

# A Pseudospectral Legendre Method for Hyperbolic Equations with an Improved Stability Condition\*

HILLEL TAL-EZER

*Tel-Aviv University, Ramat Aviv, 69 978 Tel-Aviv, Israel and  
Institute for Computer Applications in Science and Engineering,  
NASA Langley Research Center, Hampton Virginia 23665*

Received October 16, 1984; revised December 31, 1985

This paper introduces a new pseudospectral method for solving hyperbolic partial differential equations. This method uses different grid points from previously used pseudospectral methods. In fact, the grid points are related to the zeroes of the Legendre polynomials. The main advantage of this method is that the allowable time-step is proportional to the inverse of the number of grid points  $1/N$  rather than to  $1/N^2$  (as in the case of other pseudospectral methods applied to mixed initial boundary value problems). A highly accurate time discretization suitable for these spectral methods is discussed. © 1986 Academic Press, Inc.

## 1. INTRODUCTION

This article discusses some aspects of spectral methods for the solution of initial boundary value problems. The model problem can be formulated in the following way:

$$\begin{aligned} \frac{\partial U}{\partial t} - GU &= 0, \\ U(x, 0) &= U^0(x), \end{aligned} \tag{1.1}$$

where for each  $t$ ,  $U(t)$  belongs to a Hilbert space  $H$  so that  $U(t)$  satisfies homogeneous boundary conditions and  $G$  is a linear spatial differential operator. There are three commonly used spectral methods for the space discretization of (1.1): Galerkin, Tau, and pseudospectral (collocation). Each one of three methods

\* Research supported in part by the National Aeronautics and Space Administration under NASA Contract NAS1-17070. Part of this research was carried out while the author was a visitor at ICASE, NASA Langley Research Center, Hampton, Virginia 23665. The research is also supported in part by the Air Force Office of Scientific Research under Contract AFO5R 83-0089. This work was done toward partial fulfillment of the requirements of Ph.D. degree in the Department of Mathematics, Tel-Aviv University, Tel-Aviv, Israel.

can be characterized by specifying a finite dimensional subspace  $B_N \subset H$  and a projection operator  $P_N$ ,

$$P_N: H \rightarrow B_N, \quad (1.2)$$

such that

$$\lim_{N \rightarrow \infty} \|P_N u - u\| = 0. \quad (1.3)$$

Using the operator  $P_N$  results in a semidiscrete approximation of (1.1)

$$\begin{aligned} \frac{\partial}{\partial t} U_N - G_N U_N &= 0, \\ U_N(x, 0) &= U_N^0(x), \end{aligned} \quad (1.4)$$

while

$$\begin{aligned} U_N &= P_N U; & U_N^0 &= P_N U^0, \\ G_N &= P_N G P_N. \end{aligned} \quad (1.5)$$

The commonly used basis functions of the subspace  $B_N$  are related to Chebyshev or Legendre polynomials.

$G_N$  is an operator from  $B_N$  to  $B_N$ ; thus it can be viewed in the numerical procedure as an  $N \times N$  matrix. The formal solution of (1.4) is

$$U_N(t) = \exp(tG_N) U_N^0. \quad (1.6)$$

When (1.4) is discretized in time by means of an explicit finite difference scheme, the time-step is limited by a stability condition. It has been observed that the restriction on the time-step  $\Delta t$ , for Chebyshev or Legendre methods is of the form

$$\Delta t = O\left(\frac{1}{N^2}\right).$$

In fact when Eq. (1.4) (for  $G = \partial/\partial x$ ) is discretized in space by the pseudospectral Chebyshev method and in time by the modified Euler scheme, then one encounters the stability condition [1],

$$\Delta t \leq \frac{8}{N^2}. \quad (1.7)$$

The stability condition (1.7) is very stringent and has forced researchers to resort to implicit or semi-implicit time marching techniques thus complicating the program and reducing the efficiency of the method. The stability condition (1.7) had been attributed to the well-known CFL condition. Since the distribution of the grid

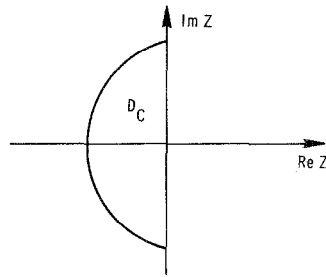


FIG. 1. Chebychev domain  $D_C$ .

points in any pseudospectral method is not uniform and  $\Delta x_{\min} = O(N^{-2})$ , then one should expect a stability condition of the form  $\Delta t \sim \Delta x_{\min}$  which agrees with (1.7). However, spectral methods are global in nature since the solution at time-step  $n + 1$  at a certain grid point depends on the solution at time-step  $n$  at all the grid points. Therefore, an argument based on domain of dependence is not valid here.

In this paper we analyze a pseudospectral method that does not have this severe limitation on the time step. This scheme is based on results obtained by M. Dubiner.<sup>1</sup> In this paper, Dubiner has carried out a detailed analysis of the spectrum of the matrix  $G_N$  for the inflow problem

$$\begin{aligned} (U_N)_t - (U_N)_x &= 0, & -1 < x < 1, \\ U_N(x, 0) &= U_N^0(x), \\ U_N(1, t) &= 0, \end{aligned} \tag{1.8}$$

for several matrices  $G_N$  resulting from various spectral approximations. He shows that if one uses the Tau method to solve (1.8) with Jacoby polynomials  $P_n^{(\alpha, \beta)}(x)$  as basis functions then the eigenvalues of  $G_N$ ,  $\lambda_i^N$ , behave asymptotically in the following way:

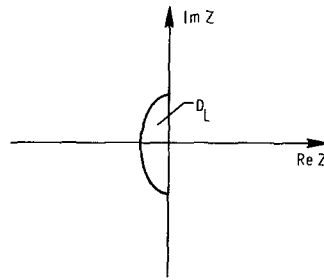
$$\begin{aligned} \lambda_i^N &= O(N^2), & \alpha \neq 0, \\ \lambda_i^N &= O(N), & \alpha = 0. \end{aligned} \tag{1.10}$$

Using this result, we propose to show that it is possible to construct a pseudospectral (collocation) algorithm for the solution of (1.8) such that the limitation on the time-step is of the form

$$\Delta t = O\left(\frac{1}{N}\right). \tag{1.11}$$

It follows from Dubiner's result (1.10) that in the Chebyshev case ( $\alpha = -\frac{1}{2}$ ), the

<sup>1</sup> M. Dubiner, 1983, Tel-Aviv University, Tel-Aviv, Israel, personal communication.

FIG. 2. Legendre domain  $D_L$ .

domain  $D_c$  (Fig. 1) in the complex plane which includes all the eigenvalues of  $G_N$ , has the size of order  $N^2$ . While in the Legendre case ( $\alpha = 0$ ), the size of the domain  $D_L$  (Fig. 2) is of order  $N$ . It is this difference in the size of the domains which results in different stability conditions.

The choice of  $\alpha = 0$  in order to get (1.11), is because the inflow is from the right boundary.<sup>1</sup> When the inflow is from the left boundary, one should choose an orthogonal polynomial with  $\beta = 0$ . For the case of inflow from both boundaries, the only choice is Legendre polynomial  $P_N^{(0,0)}$ .

In Section 2 we derive a pseudospectral method that yields the same matrix  $G_N$  that corresponds to the Tau-Legendre method and proves the stability of the exact evolution operator  $\exp(G_N t)$ .

