JOURNAL OF COMPUTATIONAL PHYSICS 35, 408–425 (1980)

Solution of Poisson Equations on a Nonuniform Grid

L. FARNELL*

Geophysical Fluid Dynamics Laboratory, Meteorological Office, Bracknell, Berkshire, England

Received October 31, 1978; revised April 6, 1979

The solution of a Poisson equation expressed in finite-difference form on a nonuniform multidimensional mesh of gridpoints in an orthogonal coordinate system is discussed. The discussion is specifically directed toward problems which require the solution of many such equations with the gridpoint distribution and boundary conditions fixed, which implies that in comparing different techniques the task of generating required constants may be neglected. Extension of the matrix decomposition technique to cover the case of smoothly varying grid-intervals is considered when zero-gradient, zero-value, or (under certain conditions) periodic boundary conditions are incorporated. This method is compared in particular with the widely used Alternating Direction Implicit procedure, and it is concluded that for sufficiently small meshes, not exceeding about 40 points in any direction, it is both faster and more precise than ADI.

1. INTRODUCTION

The point has recently been made [10] that with the increasing use of three-dimensional numerical models of fluid flows there is a need for fast algorithms for solving three-dimensional Poisson equations expressed in finite-difference form; indeed this equation and others of the elliptic type are frequently met in other fields also, such as astrophysics and plasma physics. Several different methods have been developed, including cyclic reduction [8, 16], alternating direction implicit (ADI) [5], as optimised by Wachspress [9], multiple grid techniques [14, 15], matrix decomposition [3, 16], and generalized marching algorithms [17].

In the type of problem under consideration [1, 6, 12, 13] the basic equations (i.e., the Navier–Stokes equations in fluid mechanics) are solved by integration over time, and this necessitates solving at least one elliptic equation in each timestep. We therefore assume that we require to solve many such equations with the distribution of gridpoints and the boundary conditions being fixed. This enables us to discount the task of generating items which are functions of these alone when we compare methods. It is also implied that, apart from the first few timesteps, a quite accurate initial guess is available for iterative techniques, viz., an extrapolation from the solution for previous timesteps.

* Present address: Research School of Chemistry, Australian National University, P.O. Box 4, Canberra, A.C.T. 2600, Australia.

0021-9991/80/060408-18$02.00/0

In many fluid problems, computer storage can be reduced by use of nonuniformly spaced grids [4, 7]. This allows gridpoints to be concentrated in regions such as boundary layers where large gradients of the variables are expected. If this is not done, the range of problems which may be investigated is restricted; e.g., Williams [12] examined a physical system which was expected to possess relatively thick boundary layers. The methods listed above have all been developed basically in terms of a grid which is uniformly spaced, and extension to irregular grids is not trivial. Quon [6] has noted that adapting cyclic reduction is relatively complex, and accordingly uses ADI instead. Furthermore, Wilhelmson and Erickson [10] have concluded that in three dimensions the use of cyclic reduction alone (for it is possible to apply different techniques in different directions if the equation is separable) is never optimal, so we will therefore leave it out of consideration. Multiple grid techniques are admitted to be complex to implement [15], and with a good starting guess would seem to have no advantage over ADI. In any case the type of nonuniform grid considered by Brandt [14], where a square mesh is subdivided into smaller squares in certain regions, is different in character from the one we wish to use, where the grid-interval is varied smoothly.

Matrix decomposition makes use of discrete fast Fourier transforms [3]. The problem with this is that trigonometric functions are not orthogonal on a nonuniform grid (in the sense that the scalar product of two different vectors is not zero; see below for further discussion) and nonuniformity also precludes the use of "fast" transforms [2]. The latter point is not crucial for small numbers of gridpoints because the lesser complexity of "slow" transform algorithms compensates for the greater number of arithmetic operations relative to "fast" transforms.

We followed Quon and others in selecting ADI as the easiest method to implement as part of a scheme for numerical determination of fluid flows [1], but some disadvantages became apparent. (a) A minimum of eight iterations is necessary to provide a pressure field sufficiently accurate to maintain stability in the overall integration scheme, and under some circumstances even more. This occurs despite correcting for errors in the same manner as that of Williams [12]. (b) For any acceptable number of iterations, ADI leaves a large error compared to noniterative methods, and even if the catastrophic effects noted under (a) are avoided, this can lead to errors in the conservation of integral properties such as kinetic energy and angular momentum. (c) Since an iteration parameter, which may be 1000 times larger than the solution we are looking for, is added in and later subtracted out, rather more precision is required locally than is necessary overall.

