factorization possible for these equations. This motivates the introduction of a fixed preconditioner for which the linear equations do have an AD factorization.

We define two orderings on the nodes in $\Omega = [0,1]^2$. The first is a global ordering which assigns one of the numbers 1, 2, $\cdots$, M to each node in Ω. The second is a tensor product ordering of the M nodes. Grid lines of the form $x = x_j$, $0 \leq x_j \leq 1$, are numbered 1, 2, $\cdots$, M_x while grid lines of the form $y = y_j$, $0 \leq y_j \leq 1$ are numbered 1, 2, $\cdots$, M_y. With each node i, we associate an x-grid line and a y-grid line. The tensor product index of the node i is the pair $(m(i), n(i))$, where $m(i)$ is the index of the x-grid line and $n(i)$ is the index of the y-grid line. We then denote the tensor product basis as

$$
B_i(\vec{x}) = \Phi_{m(i)}(x)\, \psi_{n(i)}(y) = \Phi_m(x)\, \psi_n(y) \qquad , 1 \leq i \leq M , \tag{5.3}
$$

where $\{\Phi_m(x)\}_{m=1}^{M_x}$ and $\{\psi_n(y)\}_{n=1}^{M_y}$ are bases for the one-dimensional spaces $\overset{o}{M}_h[0,1]$ for x or y in [0,1], respectively.

Let U^p from (5.2) be written as

$$U^p = \sum_{i=1}^{M} \xi_i^p B_i(\vec{x}) = \sum_{m=1}^{M_x} \sum_{n=1}^{M_y} \xi_{mn}^p \Phi_m(x) \psi_n(y) . \tag{5.4}$$

Using (5.4), (5.2) with $\mu = 3$ can be written as

$$\begin{aligned} L^{n+1}\{\xi^{n+1} - \xi^n\} &= C^n(\xi)\{\sum_{j=1}^{2} \alpha_j \delta \xi^{n+1-j}\} + k\{F_1^n(\xi) + F_2^n(\xi)\} \\ &\equiv F^n(\xi) \end{aligned} \tag{5.5}$$

where the matrices and vectors in (5.5) are defined by

$$\text{a)}\quad L^n = C^n + k A^n + k^2 G^n ,$$

$$\text{b)}\quad C^n = \left((c\,(E(3)\,U^{n+1})\,B_j, B_i\right) ,$$

$$\text{c)}\quad A^n = \beta\left((a_x\,(E(3)\,U^{n+1})\frac{\partial}{\partial x}B_j, \frac{\partial}{\partial x}B_i\right) + \left(a_y\,(E(3)\,U^{n+1})\frac{\partial}{\partial y}B_j, \frac{\partial}{\partial y}B_i\right)\right) ,$$

$$\text{d)}\quad G^n = \frac{\beta^2 a_1 a_2}{c_o}\left(\left(\frac{\partial^2}{\partial x \partial y}B_j, \frac{\partial^2}{\partial x \partial y}B_i\right)\right) , \tag{5.6}$$

$$\text{e)}\quad F_1^n(\xi) = -A^{n+1}\xi^{n+1} + \beta\left(\left(f\,(t^{n+1}, E(3)\,U^{n+1}), B_j\right)\right) ,$$

$$\text{f)}\quad F_2^n(\xi) = G^n\left[\xi^n - \xi^{n+1}\right] ,$$

for $i, j = 1, 2, \cdots, M$.

Instead of solving (5.5) exactly, we shall approximate its solution by using an iterative procedure which has been preconditioned by $\bar{L}^o$ the matrix (5.6.a) with c, a_x, and a_y replaced by c_o, a_1, and a_2, respectively. Since the matrix $\bar{L}^o$ has constant coefficients, we can use the tensor product property of the basis to factor $\bar{L}^o$ into the product

$$(5.7) \qquad \bar{L}^{o} = (C_x + k A_x)(C_y + k A_y)$$

where

a) $C_x = \left((c_o^{1/2} \Phi_i'(x), \Phi_i(x)\right)$

b) $A_x = \left(\beta a_1 c_o^{-1/2} (\Phi_j'(x), \Phi_i'(x))\right)$

c) $C_y = \left((c_o^{1/2} \psi_n(y), \psi_m(y)\right)$

d) $A_y = \left(\beta a_2 c_o^{-1/2} (\psi_n'(y), \psi_m'(y)\right)$

for $i, j = 1, \cdots, M_x$, and $m, n = 1, \cdots, M_y$. Thus inverting $\bar{L}^{o}$ corresponds to solving two one-dimensional problems successively.