In Section 3 we analyze the solution of the fully discrete problem. Since spectral methods in space are highly accurate, it is desirable to have a similar accuracy in time as well. A scheme which has this property is explored in Section 4. And a slightly different approach for constructing  $G_N$  is described in Section 5. The algorithm based on this approach has some advantages over the previous one from a programming point of view. On the other hand, instabilities occur when applied to a system of hyperbolic equations unless the boundary conditions are modified.

In Section 6 we describe this phenomenon of instability and try to explain its origin. We also prove in this section that the first approach is stable without any modification of the boundary conditions. We conclude with Section 7 giving some numerical results.

## 2. THE NEW PSEUDOSPECTRAL METHOD

It has been shown [1] that when the Tau method is applied to the constant coefficient hyperbolic problem

$$\begin{aligned} U_t - U_x &= 0, \\ U(x, 0) &= U^0(x), \\ U(1, t) &= 0, \end{aligned}$$

then the numerical approximation  $U_N$  satisfies exactly the equation

$$\frac{\partial U_N}{\partial t} = \frac{\partial U_N}{\partial x} + \tau(t) Q_N(x),$$

where

$$U_N = \sum_{k=0}^N A_k Q_k(x),$$

$Q_k(x)$  are any orthogonal polynomials. This has led many researchers to identify the Tau method with a collocation method where the collocation points  $x_j$  are the zeroes of  $Q_N(x)$  [2]. Note that the  $x_j$ 's lie in the open interval  $(-1, 1)$ . To construct this pseudospectral method, we define the following basis functions

$$g_j(x) = \frac{F_N(x)}{F'_N(x_j)(x - x_j)}, \quad 1 \leq j \leq N, \tag{2.1}$$

where

$$F_N(x) = (x - x_1) \cdots (x - x_N)(x - 1) = (x - 1) Q_N(x). \tag{2.2}$$

It is easily verified that

$$g_j(x_k) = \delta_{jk}, \quad 1 \leq j, k \leq N \tag{2.3}$$

and

$$g_j(1) = 0.$$

Thus  $g_j(x)$  satisfies the right-hand boundary condition. Using  $g_j(x)$  we get the interpolation polynomial of the function  $U(x)$ ,

$$P_N U(x) = \sum_{j=1}^N U(x_j) g_j(x) \tag{2.4}$$

and its derivative

$$[P_N U(x)]' = \sum_{j=1}^N U(x_j) g'_j(x). \tag{2.5}$$

Now, we solve (1.8) by substituting  $P_N U(x)$  instead of  $U_N$  and satisfying the differential equation at the interior points  $x_j$ . This, together with the homogeneous boundary condition (see remark at the end of the section for the treatment of non-homogeneous boundary conditions) results in the following set of equations:

$$\frac{dU_k}{dt} = \sum_{j=1}^N g'_j(x_k) U_j, \quad 1 \leq k \leq N, \tag{2.6}$$

where

$$U_j = U(x_j). \quad (2.7)$$

Equation (2.6) can be written in the matrix form

$$\frac{d}{dt} \bar{U}_N = G_N \bar{U}_N, \quad (2.8)$$

where  $\bar{U}_j^T$  is the vector

$$\bar{U}_N^T = (U_1, U_2, \dots, U_N) \quad (2.9)$$

and  $G_N$  is the matrix

$$(G_N)_{kj} = g'_j(x_k). \quad (2.10)$$

For  $j \neq k$ , it is easily verified that

$$g'_j(x_k) = \frac{(x_k - 1) Q'_N(x_k)}{(x_j - 1) Q'(x_j)} \frac{1}{x_k - x_j}. \quad (2.11)$$

The expression for  $(G_N)_{jj}$  is more involved. Define

$$R_i(x) = x - x_i, \quad (2.12)$$

then for  $j = k$  we get

$$\begin{aligned} g'_j(x_j) &= \frac{1}{F'_N(x_j)} \lim_{x \rightarrow x_j} \left[ \frac{(x - x_j) F'_N(x) - F_N(x)}{(x - x_j)^2} \right] \\ &= \frac{1}{F'_N(x_j)} \lim_{x \rightarrow x_j} \left[ \frac{Q_N(x) + (x - 1) Q'_N(x)}{x - x_j} - \frac{x - 1}{x - x_j} \prod_{\substack{i=1 \\ i \neq j}}^N R_i \right] \\ &= \frac{1}{F'_N(x_j)} \lim_{x \rightarrow x_j} \left\{ \prod_{\substack{i=1 \\ i \neq j}}^N R_i + \frac{x - 1}{x - x_j} \left[ \sum_{k=1}^N \prod_{\substack{i=1 \\ i \neq k}}^N R_i - \prod_{\substack{i=1 \\ i \neq j}}^N R_i \right] \right\} \\ &= \frac{1}{F'_N(x_j)} \lim_{x \rightarrow x_j} \left\{ \prod_{\substack{i=1 \\ i \neq j}}^N R_i + \frac{x - 1}{x - x_j} \left[ \sum_{\substack{k=1 \\ k \neq j}}^N \prod_{\substack{i=1 \\ i \neq k}}^N R_i \right] \right\} \\ &= \frac{1}{F'_N(x_j)} \lim_{x \rightarrow x_j} \left\{ \prod_{\substack{i=1 \\ i \neq j}}^N R_i + (x - 1) \left[ \sum_{\substack{k=1 \\ k \neq j}}^N \prod_{\substack{i=1 \\ i \neq kj}}^N R_i \right] \right\} \\ &= \frac{1}{F'_N(x_j)} \left\{ \prod_{\substack{i=1 \\ i \neq j}}^N R_i(x_j) + (x_j - 1) \left[ \sum_{\substack{k=1 \\ k \neq j}}^N \prod_{\substack{i=1 \\ i \neq kj}}^N R_i(x_j) \right] \right\} \\ &= \frac{1}{x_j - 1} + \left[ \sum_{\substack{k=1 \\ k \neq j}}^N \prod_{\substack{i=1 \\ i \neq kj}}^N R_i(x_j) \right] / \left[ \prod_{\substack{i=1 \\ i \neq j}}^N R_i(x_j) \right] = \sum_{\substack{k=1 \\ k \neq j}}^{N+1} \frac{1}{x_j - x_k} \quad (x_{N+1} = 1). \end{aligned} \quad (2.13)$$

Thus, we finally have the following expressions for  $(G_N)_{kj}$

$$(G_N)_{kj} = \begin{cases} \frac{(x_k - 1) Q'_N(x_k)}{(x_j - 1) Q'_N(x_j)} \frac{1}{x_k - x_j}, & j \neq k \\ \sum_{\substack{i=1 \\ i \neq j}}^{N+1} \frac{1}{x_j - x_i}, & j = k. \end{cases} \quad (2.14)$$

From the theory of the zeroes of Jacobi polynomials [6], we have the following identity:

$$\sum_{\substack{i=1 \\ i \neq j}}^N \frac{1}{x_j - x_i} + \frac{\alpha + 1}{2(x_j - 1)} + \frac{\beta + 1}{2(x_j + 1)} = 0, \quad (2.15)$$

where  $\alpha$  and  $\beta$  are the powers in the expression of the weight function  $w(x) = (1 - x)^\alpha(1 + x)^\beta$  of the Jacobi polynomials. In the Legendre case we have  $\alpha = \beta = 0$ ; thus expression (2.14) can be simplified

$$(G_N)_{kj} = \begin{cases} \frac{(x_k - 1) Q'_N(x_k)}{(x_j - 1) Q'_N(x_j)} \frac{1}{x_k - x_j}, & j \neq k \\ \frac{1}{x_j^2 - 1}, & j = k. \end{cases} \quad (2.16)$$

From the programming point of view it is convenient to define

$$(H_N)_{ij} = (x_j - 1) Q'_N(x_j) \delta_{ij}, \quad (2.17)$$

then

$$G_N = H_N \bar{G}_N H_N^{-1}, \quad (2.18)$$

where

$$(\bar{G}_N)_{kj} = \begin{cases} \frac{1}{x_k - x_j}, & j \neq k \\ \frac{1}{x_j^2 - 1}, & j = k. \end{cases} \quad (2.19)$$

To use the operator  $G_N$ , one has to store only two vectors  $\bar{x}$ ,  $\bar{v}$  where

$$\begin{aligned} (\bar{x})_k &= x_k, \\ (\bar{v})_k &= Q'_N(x_k). \end{aligned} \quad (2.20)$$

The number of multiplications is

$$N + N^2 + N = N^2 + 2N \sim N^2. \quad (2.21)$$

This number should be compared with  $CN \log N$  in the Chebyshev case (using FFT). For small  $N$  (up to  $N=64$ ) the two results are of the same magnitude.

