JOURNAL OF COMPUTATIONAL PHYSICS 112, 211–225 (1994)

The Galerkin–Collocation Method for Hyperbolic Initial Boundary Value Problems

P. DUTT* AND A. K. SINGH

Department of Mathematics, Indian Institute of Technology at Kanpur, 208016, India

Received June 17, 1991; revised November 23, 1992

In this paper we implement a spectral method for solving initial boundary value problems which is in between the Galerkin and collocation methods. In this method the partial differential equation and initial and boundary conditions are collocated at an overdetermined set of points and the approximate solution is chosen to be the least-squares solution to this system of equations. The solution is obtained using preconditioned residual minimization. Numerical results for linear and nonlinear hyperbolic problems are provided.

1. INTRODUCTION

Current formulations of spectral methods for solving initial boundary value problems employ a spectral discretization only in space and rely on finite-difference techniques for advancing in time. As a result the global accuracy of the method is reduced to only finite order unless very small time steps are used, which is not always practicable. Further, a proper theoretical underpinning to the method is lacking at present.

In the early 1980s Morchoisne [5, 6] had proposed a method for solving such systems of equations which was spectral in both space and time. However, even though his numerical results were impressive, the method has not acquired general acceptance so far. One reason for this was that it required considerably more memory than conventional spectral methods. However, with the increasing availability and use of parallel computing systems this is no longer a serious constraint, a point which was made by Morchoisne in one of his papers [6]. Another possible reason for the neglect of this approach was that it lacked a theoretical justification.

Recently, Dutt [2] proposed a method for solving initial boundary value problems which is similar to Morchoisne's approach in that it employs a spectral discretization in both space and time; henceforth we shall refer to it as the Galerkin–collocation method. The method is set in a Galerkin formulation as it seeks an approximate solution which minimizes a weighted sum of the residuals in a filtered version of the partial differential equations and initial and boundary conditions.

The solution process, however, effectively amounts to collocating the filtered version of the partial differential equation and initial and boundary conditions at an overdetermined set of points. We show in this paper that *the filtering can be dispensed with*, and it suffices to collocate the partial differential equation and initial and boundary conditions at the overdetermined set of points. The solution is then obtained by finding a least-squares solution to the overdetermined set of equations. It has been proved that the solution thus obtained converges to the actual solution at a spectral rate of accuracy in both space and time. In practice, the huge, full, and overdetermined set of equations is solved by iterative techniques in which a low order finite difference solver is used as a preconditioner, as was first proposed by Orszag [7] and Morchoisne [5, 6]. Zang, Wong, and Hussaini [10] have experimented with multigrid versions of this approach in order to accelerate convergence of the solution procedure.

We now outline the contents of this paper. In Section II we present a brief discussion of the proposed method and theoretical results pertaining to it. In Section III we describe the preconditioning technique we adopt for the scalar problem and report the computational results we have obtained. In Section IV we investigate preconditioning techniques for the system case and highlight in particular one technique which uses an approximate treatment of the boundary conditions to effectively decouple the system of equations. This, combined with a diagonal implicit factorization technique [8, 9], causes a considerable reduction in computational effort and may have important applications to solving such systems on parallel computers using spectral techniques along with domain decomposition. Finally, in Section V we present computational results for nonlinear problems.

* Research partially supported by C.D.A.C. (Center for Development of Advanced Computing), Pune.

0021-9991/94 $6.00

II. DISCUSSION OF METHOD AND THEORETICAL RESULTS

In this paper we restrict ourselves to the case of one space dimension. The method we describe, however, is applicable to any number of space dimensions.

We shall shift our initial time from $t=0$ to $t=-1$ as this will considerably simplify our presentation. Consider the differential operator

$$Lu \equiv u_t - Au_x - Bu. \tag{1}$$

Here u is a vector-valued function with k components and A and B are $k \times k$ matrix-valued functions which are smooth functions of x and t. We assume the system (1) is strictly hyperbolic. We consider the initial boundary value problem,

$$Lu(x, t) = F(x, t) \qquad \text{for} \quad -1 \leqslant x \leqslant 1, \quad -1 \leqslant t \leqslant 1, \tag{2.a}$$

with boundary conditions,

$$\mathbf{M}u(-1, t) = g(t) \qquad \text{for} \quad -1 \leqslant t \leqslant 1, \tag{2.b}$$

$$\mathbf{P}u(1, t) = h(t) \qquad \text{for} \quad -1 \leqslant t \leqslant 1, \tag{2.c}$$

and initial conditions,

$$u(x, -1) = f(x) \qquad \text{for} \quad -1 \leqslant x \leqslant 1. \tag{2.d}$$

If there are l inflow variables at the boundary $x=-1$ then $\mathbf{M}$ is an $l \times k$ matrix-valued function which prescribes the l inflow variables in terms of the $(k-l)$ outflow variables. Similarly if there are s inflow variables at the boundary $x=1$ then $\mathbf{P}$ is an $s \times k$ matrix-valued function. Both $\mathbf{M}$ and $\mathbf{P}$ are smooth functions of t. We assume that the initial and boundary data f, g, h and forcing function F are smooth and satisfy the compatibility conditions which must hold at the space-time corners for the solution u to be smooth. Finally we assume that the above initial boundary value problem (IBVP) satisfies the uniform Kreiss condition. If the uniform Kreiss condition is satisfied then the IBVP is well posed; i.e., the solution u depends continuously on its data. More precisely, it has been proved that the estimate

$$\int_{-1}^{1} \int_{-1}^{1} \|u(x, t)\|^2 \, dx \, dt + \int_{-1}^{1} \|u(-1, t)\|^2 \, dt + \int_{-1}^{1} \|u(1, t)\|^2 \, dt + \int_{-1}^{1} \|u(x, 1)\|^2 \, dx$$
$$\leqslant C\left[\int_{-1}^{1} \int_{-1}^{1} \|F(x, t)\|^2 \, dx \, dt + \int_{-1}^{1} \|f(x)\|^2 \, dx + \int_{-1}^{1} \|g(t)\|^2 \, dt + \int_{-1}^{1} \|h(t)\|^2 \, dt\right] \tag{3}$$

holds for some positive constant C. Here the norm $\| \; \|$ denotes the Euclidean norm.

One final remark we make is that an IBVP that is well posed is "structurally stable"; i.e., if we perturb the coefficients of the differential operator and boundary operator by a small amount then the perturbed problem continues to remain well posed. This property is crucial for proving that the approximate solution we obtain by our method converges to the actual solution of the IBVP.

The method which we now describe applies to general Geggenbauer polynomials but here we shall describe it for Chebyshev polynomials. We recall that the Chebyshev polynomials $T_j(y) = \cos(j \cos^{-1}(y))$ are orthogonal with respect to the weight function

$$\omega(y) = \frac{1}{\sqrt{1-y^2}}$$

in the interval $[-1, 1]$.

Let $S^{p,q}$ be the set of polynomials $w^{p,q}(x, t)$ of the form

$$w^{p,q}(x, t) = \sum_{i=0}^{p} \sum_{j=0}^{q} a_{ij} T_i(x) \, T_j(t), \tag{4}$$

with scalar coefficients a_{ij}. Similarly, we shall denote by $(S^{p,q})^k$ the set of polynomials $w^{p,q}$ of the form (4) if the coefficients a_{ij} are vectors with k components. Henceforth we shall assume that there exists a constant λ such that $1/\lambda \leqslant p/q \leqslant \lambda$.

We now define an interpolation operator $I^{p,q}$ which takes a continuous function $r(x, t)$ defined on $[-1, 1] \times [-1, 1]$ and projects it into $S^{p,q}$. Thus

$$I^{p,q} r(x, t) = \sum_{j=0}^{q} \sum_{i=0}^{q} b_{ij} T_i(x) \, T_j(t) = \bar{r}^{p,q}(x, t) \tag{5}$$

is the unique polynomial belonging to $S^{p,q}$ which interpolates $r(x, t)$ at the $(p+1) \times (q+1)$ points $\{(x_i^p, t_j^q)\}_{i=0,\dots,p,\, j=0,\dots,q}$. Here the points

$$x_i^p = \cos(i\pi/p), \qquad 0 \leqslant i \leqslant p,$$
$$t_j^q = \cos(j\pi/q), \qquad 0 \leqslant j \leqslant q,$$

are the Gauss–Lobatto–Chebyshev points.

In much the same way we can define a one-dimensional interpolation operator I^l which takes a continuous function $s(y)$ defined on $[-1, 1]$ and projects it into the space of polynomials of degree $\leqslant l$. Thus

$$I^l s(y) = \sum_{i=0}^{l} b_i T_i(y) = \bar{s}^l(y) \tag{6}$$

is the unique polynomial of degree $\leqslant l$ which interpolates $s(y)$ at the $(l+1)$ points $\{y_i^l = \cos(i\pi/l)\}_{i=0,...,l}$.

We can now use these interpolation operators to define a filtered version of the differential operator

$$Lu = u_t - Au_x - Bu.$$

Let

$$\bar{A}^{p,q} = I^{p,q} A,$$

$$\bar{B}^{p,q} = I^{p,q} B$$

be the polynomial interpolants of the $k \times k$ matrix-valued functions A and B. We now define the differential operator

$$L^{p,q} u = u_t - \bar{A}^{p,q} u_x - \bar{B}^{p,q} u, \tag{7}$$

which can be regarded as a perturbed version of the original differential operator Lu.

