Journal of Computational Physics **160,** 42–66 (2000)
doi:10.1006/jcph.2000.6420, available online at http://www.idealibrary.com on IDEAL®

Strict Stability of High-Order Compact Implicit Finite-Difference Schemes: The Role of Boundary Conditions for Hyperbolic PDEs, I

Saul S. Abarbanel* and Alina E. Chertock†

Department of Applied Mathematics, School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel
E-mail: *saul@math.tau.ac.il, †cheral@math.tau.ac.il and alina@math.ibl.gov

Received March 2, 1999; revised November 30, 1999

Temporal, or "strict," stability of approximation to PDEs is much more difficult to achieve than the "classical" Lax stability. In this paper, we present a class of finite-difference schemes for hyperbolic initial boundary value problems in one and two space dimensions that possess the property of strict stability. The approximations are constructed so that all eigenvalues of corresponding differentiation matrix have a nonpositive real part. Boundary conditions are imposed by using penalty-like terms. Fourth- and sixth-order compact implicit finite-difference schemes are constructed and analyzed. Computational efficacy of the approach is corroborated by a series of numerical tests in 1-D and 2-D scalar problems.

Key Words: hyperbolic PDEs; boundary conditions; stability; accuracy; error bounds.

1. INTRODUCTION

In many computational problems, including, for example, acoustics, electromagnetic wave propagation, and fluid dynamic, low-order schemes (second or lower) are not accurate enough. The advantage of high-order finite-difference schemes is twofold: they allow one either to increase accuracy while keeping the number of mesh points fixed or to reduce the computational cost by decreasing the grid dimension while preserving accuracy. And although they require more work per node, the fact that fewer points need to be stored and computed makes them more efficient than low-order methods [8].

One of the main reasons that low-order schemes are still used in practical computations is that difficulty arises for the high-order finite-difference schemes near the boundaries of computational domain. To retain the formal accuracy of the high-order scheme, boundary closures must be accomplished with the same accuracy as that of the interior scheme, or at most one order less [6, 7]. On a Cartesian mesh, it is always possible to derive nonsymmetrical

0021-9991/00 $35.00

boundary operators that fulfill the boundary conditions and maintain the overall accuracy of the scheme. The difficulty is in deriving high accuracy and *stable* operators.

While dealing with the numerical integration of time-dependent PDEs, two different limit processes can be considered. One limit is the behavior of the numerical solution as the mesh size $h \to 0$ for a fixed time T. Another is the behavior of the solution for a fixed mesh size h as the time T tends to infinity. "Classical" stability addresses the first issue: boundedness of the numerical solution as the mesh is refined at a fixed time T. In this case, Lax's equivalence theorem ensures that the scheme converges, i.e., that for a fixed time T the numerical solution converges to the analytical solution as the mesh size $h \to 0$. Nothing in this definition excludes error growth in time, and specifically it allows exponential growth of the error in time. Unfortunately, for time-dependent problems, this stability definition might be too weak, in particular if long time integration is being carried out. For long time numerical simulations to be useful, the numerical solution must be strictly stable in time. In the case of semidiscrete approximations, strict stability implies that for a fixed mesh size h, all eigenvalues of the coefficient matrix of the corresponding system of ordinary differential equations have a nonpositive real part. For calculation over long time intervals, strict stability is especially important because it prevents exponential growth in time of the error for a fixed mesh size h.

In the work by Carpenter *et al*. [3] it was shown that many high-order scalar schemes, which are stable in the classical sense, are not time stable. Moreover, it was recently found that many high-order schemes that are strictly stable in the scalar case exhibit time divergence when applied to systems of equations. The underlying reason for the error growth in time is improper imposition of boundary conditions.

For the scalar explicit central-differencing case, Kreiss and Scherer [9] have presented a method for constructing a boundary condition of accuracy one order less than the inner scheme such that a generalized summation by parts property of the differential equation is preserved. Strand [12] obtained the stability results for explicit high-order finite difference approximations using the G-K-S stability theory for the semidiscrete initial boundary value problems (IBVPs). To close the scheme near the boundary he obtained extra boundary conditions by extrapolating the outgoing characteristic variables, and by differentiating the analytic boundary conditions and using the partial differential equation for the incoming characteristic variables. However, in some cases the approximation with such boundary conditions had eigenvalues with a positive real part, and to assure the time stability of the scheme the numerical boundary conditions were modified by adding dissipative terms into the inflow part of the boundary conditions.

In the present work a methodology for constructing compact implicit high-order finite-difference schemes for hyperbolic initial boundary value problems is presented. The SAT procedure for imposing the analytical boundary conditions proposed by Carpenter *et al*. in [4] is generalized in such a way that: (i) it essentially simplifies the construction of the approximation of the desirable accuracy from the technical point of view and (ii) it allows one, in principle, to apply this technique to the solution of multidimensional problems. Temporal stability in one space dimension is achieved by constructing such approximations that all eigenvalues of the coefficient matrix of the corresponding system of ordinary differential equations have a negative real part. On the other hand, convergence of the scheme is proved directly by deriving an equation for the error and bounding the error norm. In order to solve two-dimensional scalar problems $\partial/\partial x + \partial/\partial y$ is approximated by the sum of two differentiation matrices $D_x + D_y$, where both D_x and D_y have eigenvalues with a negative

real part. Since the sum matrix $D_x + D_y$ does not necessarily preserve this virtue, strict stability of the scheme is proved by showing that $H(D_x + D_y) + [H(D_x + D_y)]^T \leq 0$ for any symmetric positive definite matrix H. Numerical studies on hyperbolic scalar IBVPs in one and two space dimensions have been performed using fourth- and sixth-order compact implicit difference schemes. Boundary conditions have been imposed using the SAT boundary procedure. The numerical results support the theoretical analysis. It has been shown that the actual numerical solution has a temporal error bounded by a constant.

In Part II of this work [13] the above procedures are implemented for the cases of 1-D and 2-D *systems of hyperbolic PDEs*. Partial reflection and/or absorbtion at the boundaries render the analysis more complex. For example, the construction of the differentiation matrices allows for nonpositiveness of the real part of the eigenvalue. Results of similar quality to those of Part I are reported there.

2. 1-D HYPERBOLIC EQUATIONS

2.1. *Description of the Method and Proof of the Main Theorem*

We consider the scalar hyperbolic equation

$$\frac{\partial u}{\partial t} + \lambda \frac{\partial u}{\partial x} = 0, \qquad 0 \leq x \leq 1, t \geq 0 \tag{2.1}$$

with initial conditions prescribed at $t = 0$,

$$u(x, 0) = f(x), \qquad 0 \leq x \leq 1. \tag{2.2}$$

For positive λ we have the boundary condition

$$u(0, t) = g(t), \qquad t \geq 0. \tag{2.3}$$

We want to solve the above problem by finite difference approximations. In this work, we will deal with compact schemes for the discretization of the spatial operator $\frac{\partial}{\partial x}$. We therefore introduce the mesh width h and divide the interval [0, 1] into subintervals of length h. We use with $j = 0, \ldots, N$ and $N = 1/h$ the notation

$$x_j = jh, \qquad u_j(t) = u(x_j, t), \tag{2.4}$$

where $u_j(t)$ is the projection of the exact solution $u(x, t)$ unto the grid. We denote by $\vec{u}$ the vector $(u_0(t), , \ldots, u_N(t))^T$ and by $\vec{v}$ the numerical approximation to the projection $\vec{u}$.

The implicit approximation for the first derivative can be written as

$$P \frac{\partial \vec{v}}{\partial x} = Q\vec{v}, \tag{2.5}$$

where $P = (p_{ij})$ and $Q = (q_{ij})$ are $(N + 1) \times (N + 1)$ Toeplitz matrices with small perturbations at the corners due to the boundary conditions (a detailed discussion regarding the construction of these matrices is given in [5]). Using (2.5), we may write the following approximation for (2.1)

$$P \frac{\partial \vec{v}}{\partial t} = -\lambda Q\vec{v}. \tag{2.6}$$

In order to satisfy the analytic boundary conditions, (2.3), we use the SAT methodology introduced in [4], which involves an indirect treatment of the boundary conditions. Using this method we do not satisfy the boundary conditions directly by imposing $v_0 = g(t)$, but add to the derivative operator a term proportional to the difference between the discrete value v_0 and the boundary term $g(t)$, and solve a derivative equation everywhere, including the boundary points. This approach will be discussed later.

Throughout this work we make three main assumptions:

1. Equation (2.5) is accurate to order m, i.e.,

$$P\frac{\partial \vec{u}}{\partial x} = Q\vec{u} + P\vec{T}, \tag{2.7}$$

where $\vec{T}$ is the truncation error due to the numerical differentiation and

$$\|\vec{T}\| = O(h^m). \tag{2.8}$$

This implies that we assume a solution of sufficient smoothness (see footnote 1, in the following).

2. The matrix P is a symmetric positive definite matrix with a simple structure that is easily invertible and there exist positive constants c_0, c_1 independent of N such that

$$c_0\|\vec{u}\|^2 \le (P\vec{u}, \vec{u}) \le c_1\|\vec{u}\|^2, \tag{2.9}$$

where $\|\vec{u}\|^2 = (\vec{u}, \vec{u}) = \sum_i u_i^2$ is the standard form of the scalar product, and c_1 is the largest eigenvalue of P, i.e., $c_1 = \|P\|$ because P is positive definite symmetric matrix.

3. The matrix Q is almost skew-symmetric except in $(n+1)\times(n+1)$ corners.

It is shown in [5] that the matrix $Q = (q_{ij})$ can be constructed in such a way that $n = 1$, that is,

$$\frac{Q+Q^T}{2} = \begin{pmatrix} q_{00} & \frac{1}{2}(q_{01}+q_{10}) & 0 & & & & \\ \frac{1}{2}(q_{01}+q_{10}) & q_{11} & 0 & & \mathbf{0} & & \\ 0 & 0 & 0 & & & & \\ & & & \ddots & & & \\ & & & 0 & 0 & 0 & \\ & \mathbf{0} & & 0 & q_{N-1N-1} & \frac{1}{2}(q_{NN-1}+q_{N-1N}) & \\ & & & 0 & \frac{1}{2}(q_{NN-1}+q_{N-1N}) & q_{NN} & \end{pmatrix}.$$

We now rewrite the semidiscrete problem for $\vec{v}$ in the form

$$P\frac{d\vec{v}}{dt} = -\lambda Q\vec{v} + \lambda\vec{S}_0(v_0 - g(t)), \tag{2.10}$$

where

$$\vec{S}_0 = \begin{pmatrix} \tau q_{00} \\ q_{01} + q_{10} \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{2.11}$$

THEOREM 2.1. *The approximation* (2.10), (2.11) *preserves the order of accuracy m of the spatial operator and is strictly stable under the following conditions on* τ *and the corner entries of the matrix Q*:

$$\begin{gathered} (1-\tau)q_{00} \geq 0, \qquad q_{11} \geq 0, \\ q_{NN}u_N^2 + (q_{N-1N} + q_{NN-1})u_N u_{N-1} + q_{N-1N-1}u_{N-1}^2 \geq 0, \qquad \forall u_N, u_{N-1} \in \mathbf{R}. \end{gathered} \tag{2.12}$$

Proof. Denote as before $\vec{u} = (u_0(t), \ldots, u_N(t))^T$, i.e., the values of the true solution at the grid points, and denote $\vec{v}$ its numerical approximation. Combining the accuracy condition found in assumption 1 with Eq. (2.10) we may write

$$P\frac{d\vec{u}}{dt} = -\lambda Q\vec{u} + \lambda\vec{S}_0(u_0(t) - g(t)) + P\vec{T}. \tag{2.13}$$

Note that $u_0(t) - g(t) = u(0,t) - g(t) = 0$. To get the equation for the solution error vector, $\vec{\epsilon}(t) = \vec{u}(t) - \vec{v}(t)$, we substract (2.10) from (2.13),

$$P\frac{d\vec{\epsilon}}{dt} = -\lambda Q\vec{\epsilon} + \lambda\vec{S}_0\epsilon_0 + P\vec{T}, \tag{2.14}$$

where $\epsilon_0 = v_0 - g(t) = v_0 - u_0$.