The stability proof for the solution of the semidiscrete problem (1.8) is straightforward. Define the following vector norm:

$$\|\bar{v}\|_w = \left[ \sum_{j=1}^N w_j v_j^2 \right]^{1/2} \quad (2.22)$$

where  $w_j$  are the weights used in the Gauss-Legendre quadrature, namely

$$w_j = \frac{2[Q_N(1)]^2}{(1-x_j^2)[Q'_N(x_j)]^2}, \quad (2.23)$$

then

$$\begin{aligned} \frac{\partial}{\partial t} \sum_{j=1}^N w_j U_N^2(x_j) &= 2 \sum_{j=1}^N w_j U_N(x_j) \frac{\partial}{\partial t} U_N(x_j) \\ &= 2 \sum_{j=1}^N w_j U_N(x_j) \frac{\partial}{\partial x} U_N(x_j). \end{aligned} \quad (2.24)$$

The function  $U_N(\partial/\partial x) U_N(x)$  is a polynomial of degree  $2N-1$ . Therefore, the Gauss-Legendre quadrature yields the exact value of the integral. Thus, we get

$$\frac{\partial}{\partial t} \sum_{j=1}^N w_j U_N^2(x_j) = 2 \int_{-1}^1 U_N \frac{\partial}{\partial x} U_N dx = [U_N^2]_{-1}^1 = U_N^2(1) - U_N^2(-1). \quad (2.25)$$

Using the boundary condition,  $U_N(1)=0$ , results in

$$\frac{\partial}{\partial t} \|UN\|_w \leq 0. \quad (2.26)$$

Since

$$\|U_N\|_w = \|\exp(G_N t) U_N^0\|_w$$

we finally have

$$\|\exp(G_N t)\|_w \leq 1. \quad (2.27)$$

However, Dubiner's paper does not carry out a detailed analysis for the other two typical problems: (1) outflow, (2) inflow from both boundaries. It demonstrates how this analysis can be done and that the results concerning the asymptotic behavior of the eigenvalues will be similar to (1.10). We would like to show how we define the operators  $P_N$  and  $G_N$  for these two problems.



(1) *Outflow.* A model problem for outflow in both boundaries is

$$\begin{aligned} (U_N)_t + x(U_N)_x &= 0, & -1 < x < 1, \\ U_N(x, 0) &= U_N^0(x). \end{aligned} \tag{2.28}$$

This problem is well posed without any boundary conditions. We therefore define the basis functions  $g_j(x)$  as

$$g_j(x) = \frac{Q_N(x)}{Q'_N(x_j)} \frac{1}{x - x_j}, \tag{2.29}$$

where

$$Q_N(x) = (x - x_1) \cdots (x - x_N) \quad (\text{Legendre polynomial}) \tag{2.30}$$

and

$$P_N U(x) = \sum_{j=1}^N U(x_j) g_j(x); \tag{2.31}$$

consequently

$$[P_N U(x)]' = \sum_{j=1}^N U(x_j) g'_j(x). \tag{2.32}$$

The elements of the matrix  $G_N$  are

$$(G_N)_{jk} = \begin{cases} -\frac{Q'_N(x_j)}{Q'_N(x_k)} \frac{x_j}{x_j - x_k}, & j \neq k \\ \frac{x_j^2}{x_j^2 - 1}, & j = k. \end{cases} \tag{2.33}$$

Using the similarity transformation

$$G_N = H_N \bar{G}_N H_N^{-1}, \tag{2.34}$$

we get

$$(H_N)_{ij} = Q'_N(x_j) \delta_{ij} \tag{2.35}$$

and

$$(\bar{G}_N)_{jk} = \begin{cases} -\frac{x_j}{x_j - x_k}, & j \neq k \\ \frac{x_j^2}{x_j^2 - 1}, & j = k. \end{cases} \tag{2.36}$$

(2) *Inflow from both boundaries.* The semidiscrete representation of the PDE is

$$\begin{aligned}(U_N)_t - x(U_N)_x &= 0, \\ U_N(x, 0) &= U_N^0(x), \\ U_N(-1, t) &= U_N(1, t) = 0.\end{aligned}\tag{2.37}$$

Since the basis functions have to satisfy

$$g_j(1) = g_j(-1) = 0,$$

we define

$$g_j(x) = \frac{S_N(x)}{S'_N(x_j)} \frac{1}{x - x_j},\tag{2.38}$$

where

$$\begin{aligned}S_N(x) &= (x - x_0)(x - x_1) \cdots (x - x_1)(x - x_{N+1}) = (x^2 - 1) Q_N(x), \\ (x_0 = -1; x_{N+1} &= 1),\end{aligned}\tag{2.39}$$

and

$$P_N U(x) = \sum_{j=1}^N U(x_j) g_j(x).\tag{2.40}$$

The elements of the matrix  $G_N$  in this case are

$$(G_N)_{jk} = \begin{cases} \frac{(x_j^2 - 1) Q'_N(x_j)}{(x_k^2 - 1) Q'_N(x_k)} \frac{x_j}{x_j - x_k}, & j \neq k \\ \frac{x_j^2}{x_j^2 - 1}, & j = k \end{cases}\tag{2.41}$$

$G_N$  is similar to  $\bar{G}_N$ ,

$$G_N = H_N \bar{G}_N H_N^{-1}\tag{2.42}$$

while

$$(H_N)_{ij} = (x_j^2 - 1) Q'_N(x_j) \delta_{ij}\tag{2.43}$$

and

$$(\bar{G}_N)_{jk} = \begin{cases} \frac{x_j}{x_j - x_k}, & j \neq k \\ \frac{x_j^2}{x_j^2 - 1}, & j = k. \end{cases}\tag{2.44}$$

*Remark.* When the boundary conditions are inhomogeneous, we have to modify our representation in the following way: For the right inflow problem we add another basis function

$$g_{N+1}(x) = \frac{F_N(x)}{F'_N(1)} \frac{1}{x-1} \tag{2.45}$$

and we have

$$g_{N+1}(1) = 1; \quad g_{N+1}(x_k) = 0, \quad 1 \leq k \leq N. \tag{2.46}$$

Thus, instead of (2.8) we get

$$\frac{d}{dt} \bar{U}_N = G_N \bar{U}_N + f(t) \bar{R}_N \tag{2.47}$$

while

$$f(t) = U(1, t) \text{ is the boundary condition} \tag{2.48}$$

and

$$\bar{R}_N^T = (g'_{N+1}(x_1), \dots, g'_{N+1}(x_N)) = \frac{1}{Q_N(1)} \bar{v} \tag{2.49}$$

