NASA Contractor Report 158947

ON SEVERAL ASPECTS AND APPLICATIONS OF THE MULTIGRID METHOD FOR SOLVING PARTIAL DIFFERENTIAL EQUATIONS

NATHAN DINAR

COLLEGE OF WILLIAM AND MARY
WILLIAMSBURG, VA 23185

NASA CONTRACT NAS1-14972-3

SEPTEMBER 1978

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

TABLE OF CONTENTS

		Page
I.	Introduction	1
II.	Multigrid Methods for Systems of Equations	4
	2.1 Cauchy-Riemann Equations	5
	2.2 Stokes Equations	9
	2.3 Navier-Stokes	12
III.	Control and Prediction Techniques in Multigrid Methods	16
IV.	Multigrid Methods for Time Dependent Parabolic Equations	21
V.	Improvements and Changes in Existing Multigrid Algorithms	27
	5.1 The Poisson Equation	27
	5.2 Distribution Relaxation	33
VI.	References	39
VII.	Appendix	40

ON SEVERAL ASPECTS AND APPLICATIONS OF THE MULTIGRID METHOD FOR SOLVING PARTIAL DIFFERENTIAL EQUATIONS

By Nathan Dinar*

College of William & Mary
Williamsburg, Virginia 23185

Research Sponsored by NASA Langley Research Center
Contract NAS1-14972-3

SUMMARY

Several aspects of multigrid methods are briefly described in this report. The main subjects include the development of very efficient multigrid algorithms for systems of elliptic equations (Cauchy-Riemann, Stokes, Navier-Stokes), as well as the development of control and prediction tools (based on local mode Fourier analysis), used to analyze, check and improve these algorithms.

Preliminary research on multigrid algorithms for time dependent parabolic equations is also described. This report deals also with improvements in existing multigrid processes and algorithms for elliptic equations.

Some partial and typical results are given. More complete and detailed information will be presented in the author's Ph.D. Thesis, to appear at the Weizmann Institute of Science, Rehovot, Israel.

I. INTRODUCTION

This report deals with several aspects concerning multigrid methods for fast solution of partial differential equations. It covers the research on this subject for the period August 1977 - August 1978, when the author was spending his sabbatical at the NASA, Langley Research Center. This research is part of a Ph.D.

---

*Research Assistant, Department of Mathematics

Thesis to appear shortly at the Weizmann Institute of Science, Rehovot, Israel, including more detailed results and conclusions.

The research on multigrid methods began in the early 1970's by the initiative of Professor A. Brandt. He is today supervising several projects on these subjects, including the present research.

The multigrid method can be applied to a wide range of problems and, therefore, it interests many people. It is known to be one of the most powerful and advanced methods used today.

The multigrid method uses the fact that the numerical discrete equations we usually want to solve are not independent. They are derived from a continuous problem whose solution we want to approximate. In the process of the solution of the discrete equations it is convenient to keep in mind the differential origin of the problem. The use of discrete operators on several levels of meshes, interacting strongly in the process of the solution, allows us to solve the problems on the finest grid very efficiently, with a minimum number of arithmetical operations ($O(N)$), where N is the number of equations in the the finest grid.

The multigrid method consists generally of several processes, performed in a given order, and defining an iterative cycle. These processes include generally:

<u>Relaxation</u>: (Usually of Gauss-Seidel type) Used only to smooth the errors, i.e. to reduce these high frequencies in the error that are not described in coarser grids.

Transfer of Residuals to a Coarse Grid: This allows us to define a problem on a coarse grid, that is similar to the original one. The solution on the coarse grid is used to improve the approximate solution given on the finer grid.

Interpolation : Used to define a new approximation on a finer grid, given an approximation solution on a coarse grid.

Moreover, we can intermix the principles of the multigrid method with the grid adaptivity principles, which mean adaption of the order of discrete approximations and mesh size, using total smoothness of the solution. This keeps N as small as possible.

These basic ideas and others, as well as treatment of theoretical and practical aspects have been extensively covered in the papers of Brandt[1,2,3,].

This report deals with a wide spectrum of problems involved in the development of multigrid methods and multigrid software and the principal parts are the following:

a) Development of multigrid methods and software for elliptic systems of equations. (Including Cauchy-Riemann equation, Stokes equation, and Navier-Stokes equation.

b) Development of control and prediction tools for various steps and processes involved in the multigrid method. (Based on local mode Fourier analysis).

c) Development of multigrid methods for time dependent parabolic equations. (Heat equation on rectangular domains).

d) Improvement and changes in existing multigrid processes and algorithms (mainly on Poisson equation) and treatment of new ideas in relaxation methods.

II. MULTIGRID METHODS FOR SYSTEMS OF EQUATIONS

The basic ideas of the multigrid method are not restricted, of course, to a unique equation, and from a theoretical point of view no special problem could be expected in implementing multigrid ideas to a system of equations. However, special and detailed algorithms for this problem did not exist, and it was extremely important to get sharp and practical proofs of the efficiency of multigrid methods for systems of equations. One of the questions we did not know the answer in advance, was for instance, what is the appropriate method of relaxation for a system?

In order to answer this and many other important questions, we developed multigrid algorithms for three typical problems, starting with a simple one, and each subsequent problem involving more and new complications, relative to the former. Moreover, since it is usually wise and necessary, in this kind of research, to isolate different questions which may arise, we developed the algorithm for simple geometrical domains (rectangles). The past experience showed that more complicated geometries did not affect the efficiency of multigrid methods (see for instance Shiftan[4]). For complicated geometries there are, of course, more programing problems.

2.1 Cauchy-Riemann Equations

The equations are:

$$U_x - V_y = f_1(x,y) \tag{1}$$

$$U_x + V_y = f_3(x,y) \tag{2}$$

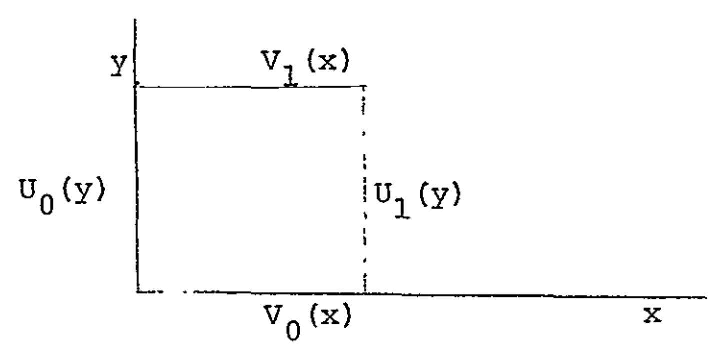
for $0 \leq x \leq 1$; $0 \leq y \leq 1$.

And the boundary conditions:

$$U(0,y) = U_0(y);\ V(x,0) = V_0(x) \tag{3}$$

$$U(1,y) = U_1(y);\ V(x,1) = V_1(x)$$

Figure 1.

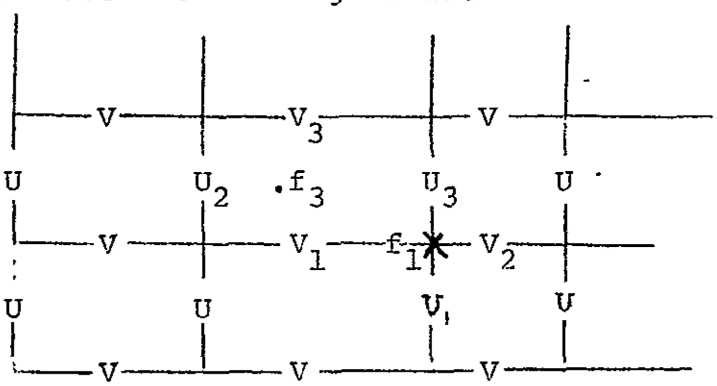


The data must fulfill an additional condition, due to the continuity equation (2), in order to assure a solution to the system. This condition is

$$\int_0^1 (U_1-U_0)\,dy + \int_0^1 (V_1-V_0)\,dx = \int_0^1\int_0^1 f_3(x,y)\,dxdy \tag{4}$$

The discretization of the problem was accomplished on a staggered grid. This method has several advantages in problems of this type. For instance, it allows us to define easily second order conservative finite differences. The grid is described in figure 2a.

Figure 2a.



The finite differences approximations for equation (1) are defined at grid intersections (X), for instance:

$$\frac{1}{h}\left[U_3-U_1-(V_2-V_1)\right] = f_1 \tag{5}$$

And the finite differences for equation (2) are defined in the middle of the cells (.), i.e.:

$$\frac{1}{h}\left[U_3-U_2+V_3-V_1\right] = f_3 \tag{6}$$

As in the continuous case, the data must fill an additional discrete condition equivalent to (4) in the finest grid, i.e.:

$$h\sum_j (U_{1j}-U_{0j}) + h\sum_i (V_{1i}-V_{0i}) = \sum_{i,j} f_{3;i,j} \tag{4a}$$

Several relaxation schemes appropriate to this system were considered and all these schemes have a good smoothing factor*, around $\bar{\eta}=.5$. The most convenient scheme was chosen because of its simplicity and its small number of arithmetical operations. This method belongs to a new approach of relaxations, developed by Professor A. Brandt, and is called "Distributed Relaxations".

This approach is characterized by the fact that when passing through a point (or cell) in the grid, for which the difference equations are defined, we change the value of <u>more</u> than one unknown, in order to make zero residual on this point.

In the Cauchy-Riemann equations separate relaxations are performed to each one of the equations, according to the schemes discribed in figures 3 and 4.

---

*Efficiency of relaxation in a multigrid process is measured by the "smoothing factor," which is defined as the asymptotic ratio between the fast components of the errors after a relaxation sweep, and the same errors before the relaxation. A more exact definition can be found in Brandt[2]. More considerations on this subject can be found in Chapter 3 of this report.

Figure 3: Relaxation of $U_x - V_y = f_1$

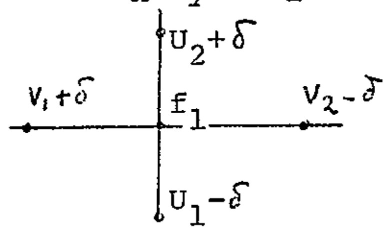
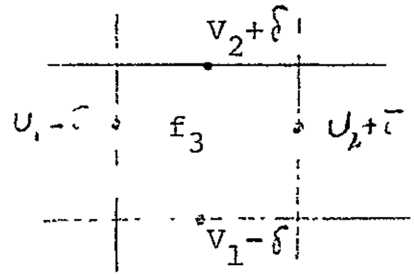


Figure 4: Relaxation of $U_x + V_y = f_3$



The values $U\pm\delta$ and $V\pm\delta$ in figures 3 and 4 represents the new values after the sweep of this point. δ is always chosen to make the residual zero at the point. The relaxation is a Gauss-Seidel type relaxation and it passes through all the points (or cells) of the grid in a usual natural order. The important property of this method relies on the fact that when relaxing each of the equations, the residual of the other equation remains unchanged.

Mode Fourier Analysis shows that the smoothing factor for this method is $\bar{\mu} = .5$, the same as in Poisson equation, which is, of course, equivalent to the Cauchy-Riemann system. The multigrid algorithm shows convergence factor $\overset{c}{\mu} \approx .56$* which is about the same value as in the Poisson case. The number of operations in the relaxation of the Cauchy-Riemann

---

*See Chapter 3 in this report.

system is, of course, higher than in Poisson equations (about two times more). On the other hand, we can say that the information we get from the solution of the Cauchy-Riemann system (functions U and V) is also two times the information we get from the solution of the Poisson equation.

The main reason for our treatment of Cauchy-Riemann system was for learning purposes. It is probably the simplest elliptic system possible and it allows us to treat complicated problems more easily. However, this system is interesting in itself, as we can see in the works of Lomax[5] and Ghil[6]. These papers describe fast algorithms for the solution of the Cauchy-Riemann system. We think that the algorithm just described in this report is preferable, because it does not depend, in principle, on the simplicity of the domain, and because it is at least as fast as Ghil algorithm, and even faster asymptotically (for finer and finer discretizations). The implementation of the algorithms for nonlinear equation is also much easier in the multigrid method[3].