The preconditioning process eliminates the need for factoring new matrices at each time step and reduces the problem to successive solution of one-dimensional problems, while the iterative procedure stabilizes the resulting problem. The stabilization process requires iteration only until a predetermined norm reduction is achieved.

Denote by

$$(5.8) \qquad V^s = \sum_{i=1}^{M} \theta_i^s B_i(\vec{x}) = \sum_{m=1}^{M_x} \sum_{n=1}^{M_y} \theta_{mn}^s \Phi_m(x) \psi_n(y) ,$$

the approximation to U^s produced by only approximately solving (5.5) using $\bar{L}^{o}$. Assume sufficiently accurate starting values have been obtained (see [3,4]). Assuming $V^o, \cdots, V^n$ have been determined, we shall determine the M-dimensional vector θ^{n+1} (and thus V^{n+1} from (5.8)) using a preconditioned iterative method to approximate ξ^{n+1} from (5.5). As an initial guess for $\xi^{n+1} - \xi^n$, we shall extrapolate from previously determined values. Specifically, for the method under consideration having time-truncation error $O(k^3)$, we shall use as an initialization for our iterative procedure

$$x_o = \left(\theta^{n+1} - \theta^n\right) - \delta^4\, \theta^{n+1} \,. \tag{5.9}$$

Since we are using previously determined θ^i in the matrix problem (5.5) to determine θ^{n+1}, our errors accumulate.

In order to analyze the cumulative error, we first consider the single step error. We define $\bar{\theta}^{n+1}$ to satisfy

$$L^{n+1}\left\{\bar{\theta}^{n+1} - \theta^n\right\} = F^n(\theta) \qquad \text{, for } n \geq 3 \,. \tag{5.10}$$

Thus $\bar{\theta}^{n+1}$ would be the exact solution of (5.5) if the computed values of θ^k from previous approximate solutions of (5.5) using $\bar{L}_o$ had been used for $k \leq n$. We can use any preconditioned iterative method which yields norm reductions of the form

$$\left\|\left(L^{n+1}\right)^{1/2}\left(\bar{\theta}^{n+1} - \theta^{n+1}\right)\right\|_e \tag{5.11}$$

$$\leq \rho_n \left\|\left(L^{n+1}\right)^{1/2}\left(\bar{\theta}^{n+1} - \theta^{n+1} + \delta^4\, \theta^{n+1}\right)\right\|_e$$

where $0 < \rho_n < 1$ and the subscript e denotes the Euclidean norm of the vector. A specific iterative procedure for obtaining (4.8) is the preconditioned conjugate gradient method analyzed in [1, 2, 13, 14, 19].

Then, letting

$$\bar{V}^s = \sum_{i=1}^{M} \bar{\theta}_i^s\, B_i(\vec{x}) = \sum_{m=1}^{M_x} \sum_{n=1}^{M_y} \bar{\theta}_{mn}^s\, \Phi_m(x)\, \psi_n(y) \,, \tag{5.12}$$

with $\bar{\theta}^s$ defined in (5.10), we see that V^{n+1} and $\bar{V}^{n+1}$ satisfy