Similarly, we define

$$\mathbf{M}^q = I^q \mathbf{M},$$

$$\mathbf{P}^q = I^q \mathbf{P}.$$

We now replace the original IBVP by a filtered version,

$$L^{p,q} \tilde{u}(x, t) \quad \text{for} \quad -1 \leqslant x \leqslant 1, \quad -1 \leqslant t \leqslant 1, \tag{8.a}$$

with boundary conditions,

$$\mathbf{M}^q \tilde{u}(-1, 1) = g(t) \quad \text{for} \quad -1 \leqslant t \leqslant 1, \tag{8.b}$$

$$\mathbf{P}^q \tilde{u}(1, t) = h(t) \quad \text{for} \quad -1 \leqslant t \leqslant 1, \tag{8.c}$$

and initial conditions,

$$\tilde{u}(x, -1) = f(x) \quad \text{for} \quad -1 \leqslant x \leqslant 1. \tag{8.d}$$

The above IBVP will be well posed if we choose p and q large enough. In fact since (8) can be regarded as a perturbation of (2) the energy estimate [2]

$$\begin{aligned}
&\int_{-1}^{1} \int_{-1}^{1} \|u^{p,q}(x, t)\|^2 \, dx \, dt \\
&\quad + \int_{-1}^{1} \|u^{p,q}(-1, t)\|^2 \, dt \\
&\quad + \int_{-1}^{1} \|u^{p,q}(1, t)\|^2 \, dt \\
&\quad + \int_{-1}^{1} \|u^{p,q}(x, +1)\|^2 \, dx \\
&\leqslant C \left[\int_{-1}^{1} \int_{-1}^{1} \|L^{p,q} u^{p,q}(x, t)\|^2 \, dx \, dt \right. \\
&\quad + \int_{-1}^{1} \|u^{p,q}(x, -1)\|^2 \, dx \\
&\quad + \int_{-1}^{1} \|\mathbf{M}^q u^{p,q}(-1, t)\|^2 \, dt \\
&\quad \left. + \int_{-1}^{1} \|\mathbf{P}^q u^{p,q}(1, t)\|^2 \, dt \right],
\end{aligned} \tag{9}$$

holds for p and q large enough, with some constant C. Henceforth we shall let C denote a generic constant.

From the above the inequality

$$\begin{aligned}
&\int_{-1}^{1} \int_{-1}^{1} \|u^{p,q}(x, t)\|^2 \, dx \, dt \\
&\leqslant C \left[\int_{-1}^{1} \int_{-1}^{1} \|L^{p,q} u^{p,q}(x, t)\|^2 \, \omega(x) \, \omega(t) \, dx \, dt \right. \\
&\quad + \int_{-1}^{1} \|u^{p,q}(x, -1)\|^2 \, \omega(x) \, dx \\
&\quad + \int_{-1}^{1} \|\mathbf{M}^q u^{p,q}(-1, t)\|^2 \, \omega(t) \, dt \\
&\quad \left. + \int_{-1}^{1} \|\mathbf{P}^q u^{p,q}(1, t)\|^2 \, \omega(t) \, dt \right],
\end{aligned} \tag{10}$$

follows immediately since the weight function $\omega \geqslant 1$.

We wish to find an approximate solution $u^{p,q}(x, t) \in (S^{p,q})^k$ to the above IBVP. Note that if $u^{p,q}(x, t) \in (S^{p,q})^k$ then

$$L^{p,q} u^{p,q}(x, t) \in (S^{2p,2q})^k,$$

$$\mathbf{M}^q u^{p,q}(-1, t) \in (S^{2q})^l,$$

$$\mathbf{P}^q u^{p,q}(1, t) \in (S^{2q})^s,$$

$$u^{p,q}(x, -1) \in (S^p)^k,$$

and this suggests that we should accordingly filter our data.

Let

$$\bar{F}^{2p,2q}(x,t)=I^{2p,2q}F(x,t),$$
$$\bar{g}^{2q}(x,t)=I^{2q}g(t),$$
$$\bar{h}^{2q}(t)=I^{2q}h(t),$$
$$\bar{f}^{2p}(x)=I^{2p}f$$

be filtered representations of the data. If we were to substitute our approximate solution into the IBVP the residuals

$$\begin{aligned}\rho^{p,q}(x,t)&=L^{p,q}u^{p,q}(x,t)-\bar{F}^{2p,2q}(x,t),\\ \sigma^{q}(t)&=\mathbf{M}^{q}u^{p,q}(-1,t)-\bar{g}^{2q}(t),\\ \eta^{q}(t)&=\mathbf{P}^{q}u^{p,q}(1,t)-\bar{h}^{2q}(t),\\ \tau^{p}(x)&=u^{p,q}(x,-1)-\bar{f}^{2p}(x)\end{aligned}\tag{11}$$

would, in general, not be zero. We would like to choose our approximate solution $u^{p,q}(x,t)$ so that it makes these residuals as small as possible and for this we need to define a functional which will measure the size of the residuals.

Accordingly we define a functional

$$\begin{aligned}\mathbf{H}^{p,q}(v^{p,q})=&\int_{-1}^{1}\int_{-1}^{1}\|L^{p,q}v^{p,q}(x,t)-\bar{F}^{2p,2q}(x,t)\|^2\\ &\times\omega(x)\,\omega(t)\,dx\,dt\\ &+\int_{-1}^{1}\|\mathbf{M}^{q}u^{p,q}(-1,t)-\bar{g}^{2q}(t)\|^2\,\omega(t)\,dt\\ &+\int_{-1}^{1}\|\mathbf{P}^{q}u^{p,q}(1,t)-\bar{h}^{2q}(t)\|^2\,\omega(t)\,dt\\ &+\int_{-1}^{1}\|v^{p,q}(x,-1)-\bar{f}^{2p}(x)\|^2\,\omega(x)\,dx,\end{aligned}\tag{12}$$

where

$$v^{p,q}(x,t)=\sum_{i=0}^{p}\sum_{j=0}^{q}b_{ij}T_i(x)\,T_j(t)\in(S^{p,q})^k.$$

We choose as our approximate solution the unique $u^{p,q}\in(S^{p,q})^k$ which minimizes a functional $H^{p,q}(v^{p,q})$ over all $v^{p,q}$, where $H^{p,q}(v^{p,q})$ is essentially equivalent to $\mathbf{H}^{p,q}(v^{p,q})$.

Now we observe that

$$\begin{aligned}\rho^{p,q}(x,t)&=L^{p,q}v^{p,q}(x,t)-\bar{F}^{2p,2q}(x,t)\in(S^{2p,2q})^k,\\ \sigma^{q}(t)&=\mathbf{M}^{q}v^{p,q}(-1,t)-\bar{g}^{2q}(t)\in(S^{2q})^l\\ \eta^{q}(t)&=\mathbf{P}^{q}v^{p,q}(1,t)-\bar{h}^{2q}(t)\in(S^{2q})^s,\\ \tau^{p}(x)&=v^{p,q}(x,-1)-\bar{f}^{2p}(x)\in(S^{2p})^k;\end{aligned}$$

so we can exactly evaluate the integrals in (12) by using the very highly accurate Gauss quadrature rules. In particular, for the Gauss–Lobatto–Chebyshev rule we have that if $s(y)$ is a polynomial of degree $\leqslant 2N-1$ then

$$\int_{-1}^{1}s(y)\,\omega(y)\,dy=\frac{\pi}{N}\sum_{j=0}^{N}\frac{s(y_j^N)}{c_j^N},\tag{13}$$

where the points y_j^N are given by

$$y_j^N=\cos(\pi j/N),\qquad 0\leqslant j\leqslant N,$$

and the weights c_j^N are given by

$$c_j^N=\begin{cases}2 & \text{if } j\neq 0 \text{ or } N,\\ 1 & \text{otherwise.}\end{cases}$$

However, there is a stronger version of this rule which we use for our particular case. Suppose $r(y)$ is a polynomial of degree $\leqslant N$. Then the inequality [1, p. 286]

$$\begin{aligned}\int_{-1}^{1}r^2(y)\,\omega(y)\,dy&\leqslant\frac{\pi}{N}\sum_{j=0}^{N}\frac{r^2(y_j^N)}{c_j^N}\\ &\leqslant 2\int_{-1}^{1}r^2(y)\,\omega(y)\,dy\end{aligned}\tag{14}$$

holds.

We can therefore replace the functional $\mathbf{H}^{p,q}(v^{p,q})$ that we are trying to minimize by an equivalent functional,

$$\begin{aligned}H^{p,q}(v^{p,q})=&\frac{\pi^2}{4pq}\sum_{j=0}^{2q}\sum_{i=0}^{2p}\\ &\times\frac{\|L^{p,q}v^{p,q}(x_i^{2p},t_j^{2q})-\bar{F}^{2p,2q}(x_i^{2p},t_j^{2q})\|^2}{c_i^{2p}\times c_j^{2q}}\\ &+\frac{\pi}{2q}\sum_{j=0}^{2q}\frac{\|\mathbf{M}^{q}v^{p,q}(-1,t_j^{2q})-\bar{g}^{2q}(t_j^{2q})\|^2}{c_j^{2q}}\\ &+\frac{\pi}{2q}\sum_{j=0}^{2q}\frac{\|\mathbf{P}^{q}v^{p,q}(1,t_j^{2q})-\bar{h}^{2q}(t_j^{2q})\|^2}{c_j^{2q}}\\ &+\frac{\pi}{2p}\sum_{i=0}^{2p}\frac{\|v^{p,q}(x_i^{2p},-1)-\bar{f}^{2p}(x_i^{2p})\|^2}{c_i^{2p}}.\end{aligned}\tag{15}$$

In fact, using (14) we conclude that

$$\mathbf{H}^{p,q}(v^{p,q})\leqslant H^{p,q}(v^{p,q})\leqslant 4\mathbf{H}^{p,q}(v^{p,q}).$$

We choose as our approximate solution $u^{p,q}\in(S^{p,q})^k$ which minimizes $H^{p,q}$.