An example of a computer output is given in the Appendix, Output No. 1:

2.2 Stokes Equations

The Stokes equation are given by

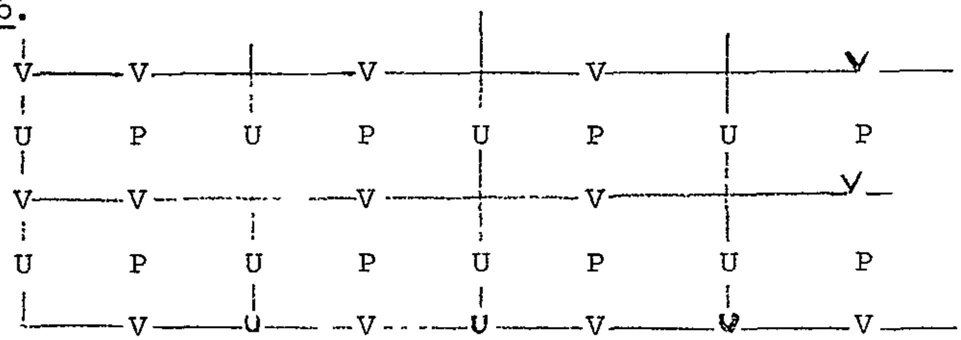
$$\Delta U - P_x = f_1 \tag{7}$$

$$\Delta V - P_y = f_2 \tag{8}$$

$$U_x + V_y = f_3 \tag{9}$$

The unknowns are U, V, and P. U and V are given on boundaries. Like in Cauchy-Riemann systems. The data must satisfy the condition (4). The discretization is also in the same staggered grids already described. P and f_3 are defined in the centers of the cells. f_1 and f_2 are defined in the same points as U and V, respectively (see figure 2b).

Figure 2b.



The difference equations for equation (7) are given by

$$\Delta_h U(x,y) - \frac{1}{h}\left[P(x+\frac{h}{2},y) - P(x-\frac{h}{2},y)\right] = f_1(x,y) \qquad (10)$$

for (x,y) where U is defined. Δ_h is the usual five points discrete Laplace operator: $\frac{1}{h^2}\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix}$

Similarly, for equation (8)

$$\Delta_h V(x,y) - \frac{1}{h}\left[P(x,y+\frac{h}{2}) - P(x,y-\frac{h}{2})\right] = f_2(x,y) \qquad (11)$$

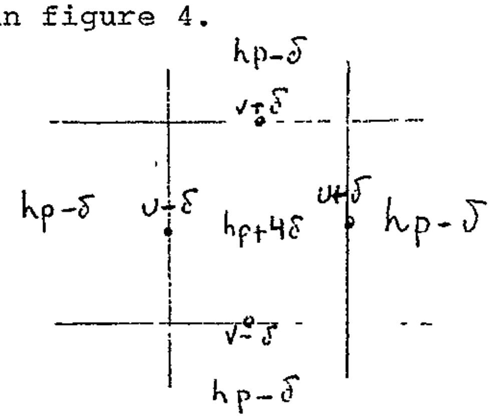
for (x,y) where V is defined.

For the continuity equation (9), the difference equations were already defined by (6).

Equations (10) and (11) hold for interior points. Near the boundary the difference equations are different, but they are simple and will not be described here.

The relaxations are based on the same principles already described. The relaxation of equations (10) and (11), the Momentum equations, are performed separately, by the usual Gauss-Seidel method corresponding to the Poisson operator. The relaxation scheme for the continuity equation (6), is described in figure 4.

Figure 4.



δ is chosen so that equation (6) is fully satisfied on the current cell, and the changes of the values of the P are chosen in a way that preserves the residuals of equations (10) and (11) when relaxing equation (6).*

The theoretical smoothing factor in the relaxation method just described can be shown to be $\bar{\eta}$ =.5, like in Poisson and Cauchy-Riemann equations.

In the numerical experiment we get a multigrid convergence factor** $\mathring{\eta} \approx .65$. This is a very good value, and assures

---

*This is exact only on unbounded domains. On bounded domains this property cannot be strictly maintained near the boundary.

**This is the reduction factor of the residuals per unit work which is equivalent to one relaxation sweep on the finest grid.

a very fast method for solving the Stokes equations. However, it was a little worse than our expectations, based on the previous experience of Poisson and Cauchy-Riemann equations.

In order to check this point, and to reduce the possibility of errors in the computer program, we performed a complete multigrid Mode Fourier Analysis (with the aid of the techniques to be described later). The results of this analysis showed complete agreement between numerical and theoretical results, leading to the conclusion that the value of $\overset{0}{\mu} \approx .65$ is intrinsic to this system, which is characterized by strong interaction between the equations. The Mode Fourier Analysis helped to rule out possible programing errors or bad influences of the boundary on interior regions.

An example of a computer output is given in the Appendix, Output No. 2:

2.3 Navier-Stokes Equations

Two-dimensional Navier-Stokes equations are given by

$$\left[\Delta - R\left(U\frac{\partial}{\partial x} + V\frac{\partial}{\partial y}\right)\right]U - P_x = f_1(x,y) \tag{12}$$

$$\left[\Delta - R\left(U\frac{\partial}{\partial x} + V\frac{\partial}{\partial y}\right)\right]V - P_y = f_2(x,y) \tag{13}$$

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = f_3(x,y) \tag{14}$$

and the same boundary condition as in Stokes equation. The parameter R is called the Reynolds number (R=0 corresponds to the Stokes equation).

These equations are more interesting, from the practical and physical point of view, than the former equations. The numerical treatment is more difficult and complicated, due to several reasons, like the nonlinearity and the fact that boundary-layers may appear for large Reynolds numbers.

These facts demand a careful choice of the difference equations, in order to keep the ellipticity of the difference operator. This can be accomplished in several ways, based usually on one-sided first differences, instead of central first differences, that may cause instability, unless we make a very drastic and unpractical restriction on the mesh size h (h must be of the order $h=0(\frac{1}{R})$). In this work, we define the following difference approximations to the first derivatives of U and V, for example for $\frac{\partial U}{\partial x}$ we define

$$\frac{\partial U}{\partial x} \approx a_x \frac{U(x+h) - U(x)}{h} + (1-a_x) \frac{U(x,y) - U(x-h,y)}{h} \tag{15}$$

for $0 \leq a_x \leq 1$ ($a_x = .5$ corresponds to central differences). a_x is defined by

$$a_x = \begin{cases} \frac{1}{2} & |\xi| \leq 1 \\ \frac{1}{2\xi} & \xi \geq 1 \\ 1 - \frac{1}{2\xi} & \xi \leq -1 \end{cases} \tag{16}$$

and $\xi = hRU(x,y)$

The approximation for $\frac{\partial U}{\partial x}$, $\frac{\partial V}{\partial x}$ and $\frac{\partial V}{\partial y}$ are defined in a similar way. It can be shown that with this approximations we get finite differences with the desired properties. The equations, defined on the same staggered grid already described, are given by

$$M_h U(x,y) - \frac{1}{h}\left[P(x+\tfrac{h}{2},y) - P(x-\tfrac{h}{2},y)\right] = f_1(x,y) \tag{18}$$

$$M_h V(x,y) - \frac{1}{h}\left[P(x,y+\tfrac{h}{2}) - P(x,y-\tfrac{h}{2})\right] = f_2(x,y) \tag{19}$$

$$\frac{1}{h}\left[U(x+\tfrac{h}{2},y) - U(x-\tfrac{h}{2},y) + V(x,y+\tfrac{h}{2}) - V(x,y-\tfrac{h}{2})\right] = f_3(x,y) \tag{20}$$

$$M_h = \frac{1}{h^2}\begin{bmatrix} & \varphi(-\eta) & \\ \varphi(\xi) & -S & \varphi(-\xi) \\ & \varphi(\eta) & \end{bmatrix} \tag{21}$$

$$\xi = hRU \; ; \; \eta = hRV \tag{22}$$

$$\varphi(\alpha) = \frac{1}{2}\left[1 + (\alpha-1)_+ + (\alpha+1)_+\right] \; ; \; g_+ = \begin{cases} g(x) & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{23}$$

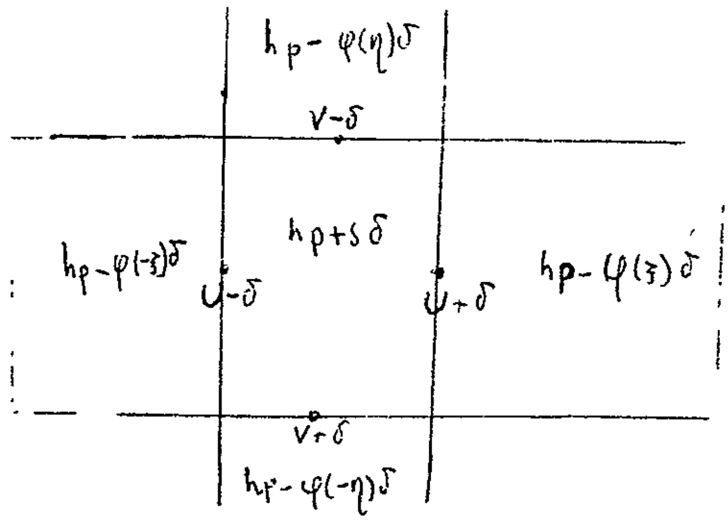
$$S = \varphi(\xi) + \varphi(-\xi) + \varphi(\eta) + \varphi(-\eta) \tag{24}$$

(These are the equations for interior points. The operator is a little different near boundaries).

The relaxation method for (18) and (19) are done as in the Stokes equation, by freezing U and V in the calculation of $\varphi(\xi)$ and $\varphi(\eta)$.

The relaxation scheme for equation (20) is described in figure 5.

Figure 5.



δ is chosen so that the residual of the continuity equation in the current cell is zero. The changes in P would keep the residuals on (18) and (19) unchanged if we froze the coefficients.

Mode Fourier Analysis shows that the smoothing factor (calculated for equations with frozen coefficients) will generally depend on the direction of the relaxation of the Momentum equations. For instance, if the solutions U and V satisfy $U \geq 0$; $V \geq 0$, and the direction of the relaxation is as usual, (say by columns and increasing y, and increasing x) then $\tilde{\eta} = .5$. If U or V are negative in some part of the domain, and R is big enough, the smoothing factor will be higher, and, therefore, the multigrid method less efficient. These difficulties may be easily overcome by making relaxations sweeps in different directions, improving in this way the smoothing factor, or better, by the technique of distributed relaxations already mentioned.

The first preliminary version of the algorithm does not include the features just pointed out, and, therefore, it works with the full multigrid efficiency only for cases where U and V do not change sign in the domain, or for general cases with R not too big (up to R=100).

These first experiments are very important in showing how the algorithms work in principle independent of secondary problems, like the direction of the relaxations, mainly technical.

The numerical results shows a good multigrid convergence factor, bounded for all R by $\hat{\mu} \approx .7-.75$. This provides a very fast algorithm for solving Navier-Stokes, and almost the same efficiency as in the linear Stokes case.

The full multigrid cycle (explained later) was also implemented. Preliminary experiments shows that only 8-9 works units* are needed to solve the equations to the level of truncations errors. (Execution time on CDC 6400 for a grid of 64x64, is about 15 seconds, and the program is not optimized, and includes all kind of calculations for debugging purposes). An example of a computer output is given in the Appendix, Output No. 3.

III. CONTROL AND PREDICTION TECHNIQUES IN MULTIGRID METHODS

One of the remarkable advantages of the multigrid method is the possibility of predicting in advance its efficiency, given the relaxation method, interpolation,

*Equivalent to relaxations sweeps on the finest grid.

etc. This can be done by means of local Fourier Mode Analysis in an infinite space, i.e. neglecting boundaries influence. This can be justified because of the fact that Fourier Analysis is very accurate in descibing the fast components behavior, while is less accurate for slower components. However, most of the calculating work in the multigrid processes is invested in the reduction of the fast components of the errors in the finest grid, and this is achieved by means of relaxation. More detailed justifications can be found in Brandt[3], who shows a very good agreement between theoretical results based on Fourier Mode Analysis and between numerical data obtained by multigrid algorithms.

We can distinguish two levels of analysis that are appropriate for multigrid methods. In the simplest one we calculate the smoothing factor $\bar{\eta}$, which depends only on the relaxation process. The more complete and complicated analysis takes into account all the multigrid processes and estimates theoretically the value of $\mathring{\eta}$ (definitions of $\bar{\eta}$ and $\mathring{\eta}$ are given in the footnotes in Chapter 2). In many cases, it is enough to perform the simpler analysis. In these cases, the formula $\mathring{\eta}_{approx.} \approx \bar{\eta}^{1-\rho^d}$, where ρ is the ratio of the number of points between coarse approximate and fine grid, (usually $\rho = \frac{1}{2}$), and d is the dimension of the problem provides a fair approximation to $\mathring{\eta}$.

The calculation of the smoothing rate involves the search for extremum of functions in bounded domains, and the calculation of eigenvalues (defined usually by a generalized eigenvalue problem $Ax=\lambda Bx$) of complex matrices of size $q \times q$, where q is the number of equations in the system. The estimation of the convergence factor involves the same kinds of calculations, but the size of the matrices are much bigger ($2^d q \times 2^d q$ where d is the dimension of the problem).*

It is clear that the kind of calculations just described, cannot be performed in a closed form, with the exception of extremely simple cases, like the smoothing rate for Poisson equations and one dimension simple problems.

