I. STABILITY OF TCHEBYSHEV COLLOCATION

II. INTERPOLATION FOR SURFACES WITH 1-D DISCONTINUITIES

III. ON COMPOSITE MESHES

Thesis by
Luis Guillermo M. Reyna

in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California
1983
(submitted October 15th , 1982)

ACKNOWLEDGEMENTS

I am particularly thankful to my thesis advisor Professor Heinz-Otto Kreiss for his continuous help and support during the development of this thesis. Working with him has been most enjoyable.

I am grateful to Dr. Gerald Browning and to Dr. Akira Kasahara for providing access to the computing facilities at the National Center for Atmospheric Research at Boulder, Colorado.

I also thank my friends from the Applied Mathematics Department and from the rest of the Institute. Special thanks are due to Dave Brown for commenting on the manuscript of this thesis and to Charles Sobrero for clarifying my English.

Financial support was provided by Institute fellowships and teaching assistantships as well as research assistantships under Office of Naval Research contract no. N00014-80-C0076.

Finally, agradezco a mis padres por su continuo apoyo durante todos estos años; es a ellos a quienes dedico esta tesis.

THESIS ABSTRACT

I. Stability of Tchebyshev Collocation

We describe Tchebyshev collocation when applied to hyperbolic equations in one space dimension. We discuss previous stability results valid for scalar equations and study a procedure that when applied to a strictly hyperbolic system of equations leads to a stable numerical approximation in the L_2-norm. The method consists of using orthogonal projections in the L_2-norm to apply the boundary conditions and smooth the higher modes.

II. On 2-D Interpolation for Surfaces with 1-D Discontinuities

This problem arises in the context of shock calculations in two space dimensions. Given the set of parabolic equations describing the shock phenomena the method proceeds by discretising in time and then solving the resulting elliptic equation by splitting. The specific problem is to reconstruct a two dimensional function which is fully resolved along a few parallel horizontal lines. The interpolation proceeds by determining the position of any discontinuity and then interpolating parallel to it.

III. On Composite Meshes

We collect several numerical experiments designed to determine possible numerical artifacts produced by the overlapping region of composite meshes. We also study the numerical stability of the method when applied to hyperbolic equations. Finally we apply it to a model of a wind driven ocean circulation model in a circular basin. We use stretching in the angular and radial directions which allow the necessary resolution to be obtained along the boundary.

Table of Contents

Part II: Interpolation for Surfaces with 1-D Discontinuities

Part III: On Composite Meshes

Part I:

Stability of Tchebyshev Collocation

1. Introduction

In this part of this thesis we discuss the numerical method of Tchebyshev collocation when applied to a system of symmetric hyperbolic equations in one space dimension. It is well known that the method is not stable for general problems unless some care is taken with the higher modes. We are interested in developing procedures to stabilize the numerical method when applied to the following problem:

$$\frac{\partial \underline{u}}{\partial t} = \mathbf{A}(x,t)\ \frac{\partial \underline{u}}{\partial x} + \mathbf{B}(x,t)\ \underline{u} + \mathbf{C}(x,t)\ , \tag{1.1a}$$

defined for $-1 \leq x \leq 1$ and for $t \geq 0$; $\underline{u}(x,t)$ is an r-dimensional vector, $\mathbf{A}(x,t)$ is a symmetric matrix and $\mathbf{B}(x,t)$ is a general square matrix. We consider equation (1.1a) with the initial values given by

$$\underline{u}(x,t=0) = \underline{u}_0(x)\ . \tag{1.1b}$$

We will also require that $\mathbf{A}(x,t)$ be nonsingular at $x = \pm 1$ and $t \geq 0$.

In order to have a well posed problem we still have to provide boundary conditions for this equation. We know that the problem is well posed if the boundary conditions specify incoming variables in terms of outgoing variables. We define incoming and outgoing variables in the following way: we know there exists a smooth orthogonal transformation $\mathbf{T}_-(t)$ such that

$$\mathbf{T}_-^{-1}(t)\ \mathbf{A}(-1,t)\mathbf{T}_-(t) = \begin{bmatrix} \mathbf{A}_-^{I} & 0 \\ 0 & \mathbf{A}_-^{II} \end{bmatrix} \tag{1.1c}$$

where $\mathbf{A}_-^{I}$ is negative definite and $\mathbf{A}_-^{II}$ is positive definite. If we introduce $\underline{v}_-$ by

$$\underline{u}_-(-1,t) = \mathbf{T}_-(t)\ \underline{v}_-(t)\ ; \tag{1.1d}$$

and if we split $\underline{v}_-(t)$ according to equation (1.1c), that is

$\underline{v}_-(t)^t = (\underline{v}_-^I(t), \underline{v}_-^{II}(t))^t$, then $\underline{v}_-^I$ are the incoming variables and $\underline{v}_-^{II}$ are the outgoing variables. The type of boundary conditions that we will consider are of the form

$$\underline{v}_-^I(t) = S_- \underline{v}_-^{II}(t) + G_-(t) \tag{1.1e}$$

where G_- is a given vector and S_- is a rectangular matrix subject to the constraint

$$A_-^{II} + S_-^t A_-^I S_- > 0 \tag{1.1f}$$

for all $t > 0$. In a similar way we introduce $T_+(t)$ and $\underline{v}_+(t)$; the boundary conditions at $x=1$ are of the type

$$\underline{v}_+^I(t) = S_+ \underline{v}_+^{II}(t) + G_+(t) \tag{1.1g}$$

where $G_+(t)$ is a given vector and S_+ is a given rectangular matrix subject to a restriction similar to (1.1f).

Equation (1.1a) with the initial value given by (1.1b) and the boundary conditions given by (1.1c)-(1.1g) is well posed in the L_2-norm. The restriction (1.1f) is given so that the proof of well-posedness reduces to integration by parts. We study modifications of the numerical method for which we can also use integration by parts in order to obtain a numerical stability result. This idea was successfully used by Kreiss and Oliger [7] in the context of Fourier collocation.

Tchebyshev collocation, as with any spectral method, has a high rate of convergence which makes it particularly attractive for problems with smooth solutions. It is also useful in problems where the solution has sharp gradients in the vicinity of the boundaries of the interval and hence high resolution is needed. Tchebyshev collocation is also attractive because of its efficient implementation using the Fast Fourier Transform algorithm.

Gottlieb and Orszag in [4] discuss the convergence rate of Tchebyshev interpolation for smooth functions and for functions with boundary layers. They also discuss the convergence rate of Tchebyshev collocation when applied to scalar hyperbolic equations. In chapter 2 we collect some results concerning this interpolation procedure.

We are interested in a method which is convergent in the sense that if we increase our efforts (the number of grid points) then the numerical solution obtained gets closer to the real solution. There is a general result due to Lax and Richtmyer which says that if a numerical method is stable and consistent then it is convergent. Thus we will address the stability of the method.

The main source of difficulties in obtaining a stability result for this method is that problem (1.1) is in general not well posed in the Tchebyshev norm which is the natural norm to consider for this numerical method. We will discuss in chapter 3 the well-posedness of the continuous equations in different norms introduced by Gottlieb and Orszag [4]

Consider the system of parabolic equations

$$\frac{\partial \underline{u}}{\partial t} = \mathrm{A}(x,t)\ \frac{\partial \underline{u}}{\partial x} + \mathrm{B}(x,t)\ \underline{u} + \mathrm{C}(x,t) + \nu\, \mathrm{D}(x,t)\ \frac{\partial^2 \underline{u}}{\partial x^2}$$

where D is a symmetric positive definite matrix, called the viscosity matrix, with eigenvalues bounded away from zero for all x and t, and ν is a positive number. Unlike systems of hyperbolic equations, systems of parabolic equations are stable in the Tchebyshev norm (provided that the ν, the viscosity, is large enough). It is not unreasonable to expect that Tchebyshev collocation applied to this problem will lead to a convergent numerical method; in fact Canuto and Quarteroni [1] and also Gottlieb [3] proved that this is the case. The estimates obtained are not valid in the singular case when the ν is much smaller than the inverse of the number of points used in the numerical method. For these

problems it is important to have a stable approximation to the hyperbolic part of the problem.

Gottlieb in [3] considers the following type of scalar hyperbolic equations

$$\frac{\partial u}{\partial t} = a \frac{\partial u}{\partial x} + b\, u + c\ , \tag{1.2}$$

with appropriate initial value and boundary conditions. He proves that the Tchebyshev collocation method applied to the above equation is stable for all $t > 0$, in different norms related to Tchebyshev norm, either when $a(x,t)$ does not change sign inside the interval $[-1,1]$ or when it has a sign change at $-1 < s(t) < 1$ and $a(x,t)/(x-s(t))$ has no sign changes in the interval. In chapter 4 we introduce smoothing operators that ensure the stability of the previous scalar problem without these restrictions. These smoothing operators can be implemented with a minimal amount of extra computational effort; the boundary conditions are imposed in a way consistent with the smoothing operator. We describe these operators in chapter 7. We use different norms and stabilizing procedures for different cases of the scalar hyperbolic equation. These norms are not equivalent to each other when considered over the entire space of functions defined in the interval [-1,1]. Therefore it is not clear whether it is possible to draw either a stability or a weaker algebraic stability result for system (1.1a) from the stability of all the different cases of scalar hyperbolic equations. In chapter 6 we will show that all the norms we will consider are algebraically equivalent to each other when restricted to the finite dimensional space of the numerical solutions. In this way if we consider the above equation with homogeneous boundary conditions the following estimate will hold:

$$\| v_N(x,t) \|_2 \leq c(t)\, N^r\, N^{ts}\, \| v_N(x,0) \|_2\ , \tag{1.3}$$

where N is the number of grid points, $v_N(x,t)$ is the numerical solution, $c(t)$ is a

positive function which depends on the coefficients of the equation and r and s are real constants that could be positive.

In order to obtain a stability result we will use a smoothing operator that guarantees stability in the L_2-norm for general hyperbolic equations (this also immediately implies a stability result for systems of hyperbolic equations). If the coefficients of equation (1.1) are not smooth, the extra amount of work can be significant; the total amount of work nevertheless remains smaller than that required by spectral methods based on other special functions. The L_2-stabilizing operator consists in truncating the higher modes of the Legendre expansion of the solution. In chapter 4 we also discuss other types of stabilizing operators which lead to more accurate numerical approximations.

Finally chapter 5 contain lemmas necessary for the final stability results for this stabilized method which are described in chapter 8.

2. Tchebyshev Interpolation

In this chapter we introduce and develop some basic results about Tchebyshev interpolation.

Given a function $u(x) \in C[-1,1]$, introduce the periodic, even $C(R)$ function $\tilde{u}(\vartheta) = u(\cos(\vartheta))$. If we now expand $\tilde{u}(\vartheta)$ in its Fourier series

$$\tilde{u}(\vartheta) = \sum_{n=0}^{\infty} u_n \cos(n\vartheta), \tag{2.1}$$

the corresponding expansion for $u(x)$,

$$u(x) = \sum_{n=0}^{\infty} u_n \cos(n \arccos(x)) \tag{2.2}$$

is called the *Tchebyshev expansion* of $u(x)$. From the recursion formula

$$\tfrac{1}{2}\cos((n+1)\vartheta) + \tfrac{1}{2}\cos((n-1)\vartheta) = \cos(\vartheta)\cos(n\vartheta) \tag{2.3}$$

it follows that $T_n(x) = \cos(n \arccos(x))$ is actually a polynomial of degree n with leading order coefficient 2^{n-1} for $n > 0$; $T_n(x)$ is called the *n^{th}-Tchebyshev polynomial*.

The coefficients u_n of this expansion can be easily computed and are given by

$$u_n = \frac{2}{\pi}\int_0^{\pi} u(\cos(\vartheta))\cos(n\vartheta)\,d\vartheta = \frac{2}{\pi}\int_{-1}^{1} u(x)\,T_n(x)\,\frac{dx}{\sqrt{1-x^2}} \qquad n \geq 0 \tag{2.4a}$$

$$u_0 = \frac{1}{\pi}\int_0^{\pi} u(\cos(\vartheta))\,d\vartheta = \frac{1}{\pi}\int_{-1}^{1} u(x)\,\frac{dx}{\sqrt{1-x^2}} \tag{2.4b}$$

This suggests introducing the following norm for functions defined in the interval $[-1,1]$

$$\|u(x)\|_T^2 = \int_{-1}^{1} u^2(x)\,\frac{dx}{\sqrt{1-x^2}} \tag{2.5}$$

which will be called the *Tchebyshev norm* or *T–norm.* We denote the corresponding inner product by $(\cdot,\cdot)_T$.

From the recursion formula

$$\sin((n+1)\vartheta) + \sin((n-1)\vartheta) = 2\cos(n\vartheta)\sin(\vartheta) \tag{2.6}$$

it follows that

$$\frac{1}{n+1}\frac{d\,T_{n+1}(x)}{d\,x} + \frac{1}{n-1}\frac{d\,T_{n-1}(x)}{d\,x} = 2\,T_n(x)\,. \tag{2.7}$$

This recurrence relation leads to an efficient way to obtain the coefficients of the Tchebyshev expansion of the derivative of a function $u(x)$ in terms of the coefficients of the Tchebyshev expansion of $u(x)$.

We normalize the Tchebyshev polynomials with respect to this new norm obtaining

$$\tilde{T}_n(x) = c_n\,T_n(x) \qquad \textit{where } c_n = \begin{cases} \sqrt{2/\pi} & n>0 \\ \sqrt{1/\pi} & n=0 \end{cases}. \tag{2.8}$$

The coefficients of this expansion decay exponentially fast for smooth functions: we have that if $u(x)\,\varepsilon\,C^p[-1,1]$ then

$$|u_n| \le 2\frac{M}{n^p} \qquad \text{where} \quad M = \max_{0\le\vartheta\le\pi}\left|\frac{\partial^p u(\cos(\vartheta))}{\partial\vartheta^p}\right|. \tag{2.9}$$

It is this fast decay of the coefficients that makes spectral methods so appealing.

We now are ready to define Tchebyshev interpolation. Let N be a natural number and introduce the sequence of points

$$x_\nu = \cos(\vartheta_\nu) = \cos(\nu\frac{\pi}{N}) \qquad \nu=0,1,2\cdots N. \tag{2.10}$$

These are called the Tchebyshev points. For a function $u(x)\,\varepsilon\,\mathbf{C}[-1,1]$, $v_N(x)$, the *Tchebyshev interpolant* of $u(x)$, is defined as the only polynomial in $P_N[x]$ such that

$$v_N(x_\nu) = u(x_\nu) \qquad \nu=0,1,2\cdots N \tag{2.11}$$

where $P_N[x]$ is the vector space consisting of all polynomials of degree less or equal to N.

We introduce the following bilinear form in $\mathbf{C}[-1,1]$

$$(v(x),w(x))_h = \frac{\pi}{N}\left\{ \tfrac{1}{2}\, v(x_0)\cdot w(x_0) + \sum_{k=1}^{N-1} v(x_k)\cdot w(x_k) + \tfrac{1}{2}\, v(x_N)\cdot w(x_N) \right\} \tag{2.12}$$

Using

$$\sum_{k=-N}^{N} e^{ink\pi/N} = 2\sum_{k=1}^{N}\cos(n\,\vartheta_k) + 1 = \begin{cases} (-1)^n(2N+1) & n/2N \text{ integer} \\ (-1)^n & \text{otherwise} \end{cases} \tag{2.13}$$

it is easily seen that if we write $v_N(x) = \sum_{n=0}^{N} v_n\ \tilde{T}_n(x)$ then

$$v_n = \begin{cases} (u(x),\tilde{T}_k(x))_h & 0\le k < N \\ \tfrac{1}{2}\,(u(x),\tilde{T}_N(x))_h & k=N \end{cases} . \tag{2.14}$$

This also can be written in the following way (which is a well know result) :

Lemma 2.1 : *Given $u(x)$ and $v(x)$ in $P_N[x]$ such that the sum of their degrees is strictly less than $2N$, then*

$$\int_{-1}^{1} u(x)\,v(x)\,\frac{dx}{\sqrt{1-x^2}} = (u(x),v(x))_h\ .$$

We now want to relate the Tchebyshev coefficients of a given function $u(x)$ and the coefficients of its Tchebyshev interpolant $v_N(x)$:

Lemma 2.2 :

$$v_n = u_n + \sum_{l=1}^{\infty} (u_{2lN-n} + u_{2lN+n}) , \qquad 0 \le n < N ;$$

$$v_N = \sum_{l=0}^{\infty} u_{(2l+1)N} .$$

Proof: We have that

$$T_{2Nl \pm n}(x_\nu) = \cos((2Nl \pm n)\frac{\nu\pi}{N}) = \cos(n\frac{\nu\pi}{N}) = T_n(x_\nu)$$

and the lemma follows.

The previous lemma implies the following error formula for the interpolation:

$$\| u(x) - v_N(x) \|_T^2 \tag{2.15}$$

$$= \sum_{k=N+1}^{\infty} u_k^2 + \left\{ \sum_{k=0}^{N-1} \left[\sum_{l=1}^{\infty} (u_{2lN+n} + u_{2lN-n}) \right]^2 + \left[\sum_{l=1}^{\infty} u_{2lN+N} \right]^2 \right\} .$$

The first term on the right hand side is called the truncation error and the second term the aliasing error; both sums can be bounded in terms of the smoothness of the function $u(x)$:

Lemma 2.3 : *Given* $u(x) \in \mathbf{C}^{\alpha}[-1,1]$, *we have*

$$\| u(x) - v_N(x) \|_T \le \frac{2M\, D_\alpha}{N^\alpha}$$

where $D_\alpha^2 = 1 + 2\sum_{j=1}^{\infty} (1/(2j-1))^{2\alpha}$ *and* $M = \max_{0 \le \vartheta \le \pi} | \frac{\partial^p u(\cos(\vartheta))}{\partial \vartheta^p} |$.

Proof: Kreiss and Oliger [7].

The natural space of functions in which the Tchebyshev expansion can be defined is the $L_2([-1,1],T)$, that is the completion of $\mathbf{C}^{\infty}[-1,1]$ with respect to

the T–norm, and its corresponding Sobolev spaces with fractional indices. The previous lemma can be extended to cover these spaces (Quarteroni [11]).

3. Well-posedness of the Continuous Problem

In general terms we can say that when solving a differential equation, or some mathematical problem that is well-posed in some sense in the Tchebyshev norm (or T-norm), a spectral or pseudo-spectral method based on Tchebyshev expansion is stable provided some special attention is paid to the higher modes in the case of pseudo-spectral methods. If we are interested in solving hyperbolic equations, the main source of difficulties is that hyperbolic equations are not in general well-posed in the T-norm. In this chapter we want to describe some norms used in the literature to prove stability results for Tchebyshev collocation.

For simplicity we will only consider the following scalar hyperbolic equation

$$u_t = a(x)\,u_x \qquad -1 \le x \le 1 \tag{3.1}$$

with initial values given by: $u(x,t=0) = u_0(x)$ and appropriate homogeneous boundary conditions; where the coefficient $a(x)\ \varepsilon\ \mathbf{C}^1[-1,1]$. Using Duhammel's principle it is easily seen that the well-posedness results we obtain for equation (3.1) remain valid for equation (1.2), a more general class of scalar problems.

We say that equation (3.1) is well-posed, for the mixed initial-boundary value problem, in a norm $\|.\|$ when given $T_0 > 0$ there exist constants M and α such that

$$\|u(x,t)\| \le M\, e^{\alpha t}\ \|u(x,t=0)\| \tag{3.2}$$

for any $0 \le t \le T_0$ and any $u_0(x) = u(x,t=0)$.

The natural norm for hyperbolic equations is the L_2-norm; in fact integrating by parts we have the following result:

Proposition 3.1 : *The equation (3.1) is well posed in the L_2-norm and we have the following estimate*

$$\|u(x,t)\|_2 \leq e^{\alpha t} \|u(x,t=0)\|_2$$

where $\alpha = \frac{1}{2} \max_{-1 \leq x \leq 1} a_x(x)$.

The T-norm is not a natural norm for hyperbolic equations; it is possible to have a problem ill-posed in the T-norm while we know that it is always well-posed in the L_2-norm. More precisely we have the following result:

Proposition 3.2 : *The equation (3.1) is not well posed in the T-norm if there is an outflow condition at either end, that is if either* $a(-1) < 0$ *or* $a(1) > 0$.

Proof: We can restrict ourselves to consider the following problem

$$u_t + u_x = 0 \qquad u(-1,t)=0$$

with the set of initial conditions given by

$$u(x,t=0;\delta) = \begin{cases} 0 & x < \delta/2 \\ 1 & -\delta/2 \leq x \leq \delta/2 \\ 0 & x > \delta/2 \end{cases}$$

where $0 < \delta << 1$. For these initial values we have

$$\|u(x,t=0;\delta)\|_T^2 \sim \delta$$

and for the solution at $t_\delta = 1-\delta/2$ we have

$$\|u(x,t_\delta;\delta)\|_T^2 = \int_{x=1-\delta}^{1} \frac{dx}{\sqrt{1-x^2}} > \frac{\delta}{\sqrt{\delta(2-\delta)}}.$$

Therefore

$$\frac{\|u(x,t_\delta;\delta)\|_T^2}{\|u(x,t=0;\delta)\|_T^2} \gtrsim \frac{1}{(2\delta)^{1/2}} \xrightarrow[\delta \to 0]{} \infty$$

so the equation is not well posed in the sense of the previous definition. This same argument can be easily modified to the case of variable coefficients.

We have a completely different situation when there is only inflow at both ends of the interval $[-1,1]$; we will refer to this as *Case I* of the equation (3.1):

Proposition 3.3 : *The equation (3.1) is well posed in the T-norm when there is an inflow condition at both ends, that is,* $a(-1) \geq 0$ *and* $a(1) \leq 0$.

Proof: We first consider the case when $a(-1) > 0$ and $a(1) < 0$: there exists a constant $\delta > 0$ such that

$$a(x) > 0 \quad \text{for } x < -1+\delta \quad \text{and} \quad a(x) < 0 \quad \text{for } x > 1-\delta$$

Multiplying both sides of equation (3.1) by $u(x)$ and integrating over the interval we obtain

$$\frac{\partial}{\partial t} \int_{-1}^{1} u^2(x) \, \frac{dx}{\sqrt{1-x^2}} = -\int_{-1}^{1} a(x)\, u(x)\, u_x(x) \, \frac{dx}{\sqrt{1-x^2}} .$$

We notice first that for any $t > 0$ we have that $u^2(x,t) \equiv 0$ in neighborhoods of either end point of the interval ; therefore integrating by parts we obtain

$$\tfrac{1}{2} \frac{d}{dt} \| u(x,t) \|_f^2 = \tfrac{1}{2} \int_{-1}^{1} u^2(x)\, a_x(x) \, \frac{dx}{\sqrt{1-x^2}} + \tfrac{1}{2} \int_{-1}^{1} u^2(x)\, a(x)\, x \, \frac{dx}{(1-x^2)^{3/2}}$$

$$\leq \tfrac{1}{2} \max_{-1 \leq x \leq 1} (a_x(x)) \int_{-1}^{1} u^2(x) \, \frac{dx}{\sqrt{1-x^2}} + \tfrac{1}{2} \int_{-1+\delta}^{1-\delta} u^2(x)\, a(x)\, x \, \frac{dx}{(1-x^2)^{3/2}}$$

$$\leq \tfrac{1}{2} \left\{ \max_{-1 \leq x \leq 1} a_x(x) + \max_{-1+\delta \leq x \leq 1-\delta} \frac{a(x) \cdot x}{1-x^2} \right\} \| u(x,t) \|_f^2$$

and the proposition follows. Now if the boundary $x = 1$ is sub-characteristic (i.e. $a(1) = 0$) we write the equation (3.1) in the form

$$u_t(x,t) + (1-x)^{\beta} \, \tilde{a}(x) \, u_x(x,t) = 0$$

and using the same method it is possible to get a similar estimate for $\beta \geq \tfrac{1}{2}$.

Gottlieb and Orszag [4] have introduced two other type of norms that are appropriate when addressing the stability of Tchebyshev collocation:

$$\|u(x)\|_{T^+}^2 = \int_{-1}^{1} u^2(x)\,\sqrt{\frac{1-x}{1+x}}\,dx \quad , \quad \|u(x)\|_{T^-}^2 = \int_{-1}^{1} u^2(x)\,\sqrt{\frac{1+x}{1-x}}\,dx \qquad (3.3a)$$

and

$$\|u(x)\|_{T^0}^2 = \int_{-1}^{1} u^2(x)\,\sqrt{1-x^2}\,dx \qquad (3.3b)$$

We also have stability results in these norms. The following propositions can be proved in the same way as the previous ones:

Proposition 3.4 : *The equation (3.1) is well posed in the T^+-norm when outflow conditions are specified at $x=1$ and either inflow conditions are specified at $x=-1$ (Case II) or the boundary at $x=-1$ is subcharacteristic; a similar result holds in the T^--norm.*

Proposition 3.5 : *The equation (3.1) is well posed in the T^0-norm when outflow conditions are specified at $x=1$ and at $x=-1$ (Case III).*

Given a scalar hyperbolic equation, we will develop a suitable modification of Tchebyshev collocation in such a way that the method obtained will be stable in either the T-norm, the T^+-norm or the T^--norm depending on the boundary conditions. We will also develop a method which ensures stability of the different cases of scalar problems under a common natural norm, the L_2-norm. This last method can be easily generalized to the system of equations (1.1) with the boundary conditions we have already discussed.

4. Tchebyshev Collocation

We are now ready to describe Tchebyshev collocation when applied to a scalar hyperbolic equation. We consider the equation

$$\frac{\partial u}{\partial t}(x,t) = a(x)\,\frac{\partial u}{\partial x}(x,t) + f(x,t) \tag{4.1a}$$

with initial conditions

$$u(x,t=0) = u_0(x). \tag{4.1b}$$

For convenience we assume that the boundary conditions are homogeneous whenever it is necessary to specify them. (The boundary conditions can be made homogeneous by subtracting a function which satisfies the boundary conditions. This introduces a new forcing function in the original differential equation.) There are three cases with different combinations of conditions at the boundaries:

$$\begin{aligned}
&\textit{Case I} && \left[\, a(-1)<0 \text{ and } a(1)>0 \,\right] && u(-1,t) = u(1,t) = 0 \quad \text{for } t>0, \\
&\textit{Case II} && \left[\, a(-1)\geq 0 \text{ and } a(1)\leq 0 \,\right] && \textit{no boundary conditions}, \\
&\textit{Case III} && \left[\, a(-1)<0 \text{ and } a(1)\leq 0 \,\right] && u(-1,t) = 0 \quad \text{for } t>0.
\end{aligned} \tag{4.1c}$$

We assume that the initial value $u_0(x)$ satisfies the corresponding boundary condition.

Given N a natural number and the Tchebyshev collocation points $x_\nu = \cos(\vartheta_\nu)$, $\nu=0,1,\cdots N$, the solution of Tchebyshev collocation, $v_N(x,t)$, is a polynomial of degree less or equal to N uniquely defined by

$$\frac{\partial v_N}{\partial t}(x_\nu,t) = a(x_\nu)\,\frac{\partial v_N}{\partial x}(x_\nu,t) + f(x_\nu,t) \quad \textit{at} \quad \nu = 0,1,\cdots N\,. \tag{4.2}$$

In *Case I* the equations corresponding to $\nu = 0$ and $\nu = N$ are replaced by

$\frac{\partial v_N}{\partial t}(x_0,t) = 0$ and $\frac{\partial v_N}{\partial t}(x_N,t) = 0$; no changes are necessary in *Case II* and in *Case III* the equation corresponding to $\nu = N$ is replaced by $\frac{\partial v_N}{\partial t}(x_N,t) = 0$.

We want to describe the collocation method in terms of operators. In order to achieve this we first introduce two polynomials $\varphi_N^+(x)$ and $\varphi_N^-(x)$ defined by

$$\varphi_N^+(1) = 1 \quad \text{and} \quad \varphi_N^+(x_\nu) = 0 \quad \text{for} \quad \nu=1,2,\cdots N$$

$$\varphi_N^-(x) = \varphi_N^+(-x)\,. \tag{4.3}$$

We also introduce the bilinear transformation $*$ defined on $P_N[x]$ by

$$* : P_N[x] \times P_N[x] \to P_N[x]$$

$$(u * v)(x_\nu) = u(x_\nu)\cdot v(x_\nu) \quad \nu = 0,1,\cdots N\,. \tag{4.4}$$

Define $a_N(x)$ as the Tchebyshev interpolation over N points of $a(x)$. The method can now be described in the following way

$$\textit{Case I} \quad \frac{\partial v_N}{\partial t} = a_N * \frac{\partial v_N}{\partial x} + f_N$$

$$- \left[a_N * v_N(-1,.) + f_N(-1,.) \right] \varphi_N^- - \left[a_N * v_N(1,.) + f_N(1,.) \right] \varphi_N^+\,,$$

$$\textit{Case II} \quad \frac{\partial v_N}{\partial t} = a_N * \frac{\partial v_N}{\partial x} + f_N\,, \tag{4.5}$$

$$\textit{Case III} \quad \frac{\partial v_N}{\partial t} = a_N * \frac{\partial v_N}{\partial x} + f_N - \left[a_N * v_N(-1,.) + f_N(-1,.) \right] \varphi_N^-\,.$$

Consider a general inner product $(\cdot,\cdot)_s$ defined on $P_N[x]$. Given M an integer less than or equal to N define $\mathrm{B}_{M,s}^N$ as the orthogonal projection with respect to the inner product $(\cdot,\cdot)_s$ of the space $P_M[x]$ onto its subspace of polynomials which satisfy the boundary conditions given in (4.1c). To avoid

encumbering the notation, we will suppress the dependence on N of $\mathbf{B}_{M,s}^{N}$.

In particular if we consider the discrete inner product introduced in chapter 2

$$(v(x),w(x))_h = \frac{\pi}{N}\left\{ \tfrac{1}{2}v(x_0)\cdot w(x_0) + \sum_{k=1}^{N-1} (x_k)\cdot w(x_k) + \tfrac{1}{2}v(x_N)\cdot w(x_N) \right\} \quad (4.6)$$

and its corresponding projection $\mathbf{B}_{N,h}$ then all three cases can be described in the following common way:

$$\frac{\partial v_N}{\partial t} = \mathbf{B}_{N,h}\,(a_N * \frac{\partial v_N}{\partial x} + f_N)\,. \quad (4.7)$$

This is the form in which Tchebyshev collocation is usually applied to practical problems.

It is possible to define a similar numerical method using the projection $\mathbf{B}_{N,T}$, orthogonal with respect to the Tchebyshev inner product, onto the same subspace. If we expand the solution of either numerical method in a Tchebyshev series

$$v_N(x,t) = \sum_{n=0}^{N} v_n(t)\; T_n(x) \quad (4.8)$$

then the numerical method based on the T-orthogonal projection differs from the previous one only in the differential equation corresponding to the last mode, $v_N(t)$ of the solution $v_N(x,t)$. In both methods the number of operations involved in imposing the boundary conditions is negligible compared to the number of operations involved in computing the term $a(x) * \frac{\partial v_N}{\partial x}(x,t)$.

The second method applied to the wave equation

$$\frac{\partial u}{\partial t}(x,t) + \frac{\partial u}{\partial x}(x,t) = 0 \quad (4.9a)$$

$$u(-1,t) = 0 \quad \text{for } t > 0 \quad \text{and} \quad u(x,0) = u_0(x)\,, \tag{4.9b}$$

or more generally to equation (4.1a) when the coefficient $a(x)$ is a linear function of x, leads to a Galerkin type method; this will not always be true for general $a(x)$. We will call methods of the type (4.7) collocation methods because of the way the advection terms are approximated. We leave open the possibility that the boundary conditions might be imposed in various ways.

The methods described above are not stable in general. This is not surprising since Tchebyshev collocation can be reduced to Fourier collocation, and that method is known to be unstable unless a smoothing operator is introduced to control the cascade of energy into the higher modes. Consider for example the following problem defined in $[-1,1]$:

$$\frac{\partial u}{\partial t}(x,t) + (1-x^2)^{\alpha}\, a(x)\, \frac{\partial u}{\partial x}(x,t) = 0\,, \quad \alpha \geq \tfrac{1}{2} \tag{4.10a}$$

$$u(x,t=0) = u_0(x) \tag{4.10b}$$

where $u_0(x)$ is a 2-periodic function. Tchebyshev collocation for this problem is exactly Fourier collocation applied to

$$\frac{\partial \tilde{u}}{\partial \vartheta}(x,\vartheta) - \sin(\vartheta)^{2\alpha-1}\, a(\cos(\vartheta))\, \frac{\partial \tilde{u}}{\partial \vartheta}(\vartheta,t) = 0 \tag{4.11a}$$

$$\tilde{u}(\vartheta,t=0) = \tilde{u}_0(\cos(\vartheta)). \tag{4.11b}$$

This new problem will have coefficients as smooth as we want if we pick α large enough, but it is not stable for general $a(x)$. Fornberg [2] has a discussion on practical aspects of Fourier collocation for hyperbolic equations.

There are two types of smoothing operators. Kreiss and Oliger [6] suggest a smoothing which enforces a given decay rate on the spectrum keeping the phases of the computed higher modes. The main feature of their smoothing

operator is that it is a projection in the sense that, if it is squared, the operator is obtained again. Majda, McDonough and Osher [8] suggest multiplying the higher modes by a smooth sequence of weights that decay to zero. Once the numerical approximation has been stabilized, the particular choice of the smoothing operator makes no practical difference to the computed solution.

Having shown the need of smoothing we we will discuss the stability for the following type of numerical approximation:

$$\frac{\partial v_N}{\partial t} = \mathbf{B}_{(1-\gamma)N,s}\ \mathbf{C}_{(1-\gamma)N,s}\ \left(a_N * \frac{\partial v_N}{\partial x} + f_N \right) \tag{4.12a}$$

and initial value given by

$$\mathbf{B}_{(1-\gamma)N,s}\ \mathbf{C}_{(1-\gamma)N,s}\ \left(u_N(x,t=0) \right) \tag{4.12b}$$

where $\mathbf{C}_{(1-\gamma)N,s}$ is the orthogonal projection with respect to the inner product $(\cdot,\cdot)_s$ of the space $P_N[x]$ onto the space of $P_{(1-\gamma)N}[x]$ and $u_N(x,t=0)$ is the Tchebyshev interpolation over N points of $u(x,t=0) = u_0(x)$. (γ could depend on N subject to the constraint that $0 < \gamma \leq \gamma_0 < 1$ for all N.)

The consistency of these numerical methods follows from the consistency of Tchebyshev collocation and from the algebraic equivalence of the different norms. Since the solution of the differential equation satisfies the boundary conditions, it is possible to prove consistency.

We are going to consider three different inner products corresponding to the norms we have already considered, that is the T-norm, the T^+-norm and the L_2-norm. In this context we are going to prove the following stability result:

Theorem: *Let* $\|.\|_s$ *be a norm in which equation (4.1a) with boundary conditions defined in (4.1c) is stable for the initial value problem. That is for any* $T_0 > 0$ *there exist constants* M *and* α *such that*

$$\|u(x,t)\|_s \leq M\, e^{\alpha t}\, \|u(x,t=0)\|_s + M \int_0^t e^{\alpha(t-\tau)}\, \|f_N(\cdot,\tau)\|_s\, d\tau$$

for any $0 \leq t \leq T_0$. *Then the numerical approximation to the equation (4.1) and its boundary conditions defined in equation (4.12) satisfies the following estimate*

$$\|v_N(x,t)\|_s \leq \left\{ M\, e^{\alpha t} + K\,(\gamma N)^{-q}\,((1-\gamma)N)^{-r} \right\} \|v_N(x,t=0)\|_s$$

$$+ M \int_0^t e^{\alpha(t-\tau)}\, \|f_N(\cdot,\tau)\|_s\, d\tau$$

where K *is a constant which depends on* $a_N(x)$. *For sufficiently rapid decay of the coefficients of the Tchebyshev expansion of* $a_N(x)$ $r+q$ *is positive. In these cases the estimate of the numerical solution converges to the corresponding estimate of the differential equation as* $N \to \infty$.

We will find the exact form of the operators $\mathbf{B}_{(1-\gamma)N,s}$, $\mathbf{C}_{(1-\gamma)N,s}$ and the necessary bounds for the proof in the following chapters. We will also write down estimates of the different constants involved in the theorem.