Taking the scalar product of $\vec{\epsilon}$ with (2.14) one gets

$$\frac{1}{2}\frac{d}{dt}(P\vec{\epsilon}, \vec{\epsilon}) = -\lambda(Q\vec{\epsilon}, \vec{\epsilon}) + \lambda(\vec{S}_0\epsilon_0, \vec{\epsilon}) + (P\vec{T}, \vec{\epsilon}). \tag{2.15}$$

We notice that $(Q\vec{\epsilon}, \vec{\epsilon}) = ((Q + Q^T)\vec{\epsilon}/2, \vec{\epsilon})$, which means that

$$\begin{aligned} (Q\vec{\epsilon}, \vec{\epsilon}) = {} & q_{00}\epsilon_0^2 + (q_{01} + q_{10})\epsilon_0\epsilon_1 + q_{11}\epsilon_1^2 + q_{NN}\epsilon_N^2 + (q_{N-1N} + q_{NN-1})\epsilon_{N-1}\epsilon_N \\ & + q_{N-1N-1}\epsilon_{N-1}^2. \end{aligned} \tag{2.16}$$

From (2.11) follows that

$$(\vec{S}_0\epsilon_0, \vec{\epsilon}) = \tau q_{00}\epsilon_0^2 + (q_{01} + q_{10})\epsilon_0\epsilon_1. \tag{2.17}$$

Using (2.16), (2.17) in (2.15) one gets

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}(P\vec{\epsilon}, \vec{\epsilon}) = {} & -\lambda(1-\tau)q_{00}\epsilon_0^2 - \lambda q_{11}\epsilon_1^2 - \lambda\big[q_{NN}\epsilon_N^2 + (q_{N-1N} + q_{NN-1})\epsilon_{N-1}\epsilon_N \\ & + q_{N-1N-1}\epsilon_{N-1}^2\big] + (P\vec{T}, \vec{\epsilon}). \end{aligned} \tag{2.18}$$

If we require (and manage to achieve by construction) that

$$q_{NN}\epsilon_N^2 + (q_{N-1N} + q_{NN-1})\epsilon_{N-1}\epsilon_N + q_{N-1N-1}\epsilon_{N-1}^2 \geq 0$$

for all $\epsilon_N, \epsilon_{N-1} \in \mathbf{R}$, then for $(1-\tau)q_{00} \geq 0$ and $q_{11} \geq 0$, and defining $\vec{T}_1 = 2\vec{T}$, the equation (2.18) leads to the inequality

$$\frac{d}{dt}(P\vec{\epsilon}, \vec{\epsilon}) \leq (P\vec{T}_1, \vec{\epsilon}). \tag{2.19}$$

We now use the inequality

$$(P\vec{T}_1, \vec{\epsilon}) \leq \sqrt{(P\vec{T}_1, \vec{T}_1)}\sqrt{(P\vec{\epsilon}, \vec{\epsilon})} \tag{2.20}$$

to obtain

$$2\frac{d}{dt}\sqrt{(P\vec{\epsilon}, \vec{\epsilon})} \leq \sqrt{(P\vec{T}_1, \vec{T}_1)}. \tag{2.21}$$

After integrating (2.21) and using (2.9) we get

$$\|\vec{\epsilon}\| \leq \frac{1}{2}\sqrt{\frac{c_1}{c_0}} \sup_{0\leq\tau\leq t} \|\vec{T}_1(\tau)\| t, \tag{2.22}$$

which proves the convergence of the scheme for all $t < \infty$ (and at most a linear temporal growth of the error).[1] The linear temporal bound on $\|\vec{\epsilon}\|$ given by (2.22) shows that the scheme is not only Lax stable but also strictly stable. ∎

Remarks. 1. The construction of the matrices P and Q is described in detail in [5]. We note that if we succeed in constructing the matrices P and Q we then know exactly the value of $q_{00}, q_{11}, q_{N-1N-1}, q_{N-1N} + q_{NN-1}, q_{NN}$. This implies that actually stability of the scheme (2.10), (2.11) depends only on τ. For example, for our sixth-order implicit scheme with five-order boundary closure the matrices P, Q were constructed in such a way that $q_{00} = -\frac{2}{3}$, $q_{11} = \frac{1}{6}$, $q_{N-1N-1} = \frac{1}{6}$, $q_{N-1N} + q_{NN-1} = \frac{1}{3}$, $q_{NN} = \frac{1}{3}$ and therefore the expression

$$\begin{aligned} &q_{NN}\epsilon_N^2 + (q_{N-1N} + q_{NN-1})\epsilon_{N-1}\epsilon_N + q_{N-1N-1}\epsilon_{N-1}^2 \\ &\quad = \frac{1}{3}\epsilon_N^2 + \frac{1}{3}\epsilon_{N-1}\epsilon_N + \frac{1}{6}\epsilon_{N-1}^2 = \frac{1}{6}(\epsilon_{N-1} + \epsilon_N)^2 + \frac{1}{6}\epsilon_N^2 \end{aligned}$$

is positive for all $\epsilon_N, \epsilon_{N-1} \in \mathbf{R}$ and the scheme is strictly stable for $\tau \geq 1$; see (2.18).

2. For negative λ we have the boundary condition at $x = 1$:

$$u(1, t) = g(t), \qquad t \geq 0. \tag{2.23}$$

[1] Note that the behavior of ϵ with h depends on the smoothness of the solution. To maintain the order of the approximation we need $u(x, t) \in C^m$, where m is the order of accuracy. If, for example, the initial data contain only a first derivative this will degrade the behavior of $\|\vec{T}\|$ with h.

In this case we will write the following semidiscrete approximation for $\vec{v}$,

$$\tilde{P}\frac{d\vec{v}}{dt} = -\lambda\tilde{Q}\vec{v} + \lambda\vec{S}_N(v_N - g(t)), \tag{2.24}$$

where

$$(\tilde{P})_{ij} = (P)_{N-i,N-j}, \quad (\tilde{Q})_{ij} = -(Q)_{N-i,N-j}, \quad (\vec{S}_N)_i = -(\vec{S}_0)_{N-i}, \qquad \forall 0 \le i, j \le N.$$

Because of the Toeplitz structure of matrices P and Q this means that the matrices $\tilde{P}$ and $(\tilde{Q} + \tilde{Q}^T)/2$ are almost identical to the matrices P and $(Q + Q^T)/2$. They differ only in the corners, which are transformed in such a way that the matrix $\tilde{P}$ still satisfies the conditions of assumption 2 with the same constants c_0, c_1 and the matrix $(\tilde{Q} + \tilde{Q}^T)/2$ is of the form

$$\frac{\tilde{Q} + \tilde{Q}^T}{2}$$
$$= \begin{pmatrix} -q_{NN} & -\frac{1}{2}(q_{N-1N} + q_{NN-1}) & & & \\ -\frac{1}{2}(q_{N-1N} + q_{NN-1}) & -q_{N-1N-1} & & \mathbf{0} & \\ & & \ddots & & \\ & \mathbf{0} & & -q_{11} & -\frac{1}{2}(q_{01} + q_{10}) \\ & & & -\frac{1}{2}(q_{01} + q_{10}) & -q_{00} \end{pmatrix}.$$

3. Sometimes it is useful to rewrite the approximation (2.10) in the matrix form

$$P\frac{d\vec{v}}{dt} = -\lambda\mathbf{Q}\vec{v} - \lambda\vec{S}_0 g(t), \tag{2.25}$$

where $\mathbf{Q}$ is an $(N+1) \times (N+1)$ matrix defined by

$$\mathbf{Q} = Q - S, \qquad S = \begin{pmatrix} \tau q_{00} & 0 & \cdots & 0 \\ q_{01} + q_{10} & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \mathbf{0} & \\ 0 & & & \end{pmatrix} \tag{2.26}$$

and the vector $\vec{S}_0$ is defined by (2.11).

Note that all the boundary information is incorporated into the matrix $\mathbf{Q}$ and that the time stability of the numerical scheme (2.25) depends directly on the properties of this matrix.

It should also be observed that if the inequalities (2.12) hold in the strict sense (as was achieved in the actual construction; see Remark 1) then inequality (2.22) can be sharpned. The argument is as follows: the matrix $\mathbf{Q}$ is positive definite, that is,

$$(\vec{v}, \mathbf{Q}\vec{v}) = \frac{1}{2}(\vec{v}, (\mathbf{Q} + \mathbf{Q}^T)\vec{v}) > 0, \qquad \forall \vec{v} \in \mathbf{R}^N.$$

This implies that the real part of each eigenvalue of the matrix $P^{-1}\mathbf{Q}$ is positive. One can verify this by writing

$$P^{-1}\mathbf{Q} = P^{-1/2}(P^{-1/2}\mathbf{Q}P^{-1/2})P^{1/2},$$

which means that the matrices $P^{-1}\mathbf{Q}$ and $P^{-1/2}\mathbf{Q}P^{-1/2}$ are similar and therefore have the same eigenvalues. Since P is a positive definite symmetric matrix there is no ambiguity about the meaning of $P^{1/2}$ (and therefore also the matrices $P^{1/2}$ and $P^{-1/2}$ are positive definite matrices), and the matrix $P^{-1/2}\mathbf{Q}P^{-1/2}$ satisfies

$$(\vec{v}, (P^{-1/2}\mathbf{Q}P^{-1/2})\vec{v}) = (P^{-1/2}\vec{v}, \mathbf{Q}(P^{-1/2}\vec{v})) > 0, \qquad \forall \vec{v} \in \mathbf{R}^N.$$

The last inequality implies that the real part of each eigenvalue of $P^{-1/2}\mathbf{Q}P^{-1/2}$ and $P^{-1}\mathbf{Q}$ is positive. Under these sharp inequalities, which are stricter than the ones assumed in Theorem 2.1, we can prove

THEOREM 2.2. *It is possible to show that the error norm is bounded for all t and, instead of* (2.22), *is given by*

$$\|\vec{\epsilon}\| \leq \frac{\sup_{0\leq\tau\leq t}\|\vec{T}\|}{\lambda\, d_0}(1 - e^{-\lambda\, d_0 t}),$$

where d_0 is the smallest real part of any of the eigenvalues of $P^{-1}\mathbf{Q}$ under the assumption that the matrix $P^{-1}\mathbf{Q}$ is diagonalized.