Because of this and other important reasons, an algorithm that performs these calculations was constructed. A general FORTRAN subroutine was written for this purpose, and the main inputs are the following:

d - The dimension of the problem.

q - The number of differential equations = the number of unknowns.

r - The number of relaxations in one multigrid iterative cycle.

---

*In the relaxation process, the Fourier components are independent. But when we take into account the transfer from grid h to grid 2h we get interdependence between groups of 2^d components.

A,B - Complex matrices that are derived from the particular difference equations and methods of relaxations. The size of these matrices is $q \times q$ and they depend on the Fourier component defined by $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$

A_θ - A function of the Fourier component θ, describing the initial distribution of amplitudes in the errors. (For a random initial distribution of errors, $A_\theta = 1$ for all θ).

L - Number of points defining a mesh in the θ domain.

ρ - The ratio between the number of points on grid and the number of points on grid .

The output contains mainly:

a) The smoothing factor $\bar{\mu}$ defined by

$$\bar{\mu} = \max_{\rho\pi \leq |\theta| \leq \pi} \mu(\theta) \tag{25}$$

where

$$\mu(\theta) = \max \left\{ \lambda : \det\left[A(\theta) - \lambda B(\theta)\right] = 0 \right\} \tag{26}$$

and $|\theta| = \max\limits_{1 \leq i \leq d} |\theta_i|$

b) The weighted smoothing factor μ_r given the number of relaxations r and the weight function A_θ.

c) The location (mode) where the smoothing factor $\bar{\mu}$ is found.

d) A distribution map of $\mu(\theta)$ (only for the two-dimensional cases, d=2).

The same subroutine, with some changes, can be used for the complete Fourier Multigrid Analysis. In this case, the user has considerably more work in defining the complex matrices A and B that include, in this case, the transfer of residuals from grid h to grid 2h, and the interpolation from grid 2h to grid h.

These programs are very important in the research of multigrid methods. It can be used for several purposes, for example:

1) Checking new relaxation methods or new multigrid processes, before the algorithm is translated into the computer.

2) Comparison of several algorithms for the purpose of optimization, i.e. looking for the faster and stable ones.

3) Debugging of multigrid computer programs.

We used the subroutine in several cases, including Cauchy-Riemann and Stokes equation. We also used it for the Poisson equation. In all these cases, we did the complete Fourier multigrid analysis, checking and comparing various methods of relaxation, residual weighting and interpolation. We got full agreement between theoretical and numerical results (in all cases considered), and this fact increases, of course, the reliability and prediction power of this theory.

An example of these facts are summarized on table 1, which include theoretical and numerical results for Stokes equation:

Table 1

Comparison Between Theoretical and Numerical Results for Stokes Equation

r (The number of relaxations on the finest grid in one iterative multigrid cycle)	ς Parameter related to the degree of accuracy in solving coarse grid correction equations	$\mathring{\mu}$ Multigrid Convergence Factor	
		Theoretical multigrid complete Fourier analysis	Numerical results
1	.4	.661	.638
1	.5	.648	.634
2	.3	.680	.695
2	.4	.710	.725
3	.1	.714	.722

The subroutine was also applied for checking a special approach to distributed relaxations, that may perhaps be applied to Navier-Stokes equations. This will be shortly described in Chapter 5. An example of a computer output is given in the Appendix, Output N^0 4.

IV. MULTIGRID METHODS FOR TIME DEPENDENT PARABOLIC EQUATIONS

A possible quite obvious way to use multigrid procedures for initial value problems is considered in the work of

Brandt[3]. By this procedure we use a multigrid algorithm for solving the implicit equations usually defined at each time step. If, for instance, we want to solve the Heat equation in two space dimensions then for each time step, an elliptic problem similar to the discrete Poisson equation is defined (and, in this case, the Gauss-Seidel relaxation has even better smoothing properties than in the Poisson case). The typical amount of work needed in advancing each time step by multigrid procedures will be, accordingly, equivalent to 5-6 relaxations (see [3]). If a solution for M time steps is required, then the total amount of work will be about 6M relaxations.

The question which naturally arise is whether we can use multigrid principles and ideas to get more efficient methods for these problems.

The answer appears to be positive, and Professor A. Brandt pointed on some approaches which eventually can develop in efficient multigrid algorithms for these problems.

A basic idea is that marching in time can be done for most of the time steps on coarser grids.

The appropriate equations on a coarse grid are carefully corrected in a way that assures the correct representation of the information from the fine grid, so that even when marching on coarser grid, the accuracy of the finer grid is kept. To update the information from the finer grid we need some sporadic and infrequent time steps on the finest grid, that constitute, of course, most of the computational work.

The base of this approach relies on the fact that fast Fourier components of the solution (represented on finer grids) converge to steady-state after a very short time. Changes in the solution after this are due mainly to slower components, that also change slower in time. This allows us to march on coarser grids with large time steps, after the influence of fast components have disappeared.

In order to check these and other ideas, we developed two different algorithms for the Heat equation, in a rectangular domain with Dirichlet Boundary Conditions. The first algorithm uses the Crank-Nicholson implicit scheme, and the second one uses the simplest explicit scheme. It must be pointed out, that the stability restriction in the explicit scheme* that makes this method very unpractical, does not appear in this case, because most of the time we march on very coarse grids, where large time steps are allowed.

a. <u>The Implicit Scheme</u>

The equations to be solved on the finest grid (with mesh size h^0 and time step k^0) are:

$$\frac{1}{k^0}\left[u^0(x,y,t+k^0) - u(x,y\ t)\right] - \frac{1}{2}\Delta_{h^0} u(x,y,t+k^0) - \frac{1}{2}\Delta_{h^0}(x,y,t) = f(x,y) \tag{27}$$

and the unknowns are $u^0(x,y,t+k^0)$ for each (x,y) defined on the grid. The index 0 in u^0,h^0,k^0 represents marching on

---

* $\frac{k}{h^2} \le \frac{1}{4}$

the finest grid. Similarly u^m, h^m, k^m will represent marching on a grid which is m levels coarser. The equations on coarser grids are given by

$$\frac{1}{k^m}\left[u^m(x,y,t+k^m)-u(x,y,t)\right]-\frac{1}{2}\Delta_{h^m}u^m(x,y,t+k^m)-\frac{1}{2}\Delta_{h^m}u^m(x,y,t)=f(x,y)+\tau_{h^m}^{h^0}(x,y,s) \quad (28)$$

where usually $h^m = 2^m h^0$ and $\tau_{h^m}^{h^0}$ represents the appropriate correction. τ has a very important significance, and it represents the spacial truncation error of grid m, relative to grid 0. $\tau_{h^m}^{h^0}$ is given by

$$\tau_{h^m}^{h^0} = \sum_{j=1}^{m}\left(\Delta_{h^j}-\Delta_{h^{j-1}}\right)u^{j-1}(x,y,s_j) \quad (29)$$

where $s=(s_1,s_2,\dots,s_m)$ satisfying $s_1 < s_2 < \dots < s_m < t$.

The transfer from a given coarse grid m to a finer grid m-1 is done by

$$u^{m-1} = u^{m-1}_{Last} + I^{m-1}_{m}(u^m - u^{m-1}_{Last}) \quad (29a)$$

where I^{m-1}_{m} means interpolation (cubic) from coarse to fine grid and "Last" represent the last marching in time on the m-1 grid.

b. The Explicit Scheme

The equations on the finer grid are:

$$\frac{1}{k^0}\left[u^0(x,y,t+k^0)-u(x,y,t)\right]-\Delta_{h^0}u^0(x,y,t)=f(x,y) \quad (30)$$

On coarser grids the following equations are defined:

$$\frac{1}{k^m}\left[u^m(x,y,t+k^m)-u^m(x,y,t)\right] - \Delta_{h^m} u^m(x,y,t) = f(x,y) + \tau_{h^m}^{h^0}(x,y,s). \quad (31)$$

In this algorithm we keep a constant ratio λ over all grids given by $\lambda = k^m/(h^m)^2$. The value of $\lambda = .2$ was chosen as very appropriate from the point of view of the fast Fourier components. τ represents here the complete relative truncation error (in space and time). τ is defined as follows:

For m=1

$$\tau_{h^1}^{h^0}(x,y,s_1) = \frac{1}{4k^0}\left[u^0(x,y,s_1+4k^0) - u^0(x,y,s_1)\right] - \Delta_{2h^0} u^0(x,y,s_1) - f(x,y) \quad (32)$$

and $s_1 < t$. Then for general m

$$\tau_{h^m}^{h^0} = \sum_{j=1}^{m} \tau_{h^j}^{h^{j-1}}(x,y,s_j) \quad (32a)$$

and s is defined as in (29). The transfer from coarse to finer grids is performed like in (29a).

It must be pointed out that this research, which deals with new approaches in implementing multigrid ideas, has a very preliminary character and the principal aim was to check, potentially, the various promising possibilities.

In this context, some numerical experiments were performed, in which we tried some criteria in order to define the times where we switch from coarse to finer grids and vice-versa. In addition, we compared the accuracy we got in these algorithms with the same schemes when we marched in time on the finest grid only. The numerical results we got from these experiments, in both algorithms, are good and encouraging. However, the explicit scheme appears to be preferable in some aspects. The numerical results in these experiments, and some theoretical considerations arising from the interpretation of these results, lead to the conclusion that applying these multigrid ideas with full efficiency, (i.e. solving the problem in an amount of work comparable with the work invested in solving elliptic problems), means using adaptive techniques where we can change and adapt the order of the discrete approximation.

But even without adaptive techniques we can use the present algorithms to solve the Heat equation very efficiently. The amount of work needed will depend on the smoothness of the solution. For instance, for a problem with smooth initial conditions, we can save 98% of the work, in relation to the same numerical scheme defined on the finest grid only (of size 64x64).

V. IMPROVEMENTS AND CHANGES IN EXISTING MULTIGRID ALGORITHMS

In this chapter, we deal with existing multigrid procedures, and we check various approaches in order to improve their efficiency. The first part deals with the practical solution of Poisson's equation in a small number of numerical operations. Although in this part we concentrate on Poisson's equation, for reason of convenience when checking and comparing numerical results, it is quite clear that the techniques developed here can be applied to other elliptic problems as well.

The second part of this chapter deals with a special approach to distributed relaxations for general five points difference operators of the form $\begin{bmatrix} & d & \\ a & -s & c \\ & b & \end{bmatrix}$; $s = a+b+c+d$.

5.1 The Poisson Equation

The first numerical multigird algorithm and program was written for the Poisson equation. For this equation there exists today more information related to multigrid methods than for any other problem.

In Brandt[1], the algorithm called there Cycle "C" for solving the Poisson equation is described with detail. In Cycle "C" the multigrid iterative cycle starts at the finest grid, where an initial approximation to the solution is defined. This approximation is then improved by solving correction equations on coarser grids. The switching from fine to coarse grid and vice versa is controlled by some internal criteria.

Cycle "C" is very useful in learning and understanding multigrid performance and several theoretical aspects, like the asymptotic convergence factor, etc. It is not generally the most efficient algorithm for solving real problems, for which we do not know in advance a good approximation on the finest grid. In addition, it is usually difficult to know in advance how many iterative cycles are needed to get the desired accuracy. One can perform several iterative cycles and then get very close to the solution of the difference equations but this will generally be wasteful, because in a real problem, we do not need more accuracy than the accuracy defined by the differential truncation error.

Because of these and other considerations a fixed algorithm for Poisson equation is described in Brandt[3]. In this algorithm we start with an approximation on the coarsest grid, and after we perform one iterative cycle on each level, the solution on each grid is used as a first approximation on the next finer grid, by means of a cubic interpolation. This algorithm is based on the knowledge and past experience in solving Poisson equation by multigrid techniques, and provides a solution to the problem up to an accuracy comparable with the truncation errors accuracy, and this solution is found in a minimal number of arithmetic operations (we do not even need to calculate the residual norms, used generally for internal criteria).