A sketch of the proof will be useful to understand what sort of estimates we are looking for. We first multiply both sides of equation (4.12a) and integrate with the appropriate weight to get:

$$\tfrac{1}{2}\, \frac{d}{dt}\, \|v_N\|_s^2 = \left(v_N\,,\, \frac{\partial v_N}{\partial t} \right)_s = \left(v_N\,,\, \mathbf{B}_{(1-\gamma)N,s}\, \mathbf{C}_{(1-\gamma)N,s}\, \left(a_N * \frac{\partial v_N}{\partial x} + f_N \right) \right)_s$$

Since the initial data satisfy the boundary conditions it follows that the solution

satisfies them for all times. Therefore

$$\tfrac{1}{2}\frac{d}{dt}\|v_N\|_s^2 = \left(v_N \,,\, C_{(1-\gamma)N,s}\, a_N * \frac{\partial v_N}{\partial x} - (a_N * \frac{\partial v_N}{\partial x}) \right)_s$$

$$+ \left(v_N \,,\, (a_N * \frac{\partial v_N}{\partial x}) - a_N \frac{\partial v_N}{\partial x} \right)_s + \left(v_N \,,\, a_N \frac{\partial v_N}{\partial x} \right)_s + \left(v_N \,,\, f_N \right)_s .$$

The first term of the right hand side is zero since the operator $C_{(1-\gamma)N,s}$ is orthogonal; the second term can be bounded in terms of the smoothness of $a(x)$; the bound for the last two terms corresponds to the usual estimate for the continuous problem.

5. Some Useful Estimates

We know that the operator of differentiation $D = \frac{\partial}{\partial x}$ is unbounded over the space of once differentiable functions; the same operator restricted to $P_N[x]$ is bounded but the constant involved depends on N. The distance between collocation points is $O(\frac{1}{N})$ in the interior of the interval $[-1,1]$ and $O(\frac{1}{N^2})$ near the end points; it is natural to expect the norm of the operator D to be proportional to the inverse of the minimum distance between collocation points: that is, the norm of D should increase like N^2 as $N \to \infty$. In the following lemma we show this result using the T-norm. It is possible to prove a similar result using the L_2-norm although we do not include this proof (which involves some manipulation of Legendre polynomials).

Lemma 5.1 : *For any $v(x)$ in $P_N[x]$, we have*

$$\| \frac{\partial v}{\partial x}(x) \|_T^2 \leq C_1 N^4 \| v(x) \|_T^2$$

where C_1 is a positive constant independent of N.

Proof: Given $v(x) = \sum_{n=0}^{N} v_n T_n(x)$, we have that

$$\frac{\partial v}{\partial x}(\cos(\vartheta)) = \sum_{n=0}^{N} n v_n \frac{\sin(n\vartheta)}{\sin(\vartheta)} .$$

It is easy to check that

$$\frac{\sin(n\vartheta)}{\sin(\vartheta)} = \sum_{k=0}^{n} \gamma_{n,k} \cos(k\vartheta)$$

where

$$\gamma_{n,k} = \begin{cases} 2 & k+n \text{ odd and } k>0 \\ 1 & n \text{ odd and } k=0 \\ 0 & k+n \text{ even} \end{cases} .$$

This result is just a different form of the recursion formula (2.7) for the derivatives of Tchebyshev polynomials. It follows that

$$\frac{\partial v}{\partial x}(\cos(\vartheta)) = \sum_{k=0}^{N-1} \left\{ \sum_{n=k+1}^{N} n\, v_n\, \gamma_{n,k} \right\} T_k(x) .$$

Now we are ready to bound the norm of $D[v]$:

$$\| \frac{\partial v}{\partial x}(x) \|_T^2 = \sum_{k=1}^{N-1} 2\pi \left\{ \sum_{\substack{n=k+1 \\ n+k \text{ odd}}}^{N} n\, v_n \right\}^2 + \pi \left\{ \sum_{\substack{n=1 \\ n \text{ odd}}}^{N} n\, v_n \right\}^2$$

$$\leq 4 \left\{ \sum_{n=1}^{N} v_n^2\, \frac{\pi}{2} \right\} \sum_{k=0}^{N-1} \left\{ \sum_{\substack{n=k+1 \\ n+k \text{ odd}}}^{N} n^2 \right\}$$

The second factor can be estimated

$$S = \sum_{k=0}^{N-1} \left\{ \sum_{\substack{n=k+1 \\ n+k \text{ odd}}}^{N-1} n^2 \right\} \sim \tfrac{1}{2} \sum_{k=0}^{N-1} \left\{ \sum_{n=k+1}^{N} n^2 \right\} \sim \frac{1}{8} N^4 \qquad as\ N \to \infty .$$

Therefore there exists a constant $C_1 \geq \frac{1}{2}$ such that

$$\| \frac{\partial v}{\partial x}(x) \|_T^2 \leq C_1 N^4 \left\{ \sum_{n=1}^{N} v_n^2\, \frac{\pi}{2} \right\} \leq C_1 N^4 \| v(x) \|_T^2$$

which is the statement of this lemma.

Remarks: (*i*) Given $\varepsilon > 0$ then the constant C_1 can be taken to be equal to $\frac{1}{2} + \varepsilon$ for N large enough (depending on epsilon).

(*ii*) The result of the lemma is sharp: consider $v(x) = \sum_{n=1}^{N} T_n(x)$. We have that

$$\| \frac{\partial v}{\partial x}(x) \|_T^2 = \pi \left\{ \sum_{\substack{n=1 \\ n \text{ odd}}}^{N} n^2 \right\} + 2\pi \sum_{k=1}^{N-1} \left\{ \sum_{\substack{n=k+1 \\ n+k \text{ odd}}}^{N} n \right\}^2 \sim \frac{\pi}{2}\, \frac{4}{15} N^5 \qquad as\ N \to \infty ,$$

and

$$\| v(x) \|_T^2 = \frac{\pi}{2} N$$

therefore

$$\frac{\| \frac{\partial v}{\partial x}(x) \|_T^2}{\| v(x) \|_T^2} \leq \frac{4}{15} N^4 .$$

The sharpness of this result can also be obtained using $v(x) = \frac{\partial T_N}{\partial x}(x)$, which is zero at all interior collocation points and equal to N^2 at the end points; this function is the natural guess for maximizing the above quotient.

We are ready now for the main lemma. We want to find a bound for the aliasing error using the rapid decay of the Tchebyshev coefficients of $a(x)$ and assuming that the last γN coefficients of $v(x)$ are zero; in this way the aliasing error only involves coefficients of $a(x)$ with index greater than γN.

Lemma 5.2 : *For any* $v(x) = \sum_{n=0}^{(1-\gamma)N} v_n \, T_n(x)$ *we have that*

$$\| (a*v)(x) - a(x).v(x) \|_T^2 \leq C_2 \, (1-\gamma)N \, \| a_2(x) \|_T^2 \, \| v(x) \|_T^2$$

where $a(x) = \sum_{n=0}^{N} a_n \, T_n(x) = \sum_{n=0}^{\gamma N} a_n \, T_n(x) + \sum_{n>\gamma N}^{N} a_n \, T_n(x) = a_1(x) + a_2(x)$ *and* C_2 *is a constant independent of* N.

Proof: For convenience introduce

$$R_a[v](x) = (a*v)(x) - a(x).v(x) .$$

From the relation $\cos(n\vartheta)\cos(m\vartheta) = \frac{1}{2}\cos((n-m)\vartheta) + \frac{1}{2}\cos((n+m)\vartheta)$, it follows that

$$T_n(x).T_m(x) = \tfrac{1}{2} T_{|n-m|}(x) + \tfrac{1}{2} T_{n+m}(x),$$

and therefore

$$T_n(x) * T_m(x) = \tfrac{1}{2} T_{|n-m|}(x) + \tfrac{1}{2} \begin{cases} T_{n+m}(x) & n+m \le N \\ T_{2N-(n+m)}(x) & n+m > N \end{cases}.$$

The norm of the aliasing error is

$$\| R_a[v] \|_T^2 = \| \sum_{\substack{n,m=0 \\ n+m>N}}^{N} a_n v_m \left(\tfrac{1}{2} T_{2N-(n+m)}(x) - \tfrac{1}{2} T_{n+m}(x) \right) \|_T^2$$

$$= \frac{1}{4} \| \sum_{\substack{n,m=0 \\ n+m>N}}^{N} a_n v_m T_{2N-(n+m)}(x) \|_T^2 + \frac{1}{4} \| \sum_{\substack{n,m=0 \\ n+m>N}}^{N} a_n v_m T_{n+m}(x) \|_T^2 .$$

Using that $(T_k(x), T_l(x))_T = 0$ if $k \neq l$ and $= \frac{\pi}{2}$ when $k = l > 0$, and Cauchy's inequality, we obtain

$$\| R_a[v] \|_T^2 = \tfrac{1}{2} \sum_{l=N+1}^{(2-\gamma)N} \left\{ \sum_{k=l-(1-\gamma)N}^{N} a_k v_{l-k} \right\}^2 \frac{\pi}{2} \le \frac{\pi}{4} \sum_{l=N+1}^{(2-\gamma)N} \left\{ \sum_{k=l-(1-\gamma)N}^{N} a_k^2 \right\} \left\{ \sum_{k=l-N}^{N} v_{l-k}^2 \right\}$$

$$\le \frac{(1-\gamma)N}{\pi} \left\{ \sum_{k>\gamma N}^{N} a_k^2 \frac{\pi}{2} \right\} \left\{ \sum_{k=1}^{(1-\gamma)N} v_k^2 \frac{\pi}{2} \right\} \le \frac{(1-\gamma)N}{\pi} \| a_2(x) \|_T^2 \| v(x) \|_T^2$$

which is the result of this lemma.

Remarks: (*i*) C_2 can be taken equal to $\frac{1}{\pi}$.

(*ii*) γ can depend on N, as long as $0 \le \gamma \le \gamma_0 < 1$ as $N \to \infty$.

(*iii*) The result of the lemma is sharp. To see this, consider $a(x) = v(x) = \sum_{n=1}^{N_1=(1-\gamma)N} T_n(x)$. We compute the aliasing error:

$$\| R_a[v](x) \|_T^2 = \tfrac{1}{2} \sum_{l=N+1}^{2N_1} (N+N_1-l+1)^2 \frac{\pi}{2} \sim \tfrac{1}{2} \frac{\pi}{2} \left\{ \frac{1}{3} ((1-\gamma)N)^3 - \frac{1}{3} (\gamma N)^3 \right\}$$

and the norm of $a(x)$

$$\| a(x) \|_f^2 = \| v(x) \|_f^2 = \frac{\pi}{2} (1-\gamma) N .$$

Hence

$$\frac{\| R_a[v](x) \|_f^2}{\| a(x) \|_f^2 \, \| v(x) \|_f^2} \sim \frac{N}{3\pi} \left\{ \frac{(1-\gamma)^3 - \gamma^3}{(1-\gamma)^2} \right\} \leq (1-\gamma) \frac{N}{3\pi}$$

for $N \to \infty$.

6. Equivalence of the Different Norms

We know that any two norms, $\|.\|_1$ and $\|.\|_2$, on a finite dimensional vector space V, are equivalent in the sense that there exist two positive constants C and D such that for any $v \in V$ the following inequalities hold

$$C \|v\|_1 \leq \|v\|_2 \leq D \|v\|_1 . \tag{6.1}$$

If we restrict any of the norms introduced in previous chapters and the T-norm to the space $P_N[x]$ then the constants C and D, involved in the definition of equivalence for these norms, depend on N. Since norms are not equivalent to each other over the space $\mathbf{C}[-1,1]$, we expect a deterioration of the constants involved as N, the dimension of the space, tends to infinity. We will show that this deterioration is at most algebraic for the different cases; we will also write down the coefficients of the corresponding inner products when the Tchebyshev polynomials are used as a basis for the space $P_N[x]$.

Specifically, given $v(x) = \sum_{n=0}^{N} v_n T_n(x)$ and $w(x) = \sum_{m=0}^{N} w_m T_m(x)$ then

$$(v(x),w(x))_2 = \int_{-1}^{1} v(x)\, w(x)\, dx = V^t A^2 W \tag{6.2}$$

where $V^t = (v_0,v_1,\cdots,v_N)$, $W^t = (w_0,w_1,\cdots,w_N)$ and $A^2 = [A^2_{i,j}]$ has the following coefficients:

$$A^2_{i,j} = \int_{-1}^{1} T_i(x)\, T_j(x)\, dx = \int_{0}^{\pi} \cos(i\vartheta)\cos(j\vartheta)\sin(\vartheta)\, d\vartheta$$

$$= \begin{cases} \dfrac{1}{1-(i-j)^2} + \dfrac{1}{1+(i+j)^2} & i+j \ \ even \\ 0 & i+j \ \ odd \end{cases} . \tag{6.3}$$

We will need this matrix when we implement the boundary conditions.

For the equivalence of the L_2-norm and the T-norm we have the following result:

Lemma 6.1 : *Given $v(x)$ in $P_N[x]$, the following inequalities hold*

$$\frac{C_3}{N} \| v \|_T^2 \leq \| v \|_2^2 \leq \| v \|_T^2$$

where C_3 is a constant which does not depend on N.

Proof: The second inequality follows immediately from the definition of each norm. In order to prove the first inequality we introduce two bases for $P_N[x]$, each one orthonormal in a different norm.

$$\tilde{P}_n(x) = (n+\tfrac{1}{2})^{\frac{1}{2}} P_n(x)$$

$$\tilde{T}_n(x) = c_n \; T_n(x)$$

where $P_n(x)$ is the n^{th}-Legendre polynomial and $T_n(x)$ is the n^{th}-Tchebyshev polynomial. We introduce now the matrix corresponding to the change of basis:

$$\tilde{P}_n(x) = \sum_{m=0}^{n} \tilde{h}_{m,n} \; \tilde{T}_n(x)$$

Using the generating function for the Legendre polynomials

$$(1 - 2s\cos\vartheta + s^2)^{-\frac{1}{2}} = \sum_{m=0}^{\infty} s^m \, P_m(\cos\vartheta)$$

it is possible to obtain an analytic formula for the coefficients $\tilde{h}_{m,n}$ (Whittaker and Watson [13]):

$$\tilde{h}_{n-2m,n} = 2 \, \frac{d_{n-2m}}{c_{n-2m}} (n+\tfrac{1}{2})^{\frac{1}{2}} \, \frac{(2m-1)!!}{m! \, 2^m} \, \frac{[2(n-m)-1]!!}{(n-m)! \, 2^{n-m}} \qquad n \geq 0 \, , \, n-2m \geq 0$$

$$\tilde{h}_{n-2m-1,n} = 0 \qquad n \geq 0 \, , \, n-2m-1 \geq 0$$

here

$$d_k = \begin{cases} 1 & \text{if } k > 0 \\ \frac{1}{2} & \text{if } k = 0 \end{cases}$$

and $m!! = m\ (m-2)\ (m-4)\ \cdots$ 5.3.1 with the usual convention $(-1)!!=1$. For convenience we also define

$$\tilde{h}_{m,n} = 0 \qquad m > n\ .$$

Now given $v(x) = \sum_{n=0}^{N} v_n\ \tilde{P}_n(x)$ we can compute the different norms and reduce the lemma to an algebraic problem. For the L_2-norm we have

$$\| v \|_2^2 = \int_{-1}^{1} v^2(x)\ dx = \sum_{n,m=0}^{N} v_n\ v_m \int_{-1}^{1} \tilde{P}_n(x)\ \tilde{P}_m(x)\ dx$$

$$= \sum_{n,m=0}^{N} v_n\ v_m\ \delta_{n,m} = \sum_{n=0}^{N} v_n^2 = V^t\ V$$

where $V^t = (v_0, v_1, \ldots, v_N)$; for the Tchebyshev norm,

$$\| v \|_T^2 = \int_{-1}^{1} v^2(x)\ \frac{dx}{\sqrt{1-x^2}} = \sum_{n,m=0}^{N} v_n\ v_m \int_{-1}^{1} \tilde{T}_m(x)\ \tilde{T}_m(x)\ \frac{dx}{\sqrt{1-x^2}}$$

$$= \sum_{n,m,k=0}^{N} v_n\ v_m\ \tilde{h}_{k,n}\ \tilde{h}_{k,m} = V^t\ \tilde{H}^t\ \tilde{H}\ V$$

where $\tilde{H}$ is the $N+1$ square matrix with entries $\tilde{h}_{n,m}$, for $0 \le n, m \le N$. Thus we can compute

$$\max_{v(x) \neq 0} \frac{\| v \|_T^2}{\| v \|_2^2} = \max_{V \neq 0} \frac{V^t\ \tilde{H}^t\ \tilde{H}\ V}{V^t\ V} = \lambda_{\max}(\tilde{H}^t \tilde{H}).$$

where $\lambda_{\max}(\tilde{H}^t \tilde{H})$ is the maximum eigenvalue of $\tilde{H}^t \tilde{H}$. To obtain the result in the lemma we must now find an upper bound for this eigenvalue. We know that

$$\lambda_{\max}(\tilde{H}^t\tilde{H}) \le \|\tilde{H}^t\tilde{H}\| \le \|\tilde{H}^t\| \, \|\tilde{H}\|$$

where $\| \ \|$ is any matrix norm induced by a vector norm. In particular if we consider the maximum vector norm we obtain

$$\lambda_{\max}(\tilde{H}^t\tilde{H}) \le \max_{0\le n\le N}\left\{\sum_m \tilde{h}_{m,n}\right\} \max_{0\le n\le N}\left\{\sum_m \tilde{h}_{n,m}\right\}.$$

That is, $\lambda_{\max}$ is bounded by the product of the maximum row sum and the maximum column sum of $\tilde{H}$. (No absolute values are necessary since $\tilde{h}_{n,m} \ge 0$ for all n,m.)

First we estimate the maximum column sum. We have

$$\tilde{P}_n(1) = (n+\tfrac{1}{2})^{\frac{1}{2}} \quad \text{and} \quad \tilde{T}_n(1) = c_n$$

therefore

$$(n+\tfrac{1}{2})^{\frac{1}{2}} = (c_1-c_0)\,\tilde{h}_{0,n} + c_1 \sum_{m=0}^{N} \tilde{h}_{m,n}$$

and hence

$$\sum_{m=0}^{N} \tilde{h}_{m,n} = \sqrt{\frac{\pi}{2}}\,(n+\tfrac{1}{2})^{\frac{1}{2}} + \left(\frac{\sqrt{2}}{2}-1\right)\tilde{h}_{0,n}\,.$$

Using Stirling's asymptotic formula for the factorial

$$n! = (2\pi n)^{\frac{1}{2}}\,(n/e)^n\,\left(1+O\left(\frac{1}{n}\right)\right)$$

we obtain

$$\tilde{h}_{0,2n} = \sqrt{\pi}\,(2n+\tfrac{1}{2})^{\frac{1}{2}}\left\{\frac{(2n-1)!!}{n!\,2^n}\right\}^2 = \frac{1}{(\pi\,(2n+\frac{1}{2}))^{\frac{1}{2}}}\left(1+O\left(\frac{1}{n}\right)\right)$$

and $\tilde{h}_{0,2n+1} = 0$ for $n \ge 0$.

Hence we obtain a bound for the growth of the maximum column sum:

$$\max_n \sum_{m=0}^{N} \tilde{h}_{m,n} = \sqrt{\frac{\pi}{2}}\,(N+\tfrac{1}{2})^{\frac{1}{2}}\,(1+O(\frac{1}{N}))$$

Now we estimate the maximum row sum. First we notice that the coefficients of the matrix $\tilde{H}$ when appropriately weighted decrease in size as n and m simultaneously increase:

$$\frac{\tilde{h}_{n+1-2m,n+1}}{\tilde{h}_{n-2m,n}}\;\frac{c_{n+1-2m}/d_{n+1-2m}}{c_{n-2m}/d_{n-2m}}$$

$$= \frac{(n+3/2)^{\frac{1}{2}}}{(n+\frac{1}{2})^{\frac{1}{2}}}\;\frac{2(n-m)+1}{2(n-m+1)} \le \left\{\frac{n+3/2}{n+1/2}\right\}^{\frac{1}{2}}\;\frac{2n+1}{2n+2} \le 1$$

therefore

$$\max_{0\le n\le N} \sum_{m=0}^{N} \tilde{h}_{n,m} = \frac{d_1/c_1}{d_0/c_0} \sum_{m=0}^{N} \tilde{h}_{0,m}\ .$$

Using again the asymptotic behaviour of $\tilde{h}_{0,m}$ for large m , we obtain

$$\sum_{m=0}^{N} \tilde{h}_{0,m} = \frac{1}{\sqrt{\pi}}\,(N+\tfrac{1}{2})^{\frac{1}{2}}\,(1+O(\frac{1}{N}))\ ,$$

thus

$$\max_{0\le n\le N} \sum_{m=0}^{N} \tilde{h}_{n,m} = \sqrt{\frac{2}{\pi}}\,(N+\tfrac{1}{2})\,(1+O(\frac{1}{N}))$$

and so finally,

$$\lambda_{\max}\,(\tilde{H}^t\tilde{H}) \le\ (N+\tfrac{1}{2})\,(1+O(\frac{1}{N}))$$

and the desired result follows.

Remarks: (*i*) Given $\varepsilon > 0$, the constant C_3 can be taken equal to $1-\varepsilon$ for N large enough.

(*ii*) We were not able to show the first inequality of the lemma to be sharp. In Table 1 we show the result of the numerical calculation corresponding to the maximum eigenvalue of $\tilde{H}^t \tilde{H}$ for different values of N. Instead of generating $\tilde{H}^t \tilde{H}$ we introduce the matrix $\tilde{A} = (\tilde{H}\tilde{H}^t)^{-1}$ whose coefficients are related to the coefficients of A^2 by :

$$\tilde{A}_{i,j} = c_i \, c_j \, A^2_{i,j}$$

The eigenvalues were computed using the inverse power method using $\lambda_{min}(\tilde{A})=0$ as initial guess. The numerical computations suggest the following asymptotic result:

$$\lambda_{max}(\tilde{H}\tilde{H}^t) = 0.863\, N \left(1+\frac{.95}{N} +o\left(\frac{1}{N}\right) \right) \qquad as\ N\to\infty$$

which has to be compared with the bound obtained in the lemma:

$$\lambda_{max}\,(\tilde{H}^t \tilde{H}) \le N\,(1+O(\frac{1}{N}))\,.$$

We now show a result equivalent to lemma (6.1) for the T^+-norm and for the T^0-norm.

We first compute the matrix $A^{T^+} = [A^{T^+}_{i,j}]$ corresponding to the inner product associated to the T^+-norm. From the definition of the coefficients

$$A^{T^+}_{i,j} = \int_{-1}^{1} T_i(x)\; T_j(x)\; (1-x)\; \frac{dx}{\sqrt{1-x^2}} \tag{6.4}$$

we obtain

$$A^{T^+} = \pi \begin{bmatrix} 1 & -\frac{1}{2} & 0 & . & 0 \\ -\frac{1}{2} & \frac{1}{2} & -\frac{1}{4} & 0 & . \\ 0 & -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} & . \\ . & . & . & . & . \\ . & 0 & -\frac{1}{4} & \frac{1}{2} & -\frac{1}{4} \\ 0 & . & 0 & -\frac{1}{4} & \frac{1}{2} \end{bmatrix} \tag{6.5}$$

Using this matrix we can prove the following result:

Lemma 6.2 : *Given $v(x)$ in $P_N[x]$, the following inequalities hold*

$$\frac{C_4}{N^2} \| v \|_T^2 \leq \| v \|_{T^+}^2 \leq 2 \| v \|_T^2$$

where C_4 is a constant which does not depend on N.

Proof: The second inequality follows from the definition of each norm. In order to prove this lemma we have to compute

$$\min_{v(x)\neq 0} \frac{\| v(x) \|_T^2}{\| v(x) \|_{T^+}^2} = \min_{V\neq 0} \frac{V^t \, A^{T^+} \, V}{\pi \left(v_0^2 + \frac{1}{2} \sum_{n=1}^{N} v_n^2 \right)} = \lambda_{\min}(B)$$

where $\lambda_{\min}(B)$ is the minimum eigenvalue of the following symmetric matrix

$$B = \begin{bmatrix} 1 & -\sqrt{2}/2 & 0 & . & 0 \\ -\sqrt{2}/2 & 1 & -\frac{1}{2} & 0 & . \\ 0 & -\frac{1}{2} & 1 & -\frac{1}{2} & . \\ . & . & . & . & . \\ . & 0 & -\frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & . & 0 & -\frac{1}{2} & 1 \end{bmatrix}.$$

If we introduce the diagonal matrix $T=[t_{i,j}]$ defined by

$$t_{n,n} = \begin{cases} \sqrt{2}/2 & n=0 \\ 1 & n\neq 0 \end{cases}$$

then

$$\lambda_{\min}(B) \geq \frac{\lambda_{\min}(TBT)}{\left[\lambda_{\max}(T) \right]^2}.$$

Using standard methods for banded matrix with constant coefficients along each diagonal it is easy to prove that

$$\lambda_{\min}(TBT) = 1 - \cos(\frac{2\pi}{2N+3});$$

from which it follows that

$$\lambda_{\min}(B) \gtrsim \frac{\pi^2}{8N^2}.$$

which shows the result of this lemma.

Remarks: (*i*) The estimate is sharp.

(*ii*) Given $\varepsilon > 0$, the constant C_4 can be taken equal to $\pi^2/8 - \varepsilon$ for N large enough.

Finally we obtain a similar result for the T^0-norm. Using again the Tchebyshev polynomials as a basis for $P_N[x]$ we can find $A^{T^0} = [A^{T^0}_{i,j}]$, the matrix of the inner product corresponding to the T^0-norm :

$$A^{T^0} = \frac{\pi}{2}\begin{bmatrix} 1 & 0 & -\frac{1}{2} & 0 & . & . & . \\ 0 & -\frac{1}{4} & 0 & -\frac{1}{4} & . & . & . \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & -\frac{1}{4} & 0 & . \\ 0 & -\frac{1}{4} & 0 & \frac{1}{2} & 0 & -\frac{1}{4} & 0 \\ . & . & . & . & . & . & . \\ . & . & . & -\frac{1}{4} & 0 & \frac{1}{2} & 0 \\ 0 & . & . & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} \end{bmatrix}. \qquad (6.6)$$

We have a similar result for the equivalence of the T^0-norm and the T-norm:

Lemma 6.3 : *Given $v(x)$ in $P_N[x]$, the following inequalities hold*

$$\frac{C_5}{N^2} \| v \|_T^2 \leq \| v \|_{T^0}^2 \leq \| v \|_T^2$$

where C_5 is a constant which does not depend on N.

Proof: We notice that Tchebyshev polynomials of even degree are T^0-orthogonal to Tchebyshev polynomials of odd degree. If we permute rows and columns putting first the coefficients corresponding to odd polynomials followed by the coefficients corresponding to even polynomials then the lemma is reduced to finding the minimum eigenvalues of two tridiagonal matrices with similar, and almost constant, coefficients. The proof follows as in the previous lemma.

Remarks: (*i*) The estimate is sharp.

(*ii*) Given $\varepsilon > 0$, the constant C_5 can be taken equal to $\pi^2/4 - \varepsilon$ for N large enough.

7. Projection Operators

We want now to describe the projection operators $\mathbf{C}_{M,s}$ and $\mathbf{B}_{M,s}$ in the various cases corresponding to different inner products. The first operator projects $P_N[x]$ onto $P_M[x]$ (where $M \leq N$) and in this way can be considered a chopping or smoothing operator. The second operator projects $P_M[x]$ onto the subspace of polynomials which satisfy the boundary conditions given in (4.1c).

We start with $\mathbf{C}_{M,2}$: we know that the Legendre polynomials form an L_2-orthogonal basis of $P_N[x]$; this implies that the L_2-orthogonal projection of $u(x)$ in $P_N[x]$ onto the subspace $P_M[x]$ amounts to finding the Legendre expansion of $u(x)$ and then eliminating the last $N-M$ terms. We want to determine the matrix corresponding to this projection when the Tchebyshev polynomials are taken as a basis for $P_N[x]$. Since $P_M[x]$ is the set of all polynomials of degree less than or equal to M, the projection does not modify the first M terms of the Tchebyshev expansion of $u(x)$. The matrix that transforms Tchebyshev expansions to Legendre expansions was already introduced in lemma 6.1 : if we write

$$P_n(x) = \sum_{m=0}^{n} h_{m,n}\, T_m(x) \,, \tag{7.1}$$

it follows from the proof of that lemma that

$$h_{n-2m,n} = 2\, d_{n-2m}\, \frac{(2m-1)!!}{m!\, 2^m}\, \frac{[2(n-m)-1]!!}{(n-m)!\, 2^{n-m}} \qquad m \geq 0 \,,\ n-2m \geq 0 \,, \tag{7.2a}$$

$$h_{n-2m-1,n} = 0 \qquad m \geq 0 \,,\ n-2m-1 \geq 0 \qquad \text{and} \qquad h_{n,m} = 0 \qquad n > m \,. \tag{7.2b}$$

Introduce $H^{-1} = [h^{-1}_{n,m}]$, the inverse of the matrix H. Now we can find the projection operator $\mathbf{C}_{M,2}$:

$$\mathbf{C}_{M,2}\,[T_m] = \mathbf{C}_{M,2}[\sum_{n=0}^{m} h^{-1}_{n,m}\, P_n\,] = \sum_{n=0}^{M} h^{-1}_{n,m}\, P_n \tag{7.3a}$$

$$= \sum_{l=0}^{M} \left[\sum_{n=l}^{M} h_{l,n}\, h_{n,m}^{-1} \right] T_l \qquad \text{for } m > M,$$

and

$$\mathbf{C}_{M,2}[T_m] = T_m \qquad \text{for } m \leq M. \tag{7.3b}$$

The matrix of this projection operator is a full matrix, therefore the number of operations involved in applying this operator is $(N-M)\times M$. This can be a costly operation depending on the size of $N-M$. Our final stability result includes an estimate of $N-M$ in terms of the smoothness of the coefficients of the equation. In practice the actual size of this difference is determined empirically and for most problems can be taken to be equal to a small integer. We want to point out that for our proof there is an actual need for this projection since

$$\| \mathbf{C}_{M,2} - \mathbf{C}_{M,T} \|_2 = \mathbf{O}(1)\ ; \tag{7.4}$$

if, on the other hand

$$\| \mathbf{C}_{M,2} - \mathbf{C}_{M,T} \|_2 = \mathbf{O}(M^{-\alpha})\ ,$$

with $\alpha > 2$ were to hold, then there would be no need to use $\mathbf{C}_{M,2}$ instead of $\mathbf{C}_{M,T}$ (which is much simpler to apply).

We now want to describe the boundary condition operator $\mathbf{B}_{M,2}$. There are two cases for this operator: (*i*) the solution should vanish at one end of the interval, say at $x=-1$, and (*ii*) the solution should vanish at both ends. In the first case the operator $\mathbf{B}_{M,2}$ can be written in the following way:

$$\mathbf{B}_{M,2}\,[u](x) = u(x) - (u(y), \varphi_{-}^{2}(y))_2 \ \ \varphi_{-}^{2}(x) \tag{7.5}$$

where the polynomial $\varphi^2_{\pm}(x) \in P_M[x]$ is of unity L_2-norm, and L_2-orthogonal to any polynomial which vanishes at $x=-1$. If we write $u(x) = \sum_{n=0}^{N} u_n T_n(x)$ and introduce $U^t = (u_0, u_1, \cdots, u_N)$ and $\Psi^t_{\pm} = (1, -1, \cdots, \pm 1)$, then it follows that $u(-1) = U^t \Psi_-$. From the definition of $\mathbf{B}_{M,2}$ it follows that

$$(u(x), \varphi^2_{\pm}(x))_2 \, (\Psi^t_{\pm} \Phi^2_{\pm}) = (U^t A^2 \Phi^2_{\pm}) (\Psi^t_{\pm} \Phi^2_{\pm}) = \Psi^t_{\pm} U \tag{7.6}$$

where A^2 is the matrix of the L_2-inner product and $\Phi^2_{\pm}$ is the vector containing the Tchebyshev coefficients of $\varphi^2_{\pm}(x)$. The previous equation has to hold for any $u(x)$; hence,

$$(\Psi^t_{\pm} \Phi^2_{\pm}) (A^2 \Phi^2_{\pm}) = \Psi_- , \tag{7.7}$$

which determines $\Phi^2_{\pm}$ up to a normalization constant. In this case the final form of the operator is :

$$\mathbf{B}_{M,2}[u](x) = u(x) - u(-1) \, \beta^2_{\pm}(x) \tag{7.8}$$

where $\beta^2_{\pm}(x) = \varphi^2_{\pm}(x) \, / \, \varphi^2_{\pm}(-1)$.

If the boundary condition were imposed at the other end of the interval, that is $x=1$, from the symmetry of the problem it follows that

$$\mathbf{B}_{M,2}[u](x) = u(x) - u(1) \, \beta^2_{+}(x) \tag{7.9}$$

where $\beta^2_{+}(x) = \beta^2_{\pm}(-x)$.

Finally if the boundary conditions were imposed at both ends, it is easy to see that the operator can be written as

$$\mathbf{B}_{M,2}[u](x) = u(x) - u(1) \, \beta^2_{0,+}(x) - u(-1) \, \beta^2_{0,-}(x) \tag{7.10a}$$

where

$$\beta^2_{0,+}(x) = \frac{\beta^2_+(x) - \beta^2_+(-1)\,\beta^2_-(x)}{1 - \beta^2_-(-1)\,\beta^2_-(1)} \quad \text{and} \quad \beta^2_{0,-}(x) = \beta^2_{0,+}(-x)\,. \qquad (7.10b)$$

This completes the discussion of the boundary condition operator L_2-orthogonal for scalar equations.

It is also possible to find the corresponding smoothing projection and boundary condition operator for the system of equations (1.1a) and its boundary conditions defined by (1.1c)-(1.1g) . These projections have to be orthogonal with respect to the natural extension of the L_2-norm to vectors consisting of functions; that is if $\underline{u}(x)$ and $\underline{v}(x)$ belong to $(C[-1,1])^r$ then

$$(\underline{u}(x) , \underline{v}(x))_2 = \int_{-1}^{1} \underline{u}^t(x)\, \underline{v}(x)\, dx\,. \qquad (7.11)$$

For simplicity we assume the boundary values $G_\pm(t)$ to be zero for all $t \geq 0$ (it is always possible to transform the system (1.1a) into this form by a smooth affine transformation).

Using this inner product, the smoothing projection for mapping a vector of polynomials $\underline{v}(x) \in (P_N[x])^r$ onto $(P_M[x])^r$ is obtained by simply applying the smoothing operator we have already found for the scalar problem to each of the components of $\underline{v}(x)$.

We now want to discuss the boundary condition operator: the polynomial $\underline{u}(x)$ needs $r\times(M+1)$ coefficients to be determined; if S_- is a $r_-\times(M-r_-)$ matrix and S_+ is a $r_+\times(M-r_+)$ matrix then the subspace of vectors that satisfy the boundary conditions is of codimension $(r_-+r_+)\times(M+1)$. From this it is easily seen that the evaluation of this projection involves $O(\,M\times(r_++r_-))$ operations.

We describe now the operators $C_{M,T}$ and $B_{M,t}$ corresponding to the T-norm. In this case it is clear that

$$\mathbf{C}_{M,T}[u](x) = \sum_{m=0}^{M} u_m \, T_m(x) \, . \tag{7.12}$$

We will use the T-norm only for the pure outflow condition in which case the operator $\mathbf{B}_{M,T}$ reduces to the identity operator.

For the $T^{\pm}$ and T^0 norms the computation of the projection operators is simpler than in the L_2 case. This is due to the fact that the matrices corresponding to the remaining inner products are banded matrices.