$$
\begin{aligned}
&k^{-1}\left(c\left(x, E(\mu)\, V^{n+1}\right)\delta V^{n+1}, \chi\right) \\
&\quad + \beta\left(a_x\left(x, E(\mu)\, V^{n+1}\right)\frac{\partial}{\partial x} V^{n+1}, \frac{\partial}{\partial x}\chi\right) \\
&\quad + \beta\left(a_y\left(x, E(\mu)\, V^{n+1}\right)\frac{\partial}{\partial y} V^{n+1}, \frac{\partial}{\partial y}\chi\right) \\
&\quad + \frac{k\beta^2 a_1 a_2}{c_o}\left(\frac{\partial^2}{\partial x \partial y} D(\mu)\, V^{n+1}, \frac{\partial^2}{\partial x \partial y}\chi\right) \\
&\quad = k^{-1}\left(c\left(x, E(\mu)\, V^{n+1}\right)\left[\alpha_1 \delta V^{n} + \alpha_2 \delta V^{n-1}\right], \chi\right) \\
&\quad + \beta\left(f\left(x, t^{n+1}, E(\mu)\, V^{n+1}\right), \chi\right) \\
&\quad + k^{-1}\left(c\left(x, E(\mu)\, V^{n+1}\right)\left[V^{n+1} - \bar{V}^{n+1}\right], \chi\right) \\
&\quad + \beta\left(a_x\left(x, E(\mu)\, V^{n+1}\right)\frac{\partial}{\partial x}\left(V^{n+1} - \bar{V}^{n+1}\right), \frac{\partial}{\partial x}\chi\right) \\
&\quad + \beta\left(a_y\left(x, E(\mu)\, V^{n+1}\right)\frac{\partial}{\partial y}\left(V^{n+1} - \bar{V}^{n+1}\right), \frac{\partial}{\partial y}\chi\right) \\
&\quad + \frac{k\beta^2 a_1 a_2}{c_o}\left(\frac{\partial^2}{\partial x \partial y}\left[V^{n+1} - \bar{V}^{n+1}\right], \frac{\partial^2}{\partial x \partial y}\chi\right) \qquad , \chi \in M_h ,
\end{aligned}
\tag{5.13}
$$

where the last four terms measure the single step error arising from the iterative stabilization. We must iterate only sufficiently often to control these terms in the analysis. Since $\bar{L}_o$ is a sequence of one-dimensional operators, we can very efficiently update $\bar{L}_o$ if L_n drifts far away from $\bar{L}_o$. Analysis and details will appear in [3].

Note that in preconditioned iterative methods, only the preconditioner is inverted. In this case, that is only a sequence of one-dimensional problems. If the basis functions in the one-dimensional problem are linear (tensor products of linears for the basis for M_h) the matrices to be inverted are tridiagonal and if the basis functions are quadratic the matrices are pentadiagonal. Thus if $d = 2$ or 3 the work estimate is $O(M_x M_y)$ or $O(M_x M_y M_z)$,

respectively. Thus the work is proportional to the total number of unknowns in the problem and optimal order work estimates are obtained (see [11, 14, 24, 27, 28]).

The storage requirements are also very attractive for AD methods. Since the matrix problem is treated as a series of one-dimensional problems, only the data corresponding to one grid line are required in core at any given time. In two dimensions the storage requirements for these AD methods are comparable to those of a frontal elimination solver, but these methods require considerably less I/O. In three dimensions the frontal elimination solvers require that a plane of data be in core, while these methods only require one line of data. Clearly all of the above remarks apply to each of the AD methods presented here, not only to the iterative variants.

The author has applied iterative stabilization methods to problems of hyperbolic and Sobolev types in [19, 20]. The extension of these iterative ideas to AD methods for equations of these types follows from the ideas presented above for parabolic problems.

TABLE 1: BACKWARD DIFFERENTIATION MULTISTEP METHODS

μ	$\beta(\mu)$	$\alpha_1(\mu)$	$\alpha_2(\mu)$	$\alpha_3(\mu)$	$E(\mu)\ U^{n+1}$
1	1	0	0	0	$U^{n+1} - \delta\ U^{n+1}$
2	2/3	1/3	0	0	$U^{n+1} - \delta^2\ U^{n+1}$
3	6/11	7/11	-2/11	0	$U^{n+1} - \delta^3\ U^{n+1}$
4	12/25	23/25	-13/25	3/25	$U^{n+1} - \delta^4\ U^{n+1}$