In the present work a similar but more general algorithm is constructed. This algorithm can serve for learning purposes as well as for solving practical problems. It can also be easily applied to other elliptic operators. The algorithm was implemented in the FAS (Full Approximation Storage) mode. (For a detailed description of FAS see Brandt[2]). In this mode we can look at the coarse grid correction equations in a somehow different, but equivalent way, as follows:

$$L_{2h} U^{2h} = f_h + \tau_h^{2h}$$

$$\tau_h^{2h} = (L_{2h} - L_h) U^h \tag{33}$$

U^{2h} is a new and better approximation than U_h to the exact solution u^h of the difference equation

$$L_h u^h = f_h \tag{34}$$

τ_h^{2h} given by (33) describes the truncation error of the grid 2h, relative to grid h. It is well known that the exact solution u of the differential equation Lu=f satisfy the following

$$L_{2h}u = f + \tau_{2h} \tag{35}$$

and τ_{2h} is the differential truncation error of the 2h grid. The comparison between (35) and (33), that arises naturally during the FAS Multigrid Cycle without additonal investment of computational work, shows the remarkable differential aspect of the multigrid method. This is in sharp contrast with other methods, where the algebraic aspect is usually the most important and central one. These properities open new and interesting possibilities in implementing the FAS algorihm: τ_h^{2h} in (33) satisfies the following

$$\tau_h^{2h} \approx \tau_{2h} - \tau_h \tag{36}$$

and the truncation errors can be represented by $\tau_h = \lambda h^2 + O(h^4)$ (provided additional smoothing conditions). Then we have

$$\tau_{2h} = \frac{4}{3}\, \tau_h^{2h} + O(h^4) \quad . \tag{37}$$

Then we can change the FAS algorithm by defining the following correction equation on a coarse grid (instead of 33).

$$L_{2h} U^{2h} = f + \frac{4}{3}\, \tau_h^{2h} \tag{38}$$

A series of numerical experiments were performed to check these interesting points, including printouts of $\|\tau\|_{L_2}$ and $\|u - v^h\|_{L_\infty}$ (we chose for this purpose problems where the exact differential solution u is known). As a result of these experiments, we can point out the following conclusions:

a) The assumption that after only one multigrid cycle (5-6 work units) we get the needed accuracy in the solution (truncation error accuracy) was confirmed. In addition, truncation error behaves clearly like $0(h^2)$.

b) We can get the same desired accuracy without even performing a complete multigrid cycle, i.e. we can stop the process before we interpolate back to finer grids. The solution is then defined on coarse grid points only, but the accuracy is the <u>finest grid accuracy</u>. This feature can be very helpful in the development of multigrid methods that use small amount of storage, much smaller than the number of unknowns in the finest grid.

c) Using the τ-extrapolation (38 instead of 33) we can get a much better approximation to the solution of the differential equation on the finest grid, much better in fact than the approximation achieved by the exact solution of the difference equations in this grid, where the accuracy is bounded by truncation errors. In order to get this better approximation, we do not have to invest any additional work! Our numerical results do not show, however, that the approximation is $0(h^4)$, in spite of (37). This is because of the fact that at least 5-th order interpolation is needed for this purpose.

Nevertheless, the improvement we can get by this small change in the FAS algorithm is impressive. The improvement will generally depend on the smoothness of the solution. The more smoothness the better approximations.

In table 2 we can compare the results we get in the different variations we tried. The problem is $\Delta u = f$; the solution is known, given by $u = \sin(3x+3y)$; $(0 \le x \le 3;\ 0 \le y \le 2)$

Table 2

Size of the Finest Grid	$\|\tilde{v}^h - u\|_{L_\infty}$			
	Experiment 1	Experiment 2	Experiment 3	Experiment 4
16 x 24	1.4×10^{-2} (23)	1.8×10^{-2} (5.9)	5.3×10^{-3} (5.9)	2.3×10^{-3} (9.1)
32 x 48	3.4×10^{-3} (23)	4.2×10^{-3} (5.6)	4.7×10^{-4} (5.5)	2.3×10^{-4} (8.9)
64 x 96	8.6×10^{-4} (23)	1.0×10^{-3} (5.4)	5.2×10^{-5} (5.4)	1.8×10^{-5} (8.9)

Key to table 2:

Experiment 1 - $\tilde{U}^h$ is the exact solution of the difference equations (up to randoff errors).

Experiment 2 - $\tilde{U}^h$ is the approximation after only one cycle in all grids.

Experiment 3 - $\tilde{U}^h$ is the approximation after only one cycle in all grids, but using (38) instead of (33).

Experiment 4 - Like experiment 3, but the number of relaxation in each level was doubled.

The numbers in parenthesis represent the number of work units which are equivalent to the total number of relaxations on the finest grid (printed in each line of the table).

5.2 Distributed Relaxation

One of the important application of the method of distributed relaxations is in finding relaxation schemes with good smoothing properties. This is generally hard for very asymmetric operators. In these cases, the usual Gauss-Seidel relaxation may be very good or very bad, depending on the direction in which the relaxation is performed. The problem is even more complicated in case of non-linear difference operators, because in this case we may need one direction for some parts of the domain, and an opposite direction for other parts. In these cases we can perform, as we explained in a previous chapter, a series of alternating direction relaxation sweeps, but this cannot generally be an ideal solution. If we think, for example, of a three-dimensional problem, we need there at least $2^3=8$ relaxations on each level for a multigrid iterative cycle, in order to do relaxations in all possible directions. This may be too much, because we know that in most multigrid procedures, we use only 2-3 relaxations for achieving optimal multigrid efficiency. So, alternating direction relaxations may in some cases reduce considerably the efficiency and, in addition, it will cause more programming difficulties.

Because of this and other reasons it is convenient to find relaxations, with good smoothing properties, that are to some extent independent on the direction of the relaxation, so we can relax always in the same direction, no matter how the operator behaves.

As a first example, we consider the one-dimensional operator represented by $[1, -2-\eta, 1+\eta]$; $0 \le \eta < \infty$ which can be derived, for example, from the differential equation $y'' + R y' = 0$ (by taking one-sided first differences). The smoothing factor can be easily calculated in this case, and for extreme values of η it behaves in the following way:

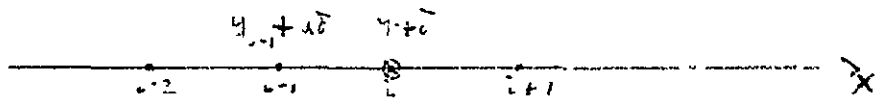
$$\lim_{\eta \to \infty} \bar{\eta} = \begin{cases} 1 & \text{Relaxation from left to right.} \\ \\ 0 & \text{Relaxation from right to left.} \end{cases} \tag{39}$$

for $\eta = 0$, $\bar{\eta} = .5$.

In the present work, we check a special principle for the construction of distributed relaxations. The relaxation is always done from left to right according to the scheme represented in figure 6.

Figure 6.

Relaxation $\longrightarrow$



y_{i-1} and y_i in figure 6 represent the values of the approximation before relaxing on point i, and the new values after the relaxation on this point are represented by $y_i+\delta$, $y_{i-1}+\alpha\delta$. δ is chosen so that the residual at point i is zero. The changes performed at points i and i-1 make the residuals at the points i-1 and i-2 nonzero, although they were previously zero when we relaxed these points. So we can choose the parameter α so that the deterioration of these residuals is minimal, say in the L_2 norm.

If we chose α in this way it can be shown that the relaxation scheme is always stable and satisfies $\bar{\mu} \le \frac{1}{\sqrt{2}} \simeq .7$ for all $\eta \ge 0$. For $\eta = 0$, $\bar{\mu} \simeq .24$. The optimum parameter α depends, of course, on η but satisfies $.4 \le \eta \le .5$ and, in fact, we can chose a constant suboptimal α , say $\alpha = .45$, without significant lose of efficiency.

It must be pointed out that this scheme is not necessarily the best one for all values of η . In fact, for large η , there exists better stable schemes (Brandt[7]).

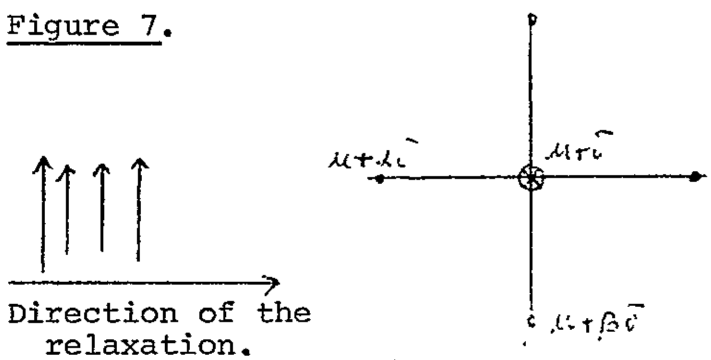
Using the same principles just stated, we calculated distributed relaxation schemes for general five points operators in the form

$$\begin{bmatrix} & d & \\ a & -s & c \\ & b & \end{bmatrix}$$

where a,b,c,d≥0 and a+b+c+d=s.

Operators of this type are typical, for instance, in Navier-Stokes problem. The relaxation scheme is described in figure 7.

<u>Figure 7</u>.



α and β are the solutions of an appropriate optmization problem. As in the one-dimensional case, $\mu+\tilde{\sigma}$, $\mu+\alpha\tilde{\sigma}$, $\mu+\beta\tilde{\sigma}$ represent the new values after the relaxation of the central point of figure 7.

The results (calculated by the algorithm described in Chapter 3) show that the smoothing factor is bounded far below 1, even in the cases of great asymmetry of the operator, except in the cases where the problem degenerate practically to a one-dimensional problem (like the case $a \simeq c \simeq 0$).
In the case of Poisson's equation (a=b=c=d=1) $\bar{\eta}$=.315, which

is significantly better than in the normal Gauss-Seidel relaxation ($\bar{\mu}$=.5). For this Poisson operation we get $\alpha=\beta=\frac{4}{19}$.* For other nonsymmetrical operators, the comparison between this distributed relaxation and the usual Gauss-Seidel is even more remarkable, as we can see on table 3.

Table 3

Operator				Smoothing Factor $\bar{\mu}$	
a	b	c	d	Gauss-Seidel Relaxations	Distributed Relaxations
.5	1.5	1.5	.5	.631	.313
10.5	.5	10.5	.5	.913	.449
.5	100.5	100.5	.5	.990	.481
.5	1000.5	1000.5	.5	.999	.484
.5	.5	1000.5	1000.5	1.000	.752

Like in the one-dimensional case, it may be possible to find more efficient scheme for the extremely asymmetric cases. But the present methods assures a good smoothing factor in all the cases, bounded far below 1, and consequently, an efficient multi-grid algorithm. Moreover, if the problem is nonlinear, we do not have to know in advance the behavior of the solution, in order to define an appropriate relaxation scheme. In addition,

---

*This distributed relaxation scheme for Poisson's equation was implemented in a multigrid computer program, and we get perfect agreement with the theoretical smoothing factor $\bar{\mu}$ = .315.

as in the one-dimensional case, it is not necessary to invest extra work in the exact determination of the parameters and They can be estimated in a suboptimal way, without significant loss of efficiency.

It must be pointed out that in the case of the Poisson equation the Gauss-Seidel relaxation can be still considered a little more efficient, if we take into account the number of arithmetic operations needed for each relaxation method. On the other hand, for all other operators, even if they are small perturbations of the Laplace operator, the proposed distributed relaxations are much more efficient than the Gauss-Seidel relaxation.

REFERENCES

1. A. Brandt, Multi-Level Adaptive Techniques (MLAT) for Fast Numerical Solution to Boundary-Value Problems. Proc. 3rd. International Conference on Numerical Methods in Fluid Mechanics (Paris 1972), Lecture Notes in Physics, Vol. 18, Springer-Verlag, Berlin and New York, 1973, pp. 82-89.

2. A. Brandt, Multi-Level Adaptive Solutions to Boundary-Value Problems. Research Report RC-6159, IBM Thomas J. Watson Research Center, N.Y., 1976.

3. A. Brandt, Multi-Level Adaptive Solutions to Boundary-Value Problems. Mathematics of Computation, 13, 1977.

4. Y. Shiftan, Multigrid Method for Solving Elliptic Difference Equations M. Sc. Thesis, Weizmann Institute of Science, Rehovot, Israel, 1972 (Hebrew).

5. H. Lomax, E. Martin, Fast Direct Numerical Solution of the Nonhomogeneous Cauchy-Riemann Equations, Journal of Computational Physics, 15, 1974.

6. M. Ghil and R. Balgovind, A Fast Cauchy-Riemann Solver with Nonlinear Applications, Research Report Courant Institute of Mathematical Sciences, New York University, 1978.