Recall that the matrix corresponding to the T^+ inner product is a tridiagonal matrix. This implies that:

$$(T_n(x), T_m(x)) = 0 \quad \text{for } n \geq M+2 \text{ and } m \leq M \, , \tag{7.13}$$

therefore

$$\mathbf{C}_{M,T^+}[u](x) = \mathbf{C}_{M,T^+}\left[\sum_{n=0}^{N} u_n \, T_n \right] = \sum_{m=0}^{M} u_m \, T_m(x) + u_{M+1} \, \mathbf{C}_{M,T^+}[T_{M+1}](x) \tag{7.14}$$

where $\mathbf{C}_{M,T^+}[T_{M+1}](x)$ must be determined. If we write $\mathbf{C}_{M,T^+}[T_{M+1}](x) = \sum_{m=0}^{M} c_m^+ \, T_m(x)$, then since the projection is T^+-orthogonal, it follows that

$$(\mathbf{C}_{M,T^+}[T_{M+1}] \, , \, T_l \,)_{T^+} = \sum_{m=0}^{M} c_m^+ \, (\, T_m \, , \, T_l \,)_{T^+} = \sum_{m=0}^{M} c_m^+ \, A_{m,l}^{T^+} \tag{7.15}$$

$$= (T_{M+1}, T_l)_{T^+} = a_{M+1,l}^{T^+} \quad \text{for } l \leq M.$$

This equation can be written as

$$A^{T^+} \, C_+ = R^{T^+} \, , \tag{7.16}$$

where $C_+ = (c_0^+, c_1^+, \cdots, c_M^+)^t$ and $R^{T^+} = (a_{M+1,1} \, , \, a_{M+1,2}, \cdots, \, a_{M+1,M})^t$. In this way

finding the smoothing operator has been reduced to solving a tridiagonal linear system of equations.

As before, we write the boundary condition operator in the following way

$$\mathbf{B}_{M,T^+}[u](x) = u(x) - u(-1)\,\beta^+(x) \tag{7.17}$$

where $\beta^+(x)$ has to be T^+-orthogonal to any polynomial in $P_M[x]$ which vanishes at $x=-1$. If we write $\beta^+(x) = \sum_{m=0}^{M} b_m^+\, T_m(x)$ and introduce $\beta_+^t = (b_0^+, \cdots, b_M^+)$ then the coefficients of $\beta^+(x)$ are determined by

$$A^{T^+}\,\beta_+ = (\,\beta_+^t\, A^{T^+}\,\beta_+\,)\,\Psi_-\,. \tag{7.18}$$

To obtain the operators for the T^0-norm recall that

$$(\,T_l\,,\,T_m\,)_{T^0} = 0 \quad \text{for } l+m \text{ odd} \quad \text{or} \quad |l-m| > 2\,. \tag{7.19}$$

This implies that the projection operator can be written as

$$\mathbf{C}_{M,T^0}[u](x) = \sum_{m=0}^{M} u_m T_m + u_{M+1}\,\mathbf{C}_{M,T^0}[T_{M+1}](x) + u_{M+2}\,\mathbf{C}_{M,T^0}[T_{M+2}](x). \tag{7.20}$$

The coefficients of the last two polynomials can be found by solving tridiagonal systems of linear equations.

In a similar way the boundary condition operator can be written as

$$\mathbf{B}_{M,T^0}[u](x) = u(x) - u(-1)\,\beta_-^0(x) - u(\,1)\,\beta_+^0(x)\,, \tag{7.21}$$

where $\beta_+^0(-1) = 0$,$\beta_+^0(\,1) = 1$ and, from the symmetry of the problem, $\beta_+^0(x) = \beta_-^0(-x)$.

If we introduce $\beta_\pm^0$, the vectors of the Tchebyshev coefficients of $\beta_\pm^0(x)$, then these coefficients are determined by:

$$A^{T^0}\,\beta_+^0 = (\beta_+^0\,A^{T^0}\,\beta_-^0)\,\Psi_- + (\beta_+^0\,A^{T^0}\,\beta_+^0)\,\Psi_+ \qquad (7.22a)$$

$$A^{T^0}\,\beta_-^0 = (\beta_-^0\,A^{T^0}\,\beta_-^0)\,\Psi_- + (\beta_-^0\,A^{T^0}\,\beta_+^0)\,\Psi_+ , \qquad (7.22b)$$

where $\Psi_\pm^t = (1,1,\cdots,1)$. As usual this system of equations can be reduced to tri-diagonal system of linear equations.

8.1 Stability Results in the L_2-Norm

We are now ready to prove our stability theorems. Consider the following scalar one dimensional hyperbolic equation:

$$\frac{\partial u}{\partial t}(x,t) = a(x)\,\frac{\partial u}{\partial x}(x,t) + b(x,t)\,u(x,t) + f(x,t) \tag{8.1a}$$

with appropriate homogeneous boundary conditions and initial value given by

$$u(x,0) = g(x)\,. \tag{8.1b}$$

Introduce N, the number of collocation points, and $M=(1-\gamma)N$ where $0<\gamma\leq\gamma_0<1$. Given $a_N(x)$, the Tchebyshev interpolation of $a(x)$ in $P_N[x]$, define $a_{N,1}(x)$ and $a_{N,2}(x)$ in the following way:

$$a_N(x) = \sum_{n=0}^{N} \tilde{a}_{N,n}\; T_n(x)$$

$$= \sum_{n=0}^{\gamma N} \tilde{a}_{N,n}\; T_n(x) + \sum_{n>\gamma N}^{N} \tilde{a}_{N,n}\; T_n(x) = a_{N,1}(x) + a_{N,2}(x)\,; \tag{8.2}$$

in a similar way define $b_N(x,t)$, $b_{N,1}(x,t)$, $b_{N,2}(x,t)$, $f_N(x,t)$ and $g_N(x)$.

We define the *s-stabilized Tchebyshev collocation method*, when applied to equation (8.1), as follows:

$$\frac{\partial v_N}{\partial t}(x,t) = \mathbf{B}_{M,s}\;\mathbf{C}_{M,s}\left[\, a_N(x) * \frac{\partial v_N}{\partial x} + b_N(x,t) * v_N(x,t) + f_N(x,t)\,\right] \tag{8.3a}$$

with initial values given by

$$v_N(x,t=0) = \mathbf{B}_{M,s}\;\mathbf{C}_{M,s}\left[\, g_N(x)\,\right] \tag{8.3b}$$

and where s denotes any of the previously considered inner products.

We are ready now for our stability theorems:

Theorem 8.1 : *The L_2-stabilized Tchebyshev collocation method applied to equation (8.1) satisfies the following estimate :*

$$\frac{d}{dt}\,\|v_N(\cdot,t)\|_2 \le \left\{ \max_{-1\le x\le 1}\left[b_N(x,t) - \tfrac{1}{2}\,\frac{da_N}{dx}(x) \right] \right\} \|v_N(\cdot,t)\|_2$$

$$+ \left\{ C_6\,[(1-\gamma)N]^3\,\|a_{N,2}\|_T + C_7[(1-\gamma)N]\,\|b_{N,2}(\cdot,t)\|_T \right\} \|v_N(\cdot,t)\|_2$$

$$+ \|f_N(\cdot,t)\|_2 ,$$

where C_6 and C_7 are constants which do not depend on either N or γ.

Proof: We notice that $v_N(x,t)$, the solution of the numerical approximation to equation (8.1), satisfies

$$v_N(x,t) = \mathbf{B}_{M,2}\,\mathbf{C}_{M,2}\left[v_N(x,t) \right] \tag{8.4}$$

for all $t \ge 0$. This implies that

$$\left(v_N(\cdot,t)\,,\ \frac{\partial v_N}{\partial t}(\cdot,t) \right)_2 = \left(v_N(\cdot,t)\,,\ a_N * \frac{\partial v_N}{\partial x}(\cdot,t) + b_N(\cdot,t) * v_N(\cdot,t) + f_N(\cdot,t) \right)_2 \tag{8.5}$$

Now we are going to estimate each of the terms in the right hand side of this equation. We start with the last one:

(*i*) Schwartz's inequality implies

$$|\,(v_N(\cdot,t)\,,\,f_N(\cdot,t))_2\,| \le \|v_N(\cdot,t)\|_2\,\|f_N(\cdot,t)\|_2 \tag{8.6}$$

(*ii*) Using Schwartz's inequality it follows that

$$\Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) * v_N(\cdot,t) \Big)_2 =$$

$$\Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) \cdot v_N(\cdot,t) \Big)_2 + \Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) * v_N(\cdot,t) - b_N(\cdot,t) \cdot v_N(\cdot,t) \Big)_2$$

$$\leq \Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) \cdot v_N(\cdot,t) \Big)_2 + \| b_N(\cdot,t) * v_N(\cdot,t) - b_N(\cdot,t) \cdot v_N(\cdot,t) \|_2 \ \| v_N(\cdot,t) \|_2 \ .$$

From lemma (5.2) we have that

$$\| b_N(\cdot,t) * v_N(\cdot,t) - b_N(\cdot,t \ \cdot v_N(\cdot,t) \|_2 \ \leq \ C_2^{\frac{1}{2}} \ [(1-\gamma)N]^{\frac{1}{2}} \ \| b_{N,2}(\cdot,t) \|_T \ \| v_N(\cdot,t) \|_T \ .$$

and using now the equivalence between the L_2-norm and the T-norm we obtain

$$\Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) * v_N(\cdot,t) \Big)_2 \qquad (8.7)$$

$$\leq \Big(v_N(\cdot,t)\ ,\ b_N(\cdot,t) \cdot v_N(\cdot,t) \Big)_2 + \left[\frac{C_2}{C_3} \right]^{\frac{1}{2}} [(1-\gamma)N]^{\frac{1}{2}} \ \| b_{N,2}(\cdot,t) \|_T \ \| v_N(\cdot,t) \|_2 \ .$$

(*iii*) We find the bound corresponding to the higher order term of the equation; using again Schwartz's inequality we obtain

$$\left(v_N(\cdot,t)\ ,\ a_N * \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2 \qquad (8.8)$$

$$= \left(v_N(\cdot,t)\ ,\ a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2 + \left(v_N(\cdot,t)\ ,\ a_N * \frac{\partial v_N}{\partial x}(\cdot,t) \ v_N(\cdot,t) - a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2$$

$$\leq \left(v_N(\cdot,t)\ ,\ a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2 + \| a_N * \frac{\partial v_N}{\partial x}(\cdot,t) - a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \|_2 \ \| v_N(\cdot,t) \|_2$$

Using the results from lemmas (6.1), (5.2) and (5.1) we obtain

$$\| a_N * \frac{\partial v_N}{\partial x}(\cdot,t) - a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \|_2 \leq \| a_N * \frac{\partial v_N}{\partial x}(\cdot,t) - a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \|_T \qquad (8.9)$$

$$\leq C_2^{1/2} [(1-\gamma)N]^{1/2} \| a_{N,2} \|_T \| \frac{\partial v_N}{\partial x}(\cdot,t) \|_T \leq [C_2 C_1]^{1/2} [(1-\gamma)N]^{5/2} \| a_{N,2} \|_T \| v_N(\cdot,t) \|_T$$

$$\leq \left[\frac{C_2 C_1}{C_3} \right]^{1/2} [(1-\gamma)N]^3 \| a_{N,2} \|_T \| v_N(\cdot,t) \|_2 .$$

Integrating by parts we obtain

$$\left(v_N(\cdot,t) , a_N \cdot \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2 \leq -\frac{1}{2} \left(v_N(\cdot,t) , \frac{da_N}{dx} v_N(\cdot,t) \right)_2 . \qquad (8.10)$$

From equations (8.8)-(8.10) it follows that

$$\left(v_N(\cdot,t) , a_N * \frac{\partial v_N}{\partial x}(\cdot,t) \right)_2 \qquad (8.11)$$

$$\leq -\frac{1}{2} \left(v_N(\cdot,t) , \frac{da_N}{dx} v_N(\cdot,t) \right)_2 + \left[\frac{C_2 C_1}{C_3} \right]^{1/2} [(1-\gamma)N]^3 \| a_{N,2} \|_T \| v_N(\cdot,t) \|_2 .$$

(iv) Finally we notice that given any two functions $w(x)$ and $c(x)$ defined in $[-1,1]$ the definition of the L_2-norm implies that

$$\left(w(x) , c(x) \cdot w(x) \right)_2 \leq \left\{ \max_{-1 \leq x \leq 1} c(x) \right\} \| w(x) \|_2^2 . \qquad (8.12)$$

The theorem then follows from equations (8.5), (8.6), (8.7), (8.11) and (8.12).

Remarks: *(i)* A similar proof can be used to show that the L_2-stabilized Tchebyshev collocation applied to system (1.1) produces a stable numerical approximation. The first step in the proof is to find a smooth transformation which diagonalizes the matrix $\mathbf{A}(x,t)$ for all x and t, this reduces the problem to a scalar problem for which we have all the necessary estimates.

(*ii*) Given $\varepsilon > 0$ then C_6 can be taken greater than $\frac{1}{2\pi} + \varepsilon$ and C_7 can be taken greater than $\frac{1}{\pi} + \varepsilon$ for N large enough (depending on ε).

(*iii*) The theorem is a stability result assuming that $a(\cdot)$ and $b(\cdot,t)$ belong to Sobolev spaces with high enough indices; that is, we assume that if we expand these functions in Tchebyshev series

$$a(x) = \sum_{n=0}^{\infty} \tilde{a}_n \, T_n(x) \tag{8.13a}$$

$$b(x,t) = \sum_{m=0}^{\infty} \tilde{b}_m(t) \, T_m(x) \, , \tag{8.13b}$$

then there exist constants $\tilde{M}_a$ and $\tilde{M}_b = \tilde{M}_b(t)$ such that for $n,m > 0$ the following inequalities hold:

$$| \tilde{a}_n | \leq \frac{2 \tilde{M}_a}{n^p} \quad \text{and} \quad | \tilde{b}_m(t) | \leq \frac{2 \tilde{M}_b(t)}{m^q} \, , \tag{8.14}$$

where $p \, , q > \frac{1}{2}$. Using the result from lemma (2.3) we also obtain that

$$| a_n | \leq \frac{2 M_a}{n^p} \quad \text{and} \quad | b_m(t) | \leq \frac{2 M_b(t)}{m^q} \, . \tag{8.15}$$

where $M_a \leq \tilde{M}_a \, (1+2D_\alpha)$. Using Schwartz's inequality it follows that

$$\| a_{N,2} \|_T \leq \pi \, M_a \left\{ \sum_{n>\gamma N}^{N} n^{-2p} \right\}^{\frac{1}{2}} \leq \frac{\pi \, M_a \, K_p}{[\gamma N]^{p-\frac{1}{2}}} \tag{8.16a}$$

where $K_p \leq \sqrt{\frac{2p}{2p-1}}$. In a similar way we obtain

$$\| b_{N,2}(\cdot,t) \|_T \leq \frac{\pi \, M_b(t) \, K_q}{[\gamma N]^{q-\frac{1}{2}}} \, . \tag{8.16b}$$

We now rewrite the previous theorem showing that the smoother the functions $a_N(x)$ and $b_N(x,t)$ are, the less smoothing is necessary in the numerical procedure in order to guarantee numerical stability:

Corollary: *The L_2-stabilized Tchebyshev collocation method applied to equation (8.1) satisfies the following estimate :*

$$\frac{d}{dt}\|v_N(\cdot,t)\|_2 \le \left\{ \max_{-1\le x\le 1}\left[b_N(x,t) - \tfrac{1}{2}\frac{da_N}{dx}(x)\right]\right\}\|v_N(\cdot,t)\|_2$$

$$+\left\{ C_6\,\pi M_a K_p\,\frac{[(1-\gamma)N]^3}{[\gamma N]^{p-\frac{1}{2}}} + C_7\,\pi M_b(t) K_q\,\frac{[(1-\gamma)N]}{[\gamma N]^{q-\frac{1}{2}}}\right\}\|v_N(\cdot,t)\|_2 + \|f_N(\cdot,t)\|_2 .$$

If we take $\gamma = \alpha\, N^{-s}$, where $\alpha > 0$ and $0 \le s < 1$ are constants independent of N, then for

$$p > \frac{3}{1-s} + \tfrac{1}{2} \ge 7/2 \quad \textit{and} \quad q > \frac{1}{1-s} + \tfrac{1}{2} \ge 3/2$$

this estimate converges to the corresponding estimate for equation (8.1) as $N \to \infty$.

For comparison we now write a particular case of a similar theorem proved by Kreiss and Oliger [7] in the case of Fourier collocation (the original theorem allows a more general type of smoothing operator):

Theorem: *The solution of stabilized Fourier collocation applied to equation (8.1a) with periodic boundary conditions satisfies the following estimate*

$$\frac{d}{dt}\|v_N\|_2 \le \left\{ \max_{0\le x\le 1}\left[b_N(x,t) - \tfrac{1}{2}\frac{da_N}{dx}\right] + \frac{3\,M_a}{(2\pi)^{p-1}}\,\frac{N^2}{[\gamma N]^p}\right\}\|v_N\|_2 + \|f_N(\cdot,t)\|_2 ,$$

If $p > 2$ then the estimate of this theorem converges to the corresponding estimate for the continuous problem as $N \to \infty$.

8.2 Stability Results in the Remaining Norms

We have similar stability theorems for the remaining norms. In all of these cases there is no need to use the smoothness of $b(x,t)$ in order to get an estimate for the lower order term $b(x,t) * v_N(x)$ of the numerical approximation (8.3a).

Given $b(x)$ in $P_N[x]$ we define the following bilinear form

$$Q_{N,b,s}[v,w](x) = (v(x), b(x) * w(x))_s \,, \tag{8.17}$$

where $v(x)$ and $w(x)$ are in $P_M[x]$ ($M < N$) and $(\cdot,\cdot)_s$ is any of the T-related inner products. The following lemma provides such estimate for the remaining norms:

Lemma 8.2 : *The following inequalities hold:*

$$Q_{N,b,s}[v,w](x) \le \left\{ \max_{-1\le x\le 1} |b(x)| \right\} \|v(x)\|_s \, \|w(x)\|_s$$

$$\textit{where} \begin{cases} M \le N-1 & \text{for } s = T \\ M \le N-2 & \text{for } s = T^+ \\ M \le N-3 & \text{for } s = T^0 \end{cases}.$$

Proof: These estimates are a direct consequence of lemma (2.1) :

$$(v(x), b(x) * w(x))_T = \int_{-1}^{1} v(x)\, b(x) * w(x)\, \frac{dx}{\sqrt{1-x^2}}$$

$$= \frac{\pi}{2} \left[\tfrac{1}{2} v(x_0) b(x_0) w(x_0) + \sum_{n=1}^{N-1} v(x_n) b(x_n) w(x_n) + \tfrac{1}{2} v(x_N) b(x_N) w(x_N) \right].$$

$$\le \left\{ \max_{-1\le x\le 1} |b(x)| \right\} \|v(x)\|_T \, \|w(x)\|_T \,.$$

The results for the T^+-norm and the T^0-norm follow in a similar way.

Using the results from Lemmas (5.1), (5.2) and (8.2), and the technique introduced in the proof of theorem (8.1), it is possible to obtain the following stability theorem for the T-norm:

Theorem 8.3 : *The T-stabilized Tchebyshev collocation applied to Case I of equation (8.1) (that is with inflow conditions at both $x=-1$ and $x=1$) satisfies the following estimate :*

$$\frac{d}{dt}\,\|v_N(\cdot,t)\|_T \le \left\{ C_6\,[(1-\gamma)N]^{5/2}\,\|a_{N,2}\|_T \right\} \|v_N(\cdot,t)\|_T + \|f_N(\cdot,t)\|_T$$

$$+ \left\{ \max_{0\le x\le N}\left(b_N(x,t) - \tfrac{1}{2}\,\frac{da_N}{dx}(x) \right) + \max_{-1+\delta_a\le x\le 1-\delta_a}\left[\frac{-a(x)\cdot x}{1-x^2} \right] \right\} \|v_N(\cdot,t)\|_T \,.$$

where $\delta_a > 0$ depends on the behaviour of $a(x)$ at the boundaries.

If we again assume that there exists a constant $\tilde{M}_a$ such that that the Tchebyshev coefficients of $a(x)$ satisfy

$$|\,\tilde{a}_n\,| \le \frac{2\,\tilde{M}_a}{n^p}\,, \tag{8.18}$$

for $n > 0$, we obtain the result :

Corollary: *The following estimate holds :*

$$\frac{d}{dt}\,\|v_N(\cdot,t)\|_T \le C_6\,\pi M_a K_p\,\frac{[(1-\gamma)N]^{5/2}}{[\gamma N]^{p-\frac{1}{2}}}\,\|v_N(\cdot,t)\|_T + \|f_N(\cdot,t)\|_T$$

$$+ \left\{ \max_{0\le x\le N}\left(b_N(x,t) - \tfrac{1}{2}\,\frac{da_N}{dx}(x) \right) + \max_{-1+\delta_a\le x\le 1-\delta_a}\left[\frac{-a(x)\cdot x}{1-x^2} \right] \right\} \|v_N(\cdot,t)\|_T$$

If we take $\gamma = \alpha\, N^{-s}$, where $\alpha > 0$ and $0 \le s < 1$ are constants independent of N, then for

$$p > \frac{5/2}{1-s} + \tfrac{1}{2} \geq 3 ,$$

this estimate converges to the corresponding estimate for equation (8.1) as $N \to \infty$.

Using the results from Lemmas (5.1), (5.2), (6.2) and (8.2) it is possible to obtain a corresponding stability theorem for the T^+-norm :

Theorem 8.4 : *The T^+-stabilized Tchebyshev collocation applied to Case II of equation (8.1) (that is with inflow condition at $x=-1$ and outflow condition at $x=1$) satisfies the following estimate :*

$$\frac{d}{dt} \| v_N(\cdot,t) \|_{T^+} \leq \left\{ \frac{4C_6}{\pi} [(1-\gamma)N]^{7/2} \| a_{N,2} \|_T \right\} \| v_N(\cdot,t) \|_{T^+} + \| f_N(\cdot,t) \|_{T^+}$$

$$+ \left\{ \max_{0 \leq x \leq N} \left| b_N(x,t) - \tfrac{1}{2} \frac{da_N}{dx}(x) \right| + \max_{-1+\delta_a \leq x \leq 1-\delta_a} \left| \frac{a(x)}{1-x^2} \right| \right\} \| v_N(\cdot,t) \|_{T^+} ,$$

where $\delta_a > 0$ depends on the behaviour of $a(x)$ at the boundaries.

Corollary: *The following estimate holds :*

$$\frac{d}{dt} \| v_N(\cdot,t) \|_{T^+} \leq C_6 \, 4M_a K_p \, \frac{[(1-\gamma)N]^{7/2}}{[\gamma N]^{p-\frac{1}{2}}} \| v_N(\cdot,t) \|_{T^+} + \| f_N(\cdot,t) \|_{T^+}$$

$$+ \left\{ \max_{0 \leq x \leq N} \left| b_N(x,t) - \tfrac{1}{2} \frac{da_N}{dx}(x) \right| + \max_{-1+\delta_a \leq x \leq 1-\delta_a} \left| \frac{a(x)}{1-x^2} \right| \right\} \| v_N(\cdot,t) \|_{T_+} ,$$

If we take $\gamma = \alpha N^{-s}$, where $\alpha > 0$ and $0 \leq s < 1$ are constants independent of N, then for

$$p > \frac{7/2}{1-s} + \tfrac{1}{2} \geq 4 ,$$

this estimate converges to the corresponding estimate for equation (8.1) as

$N \to \infty$.

Finally using the results from Lemmas (5.1), (5.2), (6.3) and (8.2) it is possible to obtain the following stability theorem for the T^0-norm :

Theorem 8.6 : *The T^0-stabilized Tchebyshev collocation applied to Case III of equation (8.1) (that is with outflow condition at both $x=-1$ and $x=1$) satisfies the following estimate :*

$$\frac{d}{dt} \| v_N(\cdot,t) \|_{T^0} \leq \left\{ \frac{2C_6}{\pi} [(1-\gamma)N]^{7/2} \| a_{N,2} \|_T \right\} \| v_N(\cdot,t) \|_{T^0} + \| f_N(\cdot,t) \|_{T^0}$$

$$+ \left\{ \max_{0 \leq x \leq N} \left(b_N(x,t) - \tfrac{1}{2} \frac{da_N}{dx}(x) \right) + \max_{-1+\delta_a \leq x \leq 1-\delta_a} \left[\frac{a(x) \cdot x}{1-x^2} \right] \right\} \| v_N(\cdot,t) \|_{T^0} ,$$

where $\delta_a > 0$ depends on the behaviour of $a(x)$ at the boundaries.

Corollary: *The following estimate holds :*

$$\frac{d}{dt} \| v_N(\cdot,t) \|_{T^0} \leq C_6 \, 2M_a K_p \, \frac{[(1-\gamma)N]^{7/2}}{[\gamma N]^{p-\frac{1}{2}}} \| v_N(\cdot,t) \|_{T^0} + \| f_N(\cdot,t) \|_{T^0} ,$$

$$+ \left\{ \max_{0 \leq x \leq N} \left(b_N(x,t) - \tfrac{1}{2} \frac{da_N}{dx}(x) \right) + \max_{-1+\delta_a \leq x \leq 1-\delta_a} \left[\frac{a(x) \cdot x}{1-x^2} \right] \right\} \| v_N(\cdot,t) \|_{T^0} .$$

If we take $\gamma = \alpha N^{-s}$, where $\alpha > 0$ and $0 \leq s < 1$ are constants independent of N, then for

$$p > \frac{7/2}{1-s} + \tfrac{1}{2} \geq 4 ,$$

this estimate converges to the corresponding estimate for equation (8.1) as $N \to \infty$.

Maximum eigenvalues of $H^T H$		
N	λ_{max}	$\lambda_{max} / (N+\frac{1}{2})$
4	4.8633	1.0807
8	8.2801	0.9741
16	15.162	0.9189
32	28.963	0.8912
64	56.591	0.8774
128	111.86	0.8705
256	222.46	0.8673

Table 1

References

[1] Canuto,C. and Quarteroni,A., *Approximation Results for Orthogonal Polynomials in Sobolev Spaces*, Math. Comp., 38 (1982), pp. 67-86.

[2] Fornberg,B., *On a Fourier Method for the Integration of Hyperbolic Equations*, SIAM J. Numer. Anal., 12 (1975), pp. 509-528.

[3] Gottlieb,D., *The Stability of Pseudospectral-Chebyshev Methods*, Math. Comp., 36 (1981), pp. 107-118.

[4] Gottlieb,D. and Orszag,S.A., *Numerical Analysis of Spectral Methods: Theory and Applications*, Regional Conf. Series in Applied Math., SIAM, Philadelphia, Pa. (1977).

[5] Gottlieb,D., Orszag,S.A. and Turkel,E., *Stability of Pseudospectral and Finite-Difference Methods for Variable Coefficients Problems*, Math. Comp., 37 (1981), pp. 293-305.

[6] Kreiss,H.O. and Oliger,J., *Methods for the Approximate Solution of Time Dependent Problems*, GARP Publications Series, no. 10 (1973).

[7] Kreiss,H.-O. and Oliger,J., *Stability of the Fourier Method*, Math. Comp., 16 (1979), pp. 421-433.

[8] Majda,A. McDonough,J. and Osher,S., *The Fourier Method for Nonsmooth Initial Data*, Math. Comp., 32 (1978), pp. 1041-1081.

[9] Orszag,S.A., *Comparison of Pseudospectral and Spectral Approximations*, Studies in Applied Math., 51 (1972), pp. 253-259.

[10] Pasciak,J.E., *Spectral and Pseudo Spectral Methods for Advection Equations*, Math. Comp., 35 (1980), pp. 1081-1092.

[11] Quarteroni,A., *Theoretical and Computational Aspects of Spectral Methods*, Proceedings of the Fifth International Conference of Computing Methods in Applied Sciences and Engineering, Pub. by North-Holland, 1982.

[12] Rivlin,T.J., *The Chebyshev Polynomials*, Wiley, New York (1974).

[13] Whittaker,E.T. and Watson,G.N., *A Course in Modern Analysis*, Cambridge Univ. Press, (1927). 16 (1979), pp. 421-433.

Part II:

Interpolation for Surfaces with 1-D Discontinuities

1. Introduction

In this part we discuss an interpolation problem that arises in the context of numerically resolving shock profiles in two space dimensions. The problem, described in simple terms, is how to accurately reconstruct a function defined on the plane which is known only on a finite number of appropriately chosen parallel lines, or cross sections. We assume that on each line we know the function exactly; in practical terms we mean that the function is fully resolved on each line by a sufficiently refined mesh. If the function being considered is smooth, the problem can be solved in many standard ways. On the other hand, we are interested in solving the problem when the function has structure on a scale which is much smaller than the average distance between these parallel lines. The interpolation problem with this last feature cannot be solved for general functions; it is necessary to have information about the specific structure of the function we want to reconstruct.

We know that solutions of singular parabolic equations can have sharp gradients confined to one-dimensional regions, usually called shocks. This type of function can be reconstructed from a finite number of its cross sections. The idea of the method is to locate these regions and then perform the interpolation using function values all lying only on one side of the shock; if this is not possible then we perform the interpolation using values which are all placed on a curve parallel to the shock. In this way we interpolate using function values which are close to each other.

First, we describe how the interpolation problem arises in the context of calculating shocks in two space dimensions. Our approach is one of several possible but has the advantage that it is easy to implement. Brown [1] discusses two alternative approaches to the problem, one due to Oliger [12], Berger [2] and Gropp [3] and a second one by B. Kreiss [6]. The origin of the problem justifies the assumptions we make regarding the specific structure of the function; it

also gives an idea of what error estimates may be used.

Secondly, we describe a second order interpolation formula where we fit the shock locally using straight lines. This may not be enough for problems where the equation is sensitive to corners in the shock fronts. We study the error of the method and we obtain a criterion for placing the cross sections so as to minimize the errors committed in the procedure. The criterion involves the orientation and the curvature of the shock as well as the distance between the parallel sections. It is possible to fit the shock using higher order formulas, but the above relation must nevertheless hold for the interpolation to be meaningful.

Finally, we successfully test the method for some model functions and we apply it to Burgers' equation in two space dimensions for two sets of initial data. The numerical calculations corresponding to Burgers' equation were performed with Brown [1]. The method still needs to be tested on more challenging problems; we would also like to study the performance of higher order interpolation methods.

2. Adapting Splitting for Problems with Shocks

We present a different approach to the conventional splitting technique that allows us to solve shock problems in two space dimensions. Let us consider

$$\frac{\partial u}{\partial t} + A\,\frac{\partial u}{\partial x} + B\,\frac{\partial u}{\partial y} = \varepsilon^2\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{2.1}$$

for $-\infty < x,y < \infty$ and $t > 0$, where $A = A(u,x,y,t)$, $B = B(u,x,y,t)$ and $\varepsilon > 0$ is a small parameter. We first discretize in time using any of Gear's methods. (It is important to use a stiffly stable method in this step.) In order to minimize the change in the solution per time step we use a local moving coordinate system with velocity defined by the functions $U = U(x,y,t)$ and $V = V(x,y,t)$. For simplicity we now use implicit Euler (also called backward Euler) in time and obtain

$$\begin{aligned}
&\varepsilon^2\left(\frac{\partial^2 u}{\partial x^2}(\cdot,t+k) + \frac{\partial^2 u}{\partial y^2}(\cdot,t+k)\right) - \left(A(u(\cdot,t+k),\cdot,t+k) - U\right)\frac{\partial u}{\partial x}(\cdot,t+k) \\
&\quad - \left(B(u(\cdot,t+k),\cdot,t+k) - V\right)\frac{\partial u}{\partial y}(\cdot,t+k) - \frac{1}{k}u(\cdot,t+k) \qquad (2.2) \\
&\quad = -\frac{1}{k}u(\cdot,t)
\end{aligned}$$

where k is the time step. (We are intentionally using the same notation $u(\cdot,t+k) = u(x,y,t+k)$ for the solution of both the continuous and the time-discrete problems.)

Equation (2.2) is a nonlinear singular elliptic problem for $u(\cdot,t+k)$ that has to be solved at each time step. In order to solve this problem there is a need to generate a mesh on which the solution $u(\cdot,t+k)$ is resolved and at the same time the different terms of the equation are correctly approximated. It is in the method of generating this mesh that our splitting method differs from previous approaches. The question is how to locate and orient the mesh to align it with the curves on which the solution exhibits sharp gradients. We reduce the

problem of determining $u(\cdot,t+k)$ to that of solving a sequence of one dimensional equations. We apply splitting to (2.2) to obtain

$$\varepsilon^2 \left(\frac{\partial^2 \tilde{u}}{\partial x^2}(\cdot,t+k) - \left(A(\tilde{u}(\cdot,t+k),\cdot,t+k) - U(\cdot,t+k) \right) \frac{\partial \tilde{u}}{\partial x}(\cdot,t+k) \right)$$

$$- \frac{1}{k} \tilde{u}(\cdot,t+k) = - \frac{1}{k} u(\cdot,t) \qquad (2.3a)$$

$$\varepsilon^2 \left(\frac{\partial^2 u}{\partial y^2}(\cdot,t+k) - \left(B(u,\cdot,t+k) - V(\cdot,t+k) \right) \frac{\partial u}{\partial y}(\cdot,t+k) \right)$$

$$- \frac{1}{k} u(\cdot,t+k) = - \frac{1}{k} \tilde{u}(\cdot,t+k) . \qquad (2.3b)$$

The errors we commit in this step are $\mathbf{O}(k)$.

So far we have not discussed the practical problem of computing (2.1) in a finite domain with specified boundary conditions. It is not always possible to split the differential operator in a manner consistent with the boundary conditions. This is usually the case for non-rectangular domains. Even in these cases it may be possible to use a composite mesh technique, coupled with an iteration procedure between the different meshes, which transforms the problem to a set of two new singular equations each one defined on rectangular domains where the splitting can be done.

We notice that the spatial variable y only appears as a parameter in equation (2.3a), so we can solve this nonlinear singular perturbation problem for different values of y, say $y_1, y_2, \cdots\cdots, y_M$ where these values are appropriately chosen. The function $U_\nu(x,t+k) = U(x,y_\nu,t+k)$ is determined when solving for $\tilde{u}_\nu(x,t+k) = \tilde{u}(x,y_\nu,t+k)$, one of the cross sections of $\tilde{u}(x,y,t+k)$. Brown [1] and Hyman [4] discuss two different ways of determining this function at each time step. We are assuming that we have full knowledge of $u(x,y,t)$, the right hand side of (2.3a).

The nonlinear two point boundary value problem can be reduced to a linear one using Newton iteration on the differential equation. It is important to perform Newton iteration on the continuous problem and then discretize the linear problems as opposed to discretizing and then performing Newton iteration to solve the nonlinear system of algebraic equations. Using the second approach, we may converge to a solution of the algebraic system which is not close to the solution of the continuous problem.

Each Newton iteration reduces to solving a linear singular two point boundary value problem. There is extensive literature on this type of problem; we refer to H.O. Kreiss [8], Keller and Cebeci [5] and B. Kreiss and H.O. Kreiss [7]. These methods are based on choosing grid points in the x-direction, $x_1, x_2, \cdots, x_{N_\nu}$ (a different set of grid points for each value of y_ν) and a suitable discretization of the spatial operators in such a way that we obtain an accurate numerical solution.

If the initial values are given such that the solution of (2.1) exhibits a shock structure we will need extra points to resolve this jump. The number of extra points due to the singular behaviour depends on the specific structure of the shock. Numerical results obtained by H.O. Kreiss [8] suggest that the extra number of points required, for each value of y_ν, to resolve the transition region grows like $\mathbf{O}(\ln\varepsilon)$ as $\varepsilon \to 0$, for a given error bound on the solution.

We also notice that x only appears as a parameter in equation (2.3b) and introduce discrete values $x_1, x_2, \cdots, x_N$. In the process of solving this equation we need the function values of $\vec{U}(x, y, t+k)$ for values of y that in general we should not expect to have computed, i.e., for y not in the set $\{y_1, y_2, \cdots, y_M\}$.

Our interpolation problem is to determine $\tilde{u}(x,y,t+k)$ for any value of y, knowing $\tilde{u}(x,y_\nu,t+k)$, for $\nu = 1,2,\cdots,M$. (See Figure 1).