Proof. Consider Eq. (2.14) for the solution error vector in the matrix form. i.e.,

$$\frac{d\vec{\epsilon}}{dt} = -\lambda P^{-1}\mathbf{Q}\vec{\epsilon} + \vec{T}. \tag{2.27}$$

Multiplying both sides of this equation by ϵ, using the scalar product $(\cdot, \cdot)$ yields

$$\|\vec{\epsilon}\|\frac{d}{dt}\|\vec{\epsilon}\| = -\lambda(P^{-1}\mathbf{Q}\vec{\epsilon}, \vec{\epsilon}) + (\vec{T}, \vec{\epsilon}). \tag{2.28}$$

We denote now by $\{\mu_i\}_{i=0}^N$ eigenvalues of the matrix $P^{-1}\mathbf{Q}$ and by $\{\vec{e}_i\}_{i=0}^N$ the full orthonormal eigenvalue basis of $P^{-1}\mathbf{Q}$ and consider the scalar product

$$\begin{aligned}(P^{-1}\mathbf{Q}\vec{\epsilon}, \vec{\epsilon}) &= \mathcal{R}e(P^{-1}\mathbf{Q}\vec{\epsilon}, \vec{\epsilon}) = \mathcal{R}e\left(P^{-1}\mathbf{Q}\sum_{i=0}^N a_i\vec{e}_i, \sum_{i=0}^N a_i\vec{e}_i\right)\\ &= \mathcal{R}e\left(\sum_{i=0}^N a_i\mu_i\vec{e}_i, \sum_{i=0}^N a_i\vec{e}_i\right) = \mathcal{R}e\left(\sum_{i=0}^N a_i^2\mu_i\right) = \sum_{i=0}^N a_i^2\mathcal{R}e(\mu_i).\end{aligned}$$

Using the fact that all $\mathcal{R}e(\mu_i) > 0$ and denoting $d_0 = \min_{1\leq i\leq N}|\mathcal{R}e(\mu_i)|$ we get

$$(P^{-1}\mathbf{Q}\vec{\epsilon}, \vec{\epsilon}) > d_0\sum_{i=0}^N a_i^2 = d_0(\vec{\epsilon}, \vec{\epsilon}) = d_0\|\vec{\epsilon}\|^2. \tag{2.29}$$

Substituting (2.29) and the estimate $(\vec{T}, \vec{\epsilon}) \leq \|\vec{T}\|\,\|\vec{\epsilon}\|$ into (2.28) and dividing by $\|\vec{\epsilon}\|$ yields

$$\frac{d}{dt}\|\vec{\epsilon}\| \leq -\lambda d_0\|\vec{\epsilon}\| + \|\vec{T}\|. \tag{2.30}$$

From Gronwall's lemma and the fact that $\vec{\epsilon}(0) = 0$ follows that

$$\|\vec{\epsilon}\| \le \frac{\sup_{0\le\tau\le t}\|\vec{T}\|}{\lambda\, d_0}(1 - e^{-\lambda\, d_0 t}).$$

■

In the next subsection we show a graphical representation of this fact. Figures 7 and 8 show the eigenvalue spectrum of $-P^{-1}\mathbf{Q}$ for fourth-order and sixth-order approximation, respectively, for various grids. All eigenvalues of these matrices ($N = 20, 40, 60, 80$) are distinct and no eigenvalues with a positive real part exist.

4. In a similar fashion, if we define

$$\tilde{\mathbf{Q}} = \tilde{Q} - \tilde{S}, \qquad \tilde{S} = \begin{pmatrix} & & & 0 \\ & \mathbf{0} & & \vdots \\ & & & 0 \\ 0 & \cdots & 0 & -(q_{01}+q_{10}) \\ 0 & \cdots & 0 & -\tau q_{00} \end{pmatrix}, \qquad \vec{S}_N = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -(q_{01}+q_{10}) \\ -\tau q_{00} \end{pmatrix}, \tag{2.31}$$

we can rewrite the approximation (2.24) for $\lambda < 0$ as

$$\tilde{P}\frac{d\vec{v}}{dt} = -\lambda\tilde{\mathbf{Q}}\vec{v} - \lambda\vec{S}_N g(t). \tag{2.32}$$

In this case it can be shown that the matrix $\tilde{\mathbf{Q}}$ is negative definite and all eigenvalues of the matrix $\tilde{P}^{-1}\tilde{\mathbf{Q}}$ have a negative real part.

2.2. *Numerical Results*

In this subsection we consider the scalar model problem

$$u_t(x,t) + u_x(x,t) = 0, \qquad 0 \le x \le 1, t \ge 0 \tag{2.33}$$

$$u(x,0) = f(x), \qquad 0 \le x \le 1, \tag{2.34}$$

$$u(0,t) = g(t), \qquad t \ge 0 \tag{2.35}$$

with $f(x) = \sin\omega x$, $g(t) = -\sin\omega t$.

The exact solution is

$$u(x,t) = \sin\omega(x-t), \qquad 0 \le x \le 1, t \ge 0. \tag{2.36}$$

In order to highlight the difference in the quality of results obtained using standard and SAT-type boundary conditions, we solve the scalar model equation using both types of boundary conditions.

To solve the model problem (2.33), (2.34), (2.35) we use two different difference operators: fourth-order compact and six-order compact (see [5] for details). Here the order of the difference operator refers to the order of the global accuracy that the theory of Gustafsson [6, 7] predicts. There it is proved that in our case boundary conditions of at least order $m-1$ must be imposed to retain mth-order global accuracy. Therefore we use a fourth-order difference operator, which is of order three at the boundary and order four in the interior, and

TABLE I
Grid Convergence of Two High-Order Schemes on $u_t + u_x = 0$, Using Conventional Implementation of Boundary Conditions

	Fourth-order compact		Sixth-order compact	
Grid	$\log_{10}(L_2)$	Rate	$\log_{10}(L_2)$	Rate
21	−2.798		−3.510	
31	−3.431	3.60	−4.535	5.82
41	−3.901	3.76	−5.331	6.37
61	−4.580	3.86	−6.408	6.12
81	−5.069	3.91	−7.169	6.09

Note. Here $\omega = 2\pi$, CFL $= 0.1$. $T = 10$ for the fourth-order scheme and $T = 0.5$ for the sixth-order scheme.

a sixth-order difference operator of order five at the boundary and order six in the interior. The standard fourth-order Runge–Kutta method is used for time integration in the case of the fourth-order difference operator, and a sixth-order Runge–Kutta method (developed by Butcher [1, 2]) is used in case of the sixth-order difference operator. The time step is chosen small enough to ensure the local stability of the Runge–Kutta method. In the case of conventional implementation of boundary conditions we overwrite the value of the solution at the boundary point with the analytic boundary condition at the end of each Runge–Kutta stage.

Conventional boundary conditions. Table I shows a grid convergence study for both spatial discratizations. The absolute error $\log_{10}(L_2)$ at a fixed time $t = T$ and the convergence rate between two grids are plotted. The convergence rate is computed as

$$\log_{10}\left(\frac{\|u - v^{h_1}\|_2}{\|u - v^{h_2}\|_2}\right) \Big/ \log_{10}\left(\frac{h_1}{h_2}\right), \tag{2.37}$$

where $u = (u(x_0, t), u(x_2, t), \ldots, u(x_N, t))^T$ is the projection of the exact solution, v^h is the numerical solution with mesh width h, and $\|u - v^h\|_2$ is the discrete L_2 norm of the absolute error.

We see in this table that for relative short time integration ($T = 0.5$) the convergence rate of sixth-order scheme is approximately 6. The convergence rate of fourth-order scheme asymptotes to the theoretical value of 4. For the schemes to be strictly stable no eigenvalues with a positive real part are allowed to exist. Therefore we investigated numerically whether the schemes are strictly stable by both measuring the error for long time integration and computing eigenvalues of the ODE system obtained after semidiscretization. Figures 1 and 2 show the error as a function of time for the fourth-order compact scheme and the sixth-order compact scheme respectively for different grids. Clearly there is an exponential growth in time for the sixth-order scheme, but not for the fourth-order one. Figures 3 and 4 show the semidiscrete eigenvalue spectrum of the ODE system. In Fig. 3 we see that for the fourth-order scheme there are no eigenvalues with a positive real part. This fortuitous situation fails when one consider the case of a system of equations rather than the scalar partial differential equation with conventional boundary conditions (see Part II). In Fig. 4 we see that the eigenvalue spectra of the ODE system for the sixth-order scheme stretches into

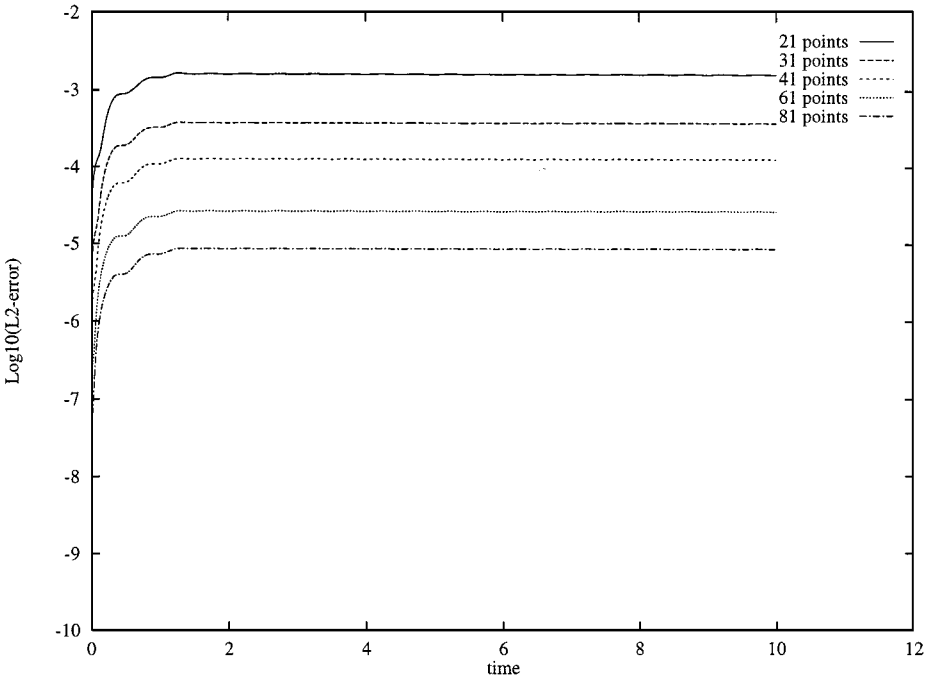

FIG. 1. The L_2-error as a function of time for the fourth-order approximation using conventional implementation of boundary conditions with CFL $= 0.1$, $\omega = 2\pi$.