( $\bar{v}$  is defined by (2.20)). When we have boundary conditions on both sides of the interval, we add two basis functions

$$\begin{aligned} g_0(x) &= \frac{S_N(x)}{S'_N(-1)} \frac{1}{x+1}, \\ g_{N+1}(x) &= \frac{S_N(x)}{S'_N(1)} \frac{1}{x-1}, \end{aligned} \tag{2.50}$$

hence

$$\begin{aligned} g_0(-1) &= 1; & g_0(x_k) &= 0, & 1 \leq k \leq N+1 & \quad (x_{N+1} = 1), \\ g_{N+1}(1) &= 1; & g_{N+1}(x_k) &= 0, & 0 \leq k \leq N & \quad (x_0 = -1). \end{aligned} \tag{2.51}$$

Thus, instead of (2.8) we get

$$\frac{d}{dt} \bar{U}_N = G_N \bar{U}_N + f(t) \bar{V}_N^L + g(t) \bar{V}_N^R \tag{2.52}$$

while

$$\begin{aligned} f(t) &= U(-1, t) && \text{is the left boundary condition,} \\ g(t) &= U(1, t) && \text{is the right boundary condition} \end{aligned}$$

and

$$(\bar{V}_N^L)_k = g'_0(x_k) = -\frac{1}{2Q_N(-1)}(x_k - 1)Q'_N(x_k), \quad (2.53)$$

$$(\bar{V}_N^R)_k = g'_{N+1}(x_k) = \frac{1}{2Q_N(1)}(x_k + 1)Q'_N(x_k). \quad (2.54)$$

### 3. THE FULLY DISCRETE SOLUTION

The exact solution of (1.4) is

$$U_n(x, t) = \exp(tG_N) U_N^0. \quad (3.1)$$

In [4] it has been shown that any explicit time algorithm can be represented as a polynomial approximation of the exact evolution operator  $\exp(tG_N)$ ; thus the fully discrete solution of (1.1) is

$$V_N^M(x, t) = H_M(tG_N) U_N^0, \quad (3.2)$$

where  $H_M(z)$  is polynomial of degree  $M$  which converges to  $e^z$  in the domain that includes all the eigenvalues of the matrix  $tG_N$ . The eigenvalues of  $tG_N$  are distinct;<sup>1</sup> therefore the corresponding eigenvectors are linearly independent and we can define a matrix  $S_N$  whose columns are the eigenvectors of  $G_N$  such that

$$U_N - V_N^M = [\exp(tG_N) - H_M(tG_N)] U_N^0 = (S_N E_N S_N^{-1}) U_N^0, \quad (3.3)$$

where  $E_N$  is a diagonal matrix whose elements are

$$(E_N)_{kk} = e^{\lambda_k t} - H_M(\lambda_k t), \quad (3.4)$$

Therefore, if

$$|e^z - H_M(z)| \xrightarrow{M \rightarrow \infty} 0, \quad z \in I_N \quad (3.5)$$

while  $I_N$  is the domain in the complex plane which includes all the eigenvalues of  $tG_N$ , then

$$\|U_N - V_N^M\| \xrightarrow{N, M \rightarrow \infty} 0. \quad (3.6)$$

The relation between  $M$  and  $N$  depends mainly on three factors:

- (1) The rate of convergence of  $H_M(z)$  to  $e^z$ .
- (2) The size of the domain  $I_N$ .
- (3) The norm of the matrices  $S_N$ ,  $S_N^{-1}$ .

In [4] we find that for periodic problems where  $\|S_N\| = \|S_N^{-1}\| = 1$  (the eigenvectors

are orthogonal) one has to choose  $M$  such that the scalar function  $H_M(z)$  resolves the exponent function  $e^z$  for  $z \in I_N$ . In the case of boundary value problems, the analysis is much more complicated since the eigenvectors are not orthogonal. We were not able to get an expression for the norms of  $S_N$  and  $S_N^{-1}$ . However, numerical experiments verify the assumption that asymptotically one can get a sufficient condition relating  $M$  and  $N$  by carrying out an analysis based on the concept of resolution.

Consider for example the modified Euler scheme. In the constant coefficients case, it is equivalent to the second or Taylor series method

$$V_N^{k+1} = (I + \Delta t G_N + \frac{1}{2}(\Delta t)^2 G_N^2) V_N^k \tag{3.7}$$

or

$$V_N^k = (I + \Delta t G_N + \frac{1}{2}(\Delta t)^2 G_N^2)^k U_N^0. \tag{3.8}$$

If  $n$  is the number of time-steps required to march to time level  $t$ , i.e.,

$$\Delta t = t/n \tag{3.9}$$

then

$$V_N(t) = V_N^n = \left( I + \frac{1}{n} (G_N t) + \frac{1}{2n^2} (G_N t)^2 \right)^n U_N^0.$$

Thus, the numerical evolution operator is

$$H_M(G_N t) = \left( I + \frac{1}{n} (G_N t) + \frac{1}{2n^2} (G_N t)^2 \right)^n \quad (M = 2n); \tag{3.10}$$

upon substituting  $z$  for  $G_N t$  we get

$$H_M(z) = \left( 1 + \frac{1}{n} z + \frac{1}{2n^2} z^2 \right)^n. \tag{3.11}$$

Since

$$e^z = (e^{z/n})^n = \left[ 1 + \frac{z}{n} + \frac{1}{2} \left( \frac{z}{n} \right)^2 + R \right]^n = [H_M(z)^{1/n} + R]^n \tag{3.12}$$

while

$$R = \left( \exp \left( \theta \frac{z}{n} \right) \left( \frac{z}{n} \right)^3 \right) / 6, \quad 0 \leq \theta \leq 1 \tag{3.13}$$

we get

$$|e^z - H_M(z)| = n(H_M(z))^{(n-1)/n} R + \text{low order terms} \tag{3.14}$$

substituting  $(\exp(z/n) - R)$  for  $[H_M(z)]^{1/n}$  results in the following expression for the relative error  $E$

$$E \approx \frac{n[\exp(z/n) - R]^{n-1} R}{\exp(z)} \approx \frac{nR \exp(z(n-1)/n)}{\exp(z)} \quad \left( \text{for } \left| \frac{z}{n} \right| < 1 \right). \quad (3.15)$$

Using (3.13) in (3.15) gives

$$E \approx \frac{1}{6} \frac{z^3}{n^2} \exp\left(\left(\theta - 1\right) \frac{z}{n}\right). \quad (3.16)$$

Thus, resolution of  $e^z$  by  $H_M(z)$  is achieved when

$$\left| \frac{1}{6} \frac{z^3}{n^2} \exp\left(\left(\theta - 1\right) \frac{z}{n}\right) \right| < \varepsilon. \quad (3.17)$$

(The magnitude of  $\varepsilon$  is problem dependent.) From (1.10), (3.9), and (3.10) it follows that in order to satisfy (3.17) we have to choose  $M$  such that

$$M = O(N^{3/2}) \quad (3.18)$$

or, equivalently

$$\Delta t = O\left(\left(\frac{1}{N}\right)^{3/2}\right). \quad (3.19)$$

The power  $\frac{3}{2}$  is due to the fact that the modified Euler scheme is second order accurate in time. A similar analysis for any explicit scheme which is  $p$  order accurate in time will yield the following condition

$$M = O((N^{(p+1)/p}). \quad (3.20)$$

It is obvious from this expression that using a scheme which has high accuracy in time will lead to the desired condition

$$M = O(N); \quad (3.21)$$

such a scheme is described in the next section.

*Remark.* Since we assume that resolution implies stability, conditions (3.18)–(3.10) are sufficient but not necessary. It is possible to get stable results while  $M$  satisfies  $M = O(N)$  even for second order in time scheme as shown by the numerical results presented in Section 7.