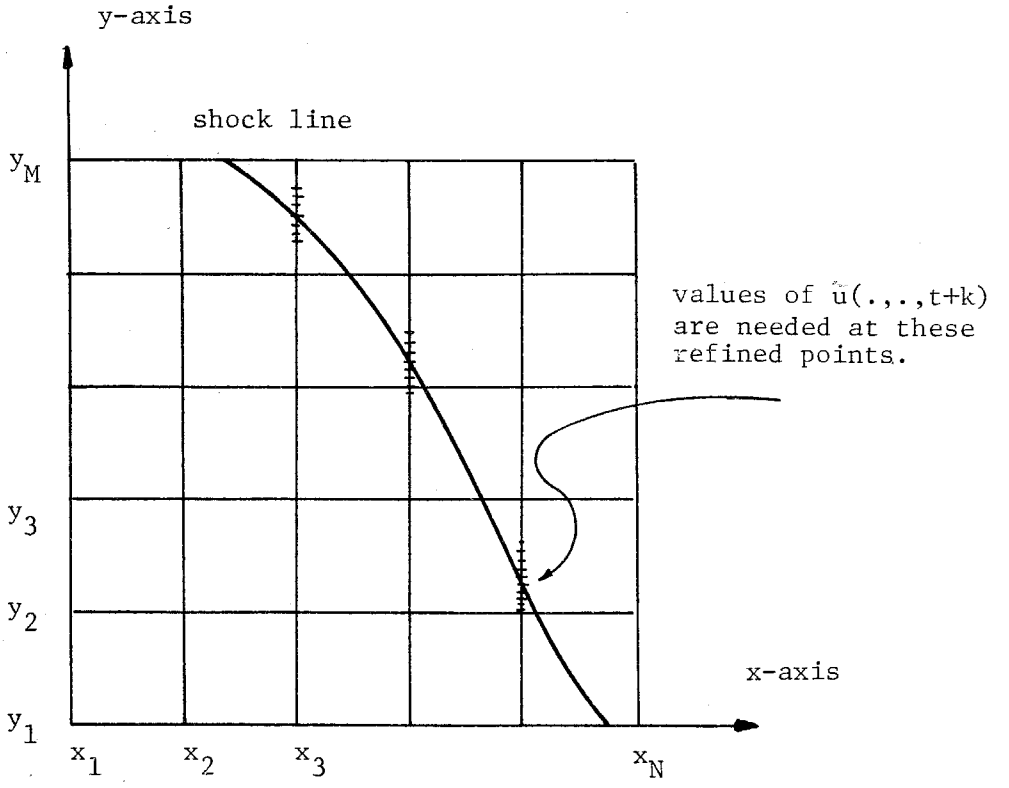


Figure 1

The next immediate problem is determining in which norm we should get bounds for the error of the procedure. This is equivalent to the question of in which norms equation (2.1) is well-posed. Actually, from the computational point of view, we need well-posedness of (2.1) and the corresponding norm estimate must be independent of ε, the small parameter. This involves obtaining stability results for the shape of the shock fronts of equation (2.1) with $\varepsilon = 0$.

Majda [10] proves a result concerning the existence of shock front solutions for nonlinear hyperbolic system of conservations laws in several space dimensions. If we define the position of the front at time t as the solution of $S(\underline{x},t) = 0$ then the result says that for smooth initial data which exhibits a jump discontinuity across the $\mathbf{C}^2$ hypersurface $S(\underline{x},t=0) = 0$, then for short

enough time the solution exists and the front remains a C^2 hypersurface. For two space dimensions Majda's result states that if the initial data belong to H^p away from the shock, where H^p denotes a Sobolev space and $p > 10$, then $S(x_0,y_0,t)$ as a function of t also belongs to H^p. In other words the rate at which the shock is deformed is determined by the smooth part of the initial values in a small enough time interval.

A different stability result is due to Oleinik [11]. In this case the result refers to one-dimensional quasi-linear hyperbolic equation.

A more complete knowledge of the solutions of equation (2.1) will determine how sensitive the numerical solution is to the errors introduced in the interpolation procedure.

3. Description of the Numerical Method

Consider the function $u_0(x,y)$ defined on the rectangle $[0,1]\times[0,1]$. We now want to determine $I(u_0)(x_0,y_0)$, the interpolant of the function $u_0(x,y)$ at (x_0,y_0), from the function values of $u_0(x,y)$ for any x and for values of y that belong to the sequence $0=y_1<\ldots.<y_N=1$. For simplicity we consider the problem without boundaries, that is we assume that $u_0(x,y_\nu)$ is defined for $-\infty<x<\infty$ and $\nu=1,2,\cdots N$.

The interpolation method we describe is a local procedure; this is a useful feature when the singular domain is topologically complex, for example when two or more shocks come together to a common point, when any of the shocks is not a simple curve, or when there is more than one shock in the computational domain.

Our aim is to interpolate using function values obtained from points which can be joined by a smooth curve entirely lying on a same smooth part of the solution. It is also important for the points to be relatively close to each other. We introduce h, a small positive number, where h^2 is the prescribed error tolerance for the interpolation.

We now present the assumptions on $u_0(x,y)$ on which the method is based.

(*i*) The function should be smooth at distances greater than ε away from a one dimensional region where there is singular behaviour. Here, ε is a positive number much smaller than the natural scale corresponding to changes of $u_0(x,y)$ outside of this region. If the function u_0 is only known at discrete values, we then assume that away from this singular region a mesh of size h completely resolves the function, where $h>>\varepsilon$.

(*ii*) We assume that the singular region is the union of a finite number of smooth curves. In this way the curves can be isolated from each other, and if they intersect, the number of possible intersections is finite.

(*iii*) We assume that the singular behaviour is of the shock type, i.e., an actual jump and not a high frequency oscillation which matches two different smooth states. We use this assumption in the method to define the local orientation of the shock.

We restrict ourselves to describing a second order interpolation procedure; the shape of the front is approximated by piecewise straight lines and the smooth parts of the solution by piecewise linear functions.

In this second order interpolation method, we make use of only two cross sections of the function $u_0(x_0,y_0)$ to determine $I(u_0)(x_0,y_0)$, that is we only use the values $u_0(x,y_{m+1})$ and $u_0(x,y_m)$, where $y_m \le y_0 \le y_{m+1}$.

Denote $P_m = (x_0,y_m)$ and $P_{m+1} = (x_0,y_{m+1})$ (see Figure 2) and introduce the jump in function values from top ($y = y_{m+1}$) to bottom ($y = y_m$) lines and the horizontal curvatures:

$$\delta u_0(\tilde{x},m) = |u_0(\tilde{x},y_{m+1}) - u_0(\tilde{x},y_m)| , \tag{3.1a}$$

$$\kappa(\tilde{x},m) = |\frac{\partial^2 u_0}{\partial x^2}(\tilde{x},y_{m+1}) - \frac{\partial^2 u_0}{\partial x^2}(\tilde{x},y_m)| , \quad \text{and} \tag{3.1b}$$

$$T(\tilde{x},m) = \max (\delta u_0(\tilde{x},m) , \kappa(\tilde{x},m)) . \tag{3.1c}$$

We have two general cases:

Case I. $T(x_0,m) \le \beta h$, where β is some positive constant (fudge factor). We are assuming that away from the singular region, the magnitude of the gradients of $u_0(x,y)$ are strictly bounded by β. In this case we assume that there is no shock structure nearby and we perform linear interpolation:

$$I(u_0)(x_0,y_0) = \frac{y_0-y_m}{y_{m+1}-y_m} u_0(x_0,y_{m+1}) + \frac{y_{m+1}-y_0}{y_{m+1}-y_m} u_0(x_0,y_m) . \tag{3.2}$$

It is usually the case that there is no need to interpolate in this case; there

would be no need to know the values of the right hand of (2.3a) or (2.3b) if there is no shock structure nearby. It is important to look at the values of the horizontal second derivatives since ignoring them will cause the top and bottom part of the jump to be deformed. (In a higher order approximation we should look at the $T(\tilde{x},\tilde{m})$, for $\tilde{x} = x_0-h, x_0$ and x_0+h, as well as for $\tilde{m} = m-1, m$ and $m+1$, to determine the existence of a shock in the vicinity of the points where we are interpolating.)

Case II. $T(x_0,m) > \beta h$ (there is a singular structure in the vicinity). We have two different cases according to size of the jump $\delta u_0(x_0,m)$.

Case II.a. (See Figure 2) If $\delta u_0(x_0,m) > \beta h$, then we know the shock line crosses the segment $[P_m,P_{m+1}]$. Our aim now is to isolate the region of singular behaviour. Practical experience has shown that it is best to define this region with two curves. In Figure 2 these are the lines defined by $[(x_1^0,y_m),(x_0^1,y_{m+1})]$ and $[(x_0^0,y_m),(x_1^1,y_{m+1})]$. For convenience we assume that

$$\delta_1 = u_0(x_0,y_{m+1}) - u_0(x_0,y_m) > 0 . \tag{3.3}$$

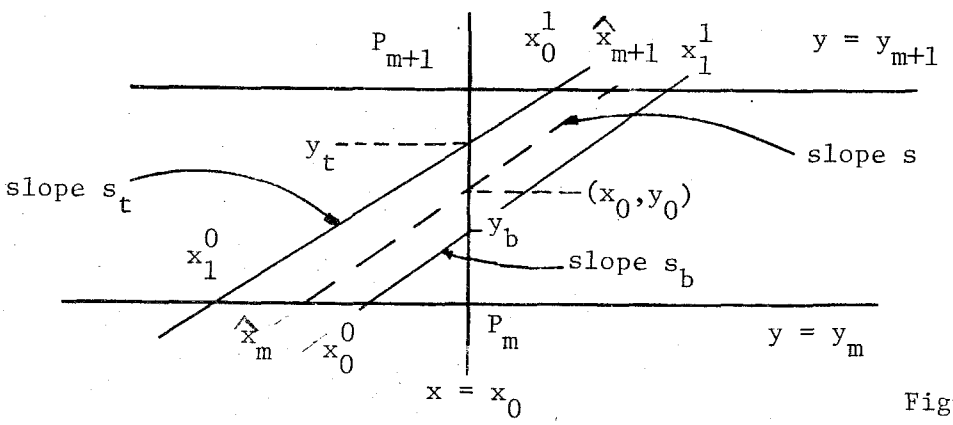


Figure 2

We now determine x_γ^1, for $\gamma = 0,1$, according to the following equation

$$u_0(x_\gamma^1,y_{m+1}) = u_0(x_0,y_{m+1}) - l_\gamma\delta_1 , \tag{3.4a}$$

where $l_0 = 1/3$ and $l_1 = 2/3$. In a similar way we determine x_γ^0 by

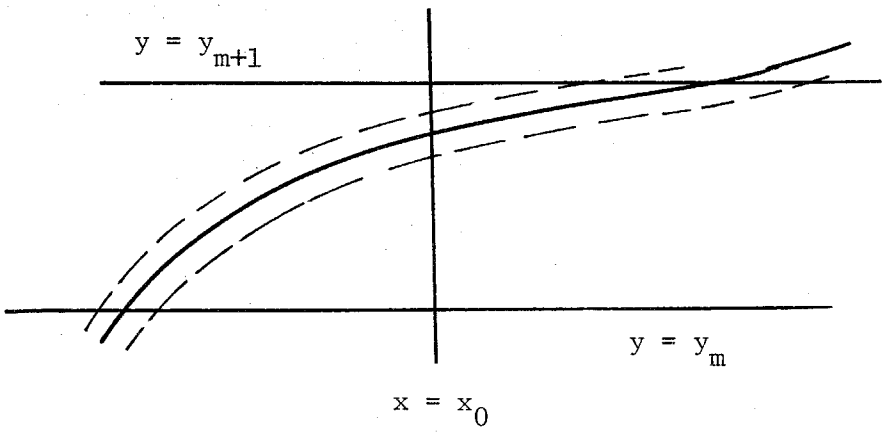


Figure 3

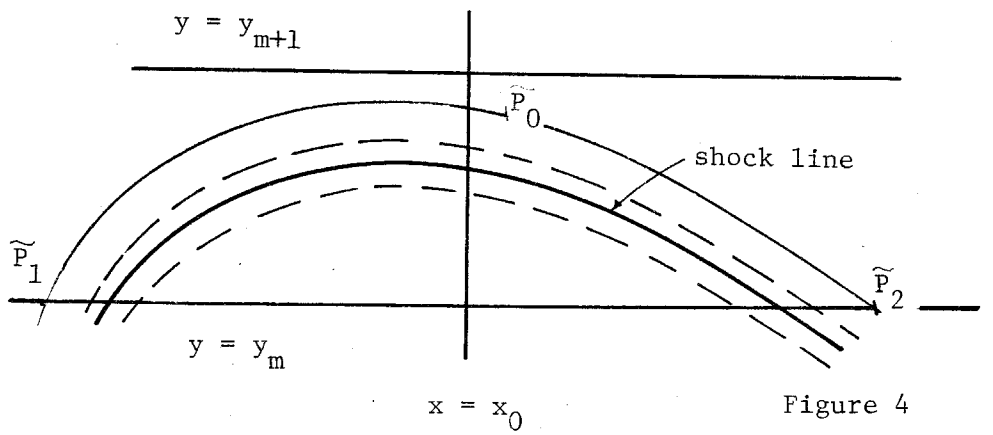


Figure 4

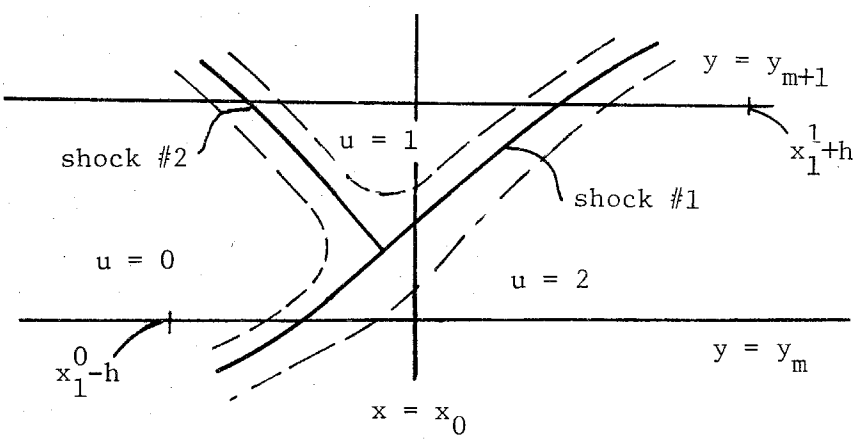


Figure 5

$$u_0(x_\gamma^0, y_m) = u_0(x_0, y_m) + l_\gamma \delta_1 \ . \tag{3.4b}$$

The slope in the shock region is of $\mathbf{O}(\varepsilon^{-1})$ and a change in the constants l_γ will only produce a change of $\mathbf{O}(\varepsilon)$ in the determination of the points x_γ^σ which themselves determine the direction of the shock.

We now make sure that the points obtained are close enough, that is we compute

$$(d_t)^2 = (y_{m+1} - y_m)^2 + (x_1^0 - x_0^1)^2 \tag{3.5a}$$

$$(d_b)^2 = (y_{m+1} - y_m)^2 + (x_0^0 - x_1^1)^2 \ . \tag{3.5b}$$

If either d_t or d_b are greater than $2h$, or it is not possible to determine any of the points x_γ^σ, for $\gamma = 0,1$ and $\sigma = 0,1$, then we decide there is not enough information to perform an accurate interpolation. In this case we need to obtain the values of $u_0(x, \tilde{y})$, where $\tilde{y} = \frac{1}{2}(y_m + y_{m+1})$ and $-\infty < x < \infty$. In the shock problem this has to be done going back to the last half time step computation. (See Figures 3 and 4 for some possible situations in which this procedure asks for more information).

We must still consider the possibility illustrated in Figure 5, that is when two or more shocks come together at one point. In order for the method to recognize this situation we also make sure that

$$| \ u_0(x_1^1 + h, y_{m+1}) - u_0(x_0, y_m) | \ \leq \ \beta h$$

and similarly that

$$| \ u_0(x_1^0 - h, y_m) - u_0(x_0, y_{m+1}) | \ \leq \ \beta h \ .$$

(The example in Figure 5 would fail the second test because of the presence of shock #2.) If either of these conditions were not to hold we ask for more

information again.

Having ruled out the anomalous cases we now determine the slope of the segments joining the points that lie on the same part of the smooth solution

$$s_t = \frac{y_{m+1}-y_m}{x_1^0 - x_0^1} \tag{3.6a}$$

$$s_b = \frac{y_{m+1}-y_m}{x_1^1 - x_0^0} \tag{3.6b}$$

and the intersections of the segments with the vertical line through P_m and P_{m+1}

$$y_t = s_t\,(x - x_0^1) + y_m \tag{3.7a}$$

$$y_b = s_b\,(x - x_0^0) + y_m\,. \tag{3.7b}$$

Defining s in the following way

$$s = \begin{cases} s_t & \text{for } y_{m+1} \geq y_0 \geq y_t \\ \dfrac{y_0-y_b}{y_t-y_b}\,s_t + \dfrac{y_t-y_0}{y_t-y_b}\,s_b & \text{for } y_t \geq y_0 \geq y_b \\ s_b & \text{for } y_b \geq y_0 \geq y_m \end{cases} \tag{3.8}$$

we compute the intersections of a straight line through the point (x_0,y_0) with slope s with the top and bottom lines $y = y_m$ and $y = y_{m+1}$

$$\hat{x}_{m+1} = x_0 + \frac{y_{m+1}-y_0}{s} \tag{3.9a}$$

$$\hat{x}_m = x_0 + \frac{y_0-y_m}{s} \tag{3.9b}$$

and finally perform linear interpolation using the function values of u_0 at the points $(\hat{x}_m,y_m)$ and $(\hat{x}_{m+1},y_{m+1})$:

$$I(u_0)(x_0,y_0) = \frac{y_0-y_m}{y_{m+1}-y_m}\, u_0(\hat{x}_{m+1},y_{m+1}) + \frac{y_{m+1}-y_0}{y_{m+1}-y_m}\, u_0(\hat{x}_m,y_m)\,. \qquad (3.10)$$

Case II.b. If $\delta u_0(x_0,m) \le \beta h$, we then have a shock close to either point (or close to both points). Introduce

$$\tilde{\delta}(\tilde{x},m,n) = \max\,(\,|\,u_0(\tilde{x},y_m) - u_0(\tilde{x}+h,y_n)\,|\,,\,|\,u_0(\tilde{x},y_m) - u_0(\tilde{x}-h,y_n)\,|\,)\,;$$

and define $\delta_2 = \max\,(\,\tilde{\delta}(x_0,m,m+1)\,,\,\tilde{\delta}(x_0,m+1,m)\,)$. If $\delta_2 \le \beta h$ then we need more information (see the discussion following equations (3.5)). Otherwise we proceed as in *Case II.a* looking for either a vertical or an oblique or curved shock, as in Figure 4. If it is not possible to find any such structure we again need extra information.

We notice that the interpolation procedure does not produce a continuous function of (x_0,y_0); specifically it may be discontinuous at (x_0,y_0) when $x_0 = x_b$ or $x_0 = x_t$ and $y_m < y_0 < y_{m+1}$. The interpolation nevertheless produces a continuous function along the y-direction for any given x, and this is all that is needed since the interpolant is computed at only a discrete set of x values. In principle, the discontinuities can be eliminated from the definition of the interpolant. For example, we can use a "buffer" zone of width $s\,\delta y = s\,(y_{m+1}-y_m)$ to switch from an interpolation using points along a segment of slope s to an interpolation using a vertical segment. Even though this modification produces a smooth interpolant, it is of no practical use; the buffer zone introduces the extra complexity that in order to determine the orientation of the interpolation segment, we would have to test for a shock in an interval of width d where $d^2 = 4h^2 - \delta y^2$ (see equation (3.5)). This testing would involve a substantial amount of work.

We now discuss the possibility of using higher order methods for fitting the shape of the front. In the cases corresponding to Figures 3 and 4 the

interpolation method will fail and will ask for extra information until δy, the distance in between two consecutive horizontal lines, gets to be of $\mathbf{O}(\varepsilon)$, the width of the shock. If we use a higher order fitting method and even if we assume that the shape of the front is known exactly, we will still need to add extra horizontal lines. In order to find the value of the interpolant at $P_0 = (x_0, y_0)$, we will perform some interpolation along a curve parallel to the front. The interpolation formula will link values of the functions at points like $\tilde{P}_1$ and $\tilde{P}_2$, and possibly points on other horizontal lines. The distance between these points is $\mathbf{O}(\delta y^{1/2})$ in the cases of Figures 3 and 4. This implies that if we only restrict ourselves to interpolate using points that are $\mathbf{O}(h)$ apart from each other then we must restrict δy to be $\mathbf{O}(h^2)$. Hence,a higher order fitting method will reduce the number of operations when compared to the second order fitting method we have described; but, there will be no saving in the numerical effort for $\varepsilon << h^2$. if the shock lies horizontally (s small) and $\varepsilon >> h^2$.

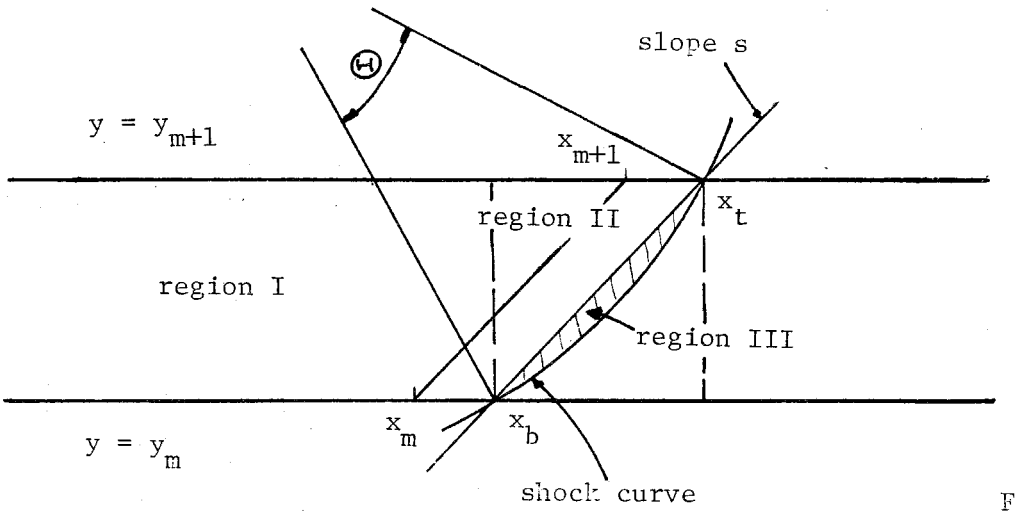


Figure 6

In order to study the error, we consider the problem corresponding to $\varepsilon = 0$ (that is an actual discontinuity) and to only one shock. In this case $x_0^0 = x_1^0 = x_b$, $x_0^1 = x_1^1 = x_t$ and $s_t = s_b = s$ where x_b corresponds to the intersection of the shock line with the line $y = y_m$ and similarly for x_t (see Figure 6). Without any loss of generality we can consider $x_b < x_t$. In this case the

interpolant evaluated at (x_0,y_0) (for $y_m \le y_0 \le y_{m+1}$) is defined by equation (3.2) when $x_0 \ge x_t$ or $x_0 \le x_b$, and by equation (3.10) when $x_b < x_0 < x_t$.

It is not possible to obtain an error formula in the maximum norm. The existence of such an error formula would imply that it is possible to determine the shape of an arbitrary curve on the plane from a finite number of its points. Nevertheless we have the following obvious local error estimates: the error is $\mathbf{O}(\delta y^2)$ away from the shock (region *I*), $\mathbf{O}(\delta y^2\sqrt{1+s^{-2}})$ near the shock region (region *II*), and $\mathbf{O}(1)$ between the shock and the chord $[(x_b,y_m),(x_t,y_{m+1})]$ (region *III*). The area of this last region is $A = \frac{1}{2}\kappa^2 (\vartheta - \sin(\vartheta)) (1+\mathbf{O}(\delta y))$ where κ is a value characteristic of the curvature of the shock front when the front lies between the lines $y = y_m$ and $y = y_{m+1}$, and ϑ is the change in the angle of the tangents of the shock front as the shock moves from $y = y_m$ to $y = y_{m+1}$. The angle ϑ is determined by $\sin(\frac{1}{2}\vartheta) = \frac{1}{2}\kappa\,\delta y\,\sqrt{1+s^{-2}}$.

In order to have an accurate interpolation, the shape of the front has to be resolved. We can assume that this has been achieved when A_{III}, the total area of regions of type *III*, is $\mathbf{O}(h^2)$. Now, when ϑ is small and $|s| > \delta y$ we have that $A \approx \kappa\,(\sqrt{1+s^{-2}}\,\delta y)^3$. Thus in order to resolve the shock we need $A\,\delta y / s = \mathbf{O}(h^2)$. In this way when $s = \mathbf{O}(1)$ and when the shock is a smooth curve (i.e. when we have an upper bound for κ), the condition on the area amounts to $\delta y / s$ being $\mathbf{O}(h)$. This relation between the orientation of the shock and the distance between two consecutive horizontal lines was enforced by making d_t and d_b smaller than $2h$. On the other hand when $|s| < \delta y$, we have that $A \approx \delta y$; hence in this case we need $\delta y = \mathbf{O}(h^2)$. This implies that we should stop adding extra horizontal lines when δy is $\mathbf{O}(h^2)$.

4. Numerical Experiments

In this chapter we present a test problem for the interpolation procedure and we include an application of the method to a time dependent problem.

Example 1. We introduce a set of parallel horizontal (that is $y=constant$) cross sections uniformly distributed in the y-direction in the interval $[0,1]$, each cross section corresponds to $y=y_0,y_1,\cdots,y_N$, where N is a natural number and $y_k=k/N$ for $k=0,1,\cdots N$. On each cross section the function values are specified at N^2 uniformly distributed mesh points $x=x_0,x_1,\cdots,x_{N^2}$, where $x_k=k/N^2$ and $k=0,1,\cdots N^2$.

In the test problem we considered different functions $u_0(x,y)$. Each function was evaluated at the N^2 mesh points of the N cross sections, and at the $4N^2$ points corresponding to the function values on the boundary of the region $0\leq x,y\leq 1$.

In the first step the function $u_0(x,y)$ was interpolated over $N-1$ vertical cross sections (that is $x=constant$) uniformly distributed inside the interval $[0,1]$ (there is no need to perform interpolation on the boundaries) each one consisting of N^2 points. We obtain a function $u_1(x,y)$ which is defined only on a mirror image of the mesh points where the original function $u_0(x,y)$ was evaluated.

In the second step $u_1(x,y)$ was interpolated to the original horizontal cross sections to obtain a second function $u_2(x,y)$.

We introduce the maximum norm, the L_1-norm and the L_2-norm for the error function

$$\|e(x,y)\|_\infty = \max_{x_k,y_l} |\, e(x_k,y_l)\,|$$

$$\|e(x,y)\|_2^2 = \sum_{k,l} (\, e(x_k,y_l)\,)^2 \cdot \frac{1}{N^3} \tag{4.1}$$

$$\|e(x,y)\|_1 = \sum_{k,l} |\, e(x_k,y_l)\,| \cdot \frac{1}{N^3}\,,$$

where $k=0,1,\cdots,N^2$ and $l=0,1,\cdots N$. According to the discussion of the previous chapter we expect that

$$\|e(x,y)\|_\infty = \mathbf{O}(1)$$

$$\|e(x,y)\|_2^2 = \mathbf{O}(N^{-4})+\mathbf{O}(\kappa N^{-2}) \tag{4.2}$$

$$\|e(x,y)\|_1 = \mathbf{O}(N^{-2}) + \mathbf{O}(\kappa N^{-2})$$

where κ is some measure of the curvature of the shock.

We show two numerical examples: Figure 7 corresponds to 11 cross sections of the function

$$u_0(x,y) = -\tanh\,(S(x,y)/\varepsilon)\,, \tag{4.3}$$

where $S(x,y) = y - \frac{1}{2}x^2 - \frac{1}{2}x$ and $\varepsilon = .02$; Figure 8 shows the function $u_0(x,y)$ after it has been interpolated twice. The corresponding errors are:

$$\|e(x,y)\|_\infty = 0.18\,, \quad \|e(x,y)\|_2 = .012 \quad \text{and} \quad \|e(x,y)\|_1 = .0022\,. \tag{4.4}$$

As a second example we considered the function

$$u_0(x,y) = -\tanh\,(S(x,y)/\varepsilon) + \tfrac{1}{4}\sin(\pi(x+y))\,, \tag{4.5}$$

where $S(x,y) = y - x$ and $\varepsilon = .02$. Figure 9 shows again 11 cross sections of this function and Figure 10 corresponds to the same function after two interpolations.

The errors obtained were:

$$\|e(x,y)\|_\infty = 0.043 \quad , \quad \|e(x,y)\|_2 = .0067 \quad \text{and} \quad \|e(x,y)\|_1 = .0026 \, . \tag{4.6}$$

Example 2. We consider the numerical solution of Burgers' equation in two space dimensions

$$\frac{\partial u}{\partial t} + u \cdot \frac{\partial u}{\partial x} + u \cdot \frac{\partial u}{\partial y} = \varepsilon \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) , \tag{4.7}$$

for $0 \le x, y \le 1$ and $t > 0$. The calculations were performed using the code developed by Brown [1]. Two different initial values were considered.

The first set of initial values (Figure 11) corresponds to a ramp connecting the constant values $u = \pm 1$. If there were no boundaries involved, then this problem could be solved exactly by reducing equation (4.7) to Burgers' equation in one space dimension. The solution corresponds to a stationary shock with planar front. In Figure 12 we show the solution at time $t = 1$. The boundary conditions were given such that the shock did not develop high curvature at the boundaries.

The second initial values (Figure 13) correspond to two ramps that intersect each other producing a wedge. The ramps connect the constant values $u = \pm 1$. The solution corresponding to the inviscid case of (4.7) (that is $\varepsilon = 0$) consists in a contact discontinuity and the development of a shock. The effect of the viscosity is to smear out in time the contact discontinuity; simultaneously the corner of the wedge gets rounded. In Figure 14 we show the solution at $t = 0.20$, and in Figure 15 at $t = 0.50$. For further details refer to Brown [1].

Original Function

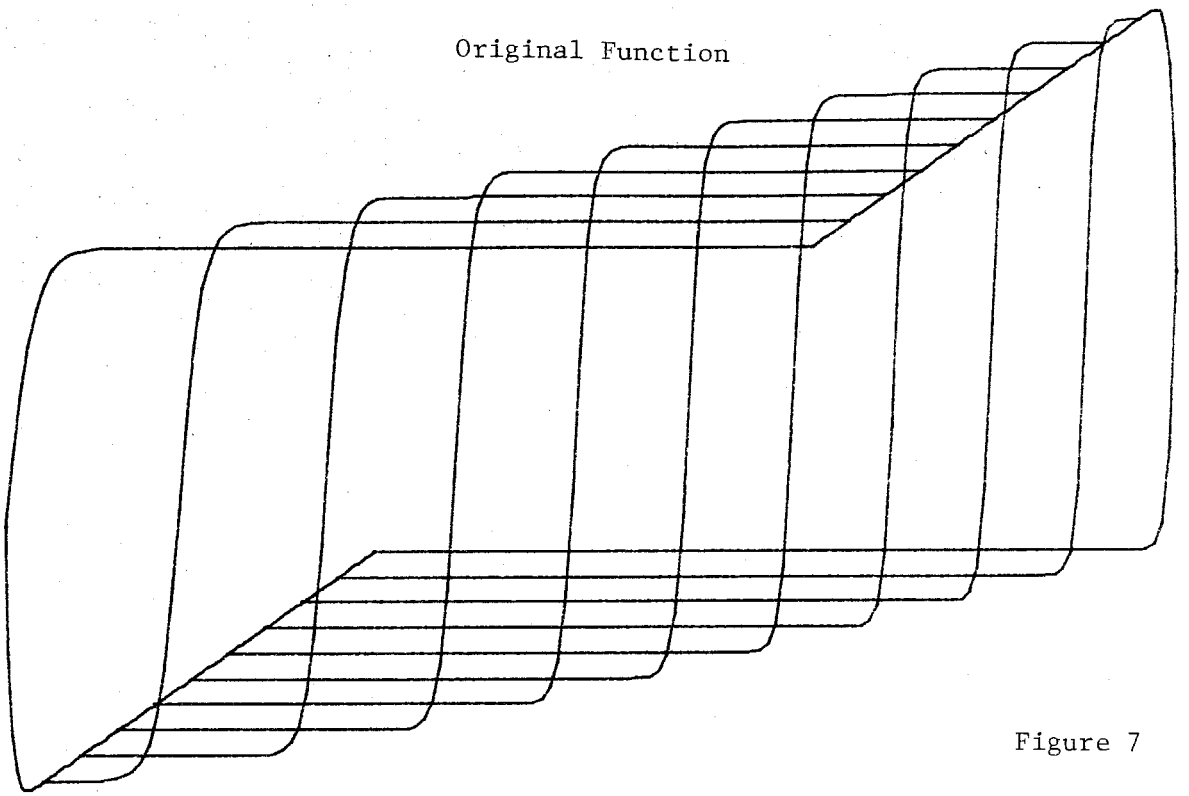
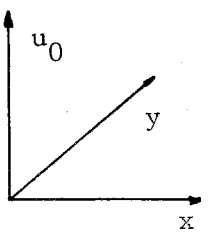


Figure 7



Interpolated Function

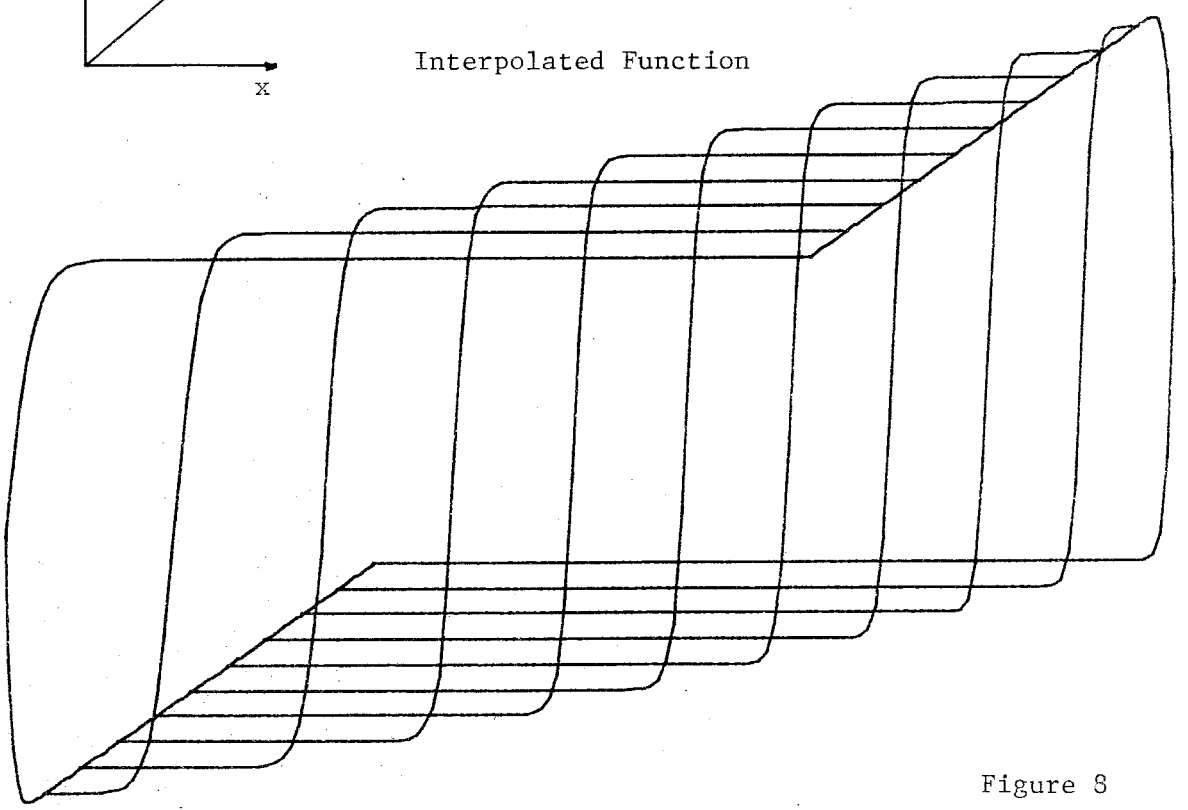


Figure 8

Original Function

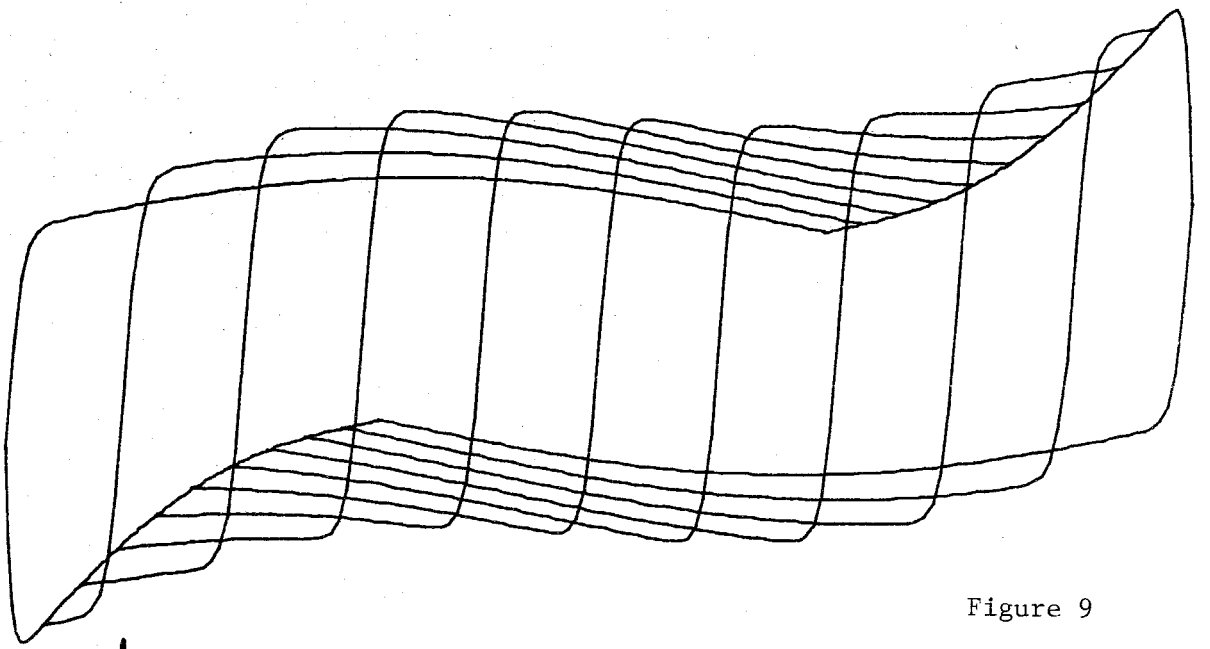
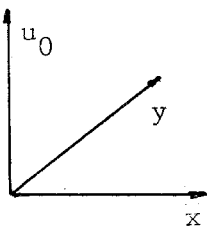