In other words, our solution $u^{p,q}$ is given by a least-squares solution to the overdetermined system of equations,

$$\begin{aligned}
&\left(\frac{\pi^2}{4pqc_i^{2p}c_j^{2q}}\right)^{1/2} \{L^{p,q}u^{p,q}-F)(x_i^{2p}, t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant i \leqslant 2p, \quad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2qc_j^{2q}}\right)^{1/2} \{\mathbf{M}^q(t_j^{2q})\, u^{p,q}(-1, t_j^{2q}) - g(t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2qc_j^{2q}}\right)^{1/2} \{\mathbf{P}^q(t_j^{2q})\, u^{p,q}(-1, t_j^{2q}) - h(t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2pc_i^{2p}}\right)^{1/2} \{u^{p,q}(x_i^{2p}, -1) - f)(x_i^{2p})\} \\
&\quad =0, \qquad 0 \leqslant i \leqslant 2p.
\end{aligned} \tag{16}$$

Here, we have used the fact that $\bar{F}^{2p,2q}(x_i^{2p}, t_j^{2q}) = F(x_i^{2p}, t_j^{2q})$ etc., so we can work with point values of the *original data*. We may write the system (16) in the form

$$D^{p,q}U^{p,q} = Z^{p,q}, \tag{17}$$

where $D^{p,q}$ is a $\lambda \times v$ matrix, $U^{p,q}$ is a v column vector formed by concatenating the point values $\{u^{p,q}(x_i^p t_j^q)\}_{i=0,\ldots,p;\, j=0,\ldots,q}$ and $Z^{p,q}$ is a λ-column vector with

$$\begin{aligned}
\lambda &= k(2p+1)(2q+1) + (l+s)(2q+1) + k(2p+1), \\
v &= k(p+1)(q+1).
\end{aligned}$$

We emphasize that $U^{p,q}$ denotes the v column vector defined above and $u^{p,q}(x, t)$ is the polynomial belonging to $(S^{p,q})^k$ whose point values are the components of $U^{p,q}$. We wish to find a least-squares solution to the problem (17). Clearly, $U^{p,q}$ must satisfy the linear system of equations

$$[(D^{p,q})^{\mathrm{T}}\, D^{p,q}]\, U^{p,q} = (D^{p,q})^{\mathrm{T}}\, Z^{p,q}.$$

In [2] it has been shown that the matrix $(D^{p,q})^{\mathrm{T}}\, (D^{p,q})$ has an inverse for p and q large enough. Hence the solution to the minimization problem is unique.

To store the filtered representations of the coefficient matrices $A^{p,q}$, $B^{p,q}$ etc. would place a prohibitive overhead on memory requirements for realistic problems; there is a way of getting around this, however. Instead of solving the system (17) we choose $\tilde{U}^{p,q}$ which is the least-squares solution to the unfiltered system of equations

$$\begin{aligned}
&\left(\frac{\pi^2}{4(pqc_i^{2p}c_j^{2q})}\right)^{1/2} \{L\tilde{u}^{p,q}(x_i^{2p}, t_j^{2q}) - F(x_i^{2p}, t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant i \leqslant 2p, \quad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2qc_j^{2q}}\right)^{1/2} \{\mathbf{M}(t_j^{2q})\, \tilde{u}^{p,q}(-1, t_j^{2q}) - g(t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2qc_j^{2q}}\right)^{1/2} \{\mathbf{P}(t_j^{2q})\, \tilde{u}^{p,q}(1, t_j^{2q}) - h(t_j^{2q})\} \\
&\quad =0, \qquad 0 \leqslant j \leqslant 2q, \\
&\left(\frac{\pi}{2pc_j^{2q}}\right)^{1/2} \{\tilde{u}^{p,q}(x_i^{2p}, -1) - f(x_i^{2p})\} \\
&\quad =0, \qquad 0 \leqslant i \leqslant 2p,
\end{aligned} \tag{18}$$

as our approximate solution.

Note that the system (19) may be written as

$$\tilde{D}^{p,q}\tilde{U}^{p,q} = Z^{p,q} \tag{19}$$

which is the same as (17), except that the matrix $D^{p,q}$ has been replaced by $\tilde{D}^{p,q}$, where $\tilde{D}^{p,q}$ may be regarded as a perturbed version of $D^{p,q}$. If $A(x, t)$ is a smooth function then we know that

$$|A(x_i^{2p}, t_j^{2q}) - \bar{A}^{p,q}(x_i^{2p}, t_j^{2q})|$$

is spectrally small for all i and j. Using this we conclude that the matrix $\tilde{D}^{p,q}$ differs from $D^{p,q}$ by a spectrally small amount and hence the difference between $U^{p,q}$ and $\tilde{U}^{p,q}$, the least-squares solutions of (17) and (19), respectively, is spectrally small.

It is the system of Eqs. (18), (19) *that we will be solving in the remainder of this paper.* We make the arguments we have outlined above rigorous in the following lemmas and theorem.

LEMMA 1. *Let $v^{p,q}$ belong to the space of polynomials $S^{p,q}$ defined in* (4). *Then*

$$\begin{aligned}
&\left(\int_{-1}^{1}\int_{-1}^{1} (v^{p,q})^2\, \omega(x)\, \omega(t)\, dx\, dt\right) \\
&\qquad \leqslant C_1 \left(\int_{-1}^{1}\int_{-1}^{1} (v^{p,q})^2\, dx\, dt\right),
\end{aligned} \tag{20}$$

where $C_1 = E_\alpha(pq)^{2-2/\alpha}$, for any $\alpha > 2$. In particular, choosing $\alpha = 4$ we obtain $C_1 = E_4(pq)^{3/2}$.

Using Holder's inequality we obtain

$$\left(\int_{-1}^{1}\int_{-1}^{1}(v^{p,q})^2\,\omega(x)\,\omega(t)\,dx\,dt\right)$$
$$\leqslant\left(\int_{-1}^{1}\int_{-1}^{1}((v^{p,q})^2)^\alpha\,dx\,dt\right)^{1/\alpha}$$
$$\times\left(\int_{-1}^{1}\int_{-1}^{1}(\omega(x)\,\omega(t))^\beta\,dx\,dt\right)^{1/\beta},$$

where $1/\alpha+1/\beta=1$. Now

$$\int_{-1}^{1}(\omega(x))^\beta\,dx=\int_{-1}^{1}\frac{1}{(1-x^2)^{\beta/2}}\,dx$$

is finite for $1<\beta<2$.

For a fixed value of β the right-hand side of the last equation becomes a constant which we shall denote by D_β. Hence we can conclude that

$$\left(\int_{-1}^{1}\int_{-1}^{1}(\omega(x)\,\omega(t))^\beta\,dx\,dt\right)^{1/\beta}=D_\beta^{2/\beta}.$$

Now if $s(y)$ is a polynomial of degree m then by Nikolskii's inequality [1, p. 288]

$$\left(\int_{-1}^{1}(s(y))^\alpha\,dy\right)^{1/\alpha}\leqslant Km^{2(1/\beta-1/\alpha)}\left(\int_{-1}^{1}(s(y))^\beta\,dy\right)^{1/\beta},$$

for $1\leqslant\beta<\alpha$. Thus we obtain

$$\left(\int_{-1}^{1}\int_{-1}^{1}((v^{p,q})^2)^\alpha\,dx\,dt\right)^{1/\alpha}$$
$$\leqslant K^2(4pq)^{2(1-1/\alpha)}\left(\int_{-1}^{1}\int_{-1}^{1}((v^{p,q})^2)\,dx\,dt\right)$$

Putting $E_\alpha=K^2(4D_\beta)^{2/\beta}$ we obtain the required result. ∎

LEMMA 2. *There are constants K_1 and K_2 depending on p and q such that the estimate*

$$K_1\,\|V^{p,q}\|^2\leqslant\|D^{p,q}V^{p,q}\|^2\leqslant K_2\,\|V^{p,q}\|^2. \tag{21}$$

holds. Here

$$K_1=C/p^5,$$
$$K_2=Cp^2,$$

where C denotes a generic constant.

Let $v^{p,q}(x,t)=\sum_{j=0}^{q}\sum_{i=0}^{p}a_{i,j}T_i(x)\,T_j(t)$ be the polynomial $\in(S^{p,q})^k$ such that $v^{p,q}(x_i^p,t_j^q)=\{V^{p,q}\}_{i,j}$ for $i=0,\ldots,p$, $j=0,\ldots,q$. We have that

$$\|D^{p,q}V^{p,q}\|^2=\frac{\pi^2}{4pq}\sum_{j=0}^{2q}\sum_{i=0}^{2p}\frac{\|L^{p,q}v^{p,q}(x_i^{2p},t_j^{2q})\|^2}{c_i^{2p}c_j^{2q}}$$
$$+\frac{\pi}{2q}\sum_{j=0}^{2q}\frac{\|\mathbf{M}^q v^{p,q}(-1,t_j^{2q})\|^2}{c_j^{2q}}$$
$$+\frac{\pi}{2q}\sum_{j=0}^{2q}\frac{\|\mathbf{P}^q v^{p,q}(1,t_j^{2q})\|^2}{c_j^{2q}}$$
$$+\frac{\pi}{2p}\sum_{i=0}^{2p}\frac{\|v^{p,q}(x_i^{2p},-1)\|^2}{c_i^{2p}}.$$

Then by (14) we obtain

$$\|D^{p,q}V^{p,q}\|^2\leqslant 4\left[\int_{-1}^{1}\int_{-1}^{1}\|L^{p,q}v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt\right.$$
$$+\int_{-1}^{1}\|\mathbf{M}^q v^{p,q}(-1,t)\|^2\,\omega(t)\,dt$$
$$+\int_{-1}^{1}\|\mathbf{P}^q v^{p,q}(-1,t)\|^2\,\omega(t)\,dt$$
$$\left.+\int_{-1}^{1}\|v^{p,q}(x,-1)\|^2\,\omega(x)\,dx\right].$$

Now by the inverse inequality for differentiation [1, p. 295] if $v^{p,q}\in(S^{p,q})^k$ then

$$\int_{-1}^{1}\int_{-1}^{1}\left\|\frac{\partial}{\partial x}v^{p,q}(x,t)\right\|^2\omega(x)\,\omega(t)\,dx\,dt$$
$$\leqslant Cp^4\int_{-1}^{1}\int_{-1}^{1}\|v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt,$$

and

$$\int_{-1}^{1}\int_{-1}^{1}\left\|\frac{\partial}{\partial t}v^{p,q}(x,t)\right\|^2\omega(x)\,\omega(t)\,dx\,dt$$
$$\leqslant Cq^4\int_{-1}^{1}\int_{-1}^{1}\|v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt,$$

where by C we denote a generic constant.