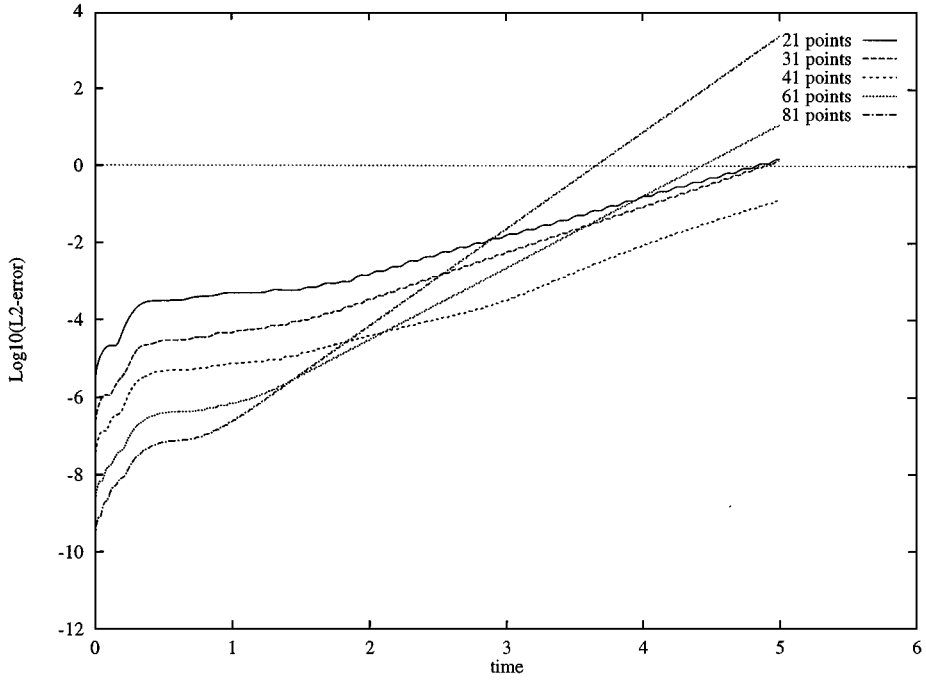

FIG. 2. The L_2-error as a function of time for the sixth-order approximation using conventional implementation of boundary conditions with CFL $= 0.1$, $\omega = 2\pi$.

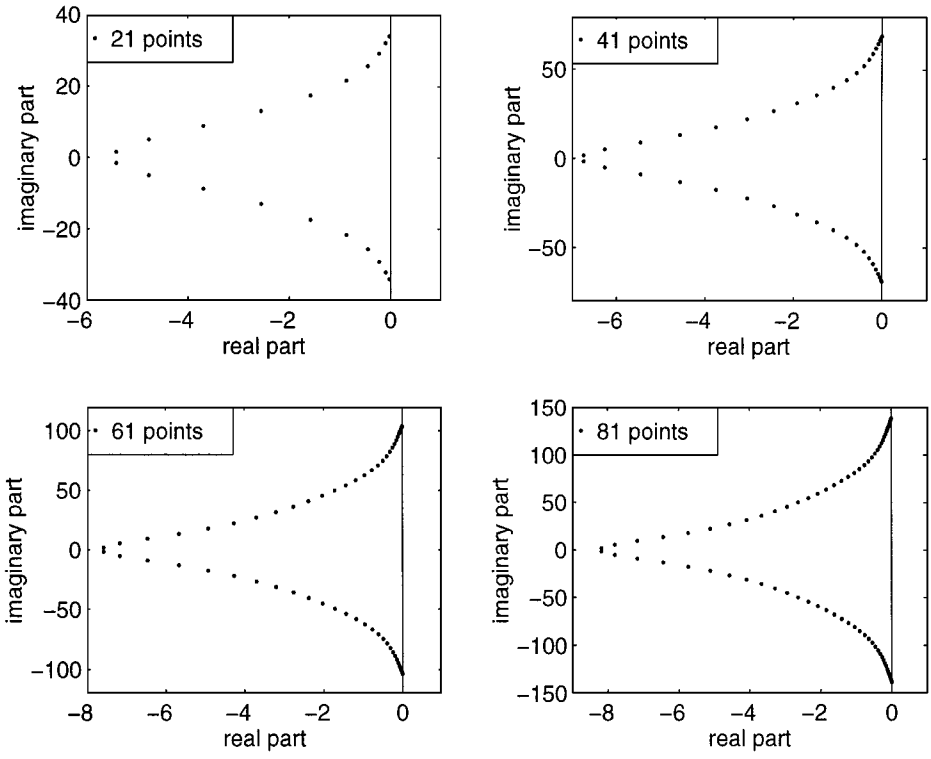

FIG. 3. Semidiscrete eigenvalue spectrum for the fourth-order approximation using conventional implementation of boundary conditions.

the right half-plane and since the exponential growth is caused by the eigenvalues having a positive real part we get the unwanted growth. The time divergence seen in the sixth-order scheme is a result of imposing the conventional boundary conditions.

SAT boundary conditions. We now solve the model problem (2.33), (2.34), (2.35) using the SAT method for treating the boundary conditions.

Table II shows a grid refinement study for the fourth-order and the sixth-order compact difference operators with different SAT parameters τ. As in the case of conventional

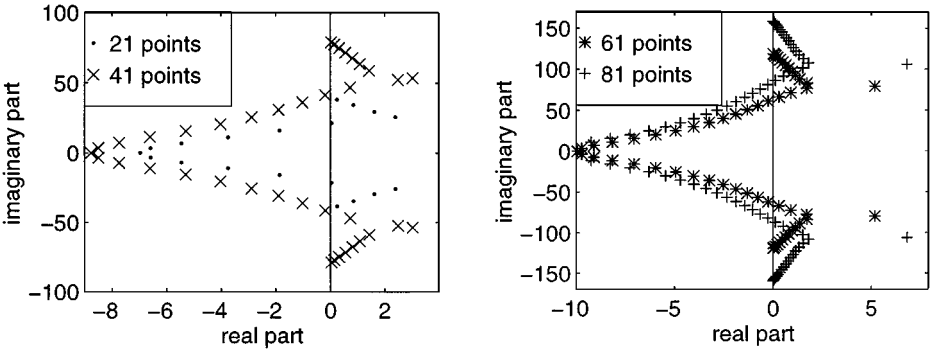

FIG. 4. Magnification of semidiscrete eigenvalue spectrum close to imaginary axis for the sixth-order approximation using conventional implementation of boundary conditions.

TABLE II
Grid Convergence of Two High-Order Schemes on $u_t + u_x = 0$, Using SAT Implementation of Boundary Conditions with the SAT Parameter $\tau = 2$

	Fourth-order compact		Sixth-order compact	
Grid	$\log_{10}(L_2)$	Rate	$\log_{10}(L_2)$	Rate
21	−3.632		−5.012	
31	−4.315	3.99	−6.203	6.75
41	−4.816	4.00	−7.044	6.73
61	−5.541	4.12	−8.170	6.39
81	−6.061	4.16	−8.949	6.23

Note. Here $\omega = 2\pi$, CFL = 0.1, T = 10.

boundary conditions we plot the absolute error $\log_{10}(L_2)$ at the time $t = T = 10$ (extracted from computation run to $T = 100$) and the convergence rate computed as in (2.37). We see that the SAT procedure for boundary treatment does not destroy the formal accuracy of spatial discretization. The numerical results agree well with the theory of Gustafsson [6, 7] and give the predicted accuracy. Numerical experimentation has shown that the choice of $\tau = 2$ is efficacious.

It was proved in Section 2.1 that semidiscrete approximation (2.10) and (2.11) obtained with the SAT method is strictly stable. From results of Kreiss and Wu [10] and Levi and Tadmor [11] follow that the fully discrete approximation is stable if a locally stable Runge–Kutta method is used for time integration. Again, the standard fourth-order Runge–Kutta method is used for time integration in the case of the fourth-order spatial difference operator, and the sixth-order Runge–Kutta method (developed by Butcher [1, 2]) is used in case of the sixth-order spatial difference operator. The time step is chosen small enough to ensure the local stability of the Runge–Kutta method. Figures 5 and 6 show the error as a function of time for fourth- and sixth-order schemes, respectively, with different SAT parameters τ, different grids, and different CFL numbers. In all cases the error remained bounded for all grids and CFLs for time as large as $T = 100$. No exponential growth was found for the SAT method, indicating time (strict) stability. Figures 7 and 8 show semi-discrete eigenvalues spectrum for this method, i.e., the eigenvalues of the matrix $-P^{-1}\mathbf{Q}$ defined by (2.25), (2.26) (see Section 1.1). As we can see in these figures no eigenvalues with positive real part exist.

We also solved the problem (2.33), (2.34), (2.35) for different values of ω. In Figs. 9 and 10 we show the approximate solution of the problem computed at the time $t = 10$ using the sixth-order compact scheme with $\tau = 2$, CFL $= 0.1$, $\omega = 30\pi$ and the number of grid points $N = 80$.

3. 2-D HYPERBOLIC EQUATIONS

3.1. *Description of the Method and Proof of Main Results*

In this section we show how to use the one-dimensional scheme, whose properties were described in the previous section, for the two-dimensional case. We consider the following linear differential equation, with constant coefficients, in a rectangular domain Ω with

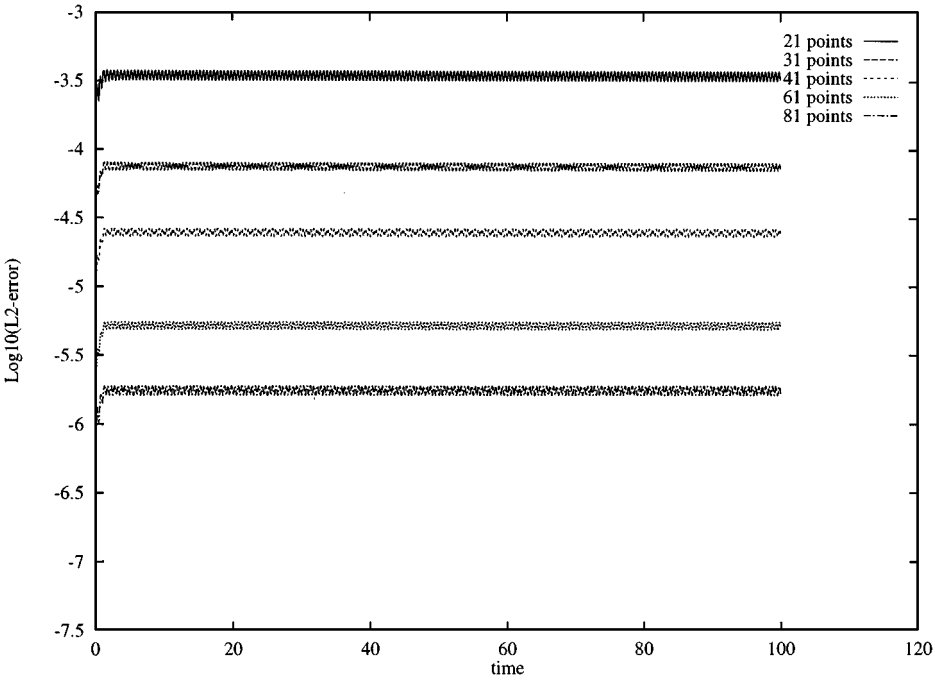

FIG. 5. The L_2-error as a function of time for the fourth-order approximation using SAT implementation of boundary conditions with $\tau = 1$, CFL $= 0.5$, $\omega = 2\pi$.