Figure 9



Interpolated Function

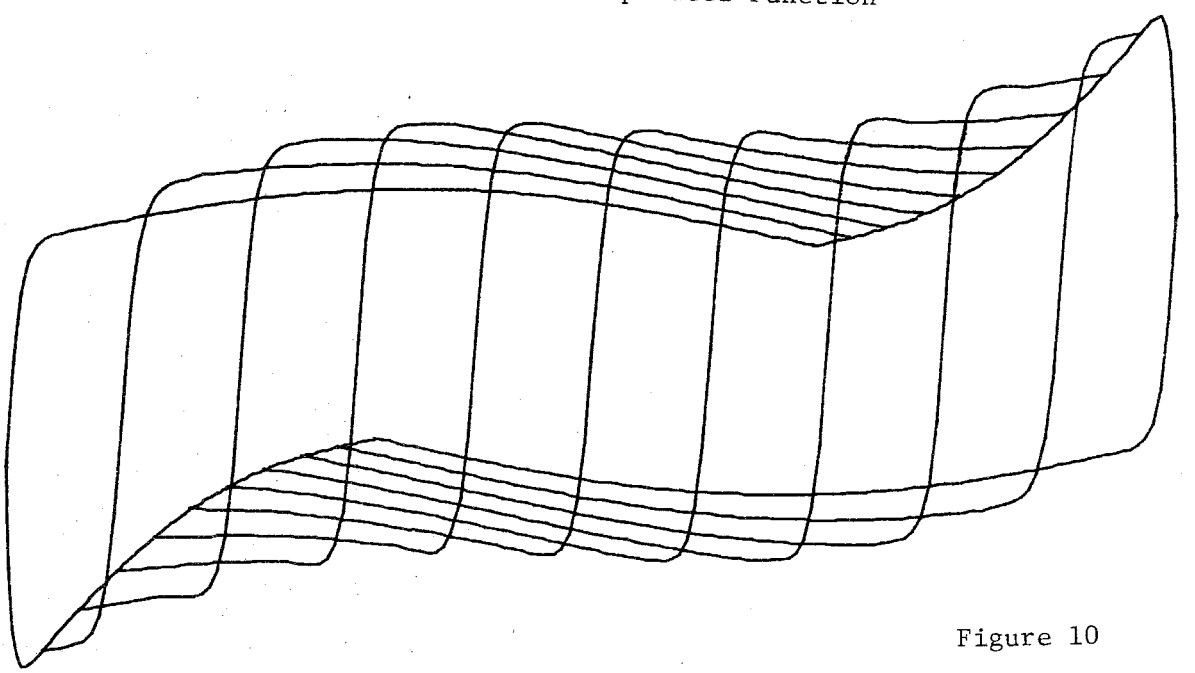


Figure 10

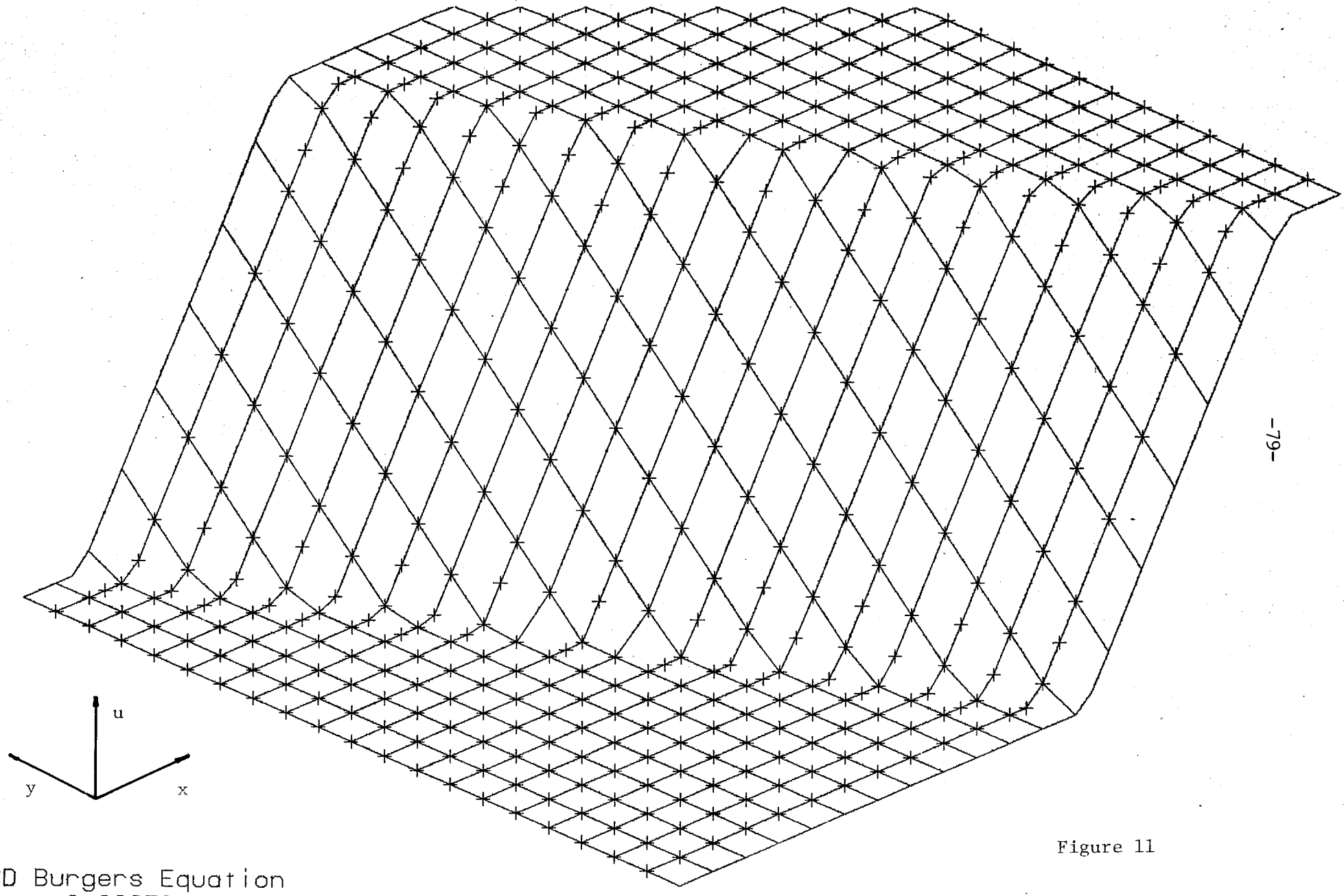


Figure 11

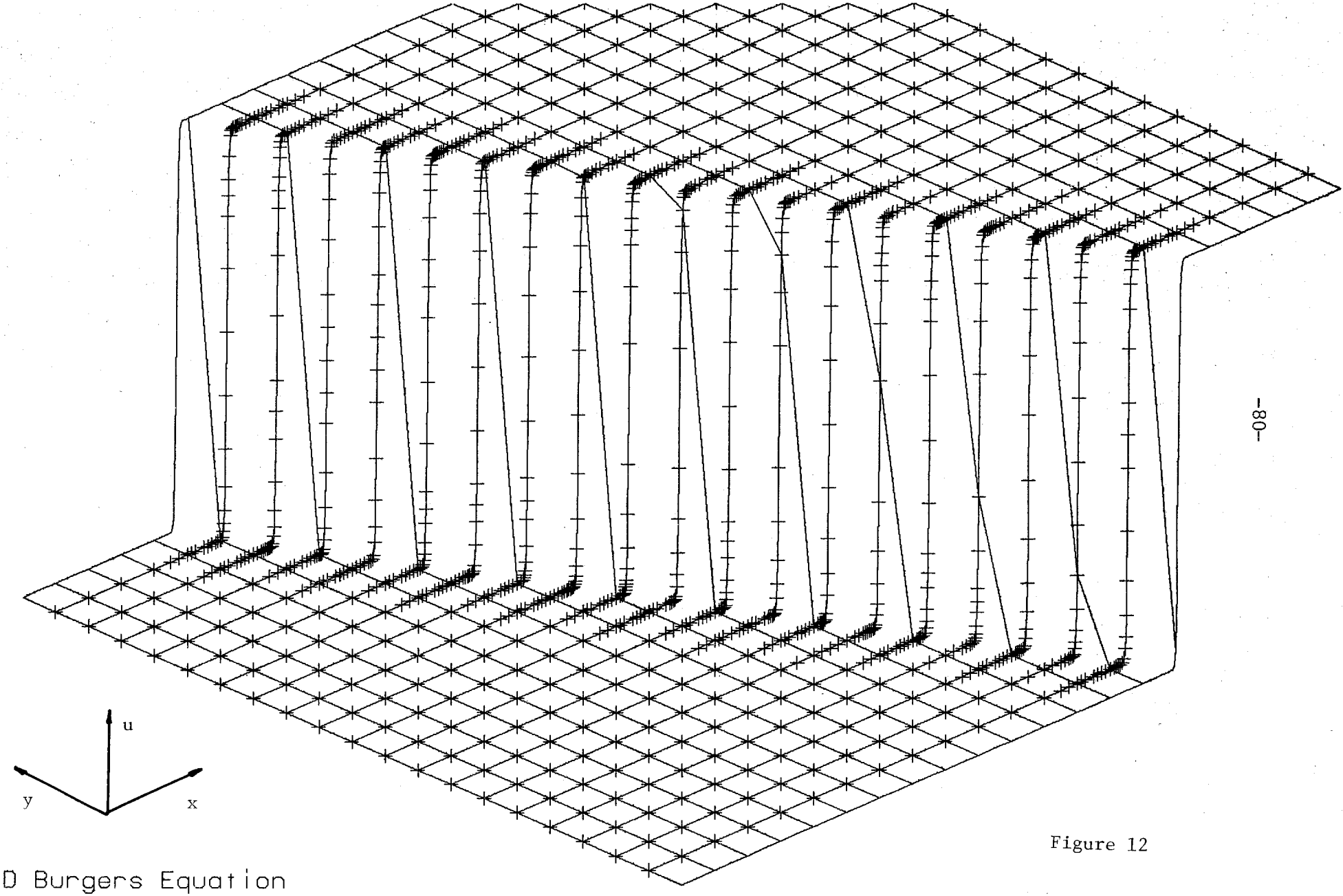


Figure 12

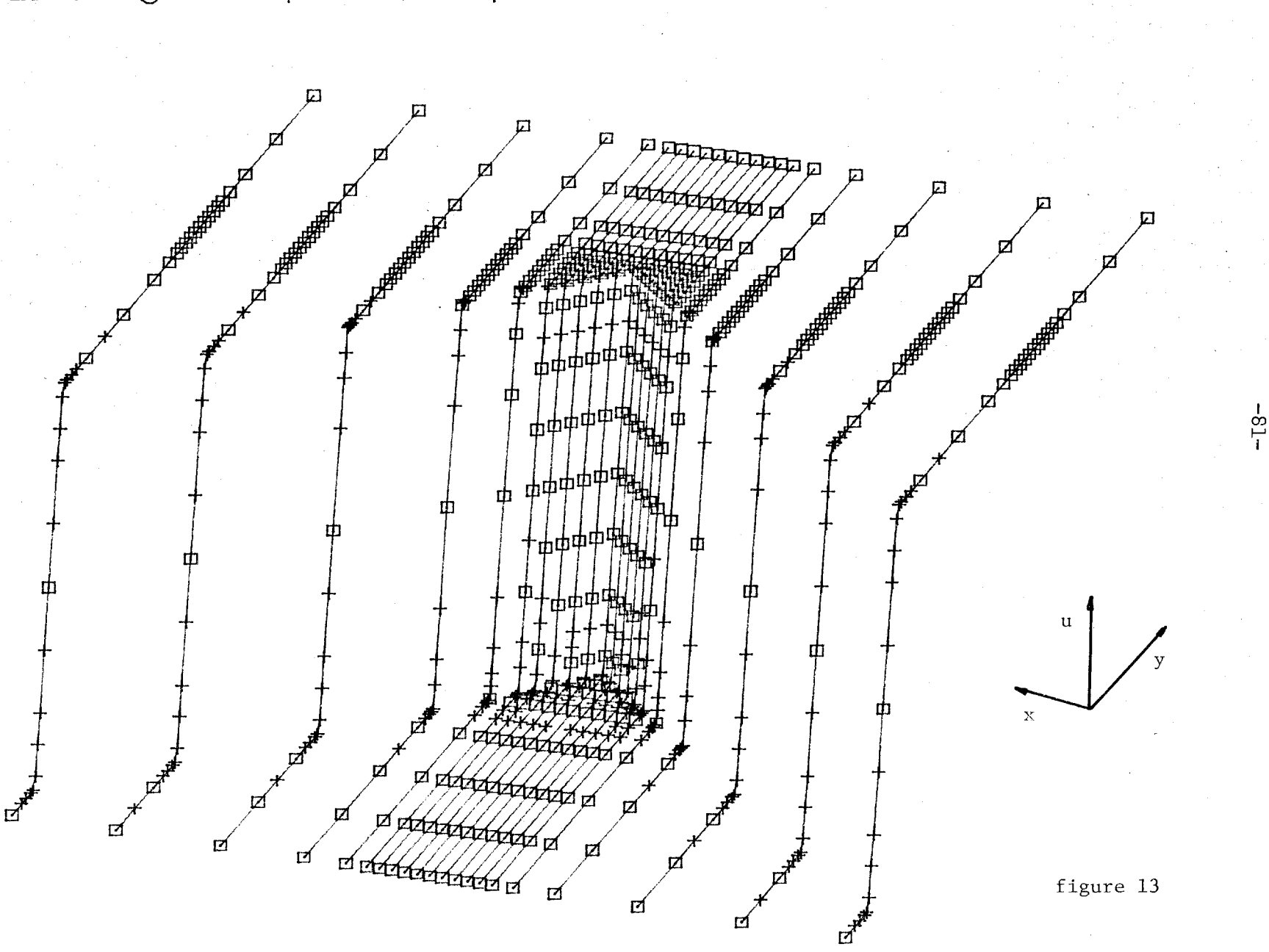


figure 13

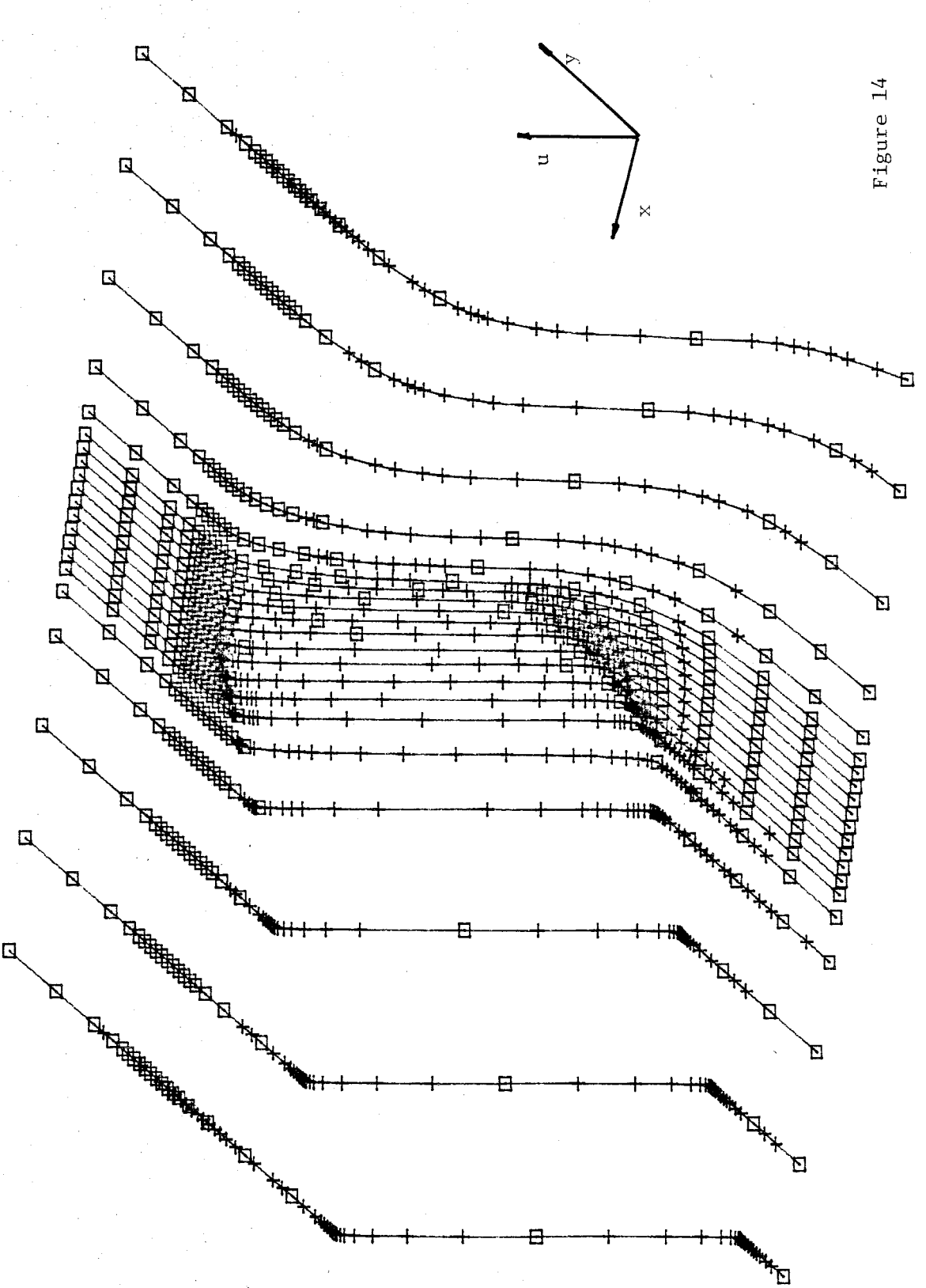


Figure 14

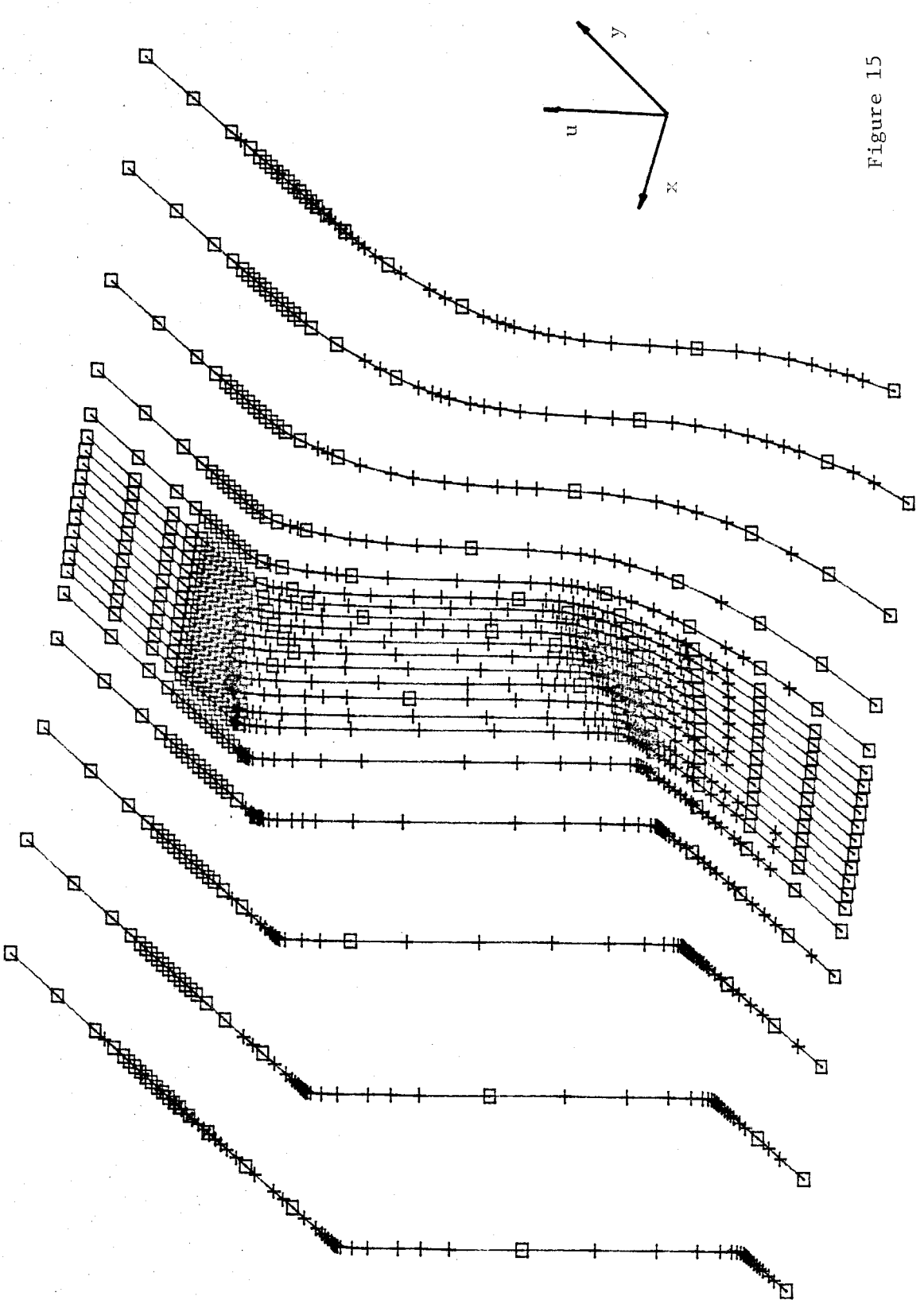


Figure 15

References

[1] Brown,D.L., *Ph.D. Thesis*, Calif. Inst. of Tech. Dept. of Applied Math., (1982).

[2] Berger,M., *Ph.D. Thesis*, Stanford University Dept. of Computer Science, (1982).

[3] Gropp,W.D., *A Test of Moving Mesh Refinement for 2-D Scalar Hyperbolic Problems*, SIAM J. Sci. Stat. Comut., 1 (1980) pp. 191-197.

[4] Hyman,J.M., *A Method of Lines Approach to the Numerical Solution of Conservation Laws*, Los Alamos Preprint, LA-UR-79-837, (1979).

[5] Keller,H.B. and Cebeci,T., *Accurate Numerical Methods for Boundary-layer Flows, II*, AIAA J., 10 (1972) pp. 1193-1199.

[6] Kreiss,B., *Construction of Curvilinear Grids*, Uppsala University Dept. of Comp. Sci. Rept. no. 89, (1981).

[7] Kreiss,B. and Kreiss,H.O., *Numerical Methods for Singular Perturbation Problems*, SIAM J. Num. Anal., 18 (1981) pp. 262-276.

[8] Kreiss,H.O., *Shock Calculations and the Numerical Solution of Singular Perturbation Problems*, Transonic, Shock, and Multidimensional Flows: Advances in Scientific Computing, (1982), pp. 289-311, pub. Academic Press.

[9] Kreiss,H.O., *Numerical Solution of Conservation Laws*, in Advances in Fluid Mechanics, Lecture Notes in Physics, 148 (1981), pp. 15-37, pub. Springer-Verlag.

[10] Majda,A., *The Existence of Multi-Dimensional Shock Fronts*, report Center for Pure and Applied Math, Univ. of Calif., Berkeley.

[11] Oleinik,O., *Uniqueness and Stability of the Generalized Solution of the Cauchy Problem for a Quasilinear Equation*, Amer. Math. Soc. Trans., Ser 2, 33 (1963) pp. 285-290.

[12] Oliger,J., *Adaptive Composite Grid Methods for Time Dependent Problems*, presented at the Adaptive Mesh Workshop, Center for Nonlinear Studies, Los Alamos National Lab, (1981).

[13] Swartz,B., *Courant-like Conditions Limit Reasonable Mesh Refinement to Order H^2*, Los Alamos Preprint, LA-UR-81-2037, (1981).

Part III:

On Composite Meshes

1. Introduction

In this part, we present a study of the composite mesh technique when applied to two dimensional problems. The method arises as a possible solution to the problem of numerically approximating a differential equation defined on a domain of complicated geometry. Specifically, we are interested in the generation of meshes on which the differential operators can be approximated in an accurate way. The mesh generation technique has to be general enough that it can be applied to any reasonable geometry. From a practical point of view, it is important to have meshes with simple storage structure; in this way any differential operator can be easily approximated. Such meshes are particularly convenient for computers with vector processors.

The composite mesh technique involves the use of rectangular meshes to cover the domain. A transformation of the independent variables is used in order to accommodate each mesh to the boundary and to the shape of any region where the solution exhibits singular behaviour. An interpolation procedure connects the numerical solutions corresponding to the various meshes.

There are two other approaches to the problem of generating meshes for domains with complicated geometry.

One approach uses the finite element method where the mesh is generated from an arbitrary distribuition of points inside the domain, so there is much freedom in the choice of the mesh. Its disadvantage is that the development of numerical approximations to the differential operators in non-regular meshes becomes extremely involved. Even though the method provides a framework where approximations can be generated, it is difficult to define higher order approximations. There is an extensive literature on these methods; we mention Gelinas, Doss and Miller [13], Bank [2] and Alexander, Manselli and Miller [1].

A second approach involves the introduction of an auxiliary problem to generate the mesh. A simple idea is to find a conformal transformation to map the domain of computation into a simpler geometry, for example a circle or a square. For problems that are essentially harmonic, this is a reasonable approach, but it is not clear how useful the method is for other types of problems: the distribution of points is determined by properties of analytic functions and not by the problem we want to solve. (Fornberg [12] has developed and successfully used a fast method to compute these transformations. Other computational approaches have been developed by Ives and Livtermoza [17], and Davis [10].) Finding the transformation becomes difficult when the geometry is complicated, and this presents us with a problem harder than the original one.

Another way to generate orthogonal meshes is by solving an appropriate nonlinear hyperbolic system of equations (see Sorenson [24] and Starius [25]), but these methods run into the familiar problem of the formation of shocks.

Finally Steger and Sorenson [28] consider the problem of improving a given mesh by solving a Poisson equation with a forcing function chosen to produce a high density of points where needed. This method is particularly popular for solving low Reynold's number flows around airfoils.

Composite mesh techniques seem to be a reasonable way to approach this problem; they have enough flexibility to produce a high concentration of mesh points wherever we need them. The generation of the mesh is fast and in this way the mesh can be regenerated to follow shocks or any singular behaviour of the solution.

In chapter 2 we describe the method and apply it to some simple problems. We pay special attention to possible numerical artifacts.

In chapter 3 we study the numerical stability of the composite mesh technique when applied to hyperbolic equations. We also show an example of a

simple interpolation procedure between the meshes which defines an unstable numerical method.

Finally in chapter 4 we apply the method to a model of a wind-driven ocean circulation in a circular basin. The problem in a rectangular geometry has been considered by Beardsley [3-5], Beardsley and Robbins [6] and Bryan [8] using different numerical techniques. Our method allows us to efficiently place the mesh points along the boundaries and at the separation point of the current without introducing unnecessary points at the center of the circle. In order to avoid one-sided formulas to update the value of the vorticity on the wall and at the same time keep a second order accurate approximation, we locate the boundary of the basin between two consecutive grid lines. In our numerical computations we only used an explicit method in time; we discuss a combination of implicit and explicit discretizations that will enable us to take longer time steps in the time-dependent problem.

2. Description and First Experiments on the Composite Mesh Technique.

In this section we describe how the composite method is used for a problem in which the domain of interest, Ω, is a simply connected subset of $\mathbf{R}^2$ whose boundary, $\partial\Omega$, is a simple smooth curve. We consider the case when the use of two regular grids, G_b and G_i is enough to cover Ω; one grid is used to follow the boundary and the second grid to fill the interior (see Figure 3a).

Each regular grid consists of a rectangular grid defined in the unit square $S = [0,1]\times[0,1]$ and a smooth function T which maps the unit square onto a subset of the plane; the transformation T is one to one and its Jacobian is never singular. Each rectangular mesh is generated from uniform grids defined in the interval $[0,1]$ with grid spacing M^{-1}, for some natural number M.

The rectangular grid R_b corresponding to the boundary grid consists of $(N_b+1)\times(M_b+1)$ points of the form $(\frac{i}{N_b}, \frac{j}{M_b})$, uniformly distribu ted in S, where $i = 0,1,\cdots,N_b$ and $j = 0,1,\cdots,M_b$. The interior of the rectangular grid, R_b^o, consists of the points $(\frac{i}{N_b}, \frac{j}{M_b})$, where $i = 1,2,\cdots,N_b-1$ and $j = 1,2,\cdots,M_b-1$, we define the boundary of this grid, ∂R_b, a similar way. The transformation from R_b to the original curvilinear coordinate system is denoted by T_b. Note that,in particular, T_b must satisfy

$$T_b(0,y) = T_b(1,y) \tag{2.1}$$

for $0 \leq y \leq 1$; we can always assume that $T_b([0,1]\times\{0\}) = \partial\Omega$. In a similar way we introduce the interior mesh R_i. This mesh need not be the entire unit square, it may have indentations. From a practical point of view it is convenient to use a rectangular mesh for R_i and "flag" the unnecessary points. We also introduce the interior, R_i^o, and the boundary points, ∂R_i. The interior grid is picked so that we can cover the entire domain Ω with both grids, that is,

$$T_i(R_i) \cup T_b(R_b) = \Omega \ . \tag{2.2}$$

It is important that the grids are placed in such a way that the boundaries of each grid lie inside the interior of the remaining grid, that is

$$T_i(\partial R_i) \subseteq T_b(R_b^o) \quad \text{and} \quad T_b(\partial R_{b,1}) \subseteq T_i(R_i^o) \ , \tag{2.3}$$

where $\partial R_{b,1}$ is the connected part of the boundary of R_b which does not correspond to $\partial\Omega$. ($\partial R_{b,1} = T_b([0,1]\times\{1\})$).

B. Kreiss [18] has developed a numerical code that given a simply connected domain Ω defined through its boundary generates both of the grids discussed above and computes the derivatives of the transformations involved. Figure 3a was taken from [18]; Figure 3b corresponds to a composite mesh generated to study a free boundary problem associated to the displacement of oil in reservoirs. In this application a non-affine transformation was used in order to conveniently distribute the grid points (see Reinelt [23]). Both grids were generated using this code.

Example 1 In our first numerical experiment we considered the initial value problem for the two-dimensional wave equation

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \tag{2.4a}$$

defined for $t > 0$ and on $x^2 + y^2 \leq 1$; with initial values given by

$$u(x,y,t=0) = u_0(x,y) \quad \text{and} \quad \frac{\partial u}{\partial t}(x,y,t=0) = u_1(x,y) \ , \tag{2.4b}$$

and boundary conditions given in polar coordinates by

$$u(r=1,\vartheta,t) = u_b(\vartheta) \ ; \tag{2.4c}$$

here $u_0(x,y$ and $u_1(x,y)$ are given functions.

We use two regular meshes to numerically approximate equation (2.4a); a polar grid that covers the annulus $r > \frac{1}{2}$ and a cartesian grid that covers the center: $\max(|x|,|y|) < \frac{1}{2}$. We first introduce the mesh widths

$$\delta x = \frac{1}{N_x - 3} \qquad \delta y = \frac{1}{N_y - 3} \tag{2.5a}$$

$$\delta r = \frac{\frac{1}{2}}{N_r - 2} \qquad \delta \vartheta = \frac{2\pi}{N_\vartheta - 2}. \tag{2.5b}$$

The cartesian grid G_c is defined as

$$G_c = \left\{ (x_i, y_j) \mid i = 1,2,\cdots,N_x \text{ and } j = 1,2,\cdots N_y \right\} \tag{2.6a}$$

where $x_i = -\frac{1}{2} + (i-2)\,\delta x$ and $y_j = -\frac{1}{2} + (j-2)\,\delta y$. For the polar grid G_p we use

$$G_p = \left\{ (r_l \cos(\vartheta_k), r_l \sin(\vartheta_k)) \mid k = 1,2,\cdots N_\vartheta \text{ and } l = 1,2 \ldots N_r \right\} \tag{2.6b}$$

where $r_l = \frac{1}{2} + (l-2)\,\delta r$ and $\vartheta_k = (k-2)\,\delta\vartheta$. Notice the overlapping of the meshes.

We denote the values of the numerical solution $\tilde{u}(x,y,t)$ on each mesh by

$$u_{i,j}^m = \tilde{u}(x_i, y_j, m\,\delta t) \quad \text{and} \quad v_{k,l}^m = \tilde{u}(r_k, \vartheta_l, m\,\delta t)\,; \tag{2.7}$$

where δt is the time step. We consider the numerical approximation to equation (2.4a) defined using standard centered formulas, that is

$$\frac{u_{i,j}^{m+1} - 2\,u_{i,j}^m + u_{i,j}^{m-1}}{\delta t^2} \tag{2.8a}$$

$$= \frac{u_{i+1,j}^m - 2\,u_{i,j}^m + u_{i-1,j}^m}{\delta x^2} + \frac{u_{i,j+1}^m - 2\,u_{i,j}^m + u_{i,j-1}^m}{\delta y^2}$$

for $i = 2,3,\cdots,N_x - 1$ and $j = 2,3,\cdots,N_y - 1$;

$$\frac{v_{k,l}^{m+1} - 2\,v_{k,l}^{m} + v_{k,l}^{m-1}}{\delta t^{2}} \tag{2.8b}$$

$$= \frac{v_{k,l+1}^{m} - 2\,v_{k,l}^{m} + v_{k,l-1}^{m}}{\delta r^{2}} + \frac{1}{r_l}\,\frac{v_{k,l+1}^{m} - v_{k,l-1}^{m}}{2\,\delta r} + \frac{1}{r_l^{2}}\,\frac{v_{k+1,l}^{m} - 2\,v_{k,1}^{m} + v_{k-1,l}^{m}}{\delta \vartheta^{2}}$$

for $k=2,3,\cdots N_\vartheta-1$ and $l=2,3,\cdots N_r$. The initial data are imposed by providing two consecutive time levels.

We now impose ϑ-periodicity

$$v_{1,N_r} = v_{N_\vartheta-1,N_r} \quad \text{and} \quad v_{2,N_r} = v_{N_\vartheta N_r}\,, \tag{2.8c}$$

the boundary condition

$$v_{k,N_r} = u_b(\vartheta_k)\,, \tag{2.8d}$$

for $k=1,2,\cdots,N_\vartheta$.(For simplicity we are dropping the superscript $m+1$.)