APPENDIX

OUTPUT NO. 1


```
CAUCHY RIEMAN EQ ,(2) UX+VY=C+S,(1) UY-VX=F,"INJECTION", S=0 , F=X**2+Y**2   ,   SEPARATE RELAXATIONS
COARSEST GRID IS   2  X-INTERVALS,AND    2  Y-INTERVALS, H0=  .500
 NUMBER OF GRIDS      6                                                       LEVEL 6= 64x64 GRID
METHOD PARAMETERS,  ETA=  .600   DELTA=  .200  ERW= 5.000
            TOL= .100E-04
```

LEVEL	RES,NORM(1)	RES,NORM(2)	WEIGHTED RES,NOR	WORK (ACC)
6	.473E+00	.311E-12	.788E-01	1.000
6	.449E+00	.417E-12	.748E-01	2.000
5	.413E+00	.460E-12	.688E-01	2.250
5	.377E+00	.517E-12	.628E-01	2.500
4	.320E+00	.493E-12	.534E-01	2.563
4	.266E+00	.496E-12	.443E-01	2.625
3	.193E+00	.432E-12	.322E-01	2.641
3	.134E+00	.409E-12	.224E-01	2.656
2	.806E-01	.331E-12	.134E-01	2.660
2	.464E-01	.297E-12	.773E-02	2.664
2	.264E-01	.270E-12	.441E-02	2.668
3	.280E-01	.952E-02	.126E-01	2.684
3	.183E-01	.299E-02	.554E-02	2.699
4	.367E-01	.158E-01	.192E-01	2.762
4	.206E-01	.592E-02	.836E-02	2.824
5	.456E-01	.150E-01	.201E-01	3.074
5	.264E-01	.727E-02	.105E-01	3.324
6	.488E-01	.124E-01	.184E-01	4.324

6	.316E-01	.784E-02	.118E-01	5.324
5	.242E-01	.747E-02	.103E-01	5.574
5	.187E-01	.729E-02	.919E-02	5.824
4	.148E-01	.661E-02	.797E-02	5.887
4	.116E-01	.608E-02	.699E-02	5.949
3	.852E-02	.459E-02	.525E-02	5.965
3	.608E-02	.343E-02	.387E-02	5.980
2	.343E-02	.159E-02	.190E-02	5.984
2	.140E-02	.556E-03	.696E-03	5.988
3	.100E-02	.109E-02	.107E-02	6.004
4	.195E-02	.145E-02	.153E-02	6.066
5	.395E-02	.160E-02	.199E-02	6.316
6	.775E-02	.187E-02	.285E-02	7.316
6	.527E-02	.140E-02	.204E-02	8.316
5	.391E-02	.125E-02	.170E-02	8.566
5	.276E-02	.119E-02	.145E-02	8.816
4	.204E-02	.101E-02	.118E-02	8.879
4	.144E-02	.932E-03	.102E-02	8.941
3	.107E-02	.715E-03	.773E-03	8.957
3	.764E-03	.622E-03	.646E-03	8.973
2	.502E-03	.415E-03	.429E-03	8.977
2	.278E-03	.307E-03	.302E-03	8.980
1	.949E-04	.105E-03	.103E-03	8.981
1	.142E-13	.116E-12	.990E-13	8.982
2	.774E-04	.113E-03	.107E-03	8.986
3	.120E-03	.197E-03	.184E-03	9.002
4	.297E-03	.247E-03	.255E-03	9.064
5	.686E-03	.289E-03	.355E-03	9.314
6	.139E-02	.320E-03	.498E-03	10.314
6	.920E-03	.207E-03	.325E-03	11.314
5	.696E-03	.174E-03	.261E-03	11.564
5	.502E-03	.158E-03	.216E-03	11.814
4	.377E-03	.128E-03	.170E-03	11.877
4	.267E-03	.108E-03	.135E-03	11.939
3	.190E-03	.745E-04	.937E-04	11.955
3	.124E-03	.510E-04	.633E-04	11.971
2	.768E-04	.279E-04	.361E-04	11.975
2	.423E-04	.156E-04	.200E-04	11.979
2	.257E-04	.105E-04	.130E-04	11.982
1	.751E-05	.358E-05	.424E-05	11.983
1	.142E-13	.108E-12	.927E-13	11.984
2	.651E-05	.444E-05	.479E-05	11.988
3	.240E-04	.181E-04	.191E-04	12.004
4	.613E-04	.331E-04	.378E-04	12.066
5	.129E-03	.443E-04	.585E-04	12.316
6	.249E-03	.509E-04	.839E-04	13.316
6	.170E-03	.320E-04	.550E-04	14.316
5	.130E-03	.270E-04	.442E-04	14.566
5	.940E-04	.249E-04	.364E-04	14.816

4	.698E-04	.214E-04	.295E-04	14.879
4	.479E-04	.198E-04	.244E-04	14.941
3	.329E-04	.156E-04	.185E-04	14.957
3	.203E-04	.136E-04	.147E-04	14.973
2	.126E-04	.876E-05	.940E-05	14.977
2	.676E-05	.598E-05	.611E-05	14.980
1	.296E-05	.173E-05	.194E-05	14.981
1	.142E-13	.118E-12	.100E-12	14.982
2	.183E-05	.230E-05	.222E-05	14.986
3	.414E-05	.424E-05	.422E-05	15.002
4	.112E-04	.601E-05	.687E-05	15.064
5	.243E-04	.742E-05	.102E-04	15.314
6	.463E-04	.832E-05	.147E-04	16.314
6	.315E-04	.556E-05	.988E-05	17.314

CYCLE NO. 2	CYCLE EFF. .574	ACC.EFF. .574
CYCLE NO. 3	CYCLE EFF. .556	ACC.EFF. .565
CYCLE NO. 4	CYCLE EFF. .542	ACC.EFF. .558
CYCLE NO. 5	CYCLE EFF. .553	ACC.EFF. .557

↑ CONVERGENCE FACTOR FOR EACH CYCLE

↑ MEAN CONVERGENCE FACTOR

OUTPUT NO. 2

EQUATIONS STOKES. (1) DEL(U)=PY+F (2) DEL(V)=PY+G (3) UX+VY=S
F=X**3+Y**3 G=X*Y S=0.

BOUNDARY CONDITIONS AND INITIAL FUNCTION VALUES U=X*(1.+Y**2) V=-Y*(1.+Y**2/3.) P=1.

METHOD FAS,SEPARATE RELAX.,LINEAR INT.,NEW SCHEME, RES. WEIGTHING EVERYWHERE

COARSEST GRID IS 2 X-INTERVALS,AND 2 Y-INTERVALS, H0= .500
NUMBER OF GRIDS 6
METHOD PARAMETERS, ETA= .100 DELTA= .200
TOL= .100E-06
EPS .350 .350 .300

LEVEL 6 = 64X64 GRID

LEVEL	ACC. WORK	RES. NORM (WEIGHTED)	RES. NORM EQ. 1	RES. NORM EQ. 2	RES. NORM EQ. 3
6	1.000	.151E+01	.130E+01	.281E+01	.292E+00
6	2.000	.152E+01	.127E+01	.273E+01	.386E+00
END CYCLE NO. 1					
5	2.250	.147E+01	.123E+01	.261E+01	.430E+00
5	2.500	.147E+01	.119E+01	.249E+01	.618E+00
4	2.563	.136E+01	.111E+01	.229E+01	.588E+00
4	2.625	.133E+01	.102E+01	.204E+01	.814E+00
3	2.641	.111E+01	.858E+00	.173E+01	.688E+00
3	2.656	.909E+00	.702E+00	.140E+01	.800E+00
2	2.660	.635E+00	.435E+00	.895E+00	.566E+00
2	2.660	.437E+00	.282E+00	.507E+00	.539E+00
1	2.665	.131E+00	.733E-01	.171E+00	.152E+00
1	2.666	.565E-01	.443E-01	.889E-01	.330E-01
2	2.670	.328E+00	.282E+00	.469E+00	.217E+00
2	2.670	.140E+00	.123E+00	.193E+00	.970E-01
3	2.689	.485E+00	.341E+00	.601E+00	.420E+00
3	2.705	.211E+00	.134E+00	.259E+00	.246E+00
4	2.768	.451E+00	.269E+00	.614E+00	.473E+00
4	2.830	.198E+00	.102E+00	.200E+00	.306E+00
5	3.080	.394E+00	.190E+00	.462E+00	.552E+00
5	3.330	.219E+00	.830E-01	.149E+00	.459E+00
6	4.330	.429E+00	.132E+00	.328E+00	.886E+00
6	5.330	.305E+00	.918E-01	.160E+00	.856E+00
END CYCLE NO. 2					

5	5.580	.192E+00	.772E-01	[illegible]	.406E+00
5	5.830	.187E+00	.708E-01	.130E+00	.399E+00
4	5.893	.990E-01	.491E-01	.930E-01	.167E+00
4	5.955	.802E-01	.412E-01	.776E-01	.141E+00
3	5.971	.387E-01	.252E-01	.475E-01	.441E-01
3	5.986	.251E-01	.188E-01	.305E-01	.262E-01
2	5.990	.100E-01	.983E-02	.124E-01	.478E-02
2	5.994	.537E-02	.631E-02	.406E-02	.579E-02
1	5.995	.123E-02	.211E-02	.312E-03	.129E-02
1	5.996	.498E-03	.740E-03	.274E-03	.478E-03
2	6.000	.308E-02	.382E-02	.308E-02	.222E-02
3	6.016	.108E-01	.870E-02	.158E-01	.723E-02
4	6.078	.351E-01	.270E-01	.476E-01	.299E-01
5	6.328	.567E-01	.385E-01	.672E-01	.656E-01
6	7.328	.903E-01	.491E-01	.959E-01	.152E+00
6	8.328	.497E-01	.204E-01	.290E-01	.109E+00
END CYCLE NO. 3					
5	8.578	.256E-01	.140E-01	.179E-01	.475E-01
5	8.829	.225E-01	.122E-01	.153E-01	.430E-01
4	8.891	.121E-01	.662E-02	.992E-02	.186E-01
4	8.953	.971E-02	.641E-02	.710E-02	.166E-01
3	8.969	.451E-02	.360E-02	.368E-02	.650E-02
3	8.984	.282E-02	.227E-02	.197E-02	.445E-02
2	8.988	.822E-03	.865E-03	.595E-03	.100E-02
2	8.992	.434E-03	.470E-03	.412E-03	.412E-03
3	9.008	.175E-02	.157E-02	.216E-02	.161E-02
4	9.070	.438E-02	.359E-02	.512E-02	.443E-02
5	9.320	.711E-02	.470E-02	.761E-02	.924E-02
6	10.320	.208E-01	.133E-01	.263E-01	.230E-01
6	11.320	.998E-02	.425E-02	.755E-02	.195E-01
END CYCLE NO. 4					
5	11.570	.437E-02	.208E-02	.300E-02	.869E-02
5	11.820	.375E-02	.161E-02	.225E-02	.790E-02
4	11.883	.193E-02	.100E-02	.151E-02	.387E-02
4	11.945	.157E-02	.773E-03	.110E-02	.298E-02
3	11.961	.669E-03	.391E-03	.593E-03	.100E-02
3	11.977	.416E-03	.249E-03	.300E-03	.598E-03
2	11.980	.125E-03	.731E-04	.140E-03	.123E-03
2	11.988	.826E-04	.658E-04	.112E-03	.683E-04
3	12.000	.261E-03	.223E-03	.253E-03	.314E-03
4	12.063	.700E-03	.550E-03	.770E-03	.807E-03
5	12.313	.192E-02	.102E-02	.140E-02	.191E-02
6	13.313	.340E-02	.192E-02	.380E-02	.166E-02
6	14.313	.204E-02	.868E-03	.148E-02	.806E-02
END CYCLE NO. 5					
5	14.563	.963E-03	.477E-03	.698E-03	.184E-02
5	14.813	.840E-03	.380E-03	.547E-03	.172E-02
4	14.875	.437E-03	.202E-03	.356E-03	.758E-03
4	14.938	.349E-03	.174E-03	.265E-03	.650E-03
3	14.953	.154E-03	.916E-04	.144E-03	.238E-03
3	14.969	.105E-03	.689E-04	.887E-04	.166E-03