We therefore decided to implement the matrix decomposition method, and the aim of this paper is to discuss the problems that arose. The method here described has the following properties. (a) It is considerably faster than ADI for the relatively small number of gridpoints made possible by use of a nonuniform mesh. This is enhanced by the removal of the requirement of an initial guess. (b) The solution is direct and consequently obtained to within computer rounding error. (c) The method may be readily adapted to zero-gradient, zero-value, and, under certain circumstances discussed below, periodic boundary conditions.

A final point is that the solution of the Poisson equation now takes less than a quarter of the total time for each timestep. In consequence little seems to be gained by using more complex techniques such as marching algorithms, and these have not therefore been investigated.

2. Outline of Method

The essentials of the procedure will be illustrated by the solution of a two-dimensional problem in Cartesian coordinates. For definiteness, the nomenclature appropriate to the fluid-mechanical problem will be adopted. Given a source-function S, a pressure field Π is to be obtained, where both fields depend on the two spatial coordinates x, y. The Poisson equation is then

$$\frac{\partial^2 \Pi}{\partial x^2} + \frac{\partial^2 \Pi}{\partial y^2} = S \tag{1}$$

and zero-gradient boundary conditions may be employed [12].

$$\begin{aligned} \frac{\partial \Pi}{\partial x} &= 0 \qquad \text{at} \quad x = 0, x = X, \\ \frac{\partial \Pi}{\partial y} &= 0 \qquad \text{at} \quad y = 0, y = Y, \end{aligned} \tag{2}$$

where X and Y are the dimensions of the rectangular region of interest.

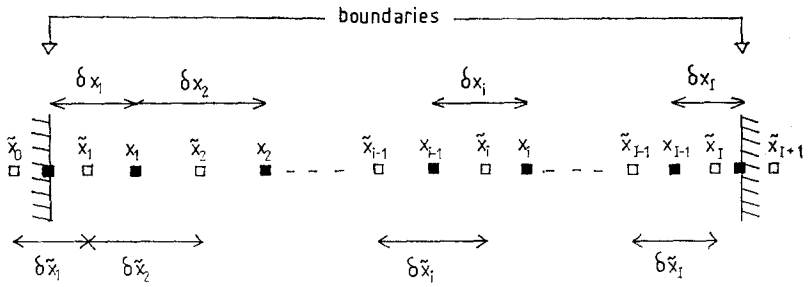

FIG. 1. Distribution of gridpoints and definitions of grid-intervals. ■, Main grid, x_i, $i = 0$ to I; □, auxiliary grid, $\tilde{x}_i$, $i = 0$ to $I + 1$.

Gridpoints for the finite-difference formulation are located as illustrated in Fig. 1 (for the x-coordinate; the y-coordinate is treated analogously). Note that two sets of points (main and auxiliary) are interleaved, and the grid-intervals lie to the left of the points with the same index, i.e., δ is the backward difference operator. (See, e.g., [4, 6, 7, 12, 18] for the advantage of this choice of grid.) Π and S are defined on the auxiliary gridpoints $\tilde{x}_i$, $\tilde{y}_k$, which are indexed 1 to I and 1 to K, respectively, within

the boundaries only. (A more usual convention includes the dummy points, here denoted $\tilde{x}_0$, $\tilde{x}_{I+1}$, $\tilde{y}_0$, and $\tilde{y}_{K+1}$ within this range of indexing, but this would be inconvenient in the following discussion.)

Using the simple five-point representation of ∇^2, the finite-difference analogs of (1) and (2) are as follows.

$$\frac{1}{\delta x_i \delta \tilde{x}_{i+1}}(\Pi_{i+1k} - \Pi_{ik}) + \frac{1}{\delta x_i \delta \tilde{x}_i}(\Pi_{i-1k} - \Pi_{ik}) \cdots$$
$$+ \frac{1}{\delta y_k \delta \tilde{y}_{k+1}}(\Pi_{ik+1} - \Pi_{ik}) + \frac{1}{\delta y_k \delta \tilde{y}_k}(\Pi_{ik-1} - \Pi_{ik}) = S_{ik}\,, \tag{3}$$

$$\begin{aligned} \Pi_{I+1k} &= \Pi_{Ik}\,; \qquad \Pi_{0k} = \Pi_{1k}\,; \\ \Pi_{iK+1} &= \Pi_{ik}\,; \qquad \Pi_{i0} = \Pi_{i1}\,. \end{aligned} \tag{4}$$

In essence the use of a Fourier transform removes derivatives with respect to the coordinate over which the transform is performed. In this example, Eq. (1) is transformed using the following functions which satisfy conditions (2).

$$H_m = \cos\left(\frac{2\pi m x}{X}\right), \qquad m = 0 \text{ to } \infty, \tag{5}$$