Further, since $\sup_{(x,t)\in[-1,1]\times[-1,1]}\|A(x,t)\|\leqslant C$, and $\sup_{(x,t)\in[-1,1]\times[-1,1]}\|B(x,t)\|\leqslant C$, we obtain

$$\int_{-1}^{1}\int_{-1}^{1}\|L^{p,q}v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt$$
$$\leqslant C(p^4+q^4)\int_{-1}^{1}\int_{-1}^{1}\|v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt$$
$$\leqslant C\frac{(p^4+q^4)}{pq}\sum_{j=0}^{q}\sum_{i=0}^{p}\|v^{p,q}(x_i^p,t_j^q)\|^2,$$

using (14). Hence we can conclude that

$$\int_{-1}^{1}\int_{-1}^{1} \|L^{p,q}v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt \leqslant C\frac{(p^4+q^4)}{pq}\|V^{p,q}\|^2.$$

Now

$$\int_{-1}^{1} \|v^{p,q}(x,-1)\|^2\,\omega(x)\,dx \leqslant \frac{\pi}{q}\sum_{j=0}^{q}\|v^{p,q}(x^p,-1)\|^2 \leqslant \frac{Cp}{pq}\sum_{j=0}^{q}\sum_{i=0}^{p}\|v^{p,q}(x_i^p,t_j^q)\|^2.$$

Hence we can conclude that

$$\int_{-1}^{1} \|v^{p,q}(x,-1)\|^2\,\omega(x)\,dx \leqslant \frac{Cp}{pq}\|V^{p,q}\|^2,$$

and, by similar arguments, that

$$\int_{-1}^{1} \|\mathbf{M}^q v^{p,q}(x,-1)\|^2\,\omega(x)\,dx \leqslant \frac{Cq}{qp}\|V^{p,q}\|^2,$$

$$\int_{-1}^{1} \|\mathbf{P}^q v^{p,q}(x,-1)\|^2\,\omega(x)\,dx \leqslant \frac{Cq}{qp}\|V^{p,q}\|^2.$$

Combining all the above inequalities we conclude that

$$\|D^{p,q}V^{p,q}\|^2 \leqslant C\left(\frac{p^4+q^4}{pq}\right)\|V^{p,q}\|^2.$$

Now using the condition that there is a constant λ such that

$$\frac{1}{\lambda} \leqslant \frac{p}{q} \leqslant \lambda,$$

we obtain $\|D^{p,q}V^{p,q}\|^2 \leqslant Cp^2\|V^{p,q}\|^2$.

Next, we have to bound $\|D^{p,q}V^{p,q}\|^2$ from below. Using (14) we have that

$$\|D^{p,q}V^{p,q}\|^2 \geqslant \int_{-1}^{1}\int_{-1}^{1} \|L^{p,q}v^{p,q}(x,t)\|^2\,\omega(x)\,\omega(t)\,dx\,dt + \int_{-1}^{1} \|\mathbf{M}^q v^{p,q}(-1,t)\|^2\,\omega(t)\,dt + \int_{-1}^{1} \|\mathbf{P}^q v^{p,q}(1,t)\|^2\,\omega(t)\,dt + \int_{-1}^{1} \|v^{p,q}(x,-1)\|^2\,\omega(x)\,dx.$$

And so by (10) we can conclude that

$$\|D^{p,q}V^{p,q}\|^2 \geqslant C\int_{-1}^{1}\int_{-1}^{1} \|v^{p,q}(x,t)\|^2\,dx\,dt,$$

and this, together with Lemma 1 and (14), gives us

$$\|D^{p,q}V^{p,q}\|^2 \geqslant \frac{C}{p^5}\|V^{p,q}\|^2. \quad \blacksquare$$

THEOREM 1. *The difference between $U^{p,q}$ and $\tilde{U}^{p,q}$, the solutions of Eqs.* (17) *and* (19), *respectively, is spectrally small.*

Proof. We have that

$$U^{p,q} = \{(D^{p,q})^{\mathrm{T}} D^{p,q}\}^{-1}(D^{p,q})^{\mathrm{T}} Z^{p,q},$$
$$\tilde{U}^{p,q} = \{(\tilde{D}^{p,q})^{\mathrm{T}} \tilde{D}^{p,q}\}^{-1}(\tilde{D}^{p,q})^{\mathrm{T}} Z^{p,q}.$$

Hence

$$\begin{aligned}\|U^{p,q}-\tilde{U}^{p,q}\| &\leqslant \|\{(D^{p,q})^{\mathrm{T}} D^{p,q}\}^{-1}\|\,\|(D^{p,q})^{\mathrm{T}}-(\tilde{D}^{p,q})^{\mathrm{T}}\|\,\|Z^{p,q}\| \\ &\quad + \|\{(\tilde{D}^{p,q})^{\mathrm{T}}\tilde{D}^{p,q}\}^{-1}-\{(D^{p,q})^{\mathrm{T}} D^{p,q}\}^{-1}\| \\ &\quad \times \|(\tilde{D}^{p,q})^{\mathrm{T}}\|\,\|Z^{p,q}\|. \end{aligned} \tag{22}$$

Now we know that $\|\tilde{D}^{p,q}-D^{p,q}\| = O(1/p^s)$ for any $s>0$. And in Lemma 2 we have shown that

$$K_1\|V^{p,q}\|^2 \leqslant \|D^{p,q}V^{p,q}\|^2 \leqslant K_2\|V^{p,q}\|^2$$

which implies that

$$1/K_1 \geqslant \|\{(D^{p,q})^{\mathrm{T}} D^{p,q}\}^{-1}\| \geqslant 1/K_2.$$

Hence

$$\|\{(D^{p,q})^{\mathrm{T}} D^{p,q}\}^{-1}\|\,\|(D^{p,q})^{\mathrm{T}}-(\tilde{D}^{p,q})^{\mathrm{T}}\|\,\|Z^{p,q}\| \leqslant 1/K_1\,O\left(\frac{1}{p^s}\right)\|Z^{p,q}\|. \tag{23}$$

We now estimate the second term in the RHS of (22). Clearly,

$$\begin{aligned}\|(\tilde{D}^{p,q})\| &\leqslant \|(D^{p,q})\| + \|(\tilde{D}^{p,q})-(D^{p,q})\| \\ &\leqslant 2\sqrt{K_2} + O\left(\frac{1}{p^s}\right) \\ &\leqslant 2\sqrt{K_2} \qquad \text{for } p, q \text{ large enough.}\end{aligned}$$

Put

$$M=(D^{p,q})^{\mathrm{T}}\,D^{p,q},$$
$$N=\{(\tilde{D}^{p,q})^{\mathrm{T}}\,\tilde{D}^{p,q}.$$

Let

$$\Delta M=(\tilde{D}^{p,q})^{\mathrm{T}}\,\tilde{D}^{p,q}-(D^{p,q})^{\mathrm{T}}\,D^{p,q}.$$

Then $N=M+\Delta M$. It is easy to show that $\|\Delta M\|=O(1/p^s)$, for all $s>0$. Hence,

$$\|N^{-1}-M^{-1}\|\leqslant\frac{\|M^{-1}\|^2\,\|\Delta M\|}{1-\|M^{-1}\|\,\|\Delta M\|}\leqslant 2\,\|M^{-1}\|^2\,\|\Delta M\|$$

for p, q large enough. So we obtain

$$\|N^{-1}-M^{-1}\|\leqslant\frac{2}{K_1^2}\,O\left(\frac{1}{p^s}\right).$$

Thus

$$\|\{(\tilde{D}^{p,q})^{\mathrm{T}}\,\tilde{D}^{p,q}\}^{-1}-\{(D^{p,q})^{\mathrm{T}}\,D^{p,q}\}^{-1}\|\;\|(\tilde{D}^{p,q})^{\mathrm{T}}\|\;\|Z^{p,q}\|$$
$$\leqslant 4\frac{\sqrt{K_2}}{K_1^2}\,O\left(\frac{1}{p^s}\right)\|Z^{p,q}\| \qquad (24)$$

and

$$\|Z^{p,q}\|\leqslant\sup_{(x,t)\in[-1,1]\times[-1,1]}\|F(x,t)\|+\sup_{t\in[-1,1]}\|g(t)\|$$
$$+\sup_{t\in[-1,1]}\|h(t)\|+\sup_{x\in[-1,1]}\|f(x)\|.$$

Now combining (22), (23), and (24) we obtain the result:

$$\|U^{p,q}-\tilde{U}^{p,q}\|\leqslant O\left(\frac{1}{p^s}\right)\qquad\text{for all}\quad s>0.\ \blacksquare$$