ORIGINAL PAGE IS OF POOR QUALITY

2	14.973	.403E-04	.301E-04	.436E-04	.485E-04
2	14.977	.258E-04	.250E-04	.293E-04	.332E-04
1	14.979	.356E-05	.367E-05	.194E-05	.531E-05
2	14.981	.170E-04	.184E-04	.195E-04	.125E-04
3	14.997	.523E-04	.455E-04	.470E-04	.660E-04
4	15.060	.147E-03	.113E-03	.151E-03	.182E-03
5	15.310	.322E-03	.216E-03	.355E-03	.430E-03
2	16.310	.741E-03	.475E-03	.706E-03	.102E-02
6	17.310	.477E-03	.210E-03	.361E-03	.425E-03
END CYCLE NO. 6					
5	17.560	.218E-03	.101E-03	.166E-03	.414E-03
5	17.810	.189E-03	.800E-04	.128E-03	.349E-03
4	17.872	.993E-04	.516E-04	.830E-04	.171E-03
4	17.935	.795E-04	.379E-04	.598E-04	.151E-03
3	17.950	.332E-04	.181E-04	.298E-04	.550E-04
3	17.966	.203E-04	.119E-04	.154E-04	.358E-04
2	17.970	.569E-05	.372E-05	.476E-05	.407E-05
2	17.974	.387E-05	.415E-05	.301E-05	.454E-05
3	17.989	.124E-04	.100E-04	.120E-04	.155E-04
4	18.052	.351E-04	.269E-04	.356E-04	.439E-04
5	18.302	.690E-04	.469E-04	.661E-04	.995E-04
6	19.302	.180E-03	.991E-04	.172E-03	.283E-03
6	20.302	.120E-03	.496E-04	.886E-04	.238E-03
END CYCLE NO. 7					
5	20.552	.546E-04	.255E-04	.414E-04	.104E-03
5	20.802	.459E-04	.196E-04	.300E-04	.952E-04
4	20.864	.241E-04	.127E-04	.194E-04	.428E-04
4	20.927	.195E-04	.956E-05	.143E-04	.374E-04
3	20.942	.836E-05	.483E-05	.713E-05	.139E-04
3	20.958	.553E-05	.358E-05	.401E-05	.956E-05
2	20.962	.170E-05	.120E-05	.135E-05	.270E-05
2	20.966	.121E-05	.117E-05	.107E-05	.143E-05
1	20.967	.244E-06	.601E-07	.391E-06	.285E-06
1	20.968	.105E-06	.133E-06	.131E-06	.434E-07
2	20.972	.826E-06	.123E-05	.692E-06	.515E-06
3	20.987	.308E-05	.261E-05	.351E-05	.314E-05
4	21.050	.842E-05	.692E-05	.890E-05	.981E-05
5	21.300	.163E-04	.108E-04	.161E-04	.223E-04
6	22.300	.462E-04	.270E-04	.490E-04	.652E-04
6	23.300	.309E-04	.140E-04	.256E-04	.568E-04
END CYCLE NO. 8					
5	23.550	.132E-04	.644E-05	.104E-04	.246E-04
5	23.800	.109E-04	.488E-05	.727E-05	.221E-04
4	23.862	.556E-05	.302E-05	.459E-05	.975E-05
4	23.925	.437E-05	.223E-05	.318E-05	.827E-05
3	23.940	.176E-05	.104E-05	.152E-05	.286E-05
3	23.956	.114E-05	.722E-06	.102E-05	.178E-05
2	23.960	.370E-06	.264E-06	.377E-06	.448E-06
2	23.964	.173E-06	.222E-06	.130E-06	.168E-06
3	23.979	.566E-06	.445E-06	.685E-06	.569E-06
4	24.042	.199E-05	.160E-05	.222E-05	.217E-05

5	24.292	.392E-05	.277E-05	.396E-05	.521E-05
6	25.292	.122E-04	.782E-05	.139E-04	.153E-04
6	26.292	.795E-05	.390E-05	.710E-05	.137E-04

END CYCLE NO. 9

5	26.542	.315E-05	.154E-05	.259E-05	.567E-05
5	26.792	.255E-05	.115E-05	.179E-05	.507E-05
4	26.854	.131E-05	.716E-06	.111E-05	.225E-05
4	26.917	.104E-05	.532E-06	.783E-06	.192E-05
3	26.933	.422E-06	.266E-06	.353E-06	.687E-06
3	26.948	.275E-06	.189E-06	.218E-06	.443E-06
2	26.952	.813E-07	.663E-07	.634E-07	.120E-06
2	26.956	.493E-07	.532E-07	.430E-07	.521E-07
3	26.972	.149E-06	.106E-06	.184E-06	.156E-06
4	27.034	.463E-06	.356E-06	.535E-06	.503E-06
5	27.284	.943E-06	.649E-06	.966E-06	.126E-05
6	28.284	.327E-05	.191E-05	.307E-05	.461E-05
6	29.284	.211E-05	.999E-06	.105E-05	.370E-05

END CYCLE NO. 10

5	29.534	.807E-06	.396E-06	.684E-06	.143E-05
5	29.784	.619E-06	.279E-06	.438E-06	.123E-05
4	29.847	.317E-06	.171E-06	.268E-06	.544E-06
4	29.909	.250E-06	.126E-06	.189E-06	.465E-06
3	29.925	.105E-06	.635E-07	.903E-07	.170E-06
3	29.940	.741E-07	.486E-07	.636E-07	.117E-06
2	29.944	.216E-07	.173E-07	.174E-07	.316E-07
2	29.948	.128E-07	.125E-07	.962E-08	.168E-07
3	29.964	.391E-07	.276E-07	.439E-07	.470E-07
4	30.026	.114E-06	.655E-07	.132E-06	.127E-06
5	30.276	.231E-06	.154E-06	.235E-06	.315E-06
6	31.276	.915E-06	.530E-06	.100E-05	.127E-05
6	32.276	.587E-06	.281E-06	.536E-06	.100E-05

END CYCLE NO. 11

5	32.526	.213E-06	.104E-06	.183E-06	.375E-06
5	32.776	.155E-06	.689E-07	.110E-06	.308E-06
4	32.839	.789E-07	.421E-07	.657E-07	.137E-06
4	32.901	.630E-07	.316E-07	.466E-07	.119E-06
3	32.917	.274E-07	.161E-07	.237E-07	.451E-07
3	32.933	.203E-07	.124E-07	.175E-07	.324E-07
2	32.937	.696E-08	.476E-08	.505E-08	.475E-08
2	32.940	.375E-08	.327E-08	.326E-08	.490E-08
3	32.956	.105E-07	.752E-08	.109E-07	.136E-07
4	33.019	.280E-07	.199E-07	.321E-07	.328E-07
5	33.269	.574E-07	.375E-07	.575E-07	.805E-07
6	34.269	.261E-06	.163E-06	.306E-06	.322E-06
6	35.269	.166E-06	.859E-07	.161E-06	.266E-06

END CYCLE NO. 12.

5	35.519	.559E-07	.276E-07	.490E-07	.970E-07
5	35.769	.395E-07	.176E-07	.282E-07	.783E-07
4	35.831	.198E-07	.104E-07	.164E-07	.349E-07
4	35.894	.160E-07	.764E-08	.117E-07	.306E-07
3	35.909	.712E-08	.394E-08	.635E-08	.117E-07

3	35.925	.525E-08	.294E-08	.470E-08	.860E-08
2	35.929	.163E-08	.111E-08	.168E-08	.217E-08
2	35.933	.103E-08	.759E-09	.120E-08	.114E-08
3	35.948	.265E-08	.179E-08	.280E-08	.349E-08
4	36.011	.675E-08	.465E-08	.762E-08	.818E-08
5	36.261	.143E-07	.937E-08	.142E-07	.203E-07
6	37.261	.748E-07	.472E-07	.861E-07	.938E-07
6	38.261	.470E-07	.207E-07	.458E-07	.745E-07

END CYCLE NO. 13

CYCLE NO. 2	CYCLE EFF. .641	ACC.EFF. .641
CYCLE NO. 3	CYCLE EFF. .524	ACC.EFF. .583
CYCLE NO. 4	CYCLE EFF. .585	ACC.EFF. .583
CYCLE NO. 5	CYCLE EFF. .588	ACC.EFF. .584
CYCLE NO. 6	CYCLE EFF. .616	ACC.EFF. .591
CYCLE NO. 7	CYCLE EFF. .630	ACC.EFF. .597
CYCLE NO. 8	CYCLE EFF. .636	ACC.EFF. .602
CYCLE NO. 9	CYCLE EFF. .635	ACC.EFF. .606
CYCLE NO. 10	CYCLE EFF. .642	ACC.EFF. .610
CYCLE NO. 11	CYCLE EFF. .652	ACC.EFF. .614
CYCLE NO. 12	CYCLE EFF. .656	ACC.EFF. .618
CYCLE NO. 13	CYCLE EFF. .655	ACC.EFF. .621

CONVERGENCE FACTOR FOR EACH CYCLE ↑

↑ MEAN CONVERGENCE FACTOR

OUTPUT No 3

```
EQUATIONS                                   NAVIER STOKES (1)(DEL-RUD/DX-RVD/DY) U-PX=F1
                                            (2) (DEL-RUD/DX-RVD/DY)V - PY=F2 -- (3)  UX+VY=F3     F1=F2=F3=0,

BOUNDARY CONDITIONS AND INITIAL FUNCTION VALUES   U=Y*SQRT(R)*FXY   V=X*SQRT(R)*FXY   P=-R**2/2(X**2+Y**2)   (FXY=1+X(1-X)Y(1-Y)  )

METHOD                                      FAS,LIKE INJECTION,SEPARATE RELAXATIONS,LINEAR INT

COARSEST GRID IS     2  X-INTERVALS,AND     2  Y-INTERVALS, H0=  .500
 NUMBER OF GRIDS          6
METHOD PARAMETERS,   ETA=  .200    DELTA=  .300
              TOL=  .100E-04
          ERR  .350  .350  .300
  REYNOLDS NUMBER      .410E+04
```

LEVEL 6 = 64X64 GRID.

LEVEL	ACC. WORK	RESIDUAL NORMS WEIGHTED	EQ. 1	EQ 2	EQ 3
6	1.000	.195E+08	.374E+08	.183E+08	.888E+03
6	2.000	.745E+07	.139E+08	.738E+07	.371E+03
END CYCLE NO. 1					
5	2.250	.653E+06	.140E+07	.471E+06	.478E+02
6	3.250	.330E+06	.665E+06	.278E+06	.114E+03
6	4.250	.147E+05	.359E+05	.616E+04	.693E+02
6	5.250	.416E+03	.944E+03	.198E+03	.525E+02
6	6.250	.160E+03	.256E+03	.164E+03	.430E+02
END CYCLE NO. 2					
5	6.500	.513E+03	.918E+02	.136E+04	.142E+02
5	6.750	.236E+03	.408E+03	.258E+03	.985E+01
4	6.813	.318E+03	.110E+03	.797E+03	.281E+01
4	6.875	.233E+03	.392E+03	.273E+03	.157E+01
3	6.891	.153E+03	.725E+02	.364E+03	.399E+00
3	6.906	.158E+03	.267E+03	.183E+03	.182E+00
2	6.910	.570E+02	.357E+02	.127E+03	.401E-01
2	6.914	.719E+02	.129E+03	.766E+02	.151E-01
1	6.915	.127E+02	.194E+02	.170E+02	.290E-02
2	6.919	.526E+03	.798E+03	.706E+03	.146E-01
2	6.923	.917E+02	.156E+03	.106E+03	.754E-02
2	6.927	.771E+02	.126E+03	.941E+02	.294E-02
1	6.928	.937E+01	.208E+02	.640E+01	.444E-03

2	6.932	.106E+03	.157E+03	.146E+03	.255E-02
2	6.936	.210E+02	.434E+02	.166E+02	.176E-02
3	6.951	.159E+04	.290E+04	.163E+04	.101E+00
3	6.967	.106E+03	.188E+03	.115E+03	.403E-01
3	6.982	.289E+02	.424E+02	.402E+02	.214E-01
4	7.045	.803E+04	.146E+05	.834E+04	.942E+00
4	7.107	.206E+03	.467E+03	.120E+03	.380E+00
4	7.170	.500E+02	.820E+02	.606E+02	.207E+00
5	7.420	.274E+05	.510E+05	.274E+05	.659E+01
5	7.670	.610E+03	.154E+04	.198E+03	.274E+01
5	7.920	.627E+02	.109E+03	.693E+02	.155E+01
5	8.170	.336E+02	.598E+02	.352E+02	.101E+01
6	9.170	.561E+05	.103E+06	.573E+05	.255E+02
6	10.170	.182E+04	.452E+04	.674E+03	.120E+02
6	11.170	.637E+02	.127E+03	.487E+02	.778E+01
6	12.170	.346E+02	.642E+02	.297E+02	.580E+01

END CYCLE NO. 3

5	12.420	.680E+02	.186E+02	.174E+03	.188E+01
5	12.670	.339E+02	.684E+02	.272E+02	.129E+01
4	12.732	.317E+02	.204E+02	.699E+02	.372E+00
4	12.795	.409E+02	.706E+02	.462E+02	.204E+00
3	12.811	.164E+02	.201E+02	.269E+02	.448E-01
3	12.826	.461E+02	.580E+02	.738E+02	.235E-01
2	12.830	.107E+02	.168E+02	.138E+02	.098E-02
3	12.846	.267E+03	.403E+03	.359E+03	.165E-01
3	12.861	.568E+02	.120E+03	.425E+02	.109E-01
2	12.865	.344E+01	.532E+01	.451E+01	.225E-02
3	12.881	.733E+02	.131E+03	.781E+02	.630E-02
3	12.896	.131E+02	.235E+02	.138E+02	.407E-02
3	12.912	.561E+01	.608E+01	.993E+01	.308E-02
4	12.975	.852E+03	.122E+04	.122E+04	.726E-01
4	13.037	.449E+02	.101E+03	.276E+02	.370E-01
4	13.100	.124E+02	.178E+02	.179E+02	.232E-01
3	13.115	.404E+01	.617E+01	.536E+01	.547E-02
3	13.131	.538E+01	.744E+01	.794E+01	.375E-02
2	13.135	.155E+01	.308E+01	.135E+01	.108E-02
3	13.150	.253E+02	.268E+02	.454E+02	.232E-02
3	13.166	.998E+01	.197E+02	.879E+01	.180E-02
2	13.170	.965E+00	.141E+01	.140E+01	.547E-03
3	13.186	.190E+02	.256E+02	.288E+02	.114E-02
3	13.201	.649E+01	.140E+02	.592E+01	.971E-03
2	13.205	.940E+00	.127E+01	.144E+01	.315E-03
3	13.221	.513E+01	.683E+01	.781E+01	.689E-03
3	13.236	.295E+01	.579E+01	.265E+01	.635E-03
4	13.299	.843E+02	.116E+03	.125E+03	.106E-01
4	13.361	.860E+01	.186E+02	.598E+01	.591E-02
5	13.611	.346E+04	.494E+04	.495E+04	.487E+00
5	13.861	.126E+03	.332E+03	.287E+02	.295E+00
5	14.111	.102E+02	.167E+02	.123E+02	.211E+00
6	15.111	.106E+05	.159E+05	.143E+05	.355E+01
6	16.111	[illegible]	[illegible]		