Expansion of Π and S as series

$$\Pi = P_m H_m\,; \qquad S = Q_m H_m\,, \tag{6}$$

where the summation convention for repeated indices is assumed, yields the following equations for the expansion coefficients P_m and Q_m.

$$\frac{\partial^2 P_m H_m}{\partial x^2} + \frac{\partial^2 P_m H_m}{\partial y^2} = Q_m H_m\,, \tag{7}$$

$$\therefore P_m \frac{\partial^2 H_m}{\partial x^2} + H_m \frac{\partial^2 P_m}{\partial y^2} = Q_m H_m\,, \tag{8}$$

$$\therefore \left(-m^2 P_m + \frac{\partial^2 P_m}{\partial y^2}\right) H_m = Q_m H_m\,. \tag{9}$$

Multiplication of (9) by H_n followed by integration over x gives finally

$$\frac{\partial^2 P_n}{\partial y^2} - n^2 P_n = Q_n\,, \qquad n = 0 \text{ to } \infty. \tag{10}$$

Thus the dimensionality has been reduced by one at the expense of multiplying the number of equations. While a second transform could be done to solve Eq. (10), it is always faster to solve the one-dimensional finite-difference equations by a form of Gaussian elimination adapted to take advantage of the tridiagonality of the matrices

[3, 12] so that this stage is standard. The outstanding problem is to find a finite-difference analog of (5)–(10) when the gridpoints are nonuniformly distributed.

The essential property of the functions H_m is that they are eigenfunctions of the $\partial^2/\partial x^2$ operator. On a uniform mesh, the vectors obtained by using Eq. (5) with the relevant values of x are also eigenfunctions of the analogous finite-difference operator. However, no analytic formula is available when the gridpoints are nonuniformly distributed, and the vectors must be determined numerically. This is discussed in Section 3. Here we will assume that the values $H_{m;i}$ (where we take subscripts before the semicolon to refer to wave-space, and those after to refer to gridpoint space) are known, as also are the corresponding eigenvalues λ_m, which correspond to the coefficients $(-m^2)$ in Eq. (9).

As previously noted the scalar products $H_{m;i}H_{n;i}$ are not zero for $m \neq n$, which is a necessary requirement for the process (9)–(10). However, it is always possible to find a reciprocal set of vectors $h_{m;i}$ for which this property is true, viz.,

$$H_{m;i}h_{n;i} = \delta_{mn} . \tag{11}$$

where δ_{mn} is the Kronecker delta. This is also dealt with in Section 3.

We can now list the steps required to solve the problem given by Eqs. (3) and (4).

(a) The source function is expanded in terms of the vectors, analogous to (6) above.

$$S_{ik} = Q_{m;k}H_{m;i} , \tag{12}$$

whence

$$h_{n;i}S_{ik} = Q_{m,k}h_{n;i}H_{m;i} = Q_{n;k} \tag{13}$$

so that $Q_{n;k}$ are determined.

(b) The finite-difference analogs of (8) and (9) above are as follows.

$$Q_{m;k}H_{m;i} = P_{m;k}\left[\frac{H_{m;i+1} - H_{m;i}}{\delta x_i \delta \tilde{x}_{i+1}} + \frac{H_{m;i-1} - H_{m;i}}{\delta x_i \delta \tilde{x}_i}\right] \cdots$$

$$+ H_{m;i}\left[\frac{P_{m;k+1} - P_{m,k}}{\delta y_k \delta \tilde{y}_{k+1}} + \frac{P_{m;k-1} - P_{m;k}}{\delta y_k \delta \tilde{y}_k}\right] \tag{14}$$

$$= H_{m;i}\left[\lambda_m P_{m;k} + \frac{P_{m;k+1} - P_{m;k}}{\delta y_k \delta \tilde{y}_{k+1}} + \frac{P_{m;k-1} - P_{m;k}}{\delta y_k \delta \tilde{y}_k}\right] . \tag{15}$$

Taking the scalar product of both sides of (15) with each reciprocal vector in turn, and making use of (11), yields

$$Q_{n;k} = \lambda_n P_{n;k} + \frac{P_{n;k+1} - P_{n;k}}{\delta y_k \delta \tilde{y}_{k+1}} + \frac{P_{n;k-1} - P_{n;k}}{\delta y_k \delta \tilde{y}_k} , \qquad n = 1 \text{ to } I \tag{16}$$

and this set of equations may be solved by standard techniques as previously noted to

determine all $P_{n;k}$. Note that the case $\lambda_n = 0$, which always arises with the given boundary conditions, requires special treatment.