III. PRECONDITIONING FOR SCALAR PROBLEMS

The system of equations

$$D^{p,q}U^{p,q}=Z^{p,q} \qquad (25)$$

is huge, full, and overdetermined. To obtain a least-squares solution to this problem we resort to iterative techniques. In this section we describe the numerical method for a scalar problem. In this and the next sections we shall denote our approximate solution $u^{p,q}(x,t)=\sum_{i=0}^{p}\sum_{j=0}^{q}a_{ij}^{p,q}T_i(x)\,T_j(t)$ by $u(x,t)$ and this should cause no confusion. Similarly we shall let U denote $U^{p,q}$, where $U^{p,q}=\{u^{p,q}(x_i^p,t_j^q)\}_{i=0,\dots,p,\,j=0,\dots,q}$ is the vector whose components are the $(p+1)\times(q+1)$ values of $u^{p,q}$ evaluated at the Gauss–Chebyshev Lobatto points. We can then write (25) in the equivalent form

$$D^{sp}U=Z,\qquad\text{where}\quad Z=Z^{p,q}. \qquad (26)$$

Since we wish to find a least-squares solution to this problem our solution U should minimize the residual

$$H(V)=\|D^{sp}V-Z\|^2,$$

and this suggests that we should seek the solution by using preconditioned residual minimization. For this we need to have an approximate inverse, which we shall denote by $(D^{ap})^{-1}$, to the matrix D^{sp} and we typically use a low order finite difference solver for $(D^{ap})^{-1}$. The method can then be described as:

(1) Given the current guess $U^{(n)}$ compute the residual

$$R^{(n)}=Z-D^{sp}U^{(n)}.$$

(2) Obtain an improvement $V^{(n)}$ for $U^{(n)}$ by computing

$$V^{(n)}=(D^{ap})^{-1}\,R^{(n)}.$$

(3) Update the current value of $U^{(n)}$ by putting $U^{(n+1)}=U^{(n)}+\omega_n V^{(n)}$, where ω_n is chosen as that value of ω at which $H(U^{(n)}+\omega V^{(n)})$ achieves its minimum. ω_n can be computed using the formula

$$\omega_n=\frac{(R^{(n)},\,D^{sp}V^{(n)})}{(D^{sp}V^{(n)},\,D^{sp}V^{(n)})},$$

where (,) denotes the standard inner product.

We now explain each step in more detail.

(1) Given the $(p+1)\times(q+1)$ values of $U^{(n)}$ which are the point values of $u^{(n)}(x,t)$ at the points $\{(x_i^p,t_j^q)\}_{i=0,\dots,p,\,j=0,\dots,q}$ we compute $\Gamma^{(n)}$, the coefficients in its representation as a Chebyshev series,

$$u^{(n)}(x,t)=\sum_{i=0}^{p}\sum_{j=0}^{q}\gamma_{ij}^{(n)}T_i(x)\,T_j(t).$$

This can be implemented using either a two-dimensional fast Chebyshev transform or, alternatively, by matrix multiplications. As the details of this are well known [7, 10] we do not go into it any further. Since we need to compute the residuals on a grid with $(2p+1)\times(2q+1)$ points we pad the representation of $u^{(n)}$ with zeros as

$$u^{(n)}(x,t)=\sum_{i=0}^{2p}\sum_{j=0}^{2q}\gamma_{ij}^{(n)}T_i(x)\,T_j(t), \qquad (27)$$

where $\gamma_{ij}^{(n)}=0$ for $i>p$ or $j>q$. We can now calculate the values of $u^{(n)}(x,t)$ at the $(2p+1)\times(2q+1)$ points $\{(x_i^{2p},t_j^{2q})\}_{i=0,\ldots,2p,j=0,\ldots,2q}$ by using an inverse transform or matrix multiplications. It is now an easy matter to compute the residuals

$$\begin{aligned}\rho_{ij}^{(n)}&=(u_t^{(n)}-au_x^{(n)}-bu^{(n)}-F)(x_i^{2p},t_j^{2q}),\\ \sigma_{ij}^{(n)}&=(\mathbf{M}(t_j^{2q})\,u^{(n)}(-1,t_j^{2q})-g(t_j^{2q})),\\ \eta_j^{(n)}&=(\mathbf{P}(t_j^{2q})\,u^{(n)}(-1,t_j^{2q})-h(t_j^{2q}))\\ \tau_j^{(n)}&=((u^{(n)}(x_i^{2p},-1)-f(x_i^{2p})).\end{aligned}\tag{28}$$

For the scalar problem we shall denote the matrices A and B by a and b.

The differentiations involved in computing (28) can be implemented using matrix multiplications or transform techniques. What is important to note is that these computations can be speeded up immensely using vectorization, as was pointed out by Orszag [7]. Henceforth we shall denote the vector of residuals $(\rho^{(n)},\sigma^{(n)},\eta^{(n)},\tau^{(n)})$ by $R^{(n)}$.

(2) We now seek an iterative improvement to the vector $U^{(n)}$ which we denote by the vector $V^{(n)}=\{v^{(n)}(_i^p,t_j^q)\}_{i=0,\ldots,p,j=0,\ldots,q}$ with $(p+1)\times(q+1)$ components. Let $W^{(n)}$ denote the prolongation of $V^{(n)}$ onto the grid with $(2p+1)\times(2q+1)$ points $\{(x_i^{2p},t_j^{2q})\}_{i=0,\ldots,2p,j=0,\ldots,2q}$ as described in (27). We can write this as

$$W^{(n)}=PV^{(n)},\tag{29}$$

where P denotes the prolongation operator.

It is natural to seek $V^{(n)}$ as the solution of the system of equations

$$D^{ap}V^{(n)}=D^{fd}PV^{(n)}=R^{(n)},\tag{30}$$

where D^{fd} is a finite difference discretization of the IBVP on the finer mesh. Then we have

$$V^{(n)}=P^{-1}(D^{fd})^{-1}\,R^{(n)},$$

where P^{-1} should be interpreted as the generalized inverse of the operator P. We can write this in two steps as:

(a) Compute $W^{(n)}=(D^{fd})^{-1}\,R^{(n)}$.

(b) Compute $V^{(n)}=P^{-1}W^{(n)}$.

We describe these steps further:

(a) In computing $W^{(n)}$ it is important to choose the finite difference operator so that it is easily invertible and stable. A first- or second-order implicit approximate factorization code based on central differencing ideally fulfils all these objectives [8, 9]. In fact, many of the general purpose simulation codes in use in research and industry utilize just this approach and it should be possible to modify them to perform spectral calculations. Further, since these codes involve the solution of a set of independent tridiagonal or block-tridiagonal matrix solvers the solution process can be vectorized. We refer the interested reader to [8, 9] for details.

We indicate the equations obtained from the finite difference discretization of

$$w_t^{(n)}-a(x,t)\,w_x^{(n)}-b(x,t)\,w^{(n)}=\rho^{(n)}(x,t)\tag{31}$$

at interior points of the space time square using implicit central differencing. Here $W^{(n)}$ denotes the vector with $(2p+1)\times(2q+1)$ values $\{w^{(n)}(x_i^{2p},t_j^{2q})=w_{ij}^{(n)}\}_{i=0,\ldots,2p,j=0,\ldots,2q}$. Let $a_{ij}=a(x_i^{2p},t_j^{2q})$ and $b_{ij}=b(x_i^{2p},t_j^{2q})$.

To advance the solution from time t_{j+1} to t_j we can use the implicit scheme

$$\begin{aligned}\frac{w_{i,j}-w_{i,j+1}}{\Delta t}-\frac{a_{ij}}{2}\Bigg[&\left\{\frac{w_{i-1,j+1}-w_{i,j+1}}{\Delta x}+\frac{w_{i,j+1}-w_{i+1,j+1}}{\Delta x'}\right.\\&\left.-\frac{w_{i-1,j+1}-w_{i+1,j+1}}{\Delta x'+\Delta x}\right\}\\&+\left\{\frac{w_{i-1,j}-w_{i,j}}{\Delta x}+\frac{w_{i,j}-w_{i+1,j}}{\Delta x'}\right.\\&\left.-\frac{w_{i-1,j}-w_{i+1,j}}{\Delta x+\Delta x'}\right\}\Bigg]-b_{ij}\left\{\frac{w_{i,j}+w_{i,j+1}}{2}\right\}=\rho_{i,j}\end{aligned}$$

which is second-order accurate.

This can be written as

$$\begin{aligned}&\alpha_iw_{i-1,j}+\beta_iw_{i,j}+\gamma_iw_{i+1,j}\\&\quad=\rho_{i,j}-\{\tilde{\alpha}_iw_{i-1,j+1}+\tilde{\beta}_iw_{i,j+1}+\tilde{\gamma}_iw_{i+1,j+1}\},\end{aligned}\tag{32}$$

where

$$\alpha_i=\left\{-\frac{1}{\Delta x}+\frac{1}{\Delta x+\Delta x'}\right\}\frac{a_{ij}}{2},$$

$$\tilde{\alpha}_i=\left\{-\frac{1}{\Delta x}+\frac{1}{\Delta x'+\Delta x}\right\}\frac{a_{ij}}{2}$$

$$\beta_i=\left\{-\frac{1}{\Delta t'}-\frac{a_{ij}}{2}\left(-\frac{1}{\Delta x}+\frac{1}{\Delta x'}\right)-\frac{b_{ij}}{2}\right\},$$

$$\tilde{\beta}_i=\left\{-\frac{1}{\Delta t'}-\frac{a_{ij}}{2}\left(-\frac{1}{\Delta x}+\frac{1}{\Delta x'}\right)-\frac{b_{ij}}{2}\right\},$$

$$\gamma_i=\left\{\frac{1}{\Delta x}+\frac{1}{\Delta x+\Delta x'}\right\}\frac{a_{ij}}{2},$$

$$\tilde{\gamma}_i=\left\{\frac{1}{\Delta x}-\frac{1}{\Delta x+\Delta x'}\right\}\frac{a_{ij}}{2}.$$

The above equation uses information from the six-point stencil shown in Fig. 1. Thus to advance from time level t_{j+1} to t_j we have to solve a tridiagonal system. To initialize the procedure we impose the initial conditions

$$w^{(n)}_{i,2q} = \tau^{(n)}_i, \qquad 0 \leqslant i \leqslant 2q. \tag{33}$$

We can impose the boundary conditions either implicitly or explicitly. Inflow boundary conditions pose no problem. Thus if $x = -1$ is an inflow boundary for (31) we simply impose the boundary condition

$$w^{(n)}_{2p,j} = \sigma^{(n)}_j. \tag{34}$$

If it is an outflow boundary, however, we either impose the partial differential equation at the boundary implicitly or use extrapolation techniques [3]. Our computational results show that the implicit treatment is preferable, so we shall say a few words about it.