TABLE 2: DIRECT METHODS

μ	$D(\mu)\ U^{n+1}$	$F(\mu)\ U^{n+1}$	$G(\mu)U^{n+1}$	Coefficient Constraints
1	δU^{n+1}	$\delta^2 U^{n+1}$	$\delta\ U^{n+1}$	one-sided (c_o)
2	δU^{n+1}	$\delta^3 U^{n+1}$	$\delta^2 U^{n+1}$	two-sided (c_o)
2	δU^{n+1}	$\delta^2 U^{n+1} - kc_o S_\gamma^{-1}\delta[c_n^{-1} L_n U^n]$	$\delta^2 U^{n+1}$	one-sided (c_o)
3	$\delta^2 U^{n+1}$	—	$\delta^3 U^{n+1}$	two-sided (a_o)
3	$\delta U^{n+1} - C_o S_\gamma^{-1} U^n$	—	$\delta^2 U^{n+1} - c_o S_\gamma^{-1}\delta^2 U^n$	one-sided (a_o)

REFERENCES

1. O. Axelsson, "On Preconditioning and Convergence Acceleration in Sparse Matrix Problems", CERN European Organization for Nuclear Research, Geneva, 1974.

2. O. Axelsson, "On the Computational Complexity of Some Matrix Iterative Algorithms", Report 74.06, Dept. of Computer Science, Chalmers University of Technology, Goteberg, 1974.

3. J. H. Bramble and R. E. Ewing, "Alternating Direction Multistep Methods for Parabolic Problems - Iterative Stabilization" (in preparation).

4. J. H. Bramble and R. E. Ewing, "Efficient Starting Procedures for High-Order Time-Stepping Methods for Differential Equations", (in preparation).

5. J. H. Bramble and P. H. Sammon, "Efficient Higher-Order Multistep Methods for Parabolic Problems" (in preparation).

6. B. D. Coleman and W. Noll, "The Thermodynamics of Elastic Materials with Heat conduction and Viscosity", Arch. Rat. Mech. Anal. 13 (1963), pp. 167-178.

7. C. M. Dafermos, "The Mixed Initial-Boundary Problem for Equations of Non-linear Viscoelasticity", J. of Diff. Eqn. 6 (1969), pp. 71-81.

8. J. E. Dendy, Jr., "An Analysis of some Galerkin Schemes for the Solution of Nonlinear Time-Dependent Problems", SIAM J. Numer. Anal. 12 (1975), pp. 541-565.

9. J. E. Dendy, Jr., and G. Fairweather, "Alternating-Direction Galerkin Methods for Parabolic and Hyperbolic Problems on Rectangular Polygons", SIAM J. Numer. Anal. 12 (1975), pp. 144-163.

10. J. Douglas, Jr., "Alternating-Direction Methods for Three Space Variables", Numerische Mathematik 4 (1962), pp. 41-63.

11. J. Douglas, Jr., "Effective Time-Stepping Methods for the Numerical Solution of Nonlinear Parabolic Problems", to appear in the Proceedings of MAFELAP78, Brunel University, 1978.

12. J. Douglas and T. Dupont, "Alternating-Direction Galerkin Methods on Rectangles", in Proceedings Symposium on Numerical Solution of Partial Differential Equations, II, B. Hubbard (ed.), Academic Press. New York, 1971, pp. 133-214.

13. J. Douglas, Jr., and T. Dupont, "Preconditioned Conjugate Gradient Iteration Applied to Galerkin Methods for a Mildly Nonlinear Dirichlet Problem", Sparse Matrix Calculations, J. R. Bunch and D. J. Rose, eds., Academic Press, New York, 1976, pp. 333-348.

14. J. Douglas, Jr., T. Dupont, and R. E. Ewing, "Incomplete Iteration for Time-Stepping a Galerkin Method for a Quasilinear Parabolic Problem", SIAM J. Numer. Anal. 16 (1979), pp. 503-522.

15. J. Douglas, T. Dupont, and P. Percell, "A Time-Stepping Method for Galerkin Approximations for Nonlinear Parabolic Equations", to appear in Proceedings of the Conference on Numerical Analysis held at Dundee, Scotland, June 28-July 1, 1977.