4. HIGHLY ACCURATE TIME DISCRETIZATION

The formal representation of an explicit fully discrete solution of (1.1) is given by (3.2). Since spectral methods in space are highly accurate, we would like to find a polynomial  $H_M(z)$  that will yield high accuracy in time as well. Such a polynomial is described in [4] for pure initial value problems. It is based on approximating the function  $e^z$  by orthogonal polynomials. We would like to show how to implement this approach in the case of inflow-outflow boundary conditions.

The main difference between the present case and the periodic one is the topological structure of the domain that includes the eigenvalues of  $tG_N$ .

$$|\operatorname{Re} \lambda_i^N| \leq C_1, \quad |I_m \lambda_i^N| \leq C_2(N), \tag{4.1}$$

where  $\lambda_i^N$  are the eigenvalues of  $tG_N$ . (The constant  $C_1$  does not depend on  $N$ , while  $C_2(N) = O(N)$ .) Whereas, in the boundary value case the eigenvalues satisfy<sup>1</sup>

$$|\operatorname{Re} \lambda_i^N| \leq C_1(N); \quad |I_m \lambda_i^N| \leq C_2(N). \tag{4.2}$$

(Usually  $C_{1,2}(N) = O(N^2)$ . In Section 2 we have defined projection operators such that the related eigenvalues satisfy  $C_{1,2}(N) = O(N)$ ).

Accounting for this fact we have to modify the Orthogonal Polynomials Scheme (OPS) which is described in [4].

Define

$$s = \max_i |\operatorname{Re}(\lambda_i^N)|, \tag{4.3}$$

$$R = \max_i |I_m(\lambda_i^N)|. \tag{4.4}$$

Since resolution of  $e^z$  by  $H_M(z)$  means

$$E = \max_z \left| \frac{e^z - H_M(z)}{e^z} \right| < \varepsilon, \quad z \in D \text{ (domain of the eigenvalues)} \tag{4.5}$$

for small enough  $\varepsilon$ , we would like to choose  $H_M(z)$  such that  $E$  is small for given  $M$ . Approximation based on the polynomials  $\phi_k(w)$  defined in [4] (i.e., orthogonal polynomials on the imaginary axis) will converge in  $D$  but will result in relatively large error  $E$ . Accounting for the fact that the denominator of  $E$  achieves its minimum in the left side of  $D$ , it is advisable to use a set of polynomials which are orthogonal on the line  $\operatorname{Re}(z) = -s$ . Using this set of polynomials is equivalent to approximating  $e^z$  through the following change of variables

$$\bar{z} = z + s. \tag{4.6}$$

Thus

$$e^z = e^{-s} e^{z+s} = e^{-s} e^{\bar{z}} = e^{-s} e^{R(z;R)}. \tag{4.7}$$

Therefore

$$e^z = \sum_{k=0}^{\infty} b_k \phi_k(w); \quad w = \bar{z}/R, \quad (4.8)$$

where

$$b_k = e^{-s} C_k J_k(R), \quad (4.9)$$

$$C_k = \begin{cases} 1, & k=0, \\ 2, & k \geq 1. \end{cases} \quad (4.10)$$

$J_k(R)$  is Bessel function of order  $k$ .  $\phi_k(w)$  satisfy the following recurrence relation

$$\begin{aligned} \phi_k(w) &= 2w\phi_{k-1}(w) + \phi_{k-2}(w), \\ \phi_0(w) &= 1; \quad \phi_1(w) = w. \end{aligned} \quad (4.11)$$

Thus, substituting the operator  $\bar{G}_N$

$$\bar{G}_N = \frac{1}{R} (tG_N + sI) \quad (4.12)$$

instead of  $w$  in (4.8) results in the following approximation of the evolution operator

$$e^{tG_N} \approx H_M(tG_N) = \sum_{k=0}^M b_k \phi_k(\bar{G}_N) \quad (4.13)$$

and the fully discrete numerical solution is

$$V_N^M = \left[ \sum_{k=0}^M b_k \phi_k(\bar{G}_N) U_N^0 \right], \quad (4.14)$$

where

$$\begin{aligned} \phi_k(\bar{G}_N) U_N^0 &= 2\bar{G}_N \phi_{k-1}(\bar{G}_N) U_N^0 + \phi_{k-2}(\bar{G}_N) U_N^0, \\ \phi_0(\bar{G}_N) U_N^0 &= U_N^0, \\ \phi_1(\bar{G}_N) U_N^0 &= \bar{G}_N U_N^0. \end{aligned}$$

Numerical experiments show that while using the pseudospectral projection operator defined in Section 2 for the solution of the problems: (1) outflow, (2) inflow from both boundaries, we have

$$s \gg R. \quad (4.15)$$

Since, in this case the eigenvalues are grouped close to the real axis, the scheme



described in [5] (for parabolic problems) will perform better than (4.14). (This conclusion is valid for any problem where we have an a priori information about the domain  $D$  similar to (4.15).) Hence we approximate the evolution operator  $\exp(tG_N)$  in the following way:

$$\exp(tG_N) \approx H_n(tG_N) = \sum_{k=0}^n d_k T_k(\bar{G}_N), \tag{4.16}$$

where

$$\bar{G}_N = \frac{2}{S} \left( tG_N + \frac{S}{2} I \right) \tag{4.17}$$

$$d_k = e^{-s/2} I_k \left( \frac{s}{2} \right). \tag{4.18}$$

$I_k$  is modified Bessel function of order  $k$  and  $T_k(x)$  is Chebyshev polynomial. The numerical solution at time level  $t$  is

$$V_N^M = \left[ \sum_{k=0}^n d_k T_k(\bar{G}_N) U_N^0 \right]. \tag{4.19}$$

$T_k(\bar{G}_N) U_N^0$  is computed by using the recurrence relation

$$\begin{aligned} T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x), & k \geq 2, \\ T_0(x) &= 1; & T_1(x) = x. \end{aligned} \tag{4.20}$$

Thus

$$\begin{aligned} T_k(\bar{G}_N) U_N^0 &= 2\bar{G}_N T_{k-1}(\bar{G}_N) U_N^0 - T_{k-2}(\bar{G}_N) U_N^0, \\ T_0(\bar{G}_N) U_N^0 &= U_N^0; & T_1(\bar{G}_N) U_N^0 &= \bar{G}_N U_N^0. \end{aligned} \tag{4.21}$$

*Remark.* (4.1) has been proven in [5] for the periodic problem

$$\begin{aligned} U_t - a(x) U_x &= 0, \\ U(x, 0) &= U^0(x), \end{aligned}$$

where  $a(x) = \sin 2x$ . Similar technique can be applied to prove (4.1) when  $a(x)$  is any second degree trigonometric polynomial. Numerical experiments verify the assumption that (4.1) is valid for the general case, when  $a(x)$  can be represented as a finite degree trigonometric polynomial.

### 5. MODIFICATION OF THE PSEUDOSPECTRAL METHOD

The operator  $P_N$  defined in Section 2 leads to some complexity when the boundary conditions are inhomogeneous, as shown by (2.45)–(2.54). It is possible to overcome this difficulty by using a slightly different operator  $P_N$ .