Finally we update the interior rim ($k=1$) and the boundary of the cartesian grid. First we find the position of the the points of the interior rim in the rectangular grid, that is we find $i(k)$ and $j(k)$ such that

$$x_{i(k)} \le r_1\cos(\vartheta_k) < x_{i(k)+1} \quad \text{and} \quad y_{j(k)} \le r_1\sin(\vartheta_k) < y_{j(k)+1}\,, \tag{2.8e}$$

for $k=1,2,\cdots,N_\vartheta$ From the way the grids were defined it follows that $1<i(k)<N_x-1$ and $1<j(k)<N_y-1$. In this way we can perform linear interpolation using the values corresponding to the cartesian grid $u_{i(k),j(k)}^{m+1}$, $u_{i(k)+1,j(k)}^{m+1}$ $u_{i(k),j(k)+1}^{m+1}$ and $u_{i(k)+1,j(k)+1}^{m+1}$ to define $v_{k,1}^{m+1}$. That is, we define

$$v_{k,1} = w_1(\vartheta_k,r_1)\,u_{i(k),j(k)} + w_2(\vartheta_k,r_1)\,u_{i(k)+1,j(k)} \tag{2.8f}$$

$$+ w_3(\vartheta_k,r_1)\,u_{i(k),j(k)+1} + w_4(\vartheta_k,r_1)\,u_{i(k)+1,j(k)+1}$$

were the weights w are appropriately chosen.

In our numerical experiment we considered the initial values given by

$$u(r,\vartheta,t=0) = J_1(\alpha_0 r)\cos(\vartheta)\,, \tag{2.9a}$$

$$\frac{\partial u}{\partial t}(r,\vartheta,t=0) = -J_1(\alpha_0 r)\sin(\vartheta)\,, \tag{2.9b}$$

where J_1 is the Bessel function of order one, α_0 its smallest real zero ($\alpha_0 \approx 3.83171$), r and ϑ are the usual polar coordinates; the boundary condition given is given by

$$u(r=1,\vartheta,t) = 0\,. \tag{2.9c}$$

The analytic solution corresponding to equations (2.4) and (2.9) is given by

$$u(r,\vartheta,t) = J_1(\alpha_0 r)\cos(\vartheta+t)\,. \tag{2.10}$$

The method as described is extremely efficient to compute. The interpolation coefficients are computed once at the beginning of the program and stored with the corresponding pointers. All important inner loops of the numerical approximation defined in (2.8) vectorize, this is not the case for the loops corresponding to the interpolation procedure.

In Figures 4a and 4b we show a contour map of the initial data computed using defined by

$$N_r = 20 \quad N_\vartheta = 68 \quad N_x = 23 \quad N_y = 23\,. \tag{2.11}$$

Because of our plotting package we have to show the interior mesh and the boundary mesh separately. In Figures 5a and 5b we show the computed solution at $t=9$ (a little less than 3 revolutions of the initial data) and in Figures 6a and 6b the corresponding errors. (In these figures N is the number of time iterations that were used.)

The results shown are typical of what we observed in our calculations: the error plots correspond to smooth functions and there are no numerical artifacts introduced by the composite mesh technique.

Example 2 Browning, Kreiss and Oliger [7] pointed out, and showed numerical evidence, the possibility of diffraction at the interface between two grids. The following experiment was designed to detect this numerical artifact. Trefethen [29] and [30] discuss a related problem: the dependence of the numerical group velocity on the orientation of the wave front and the mesh.

We consider the scalar hyperbolic equation

$$\frac{\partial u}{\partial t} = \frac{\partial u}{\partial \vartheta} \tag{2.11a}$$

valid for $t > 0$ and on $r \leq 1$, with initial values given by

$$u(r,\vartheta,t=0) = u_0(r,\vartheta) . \tag{2.11b}$$

(No boundary conditions are necessary.) The analytic solution to equation (2.11) is given by

$$u(r,\vartheta,t) = u_0(r,\vartheta+t) . \tag{2.12}$$

In the experiment we used $u_0(r,\vartheta) = J_1(\alpha_0 r)\cos\vartheta$. We again consider standard centered schemes in space and time.

We first consider the mesh defined by

$$N_r = 13 \quad N_\vartheta = 120 \quad N_x = 23 \quad N_y = 23 . \tag{2.13a}$$

In Figures 7a and 7b we show the contour lines of the numerical solution at $t = 3\pi$. Notice that the angular speed of propagation in the innermost lines of 7b is slower than the exact solution, as if the solution were dragged by the interior mesh. In Figures 8a and 8b we display the corresponding errors. The actual

numerical velocity of propagation is smaller than the velocity corresponding to the continuous problem. This effect is due to the lack of resolution in the interior mesh; the artifact disappears as we increase the resolution of the method: in Figures 9a and 9b we show the computed solution over the grid defined by

$$N_r = 20 \quad N_\vartheta = 120 \quad N_x = 23 \quad N_y = 23 \,. \tag{2.13b}$$

and in Figures 10a and 10b the corresponding errors.

Example 3 Finally we consider Laplace's equation

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{2.14a}$$

defined on the unit circle, with Dirichlet boundary conditions

$$u(r=1,\vartheta) = u_0(\vartheta) \,, \tag{2.14b}$$

where u_0 is a given function. We consider again centered formulas to approximate equations (2.14); these formulas correspond to the steady state of equations (2.8a) and (2.8b). We again need the interpolation formula (2.8f) to obtain equations for the internal boundaries associated with this composite method. As a result we obtain a linear system of equations to be solved to determine the numerical solution:

$$A \, \underline{u} = \underline{r} \tag{2.15a}$$

where $A = [a_{m,n}]$ is a sparse matrix, $\underline{r}$ imposes the boundary conditions and $\underline{u}$ is the numerical solution. The vector $\underline{u}$ is defined by

$$\underline{u}_{i+(j-1)N_x} = u_{i,j} \quad \text{and} \quad \underline{u}_{N_x N_y + k + (l-1)N_\vartheta} = v_{k,l} \,, \tag{2.15b}$$

where i,j,k,l are in their usual ranges.

We notice that due to the interpolation equations (2.8) the matrix A is not symmetric. However, the matrix is diagonally dominant, so that

$$\sum_{n \neq m} |a_{m,n}| \leq |a_{m,m}| \tag{2.15c}$$

for all n; we have strict inequality for the equations corresponding to the boundary conditions and equality for the remaining equations. This equation holds because the coefficients were chosen to be all positive. It is readily seen that this matrix cannot be singular.

Note that the band width of this system is $\mathbf{O}(N_e)$, where N_e^2 is the total number of equations, when the overlapping region is only a few cells wide.

Starius [25] suggests using an iteration technique to solve the linear system of equations arising from composite mesh techniques. The method consists of first guessing the solution at the boundary of the interior mesh, then solving Laplace's equation in the interior of this mesh. The next step is to interpolate from this solution to obtain values for the interior rim of the boundary grid. We now solve in the annulus, imposing the necessary boundary condition on $\partial\Omega$, and from this solution we interpolate the values corresponding to the boundary of the interior mesh. These new boundary values are then used in the next iteration.

The rate of convergence depends on the width of the overlapping region. In order to understand this we consider the following iteration

$$\Delta u_{n+1} = 0 \tag{2.16a}$$

on $r \leq r_1$ with boundary condition given by $u_{n+1}(r=r_1) = \tilde{u}_n(r=r_1)$, and

$$\Delta \tilde{u}_{n+1} = 0 \tag{2.16b}$$

on the annulus $r_0 \leq r \leq 1$ with boundary conditions $\tilde{u}_{n+1}(r=r_0,\vartheta) = u_{n+1}(r=r_0,\vartheta)$

and $u_{n+1}(r=1,\vartheta) = 0$, here $0<r_0<r_1<1$. If we start with $u_0(r=r_1,\vartheta) = \cos(\omega\vartheta)$ it easily seen that $u_n(r=r_1) = \sigma(\omega)^n\cos(\omega\vartheta)$ where $\sigma(\omega) = (\frac{r_0}{r_1})^n \frac{1-r_1^{2\omega}}{1-r_0^{2\omega}}$. This implies that modes corresponding to small ω are slow in convergence unless r_1-r_0 is small (a case in which we are not usually interested). This example suggests that the use of a multigrid method might be appropriate to solve the system (2.15a), see Linden [21]).

We decided to use a sparse matrix solver in our numerical experiments. In order to obtain the LU-decomposition of A we used the package developed at Yale University which has a reordering routine to minimize the overall number of operations, The package itself has no pivoting strategy. The numerical experiments we present were performed using single precision arithmetic on a VAX11/780 computer; this machine carries 7 digits for each real variable. Similar experiments were performed using a Cray-1 computer, which uses 14 digits per real variable. The results obtained agree with each other to at least 4 digits; this strongly suggests that there is no need to use a pivoting strategy to guarantee numerical stability in the LU-decomposition.

We consider the following regular meshes:

$$\textit{Mesh I}: \quad N_x = N_y = 8 \quad N_\vartheta = 33 \quad N_r = 7\,, \tag{2.17a}$$

$$\textit{Mesh II}: \quad N_x = N_y = 13 \quad N_\vartheta = 65 \quad N_r = 12\,, \tag{2.17b}$$

$$\textit{Mesh III}: \quad N_x = N_y = 18 \quad N_\vartheta = 99 \quad N_r = 17\,, \tag{2.17c}$$

$$\textit{Mesh IV}: \quad N_x = N_y = 23 \quad N_\vartheta = 68 \quad N_r = 20\,, \tag{2.17d}$$

with boundary values given by

$$u(r=1,\vartheta) = \cos(\omega\,\vartheta)\,. \tag{2.18}$$

In Table 1 we show the results corresponding to these four meshes for $\omega = 1,2,\cdots,8$. We also show the storage requirement, which gives the total number of operations needed to perform a back-substitution, and the total number of equations. The result suggests that the number of operations grows at a rate smaller than $N_e^{3/2}$, where again N_e is the total number of equations. The total number of operations might be substantially reduced if the mesh points corresponding to the corners of the inner mesh get "flagged out". The L_2 error is defined as

$$\varepsilon^2 = \sum_{i,j} e(x_i,y_j)^2\ \delta x\ \delta y\ + \sum_{k,l} e(\vartheta_k,r_l)\ r_l\ \delta r\ \delta\vartheta\,, \tag{2.19}$$

were $e(x,y)$ is the error function.

In Chapter 4 of this part we will need the to solve Poisson's equation discretized on a stretched mesh. For convenience we describe the type of stretching in this chapter, at the same time we present the difference approximations and some numerical results.

We introduce the stretching transformations given by

$$\tilde{r} = f(r) = r + e^{\frac{r-1}{\varepsilon_r}} \tag{2.20a}$$

$$\tilde{\vartheta} = g(\vartheta) = \frac{\varepsilon_1}{2\varepsilon_0}\left[\ln(\cosh(\frac{\vartheta_{\min}-\vartheta}{\varepsilon_1})) - \ln(\cosh(\frac{\vartheta_{\max}-\vartheta}{\varepsilon_1}))\right] + \vartheta + \alpha\,\vartheta^2\,; \tag{2.20b}$$

where the derivatives are given by

$$\frac{d\tilde{r}}{dr} = \frac{df}{dr}(r) = 1 + \frac{1}{\varepsilon_r}\,e^{\frac{r-1}{\varepsilon_r}} \tag{2.20c}$$

$$\frac{d\tilde{\vartheta}}{d\vartheta} = \frac{dg}{d\vartheta}(\vartheta) = 1 + 2\alpha\vartheta - \frac{1}{2\varepsilon_0}\left[\tanh(\frac{\vartheta_{\min}-\vartheta}{\varepsilon_1}) - \tanh(\frac{\vartheta_{\max}-\vartheta}{\varepsilon_1})\right]. \tag{2.20d}$$

Equation (2.20a) defines an exponential stretching of characteristic length ε_r which smoothly connects to the identity transformation. Similarly equation (2.20b) defines a uniform grid between $\vartheta_{\min}$ and $\vartheta_{\max}$ and a different uniform grid in the rest of the interval $[0,2\pi]$; the mesh ratio of these grids is given by $1/\varepsilon_0$ and the transition region from one mesh width to the other has a characteristic width of ε_1.

We consider the cartesian grid defined in (2.5a) and (2.6a); for the polar grid we consider

$$\tilde{r}_l = \tilde{r}_1 + (l-2)\delta\tilde{r} = f(r_l) \tag{2.21e}$$

$$\tilde{\vartheta}_k = \tilde{\vartheta}_1 + (k-2)\delta\tilde{\vartheta} = g(\vartheta_k) \tag{2.21f}$$

where $\delta\tilde{\vartheta} = (g(2\pi)-g(0))/(N_\vartheta-2)$, $\delta\tilde{r} = (f(1)-f(\frac{1}{2}))/(N_r-2)$, $\tilde{r}_1 = f(\frac{1}{2})$ and $\tilde{\vartheta}_1 = g(0)$. The constant α was chosen so that $\vartheta_1+2\pi = \vartheta_{N_\vartheta-1}$ and $\vartheta_2+2\pi = \vartheta_{N_\vartheta}$; due to the fact that the angular stretching decays exponentially fast away from $\vartheta_{\min}$ and $\vartheta_{\max}$ the constant α is a small number. (Notice that in order to obtain each grid point it is necessary to solve a nonlinear equation.) Figure 14 shows a typical mesh that can be obtained using these transformations.

We are now ready to describe the numerical discretization. Since there is no stretching in the rectangular mesh we use the usual five point formula to approximate the Laplacian operator. To define the discretization in the polar mesh we first transform to the stretched variables $\tilde{\vartheta}$ and $\tilde{r}$ to obtain:

$$\frac{\partial^2 u}{\partial \tilde{r}^2}\left(\frac{d\tilde{r}}{dr}\right)^2 + \frac{\partial u}{\partial \tilde{r}}\left(\frac{d^2\tilde{r}}{dr^2} + \frac{1}{r}\frac{d\tilde{r}}{dr}\right) + \frac{1}{r^2}\left[\frac{\partial^2 u}{\partial \tilde{\vartheta}^2}\left(\frac{d\tilde{\vartheta}}{d\vartheta}\right)^2 + \frac{\partial u}{\partial \tilde{\vartheta}}\frac{d^2\tilde{\vartheta}}{d\vartheta^2}\right] = 0\,. \tag{2.22}$$

We approximate the derivatives of the dependent variable u using standard centered second order accurate difference appproximations. The coefficients of this equation, which involve derivatives of the stretched variables with respect to the

physical variables, can be computed using the exact form of the transformation.

In Table 2 we show the numerical results obtained using four different meshes, each one generated using N_x, N_y, N_ϑ and N_r as in equation (2.17). The L_2-error is defined as

$$\varepsilon^2 = \sum_{i,j} e(x_i,y_j)^2 \, \delta x \, \delta y + \sum_{k,l} e(\vartheta_k,r_l)^2 \, r_l \, \delta\tilde{r} \, \delta\tilde{\vartheta} \, \frac{dr}{d\tilde{r}} \, \frac{d\vartheta}{d\tilde{\vartheta}} \qquad (2.23)$$

were $e(x,y)$ is the discrete error function. The parameters used for the stretching transformation are

$$\varepsilon_r = 0.0345 \quad \varepsilon_0 = 0.1111 \quad \varepsilon_1 = 0.3333 \quad \vartheta_{\min} = \pi/2 \quad \vartheta_{\max} = \tfrac{3}{4}\pi \, . \qquad (2.24)$$

In the case of $N_\vartheta = 65$ (*Mesh II*), 8 points were placed in the third quadrant (away from the stretched region), in this way its corresponding error has to be compared to the error obtained for *Mesh I* of the uniform meshes (Table 1).

In Figures 11a and 11b we show the exact solution and the computed solution for *Mesh IV* and $\omega = 4$ and in Figure 11c we show the contour plot of the numerical errors. Notice again that no numerical artifacts show up. In this case there are enough points to compute the solution accurately away from the stretched region and we do not see any difference in the nature of the error due to the stretching, this is not the case when we increase the number of modes of the solution. (We only show the polar meshes for convenience.)

Finally, in Figures 12a and 12b we show again the exact and computed solution for *Mesh IV* and $\omega = 9$ and in Figure 12c we show the corresponding error. Notice that the error in the stretched region is an order of magnitude smaller than the error in the third quadrant.

3. Some Stability Results

We devote this chapter to the study of the numerical stability of the composite mesh technique when applied to hyperbolic equations. We consider the following model problem in one space dimension

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 , \tag{3.1a}$$

for $-\infty < x < \infty$ and $t > 0$, with initial values given by

$$u(x,t=0) = u_0(x). \tag{3.1b}$$

The approach we use to study the stability of the numerical method when applied to this problem was introduced by Browning, Kreiss and Oliger [7].

We consider two uniform grids, each one with a possibly different mesh width; the grids are defined by (See Figure 1)

$$x_\nu = x_0 - \nu h_1 \quad \text{and} \quad y_\nu = y_0 + \nu h_2 \tag{3.2}$$

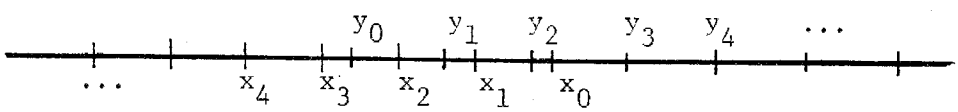


Figure 1

where $\nu = 0,1,\cdots\infty$ and where h_1 and h_2 are the mesh widths. The grids cover the entire real line and they overlap in the sense that $x_0 > y_1$ and $y_0 < x_1$. We also assume that end points of each grid do not correspond to a point on the opposite grid, that is we assume that x_0 is not a point on the y-grid and similarly for y_0. We consider the following second order centered approximation for the spatial operator of (3.1)

$$\frac{du_\nu}{dt}(t) - \frac{u_{\nu+1}(t) - u_{\nu-1}(t)}{2h_1} = 0 \,, \tag{3.3a}$$

$$\frac{dv_\mu}{dt}(t) + \frac{v_{\mu+1}(t) - v_{\mu-1}(t)}{2h_2} = 0 \,, \tag{3.3b}$$

for $\mu, \nu = 1,2, \cdots \infty$; with initial values given by

$$u_\nu(0) = u_0(x_\nu, t) \quad \text{and} \quad v_\mu(0) = u_0(y_\mu, t) \tag{3.3c}$$

for $\nu, \mu = 0,1 \cdots \infty$. We need interpolation formulae to connect the solution on each grid; they are defined by

$$u_0(t) = \alpha_k \, v_k(t) + \alpha_{k+1} \, v_{k+1}(t) \tag{3.3d}$$

$$v_0(t) = \beta_l \, u_l(t) + \beta_{l+1} \, u_{l+1}(t) \tag{3.3e}$$

where $y_k < x_0 < y_{k+1}$, $x_{l+1} < y_0 < x_l$, and

$$x_0 = \alpha_k \, y_k + \alpha_{k+1} \, y_{k+1} \quad \text{and} \quad y_0 = \beta_l \, x_l + \beta_{l+1} \, x_{l+1} \,. \tag{3.3f}$$

Because of these conditions on x_0, y_0 the constants involved in the interpolation formulae lie in the interval (0,1): that is $0 < \alpha_k, \alpha_{k+1}, \beta_l, \beta_{l+1} < 1$. Here

$$u_\nu(t) = \tilde{u}(x_\nu, t) \quad \text{and} \quad v_\mu(t) = \tilde{u}(y_\mu, t) \,, \tag{3.4}$$

define the numerical approximations to $u(x,t)$.

According to the theory developed by H.-O. Kreiss [19] and Gustafsson, Kreiss and Sundstrom [15] (often referred as the *GKS*-theory) the stability of the numerical approximation defined by (3.3a)-(3.3f) can be studied by introducing the resolvent problem defined by

$$2h_1 s\ \hat{u}_\nu - (\ \hat{u}_{\nu+1} - \hat{u}_{\nu-1}\ ) = 0\ , \tag{3.5a}$$

$$2h_2 s\ \hat{v}_\mu + (\ \hat{v}_{\mu+1} - \hat{v}_{\mu-1}\ ) = 0\ , \tag{3.5b}$$

$$\hat{u}_0 = \alpha_k\ \hat{u}_k + \alpha_{k+1}\ \hat{u}_{k+1} + \hat{f}_1\ , \tag{3.5c}$$

$$\hat{v}_0 = \beta_l\ \hat{v}_l + \beta_{l+1}\ \hat{v}_{l+1} + \hat{f}_2\ , \tag{3.5d}$$

for $\nu\ ,\ \mu = 1,2,\cdots\infty$. In the context of the *GKS*-theory equations (3.5c) and (3.5d) are called the reflection conditions. The method is stable if for any $Real(s) > 0$ there is a unique solution of equations (3.5a)-(3.5d) statisfying

$$\sum_{\nu=0}^{\infty} (|\hat{u}_\nu|^2 + |\hat{v}_\nu|^2) < \infty\ , \tag{3.5e}$$

and if that solution can be estimated in terms of the forcing functions:

$$|\hat{u}_0| + |\hat{v}_0| \le K\ (|\hat{f}_1| + |\hat{f}_2|)\ ; \tag{3.5f}$$

where the constant K does not depend on either s, $\hat{f}_1$ or $\hat{f}_2$.

Solutions of equations (3.5) can be found by introducing

$$\hat{u}_\nu = \rho\ \kappa^\nu \quad \text{and} \quad \hat{v}_\nu = \sigma\ \lambda^\nu\ , \tag{3.6}$$

for $\nu = 0,1,\cdots\infty$, where κ and λ are the roots with modulus smaller than one of

$$2\ h_1 s\ \kappa - (\ \kappa^2 - 1\ ) = 0 \quad \text{and} \quad 2\ h_2 s\ \lambda + (\ \lambda^2 - 1\ ) = 0\ . \tag{3.7}$$

When $Real(s) > 0$ we have

$$\kappa = h_1 s\ - \sqrt{1 + (h_1 s)^2}\ , \quad \text{and} \quad \lambda = -h_2 s\ + \sqrt{1 + (h_2 s)^2}\ . \tag{3.8}$$

The interpolation conditions can be written as follows:

$$\mathbf{C}(s) \begin{bmatrix} \rho \\ \sigma \end{bmatrix} = \begin{bmatrix} 1 & -\alpha_k\ \kappa^k - \alpha_{k+1}\ \kappa^{k+1} \\ -\beta_l\ \lambda^l - \beta_{l+1}\ \lambda^{l+1} & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \sigma \end{bmatrix} = \begin{bmatrix} \hat{f}_1 \\ \hat{f}_2 \end{bmatrix}\ . \tag{3.9}$$

In order to obtain the estimate (3.5f) we need to show that

$$|\det \mathbf{C}(s)| = |1 - (\beta_l \lambda^l + \beta_{l+1} \lambda^{l+1}) (\alpha_k \kappa^k + \alpha_{k+1} \kappa^{k+1})| \geq \delta > 0 , \quad (3.10)$$

for $Real(s) \geq 0$ where δ is some positive constant.

First we notice that $\kappa \to 0$ and $\lambda \to 0$ as $s \to \infty$; this implies that $\det \mathbf{C}(s) \to 1$ in the same limit. Thus we only need consider the behaviour of the determinant as $Real(s) \to 0$.

Since $0 < \beta_l, \beta_{l+1}, \alpha_k, \alpha_{k+1} < 1$ and $|\kappa|, |\lambda| \leq 1$ it follows that $\det \mathbf{C}(s) = 0$ can only hold when $\lambda = \kappa = 1$ (or equivalently when $s = 0$). But from equation (3.8) we have that $\lambda \to 1$ and $\kappa \to -1$ as $s \to 0$. The stability result follows immediately. (The stability result still holds if either β_k or β_{k+1} is equal to one.)

We now consider the numerical approximation to equation (3.1) defined by (3.3a)-(3.3c) where the grids defined by (3.2) are aligned in such a way that

$$x_0 = y_k \quad \text{and} \quad y_0 = x_l \qquad (3.11)$$

for some k and l (in such a case the ratio h_1/h_2 would be a rational number). If we now use the simple-minded interpolation between the meshes defined by

$$u_0(t) = v_k(t) \quad \text{and} \quad v_0(t) = u_l(t) \qquad (3.12)$$

then the determinant condition for this case is

$$|\det \mathbf{C}(s)| = |1 - \lambda^l \beta^k| \geq \delta > 0 . \qquad (3.13)$$

But for $s = 0$ we have $\det \mathbf{C}(0) = 1 - (-1)^l$ which is zero for l even. We have found that for this case it is not possible to draw a stability result. In this limiting case the numerical stability can be affected by the time discretization or the presence of boundaries.

We now show an example in which the use of the interpolation defined in equation (3.12) combined with the numerical approximation of equations (3.3a) and (3.3b) leads to an unstable method.

We again consider equation (3.1) this time defined for $t > 0$ and for $x \geq x_N$, where N is some natural number; we impose an homogeneous boundary condition at $x = x_N$, that is $u(x_N,t) = 0$ for all $t > 0$. The numerical approximation is defined on the grids introduced in equations (3.2); we consider $h_1/h_2 = 2$ and $x_0 = y_2$ and $y_0 = x_1$. (See Figure 2).

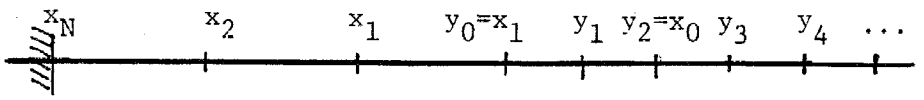


Figure 2

The numerical approximation we want to consider consists of equations (3.3a)-(3.3b) with Leap-Frog used to discretize in the time variable:

$$\frac{u_\nu^{m+1} - u_\nu^{m-1}}{2k} - \frac{u_{\nu+1}^m - u_{\nu-1}^m}{2h_1} = 0 , \tag{3.14a}$$

$$\frac{v_\mu^{m+1} - v_\mu^{m-1}}{2k} + \frac{v_{\mu+1}^m - v_{\mu-1}^m}{2h_2} = 0 , \tag{3.14b}$$

where $\nu = 1,2,\cdots N$, $\mu = 1,2,\cdots \infty$ and k is the time step, with the boundary condition

$$u_N(t) = 0 , \tag{3.14c}$$

and the interpolation formulae

$$u_0(t) = v_2(t) \quad \text{and} \quad v_0(t) = u_1(t) . \tag{3.14d}$$

The corresponding initial values given by equation (3.3c).

$$u_\nu^0 = u_0(x_\nu) \quad \text{and} \quad v_\mu^0 = u_0(y_\mu) , \tag{3.14e}$$

for $\mu = 1,2,\cdots\infty$ and $\nu = 1,2,\cdots N$. We are only interested in solutions that belong to l_2, that is

$$\sum_{\mu=0}^{\infty} |v_\mu^m|^2 < \infty \tag{3.14f}$$

for all $m \geq 0$.

We want to show that the problem defined in (3.14) admits a solution which grows exponentially at a rate which depends on N. We mention that the instability of the method is a result of the combination of the boundary at $x = x_N$ and the use of Leap-Frog as a discretization in time; Lax-Wendroff in the same situation will lead to a stable method.

We seek a solution of equations (3.14) of the type

$$u_\nu^m = (\rho_1 \kappa_1^\nu + \rho_2 \kappa_2^\nu) z^m , \tag{3.15a}$$

$$v_\mu^m = \sigma \lambda^\nu z^m . \tag{3.15b}$$

Here κ_1, κ_2 and λ are determined by

$$\kappa^2 - 1 - \frac{h_1}{k} \frac{z^2-1}{z} \kappa = 0 , \tag{3.16a}$$

$$\lambda^2 - 1 + \frac{h_2}{k} \frac{z^2-1}{z} \lambda = 0 . \tag{3.16b}$$

Recall that, since equation (3.14f) must be satisfied, we have to pick the root λ which satisfies $|\lambda| < 1$. The boundary condition at $x = x_N$ implies that

$$u_\nu^m = \rho (\kappa_1^{\nu-N} - \kappa_2^{\nu-N}) z^m \tag{3.17a}$$

$$v_\mu^m = \sigma \lambda^\mu z^m ; \tag{3.17b}$$

and if we now impose the interpolation conditions on (3.15) we then obtain

$$\mathbf{C}(z)\begin{bmatrix}\rho\\ \sigma\end{bmatrix} = \begin{bmatrix}\kappa_1^{1-N} - \kappa_2^{1-N} & -1\\ \kappa_1^{-N} - \kappa_2^{-N} & -\lambda^2\end{bmatrix} = \underline{0}. \qquad (3.18)$$

Using that $\kappa_1 \cdot \kappa_2 = -1$ the condition $\det \mathbf{C}(z) = 0$ can be written as

$$(\,1 - (-1)^N \kappa_1^{2N}\,) - \kappa_1 \lambda^2\,(\,1 + (-1)^N \kappa_1^{2N-2}\,) = 0\,. \qquad (3.19)$$

In order to show that the method is unstable, we need to find a root z of equation (3.19) with $|z| > 1$. The stability analysis for the model problem suggests that instability may occur at λ close to -1. From equation (3.16b) it follows that this situation corresponds to $z^2 - 1 \approx 0$ and $Real(z - 1/z) < 0$, which implies that $z \approx -1$. In this case we have

$$\lambda = -\frac{h_2}{2k}(z - 1/z) - (\,(\frac{h_2}{2k}(z - 1/z))^2 + 1)^{1/2}\,, \qquad (3.20a)$$

and we can always consider κ_1 to be the root defined by

$$\kappa_1 = \tfrac{1}{2}\frac{h_1}{k}(z - 1/z) - (\,(\tfrac{1}{2}\frac{h_1}{k}(z - 1/z))^2 + 1)^{1/2}\,. \qquad (3.20b)$$

Introduce $s = \frac{h_2}{k}(z - 1/z)$; under the assumption that s is small (we are considering $z^2 - 1$ to be small) we can write

$$\lambda = -e^{\frac{1}{2}s + \mathbf{O}(s^3)} \quad \text{and} \quad \kappa_1 = e^{s + \mathbf{O}(s^3)}\,. \qquad (3.21)$$

We now determine asymptotically the roots of equation (3.19). For N even the roots of (3.19) are almost roots of the following equation

$$\tilde{D}(s) = (\,1 - e^{2Ns}\,) - e^{2s}\,(\,1 + e^{(2N-2)s}\,) = 1 - 2\,e^{2Ns} - e^{2s} = 0\,. \qquad (3.22)$$

We notice that $\tilde{D}(0) = -2$ and $\tilde{D}(-\infty) = 1$, therefore there exists a root s_N, with $Real(s_N) < 0$ which behaves like

$$s_N \sim -\frac{\log N - \log 2}{2N}, \tag{3.23}$$

as $N \to \infty$. In this way we have obtained a root z_N of $\det \mathbf{C}(z) = 0$, the root behaves

$$z_N \sim -1 - \frac{k}{h_2}\frac{\log N}{8N} \quad as\ N \to \infty\ . \tag{3.24}$$

So the growth of this mode is

$$z^{t/k} \sim (-1)^{t/k}\, e^{\frac{\log N}{4Nh_1}t} = e^{\log(N)\frac{t}{4(x_0 - x_N)}}, \tag{3.25}$$

which clearly deteriorates as N increases. Hence the approximation defined by (3.14) is unstable.

In Figure 13a we show the solution of the numerical approximation defined in (3.14); the calculations were performed in the interval [0,2] . We use homogeneous initial data, and boundary conditions given by $u_N(t) = u(x=0,t) = \sin(2\pi\omega(t-\frac{1}{4}))$. In Figure 13b we show the numerical solution when the problem is considered in the interval [0,3]; we notice that the instability corresponds to the interface and the left boundary, while the solution is not affected by the conditions on the right boundary. (We use one sided difference schemes at this new boundary.) Here h_1 was taken to be 0.10 and $k/h_2 = 0.90$; the predicted ± 1 oscillation in time was observed, though for clarity we only show every other time step.

In order to avoid this instability, we can use a three point formula of the type

$$u_0(t) = \alpha_{k-1}\, v_{k-1}(t) + \alpha_k\, v_k(t) + \alpha_{k+1}\, v_{k+1}(t)\ , \tag{3.26a}$$

$$v_0(t) = \beta_{l-1}\, u_{l-1}(t) + \beta_l\, u_l(t) + \beta_{l+1}\, u_{l+1}(t)\ , \tag{3.26b}$$

where the coefficients α and β are picked to be positive, instead of (3.12). (A similar analysis can be used to show that these interpolation formulae define a stable method.)

The model problem we have just finished analyzing suggests that if we are using a composite mesh technique in the numerical approximation of a hyperbolic equation a three point formula such as (3.26) would be appropriate.

Our simple proof for the stability of the numerical method defined in (3.3) is based on the coefficients of the interpolation procedure (3.3d) and (3.3e) being positive; this condition does not hold for higher order interpolation formulae. Starius [27] proves that the composite mesh technique is stable when applied to equation (3.1) defined in the semiinfinite interval $[x_N,\infty)$. He considers the Lax-Wendroff approximation on each grid and a general interpolation formula of the type:

$$u_0(t) = \sum_{r=l-b_u}^{k+t_u} \alpha_r \, v_r(t) \,, \tag{3.27a}$$

$$v_0(t) = \sum_{s=l-b_v}^{k+t_v} \beta_s \, v_s(t) \,, \tag{3.27b}$$

to connect the solutions on each grid. He proves that this method is stable, in the norm considered by Kreiss, Gustafsson and Sundstrom [15], when either the dissipation of the method is large enough (that is for $k/h_1 = \mathbf{O}(1)$, where k is the time step), or when the width of the overlapping segment $[y_0,x_0]$ is independent of the mesh widths. More work is needed to determine which combinations of interpolation procedures and numerical approximations on each grid lead to stable methods.

4. Ocean Circulation Model

In this chapter we apply the composite mesh technique to a model of a wind driven ocean in a circular basin. The model describes a homogeneous ocean, that is, no density stratification is present, with the wind stress acting on the surface as the only source of motion. It ignores possible effects of the bottom topography and assumes the β-plane approximation which is only valid at mid-latitudes.

The dimensionless formulation of the model is described by the following mixed hyperbolic-elliptic initial boundary value problem

$$\frac{\partial \zeta}{\partial t} + R_o \left(u \frac{\partial \zeta}{\partial x} + v \frac{\partial \zeta}{\partial x} \right) + v = -S - \delta \zeta + E_k \left(\frac{\partial^2 \zeta}{\partial x^2} + \frac{\partial^2 \zeta}{\partial y^2} \right), \qquad (4.1a)$$

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = \zeta , \qquad (4.1b)$$

$$u = -\frac{\partial \psi}{\partial y} \qquad v = \frac{\partial \psi}{\partial x} . \qquad (4.1c)$$

for $x^2+y^2 \leq 1$ and $t > 0$. The boundary conditions, expressed in polar coordinates, are given by

$$\psi(r=1) = \frac{\partial \psi}{\partial r}(r=1) = 0 ; \qquad (4.1d)$$

the usual initial values are the rest state: $\zeta(x,y,t=0) \equiv 0$.

In the equation the dependent variable ζ denotes the usual vorticity, ψ the stream function, R_o a Rossby number, E_k an Ekman number, S the curl of the wind stress and δ corresponds to the bottom friction. (R_o, E_k and δ are the parameters which determine the specific problem.) The Rossby number measures the relative importance of the inertial acceleration to the Coriolis acceleration. The Ekman number compares the effects of the viscosity and the Coriolis terms; for large scale motions the value of E_k is estimated using A_H, the

horizontal kinematic eddy viscosity. The corresponding Reynolds number for this flow, defined as $R_e = R_o/E_k$, is always larger than one. For large scale dynamics E_k is $\mathbf{O}(10^{-4})$, or smaller depending on the values of A_H, and R_o is $\mathbf{O}(10^{-3})$. The bottom friction coefficient δ is $\mathbf{O}(10^{-2})$ when computed using the average depth of the main thermocline of the ocean. The scaling used for the velocity is such that the curl of the wind stress is $\mathbf{O}(1)$.

Pedlosky [22] and Greenspan [14] describe the main features of this homogeneous model for the ocean circulation. Equations (4.1) also arise in the description of a rapidly rotating cylindrical container with a sloping bottom where the wind stress term is simulated using a lid which rotates at a slightly larger angular velocity. The experiment can be easily performed and it is used as a laboratory model of the ocean. (See Beardsley and Robbins [6].)