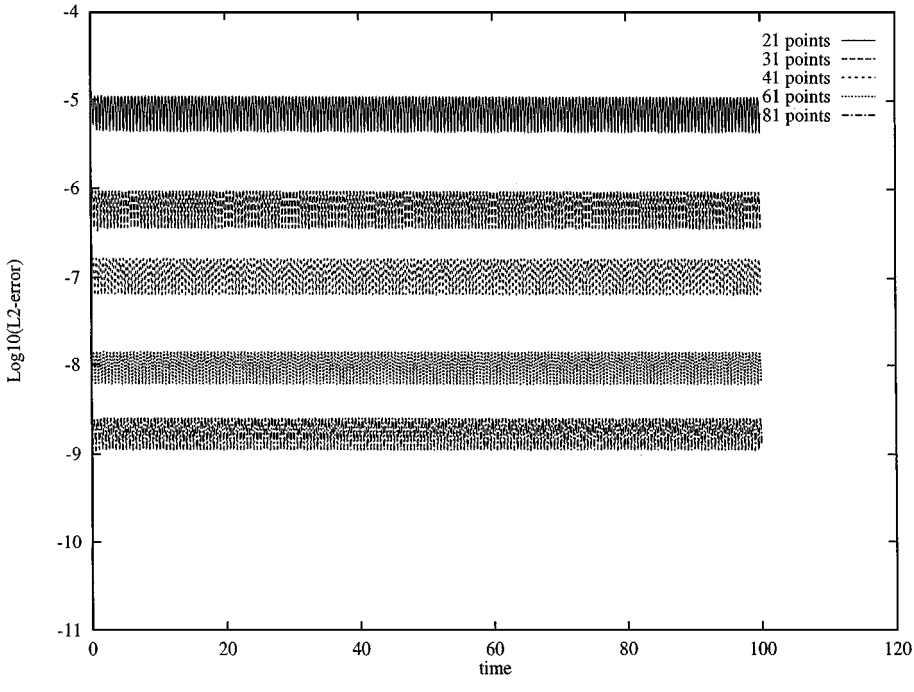

FIG. 6. The L_2-error as a function of time for the sixth-order approximation using SAT implementation of boundary conditions with $\tau = 2$, CFL $= 0.1$, $\omega = 2\pi$.

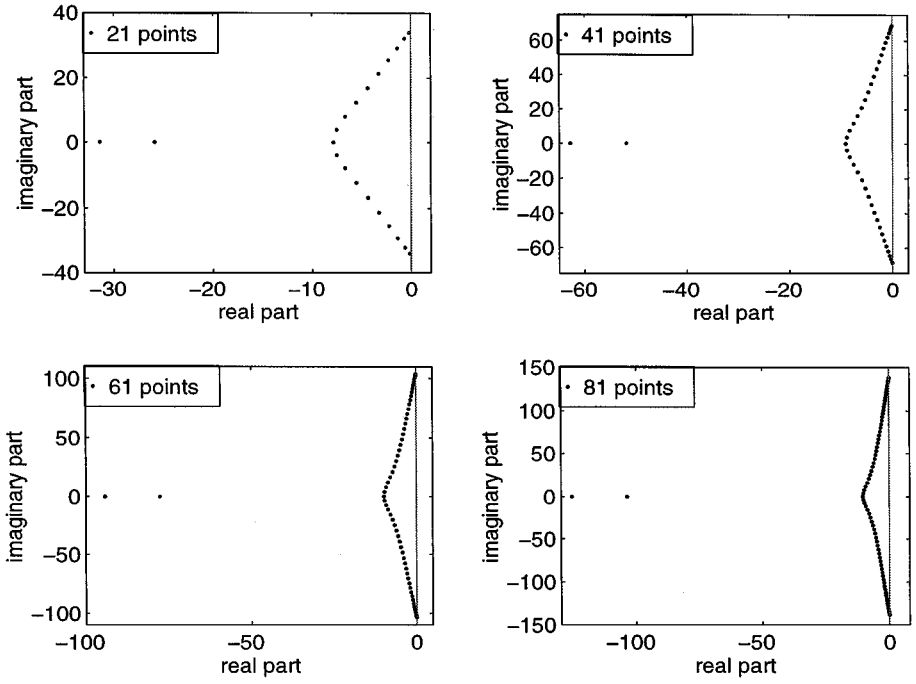

FIG. 7. Semidiscrete eigenvalue spectrum for the fourth-order approximation using SAT implementation of boundary conditions with $\tau = 2$.

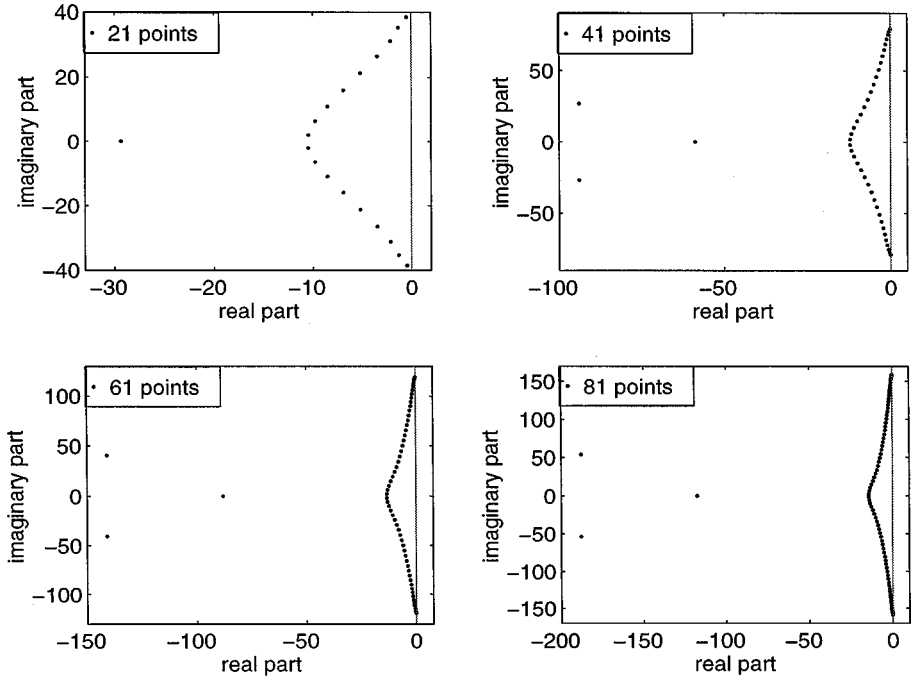

FIG. 8. Semidiscrete eigenvalue spectrum for the sixth-order approximation using SAT implementation of boundary conditions with $\tau = 2$.

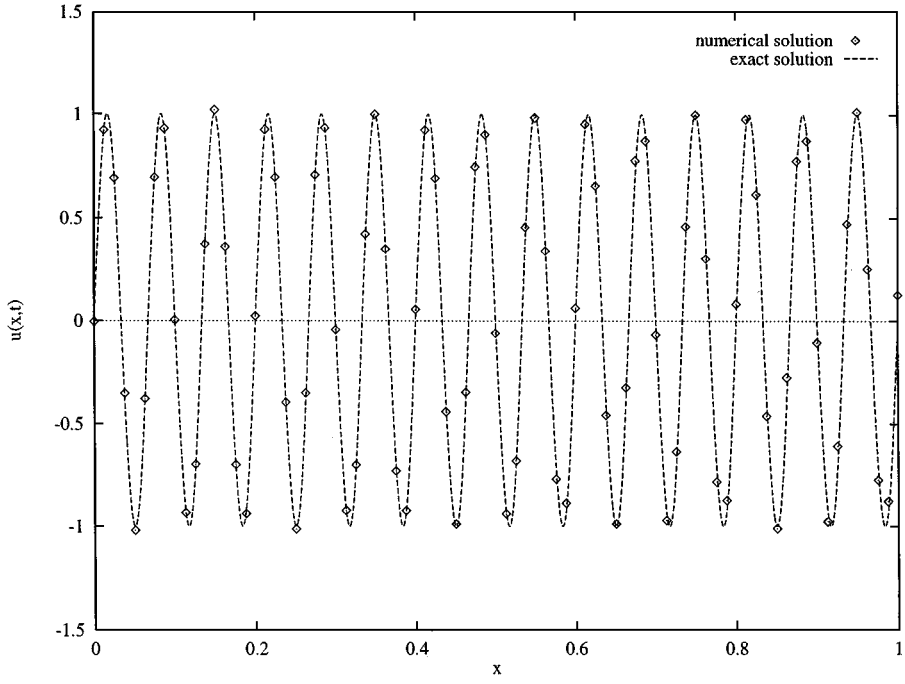

FIG. 9. The numerical solution at the time $t = 10$ obtained with the sixth-order scheme using SAT implementation of boundary conditions with $\tau = 2$, CFL $= 0.1$, $\omega = 30\pi$, $N = 81$.

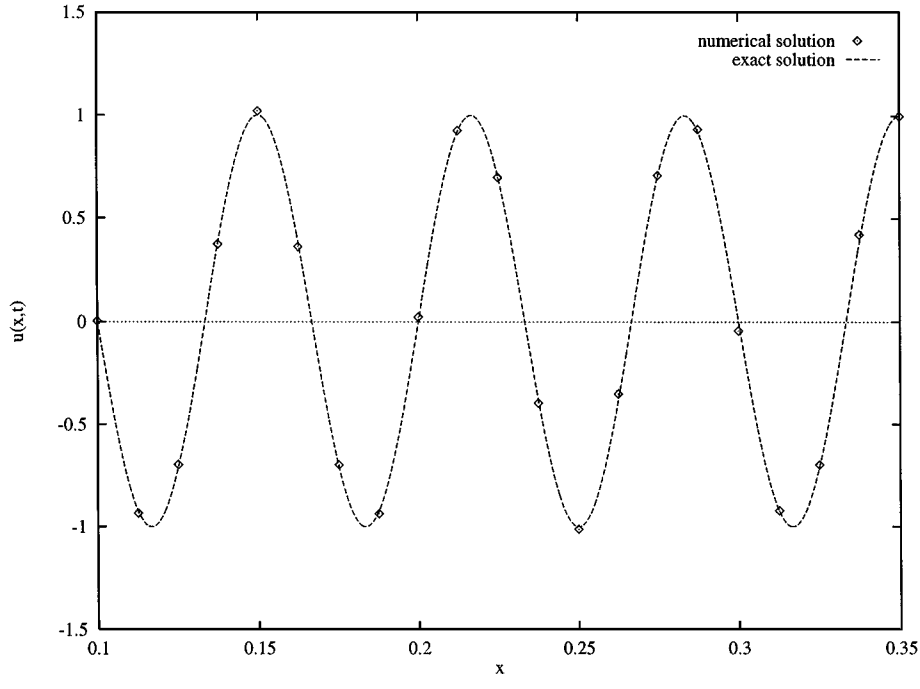

FIG. 10. Magnification of the numerical solution at the time $t = 10$ obtained with the sixth-order scheme using SAT implementation of boundary conditions with $\tau = 2$, CFL $= 0.1$, $\omega = 30\pi$, $N = 81$.