Define the basis functions  $g_j(x)$

$$g_j(x) = \frac{F_N(x)}{F'_N(x_j)(x-x_j)}, \quad 0 \leq j \leq N+1 \quad (5.1)$$

$$(x_0 = -1; x_{N+1} = 1),$$

where

$$F_N(x) = (x+1)(x-x_1) \cdots (x-x_N)(x-1) = (x^2-1) Q^N(x), \quad (5.2)$$

then

$$U_N(x) = P_N U(x) = \sum_{j=0}^{N+1} U(x_j) g_j(x) \quad (5.3)$$

is polynomial of order  $N+1$  interpolating  $U$  at the points  $x_j, j=0, 1, \dots, N, N+1$ . Its derivative is given by

$$[P_N U(x)]' = \sum_{j=0}^{N+1} U(x_j) g'_j(x). \quad (5.4)$$

By using this projection operator we get the matrix  $D_N$  (the numerical derivative operator) whose elements are

$$(D_N)_{kj} = \begin{cases} \frac{F'_N(x_k)}{F'_N(x_j)} \frac{1}{x_k - x_j}, & k \neq j \\ \frac{x_j}{x_j^2 - 1}, & j = k. \end{cases} \quad (5.5)$$

The matrix  $D_N$  can be written as

$$D_N = H_N \tilde{D}_N H_N^{-1}, \quad (5.6)$$

where

$$(\tilde{D}_N)_{kj} = \begin{cases} \frac{1}{x_k - x_j}, & k \neq j, \\ \frac{x_j}{x_j^2 - 1}, & 0 < j = K < N+1, \\ \frac{Q'_N(-1)}{Q_N(-1)} - \frac{1}{2}, & j = k = 0, \\ \frac{Q'_N(1)}{Q_N(1)} + \frac{1}{2}, & j = k = N+1, \end{cases} \quad (5.7)$$

and  $H_N$  is a diagonal matrix

$$(H_N)_{kk} = F'_N(x_k) = \begin{cases} -2Q_N(-1), & k=0, \\ (x_k^2 - 1) Q'_N(x_k), & 0 < k < N+1, \\ 2Q_N(1), & k = N+1. \end{cases} \quad (5.8)$$

Thus

$$G_N = B_N D_N B_N = B_N H_N \tilde{D}_N H_N^{-1} B_N, \quad (5.9)$$

where  $B_N$  is a diagonal matrix

$$(B_N)_{kk} = \begin{cases} 1, & 0 \leq k \leq N, \\ 0, & k = N+1. \end{cases} \quad (5.10)$$

Thus, we find that the algorithm is stable and the eigenvalues of  $G_N$  are  $O(N)$ . The main difference between the strategy used in Section 2 and the present one is the following: In the first case we follow exactly the PDE and satisfy the equation only in the interior of the domain. The boundary conditions are satisfied by a proper choice of the basis functions. In the second case we satisfy the equation in the interior and boundary domain while imposing the boundary conditions at the end of each time-step. Apparently, the first approach follows the PDE more accurately than the second one. This statement will be made clear in the next section where we describe the solution of system of equations.

### 6. SYSTEM OF HYPERBOLIC EQUATIONS

Consider the symmetric system

$$\begin{aligned} \begin{pmatrix} U \\ V \end{pmatrix}_t &= \begin{pmatrix} \frac{1}{2} & 1 \\ 1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} U \\ V \end{pmatrix}_x, & -1 < x < 1, \\ U(-1) &= f(t); & U(1) &= g(t), \end{aligned} \quad (6.1)$$

$$\begin{pmatrix} U \\ V \end{pmatrix}_{x=0} = \begin{pmatrix} U^0(x) \\ V^0(x) \end{pmatrix}.$$

It is well known [3] that using a matrix of the type  $D_N$  defined in the previous section to numerically compute the spatial derivatives leads to instability although the differential equations (6.1) are well posed. It is shown in [3] how to stabilize the algorithm by adding numerical boundary conditions for the function  $V(x)$ . This approach is based on the following argument.

The characteristic variables are  $U + V$  and  $U - V$  and (6.1) is equivalent to the following diagonal set of equations

$$\begin{aligned}(U + V)_t &= \frac{3}{2}(U + V)_x, \\ (U - V)_t &= -\frac{1}{2}(U - V)_x,\end{aligned}\tag{6.2}$$

$U + V$  is constant on the characteristic  $dx/dt = -\frac{3}{2}$  and  $U - V$  is constant on the characteristic  $dx/dt = \frac{1}{2}$ . Therefore,  $U + V$  should be given on the right boundary and should be determined by the scheme on the left boundary. Similarly  $U - V$  should be given on the left boundary and should be determined by the scheme on the right boundary.

The scheme is stabilized by requiring that the values of  $U + V$  at  $x = -1$  and  $U - V$  at  $x = 1$  are not changed as a result of imposing the boundary conditions  $U(1)$  and  $U(-1)$ .

It seems that this instability can be traced to the fact that by using the matrix  $D_N$  we use the PDE in the closed interval instead of the open interval as indicated by (6.1). By doing so we get errors on the boundaries for both  $U(x)$  and  $V(x)$ . While the error in the function  $U(x)$  is immediately corrected by imposing the boundary conditions, the error in  $V(x)$  penetrates into the system through the characteristics and causes the instability. On the other hand, using the approach described in Section 2, we follow exactly the PDE without imposing it on the boundaries, thus we do not expect this phenomenon of instability. This assumption is proved by the following theorem:

**THEOREM.** *The solution of the semidiscrete problem*

$$\begin{aligned}\begin{pmatrix} U_N \\ V_N \end{pmatrix}_t &= \begin{pmatrix} \frac{1}{2} & 1 \\ 1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} U_N \\ V_N \end{pmatrix}_x, & -1 < x < 1, \\ U_N(-1) &= 0; & U_N(1) = 0,\end{aligned}\tag{6.3}$$

$$\begin{pmatrix} U_N \\ V_N \end{pmatrix}_{x=0} = \begin{pmatrix} U_N^0(x) \\ V_N^0(x) \end{pmatrix}$$

discretized by the projection operator (2.38)–(2.40) for  $U$  and (2.29)–(2.31) for  $V$ , is stable.

*Proof.* From (2.38)–(2.40) it follows that  $U_N$  is a polynomial of degree  $N + 1$ . It can be represented as

$$U_N = \sum_{k=0}^{N+1} \hat{u}_k P_k(x),\tag{6.4}$$

<sup>2</sup>I would like to acknowledge my advisor, Professor David Gottlieb, for this proof.

where the  $P_k(x)$ 's are the normalized Legendre polynomials [ $P_k(\pm 1) = (\pm 1)^k$ ]. From (2.29)–(2.31) it follows that  $V_N$  is a polynomial of degree  $N - 1$  and therefore

$$V_N = \sum_{k=1}^{N-1} \hat{v}_k P_k(x). \tag{6.5}$$

Accounting for (6.4) and (6.5) the polynomials  $U_N$  and  $V_N$  satisfy *exactly* the following equations

$$(U_N)_t = \frac{1}{2}(U_N)_x + (V_N)_x + E_{N+1}(x), \tag{6.6a}$$

$$(V_N)_t = (U_N)_x + \frac{1}{2}(V_N)_x + F_N(x), \tag{6.6b}$$

where  $E_{N+1}$ ,  $F_N$  are polynomials of degree  $N + 1$ ,  $N$  respectively, which vanish at the grid points. Therefore, we can write

$$E_{N+1}(x) = (a_1 x + b_1) P_N(x), \tag{6.7}$$

$$F_N(x) = a_2 P_N(x). \tag{6.8}$$

In fact  $a_1$ ,  $b_1$  and  $a_2$  are given by

$$a_1 = \frac{2N + 1}{N + 1} \frac{d}{dt} \hat{u}_{N+1}, \tag{6.9a}$$

$$b_1 = \frac{d}{dt} \hat{u}_N - \frac{1}{2} (2N + 1) \hat{u}_{N+1}, \tag{6.9b}$$

$$a_2 = -(2N + 1) \hat{u}_{N+1}. \tag{6.9c}$$

Equations (6.9) are proved by making use of the following relation satisfied by Legendre polynomials [6],

$$x P_N(x) = \frac{N + 1}{2N + 1} P_{N+1}(x) + \frac{N}{2N + 1} P_{N-1}(x) \tag{6.10}$$

and the fact that  $(U_N)_x$  is a polynomial of degree  $N$  whose highest coefficient is  $(2N + 1) \hat{u}_{N+1}$ . Thus, equating the coefficients of  $P_{N+1}$  in (6.6a) results in (6.9a). Similarly, equating the coefficients of  $P_N$  in (6.6a) results in (6.9b). Finally, equating the coefficients of  $P_N$  in (6.6b) results in (6.9c).