We use the four-point stencil shown in Fig. 2 to obtain the equation

$$-\frac{1}{2}\left\{\frac{w_{2p,j}-w_{2p,j+1}}{\Delta t}+\frac{w_{2p-1,j}-w_{2p-1,j+1}}{\Delta t}\right\}$$
$$-\frac{a_{2p,j}}{2}\left\{\frac{w_{2p-1,j}-w_{2p,j}}{\Delta x}+\frac{w_{2p-1,j+1}-w_{2p,j+1}}{\Delta x}\right\}$$
$$-b_{2p,j}\left\{\frac{w_{2p,j}+w_{2p,j+1}+w_{2p-1,j}+w_{2p-1,j+1}}{4}\right\}=\sigma_j.$$

This is of the form

$$\alpha_{2p} w_{2p-1,j} + \beta_{2p} w_{2p,j} = \sigma_j - \{\tilde{\alpha}_{2p} w_{2p-1,j+1} + \tilde{\beta}_{2p} w_{2p,j+1}\}, \tag{35}$$

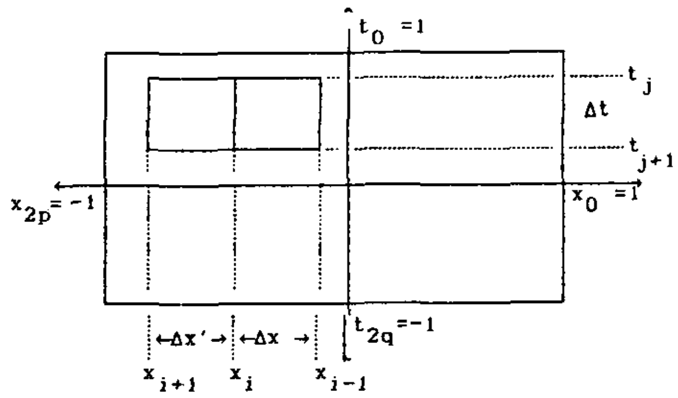

FIGURE 1

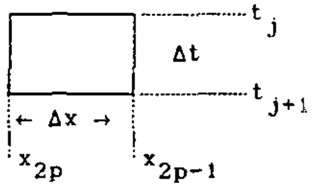

FIGURE 2

where

$$\alpha_{2p} = \frac{1}{2\Delta t} - \frac{a_{2p,j}}{2\Delta x} - \frac{b_{2p,j}}{4},$$

$$\tilde{\alpha}_{2p} = \frac{1}{2\Delta t} - \frac{a_{2p,j}}{2\Delta x} - \frac{b_{2p,j}}{4}$$

$$\beta_{2p} = -\frac{1}{2\Delta t} + \frac{a_{2p,j}}{2\Delta x} - \frac{b_{2p,j}}{4},$$

$$\tilde{\beta}_{2p} = -\frac{1}{2\Delta t} + \frac{a_{2p,j}}{2\Delta x} - \frac{b_{2p,j}}{4}.$$

Note, that with this treatment of the boundary condition our system of equations remains tridiagonal; and with this we conclude our discussion of (a).

Computation for the Scalar Case

EXAMPLE 1.

$$u_t - (x + \varepsilon(x,t))\, u_x = F(x,t), \qquad -1 \leqslant x \leqslant 1, \quad -1 \leqslant t \leqslant 1$$

$$F(x,t) = \varepsilon(x,t) \sin\left(\frac{\pi}{16} x e^t\right) \frac{\pi}{16} e^t,$$

with initial conditions,

$$u(x,0) = \sin\left(\frac{\pi}{16} x\right),$$

and boundary condition,

$$u(-1,t) = u(1,t) = \sin\left(\frac{\pi}{16} e^t\right).$$

The results obtained for three different values of $\varepsilon(x)$:

(1) $\varepsilon_1(x,t) = 0.5 \times \sin(3\pi(x+t))$,
(2) $\varepsilon_2(x,t) = 0.5 \times \sin(4\pi(x+t))$,
(3) $\varepsilon_3(x,t) = 0.5 \times \sin(5\pi(x+t))$,

are shown in Table I.

TABLE I

N	$0.5\times\sin(3\pi(x+t))$	$0.5\times\sin(4\pi(x+t))$	$0.5\times\sin(5\pi(x+t))$
33	⟨2⟩ (5.03×10^{-03})	⟨2⟩ (6.69×10^{-03})	⟨3⟩ (6.96×10^{-03})
	⟨26⟩ (2.69×10^{-10})	⟨30⟩ (2.32×10^{-10})	⟨43⟩ (3.75×10^{-10})

Note. N = number of collocation points.

EXAMPLE 2.

$$u_t + xu_x = 0$$

with initial condition,

$$u(x, 0) = f(x).$$

Results obtained for three sets of initial data are given in Table II:

(i) $f(x) = \sin((\pi/16)x)$
(ii) $f(x) = \sin((\pi/33)x)$
(iii) $f(x) = \sin((\pi/100)x)$.

Note that in Example No. 1 u is an inflow variable at $x = -1$ and $x = 1$; that is why the value of u is prescribed at both the boundaries. In Example No. 2 u is an outflow variable at $x = -1$ and $x = 1$, so the value of u at both the boundaries is obtained by enforcing the partial differential equation there. Orszag [7] had advocated a filter, in which the top one-third of the frequency components of the numerical solution are removed and which he has referred to as the two-thirds rule, for the preconditioning to be effective. In Table III we give comparative results for the two-thirds filter and the one-half filter, advocated by us.

It can be seen from Table III that the one-half filter performs better than the two-thirds filter.

TABLE II

N	$\sin\left(\frac{\pi}{100}x\right)$	$\sin\left(\frac{\pi}{33}x\right)$	$\sin\left(\frac{\pi}{16}x\right)$
33	⟨2⟩ (2.11×10^{-08})	⟨2⟩ (4.65×10^{-08})	⟨2⟩ (8.86×10^{-08})
	⟨9⟩ (5.01×10^{-14})	⟨13⟩ (8.58×10^{-14})	⟨12⟩ (1.75×10^{-12})

Note. N = number of collocation points.

IV. PRECONDITIONING FOR THE SYSTEM CASE

Consider the hyperbolic system,

$$w_t - Aw_x - Bw = F, \qquad -1 \leqslant x, \quad t \leqslant 1, \tag{36.a}$$

where

$$w = (w_1, w_2),$$
$$A = \begin{pmatrix} a_{11}(x,t) & a_{12}(x,t) \\ a_{21}(x,t) & a_{22}(x,t) \end{pmatrix},$$
$$B = \begin{pmatrix} b_{11}(x,t) & b_{12}(x,t) \\ b_{21}(x,t) & b_{22}(x,t) \end{pmatrix},$$
$$F = (F_1(x,t), F_2(x,t))^{\mathrm{T}}.$$

We prescribe boundary conditions,

$$\mathbf{M}w(-1, t) = g(t) \tag{36.b}$$
$$\mathbf{P}w(1, t) = h(t), \tag{36.c}$$

and initial condition,

$$w(x, -1) = f(x). \tag{36.c}$$

Here $\mathbf{M}$ is an $l\times 2$ and P is an $s\times 2$ matrix, where $0 \leqslant l, s \leqslant 2$;

TABLE III

$f(x)$	$a = x$: 1/2 filter	$a = x$: 2/3 filter	$a = -x$: 1/2 filter	$a = -x$: 2/3 filter
$\sin\frac{\pi}{100}x$	⟨2⟩ (1.71×10^{-06})	⟨2⟩ (5.22×10^{-6})	⟨2⟩ (2.11×10^{-08})	⟨2⟩ (2.09×10^{-08})
	⟨58⟩ (4.46×10^{-14})	⟨65⟩ (3.32×10^{-7})	⟨9⟩ (5.01×10^{-14})	⟨10⟩ (6.43×10^{-10})
$\sin\frac{\pi}{33}x$	⟨2⟩ (5.21×10^{-06})	⟨2⟩ (1.55×10^{-5})	⟨2⟩ (4.65×10^{-08})	⟨2⟩ (4.57×10^{-08})
	⟨51⟩ (8.67×10^{-13})	⟨60⟩ (3.07×10^{-7})	⟨13⟩ (8.98×10^{-14})	⟨16⟩ (1.74×10^{-08})
$\sin\frac{\pi}{16}x$	⟨3⟩ (6.60×10^{-06})	⟨2⟩ (3.03×10^{-5})	⟨2⟩ (8.86×10^{-08})	⟨2⟩ (1.16×10^{-07})
	⟨50⟩ (2.31×10^{-12})	⟨50⟩ (1.75×10^{-6})	⟨12⟩ (1.75×10^{-12})	⟨15⟩ (7.29×10^{-08})

Note. For $u_t - au_x = 0$ and $u(x, 0) = f(x)$ (Example 2), where the number of collocation points is 33.

$l=0$ means there are no boundary conditions at $x=-1$ and $s=0$ means that there are no boundary conditions at $x=1$.

We use the central differenced discretization of

$$w_t^{(n)} - A(x,t)\, w_x^n - B(x,t)\, w^n = \rho^{(n)}(x,t) \tag{37}$$

at interior points of the space-time square to advance the solution from time level t_{j+1} to t_j.

