# Direct Solution of Elliptic Equations by Block Cyclic Reduction and Factorization

THOMAS E. ROSMOND

*Environmental Prediction Research Facility, Monterey, Calif. 93940*

FRANK D. FAULKNER

*U. S. Naval Postgraduate School, Monterey, Calif. 93940*

(Manuscript received 17 July 1975, in revised form 21 November 1975 and 14 January 1976)

ABSTRACT

Poisson's and Helmholtz's equations are perhaps the most frequently occurring and important types of partial differential equations encountered in the atmospheric sciences. This paper presents a very fast, accurate technique for finding the numerical solution known as cyclic reduction and factoralization. This method has not heretofore been brought to the attention of the meteorological community at large.

This direct method essentially reduces the solution of a separable two-dimensional elliptic equation on an $N\times M$ grid to $N\log_2 N$ tri-diagonal systems of order $M$ which are solved by Gaussian elimination. In its simplest form, as described here, the cyclic reduction procedure can be applied if $N$ is $2^n-1$, $2^n$ or $2^n+1$, depending on boundary conditions. However, extensions of the method have been developed which have removed this restrictive limitation. The method is also easily generalized to higher dimensional problems.

The mathematical development of the cyclic reduction method is presented here in complete detail, along with the modifications necessary to make it computationally stable. The results of two numerical experiments comparing optimized SOR versus the direct method for the solution of Poisson's equation are presented. For Dirichlet boundary conditions the direct method is up to 50 times faster than successive over-relaxation (SOR) for $N=M=128$. For Neumann boundary conditions, the direct method has even a greater advantage over SOR. The margin of superiority increases as the size of the array increases.

## 1. Introduction

The need to solve a separable, elliptic, partial differential equation of the form

$$\frac{\partial^2\psi}{\partial y^2}+K_1(x)\frac{\partial^2\psi}{\partial x^2}+K_2(x)\frac{\partial\psi}{\partial x}-K_3(x)\psi=F(x,y), \quad (1)$$

is one of the most frequently encountered mathematical problems in the atmospheric sciences. In this paper, several typical elliptic equations which meteorologists must often solve are described and a very fast, accurate direct solution technique is presented which can be applied to these problems. In almost all applications, this method will be shown to yield remarkable benefits in computation time and accuracy when compared to the commonly used iterative methods (e.g., successive over-relaxation).

Among the most frequently encountered elliptic equations in meteorology are the following:

(i) Two-dimensional time integrations of stream-function ($\psi$)-vorticity ($\zeta$) models: an equation of the form

$$\nabla^2\psi(x,y)=-\zeta(x,y), \quad (2)$$

a Poisson equation which is usually solved with Dirichlet boundary conditions.

(ii) Barotropic, free-surface, primitive equation models with semi-implicit time differencing:

$$\nabla_H{}^2h(x,y)-\alpha h(x,y)=F(x,y), \quad (3)$$

a two-dimensional (horizontal plane) Helmholtz equation where $h$ is the free surface height, $\alpha$ a coefficient which may or may not be variable, and $F(x,y)$ a forcing function. This equation is usually solved with a combination of Neumann and periodic boundary conditions.

In the filtered meteorological equations, the diagnostic equations for the geopotential tendency and the vertical motion field (omega) are also of the form of (3).

(iii) Primitive equation, stratified fluid models with semi-implicit time differencing:

$$\frac{\partial^2P}{\partial x^2}+\frac{\partial}{\partial z}\left[\lambda(z)\frac{\partial P}{\partial z}\right]=F(x,y), \quad (4)$$

a modified Poisson equation, where $P$ is the perturbation pressure, $\lambda$ a stability parameter, and $F$ a forcing function. This equation is usually solved with Neumann boundary equations.

(iv) The linear balance equation:

$$f\nabla^2\psi+\nabla f\cdot\nabla\psi=\nabla^2\phi, \quad (5)$$

a modified Poisson equation which describes the relationship between the rotational wind component and

the geopotential field. It is widely used in initialization procedures to find an initial wind field when the geopotential is given.

In addition, elliptic equations occur in the solution of variational problems associated with initialization. (Sasaki, 1970).

Direct solution of elliptic equations through matrix manipulation is a fairly recent development in the field of numerical mathematics. Fourier transform methods (Hockney, 1965) are well known to many meteorologists and have been shown to yield considerable benefits in computer time saving when repeated solution of a Poisson or Helmholtz equation is required, such as in time integrations of the primitive equations (Williams, 1969). However, because the fast Fourier transform (Cooley and Tukey, 1965) can be applied only to a fairly restrictive system of equations, its use has not been too widespread.

The block-cyclic reduction method, to be described here, was introduced by Buneman (1969), and is described in considerable detail by Buzbee *et al.* (1970) and Sweet (1973a) for the various boundary conditions.

While the description in this paper will be primarily directed at the direct solution of two-dimensional elliptic equations on rectangular domains, more complicated problems of this type have recently received considerable attention. The direct solution of Poisson's equation on irregular regions has been described by Buzbee *et al.* (1971). Buzbee and Dorr (1974) discuss the solution of the biharmonic equation by direct methods. The solution of two-dimensional elliptic equations of the form of (1) in which the coefficients $K$ may be functions of $x$ and $y$ (non-separable) has been described by Concus and Golub (1973) and Faulkner and Rosmond (1975). These methods require the iterative application of a direct solver.

The solution of three-dimensional elliptic equations is of considerable interest to meteorologists because of the necessity of solving this type of equation in cloud modeling problems and three-dimensional, semi-implicit time integrations of the primitive equations. Solution of a three-dimensional elliptic equation requires the repeated application of a two-dimensional solver, the number of applications being proportioned to $L\log_2 L$, where $L$ is the number of intervals in the third dimension.

Martin (1973) has published a three-dimensional algorithm for solving the Dirichlet problem. In meteorology, however, because three-dimensional problems usually yield a Poisson equation for the pressure or geopotential field, Neumann boundary conditions are more appropriate. The benefits of direct solution methods are most apparent with Neumann conditions because the singular nature of the resulting matrix equations makes iterative methods extremely slow to converge. Rosmond (1975) has developed a direct solution algorithm for solving the three-dimensional Poisson equation with Neumann boundary conditions.