boundary curve $\partial\Omega$,

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} + b\frac{\partial u}{\partial y} = 0, \qquad x, y \in \Omega, t \geq 0, \tag{3.1}$$

with initial condition prescribed at $t = 0$,

$$u(x, y, 0) = f(x, y), \qquad x, y \in \Omega, \tag{3.2}$$

and the boundary condition

$$u(x, y, t)|_{\partial\Omega} = g_B(t), \qquad t \geq 0. \tag{3.3}$$

Without loss of generality we assume that Ω is a square

$$\Omega = \{(x, y) \in \mathbf{R}^2 \,|\, 0 \leq x \leq 1, 0 \leq y \leq 1\},$$

and if, for example, $a > 0$, $b < 0$ then we have the boundary conditions

$$u(0, y, t) = g^{(1)}(y, t) \tag{3.4}$$

$$u(x, 1, t) = g^{(2)}(x, t), \qquad t \geq 0. \tag{3.5}$$

We begin by dividing the continuous domain Ω into N^2 uniform intervals of width h, where $h = \Delta x = \Delta y = 1/N$. We use for $i = 0, \ldots, N$ and $j = 0, \ldots, N$ the notation

$$x_i = ih, \qquad y_j = jh, \qquad u_{ij}(t) = u(x_i, y_j, t), \tag{3.6}$$

where $u_{ij}(t)$ is the projection of the exact solution $u(x, y, t)$ unto the grid. We arrange the solution projection array in vectors according to rows, starting from the bottom of the domain Ω, and denote

$$\begin{aligned}\vec{U}(t) &= (u_{00}, u_{10}, \ldots, u_{N0}; \ldots; u_{0k}, u_{1k}, \ldots, u_{Nk}; \ldots; u_{0N}, u_{1N}, \ldots, u_{NN})^T \\ &= (\vec{u}_0, \ldots, \vec{u}_k, \ldots, \vec{u}_N)^T.\end{aligned} \tag{3.7}$$

If we arrange this array by columns (instead of rows) we will have the structure

$$\begin{aligned}\vec{U}^c(t) &= (u_{00}, u_{01}, \ldots, u_{0N}; \ldots; u_{k0}, u_{k1}, \ldots, u_{kN}; \ldots; u_{N0}, u_{N1}, \ldots, u_{NN})^T \\ &= (\vec{u}^c_0, \ldots, \vec{u}^c_k, \ldots, \vec{u}^c_N)^T.\end{aligned} \tag{3.8}$$

As one can see, the vector $\vec{U}^c(t)$ is a specific permutation of $\vec{U}(t)$,

$$\vec{U}^c(t) = R\vec{U}(t), \tag{3.9}$$

where $R = R^T = R^{-1}$ is an $(N+1)^2 \times (N+1)^2$ orthogonal matrix, each row of which contains $(N+1)^2 - 1$ zeros and a single one somewhere. If the domain is not a square, then $R \neq R^T$, but still $RR^T = I$.

The continuous derivative $\partial\vec{u}_k/\partial x$ ($k = 0, \ldots, N$) is then replaced with a finite-difference representation

$$P\frac{\partial \vec{u}_k}{\partial x} = Q\vec{u}_k + P\vec{T}^{(x)}_k, \tag{3.10}$$

and the continuous derivative $\partial \vec{u}_k/\partial y$ $(k=0,\ldots,N)$ is replaced with

$$\tilde{P}\frac{\partial \vec{u}_k^c}{\partial y} = \tilde{Q}\vec{u}_k^c + \tilde{P}\vec{T}_k^{(y)}, \tag{3.11}$$

where P, $\tilde{P}$ and Q, $\tilde{Q}$ are $(N+1)\times(N+1)$ matrices which have exactly the same structure as in the one-dimensional case, and vectors $\vec{T}_k^{(x)}$, $\vec{T}_k^{(y)}$ are the truncation error due to the numerical differentiation. Recall that the superscript "$\sim$" is used when the "inflow" boundary is on the right side of the one-dimensional domain.

Using (3.9), (3.10), and (3.11) we can write

$$\begin{aligned}\left(a\frac{\partial}{\partial x}+b\frac{\partial}{\partial y}\right)u_{ij}(t) &= [aD\vec{U}+b\tilde{D}\vec{U}^c+\vec{T}^{(x)}+\vec{T}^{(y)}]_{ij}\\ &= [aD\vec{U}+bR\tilde{D}R\vec{U}+\vec{T}^{(x)}+R\vec{T}^{(x)}]_{ij},\end{aligned} \tag{3.12}$$

where D and $\tilde{D}$ are the $(N+1)^2\times(N+1)^2$ block-diagonal matrices

$$D=\begin{pmatrix} P^{-1}Q & & & \\ & P^{-1}Q & & \\ & & \ddots & \\ & & & P^{-1}Q\end{pmatrix}, \qquad \tilde{D}=\begin{pmatrix} \tilde{P}^{-1}\tilde{Q} & & & \\ & \tilde{P}^{-1}\tilde{Q} & & \\ & & \ddots & \\ & & & \tilde{P}^{-1}\tilde{Q}\end{pmatrix}, \tag{3.13}$$

and $\vec{T}^{(x)}=(\vec{T}_1^{(x)},\ldots,\vec{T}_N^{(x)})^T$ and $\vec{T}^{(y)}=(\vec{T}_1^{(y)},\ldots,\vec{T}_N^{(y)})^T$ are truncation errors.

We now define the $(N+1)^2$ vectors, $\vec{V}=(\vec{v}_0,\ldots,\vec{v}_k,\ldots,\vec{v}_N)^T$ and $\vec{V}^c=(\vec{v}_0^c,\ldots,\vec{v}_k^c,\ldots,\vec{v}_N^c)^T$, where $\vec{v}_k$ and $\vec{v}_k^c$ are the numerical approximation to the projection $\vec{u}_k$ and $\vec{u}_k^c$ $(k=0,\ldots,N)$, respectively, and write the semidiscrete problem in the following way,

$$\frac{d\vec{V}}{dt} = -[a\mathbf{D}+bR\tilde{\mathbf{D}}R]\vec{V} - a\vec{G}^{(x)} - bR\vec{G}^{(y)}, \tag{3.14}$$

where

$$\mathbf{D}=\begin{pmatrix} P^{-1}\mathbf{Q} & & & \\ & P^{-1}\mathbf{Q} & & \\ & & \ddots & \\ & & & P^{-1}\mathbf{Q}\end{pmatrix}, \qquad \tilde{\mathbf{D}}=\begin{pmatrix} \tilde{P}^{-1}\tilde{\mathbf{Q}} & & & \\ & \tilde{P}^{-1}\tilde{\mathbf{Q}} & & \\ & & \ddots & \\ & & & \tilde{P}^{-1}\tilde{\mathbf{Q}}\end{pmatrix},$$

$$\vec{G}^{(x)}=\begin{pmatrix} P^{-1}\vec{S}_0 g_0^{(1)}(t)\\ P^{-1}\vec{S}_0 g_1^{(1)}(t)\\ \vdots \\ P^{-1}\vec{S}_0 g_N^{(1)}(t)\end{pmatrix}, \qquad \vec{G}^{(y)}=\begin{pmatrix} \tilde{P}^{-1}\vec{S}_N g_0^{(2)}(t)\\ \tilde{P}^{-1}\vec{S}_N g_1^{(2)}(t)\\ \vdots \\ \tilde{P}^{-1}\vec{S}_N g_N^{(2)}(t)\end{pmatrix}, \tag{3.15}$$

and the martices $\mathbf{Q}$, $\tilde{\mathbf{Q}}$, S, $\tilde{S}$ and the vectors $\vec{S}_0$, S_N are the same as in the 1-D case; see Section 2.

Since $S\vec{u}_k - \vec{S}_0 g_k^{(1)}(t) = 0$ and also $S\vec{u}_k^c - \vec{S}_N g_k^{(2)}(t) = 0$ $(k = 0, \ldots, N)$, we may write for the vector $\vec{U}$

$$\frac{d\vec{U}}{dt} = [-a\mathbf{D} - bR\tilde{\mathbf{D}}R]\vec{U} - a\vec{G}^{(x)} - bR\vec{G}^{(y)} + \vec{T}^{(x)} + R\vec{T}^{(y)}. \tag{3.16}$$

Subtracting (3.14) from (3.16) we get

$$\frac{d\vec{E}}{dt} = [-a\mathbf{D} - bR\tilde{\mathbf{D}}R]\vec{E} + \vec{\mathbf{T}}, \tag{3.17}$$

where $\vec{E} = \vec{U} - \vec{V}$ is the two-dimensional array of the errors arranged by rows as a vector and $\vec{\mathbf{T}}$ is proportional to the truncation error.

We recall that in Section 2.1 it was proven that if the inequalities (2.12) hold then the real part of each eigenvalue of the matrix $P^{-1}\mathbf{Q}$ is positive and the real part of each eigenvalue of the matrix $\tilde{P}^{-1}\tilde{\mathbf{Q}}$ is negative. Therefore all eigenvalues of $\mathbf{D}$ have a positive real part and all eigenvalues of $R\tilde{\mathbf{D}}R$ have a negative real part. To prove the time stability of the scheme (3.14) it is sufficient to show that $H(-a\mathbf{D} - bR\tilde{\mathbf{D}}R) + [H(-a\mathbf{D} - bR\tilde{\mathbf{D}}R)]^T \le 0$ for any symmetric positive definite matrix H.

To show this, we define now a symmetric positive definite matrix, $H = P^{1/2}(R\tilde{P}R)P^{1/2}$, and consider the scalar product

$$\begin{aligned}&([H(-a\mathbf{D} - bR\tilde{\mathbf{D}}R) + (-a\mathbf{D} - bR\tilde{\mathbf{D}}R)^T H]\vec{E}, \vec{E})\\ &\quad = -a([H\mathbf{D} + \mathbf{D}^T H]\vec{E}, \vec{E}) - b([HR\tilde{\mathbf{D}}R + R\tilde{\mathbf{D}}^T RH]\vec{E}, \vec{E}).\end{aligned} \tag{3.18}$$

It can be verified by direct multiplication and by using the properties of block-diagonal matrices and of the permutation matrix R that any block-diagonal matrix M is commutative with the matrix of the form $R\tilde{\mathbf{D}}R$, i.e., for example, $MR\tilde{\mathbf{D}}R = R\tilde{\mathbf{D}}RM$. Using this information, and the fact that $RR = I$, we can write

$$\begin{aligned}H\mathbf{D} + \mathbf{D}^T H &= P^{1/2}(R\tilde{P}R)P^{1/2}P^{-1}\mathbf{Q} + \mathbf{Q}^T P^{-1}P^{1/2}(R\tilde{P}R)P^{1/2}\\ &= (R\tilde{P}R)\mathbf{Q} + (R\tilde{P}R)\mathbf{Q}^T = R\tilde{P}R(\mathbf{Q} + \mathbf{Q}^T),\\ HR\tilde{\mathbf{D}}R + R\tilde{\mathbf{D}}^T RH &= P^{1/2}(R\tilde{P}R)P^{1/2}R\tilde{P}^{-1}\tilde{\mathbf{Q}}R + R\tilde{\mathbf{Q}}^T\tilde{P}^{-1}RP^{1/2}(R\tilde{P}R)P^{1/2}\\ &= PR\tilde{\mathbf{Q}}R + PR\tilde{\mathbf{Q}}^T R = PR(\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)R.\end{aligned} \tag{3.19}$$