Define now the characteristic variables  $R_N$  and  $S_N$ ,

$$R_N = U_N - V_N = \sum_{k=0}^{N+1} \hat{s}_k P_k(x) = \sum_{k=0}^{N-1} \hat{s}_k P_k(x) + \hat{u}_N P_N(x) + \hat{u}_{N+1} P_{N+1}(x) \tag{6.11a}$$

$$S_N = U_N + V_N = \sum_{k=0} \hat{s}_k P_k(x) = \sum_{k=0} \hat{s}_k P_k(x) + \hat{u}_N P_N(x) + \hat{u}_{N+1} P_{N+1}(x), \tag{6.11b}$$

it is easily verified by (6.6), (6.7) that

$$(S_N)_t = \frac{3}{2}(S_N)_x + (a_1x + b_1 + a_2) P_N(x), \tag{6.12a}$$

$$(R_N)_t = -\frac{1}{2}(R_N)_x + (a_1x + b_1 - a_2) P_N(x), \tag{6.12b}$$

multiplying (6.12b) by 3, adding it to (6.12a) and using the technique of equating highest coefficients of  $P_{N-1}$  results in

$$a_1 = \frac{2N+1}{4N} \frac{d}{dt} (\hat{s}_{N-1} + 3\hat{r}_{N-1}). \tag{6.13}$$

Next, we use (6.12) to get

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} \int_{-1}^1 (S_N^2 + 3R_N^2) dx &= \frac{3}{4} [S_N^2 - R_N^2]_{-1}^1 + \int_{-1}^1 S_N(a_1x + b_1 + a_2) P_N dx \\ &\quad + 3 \int_{-1}^1 R_N(a_1x + b_1 + a_2) P_N dx. \end{aligned} \tag{6.14}$$

The first term on the rhs of (6.14) vanishes due to boundary conditions ( $S_N^2 = R_N^2$  on the boundaries).

From (6.9), (6.10), (6.11), (6.13) and the fact that Legendre polynomials are orthogonal, we get

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} \int_{-1}^1 (S_N^2 + 3R_N^2) dx &= 2\alpha_{N+1} \frac{d}{dt} \hat{u}_{N+1}^2 + \frac{1}{8} 2\alpha_{N-1} \frac{d}{dt} (\hat{s}_{N-1} + 3\hat{r}_{N-1})^2 \\ &\quad + 2\alpha_N \frac{d}{dt} \hat{u}_N^2, \end{aligned} \tag{6.15}$$

where

$$\alpha_k = \int_{-1}^1 P_k^2(x) dx = \frac{2}{2k+1}.$$

On the other hand we have

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} \int_{-1}^1 (S_N^2 + 3R_N^2) dx &= \frac{1}{2} \frac{d}{dt} \left[ \sum_{k=0}^{N-2} \alpha_k (\hat{s}_k^2 + 3\hat{r}_k^2) \right] + \frac{1}{2} \alpha_{N-1} \frac{d}{dt} (\hat{s}_{N-1}^2 + 3\hat{r}_{N-1}^2) \\ &\quad + 2\alpha_N \frac{d}{dt} \hat{u}_N^2 + 2\alpha_{N+1} \frac{d}{dt} \hat{u}_{N+1}^2. \end{aligned} \tag{6.17}$$

Equating (6.15) and (6.17) gives us

$$\frac{1}{2} \frac{d}{dt} \left[ \sum_{k=0}^{N-2} \alpha_k (\hat{s}_k^2 + 3\hat{r}_k^2) + \frac{3}{8} \alpha_{N-1} (\hat{s}_{N-1} - \hat{r}_{N-1})^2 \right] = 0 \tag{6.18}$$

hence

$$\sum_{k=0}^{N-2} \alpha_k (\hat{s}_k^2 + 3\hat{r}_k^2) + \frac{3}{8} \alpha_{N-1} (\hat{s}_{N-1} - \hat{r}_{N-1})^2 = C, \tag{6.19}$$

where  $C$  is constant in time.

From (6.11) and (6.19) we get

$$\sum_{k=0}^{N-2} \alpha_k (\hat{s}_k^2 + 3\hat{r}_k^2) + \frac{3}{4} \alpha_{N-1} \hat{v}_{N-1}^2 = C. \tag{6.20}$$

All the terms on the l.h.s. of (6.20) are positive; therefore using (6.11) results in

$$\begin{aligned} |\hat{u}_j(t)| &\leq C_1, & 0 \leq j \leq N-2, \\ |\hat{v}_j(t)| &\leq C_2, & 0 \leq j \leq N-1, \end{aligned} \tag{6.21}$$

while  $C_1, C_2$  are constant in time. Due to boundary conditions we have

$$\begin{aligned} U(-1) &= \sum_{k=0}^{N+1} (-1)^k \hat{u}_k = 0, \\ U(1) &= \sum_{k=0}^{N+1} \hat{u}_k = 0. \end{aligned} \tag{6.22}$$

Thus for  $N$  even

$$\begin{aligned} -\hat{u}_{N+1} + \hat{u}_N - \hat{u}_{N+1} &= -\sum_{k=0}^{N-2} (-1)^k \hat{u}_k, \\ \hat{u}_{N-1} + \hat{u}_N + \hat{u}_{N+1} &= -\sum_{k=0}^{N-2} \hat{u}_k, \end{aligned} \tag{6.23}$$

or

$$\hat{u}_N = -\frac{1}{2} \sum_{\substack{k=0 \\ k \text{ even}}}^{N-2} \hat{u}_k, \tag{6.24}$$

$$\hat{u}_{N+1} + \hat{u}_{N-1} = -\frac{1}{2} \sum_{\substack{k=1 \\ k \text{ odd}}}^{N-3} \hat{u}_k. \tag{6.25}$$

Since  $\hat{u}_k$  for  $k = 1, \dots, N-2$  are bounded (6.21), then  $\hat{u}_N$  is bounded. To show that (6.25) implies the boundedness of  $\hat{u}_{N-1}$  and  $\hat{u}_{N+1}$  we make use of (6.9a) and (6.12). Equating the two expressions for  $a_1$  results in

$$\frac{d}{dt} \hat{u}_{N+1} = \frac{N+1}{N} \frac{d}{dt} \left( \hat{u}_{N-1} - \frac{1}{2} \hat{v}_{N-1} \right). \tag{6.26}$$

$\hat{v}_{N-1}$  is bounded; therefore in the limit we get

$$\frac{d}{dt} \hat{u}_{N+1} = \frac{d}{dt} \hat{u}_{N-1} \quad (\text{for } N \rightarrow \infty). \quad (6.27)$$

Equations (6.25) and (6.27) imply the boundedness of  $\hat{u}_{N-1}$  and  $\hat{u}_{N+1}$  and the proof is concluded. Thus, we have proved stability of the semidiscrete problem (5.3), but (unfortunately) the domain of the eigenvalues of the related matrix  $tG_N$  is proportional to  $N^2$  and not to  $N$  as in the scalar case. This large domain will evidently result in a severe stability condition

$$\Delta t = O\left(\frac{1}{N^2}\right).$$

Hence, for the system case there is no difference between using Chebyshev or Legendre polynomials.