In the mathematical developments of the following sections, we shall not distinguish between Poisson and separable Helmholtz equations of the form of (1) because their differences are irrelevant insofar as the method is concerned. However, in practice the differences between them can be important because: 1) if the equation is non-separable, it cannot be solved with one application of a direct solver; and 2) if a Helmholtz equation is very diagonally dominant and a good first-guess solution is available, iterative methods converge very rapidly and may be competitive with direct methods. We shall discuss this point further in Section 6.

## 2. Matrix representation of Poisson equation

In finite difference form (1) becomes

$$\frac{\psi(i,j+1)-2\psi(i,j)+\psi(i,j-1)}{\Delta y^2}+\frac{K_1(i)[\psi(i+1,j)-2\psi(i,j)+\psi(i-1,j)]}{\Delta X^2}+\frac{K_2(i)[\psi(i+1,j)-\psi(i-1,j)]}{2\Delta X}-K_3(i)\psi(i,j)=F(i,j), \quad (6)$$

for $i=1, M, j=1, N$; or

$$A(i)\psi(i-1,j)+B(i)\psi(i,j)+C(i)\psi(i+1,j)-\psi(i,j+1)-\psi(i,j-1)=G(i,j), \quad (7)$$

where

$$A(i)=\left(\frac{\Delta y}{\Delta x}\right)^2\left[\frac{\Delta x K_2(i)}{2}-K_1(i)\right], \quad (8)$$

$$B(i)=2+\left(\frac{\Delta y}{\Delta x}\right)^2[K_3(i)\Delta X^2+2K_1(i)], \quad (9)$$

$$C(i)=\left(\frac{\Delta y}{\Delta x}\right)^2\left[-\frac{\Delta x K_2(i)}{2}-K_1(i)\right], \quad (10)$$

$$G(i,j)=-\Delta y^2 F(i,j). \quad (11)$$

The change in signs is to yield matrix forms which follow conventional notation by having positive terms on the main diagonal.

In matrix form (7) is

$$\mathbf{D}\Psi=\mathbf{G}, \quad (12)$$

representing a system of simultaneous linear equations. There are two ways to expand (12) into its component equations using matrix forms. The first method is to express $\mathbf{D}$ as a very sparse $(M\times N)^2$ matrix and $\Psi$ and $\mathbf{G}$ as column vectors with $M\times N$ elements (Dingle and Young, 1965). In principle, this form can be solved

by Gaussian elimination or some other form of matrix inversion. In practical problems, however, the quantity $M \times N$ is far too large to allow this method because of computer expense.

The second way of expanding (12) is as follows: Define

$$\mathbf{D}=\begin{bmatrix} \mathbf{E} & -\mathbf{I} & 0 & & & & \bigcirc \\ -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & & & \\ 0 & -\mathbf{I} & \mathbf{E} & \ddots & 0 & & \\ & & \ddots & \ddots & -\mathbf{I} & & \\ & & 0 & -\mathbf{I} & \ddots \mathbf{E} & -\mathbf{I} & 0 \\ & & & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} \\ \bigcirc & & & & 0 & -\mathbf{I} & \mathbf{E} \end{bmatrix}_{N\times N}, \quad (13)$$

$$\mathbf{\Psi}=\begin{bmatrix} \boldsymbol{\psi}_{i,1} \\ \boldsymbol{\psi}_{i,2} \\ \vdots \\ \boldsymbol{\psi}_{i,N-1} \\ \boldsymbol{\psi}_{i,N} \end{bmatrix}, \qquad \mathbf{G}=\begin{bmatrix} \mathbf{g}_{i,1} \\ \mathbf{g}_{i,2} \\ \vdots \\ \mathbf{g}_{i,N-1} \\ \mathbf{g}_{i,N} \end{bmatrix}, \quad (14), (15)$$

$$\mathbf{E}=\begin{bmatrix} B_1 & C_1 & 0 & & & & \bigcirc \\ A_2 & B_2 & C_2 & 0 & & & \\ 0 & A_3 & B_3 & C_3 & 0 & & \\ & & \ddots & \ddots & \ddots & & \\ & & 0 & A_i & B_i & C_i & 0 \\ & & & 0 & A_{M-1} & B_{M-1} & C_{M-1} \\ \bigcirc & & & & 0 & A_M & B_M \end{bmatrix}_{M\times M}. \quad (16)$$

Also, the individual components of the column vectors $\mathbf{\Psi}$ and $\mathbf{G}$ given by (14) and (15) are themselves column vectors of order $M$. Also,

$$\mathbf{I}=\begin{bmatrix} 1 & 0 & & & & & \bigcirc \\ 0 & 1 & 0 & & & & \\ & 0 & 1 & & & & \\ & & & 0 & 1 & 0 & \\ & & & & 0 & 1 & 0 \\ \bigcirc & & & & & 0 & 1 \end{bmatrix}_{M\times M}. \quad (17)$$

To show that the above forms combine to yield (7), we shall expand the matrix equations at one point, say $i=1, j=1$. From (13), (14) and (15) we write the expansion of the first row of (12), i.e.,

$$\mathbf{E}\psi(i,1)-\mathbf{I}\psi(i,2)=\mathbf{g}(i,1). \quad (18)$$

Since we want the equation for $i=1$, we expand each term of (18) for this index. From (16) and (17) this yields

$$B(1)\psi(1,1)+C(1)\psi(2,1)-\psi(1,2)=\mathbf{g}(1,1). \quad (19)$$

If we compare (19) to (7), we see that if

$$\mathbf{g}(1,1)=\mathbf{G}(1,1)-A(1)\psi(0,1)+\psi(1,0), \quad (20)$$

the equations are identical. But these transferred terms are simply the boundary values we must provide to solve an elliptic equation with Dirichlet boundary conditions. Therefore, (13) is the appropriate form of the matrix $\mathbf{D}$ when we have Dirichlet boundary conditions at $J=0$ and $J=N+1$. In our example the matrix $\mathbf{E}$ is also appropriate for Dirichlet conditions at $i=0$ and $i=M+1$. However, because the terms of $\mathbf{E}$ are the variable coefficients of $\psi$ in (7), by suitable modification of the coefficients at $i=1$ and $i=M$, we can make $E$ correspond to the matrix of coefficients for any nonperiodic boundary conditions at $i=0$ and $i=M+1$. We can use this fact to write quite general subroutines for solving equations of the form of (7).