6	16,111	[illegible]	[illegible]	[illegible]	.228E+01
6	17,111	.150E+02	.274E+02	.138E+02	.184E+01
6	18,111	.868E+01	.188E+02	.462E+01	.160E+01

END CYCLE NO. 4

5	18,361	.189E+02	.475E+01	.486E+02	.599E+00
5	18,611	.105E+02	.235E+02	.624E+01	.426E+00
4	18,674	.820E+01	.615E+01	.172E+02	.112E+00
4	18,736	.997E+01	.194E+02	.902E+01	.511E-01
3	18,752	.254E+01	.329E+01	.396E+01	.909E-02
4	18,814	.190E+03	.230E+03	.313E+03	.264E-01
4	18,877	.668E+01	.155E+02	.357E+01	.145E-01
4	18,939	.202E+01	.353E+01	.223E+01	.933E-02
5	19,189	.968E+03	.133E+04	.144E+04	.140E+00
5	19,439	.234E+02	.622E+02	.458E+01	.791E-01
5	19,689	.169E+01	.348E+01	.129E+01	.543E-01
6	20,689	.309E+04	.428E+04	.456E+04	.756E+00
6	21,689	.119E+03	.313E+03	.272E+02	.615E+00
6	22,689	.276E+01	.600E+01	.143E+01	.525E+00
6	23,689	.208E+01	.465E+01	.890E+00	.457E+00

END CYCLE NO. -5

5	23,939	.417E+01	.129E+01	.105E+02	.164E+00
5	24,189	.278E+01	.644E+01	.141E+01	.109E+00
4	24,252	.175E+01	.137E+01	.360E+01	.259E-01
4	24,314	.231E+01	.467E+01	.194E+01	.113E-01
3	24,330	.646E+00	.645E+00	.120E+01	.212E-02
4	24,393	.385E+02	.485E+02	.616E+02	.618E-02
4	24,455	.317E+01	.533E+01	.373E+01	.361E-02
4	24,518	.136E+01	.213E+01	.177E+01	.244E-02
3	24,533	.328E+00	.455E+00	.482E+00	.646E-03
4	24,596	.505E+01	.523E+01	.919E+01	.145E-02
4	24,658	.218E+01	.374E+01	.249E+01	.115E-02
3	24,674	.282E+00	.503E+00	.301E+00	.396E-03
4	24,736	.395E+01	.415E+01	.712E+01	.793E-03
4	24,799	.176E+01	.317E+01	.184E+01	.706E-03
3	24,814	.222E+00	.401E+00	.233E+00	.275E-03
4	24,877	.320E+01	.447E+01	.467E+01	.536E-03
4	24,939	.143E+01	.262E+01	.145E+01	.504E-03
3	24,955	.191E+00	.329E+00	.216E+00	.205E-03
4	25,018	.222E+01	.347E+01	.288E+01	.403E-03
4	25,080	.109E+01	.199E+01	.111E+01	.385E-03
3	25,096	.161E+00	.261E+00	.198E+00	.158E-03
4	25,158	.147E+01	.235E+01	.185E+01	.313E-03
4	25,221	.814E+00	.147E+01	.855E+00	.303E-03
5	25,471	.231E+03	.329E+03	.331E+03	.399E-01
5	25,721	.400E+01	.938E+01	.204E+01	.194E-01
5	25,971	.699E+00	.135E+01	.640E+00	.121E-01
5	26,221	.284E+00	.533E+00	.271E+00	.829E-02
6	27,221	.761E+03	.107E+04	.110E+04	.181E+00
6	28,221	.194E+02	.513E+02	.408E+01	.131E+00
6	29,221	.833E+00	.160E+01	.683E+00	.107E+00
6	30,221	.485E+00	.974E+00	.335E+00	.897E-01

5	30.471	.667E+00	.265E+00	.161E+01	.295E-01
5	30.721	.503E+00	.115E+01	.274E+00	.183E-01
4	30.783	.306E+00	.226E+00	.644E+00	.400E-02
4	30.846	.360E+00	.625E+00	.403E+00	.184E-02
3	30.861	.148E+00	.165E+00	.259E+00	.332E-03
3	30.877	.381E+00	.437E+00	.650E+00	.173E-03
2	30.881	.116E+00	.187E+00	.143E+00	.386E-04
2	30.885	.408E+00	.689E+00	.476E+00	.246E-04
1	30.886	.117E+00	.188E+00	.147E+00	.500E-05
2	30.890	.508E+00	.659E+00	.793E+00	.140E-04
2	30.894	.124E+00	.201E+00	.154E+00	.630E-05
1	30.895	.226E-01	.569E-01	.771E-02	.581E-06
2	30.898	.134E+00	.118E+00	.265E+00	.416E-05
2	30.902	.375E-01	.756E-01	.316E-01	.250E-05
3	30.918	.143E+01	.246E+01	.141E+01	.780E-04
3	30.934	.622E+00	.112E+01	.659E+00	.546E-04
2	30.938	.603E-01	.118E+00	.541E-01	.157E-04
3	30.953	.573E+00	.791E+00	.846E+00	.307E-04
3	30.959	.171E+00	.333E+00	.154E+00	.179E-04
2	30.973	.147E-01	.253E-01	.165E-01	.459E-05
3	30.988	.172E+00	.246E+00	.245E+00	.107E-04
3	31.004	.540E-01	.108E+00	.465E-01	.864E-05
4	31.066	.500E+01	.667E+01	.762E+01	.770E-03
4	31.129	.861E+00	.151E+01	.950E+00	.417E-03
4	31.191	.212E+00	.446E+00	.160E+00	.248E-03
3	31.207	.214E-01	.191E-01	.421E-01	.527E-04
4	31.270	.483E+00	.704E+00	.677E+00	.120E-03
4	31.332	.127E+00	.226E+00	.138E+00	.817E-04
5	31.582	.359E+02	.517E+02	.509E+02	.725E-02
5	31.832	.992E+00	.187E+01	.961E+00	.366E-02
5	32.082	.274E+00	.541E+00	.239E+00	.232E-02
4	32.145	.576E-01	.511E-01	.113E+00	.528E-03
5	32.395	.291E+01	.403E+01	.430E+01	.121E-02
5	32.645	.916E-01	.166E+00	.958E-01	.820E-03
6	33.645	.131E+03	.190E+03	.186E+03	.344E-01
6	34.645	.253E+01	.624E+01	.963E+00	.209E-01
6	35.645	.272E+00	.525E+00	.240E+00	.159E-01
6	36.645	.815E-01	.167E+00	.556E-01	.126E-01

END CYCLE NO.- 7

5	36.895	.977E-01	.380E-01	.238E+00	.390E-02
5	37.145	.717E-01	.149E+00	.543E-01	.241E-02
4	37.207	.506E-01	.312E-01	.113E+00	.569E-03
4	37.270	.575E-01	.843E-01	.796E-01	.293E-03
3	37.285	.205E-01	.184E-01	.402E-01	.590E-04
3	37.301	.598E-01	.816E-01	.892E-01	.326E-04
2	37.305	.201E-01	.319E-01	.245E-01	.860E-05
2	37.309	.702E-01	.827E-01	.118E+00	.584E-05
1	37.310	.230E-01	.392E-01	.266E-01	.129E-05
1	37.311	.301E-01	.460E-01	.400E-01	.714E-06
1	37.312	.177E-01	.263E-01	.242E-01	.275E-06
2	37.315	.947E-01	.114E+00	.157E+00	[illegible]

2	37.319	.541E-01	.976E-01	.571E-01	.211E-05
1	37.320	.257E-02	.697E-02	.380E-03	.408E-06
2	37.324	.695E-01	.777E-01	.121E+00	.190E-05
2	37.328	.162E-01	.312E-01	.151E-01	.121E-05
3	37.344	.264E+00	.351E+00	.404E+00	.174E-04
3	37.359	.142E+00	.266E+00	.139E+00	.121E-04
2	37.363	.103E-01	.230E-01	.642E-02	.292E-05
3	37.379	.964E-01	.110E+00	.165E+00	.638E-05
3	37.395	.246E-01	.457E-01	.247E-01	.390E-05
2	37.398	.264E-02	.367E-02	.387E-02	.102E-05
3	37.414	.390E-01	.595E-01	.519E-01	.268E-05
3	37.430	.630E-02	.116E-01	.644E-02	.197E-05
4	37.492	.716E+00	.117E+01	.881E+00	.111E-03
4	37.555	.155E+00	.281E+00	.160E+00	.705E-04
3	37.570	.222E-01	.387E-01	.246E-01	.158E-04
4	37.633	.152E+00	.248E+00	.186E+00	.335E-04
4	37.695	.215E-01	.376E-01	.237E-01	.211E-04
4	37.758	.891E-02	.137E-01	.118E-01	.156E-04
5	38.008	.425E+01	.609E+01	.604E+01	.872E-03
5	38.258	.223E+00	.458E+00	.177E+00	.520E-03
5	38.508	.599E-01	.103E+00	.677E-01	.360E-03
4	38.570	.162E-01	.173E-01	.288E-01	.948E-04
5	38.820	.385E+00	.568E+00	.532E+00	.189E-03
5	39.070	.194E-01	.389E-01	.166E-01	.145E-03
6	40.070	.165E+02	.240E+02	.232E+02	.502E-02
6	41.070	.443E+00	.103E+01	.237E+00	.293E-02
6	42.070	.650E-01	.112E+00	.721E-01	.220E-02
6	43.070	.292E-01	.527E-01	.293E-01	.180E-02

**END CYCLE NO. 8

5	43.320	.214E-01	.820E-02	.525E-01	.618E-03
5	43.570	.115E-01	.245E-01	.791E-02	.426E-03
4	43.633	.109E-01	.490E-02	.262E-01	.110E-03
4	43.695	.987E-02	.154E-01	.127E-01	.600E-04
3	43.711	.479E-02	.421E-02	.948E-02	.138E-04
3	43.727	.120E-01	.897E-02	.254E-01	.714E-05
2	43.730	.296E-02	.424E-02	.423E-02	.171E-05
3	43.746	.601E-01	.810E-01	.906E-01	.392E-05
3	43.762	.133E-01	.287E-01	.931E-02	.320E-05
2	43.766	.212E-02	.279E-02	.327E-02	.997E-06
3	43.781	.265E-01	.438E-01	.320E-01	.233E-05
3	43.797	.594E-02	.108E-01	.617E-02	.185E-05
2	43.801	.154E-02	.225E-02	.216E-02	.600E-06
3	43.816	.652E-02	.704E-02	.116E-01	.128E-05
3	43.832	.440E-02	.851E-02	.406E-02	.113E-05
2	43.836	.109E-02	.150E-02	.163E-02	.379E-06
3	43.852	.407E-02	.622E-02	.542E-02	.830E-06
3	43.867	.239E-02	.448E-02	.234E-02	.723E-06
4	43.930	.189E+00	.290E+00	.251E+00	.219E-04
4	43.992	.288E-01	.562E-01	.262E-01	.143E-04
4	44.055	.107E-01	.180E-01	.122E-01	.954E-05
3	44.070	.198E-02	.276E-02	.288E-02	[illegible]

3	44.070	.198E-02	.776E-02	.288E-02	.227E-05
4	44.133	.276E-01	.427E-01	.362E-01	.460E-05
4	44.195	.253E-02	.547E-02	.177E-02	.286E-05
5	44.445	.739E+00	.112E+01	.969E+00	.126E-03
5	44.695	.587E-01	.130E+00	.379E-01	.960E-04
5	44.945	.140E-01	.233E-01	.167E-01	.749E-04
4	45.008	.386E-02	.427E-02	.674E-02	.222E-04
5	45.258	.908E-01	.137E+00	.122E+00	.377E-04
5	45.508	.527E-02	.110E-01	.348E-02	.299E-04
6	46.508	.241E+01	.344E+01	.345E+01	.693E-03
6	47.508	.126E+00	.292E+00	.673E-01	.526E-03
6	48.508	.168E-01	.243E-01	.193E-01	.438E-03
6	49.508	.705E-02	.128E-01	.702E-02	.377E-03