The solutions of equations (4.1) are known to develop a thin layer of high vorticity in the western semicircle of the basin, which in the ocean corresponds to the presence of the Gulf Stream, the Kuroshio Current and other sea currents. The model is successful at predicting the observed separation of these currents from the coastline.

In our numerical approximation we need to generate a mesh on which the boundary layer can be resolved, we also need high resolution in the angular direction in the second quadrant ($\pi \leq \vartheta \leq \pi/2$) where the separation occurs.

Beardsley [4] and Beardsley and Robbins [6] considered discretizations of equation (4.1) with meshes generated in polar coordinates; they considered spectral methods in the angular direction and an appropriate stretching in the radial direction. The mesh they consider has the usual clustering of points near the origin where no resolution is needed. We approach the problem using the composite mesh technique with angular and radial stretching as described in Example 3 of the chapter 2; this method allows us to locate mesh points only

where they are necessary.

The composite mesh we consider consists of the regular cartesian mesh defined in (2.5a) and (2.6a) and the polar grid defined using the stretching functions presented in equations (2.20a) and (2.20b). The angular part of the polar mesh was defined in (2.21f); in order to facilitate the implementation of the no-slip boundary condition, we consider a radial grid that does not include the boundary as a grid line:

$$\tilde{r}_l = \tilde{r}_1 + (l - 1/2)\delta\tilde{r} = f(r_l)\,, \tag{4.2}$$

where $\delta\tilde{r} = (f(1) - f(1/2))/N_r$ and $\tilde{r}_1 = f(1/2)$. (Notice that $r_{N_r} < 1 < r_{N_r+1}$.) Figure 14 shows such a mesh.

As in our previous numerical experiments we consider an explicit numerical approximation in time and standard second order accurate centered approximations for the spatial operators of equation (4.1). In the time discretization, we use Leap-Frog for the hyperbolic and lower order terms of (4.1a), while the dissipation term is evaluated using forward Euler.

We consider a two time level approximation: given the vorticity at $t = m\delta t$ and $t = (m-1)\delta t$ and the stream function at $t = m\delta t$, we define the values of the stream function and the the vorticity at a new time level: $t = (m+1)\delta t$. We start first with the interior mesh

$$\frac{\zeta_{i,j}^{m+1} - \zeta_{i,j}^{m-1}}{2\delta t} \tag{4.3}$$

$$+ R_o \left(- \frac{\psi_{i+1,j}^m - \psi_{i-1,j}^m}{2\delta x} \frac{\zeta_{i,j+1}^m - \zeta_{i,j-1}^m}{2\delta y} + \frac{\psi_{i,j+1}^m - \psi_{i,j-1}^m}{2\delta y} \frac{\zeta_{i+1,j}^m - \zeta_{i-1,j}^m}{2\delta x} \right)$$

$$+ \frac{\psi_{i+1,j}^m - \psi_{i-1,j}^m}{2\delta x} = -S_{i,j}^m - \delta \zeta_{i,j}^m$$

$$+ E_k \left(\frac{\zeta_{i+1,j}^{m-1} - 2\zeta_{i,j}^{m-1} + \zeta_{i-1,j}^{m-1}}{\delta x^2} + \frac{\zeta_{i,j+1}^{m-1} - 2\zeta_{i,j}^{m-1} + \zeta_{i,j-1}^{m-1}}{\delta y^2} \right)$$

for $i=2,3,\cdots,N_x-1$ and $j=2,3,\cdots,N_y-1$. (It is possible to define a second order accurate formula in time for the parabolic terms using the vorticity at the previous time level: $t=(m-2)\,\delta t$. The Ekman number is a very small parameter and usually it is not necessary to use a second order accurate formula to approximate the time derivative of the viscous term.)

We use a similar formula to update the values of the vorticity on the polar mesh $v_{k,l}$, for $l=2,3,\cdots N_r-1$ and $k=2,3,\cdots N_\vartheta-1$. In order to complete the definition of the approximation to (4.1a) we need to consider the interpolation formulae and ϑ-periodicity conditions.

The Laplacian operator is approximated using the standard five point formula

$$\frac{\psi_{i+1,j}^{m+1} - 2\psi_{i,j}^{m+1} + \psi_{i-1,j}^{m+1}}{\delta x^2} + \frac{\psi_{i,j+1}^{m+1} - 2\psi_{i,j}^{m+1} + \psi_{i,j-1}^{m+1}}{\delta y^2} = \zeta_{i,j}^{m+1}, \tag{4.4a}$$

for $i=2,3,\cdots N_x-1$ and $j=2,3,\cdots,N_y-1$. For the polar mesh we have (for simplicity we write the equation when no stretching is used)

$$\frac{\psi_{k+1,l}^{m+1} - 2\psi_{k,l}^{m+1} + \psi_{k-1,l}^{m+1}}{\delta x^2} + \frac{1}{r_l} \frac{\psi_{k,l+1}^{m+1} - \psi_{k,l-1}^{m+1}}{2\delta r} \tag{4.4b}$$

$$+ \frac{1}{r_l^2} \frac{\psi_{k,l+1}^{m+1} - 2\psi_{k,l}^{m+1} + \psi_{k,l-1}^{m+1}}{\delta y^2} = \zeta_{k,l}^{m+1},$$

for $k=2,3,\cdots N_\vartheta-1$ and $r=2,3,\cdots N_r-1$. We again include the interpolation formulae and the ϑ-periodicity.

Equations (4.4) specify a large linear system of equations for the discrete values of the stream function which can be solved once we impose the necessary boundary conditions. We now use the location of the boundary between two grid lines to define a simple second order approximation to the boundary conditions by

$$\psi_{k,N_r}^{m+1} = 0 \quad \text{and} \quad \psi_{k,N_r+1} = 0 \tag{4.5}$$

for $k=1,2,\cdots N_\vartheta$. Using the first of these conditions, we can solve for the new values of the stream function.

We still need to make the tangential velocity zero along the boundary. Finally, we need to define new values for the vorticity at $l=N_r$ (that is, the vorticity at the wall). It is in obtaining these values that we use the no-slip condition: we write equation (4.2) for $l=N_r$ using the boundary condition (4.6) to obtain

$$\frac{\psi_{k,N_r}^{m+1}}{\delta r^2} + \frac{1}{r_{N_r}} \frac{\psi_{k,N_r}^{m+1}}{2\delta r} = \zeta_{k,N_r} \tag{4.6}$$

which is used to define the right hand side.

We now show the results obtained using the method just described. In the numerical experiment we consider the following parameters in equation (4.1)

$$R_0 = 2.702\times10^{-3} \quad E_k = 5.702\times10^{-5} \quad \delta = 2.531\times10^{-2} \tag{4.7}$$

and $S\equiv 1$. (The parameters were taken from Beardsley and Robbins [6] who chose them for comparison with experimental data which were available.)

The mesh is defined using

$$\varepsilon_r = 0.0345 \quad \varepsilon_0 = 0.2000 \quad \varepsilon_1 = 0.4472 \tag{4.8a}$$

$$N_r = 25 \quad N_\vartheta = 70 \quad N_x = 23 \quad N_y = 23\,. \tag{4.8b}$$

The minimum distance between two consecutive mesh circles is $\delta r_{\min} = 2.19\times10^{-3}$, and the maximum $\delta r_{\max} = 6.20\times10^{-2}$, the ratio of these distances corresponds to the stretching factor ε_r; there are 16 mesh circles with radius greater than 0.90. In the angular direction we have $\delta\vartheta_{\min} = 3.44\times10^{-2}$ and $\delta\vartheta_{\max} = 1.38\times^{-1}$, with corresponding ratio ε_0. (The mesh obtained is shown in Figure 14.)

The method we describe is explicit in time, which immediately introduces an upper limit for the time step. The problem we are solving is non linear which makes it difficult to estimate this upper bound. By numerical experimentation we found that the stability limit is $k < 0.04$; the results we show were obtained using $k = 0.025$. The calculations were performed on the Cray-1 computer at the National Center for Atmospheric Research in Boulder, Colorado with each time iteration taking 0.06 seconds.

We consider the rest state as initial values. As time increases from 0, the curl of the wind stress generates the motion by producing vorticity throughout the ocean basin. To accomodate the no-slip condition, a thin viscous layer develops along the boundary of the basin; the layer persists in time. In the interior of the basin a smooth flow develops where the Coriolis effects and the wind stress are balanced; the flow is usually referred to as the Sverdrup balance. Vorticity is transported from the interior to the west of the basin by long wavelength Rossby modes. The presence of these modes is a direct consequence of the north-south variation of the Coriolis force. (For a full discussion of the model see Greenspan [14] and Pedlosky [22].)

As the system evolves in time, energy piles up along the western boundary and a strong current or jet develops. Nonlinear effects now become important: the flow is no longer symmetric with respect to the east-west axis and the current intensifies towards the north. The width of the viscous layer is smaller than that of the jet which is due purely to inertial effects and as a result of an existing adverse pressure gradient along the wall the viscous layer separates.

Finally, if the friction terms and the dissipation of vorticity along the western boundary are strong enough to balance the vorticity generated by the wind stress, a steady state is reached. For large values of R_0/δ^2 (depending on E_k), the steady flow is no longer stable, a small region of recirculation appears and the flow becomes periodic in time. (The characteristic time scale for these processes is $1/\delta$.)

In Figures 15a-15c we show contour plots of the numerical values of the stream function and the vorticity obtained at $t=25$. Notice the high concentration of vorticity and the formation of the jet along the western portion of the boundary; notice also the smoothness of the solution in the interior of the basin and the difference in thickness of the boundary layers corresponding to the stream function and the vorticity respectively.

Figures 16a-16c show the solution at $t=50$ where the jet has already separated. Finally, in Figures 17a-17c we show the flow at $t=100$ which is now slowly evolving. Figures 17c-17d show the contour lines of the solution in the stretched variables $\tilde{\vartheta}$ and $\tilde{r}$: sharp gradients have been properly resolved. (Recall that there are 25 grid points in the radial direction and 70 points in the angular direction.) The program was not run long enough to reach the steady state which is estimated to occur at $t \approx 5/\delta$.

In order to check the accuracy of the solution, we increased N_ϑ, the number of points in the angular direction, to 128 keeping the other parameters that

define the mesh unchanged. The results obtained were unchanged to within 0.19% when the same time step was used. Each time iteration for this new mesh took 0.10 seconds.

The constraint $k < 0.05$ is impractical since the time scale on which the numerical solution evolves is $\mathbf{O}(10)$. Our next step therefore is to use the solution to study which terms should be computed implicitly in time to get a stability limit on the time step at least of $\mathbf{O}(1)$.

The local stability conditions due to the advection terms are of the form

$$\frac{kR_o|\underline{u}|}{\delta^2} < 1 , \tag{4.9a}$$

where $|u|$ is the local magnitude of the velocity field and δ is a local mesh size. In the interior mesh this condition translates into $k < 1$. Using the numerical values for $|u|$ over the polar mesh we obtain a constraint of the same order for the time step.

For the diffusion terms the stability limit is

$$\frac{2kE_k}{\delta^2} < 1 , \tag{4.9b}$$

and in the polar grid the minimum upper bound occurs at the boundary with $k < 0.04$.

This suggests a more practical numerical approximation to (4.1). The method consists in updating the vorticity in the interior using the previous explicit scheme in time. We then interpolate to obtain the values of the vorticity in the innermost circle of the polar grid. In order to update the vorticity on the annulus we use the following discretization in time (written in Cartesian coordinates)

$$\frac{1}{2k}\left(\zeta^{m+1}-\zeta^{m-1}\right)+R_o\left(\tfrac{1}{2}u^m\left(\frac{\partial\zeta^{m+1}}{\partial x}+\frac{\partial\zeta^{m-1}}{\partial x}\right)+\tfrac{1}{2}v^m\left(\frac{\partial\zeta^{m+1}}{\partial y}+\frac{\partial\zeta^{m-1}}{\partial y}\right)\right) \quad (4.10)$$

$$-v^m=-S^m-\tfrac{1}{2}\delta\left(\zeta^{m+1}+\zeta^m\right)+\tfrac{1}{2}E_k\left(\Delta\zeta^{m+1}+\Delta\zeta^{m-1}\right),$$

which is a linear equation, related to the biharmonic, for the new values of the vorticity ζ^{m+1}. The boundary values correspond to the the usual no-slip condition, that is, if we solve

$$\Delta\,\psi^{m+1}=\zeta^{m+1} \quad \text{with} \quad \psi^{m+1}(r=1)=0\,,$$

we should obtain $\psi_r^{m+1}(r=1)\equiv 0$.

In order to solve for ζ^{m+1}, we consider an iteration scheme which involves solving the Laplacian over the composite mesh and an elliptic problem on the annulus. This iteration scheme is presented in Israeli [16]. In the iteration, we guess the new values of the vorticity on the wall $\tilde{\zeta}_w^{(0)}$ using an extrapolation procedure. With these values we compute for the vorticity $\tilde{\zeta}^{(1)}$ on the annulus and then the corresponding stream function $\tilde{\psi}^{(1)}$ using the boundary conditions $\tilde{\psi}(r=1,\vartheta)\equiv 0$. We now improve the guess for the vorticity on the wall using

$$\tilde{\zeta}_w^{(1)}(\vartheta)=\tilde{\zeta}_w^{(n)}+K\,\frac{\partial\tilde{\psi}^{(1)}}{\partial r}(r=1,\vartheta)\,, \quad (4.12)$$

where the constant K has to be appropriately chosen. (See Israeli [16].)

We plan to develop a computer code using this method and use it to examine the various assumptions that have been made regarding the solution of this ocean model.

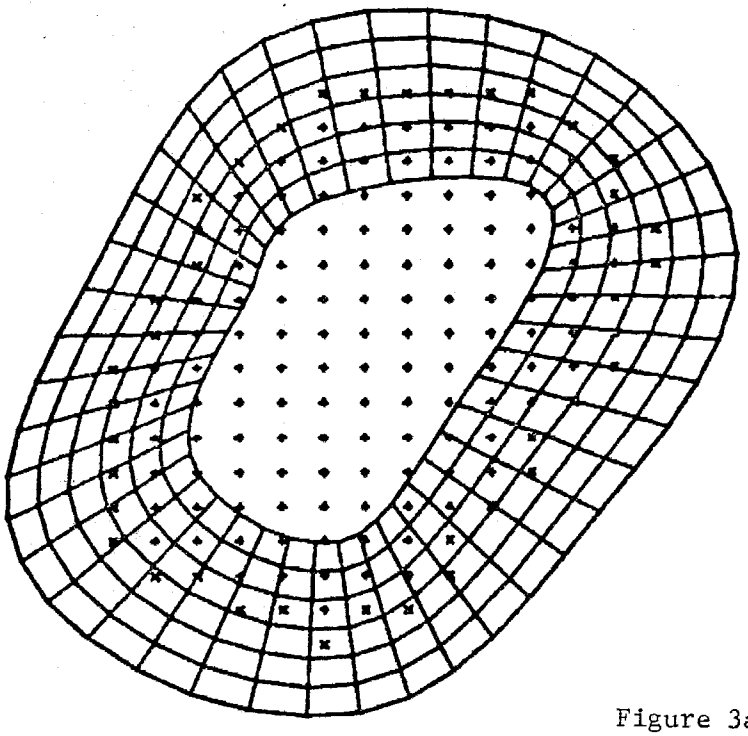


Figure 3a

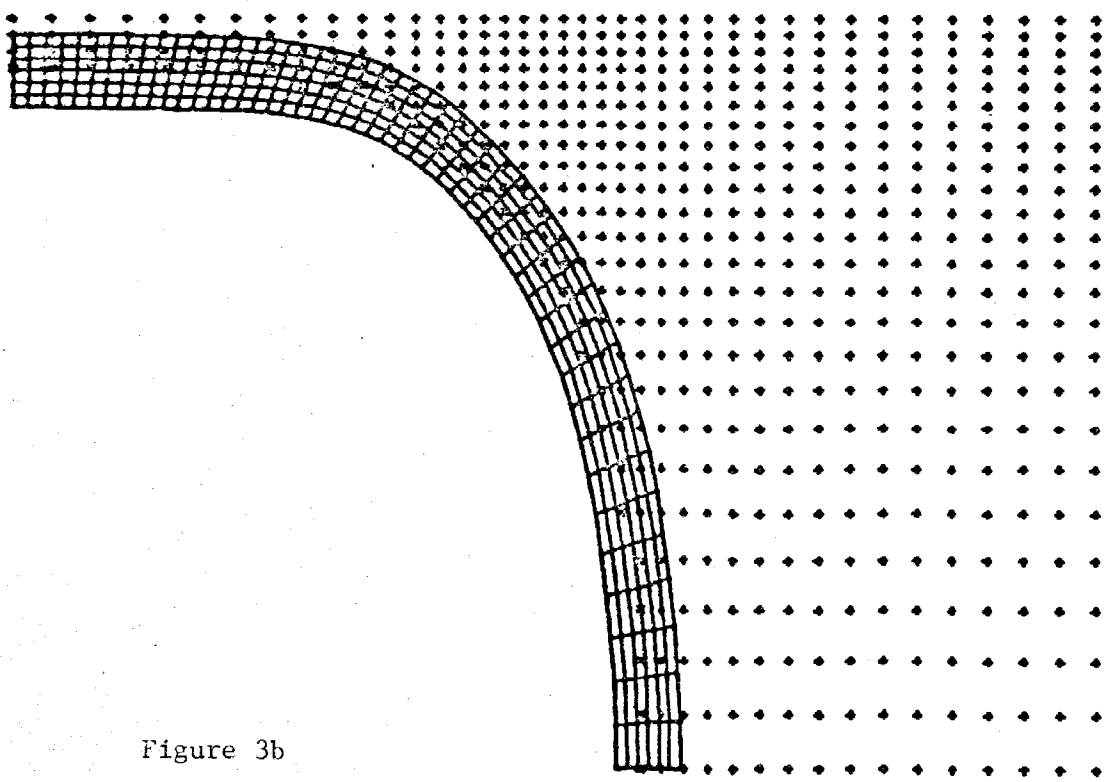


Figure 3b

T = 0.0 N = 0

cartesian mesh

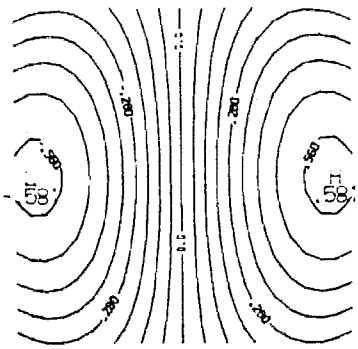


Figure 4a

INITIAL VALUES

T = 0.0 N = 0

polar mesh

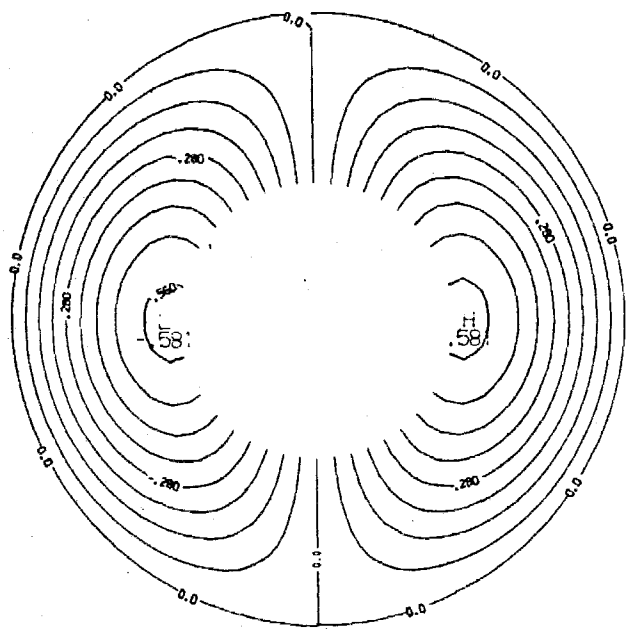


Figure 4b

contour from -0.5600 to 0.5600
contour interval of 0.70e-01

T = 9.0 N = 600 WAVE EQUATION

cartesian mesh

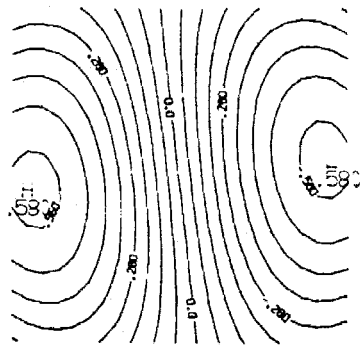


Figure 5a

COMPUTED SOLUTION

T = 9.0 N = 600

polar mesh

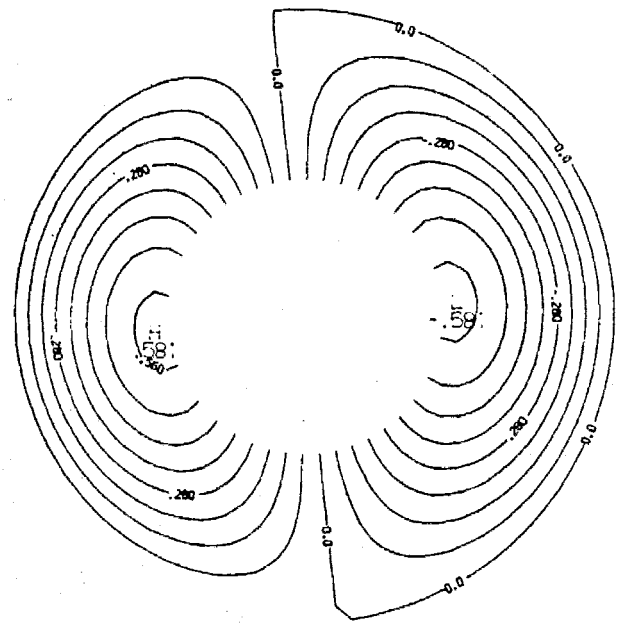


Figure 5b

contour from -0.5600 to 0.5600

contour interval of 0.70e-01

T = 9.0 N = 600 WAVE EQUATION

cartesian mesh

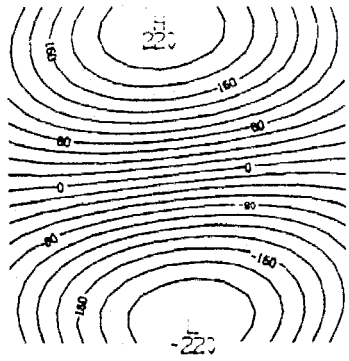


Figure 6a

COMPUTATIONAL ERROR

T = 9.0 N = 600

polar mesh

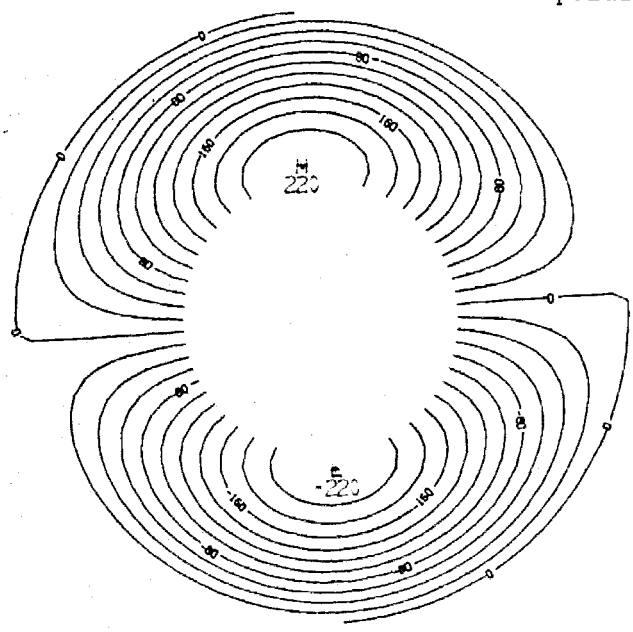


Figure 6b

contour from -0.22e-01 to 0.22e-01

contour interval of 0.87e-03

T = 9.42 N = 600

RIGID ROTATION

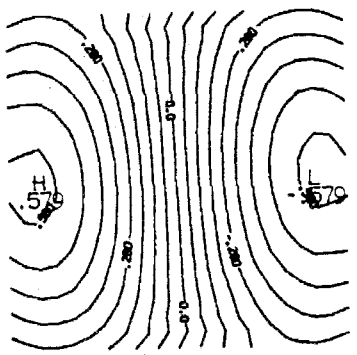


Figure 7a

COMPUTED SOLUTION

T = 9.42 N = 600

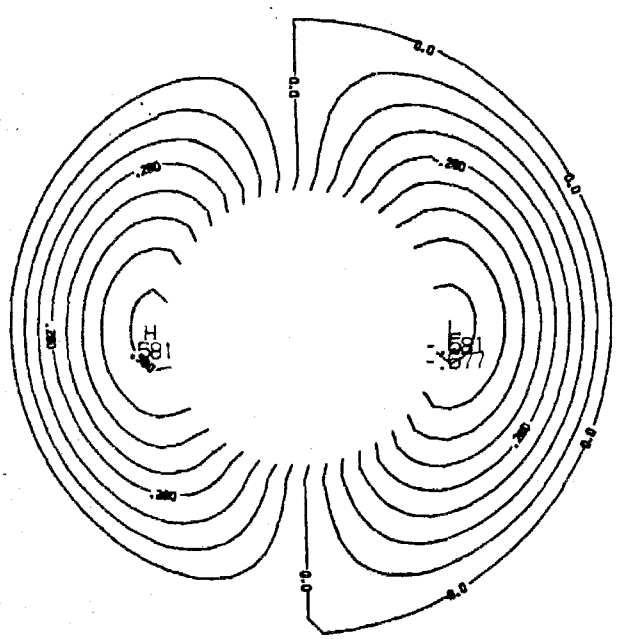


Figure 7b

contour from -0.5600 to 0.5600

contour interval of 0.70e-01

T = 9.42 N = 600 RIGID ROTATION

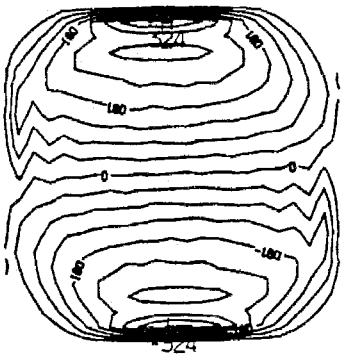


Figure 8a

contour from -0.32e-01 to 0.32e-01
contour interval of 0.40e-02

COMPUTATIONAL ERROR

T = 9.42 N = 600

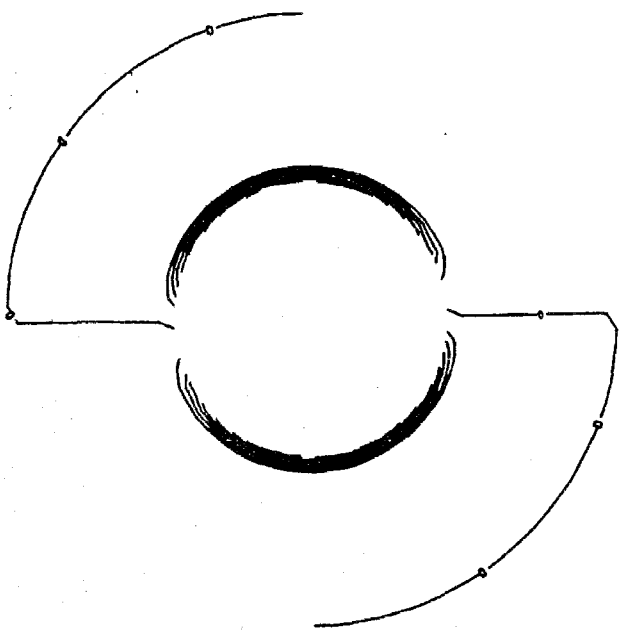


Figure 8b

contour from -0.24e-01 to 0.24e-01
contour interval of 0.30e-02

T = 9.42 N =1200

RIGID ROTATION

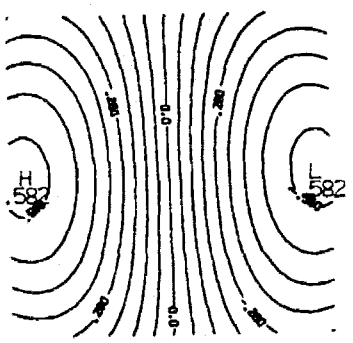


Figure 9a

COMPUTED SOLUTION

T = 9.42 N =1200

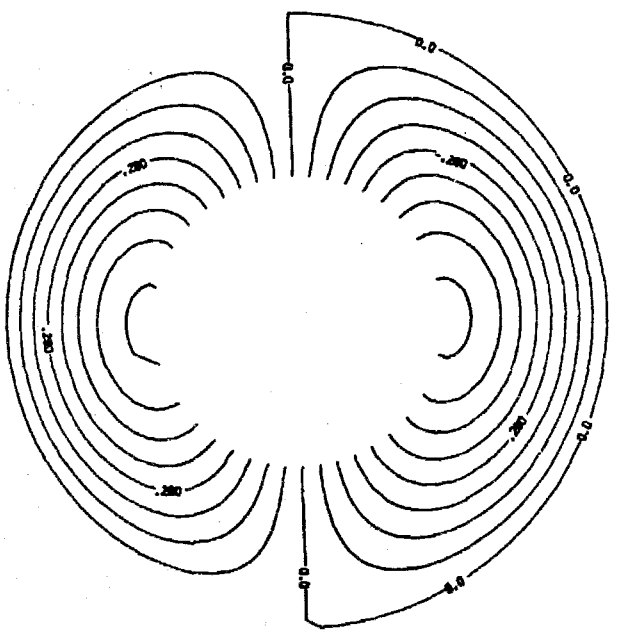


Figure 9b

contour from -0.5600 to 0.5600

contour interval of 0.70e-01

T = 9.42 N =1200 RIGID ROTATION

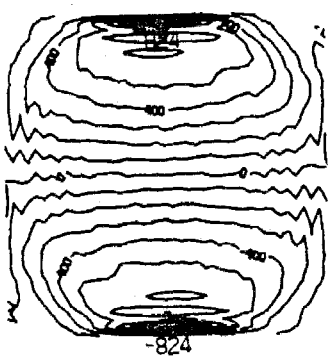


Figure 10a

contour from -0.90e-02 to 0.90e-02
contour interval of 0.10e-02

COMPUTATIONAL ERROR

T = 9.42 N =1200

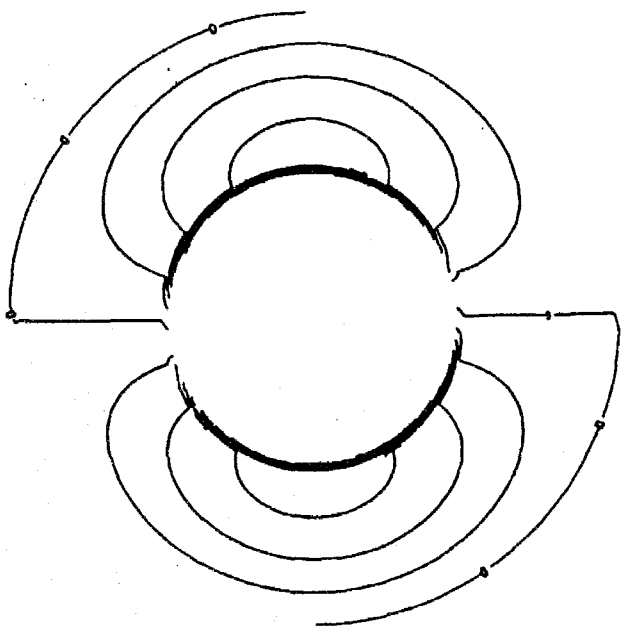


Figure 10b

contour from -0.63e-02 to 0.63e-01
contour interval of 0.70e-03

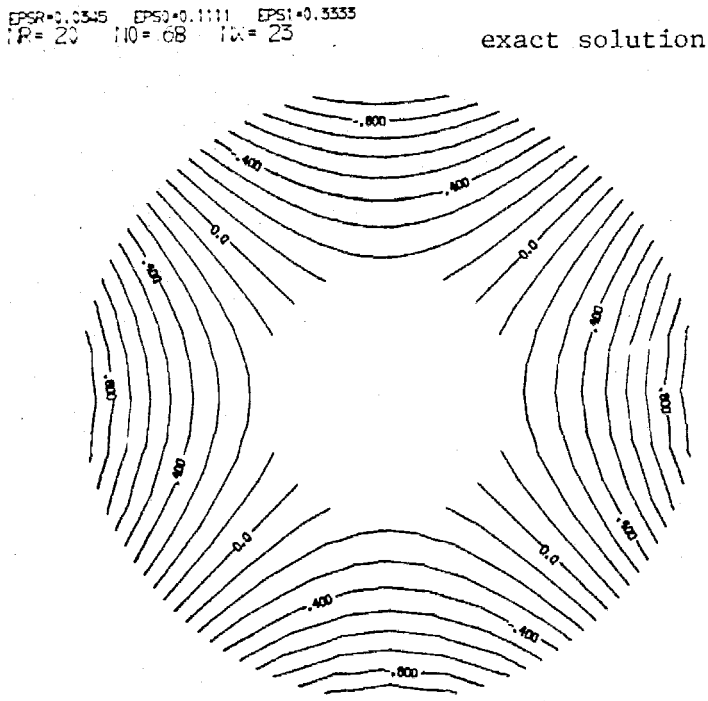


Figure 11a

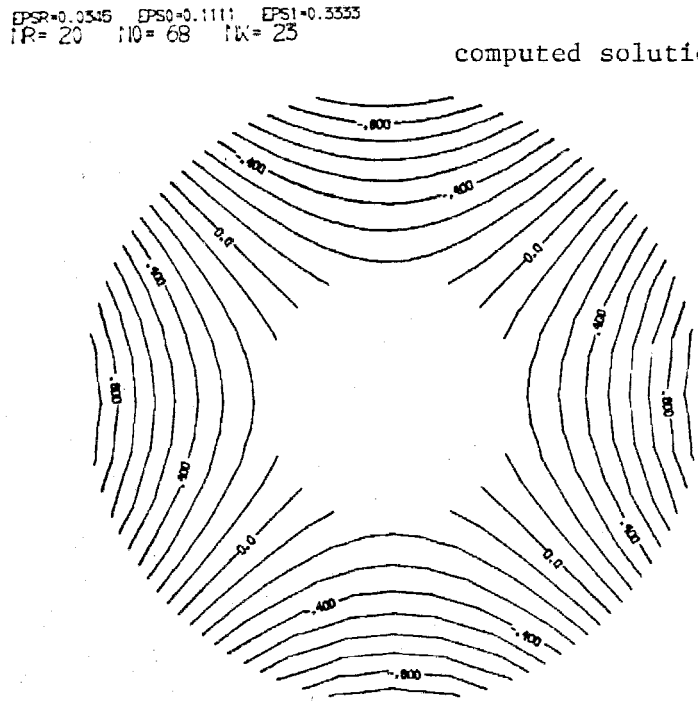


Figure 11b

contour from 0.9000 to 1.0000

contour interval of 0.1000

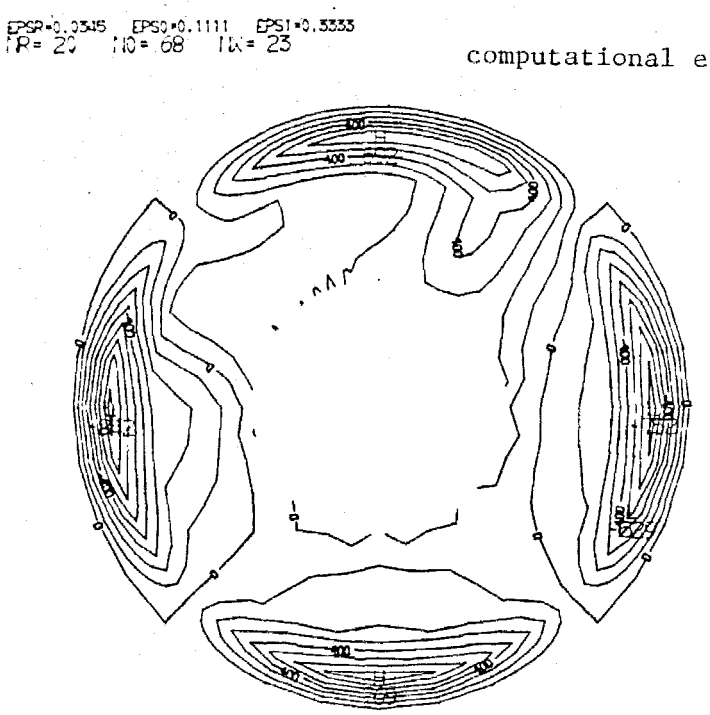


Figure 11c

contour from -0.80e-02 to 0.70e-02

contour interval of 0.10e-02

EPSR=0.0545 EPS0=0.1111 EPS1=0.3333
NR= 20 N0= 68 NK= 23

exact solution

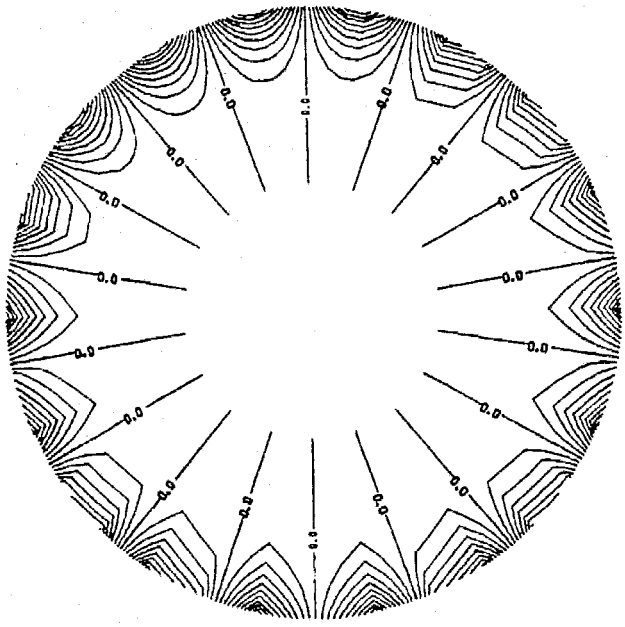


Figure 12a

EPSR=0.0545 EPS0=0.1111 EPS1=0.3333
NR= 20 N0= 68 NK= 23

computed solution

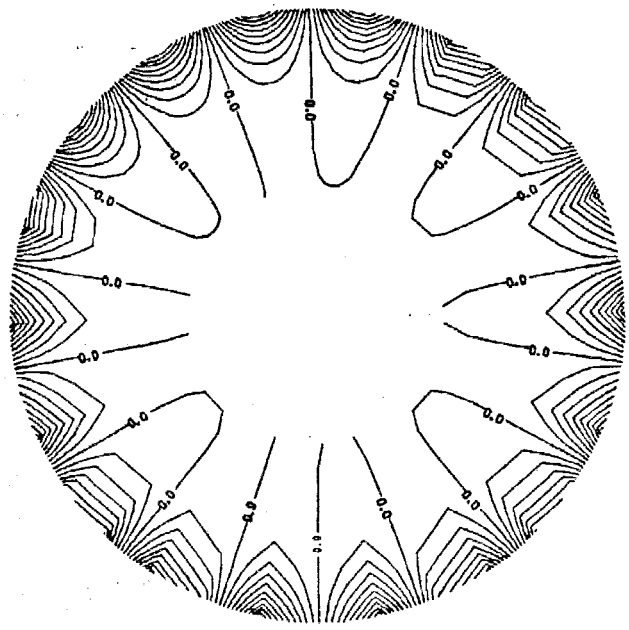
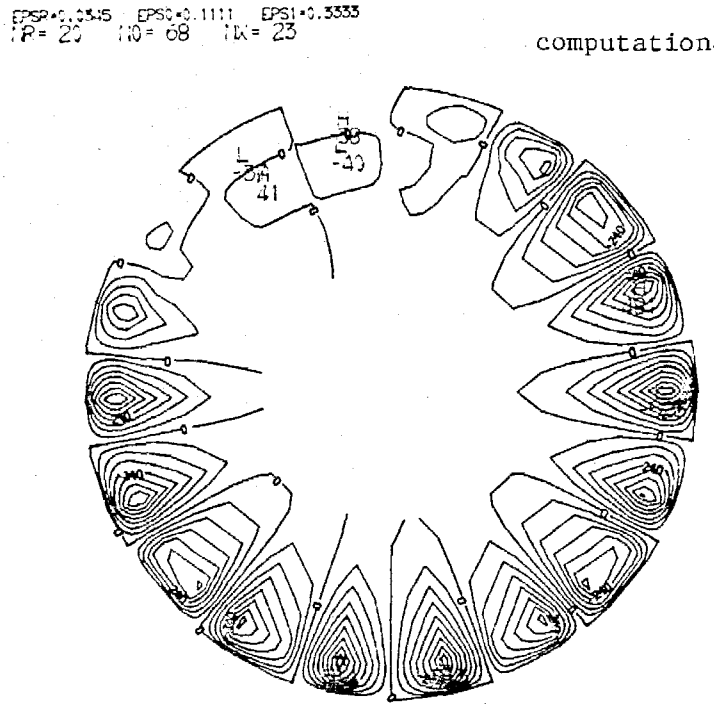