In Section 5, we shall present the forms that $\mathbf{D}$ assumes when we have Neumann or periodic boundary conditions at $j=0$ and $j=N+1$.

### 3. Block-cyclic reduction

To illustrate the method of direct solution by cyclic reduction, we use an example with $N=7$. The reason for this value of $N$ will become evident as we demonstrate the reduction procedure. We use the matrices given in Section 2 for this example, already identified as those corresponding to Dirichlet conditions at the boundaries. The matrix equation corresponding to (12) is

$$\begin{bmatrix} \mathbf{E} & -\mathbf{I} & 0 & 0 & 0 & 0 & 0 \\ -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & 0 & 0 & 0 \\ 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} \\ 0 & 0 & 0 & 0 & 0 & -\mathbf{I} & \mathbf{E} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi}_{i,1} \\ \boldsymbol{\psi}_{i,2} \\ \boldsymbol{\psi}_{i,3} \\ \boldsymbol{\psi}_{i,4} \\ \boldsymbol{\psi}_{i,5} \\ \boldsymbol{\psi}_{i,6} \\ \boldsymbol{\psi}_{i,7} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{i,1} \\ \mathbf{g}_{i,2} \\ \mathbf{g}_{i,3} \\ \mathbf{g}_{i,4} \\ \mathbf{g}_{i,5} \\ \mathbf{g}_{i,6} \\ \mathbf{g}_{i,7} \end{bmatrix}. \quad (21)$$

We multiply the even indexed rows by $\mathbf{E}$ and add to this result the respective rows above and below the even indexed rows. The even index rows are replaced by the modified rows just computed. The new matrix equation is then

$$\begin{bmatrix} \mathbf{E} & -\mathbf{I} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{E}^2-2\mathbf{I} & 0 & -\mathbf{I} & 0 & 0 & 0 \\ 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & 0 & 0 \\ 0 & -\mathbf{I} & 0 & \mathbf{E}^2-2\mathbf{I} & 0 & -\mathbf{I} & 0 \\ 0 & 0 & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 \\ 0 & 0 & 0 & -\mathbf{I} & 0 & \mathbf{E}^2-2\mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & -\mathbf{I} & \mathbf{E} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi}_{i,1} \\ \boldsymbol{\psi}_{i,2} \\ \boldsymbol{\psi}_{i,3} \\ \boldsymbol{\psi}_{i,4} \\ \boldsymbol{\psi}_{i,5} \\ \boldsymbol{\psi}_{i,6} \\ \boldsymbol{\psi}_{i,7} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{i,1} \\ \mathbf{g}_{i,1}+\mathbf{g}_{i,3}+\mathbf{E}\mathbf{g}_{i,2} \\ \mathbf{g}_{i,3} \\ \mathbf{g}_{i,3}+\mathbf{g}_{i,5}+\mathbf{E}\mathbf{g}_{i,4} \\ \mathbf{g}_{i,5} \\ \mathbf{g}_{i,5}+\mathbf{g}_{i,7}+\mathbf{E}\mathbf{g}_{i,6} \\ \mathbf{g}_{i,7} \end{bmatrix}. \quad (22)$$

Expanding the matrix for $j=4$ we get the equation

$$(\mathbf{E}^2-2\mathbf{I})\psi_{i,4}-\mathbf{I}(\psi_{i,2}+\psi_{i,6})=\mathbf{g}_{i,3}+\mathbf{g}_{i,5}+\mathbf{E}\mathbf{g}_{i,4}, \quad (23)$$

an equation which contains only the even indexed $\psi$'s. For $j=2$ and 6 similar expansions occur. Because of this result, we can write a reduced matrix equation for the even indexed $\psi$'s. This equation is

$$\begin{bmatrix} \mathbf{E}^2-2\mathbf{I} & -\mathbf{I} & 0 & \psi_{i,2} \\ -\mathbf{I} & \mathbf{E}^2-2\mathbf{I} & -\mathbf{I} & \psi_{i,4} \\ 0 & -\mathbf{I} & \mathbf{E}^2-2\mathbf{I} & \psi_{i,6} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_{i,1}+\mathbf{g}_{i,3}+\mathbf{E}\mathbf{g}_{i,2} \\ \mathbf{g}_{i,3}+\mathbf{g}_{i,5}+\mathbf{E}\mathbf{g}_{i,4} \\ \mathbf{g}_{i,5}+\mathbf{g}_{i,7}+\mathbf{E}\mathbf{g}_{i,6} \end{bmatrix}. \quad (24)$$

The odd-indexed equations make up a system we refer to as the "eliminated equations."

It is immediately obvious that (24) is similar in form to (21) and that the reduction process can therefore be repeated. We write (24) as

$$\begin{bmatrix} \mathbf{E}^{(1)} & -\mathbf{I} & 0 \\ -\mathbf{I} & \mathbf{E}^{(1)} & -\mathbf{I} \\ 0 & -\mathbf{I} & \mathbf{E}^{(1)} \end{bmatrix} \begin{bmatrix} \psi_{i,2} \\ \psi_{i,4} \\ \psi_{i,6} \end{bmatrix} = \begin{bmatrix} \mathbf{g}^{(1)}_{i,2} \\ \mathbf{g}^{(1)}_{i,4} \\ \mathbf{g}^{(1)}_{i,6} \end{bmatrix}. \quad (25)$$