16. J. Douglas, Jr., T. Dupont, and L. Wahlbin, "Optimal L_∞ Error Estimates for Galerkin Approximations to Solutions of Two-Point Boundary Value Problems", Math. Comp. 29 (1975), pp. 475-483.

17. J. Douglas, Jr., and J. E. Gunn, "A General Formulation of Alternating-Direction Methods", Numer. Math. 6 (1964), pp. 428-453.

18. J. Douglas, Jr., and H. H. Rachford, "On the Numerical Solution of Heat Conduction Problems in Two and Three Space Variables", Trans. Am. Math. Soc. 82 (1956), pp. 421-439.

19. R. E. Ewing, "Time-Stepping Galerkin Methods for Nonlinear Sobolev Partial Differential Equations", SIAM J. Numer. Anal. 15 (1978), pp. 1125-1150.

20. R. E. Ewing, "On Efficient Time-Stepping Methods for Nonlinear Partial Differential Equations, Math. Res. Cent. Rep. #1996, University of Wisconsin, Madison (1979), and Computers and Math. with Appl. 6 (1980), pp. 1-13.

21. R. E. Ewing, "Efficient Multistep Procedures for Nonlinear Parabolic Problems with Nonlinear Neumann Boundary Conditions, Math. Res. Cent. Rep. #1982, University of Wisconsin, Madison (1979), and Trans. 25th Conf. Army Math. ARO Report 80-1, and Calcolo (to appear).

22. R. E. Ewing and L. J. Hayes, "Alternating Direction Galerkin Methods for Nonlinear Sobolev Partial Differential Equations", (in preparation).

23. R. E. Ewing and L. J. Hayes, "Alternating Direction Galerkin Methods for Some Third and Fourth Order Partial Differential Equations", (in preparation).

24. R. E. Ewing and T. F. Russell, "Efficient Time-Stepping Methods for Miscible Displacement Problems in Porous Media", SIAM J. Numer. Anal. (to appear).

25. J. M. Greenberg, R. C. MacCamy, and V. J. Mizel, "On the Existence, Uniqueness, and Stability of Solutions of the Equation $\sigma'(u_x)u_{xx} + \lambda u_{xtx} = \rho_0 u_{tt}$," J. Math. and Mech. 17 (1968), pp. 707-728.

26. L. J. Hayes, "Generalization of Galerkin Alternating-Direction Methods to Nonrectangular Regions Using Isoparametric Elements", Ph.D. Thesis, University of Texas at Austin, December 1977.

27. L. J. Hayes, "Generalization of Galerkin Alternating-Direction Methods to Nonrectangular Regions Using Patch Approximations", SIAM J. Numer. Anal. (to appear).

28. L. J. Hayes and P. Percell, "Efficient Second-Order Time-Stepping Procedures for Nonlinear Parabolic Equations", SIAM J. Numer. Anal. (to appear).

29. M. Lighthill, "Dynamics of Rotating Fluids: a Survey", J. Fluid Mech. 26 (1966), pp. 411-436.

30. A. E. H. Love, A Treatise on the Mathematical Theory of Elasticity, Dover, New York, 1944.

31. R. C. MacCamy, "Existence, Uniqueness, and Stability of Solutions of the Equation $u_{tt} = \frac{\partial}{\partial x}\left(\sigma(u_x) + \lambda(u_x)u_{xt}\right)$", Report 68-18, Dept. of Math., Carnegie Inst. of Technology, Carnegie-Mellon Univeristy.

32. D. W. Peaceman and H. H. Rachford, "The Numerical Solution of Parabolic and Elliptic Differential Equations", SIAM J. 3 (1955), pp. 28-41.

33. G. W. Platzman, "The Eigenvalues of Laplace's Tidal Equations", Quart. J. of Royal Meteorological Soc. 94 (1968), pp. 225-248.

34. R. E. Showalter, "Regularization and Approximation of Second-Order Evolution Equations", SIAM J. Math. Anal. 7 (1976), pp. 461-472.