As we have seen in the scalar case the equation uses information from a six-point stencil. Thus to advance from time level t_{j+1} to t_j we have to solve a block-tridiagonal system. To initialize the procedure we impose the initial condition

$$w_{i,2q}^{(n)} = \tau_i^{(n)}, \qquad 0 \leqslant i \leqslant 2q.$$

It is evident that block-tridiagonal matrix solver constitutes the major portion of the numerical computation of the standard implicit algorithm. Equation (39) produces a 2×2 block structure for the implicit operator. Pulliam and Chaussee [8] have given an algorithm which transforms the coupled system of equations into an uncoupled diagonal form that requires considerably less computational work.

We describe this algorithm in brief for the system case. An implicit approximate factorization scheme for the system can be written as

$$\left(I - \Delta t\, A_{ij} \frac{\delta}{\delta x}\right) \Delta w_j = R_{ij}, \tag{38}$$

where $\Delta w_j = w_{j+1} - w_j$ and $A_{ij} = A(x_i, t_j)$, as we can handle the lower order term explicitly. Here $\delta/\delta x$ denotes the centered difference approximation to the differential operator $\partial/\partial x$.

The matrix A_{ij} has a set of real eigenvalues and a complete set of eigenvectors; hence a similarity transformation can be used to diagonalize $A_{i,j}$,

$$A_{ij} = T_{i,j} \wedge_{i,j} T_{i,j}^{-1}, \tag{39}$$

so we write (38) as

$$\left(T_{i,j} T_{i,j}^{-1} - \Delta t\, T_{i,j} \wedge_{i,j} T_{i,j}^{-1} \frac{\delta}{\delta x}\right) \Delta w_j = R_{ij}. \tag{40}$$

The modified form of the above equation is constructed by moving T outside the difference operator $\delta/\delta x$. This results in the *diagonal* form of the algorithm

$$T_{i,j}\left(I - \Delta t \wedge_{i,j} \frac{\delta}{\delta x}\right) T_{i,j}^{-1}\, \Delta w_j = R_{i,j}.$$

The modification has introduced an error, but Pulliam and Chaussee have shown that the error introduced by the diagonalization is first order in time. The new implicit operator $(I - \Delta t \wedge_{i,j} (\delta/\delta x))$ is still block-tridiagonal, but now blocks are diagonal in form so that the operator reduces to two independent scalar tridiagonal operators.

Numerical Treatment of Boundary Conditions

A correct treatment of the boundary conditions is essential for an effective spectral calculation. If incorrect boundary conditions are imposed on the numerical scheme the resulting errors will propagate into the computational domain. If these errors propagate and/or grow sufficiently rapidly, they will destroy the solution.

Since the system is hyperbolic, A has real eigenvalues and a complete set of eigenvectors. So there exists a matrix T such that TAT^{-1} is diagonal. Equation (36.a) can be rewritten as

$$Tw_t - TAT^{-1}Tw_x - TBT^{-1}Tw = TF$$

or

$$\tilde{w}_t - \wedge\, \tilde{w}_x - \tilde{B}\tilde{w} = \tilde{F}.$$

Here

$$\tilde{w} = Tw,$$

with

$$\tilde{w} = (\tilde{w}_1, \tilde{w}_2),$$
$$\wedge = TAT^{-1},$$
$$\tilde{B} = T_t T^{-1} - \wedge\, T_x T^{-1} - TBT^{-1},$$
$$\tilde{F} = TF.$$

The variables $\tilde{w}_1$ and $\tilde{w}_2$ are called characteristic variables.

Assume that $\tilde{w}_1$ is an inflow variable and that $\tilde{w}_2$ is an outflow variable at $x=-1$. Then the boundary operator $\mathbf{M}$ would be of the form

$$\mathbf{M}w(-1,t) = \tilde{w}_1(-1,t) - \alpha(t)\, \tilde{w}_2(-1,t), \tag{41}$$

where $\alpha(t)$ is a function of t. Hence the boundary condition at $x=-1$ could be written as

$$\mathbf{M}w(-1,t) = g(t).$$

For the outflow variable we impose the partial differential equation at the boundary implicitly. If we were to impose (41) in the form

$$\tilde{w}_1(-1,t_j) - \alpha(t_j)\, \tilde{w}_2(-1,t_j) = g(t_j), \tag{42}$$

the difference equation would no longer decouple into a set of tridiagonal equations but instead would become block-tridiagonal. We can get around this problem by an *approximate treatment of the boundary condition.* In (42) we approximate the unknown value of $\tilde{w}_2(t_j)$ by using either

(a) extrapolation or

(b) an explicit finite difference discretization of the partial differential equation.

It is easy to show that both these techniques, which we describe below, are GKSO (Gustafsson, Kreiss, Sundström, and Osher) stable for a uniform mesh.

(a.i) *Zeroth-order extrapolation.* Here we simply put $\tilde{w}_2(-1, t_j) = \tilde{w}_2(-1, t_{j+1})$ for $2q-1 \geqslant j \geqslant 1$.

(a.ii) *First-order extrapolation.* We define

$$\tilde{w}_2(-1, t_{2q-1}) = \tilde{w}_2(-1, t_{2q})$$

and

$$\tilde{w}_2(-1, t_j) = \tilde{w}_2(-1, t_{j+2}) + (\Delta t' + \Delta t) \times \frac{\tilde{w}_2(-1, t_{j+1}) - \tilde{w}_2(-1, t_{j+2})}{\Delta t},$$

$$0 \leqslant j \leqslant 2q-2.$$

(b) *Explicit difference scheme.* Here we use an explicit difference scheme to compute $\tilde{w}_1(-1, t_j)$ from the values of $\tilde{w}(-1, t_{j+1})$ and $\tilde{w}(x_{2p-1}, t_{j+1})$. Since the boundary condition (b) does not give good results we omit describing it in detail.

Computational Results for the System Case

EXAMPLE I.

$$A = \begin{pmatrix} -0.5 + 0.01 \times \sin(\pi x) & 0.01 \\ 0.5 & 0.5 + 0.05 \times \cos(\pi(x+t)) \end{pmatrix},$$

$$B = 0,$$

and $F(x, t) = \{f_1(x, t), f_2(x, t)\}$. If our characteristic variables are $\tilde{u}_1(x, t)$ and $\tilde{u}_2(x, t)$ then $\tilde{u}_1(x, t)$ is an inflow variable at $x = 1$ and $\tilde{u}_2(x, t)$ is an inflow variable at $x = -1$.

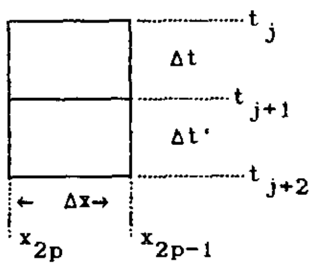

FIGURE 3

Case I. Our solution is

$$u(x, t) = \begin{pmatrix} \sin((\pi/33)\, x e^t) \\ \cos(2x - 3t) \times \varepsilon \end{pmatrix}_{\varepsilon = 0.1}$$

and the boundary conditions are of the form

$$\tilde{u}_2(-1, t) - 2 \times \sin(t)\, \tilde{u}_1(-1, t) = g(t),$$
$$\tilde{u}_1(1, t) - e^t \tilde{u}_2(1, t) = h(t)$$

with initial data

$$u(x, -1) = f(x).$$

We omit writing the rather involved expressions for F, g, h, f.

Case II. We choose

$$u(x, t) = \begin{pmatrix} \sin((\pi/16)\, x e^t) \\ \cos(2x - 3t) \times \varepsilon \end{pmatrix}_{\varepsilon = 0.1}$$

Then the boundary conditions are

$$\tilde{u}_2(-1, t) - \tilde{u}_1(-1, t) = g(t),$$
$$\tilde{u}_1(1, t) - \tilde{u}_2(1, t) = h(t)$$

with initial data

$$u(x, -1) = f(x).$$

In A and B the value of the inflow variable at the boundary is obtained using boundary conditions (a.i) and (a.ii), respectively (see Table IV).

EXAMPLE II.

$$A = \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & -0.5 \end{pmatrix}.$$

TABLE IV

	A		*B*	
	Iterations	Error	Iterations	Error
Case I	4	9.248×10^{-03}	2	2.300×10^{-03}
	56	5.276×10^{-10}	50	5.529×10^{-10}
Case II	5	8.868×10^{-03}	2	2.890×10^{-03}
	57	4.319×10^{-10}	51	2.383×10^{-10}

Note. The number of collocation points is 33.

TABLE V

	A		B	
	Iterations	Error	Iterations	Error
Case I	10	3.387×10^{-02}	2	7.348×10^{-03}
	25	2.048×10^{-07}	12	2.078×10^{-07}
Case II	45	7.521×10^{-03}	2	4.135×10^{-04}
	99	1.267×10^{-11}	50	9.393×10^{-13}
Case III	2	1.274×10^{-00}	2	8.812×10^{-05}
	25	6.163×10^{-01}	42	5.435×10^{-13}

Note. The number of collocation points is 33.

Our solution is $u(x, t) = \{u_1(x, t), u_2(x, t)\}^{\mathrm{T}}$.

Case I.

$$u_1(x, t) = x^3 + xt^2 + 100t^7 + \cos(t)$$

$$u_2(x, t) = t^4 + (x+t)^3 + \sin(x).$$

Case II.

$$u_1(x, t) = \cos\left(\frac{\pi}{100}\sin(x+t)\right)$$

$$u_2(x, t) = t^4 + (x+t)^3 + \sin(x).$$

Case III.

$$u_1(x, t) = \cos\left(\frac{\pi}{100}\sin(x+t)\right)$$

$$u_2(x, t) = \sin(2x - t).$$

with boundary conditions,

$$u_1(-1, t) - u_2(-1, t) = g(t)$$

$$u_2(1, t) - u_1(1, t) = h(t),$$

and the initial condition is

$$u(-1, t) = f(x).$$

In A and B the value of the inflow variable at the boundary is obtained using boundary conditions (b) and (a.ii), respectively. From Table V we conclude that boundary condition (b) gives poor results in general.