Performing the reduction about the middle row, we have

$$[(\mathbf{E}^{(1)})^2-2\mathbf{I}]\psi_{i,4}=\mathbf{g}^{(1)}_{i,2}+\mathbf{g}^{(1)}_{i,6}+\mathbf{E}\mathbf{g}^{(1)}_{i,4}, \quad (26)$$

or

$$\mathbf{E}^{(2)}\psi_{i,4}=\mathbf{g}^{(2)}_{i,4}. \quad (27)$$

To solve (27) we must factor the polynomial

$$\mathbf{E}^{(2)}=(\mathbf{E}^{(1)})^2-2\mathbf{I}=(\mathbf{E}^2-2\mathbf{I})^2-2\mathbf{I}, \quad (28)$$

a fourth-order polynomial in $\mathbf{E}$. Eq. (28) can be expressed as

$$\mathbf{E}^{(2)}=(\mathbf{E}-\alpha^{(2)}_1\mathbf{I})(\mathbf{E}-\alpha^{(2)}_2\mathbf{I}) \times(\mathbf{E}-\alpha^{(2)}_3\mathbf{I})(\mathbf{E}-\alpha^{(2)}_4\mathbf{I}). \quad (29)$$

To find the roots $\alpha_l$ we note from (28) that

$$\mathbf{E}^{(r+1)}=(\mathbf{E}^{(r)})^2-2\mathbf{I}, \quad (30)$$

as is easily shown by induction. This expression is identical in form to the trigonometric identity

$$2\cos(2^{r+1}\phi)=[2\cos(2^r\phi)]^2-2.$$

Therefore, we write

$$\mathbf{E}^{(r)}=2\cos(2^r\phi). \quad (31)$$

Those familiar with orthogonal polynomials will recognize the cosine function on the right-hand side of (31) as a Chebyshev polynomial of the first kind, $C_{2^r}(2\cos\phi)$ [Abramowitz and Stegun, 1965]. The zeros of this polynomial are

$$\alpha^{(r)}_j=2\cos\left[\left(\frac{2j-1}{2^{r+1}}\right)\pi\right], \quad j=1, 2, \ldots, 2^r. \quad (32)$$

Thus, we have

$$\mathbf{E}^{(r)}=\prod_{j=1}^{2^r}\left\{\mathbf{E}-2\cos\left[\left(\frac{2j-1}{2^{r+1}}\right)\pi\right]\mathbf{I}\right\}=0. \quad (33)$$

We now are able to calculate the values of the $\alpha^{(2)}$ in (29) using (32). We proceed to solve (27) by noting that it can be written as

$$(\mathbf{E}-\alpha^{(2)}_1\mathbf{I})\zeta_{i,4}=\mathbf{g}_{i,4}, \quad (34)$$

where

$$\zeta_{i,4}=(\mathbf{E}-\alpha^{(2)}_2\mathbf{I})(\mathbf{E}-\alpha^{(2)}_3\mathbf{I})(\mathbf{E}-\alpha^{(2)}_4\mathbf{I})\psi_{i,4}. \quad (35)$$

Equation (34) is solved by tri-diagonal, Gaussian elimination. We then repeat the Gaussian elimination procedure for the $(\mathbf{E}-\alpha^{(2)}_2)$ factor in (35), obtaining another intermediate solution vector. It is immediately apparent in this example that four Gaussian eliminations are necessary to obtain $\psi_{i,4}$.

To continue the process of solution we expand (25) to obtain

$$\mathbf{E}^{(1)}\psi_{i,2}=\mathbf{g}^{(1)}_{i,2}+\mathbf{I}\psi_{i,4}. \quad (36)$$

Here $\mathbf{E}^{(1)}$ is a second-order polynomial in $\mathbf{E}$ whose roots $\alpha^{(1)}_l$ are given by (32) for $r=1$. To find $\psi_{i,2}$, two Gaussian eliminations are necessary. $\psi_{i,6}$ is also found with two Gaussian eliminations performed on the equation

$$\mathbf{E}^{(1)}\psi_{i,6}=\mathbf{g}^{(1)}_{i,6}+\mathbf{I}\psi_{i,4}. \quad (37)$$

With $\psi_{i,2}$, $\psi_{i,4}$ and $\psi_{i,6}$ known, the "eliminated" equations (those with odd indices) can be solved. Each of these is a simple matrix equation in $\mathbf{E}$ itself, therefore requiring only one Gaussian elimination per equation. For our $N=7$ example, therefore, we require 12 Gaussian eliminations to solve the system. For a system with $N+1=2^k$, it is easy to show that the number of Gaussian eliminations required is $K(N+1)/2$. This is the most important factor in determining the computational effort necessary to solve an elliptic equation by cyclic reduction.

The requirement that $N+1$ be a power of 2 may be a severe restriction on the applicability of the method. It is possible, however, to obtain linear combinations of the rows of (12) in a more general fashion than is described in Section 3, thereby making it possible to reduce the system of equations with more degrees of freedom for $N$. Sweet (1973b) describes such a generalized cyclic reduction algorithm in which $N=2^l3^m5^n-1$; $l$, $m$ and $n$ being arbitrary integer constants, and more recently, Sweet (personal communication) has developed an algorithm which removes all restrictions on $N$.

## 4. The Buneman variant

In the previous example, the right-hand side terms $\mathbf{g}^{(r)}_{i,j}$ were calculated in a straightforward fashion. While mathematically correct, Buzbee *et al.* (1970) have shown that the method, as presented above, is unstable. Test examples (Faulkner, unpublished) in-

volving square domains ($M=N$), indicate that for $N=31$ the accuracy has decreased and the method is unstable for $N>31$, even when double precision is used on the IBM 360/67.

Because of this limitation on array size, Buneman (1969) developed an alternative method of calculating the right-hand sides which retains the accuracy of the method for arbitrarily large $N$. We now describe this method.