Figure 12b

contour from -0.9000 to 1.0000

contour interval of 0.1000



computational error

Figure 12c

contour from -0.48e-01 to 0.48e-01

contour interval of 0.60e-02

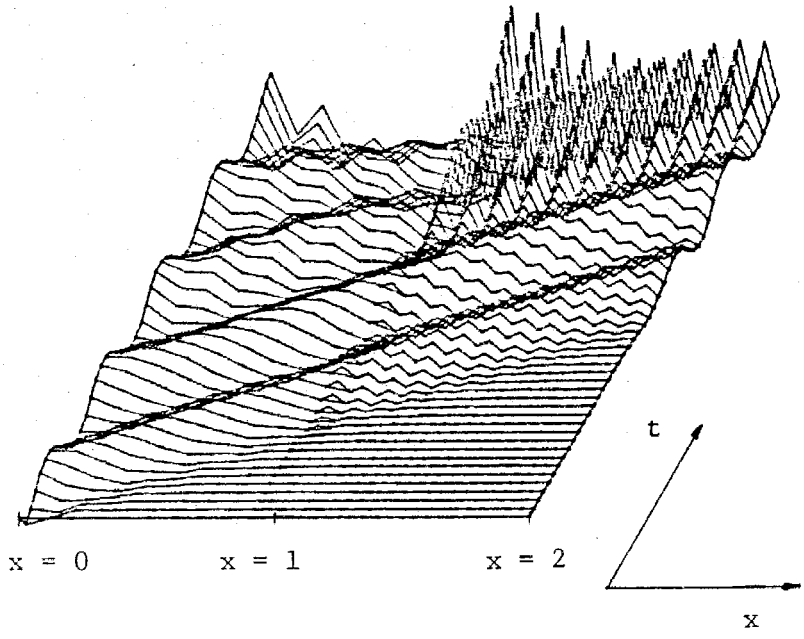


Figure 13a

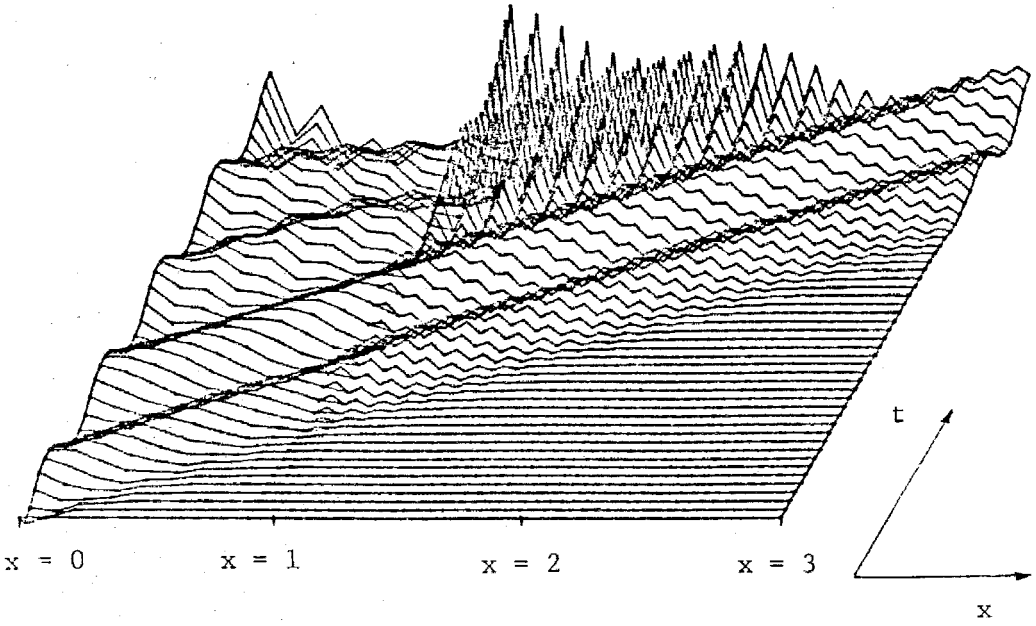
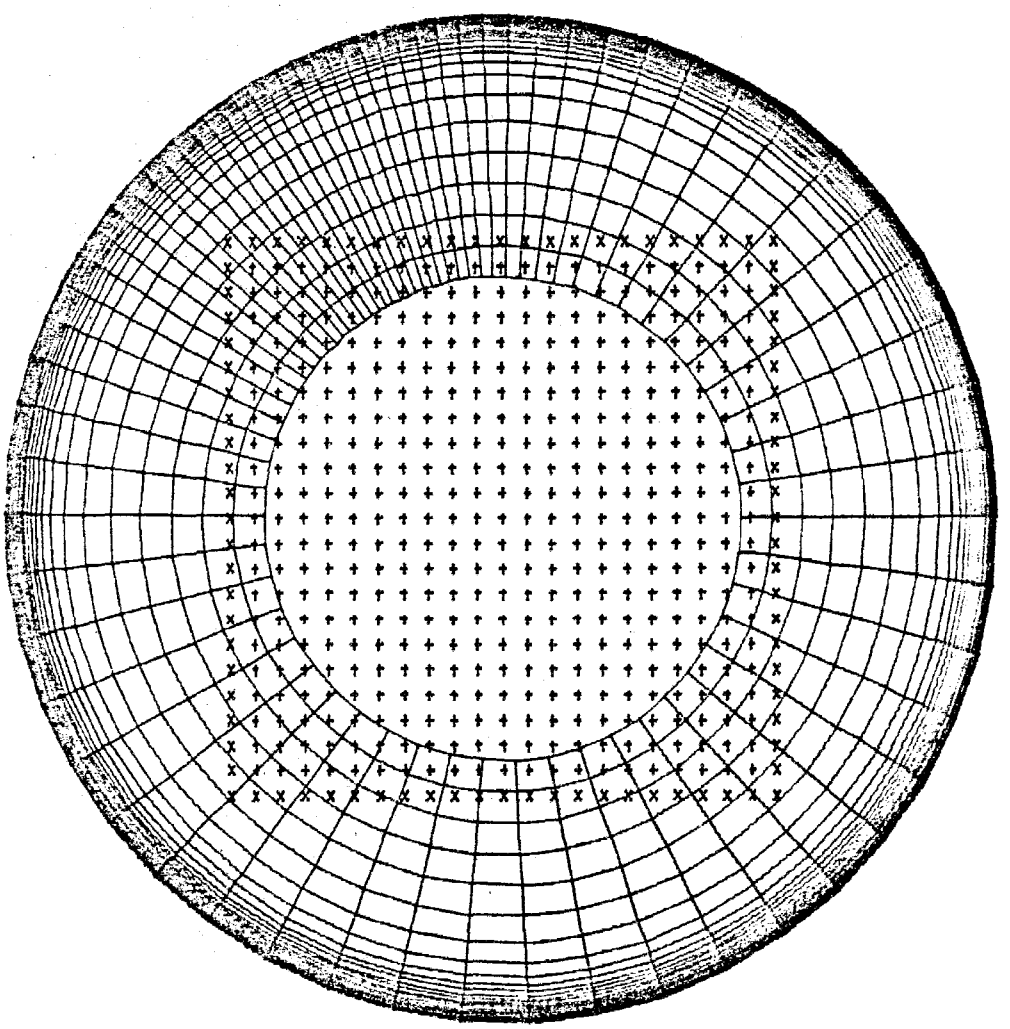


Figure 13b

Computational Mesh



epsr=0.0345 eps0=0.1111 eps1=.3333

NR= 20 NO= 68 Nx=23

Figure 14

T = 25.000 N =1000 STREAM FUNCTION
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

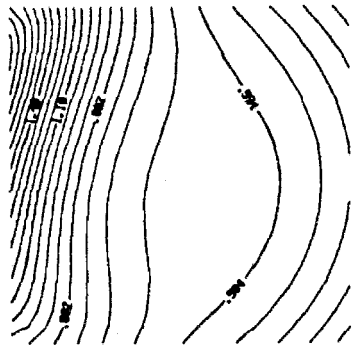


Figure 15a

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.3035 CONTOUR INTERVAL OF 0.74306E-01 PT(5,3)= 1.0110

T = 25.000 N =1000 STREAM FUNCTION
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

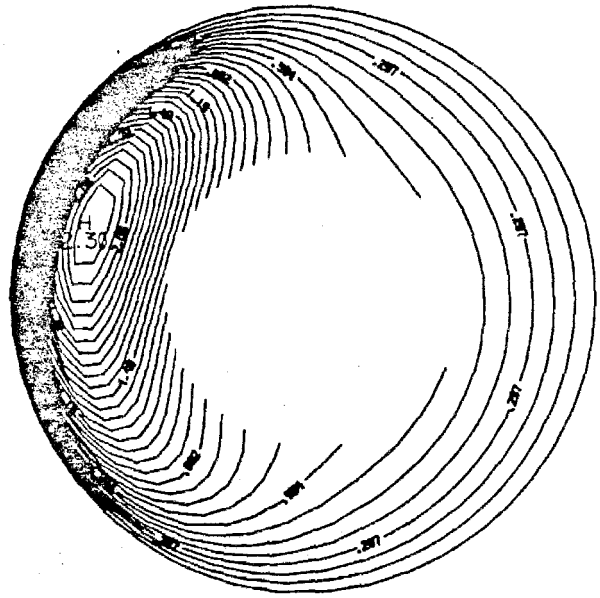


Figure 15b

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.3035 CONTOUR INTERVAL OF 0.74306E-01 PT(5,3)= 0.40070

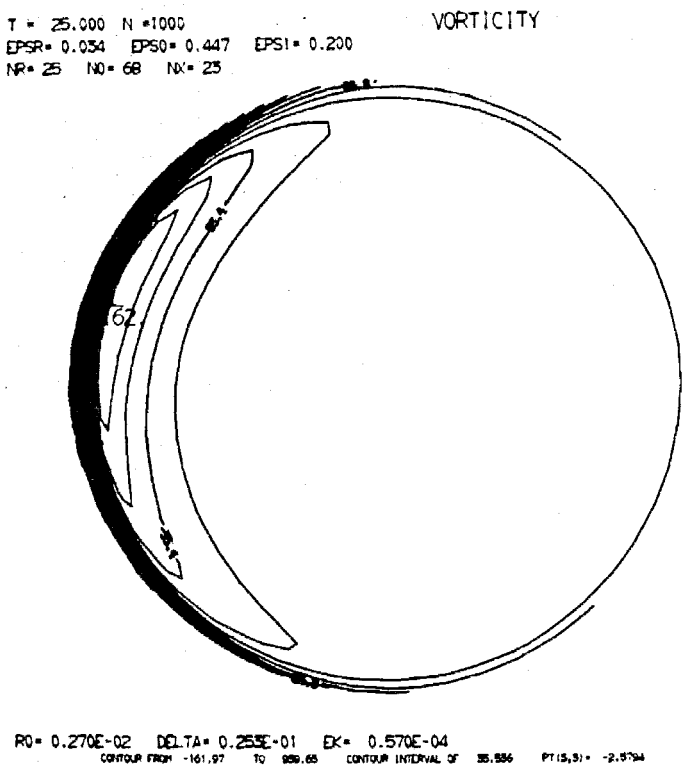

T = 25.000 N =1000
VORTICITY
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25
62
R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM -161.97 TO 959.65 CONTOUR INTERVAL OF 35.886 PT(5,5)= -2.5794

Figure 15c

T = 75.000 N =3000 STREAM FUNCTION
EPSR= 0.054 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

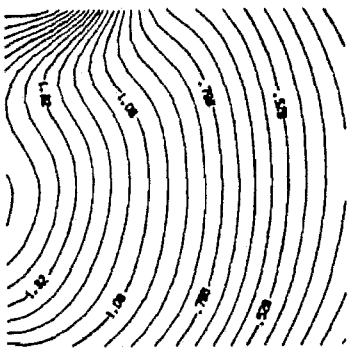


Figure 16a

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.0497 CONTOUR INTERVAL OF 0.66118E-01 PT(5,5)= 1.2542

T = 75.000 N =3000 STREAM FUNCTION
EPSR= 0.054 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

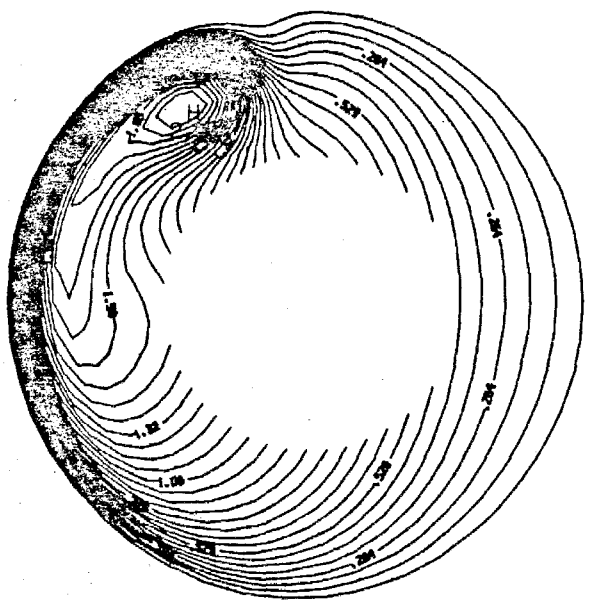


Figure 16b

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.0497 CONTOUR INTERVAL OF 0.66118E-01 PT(5,5)= 0.29055

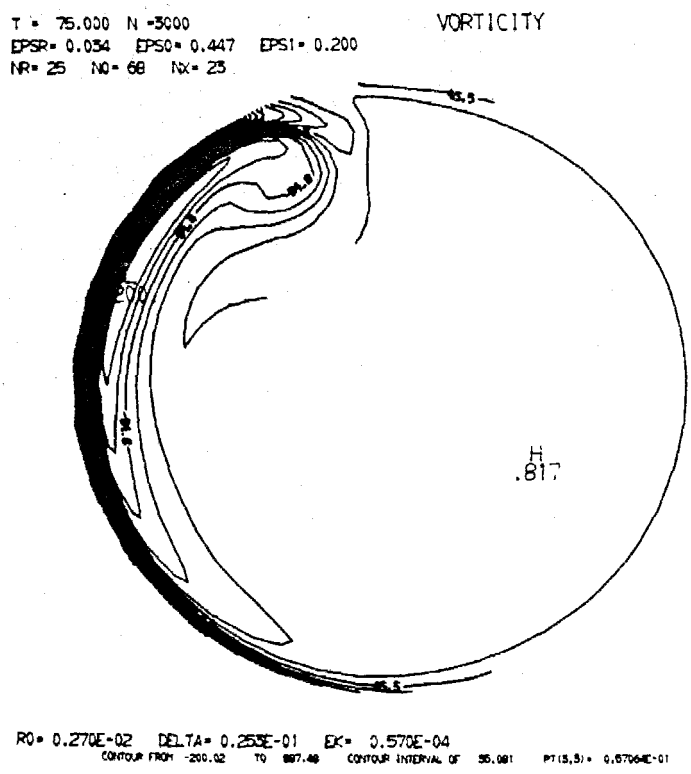

T = 75.000 N =3000
VORTICITY
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25
.817
R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM -200.02 TO 887.48 CONTOUR INTERVAL OF 55.081 PT(5,5)= 0.6706E-01

Figure 16c

T = 100.000 N =4000 STREAM FUNCTION
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

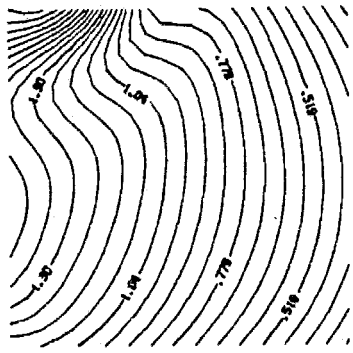


Figure 17a

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.0128 CONTOUR INTERVAL OF 0.64925E-01 PT(5,5)= 1.2165

T = 100.000 N =4000 STREAM FUNCTION
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 N0= 68 NX= 25

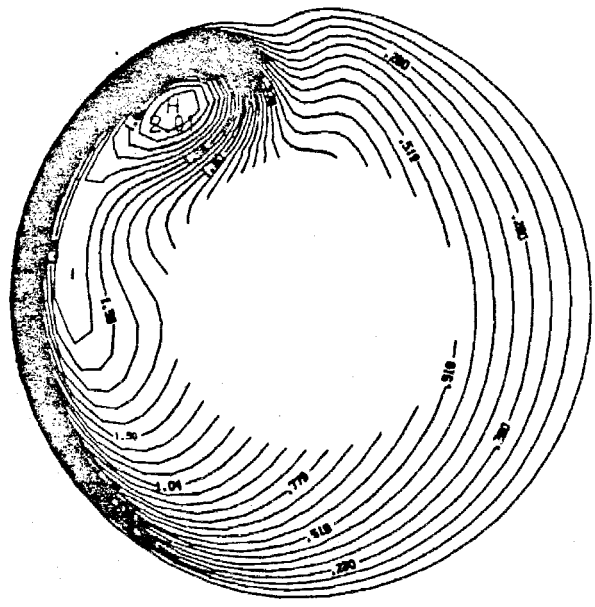


Figure 17b

R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM 0.00000 TO 2.0128 CONTOUR INTERVAL OF 0.64925E-01 PT(5,5)= 0.57435

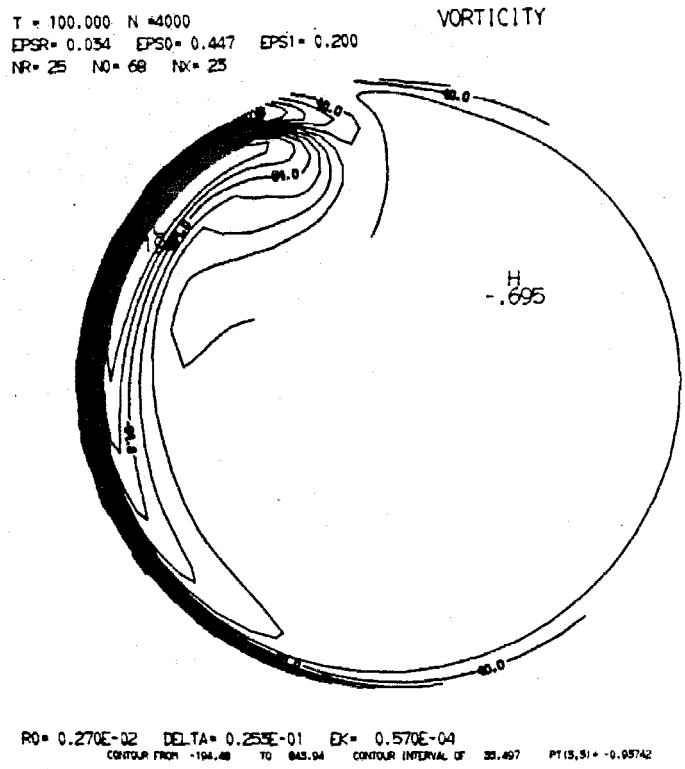

T = 100.000 N =4000
VORTICITY
EPSR= 0.034 EPS0= 0.447 EPS1= 0.200
NR= 25 NO= 68 NX= 25
H
-.695
R0= 0.270E-02 DELTA= 0.255E-01 EK= 0.570E-04
CONTOUR FROM -194.48 TO 843.94 CONTOUR INTERVAL OF 35.497 PT(5,5)= -0.05742

Figure 17c

STREAM FUNCTION

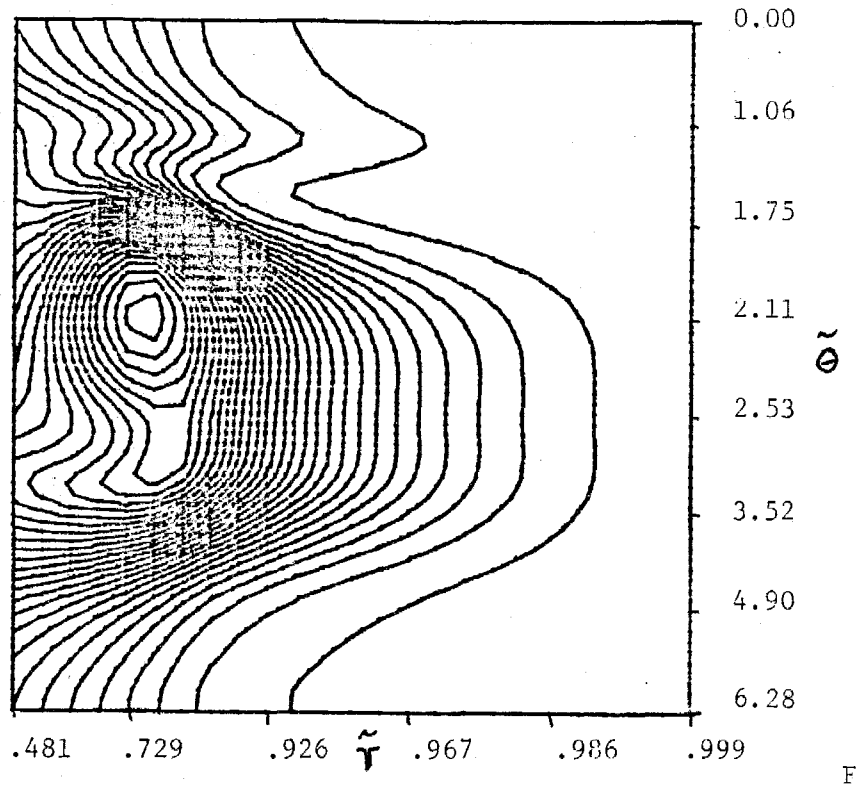


Figure 18a

VORTICITY

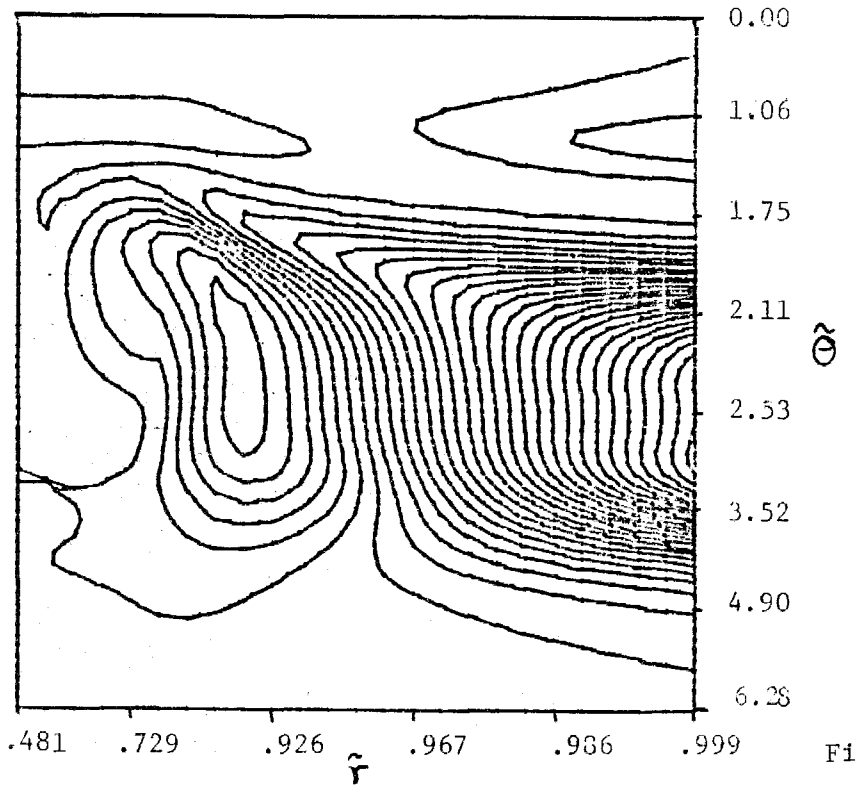


Figure 18b

L_2-Error for the Dirichlet Problem				
Uniform Meshes				
	Mesh I	*Mesh II*	*Mesh III*	*Mesh IV*
storage	*9742*	*47044*	*120820*	*129906*
N_e	*295*	*949*	*2007*	*1889*
ω=1	3.01E−03	6.41E−04	2.84E−04	5.47E−04
ω=2	5.08E−03	7.06E−04	2.54E−04	1.02E−03
ω=3	1.20E−02	1.59E−03	5.77E−04	1.12E−03
ω=4	4.49E−02	8.61E−03	3.60E−03	2.91E−03
ω=5	3.52E−02	5.74E−03	2.24E−03	3.36E−03
ω=6	3.19E−02	5.77E−03	2.37E−03	4.39E−03
ω=7	6.58E−02	7.49E−03	3.04E−03	5.57E−03
ω=8	9.64E−02	1.05E−03	4.12E−03	7.06E−03

Table 1

L_2-Error for the Dirichlet Problem				
Non-Uniform Meshes				
	Mesh I	*Mesh II*	*Mesh III*	*Mesh IV*
storage	*9092*	*43780*	*119592*	*116126*
N_e	*295*	*949*	*2007*	*1889*
α	$1.98E-05$	$3.13E-05$	$3.91E-05$	$3.31E-05$
ω=1	$5.14E-02$	$6.17E-03$	$4.33E-03$	$3.04E-03$
ω=2	$8.13E-02$	$9.83E-03$	$6.14E-03$	$5.05E-03$
ω=3	$1.16E-01$	$1.39E-02$	$6.84E-03$	$6.53E-03$
ω=4	$1.51E-01$	$2.25E-02$	$9.35E-03$	$1.04E-02$
ω=5	$1.53E-01$	$2.06E-02$	$8.76E-03$	$1.26E-02$
ω=6	$1.23E-01$	$2.21E-02$	$9.67E-03$	$1.62E-02$
ω=7	$1.52E-01$	$2.63E-02$	$1.11E-02$	$2.07E-02$
ω=8	$2.69E-01$	$3.24E-02$	$1.31E-02$	$2.56E-02$

Table 2

References

[1] Alexander,R., Manselli,P., and Miller,K., *Moving Finite Elements for the Stefan Problem in Two Dimensions*, presented at SIAM 1979 Fall Meeting, Denver, Colorado, November 12-14 (1979).

[2] Bank,R., *A Multi-Level Iterative Method for Finite Element Equations*, presented at the Adaptive Mesh Workshop, Center for Nonlinear Studies, Los Alamos Natl. Lab., August 5-7 (1981).

[3] Beardsley,R.C., *A Laboratory Model of the Wind-Driven Ocean Circulation*, J. Fluid Mech., 38 (1969), pp. 255-271.

[4] Beardsley,R.C., *A Numerical Model of the Wind-Driven Ocean Circulation in a Circular Basin*, Geoph. Fluid Dynamics, 4 (1973), pp. 211-241.

[5] Beardsley,R.C., *The 'Sliced-Cylinder' Laboratory Model of the Wind-Driven Ocean Circulation. Part 2. Oscillatory Forcing and Rossby Wave Resonance.* J. Fluid Mech., 69 (1975), pp. 41-64.

[6] Beardsley,R.C. and Robbins,K., *The 'Sliced-Cylinder' Laboratory Model of the Wind-Driven Ocean Circulation, Part 1. Steady Forcing and Topographic Rossby Wave Instability.* J. Fluid Mech., 69 (1975), pp. 27-40.

[7] Browning,G., Kreiss,H.-O. and Oliger,J., *Mesh Refinement*, Math. Comp., 27 (1973), pp. 29-39.

[8] Bryan,K., *A Numerical Investigation of a Nonlinear Model of Wind-Driven Ocean*, J. Atmos. Sci., 20 (1963), pp. 594-606.

[9] Crowley,W.P., *A Numerical Model for Viscous, Nondivergent, Barotropic, Wind-Driven, Ocean Circulation*, J. of Comput. Physics, 6 (1970), pp. 183-199.

[10] Davis,R.T., Proceedings, 4th AIAA Computational Fluid Dynamics Conference, Williamsburg, VA, July 24-26 (1979).

[11] Eiseman,P.R., *A Multi-Surface Method of Coordinate Generation*, J. of Comput. Physics, 33 (1979), pp. 118-150.

[12] Fornberg,B., *A Numerical Method for Conformal Mappings*, SIAM J. Sci. Stat. Comput., 1 (1980), pp. 386-400.

[13] Gelinas,R.J., Doss,S.K. and Miller,K., *The Moving Finite Element Method: Applications to General Partial Differential Equations with Large Gradients*, J. of Comput. Physics 40 (1981) pp. 202-249.

[14] Greenspan,M.P., *The Theory of Rotating Fluids*, Cambridge Univ. Press (1968).

[15] Gustafsson,B., Kreiss,H.-O. and Sundstrom,A., *Stability Theory of Difference Approximations for Mixed Initial Boundary Value Problems. II*, Math. Comp., 26 (1972), pp. 649-686.

[16] Israeli,M., *A Fast Numerical Method for Time Dependent Viscous Flows*, Studies in Applied Math., 49 (1970) pp. 327-349.

[17] Ives,D.C. and Livtermoza,J.F., *Analysis of Transonic cascade Flow Using Conformal Mapping and Relaxation Techniques*, AIAA Paper No. 76-370, Ninth Fluid and Plasma Dynamics Conference (1976).

[18] Kreiss,B., *Construction of Curvilinear Grids* Uppsala Univ., Dept. of Computer Sci., Report No. 89 (1981).

[19] Kreiss,H.-O., *Stability Theory for Difference Approximations of Mixed Initial Boundary Value Problems*, Math. Comp., 22 (1968), pp. 703-714.

[20] Kreiss,H.-O., *Stability Theory for Difference Approximations of Mixed Initial Boundary Value Problems. II*, Math. Comp., 26 (1972), pp. 649-685.

[21] Linden,J., *Mehrgitterverfhren fur die Poisson Gleichung in Kreis und Ringgebiet unter Verwendung lokaler Koordinaten* Diplomarbeit, Bonn, West Germany (1981).

[22] Pedlosky,J., *Geophysical Fluid Dynamics*, Springer Verlag (1979).

[23] Reinelt,D.A., Ph. D. Thesis, California Institute of Technology, Dept. of Applied Mathematics (1983).

[24] Sorenson,R.L., *A Computer Program to Generate Two-Dimensional Grids About Airfoils and Other Shapes by the Use of Poisson's Equation*, Nasa Technical Memorandum 81198.

[25] Starius,G., *Constructing Orthogonal Curvilinear Meshes by Solving Initial Value Problems*, Numer. Math., 28 (1977), pp. 25-48.

[26] Starius,G., *Composite Mesh Difference Methods for Elliptic Boundary Value Problems*, Numer. Math., 28 (1977), pp. 243-258.

[27] Starius,G., *On Composite Mesh Difference Methods for Hyperbolic Differential Equations*, Dept. of Comput. Sci., Uppsala Univ., Rep. No. 79 (1979).

[28] Steger,J.L. and Sorenson,R.L., *Automatic Mesh-Point Clustering Near a Boundary in Grid Generation with Elliptic Partial Differential Equations*, J. Comp. Phys., 33 (1979), pp. 405-410.

[29] Trefethen,L.N., *Group Velocity in Finite Difference Schemes*, SIAM Review, 24 (1982) pp. 113-136.

[30] Trefethen,L.N., *Wave Propagation and Stability for Finite Difference Schemes*, Report No. STAN-CS-82-905, Stanford University Dept. of Computer Science, (1982).

[31] Veronis,G., *Wind Driven Ocean Circulation- Part1. Linear Theory and Perturbation Analysis*. Deap-Sea Research, 13 (1966), pp. 17-29.

[32] Veronis,G., *Wind Driven Ocean Circulation- Part2. Numerical Solutions of the Nonlinear Problem*. Deap-Sea Research, 13 (1966), pp. 31-55.