Denoting $\vec{\varphi} = (R\tilde{P}^{1/2}R)\vec{E}$ and using again the fact that for any block-diagonal matrix M $(R\tilde{P}^{1/2}R)M = M(R\tilde{P}^{1/2}R)$ we obtain

$$\begin{aligned}(R\tilde{P}R(\mathbf{Q} + \mathbf{Q}^T)\vec{E}, \vec{E}) &= (R\tilde{P}^{1/2}RR\tilde{P}^{1/2}R(\mathbf{Q} + \mathbf{Q}^T)\vec{E}, \vec{E})\\ &= (R\tilde{P}^{1/2}R(\mathbf{Q} + \mathbf{Q}^T)\vec{E}, R\tilde{P}^{1/2}R\vec{E})\\ &= ((\mathbf{Q} + \mathbf{Q}^T)\vec{\varphi}, \vec{\varphi}).\end{aligned} \tag{3.20}$$

In a similar fashion, denoting $\vec{\eta} = (RP^{1/2})\vec{E}$ we get

$$\begin{aligned}(PR(\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)R\vec{E}, \vec{E}) &= (P^{1/2}R(\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)R\vec{E}, P^{1/2}\vec{E})\\ &= (R(\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)RP^{1/2}\vec{E}, P^{1/2}\vec{E})\\ &= ((\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)\vec{\eta}, \vec{\eta}).\end{aligned} \tag{3.21}$$

Taking into account (3.19), (3.20), (3.21), the fact that $a > 0$, $b < 0$, and

$$((\mathbf{Q} + \mathbf{Q}^T)\vec{\varphi}, \vec{\varphi}) \geq 0, \quad ((\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)\vec{\eta}, \vec{\eta}) \leq 0, \qquad \forall \vec{\varphi}, \vec{\eta} \in \mathbf{R}^{(N+1)^2}$$

we can conclude that if the one-dimensional inequalities (2.12) hold, then

$$\begin{aligned}&([H(-a\mathbf{D} - bR\tilde{\mathbf{D}}R) + (-a\mathbf{D} - bR\tilde{\mathbf{D}}R)^T H]\vec{E}, \vec{E})\\ &= -a((\mathbf{Q} + \mathbf{Q}^T)\vec{\varphi}, \vec{\varphi}) - b((\tilde{\mathbf{Q}} + \tilde{\mathbf{Q}}^T)\vec{\eta}, \vec{\eta}) \leq 0\end{aligned}$$

for all $\vec{E} \in \mathbf{R}^{(N+1)^2}$.

Remark. While the given form of (3.14) is advantageous for the proof, in practice we used the form

$$\frac{d}{dt}[V] = -\big[a\mathbf{D}[V] + b[V]\tilde{\mathbf{D}}^T + a\vec{G}^{(1)}(t)\vec{S}_0^T P^{-1} + b\tilde{P}^{-1}\vec{S}_N(\vec{G}^{(2)}(t))^T\big],$$

where $[V]$ is a $(N+1) \times (N+1)$ matrix with the elements v_{ij} and

$$\vec{G}^{(1)}(t) = \begin{pmatrix} g_0^{(1)}(t) \\ g_1^{(1)}(t) \\ \vdots \\ g_N^{(1)}(t) \end{pmatrix}, \qquad \vec{G}^{(2)}(t) = \begin{pmatrix} g_0^{(2)}(t) \\ g_1^{(2)}(t) \\ \vdots \\ g_N^{(2)}(t) \end{pmatrix},$$

and the matrices $\mathbf{D}$, $\tilde{\mathbf{D}}$ and P, $\tilde{P}$ and the vectors $\vec{S}_0$, S_N were defined earlier. This means that in practice the one-dimensional algorithm was implemented on each row to compute the numerical approximation to u_x and on each column to compute the numerical approximation to u_y. Note that P^{-1}, $\tilde{P}^{-1}$ are never evaluated. Rather the decompositions $P = LU$ and $\tilde{P} = \tilde{L}\tilde{U}$ are calculated. L and U ($\tilde{L}$ and $\tilde{U}$) are bidiagonal matrices with one of them having "ones" along the diagonal. Hence, the inversion of L and U ($\tilde{L}$ and $\tilde{U}$) is very cheap.

3.2. *Numerical Results*

Here we consider the problem

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0, \qquad 0 \leq x \leq 1, 0 \leq y \leq 1, t \geq 0, \tag{3.22}$$

$$u(0, y, t) = \sin\omega(y - 2t), \tag{3.23}$$

$$u(x, 0, t) = \sin\omega(x - 2t), \tag{3.24}$$

$$u(x, y, 0) = \sin\omega(x + y), \tag{3.25}$$

The analytic solution of this problem is

$$u(x, y, t) = \sin\omega(x + y - 2t). \tag{3.26}$$

We shall now use the SAT method, as well as the conventional implementation of boundary conditions, to solve the problem (3.22)–(3.25). Two difference operators were used: fourth-order with third-order boundary closure and sixth-order with fifth-order boundary closure.

TABLE III

Grid Convergence of Two High-Order Schemes on $u_t + u_x + u_y = 0$, Using Conventional Implementation of Boundary Conditions with CFL = 0.1 and $T = 0.4$ for the Sixth-Order Scheme, and CFL = 0.25 and $T = 10$ for the Fourth-Order Scheme ($\omega = 2\pi$)

	Fourth-order compact		Sixth-order compact	
Grid	$\log_{10}(L_2)$	Rate	$\log_{10}(L_2)$	Rate
21	−2.786		−3.536	
31	−3.461	3.83	−4.619	6.15
41	−3.947	3.89	−5.378	6.07
61	−4.638	3.92	−6.420	5.92
81	−5.131	3.95	−7.143	5.83

The temporal discretization was accomplished with the standard fourth-order Runge–Kutta algorithm in the case of fourth-order difference operator and with a sixth-order Runge–Kutta algorithm developed by Butcher [1, 2] in the case of sixth-order difference operator. In the case of conventional implementation of boundary conditions the value of the solution at the boundary point was overridden with the analytic boundary condition at the end of each Runge–Kutta stage.

Conventional boundary conditions. To check on the order of accuracy, the runs were repeated for $\Delta x = \Delta y = 1/20, 1/30, 1/40, 1/60$, and $1/80$. Doubling the grid at constant CFL should decrease the error at time $t = T$ by a factor $(\frac{1}{2})^p$, where $p = 4, 6$ is order of the method. The formal accuracy of each scheme was determined in this manner. Table III shows the results of this study. The $\log_{10}$ of the L_2 error at time $t = T$ and the convergence rate are the entries. $T = 10$ in the case of the fourth-order scheme and $T = 0.4$ in the case of the sixth-order scheme. As one can see for relative short time integration the convergence rate of the sixth-order scheme is approximately 6 and the convergence rate of the fourth-order scheme asymptotes to the theoretical value of 4.

The error as a function of time for the fourth-order and the sixth-order schemes is shown in Figs. 11 and 12, respectively, for different grids. CFL = 0.5 was chosen for the fourth-order

TABLE IV

Grid Convergence of Two High-Order Schemes on $u_t + u_x + u_y = 0$, Using SAT Implementation of Boundary Conditions with the SAT Parameter $\tau = 2$ and CFL = 0.1 for the Sixth-Order Scheme and the SAT Parameter $\tau = 1$ and CFL = 0.25 for the Fourth-Order Scheme ($T = 10$, $\omega = 2\pi$)

	Fourth-order compact		Sixth-order compact	
Grid	$\log_{10}(L_2)$	Rate	$\log_{10}(L_2)$	Rate
21	−3.389		−4.909	
31	−4.100	4.04	−5.991	6.14
41	−4.599	4.00	−6.757	6.14
61	−5.310	4.04	−7.835	6.06
81	−5.813	4.03	−8.575	6.00

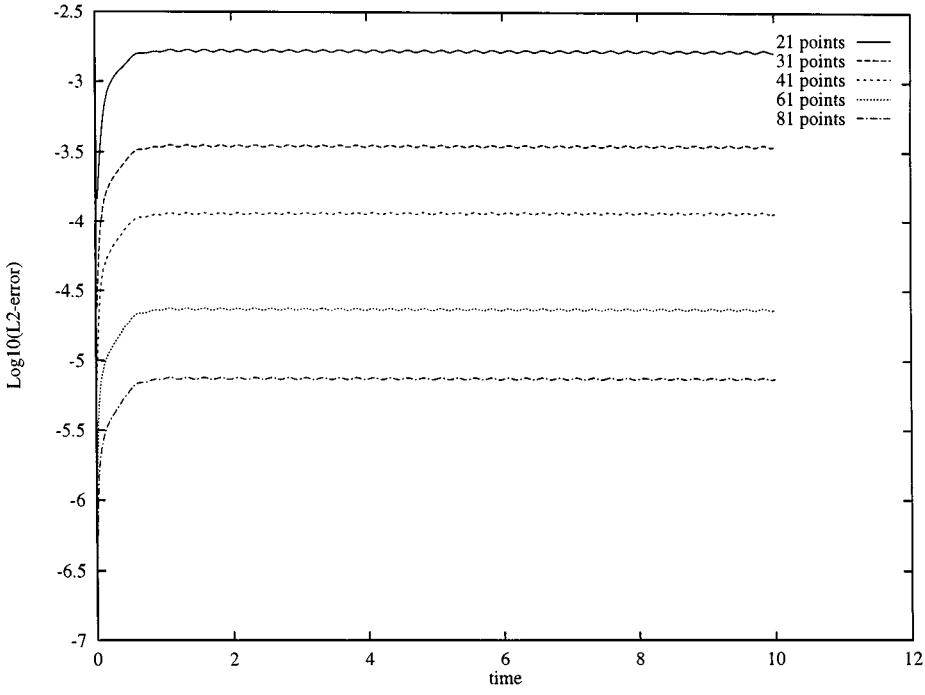

FIG. 11. The L_2-error as a function of time for the fourth-order approximation using conventional implementation of boundary conditions with CFL $= 0.25$, $\omega = 2\pi$.

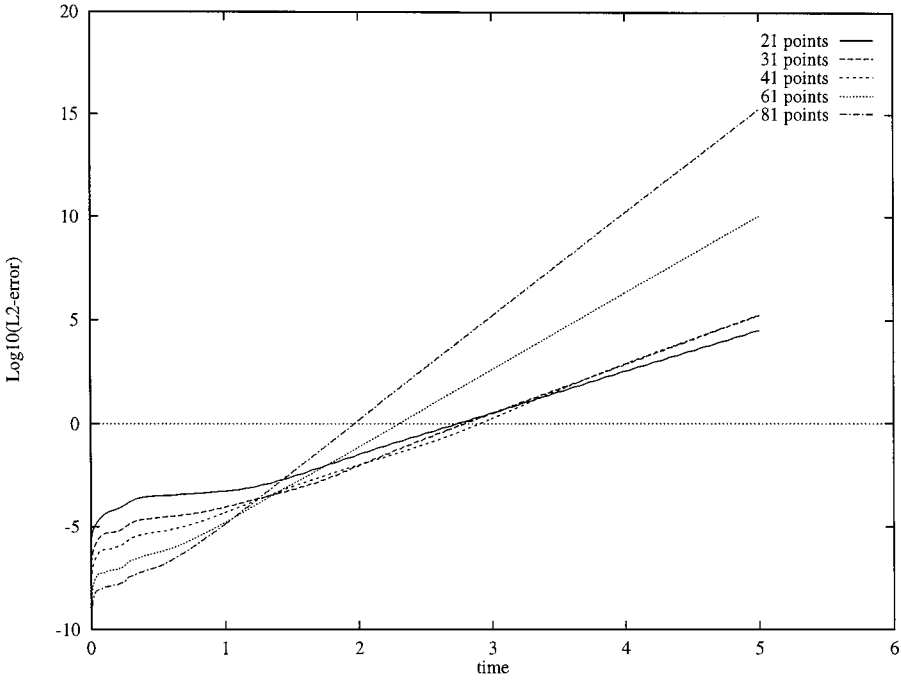

FIG. 12. The L_2-error as a function of time for the sixth-order approximation using conventional implementation of boundary conditions with CFL $= 0.1$, $\omega = 2\pi$.