As we saw in Section 3, after each reduction cycle the right-hand side terms are of the form

$$\mathbf{g}_{i,j}^{(r+1)}=\mathbf{g}_{i,j-h}^{(r)}+\mathbf{g}_{i,j+h}^{(r)}+\mathbf{E}^{(r)}\mathbf{g}_{i,j}^{(r)}, \quad h=2^r. \tag{38}$$

Buneman introduced the relationship

$$\mathbf{g}_{i,j}^{(r)}=\mathbf{E}^{(r)}\mathbf{P}_{i,j}^{(r)}+\mathbf{q}_{i,j}^{(r)}, \tag{39}$$

where $\mathbf{P}_{i,j}^{(r)}$ and $\mathbf{q}_{i,j}^{(r)}$ are auxiliary vectors which must be computed. When (39) is introduced into (38), and (30) is used, we obtain the expression

$$\begin{aligned}[(\mathbf{E}^{(r)})^2-2\mathbf{I}]\mathbf{P}_{i,j}^{(r+1)}+\mathbf{q}_{i,j}^{(r+1)} \\ =\mathbf{E}^{(r)}[\mathbf{P}_{i,j-h}^{(r)}+\mathbf{P}_{i,j+h}^{(r)}]+[\mathbf{q}_{i,j-h}^{(r)}+\mathbf{q}_{i,j+h}^{(r)}] \\ +(\mathbf{E}^{(r)})^2\mathbf{P}_{i,j}^{(r)}+\mathbf{E}^{(r)}\mathbf{q}_{i,j}^{(r)}.\end{aligned} \tag{40}$$

We separate those terms which are multiplied by a matrix operator from those which are not, and after some manipulation obtain the two expressions

$$\mathbf{P}_{i,j}^{(r+1)}=\mathbf{P}_{i,j}^{(r)}+(\mathbf{E}^{(r)})^{-1}[\mathbf{P}_{i,j-h}^{(r)}+\mathbf{P}_{i,j+h}^{(r)}+\mathbf{q}_{i,j}^{(r)}], \tag{41}$$

$$\mathbf{q}_{i,j}^{(r+1)}=\mathbf{q}_{i,j-h}^{(r)}+\mathbf{q}_{i,j+h}^{(r)}+2\mathbf{P}_{i,j}^{(r+1)}. \tag{42}$$

These two equations give recursion relations from which the arrays $\mathbf{P}_{i,j}^{(r)}$ and $\mathbf{q}_{i,j}^{(r)}$ can be found. First we solve the series of matrix equations

$$\begin{aligned}\mathbf{E}^{(r)}[\mathbf{P}_{i,j}^{(r+1)}-\mathbf{P}_{i,j}^{(r)}]=\mathbf{P}_{i,j-h}^{(r)}+\mathbf{P}_{i,j+h}^{(r)}+\mathbf{q}_{i,j}^{(r)}, \\ r=0,\ 1,\ \ldots,\ K,\end{aligned} \tag{43}$$

where

$$\mathbf{E}^{(0)}=\mathbf{E},\ \mathbf{P}_{i,j}^{(0)}=0, \text{ and } \mathbf{q}_{i,j}^{(0)}=\mathbf{g}_{i,j}^{(0)}=\mathbf{g}_{i,j},$$

and the $\mathbf{E}^{(r)}$ are the polynomials which factors are given by (32). Eq. (43) is solved in exactly the same way as was (27), as described in Section 3. We then have

$$\mathbf{P}_{i,j}^{(r+1)}=[\mathbf{P}_{i,j}^{(r+1)}-\mathbf{P}_{i,j}^{(r)}]+\mathbf{P}_{i,j}^{(r)}, \tag{44}$$

and the solution of (42) is easily obtained.

After the arrays $\mathbf{P}_{i,j}^{(r)}$ and $\mathbf{q}_{i,j}^{(r)}$ are found, we have the matrix equations

$$\mathbf{E}^{(r)}\psi_{i,j}=\mathbf{E}^{(r)}\mathbf{P}_{i,j}^{(r)}+\mathbf{q}_{i,j}^{(r)}, \tag{45}$$

or

$$\mathbf{E}^{(r)}[\psi_{i,j}-\mathbf{P}_{i,j}^{(r)}]=\mathbf{q}_{i,j}^{(r)}, \tag{46}$$

which is solved by the previously described method. The final solution is then

$$\psi_{i,j}=[\psi_{i,j}-\mathbf{P}_{i,j}^{(r)}]+\mathbf{P}_{i,j}^{(r)}. \tag{47}$$

It is possible to eliminate the $\mathbf{P}_{i,j}^{(r)}$ arrays between (41) and (42), leaving a recursion relation containing only the $\mathbf{q}_{i,j}^{(r)}$ arrays. This variant of the Buneman method is described by Buzbee, *et al.* (1970). It cuts the storage requirements in half but requires about twice as many additions in a computational algorithm. However, the extra computation time for these additions becomes negligible as $N$ increases.

## 5. Other boundary conditions

The preceding discussion is relevant to the solution of elliptic equations with Dirichlet boundary conditions at $j=0$ and $j=N+1$. The matrix $\mathbf{D}$ given by (13) is particular to these conditions and is readily amenable to the block cyclic reduction method of Section 3. We now examine the form of the matrix equations for other boundary conditions.

### *a. Periodic boundary conditions*

We have

$$\psi(i,\ N+1)=\psi(i,1) \tag{48}$$

and

$$\psi(i,0)=\psi(i,N). \tag{49}$$

Following the notation of Section 3, the block matrix analogous to (13) is

$$\mathbf{D}_{\text{per}}\begin{bmatrix} \mathbf{E} & -\mathbf{I} & 0 & \cdots & \cdots & \cdots & -\mathbf{I} \\ -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & \cdots & \cdots & 0 \\ \cdots & \ddots & \ddots & \ddots & \ddots & \ddots & \cdots \\ \cdots & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & \cdots \\ \cdots & \ddots & \ddots & \ddots & \ddots & \ddots & \cdots \\ 0 & \cdots & \cdots & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} \\ -\mathbf{I} & 0 & \cdots & \cdots & 0 & -\mathbf{I} & \mathbf{E} \end{bmatrix}_{N\times N}, \tag{50}$$

where $N=2^k$. Buzbee, *et al.* (1970), describes the reduction procedure for this matrix.