## 7. NUMERICAL RESULTS

In this section we describe some numerical experiments whose results agree with the theory written in the previous sections. Throughout this section we use the following notations:

- $N$     number of gridpoints (resolution in space),
- $M$     degree of numerical evolution operator (resolution in time),
- $\lambda_i$     eigenvalue of the related matrix  $tG_N$ .

The approximation in space is done by using the pseudospectral projection operator. The collocating points are the zeroes of the  $N$ th degree polynomial.

Table I presents the difference in the size of the domain of the eigenvalues while using Chebyshev or Legendre polynomials. The matrix  $tG_N$  whose eigenvalues we have computed is related to the problem

$$\begin{aligned} U_t - U_x &= 0, \\ U(x, 0) &= U^0(x), \\ U(1, t) &= f(t). \end{aligned} \quad (7.1)$$

We have taken  $t$  equal to 1.  
For the inflow problem

$$\begin{aligned} U_t - xU_x &= 0, \\ U(x, 0) &= U^0(x), \\ U(-1, t) &= g(t); \quad U(1, t) = f(t), \end{aligned} \quad (7.2)$$



TABLE 1

<i>N</i>	Legendre	
	Chebyshev max $ \lambda_i $	max $ \lambda_i $
8	37.57	7.0
16	150.0	14.4
32	599.6	29.8

the results were almost the same as in the previous table. For the third model problem of outflow

$$\begin{aligned}
 U_t + xU_x &= 0, \\
 U(x, 0) &= U^0(x),
 \end{aligned}
 \tag{7.3}$$

there is no difference between Chebyshev and Legendre. In both cases the eigenvalues are negative real numbers of order *N*.

In Table II we compare the amount of work needed to achieve a certain degree of accuracy for Chebyshev and Legendre polynomials. The model problem is

$$\begin{aligned}
 U_t - U_x &= 0, \\
 U^0(x) &= e^{1/(x-1)}, \\
 U(1, t) &= 0.
 \end{aligned}
 \tag{7.4}$$

The time marching technique is a fourth order Runge-Kutta. The solution is computed at time level *T* = 1.0.

In Table III we carry out a similar comparison between Chebyshev and Legendre polynomials for the inflow problem

$$\begin{aligned}
 U_t - xU_x &= 0, \\
 U(x, 0) &= \exp(1/(x^2 - 1)), \\
 U(-1, t) &= U(1, t) = 0
 \end{aligned}
 \tag{7.5}$$

The next three tables are related to Section 4. The results presented here illustrate

TABLE II

<i>L</i> <sub>2</sub> error	Chebyshev		Legendre	
	<i>N</i>	<i>M</i>	<i>N</i>	<i>M</i>
$0.32 \times 10^{-2}$	16	240	16	48
$0.246 \times 10^{-3}$	32	960	32	100

TABLE III

$L_2$ error	Chebyshev		Legendre	
	$N$	$M$	$N$	$M$
$0.645 \times 10^{-1}$	16	240	16	32
$0.188 \times 10^{-1}$	32	960	32	64

the high accuracy of the OPS compared to the second order Modified Euler scheme. Legendre polynomials are used for space approximations. In Table IV, the model problem is

$$U_t - U_x = 0,$$

$$U^0(x) = (x - 1)^{10}.$$

The solution is computed at  $t = 1.0$ .

For the OPS,  $M$  was chosen such that the time error is of the same order as space error. This table shows clearly the overall spectral rate of convergence of the OPS. To compare the OPS to the Modified Euler from the point of view of the amount of work needed to achieve a certain degree of accuracy, one can use the fact that Modified Euler is second order in time. Thus, for  $N = 16$ , for example, an error of  $0.1660 \times 10^{-5}$  is achieved by the Modified Euler scheme when  $M$  satisfies

$$M \approx 80(0.1035/0.1680 \times 10^{-5})^{1/2} \approx 16000$$

compared to 36 for OPS.

The results in the next table are related to the inflow problem.

$$U_t - xU_x = 0,$$

$$U(x, 0) = (x^2 - 1)^3,$$

$$U(-1, 0) = U(1, 0) = 0.$$

The solution is computed at  $t = 1.0$ .

TABLE IV

$N$	Modified Euler		OPS	
	$M$	$L_2$ error	$M$	$L_2$ error
16	80	0.1035	36	$0.1660 \times 10^{-5}$
32	160	$0.2388 \times 10^{-1}$	72	$0.6836 \times 10^{-9}$
64	320	$0.5749 \times 10^{-2}$	144	$0.4247 \times 10^{-12}$

TABLE V

$N$	Modified Euler		OPS	
	$M$	$L_2$ error	$M$	$L_2$ error
16	26	$0.1785 \times 10^{-1}$	16	$0.1608 \times 10^{-1}$
32	52	$0.4265 \times 10^{-2}$	32	$0.1994 \times 10^{-2}$
64	104	$0.8230 \times 10^{-3}$	64	$0.2004 \times 10^{-3}$

Since the solution is very oscillatory, the advantage of using high order approximations (in space and time) is less significant than in the previous case.

Table describes the refinement procedure for the outflow problem.

$$U_t + xU_x = 0,$$

$$U(x, 0) = (x^2 - 1)^3.$$

The solution is computed at  $t = 1.0$ .

### 8. CONCLUSION

This paper has shown that it is possible to construct a pseudospectral method for initial boundary value scalar problems with stability condition

$$\Delta t = O\left(\frac{1}{N}\right)$$

rather than the familiar condition  $\Delta t = O(1/N^2)$ . This improvement in the stability condition does not hold for the case of a system of equations or even for a scalar parabolic equation. Still, the fact that using space discretization with  $\Delta x_{\min} = O(N^{-2})$  does not necessarily imply that  $\Delta t = O(N^{-2})$  [for a hyperbolic equation] gives us hope that there may be a way to overcome this drawback of using spectral methods for the numerical solution of nonperiodic problems.

TABLE VI

$N$	Modified Euler		OPS	
	$M$	$L_2$ error	$M$	$L_2$ error
16	20	$0.9378 \times 10^{-3}$	10	$0.8819 \times 10^{-3}$
32	40	$0.2463 \times 10^{-3}$	20	$0.2357 \times 10^{-5}$
64	80	$0.6261 \times 10^{-4}$	40	$0.2242 \times 10^{-10}$

## ACKNOWLEDGMENT

I would like to thank my advisor, Professor David Gottlieb, for many helpful discussions.

## REFERENCES

1. D. GOTTLIEB AND S. ORSZAG, *Numerical Analysis of Spectral Methods* (CBMS-NSF Regional Conferences Series in Applied Mathematics, SIAM Publisher, Philadelphia, 1977).
2. D. GOTTLIEB, *Math. Comput.* **36**, 107 (1981).
3. D. GOTTLIEB, N. GUNZBURGER AND E. TURKEL, *SIAM J. Numer. Anal.* **19**, 671 (1982).
4. H. TAL-EZER, to appear in *SIAM J. Numer. Anal.* **23** (1), (1986).
5. H. TAL-EZER, Thesis, Tel-Aviv University, 1984 (unpublished).
6. G. SZEGO, *Orthogonal Polynomials* (Amer. Math. Soc., New York, 1959).