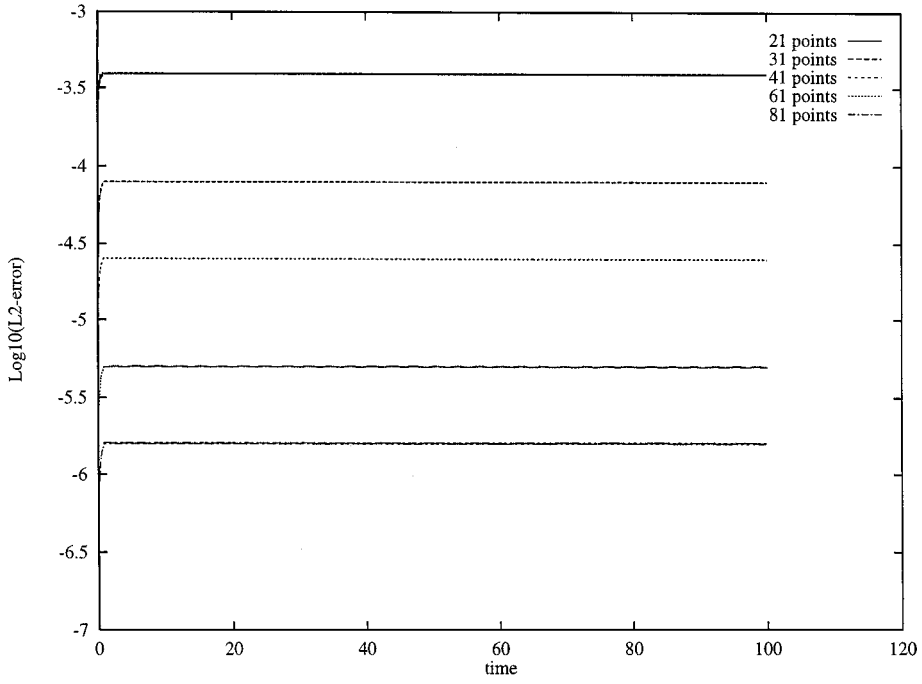

FIG. 13. The L_2-error as a function of time for the fourth-order approximation using SAT implementation of boundary conditions with CFL $= 0.25$, $\tau = 1$, $\omega = 2\pi$.

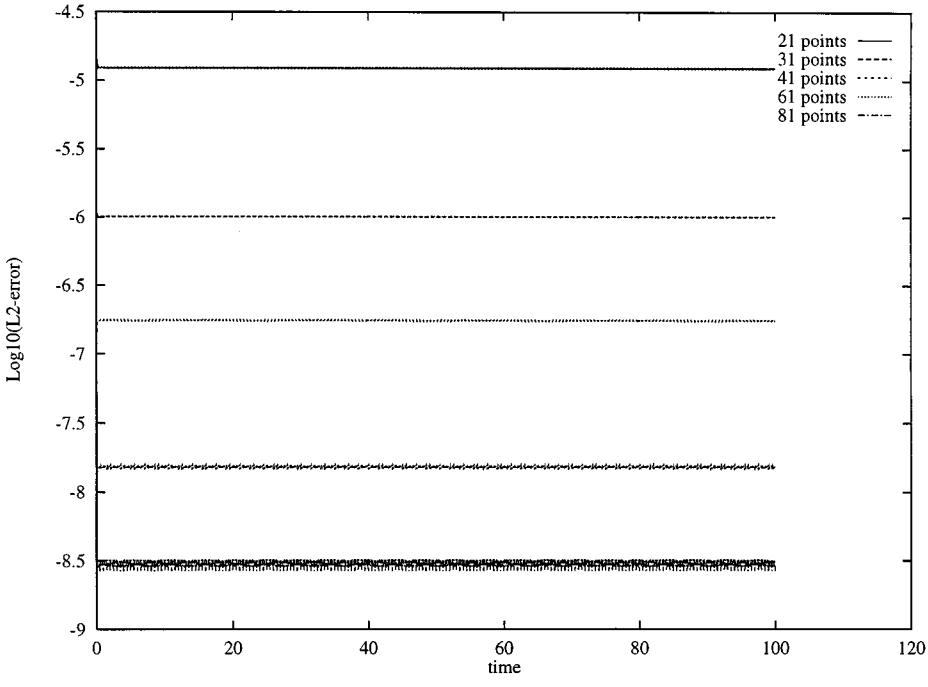

FIG. 14. The L_2-error as a function of time for the sixth-order approximation using SAT implementation of boundary conditions with CFL $= 0.1$, $\tau = 2$, $\omega = 2\pi$.

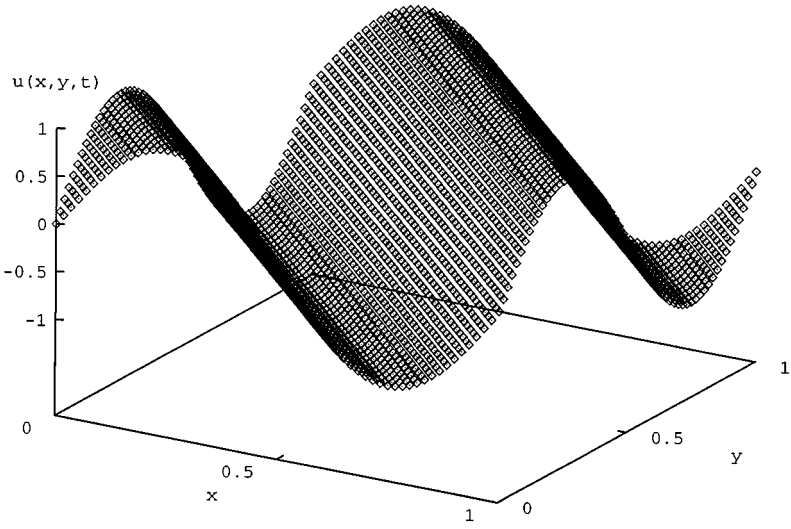

FIG. 15. Numerical solution at the time $T = 2$ obtained with the sixth-order approximation using SAT implementation of boundary conditions with $N = 60$, CFL $= 0.1$, $\tau = 2$, $\omega = 2\pi$.

scheme and CFL $= 0.1$ was chosen for the sixth-order scheme. As in the one-dimensional case, the runs are time stable in the case of the fourth-order scheme. The results obtained by using the sixth-order scheme diverge exponentially from the analytic solution.

SAT boundary conditions. To check on the order of accuracy, the runs were repeated for $\Delta x = \Delta y = 1/20, 1/30, 1/40, 1/60$, and $1/80$. Table IV shows a grid convergence study for both spatial operators. The absolute error $\log_{10}(L_2)$ at a fixed time $T = 10$ and the convergence rate are plotted. As one can see, the formal accuracy of the spatial operator is unaffected by SAT boundary treatment.

The simulations were all run to equivalent times $T = 100$ for both the fourth- and the sixth-order schemes and different grids. CFL $= 0.25$, $\tau = 1$ were chosen for the fourth-order scheme and CFL $= 0.1$, $\tau = 2$ were chosen for the sixth-order scheme. Figures 13 and 14 show a plot of the error of the solution to the problem (3.22)–(3.25) for the fourth-order and the sixth-order, respectively. The $\log_{10}$ of the L_2 error is plotted as a function of time for five grid densities: 21, 31, 41, 61, and 81 points, respectively. It is clear that both schemes give good results. No exponential growth exists, indicating time stability of the schemes.

Figure 15 shows the 3-D plot of the numerical solution at the time $T = 2$ obtained using the sixth-order scheme with $N = 60$, CFL $= 0.1$, $\tau = 2$, $\omega = 2\pi$.

4. CONCLUSIONS

- A methodology for the construction of high-order finite-difference compact schemes for hyperbolic IBVPs which are strictly stable has been developed and analyzed theoretically.
- Fourth- and sixth-order compact implicit finite-difference schemes have been constructed and analyzed.
- To close the schemes near the boundary, the SAT procedure, proposed by Carpenter *et al.* [4], but generalized and modified accordingly, has been used. It has been shown that

this procedure does not degrade the overall accuracy of the original spatial operator and it is strictly stable under some mild conditions on the parameter τ, which appears in the algorithm, and corner entries of the differentiation matrix.

- Numerical experiments on hyperbolic model problems in one and two space dimensions have been performed, and the schemes derived in this paper have been compared to conventional ones with respect to convergence rate and long time integrations. All results show good agreement with theory and demonstrate the efficacy of this methodology when applied to hyperbolic IBVPs.

REFERENCES

1. J. C. Butcher, On the integration processes of A. Huta, *J. Austral. Math. Soc.* **3**, 203 (1963).
2. J. C. Butcher, On Runge-Kutta processes of high order, *J. Austral. Math. Soc.* **4**, 179 (1964).
3. M. H. Carpenter, D. Gottlieb, and S. Abarbanel, The stability of numerical boundary treatments for compact high-order finite-difference schemes, *J. Comput. Phys.* **108**, 272 (1993).
4. M. H. Carpenter, D. Gottlieb, and S. Abarbanel, Time-stable boundary conditions for finite difference schemes solving hyperbolic systems: Methodology and applications to high-order compact schemes, *J. Comput. Phys.* **111**, 220 (1994).
5. A. Chertock, *Strict Stability of High-Order Compact Implicit Finite-Difference Schemes—The Role of Boundary Conditions for Hyperbolic PDEs*, Ph.D. thesis, Tel-Aviv University, Tel-Aviv, Israel, November 1998.
6. B. Gustafsson, The convergence rate for difference approximations to mixed initial boundary value problems, *Math. Comp.* **29**, 396 (1975).
7. B. Gustafsson, The convergence rate for difference approximations to general mixed initial boundary vlaue problems, *SIAM J. Numer. Anal.* **18**, 179 (1981).
8. H.-O. Kreiss and J. Oliger, Comparison of accurate methods for the integration of hyperbolic equations, *Tellus* **3** (1972).
9. H.-O. Kreiss and G. Scherer, On the existence of energy estimates for difference approximations for hyperbolic systems. Technical report, Department of Scientific Computing, Uppsala University, 1977.
10. H.-O. Kreiss and L. Wu, On the stability definition of difference approximations for the initial boundary value problems, *Appl. Num. Math.* **12**, 213 (1993).
11. D. Levi and A. Tadmor, From semi-discrete to fully discrete: stability of Runge–Kutta schemes by the energy method, *SIAM Rev.* **40**, 40 (1998).
12. B. Strand, Numerical studies of hyperbolic initial boundary value problems, in *Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology*, 1996.
13. S. S. Abarbanel, A. E. Chertock, and A. Yefet, Strict stability of high-order compact implicit finite-difference schemes: The role of boundary conditions for hyperbolic PDEs, II, *J. Comput Phys.* **158**, 1–21 (2000).