When (50) is reduced, the final reduction polynomial has a root $\alpha$ equal to 2.0. If the coefficient matrix $\mathbf{E}$ is appropriate for a Poisson equation with Neumann or periodic boundary conditions, then the resulting matrix equation to be solved is singular. As is well known in this case, the solution can only be determined to within an arbitrary constant. In programming this algorithm, some care is necessary to check for a singular system and specifying the arbitrary constant if necessary.

### *b. Neumann boundary conditions*

There are two ways of specifying Neumann conditions. The first method is appropriate for unstaggered grid problems such as the solution of (2) for the stream-function. We let

$$\psi(i,\ N+1)=\psi(i,\ N-1)+2\Delta y\cdot\mathbf{DR}(i), \tag{51}$$

$$\psi(i,0)=\psi(i,2)-2\Delta y\cdot\mathbf{DL}(i), \tag{52}$$

where $\mathbf{DR}[i]$ and $\mathbf{DL}(i)$ are the values of the normal derivative of $\psi$ at $j=1$ and $j=N$ respectively, and $\Delta y$ is the grid interval in the $j$ direction.

The block matrix form for this case is

$$\mathbf{D}_{\text{neu}}\begin{bmatrix} \mathbf{E} & -2\mathbf{I} & 0 & & & & \\ -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & & & \\ & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & \\ & & & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} \\ & & & & 0 & -2\mathbf{I} & \mathbf{E} \end{bmatrix}_{N\times N}, \quad (53)$$

where $N=2^k+1$. The reduction of this matrix is described by Buzbee *et al.* (1970).

As is well known, the solution of Poisson's equation with Neumann conditions on all boundaries can be determined only to within an arbitrary constant. In this case the matrix (53) is a singular matrix, and it is handled in exactly the same way as described for periodic boundary conditions.

A second form of Neumann boundary conditions is on a staggered grid [Eq. (4) is frequently of this type]. In this case we have

$$\mathbf{P}(i, N+1)=\mathbf{P}(i,N)+\Delta y\cdot\mathbf{DR}(i), \quad (54)$$

$$\mathbf{P}(i,0)=\mathbf{P}(i,1)-\Delta y\cdot\mathbf{DL}(i), \quad (55)$$

where $\mathbf{DR}(i)$ and $\mathbf{DL}(i)$ are the normal derivatives defined on the intervals between $j=N+1$ and $N$ and $j=0$ and 1, respectively.

The block matrix form for this case is

$$\mathbf{D}_{\text{neu}}=\begin{bmatrix} \mathbf{E}-\mathbf{I} & -\mathbf{I} & 0 & & & \\ -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 & & \\ & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} & 0 \\ & & 0 & -\mathbf{I} & \mathbf{E} & -\mathbf{I} \\ & & & 0 & -\mathbf{I} & \mathbf{E}-\mathbf{I} \end{bmatrix}_{N\times N}, \quad (56)$$

where again $N=2^k+1$. Sweet (1973a) describes the reduction of this matrix. As with the other case of

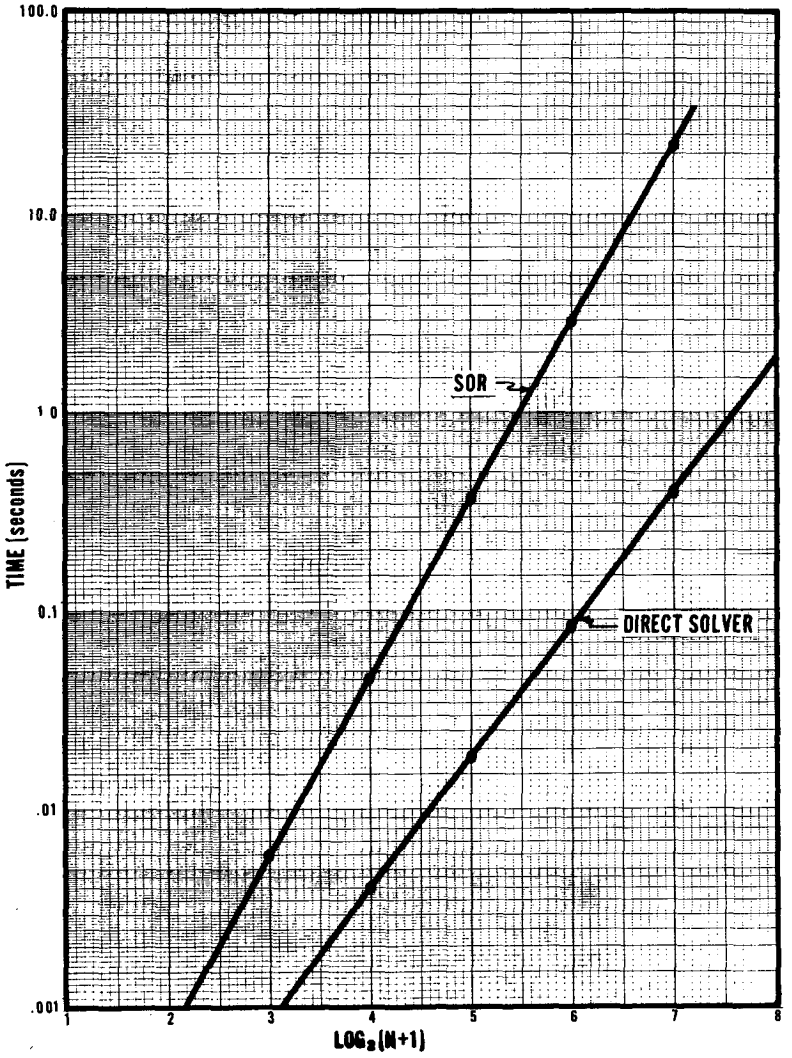


FIG. 1. Dirichlet boundary conditions.