IV. PRECONDITIONING FOR NONLINEAR PROBLEMS

We now consider the nonlinear IBVP

$$u_t - A(u)\, u_x = F(x, t), \qquad -1 \leqslant x \leqslant 1, \quad -1 \leqslant t \leqslant 1, \tag{43.a}$$

with boundary conditions,

$$\mathbf{M}u(-1, t) = g(t), \qquad -1 \leqslant t \leqslant 1, \tag{43.b}$$

$$\mathbf{P}u(1, t) = h(t), \qquad -1 \leqslant t \leqslant 1, \tag{43.c}$$

and initial condition,

$$u(x, -1) = f(x), \qquad -1 \leqslant x \leqslant 1; \tag{43.d}$$

and we assume that the solution u is smooth. We choose as our approximate solution $u^{p,q} \in (S^{p,q})^k$ which minimizes

$$\begin{aligned} H^{p,q}(v^{p,q}) = {} & \frac{\pi^2}{4pq} \sum_{j=0}^{2q} \sum_{i=0}^{2p} \|(v_t^{p,q} - A(v^{p,q})\, v_x^{p,q} - F) \\ & \times (x_i^{2p}, t_j^{2q})\|^2 \\ & + \frac{\pi}{2q} \sum_{j=0}^{2q} \|(\mathbf{M}v^{p,q}(-1, t_j^{2q}) - g(t_j^{2q})\|^2 \\ & + \frac{\pi}{2q} \sum_{j=0}^{2q} \|(\mathbf{P}v^{p,q}(1, t_j^{2q}) - h(t_j^{2q})\|^2 \\ & + \frac{\pi}{2p} \sum_{i=0}^{2p} \|(v^{p,q}(x_i^{2p}, -1) - f(x_i^{2p})\|^2 \end{aligned} \tag{44}$$

over all $v^{p,q} \in (S^{p,q})^k$. Clearly, this gives rise to a nonlinear least-squares problem, which we may write as

$$L^{sp}(U)\, U = Z. \tag{45}$$

We solve this nonlinear least-squares problem by the preconditioned residual minimization method as before. We outline the main steps:

(1) To obtain an initial guess $U^{(0)}$ for the solution we let $V^{(0)}$ be the solution obtained on the finer mesh with $(2p+1) \times (2q+1)$ points by using a first- or second-order finite difference solver for the nonlinear IBVP (43). Then we obtain $U^{(0)}$ from $V^{(0)}$ by truncating the highest half of its frequency components as before.

(2) Suppose at the nth stage of the iteration that we have an approximate solution $U^{(n)}$ corresponding to $u^{(n)}(x, t)$. We can now calculate the residuals,

$$\begin{aligned} \rho^{p,q}(x, t) &= u_t^{(n)} - A(u^{(n)})\, u_x^{(n)} - \bar{F}^{2p,2q}(x, t), \\ \sigma^q(t) &= \mathbf{M}^q u^{(n)}(-1, t) - \bar{g}^{2q}(t), \\ \eta^q(t) &= \mathbf{P}^q u^{(n)}(1, t) - \bar{h}^{2q}(t), \\ \tau^p(x) &= u^{(n)}(x, -1) - \bar{f}^{2p}(x). \end{aligned} \tag{46}$$

We wish to find a correction $v(x, t)$ to $u^n(x, t)$; corresponding to the function $v(x, t) \in (S^{p,q})^k$ we have the vector V. Then $v(x, t)$ should *approximately* satisfy

$$v_t - A(u^n)\, v_x - A(u^n)_x\, v = \rho^{(n)}(x, t), \qquad -1 \leqslant t \leqslant 1, \quad -1 \leqslant x \leqslant 1, \tag{47.a}$$

$$\left(\left[\frac{\partial \mathbf{M}}{\partial u}\right]^v_{u=u^n}\right)(-1, t) = \sigma^{(n)}(t), \qquad -1 \leqslant t \leqslant 1, \tag{47.b}$$

$$\left(\left[\frac{\partial \mathbf{P}}{\partial u}\right]^v_{u=u^n}\right)(1, t) = \eta^{(n)}(t), \qquad -1 \leqslant t \leqslant 1, \tag{47.c}$$

$$v(x, -1) = \tau^{(n)}(x), \qquad -1 \leqslant x \leqslant 1, \tag{47.d}$$

which is obtained by linearizing (43) about $u^{(n)}$. Thus v can be obtained as the solution of a linear IBVP. Hence we can use the preconditioning techniques already described to obtain a correction V. Specifically, let W be the solution obtained on the finer mesh by a finite difference solver for (47); then V is obtained by truncating the highest half of the frequency components of W. We compute the relaxation factor ω_n so as to minimize the residual:

$$\begin{aligned} H^{p,q}(\omega) = {} & \frac{\pi^2}{4pq} \sum_{j=0}^{2q} \sum_{i=0}^{2p} \|(\omega v_t - \omega A(u^{(n)})\, v_x - \rho^{(n)}) \times (x_i^{2p}, t_j^{2q})\|^2 \\ & + \frac{\pi}{2q} \sum_{j=0}^{2q} \left\| \left(\omega \left[\frac{\partial \mathbf{M}}{\partial u}\right]^v_{u=u^n}\right)(-1, t^{2q}) - \sigma^{(n)}(t_j^{2q}) \right\|^2 \\ & + \frac{\pi}{2q} \sum_{j=0}^{2q} \left\| \left(\omega \left[\frac{\partial \mathbf{P}}{\partial u}\right]^v_{u=u^n}\right)(1, t_j^{2q}) - \eta^{(n)}(t_j^{2q}) \right\|^2 \\ & + \frac{\pi}{2p} \sum_{j=0}^{2p} \|(\omega v(x_i^{2p}, -1) - \tau^{(n)}(x_i^{2p})\|^2. \end{aligned} \tag{48}$$

We then define

$$U^{(n+1)} = U^{(n)} + \omega_n V.$$

We remark that our numerical experiments indicate that it is enough to consider v as an approximate solution to the partial differential equation,

$$v_t - A(u^{(n)})\, v_x = \rho^{(n)}, \qquad -1 \leqslant t \leqslant 1, \quad -1 \leqslant x \leqslant 1,$$

along with the initial and boundary conditions (47.b)–(47.d) to obtain convergence of the numerical scheme.

Computational Results for the Nonlinear Case

EXAMPLE I. We consider the Burger's equation

$$u_t + u u_x = F(x, t), \tag{49.a}$$

with $u(x, t) = -10 + \sin((\pi/16)\, x e^t)$.

TABLE VI

	Nonlinear results	
	Iterations	Error
Example I	6	6.487×10^{-03}
	25	1.682×10^{-10}
Example II	6	6.463×10^{-03}
	25	1.571×10^{-10}

Note. The number of collocation points is 33.

Then we have to impose the boundary condition

$$u(-1, t) = g(t), \qquad -1 \leqslant t \leqslant 1, \tag{49.b}$$

and no boundary condition at $x = 1$. The initial condition is

$$u(x, -1) = f(x). \tag{49.c}$$

EXAMPLE II. We consider the isentropic Euler equation,

$$\begin{pmatrix} u \\ \rho \end{pmatrix}_t - \begin{pmatrix} u & \gamma \rho^{\gamma-2} \\ \gamma \rho & u \end{pmatrix} \begin{pmatrix} u \\ \rho \end{pmatrix}_x = F,$$

$$u(x, t) = x^2 + 10 + \cos\left(\frac{\pi}{16} \sin(x + t)\right) \tag{50}$$

$$\rho(x, t) = x^2 + 2 + \sin(2x - t)^2, \qquad \gamma = 2.0.$$

The flow is supersonic, so we do not have to specify a boundary condition at $x = -1$ or at $x = 1$ for both ρ and u.

REFERENCES

1. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer Series in Computational Physics (Springer-Verlag, New York/Berlin, 1987).
2. P. Dutt, *SIAM J. Numer. Anal.* **27** (4), 885 (1990).
3. M. Goldberg and E. Tadmor, *Math. Comput.* **32**, 1097 (1978).
4. D. Gottlieb, M. Y. Hussaini, and S. A. Orszag, "Theory and Application of Spectral Methods," in *Spectral Methods for Partial Differential Equations*, edited by R. G. Voigt, D. Gottlieb, and M. Y. Hussaini (SIAM-CBMS, Philadelphia, 1984), p. 1.
5. Y. Morchoisne, *Reach. Aerosp.* **5**, 293 (1979).
6. Y. Morchoisne, "Inhomogeneous Flow Calculations for Spectral Methods: Mono-Domain and Multi-Domain Techniques," in *Spectral Methods for Partial Differential Equations*, edited by R. G. Voigt, D. Gottlieb, and M. Y. Hussaini (SIAM-CBMS, Philadelphia, 1984), p. 181.
7. S. A. Orszag, *J. Comput. Phys.* **37**, 70 (1980).
8. T. H. Pulliam and J. Chaussee, *J. Comput. Phys.* **39**, 347 (1981).
9. J. L. Steger and P. G. Buning, "Developments in the Simulation of Compressible inviscid and viscous Flow on Supercomputers," in *Progress and Super-computing in Computational Fluid Dynamics*, edited by E. M. Murman and S. S. Abarbanel (Birkhäuser, Boston, 1985), p. 357.
10. T. A. Zang, Y. S. Wong, and M. Y. Hussaini, *J. Comput. Phys.* **54**, 489 (1984).