*********************************END CYCLE NO. 9

5	49.758	.514E-02	.162E-02	.129E-01	.136E-03
5	50.008	.223E-02	.880E-02	.150E-02	.948E-04
4	50.070	.252E-02	.123E-02	.594E-02	.261E-04
4	50.133	.229E-02	.429E-02	.224E-02	.146E-04
3	50.148	.923E-03	.906E-03	.173E-02	.308E-05
3	50.164	.203E-02	.397E-02	.183E-02	.131E-05
2	50.168	.244E-03	.410E-03	.287E-03	.228E-06
3	50.184	.113E-01	.123E-01	.201E-01	.617E-06
3	50.199	.256E-02	.563E-02	.167E-02	.362E-06
2	50.203	.172E-03	.180E-03	.312E-03	.661E-07
3	50.219	.248E-02	-.319E-02	.389E-02	.223E-06
3	50.234	.785E-03	.153E-02	.711E-03	.136E-06
2	50.238	.116E-03	.170E-03	.161E-03	.358E-07
3	50.254	.112E-02	.141E-02	.179E-02	.862E-07
3	50.270	.192E-03	.336E-03	.214E-03	.612E-07
4	50.332	.465E-01	.647E-01	.631E-01	.483E-05
4	50.395	.296E-02	.680E-02	.165E-02	.265E-05
4	50.457	.809E-03	.136E-02	.949E-03	.161E-05
3	50.473	.162E-03	.214E-03	.247E-03	.336E-06
4	50.535	.463E-02	.680E-02	.643E-02	.859E-06
4	50.598	.442E-03	.790E-03	.474E-03	.513E-06
5	50.848	.195E+00	.291E+00	.267E+00	.315E-04
5	51.098	.115E-01	.284E-01	.439E-02	.238E-04
5	51.348	.135E-02	.221E-02	.164E-02	.181E-04
6	52.348	.557E+00	.802E+00	.789E+00	.147E-03
6	53.348	.353E-01	.882E-01	.140E-01	.135E-03
6	54.348	.197E-02	.323E-02	.229E-02	.120E-03
6	55.348	.855E-03	.156E-02	.794E-03	.108E-03

*********************************END CYCLE NO. 10

5	55.598	.150E-02	.269E-03	.398E-02	.420E-04
5	55.848	.598E-03	.143E-02	.257E-03	.312E-04
4	55.910	.686E-03	.422E-03	.153E-02	.861E-05
4	55.973	.598E-03	.141E-02	.295E-03	.430E-05
3	55.988	.205E-03	.269E-03	.316E-03	.807E-06
3	56.004	.379E-03	.543E-03	.539E-03	.340E-06
2	56.008	.595E-04	.110E-03	.604E-04	.500E-07
3	56.023	.313E-02	.518E-02	.377E-02	.230E-06

3	56.039	.440E-03	.778E-03	.478E-03	.132E-06
3	56.055	.179E-03	.154E-03	.359E-03	.783E-07
4	56.117	.136E-01	.214E-01	.174E-01	.140E-05
4	56.180	.614E-03	.119E-02	.561E-03	.726E-06
4	56.242	.161E-03	.300E-03	.161E-03	.428E-06
5	56.492	.700E-01	.106E+00	.940E-01	.975E-05
5	56.742	.266E-02	.679E-02	.815E-03	.664E-05
5	56.992	.267E-03	.471E-03	.286E-03	.484E-05
5	57.242	.881E-04	.185E-03	.641E-04	.367E-05
6	58.242	.200E+00	.269E+00	.283E+00	.485E-04
6	59.242	.111E-01	.282E-01	.339E-02	.445E-04
6	60.242	.381E-03	.630E-03	.425E-03	.394E-04
6	61.242	.198E-03	.392E-03	.144E-03	.351E-04
END CYCLE NO. 11					
5	61.492	.388E-03	.103E-03	.994E-03	.131E-04
5	61.742	.225E-03	.534E-03	.102E-03	.913E-05
4	61.805	.157E-03	.129E-03	.316E-03	.225E-05
4	61.867	.158E-03	.382E-03	.674E-04	.102E-05
3	61.883	.470E-04	.530E-04	.812E-04	.180E-06
4	61.945	.378E-02	.599E-02	.482E-02	.550E-06
4	62.008	.102E-03	.161E-03	.132E-03	.333E-06
4	62.070	.720E-04	.101E-03	.105E-03	.222E-06
3	62.086	.216E-04	.269E-04	.349E-04	.537E-07
3	62.102	.784E-04	.430E-04	.181E-03	.364E-07
2	62.105	.131E-04	.243E-04	.130E-04	.931E-08
3	62.121	.492E-03	.427E-03	.940E-03	.228E-07
3	62.137	.179E-03	.378E-03	.134E-03	.197E-07
2	62.141	.125E-04	.147E-04	.162E-04	.473E-08
3	62.156	.160E-03	.233E-03	.225E-03	.929E-08
3	62.172	.440E-04	.893E-04	.363E-04	.786E-08
2	62.176	.570E-05	.677E-05	.951E-05	.217E-08
3	62.191	.511E-04	.602E-04	.858E-04	.489E-08
3	62.207	.154E-04	.266E-04	.154E-04	.422E-08
4	62.270	.702E-03	.104E-02	.967E-03	.843E-07
4	62.332	.931E-04	.191E-03	.750E-04	.499E-07
4	62.395	.259E-04	.415E-04	.325E-04	.334E-07
5	62.645	.190E-01	.293E-01	.250E-01	.258E-05
5	62.895	.443E-03	.118E-02	.871E-04	.148E-05
5	63.145	.394E-04	.729E-04	.388E-04	.989E-06
6	64.145	.648E-01	.949E-01	.901E-01	.153E-04
6	65.145	.250E-02	.654E-02	.596E-03	.129E-04
6	66.145	.673E-04	.125E-03	.575E-04	.110E-04
6	67.145	.494E-04	.104E-03	.288E-04	.951E-05
END CYCLE NO. 12					
5	67.395	.810E-04	.303E-04	.198E-03	.331E-05
5	67.645	.560E-04	.140E-03	.179E-04	.213E-05
4	67.707	.249E-04	.317E-04	.534E-04	.468E-06
4	67.770	.343E-04	.802E-04	.176E-04	.203E-06
3	67.785	.817E-05	.747E-05	.158E-04	.314E-07
4	67.848	.760E-03	.121E-02	.102E-02	.122E-06
4	67.910	.275E-04	.470E-04	.313E-04	.708E-07

OUTPUT NO. 4

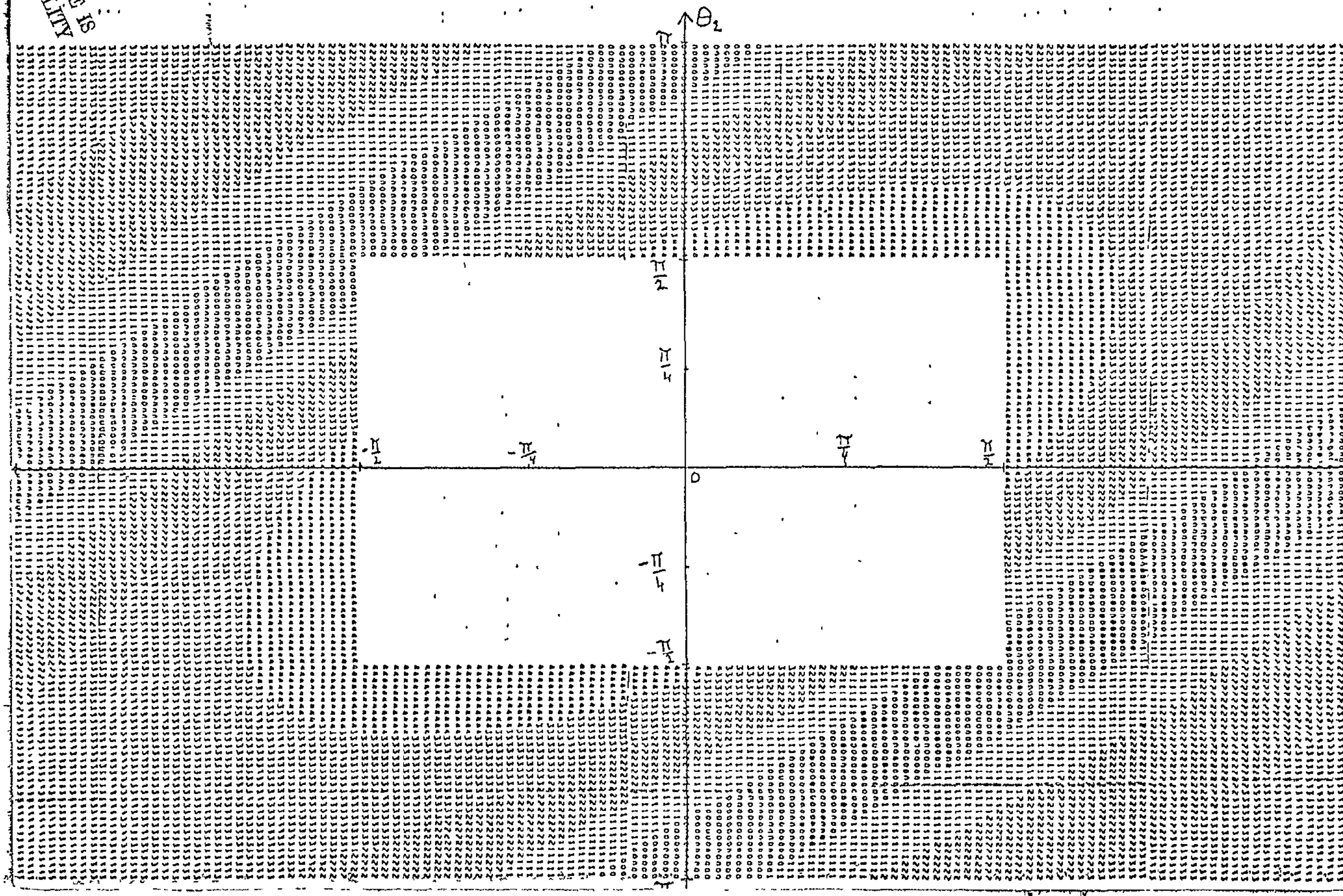
SMOOTHING FACTOR = .5000 THETA=1.571 =.638
WEIGHTED SMOOTHING .2787 ($\lambda_\theta = 1$)

POISSON OPERATOR

	1.0000	
1.0000	-4.0000	1.0000
	1.0000	

USUAL GAUSS-SEIDEL RELAXATION

EACH NUMBER J IN THE MAP REPRESENTS $\frac{J}{10} \le \mu(\theta) < \frac{J+1}{10}$



4	67.973	.115E-04	.142E-04	.187E-04	.435E-07
5	68.223	.406E-02	.617E-02	.543E-02	.628E-06
5	68.473	.669E-04	.167E-03	.242E-04	.334E-06
5	68.723	.929E-05	.179E-04	.840E-05	.215E-06
6	69.723	.158E-01	.234E-01	.217E-01	.365E-05
6	70.723	.435E-03	.115E-02	.910E-04	.279E-05
6	71.723	.147E-04	.284E-04	.116E-04	.227E-05
6	72.723	.126E-04	.231E-04	.112E-04	.190E-05

END CYCLE NO. 13

	Convergence Factor for each Cycle	Mean Convergence Factor
CYCLE NO. 2	CYCLE EFF. .080	ACC.EFF. .080
CYCLE NO. 3	CYCLE EFF. .772	ACC.EFF. .299
CYCLE NO. 4	CYCLE EFF. .792	ACC.EFF. .428
CYCLE NO. 5	CYCLE EFF. .774	ACC.EFF. .499
CYCLE NO. 6	CYCLE EFF. .800	ACC.EFF. .556
CYCLE NO. 7	CYCLE EFF. .758	ACC.EFF. .589
CYCLE NO. 8	CYCLE EFF. .852	ACC.EFF. .624
CYCLE NO. 9	CYCLE EFF. .802	ACC.EFF. .646
CYCLE NO. 10	CYCLE EFF. .697	ACC.EFF. .651
CYCLE NO. 11	CYCLE EFF. .780	ACC.EFF. .663
CYCLE NO. 12	CYCLE EFF. .790	ACC.EFF. .674
CYCLE NO. 13	CYCLE EFF. .782	ACC.EFF. .682

ORIGINAL PAGE IS OF POOR QUALITY