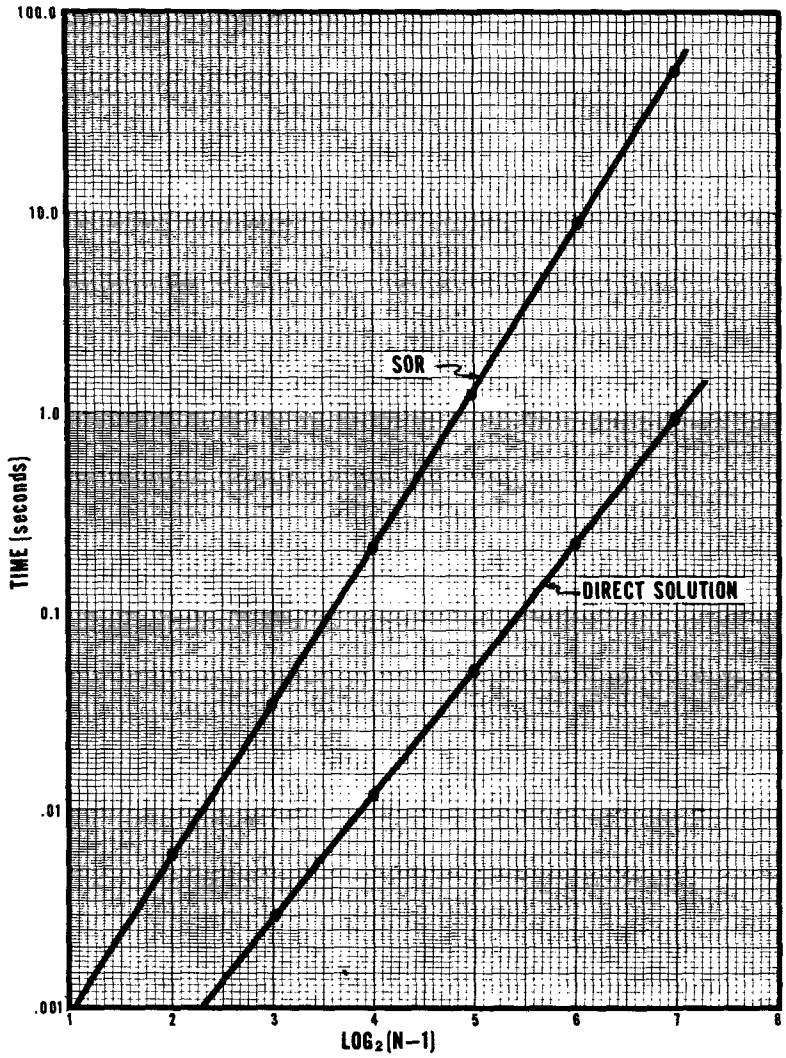


FIG. 2. Neumann boundary conditions.

Neumann conditions, the matrix **D** is singular and the solution is determined to within an arbitrary constant.

## 6. Comparison with SOR and discussion

Leslie and McAvaney (1973) have discussed the relative speeds of many different methods, both iterative and direct, for solving Helmholtz-type equations. Dorr (1970) also discusses the efficiency of iterative versus direct methods in terms of operation counts. We shall therefore not present a comparison of the direct method described in this paper with all of the possible alternatives. However, we shall compare this method with successive over-relaxation (SOR) because, in spite of the clear theoretical advantages of the direct method over SOR, potential users are still hesitant to abandon their iterative methods until they have seen a graphic demonstration of the savings possible.

Figure 1 is a graph showing relative speeds for the solution of (7). We assume a square domain, i.e., $M=N$, and let $\Delta X=\Delta Y$. In the SOR solution the over-relaxation coefficient was calculated according to

$$\omega=\frac{2}{1+\sin(\pi/N)}, \tag{57}$$

as given by Frankel (1950).

The SOR iterations were stopped when the maximum residual was less than $1.0\times10^{-5}$ [solution values of O(1)]. For $N+1=128$ this required 264 iterations, so the computational expense is not to be ignored.

Figure 2 shows the relative solution times of (7) with Neumann conditions on all boundaries using the formulation of Miyakoda (1962). There is no clear statement of what the optimum over-relaxation coefficient for the Neumann problem should be, but experience shows that somewhat larger values than for the

Dirichlet problem give the fastest convergence. Therefore, the relationship

$$\omega = 2.0/(1+\sin\pi/2N) \tag{58}$$

was used. This was chosen rather arbitrarily, but it gave faster convergence than either $N$ or $3N$ in the denominator of the sine function argument. Careful tuning by numerical experimentation would probably find more optimum values, but the convergence rates would still be much slower than for Dirichlet boundary conditions.

We again terminated the iterative solution when the maximum residual was less than $1.0\times10^{-5}$. As expected, the direct solution shows even greater advantage over SOR for Neumann boundary conditions. In addition, the direct solution results are correct to machine accuracy. Keller (1965) gives some conditions for the convergence of iterative schemes such as SOR when the matrix is singular. It was not clear how to define his iteration matrix and no comparisons were made for this case.

It should be noted that in some marching problems a good first estimate for the solution may be available, and in this case fewer iterations are needed for SOR, so it is more competitive. This does not alter the convergence criteria, however, but only the number of iterations to effect it.

In both these examples a Poisson equation was solved. However, the introduction of a Helmholtz term can change the behavior of SOR convergence considerably. For example, a Helmholtz term will usually eliminate the singular nature of the Neumann solution and therefore, may accelerate convergence. If $k_3(x)$ in (1) is positive, then the matrix equations are more diagonally dominant than for a Poisson equation and convergence is enhanced. If $k_3(x)$ in (1) is negative, however, the reverse is true, and as the absolute value of this term increases convergence of SOR is slowed, and if it becomes large enough the elliptic nature of the equation is lost and the method diverges. The cyclic reduction method, on the other hand, is quite insensitive to the degree of diagonal dominance of the matrix equations. In Section 5 the cases of singular systems arising from Poisson equations with Neumann or periodic boundary conditions were described. If such a singular system is anticipated, it can be solved with special treatment. It is possible, however, for a Helmholtz equation with Dirichlet boundary conditions to be singular in special cases of $k_3(x)$ in (1) being less than zero. If such a singularity is not anticipated erroneous results will occur. However, the method is very stable even for very nearly singular systems, whether the matrix is positive definite or not. In numerical experiments, it was necessary to use double precision arithmetic on an IBM 360/67 to locate eigenvalues which would cause any noticeable effect on the answers, and even then residuals remained quite small. Of course, this does not eliminate the need for caution when such singularities can occur.

All computations were performed on a Control Data 7600 at the Lawrence Berkeley Laboratory, Berkeley, Calif., and an IBM 360/67 at the W. R. Church Computer Center, Naval Postgraduate School, Monterey, California.

## 7. Summary

The algorithms described in this paper have been programmed in FORTRAN by Roland Sweet of the National Center for Atmospheric Research (NCAR). The subroutines are designated as POISXX, where the last two letters vary depending on the boundary conditions. For example POISDN solves a Poisson equation with Dirichlet conditions in the $j$ direction and Neumann conditions in $i$. These routines must have the number of intervals in the $j$ direction equal to a power of 2, but the number of intervals in the $i$ direction is arbitrary. They are written for constant coefficients in the Gaussian elimination procedure.

Sweet has also written a subroutine called POIS which solves a general elliptic equation for any combination of boundary conditions. For this routine the number of intervals in one direction may be $N=2^p3^q5^r$. The coefficients in the Gaussian elimination scheme may vary in this subroutine. Swarztrauber and Sweet (1973) have used POIS to write subroutines which solve the discrete Poisson equation on a disk and a sphere. All of these routines and several others are described in NCAR Technical Note IN/IA-109 and are available from the NCAR facility.

Rosmond and Faulkner of the Navy's Environmental Prediction Research Facility (EPRF) have modified the POISXX routines to allow variable coefficients in the Gaussian eliminations. These routines, as well as those of Sweet and Swarztrauber, have been used extensively by EPRF and Fleet Numerical Weather Central (FNWC) in their operational weather prediction program. Increased accuracy and considerable computer time savings have resulted from their use.

*Acknowledgments.* The authors wish to acknowledge the assistance given them by Dr. Roland Sweet of the University of Colorado, Denver, and the National Center for Atmospheric Research. His FORTRAN subroutines formed the basis upon which our work was based. He also contributed much early guidance in our study of the method and subsequent applications.

REFERENCES

Abramowitz, M., and S. A. Stegun, 1965: *Handbook of Mathematical Functions*. Dover Publ., 773–802.

Buneman, O., 1969: A compact non-iterative Poisson solver. SUIPR Rept. No. 294, Institute for Plasma Research, Stanford University, Palo Alto, Calif., 9 pp.

Buzbee, B. L., and F. W. Dorr, 1974: The direct solution of the biharmonic equation on rectangular regions and the Poisson

equation on irregular regions. *SIAM J. Num. Anal.*, **11**, 753–763.

——, G. H. Golub and C. W. Nielson, 1970: On direct methods for solving Poissons' Equations. *SIAM J. Num. Anal.*, **7**, 667–656.

——, F. W. Dorr, J. A. George and G. H. Golub, 1971: The direct solution of the discrete Poisson equation on irregular regions. *SIAM J. Num. Anal.*, **8**, 722–736.

Concus, C., and G. H. Golub, 1973: Use of fast direct methods for efficient numerical solution of nonseparable elliptic equations. *SIAM J. Num. Anal.*, **10**, 1103–1120.

Cooley, J. W., and J. W. Tukey, 1965: An algorithm for the machine calculation of complex Fourier series. *Math. Comput.*, **19**, 297–301.

Dingle, A. N., and C. Young, 1965: *Computer Applications in the Atmospheric Sciences*. University Microfilms, Ann Arbor, Mich., 256 pp.

Dorr, F. W., 1970: The direct solution of the discrete Poisson equation on a rectangle. *SIAM Rev.*, **12**, 248–263.

Faulkner, F. D., and T. E. Rosmond, 1975: The numerical solution of Helmholtz's equations by use of fast Poisson solvers. In preparation as NAVENVPREDRSCHFAC, Tech. Note.

Frankel, S. P., 1950: Convergence rates for iterative treatments of partial differential equations. *Math. Tables Nat. Res. Coun. Wash.*, **4**, 65–75.

Hockney, R. W., 1965: A fast direct solution of Poisson's equation using Fourier analysis. *J. Assoc. Comp. Mach.*, **12**, 95–113.

Keller, H. B., 1965: On the solution of singular and semidefinite linear systems by iteration. *SIAM J. Num. Anal.*, **2**, 281–290.

Leslie, L. M., and B. J. McAvaney, 1973: Comparative test of direct and iterative methods for solving Helmholtz type equations. *Mon. Wea. Rev.*, **101**, 235–239.

Martin, E. D., 1973: Direct numerical solution of three-dimensional equations containing elliptic operators. *Intern. J. Num. Methods Eng.*, **6**, 201–212.

Miyakoda, K., 1962: Contribution to the numerical weather prediction-computation with finite difference. *Jap. J. Geophys.*, **3**, 75–190.

Rosmond, T. E., 1975: Subroutines for solving three-dimensional elliptic equations. ENVPREDRSCHFAC Computer Programming Note No. 22. Naval Environmental Prediction Research Facility, Monterey, Calif., 39 pp.

Sasaki, Y., 1970: Some basic formalisms in numerical variational analysis. *Mon. Wea. Rev.*, **98**, 875–883.

Swarztrauber, P. N., and R. A. Sweet, 1973: The direct solution of the discrete Poisson equation on a disk. *SIAM J. Num. Anal.*, **10**, 900–907.

——, and ——, 1975: Efficient fortran subprograms for the solution of elliptic partial differential equations. NCAR Tech. Note IN/IA-109, 139 pp.

Sweet, R. A., 1973a: Direct methods for the solution of Poisson's equation on a staggered grid. *J. Comput. Phys.*, **12**, 422–428.

——, 1973b: A generalized cyclic reduction algorithm. *SIAM J. Num. Anal.*, **10**, 506–520.

Williams, G. P., 1969: Numerical integration of the three-dimensional Navier-Stokes equations for incompressible flow. *J. Fluid Mech.*, **37**, 727–750.