(c) The pressure field is now obtainable by summation over eigenvectors.

$$\Pi_{ik} = P_{m;k} H_{m;i} . \tag{17}$$

3. Evaluation of Numerical Vectors

Since Poisson's equation is separable each direction of a system of orthogonal coordinates may be treated independently. For the modification of Eq. (14) to yield (15) we require that the vectors satisfy the following equations.

$$\frac{1}{\delta x_i \delta \tilde{x}_{i+1}} [H_{m;i+1} - H_{m;i}] + \frac{1}{\delta x_i \delta \tilde{x}_i} [H_{m;i-1} - H_{m;i}] = \lambda_m H_{m;i} . \tag{18}$$

These may be made somewhat more general and compact by defining "leftward" and "rightward" coefficients so that (18) may be rewritten as

$$R_i [H_{m;i+1} - H_{m;i}] + L_i [H_{m;i-1} - H_{m;i}] = \lambda_m H_{m;i} , \tag{19}$$

which is recognizable as a matrix eigenvalue problem

$$\mathbf{A} \mathscr{H}_m = \lambda_m \mathscr{H}_m . \tag{20}$$

The matrix **A** is of order $I \times I$, and tridiagonal with nonzero elements as follows.

$$\begin{aligned} A_{ii} &= -L_i - R_i , & i &= 2 \text{ to } I - 1, \\ A_{ii+1} &= R_i , & i &= 1 \text{ to } I - 1, \\ A_{i+1i} &= L_{i+1} , & i &= 1 \text{ to } I - 1. \end{aligned} \tag{21}$$

The corner elements A_{11} and A_{II} are determined by the boundary conditions. Taking zero-gradient as required by the problem of Section 2, we may substitute

$$H_0 = H_1 ; \qquad H_{I+1} = H_I \tag{22}$$

into the first and last of Eqs. (19), respectively, to yield

$$A_{11} = -R_i ; \qquad A_{II} = -L_I . \tag{23}$$

There are standard methods for eigenvector problems [11] which may be applied to the solution of Eq. (20). Observe that **A** is quasi-symmetric, i.e., all elements are real and all products $A_{ii+1} A_{i+1i}$ are positive. This means that a similarity transform exists which converts **A** into a real symmetric tridiagonal matrix **B** which has the same

diagonal elements as **A** and superdiagonal elements $B_{ii+1} = (A_{ii+1}A_{i+1i})^{1/2}$. The procedure for determining the eigenvalues of **B** (which are also the eigenvalues of **A**) will be briefly summarized, since there are some points of interest that are peculiar to the present problem.

First a bound on the absolute value of all λ_m is provided by finding the sum of the absolute values of the elements in each row, and taking the largest. In practice it is found that no eigenvalue is greater than zero (compare the continuous problem $\lambda_m = -m^2$) so that the bound obtained is only required to give a lower limit. Furthermore the values become closer together as zero is approached, so that it is advantageous to determine the eigenvalues in ascending order, using each as the lower bound on the next.

Whatever the procedure adopted, the interval defined by the two bounds is bisected several times until it is reduced to within a specified tolerance. This is possible because a criterion exists (the Sturm sequence property) as to whether the mean of the old limits is an upper or lower bound on the selected eigenvalue. If this mean is Λ, the criterion requires the evaluation of the determinants of the principal leading minors of the matrix $\mathbf{B} - \Lambda\mathbf{E}$ (where **E** is the unit matrix), which will be denoted Δ_i, where $i \times i$ is the order of the minor. This may be done recursively, using the following formulas.

$$\begin{aligned} \Delta_0 &= 1, \\ \Delta_1 &= B_{11} - \Lambda, \\ \Delta_i &= (B_{ii} - \Lambda)\,\Delta_{i-1} - B_{ii-1}^2\Delta_{i-2}, \qquad i = 2 \text{ to } I. \end{aligned} \tag{24}$$

It may be noted that Δ_{i+1}/Δ_i is of order $\overline{\delta x}^{-2}$, where $\overline{\delta x}$ denotes a typical gridlength (say X/I). For more than 10–15 points this leads to problems due to the generation of numbers too large or too small for computer representation, unless $\overline{\delta x}$ is of order unity. Thus it may be necessary to multiply all coefficients L_i, R_i by $\overline{\delta x}^2$ and correct the eigenvalues afterward by division by the same factor.

Finally it is necessary to count the number of times that Δ_i and Δ_{i+1} are of the same sign (including the "dummy" determinant Δ_0), considering $\Delta_i = 0$ to have the opposite sign to Δ_{i-1} and noting that successive zeros are impossible. The number thus obtained is identical to the number of eigenvalues greater than Λ.

Once the eigenvalues of **A** are known, the eigenvectors can be found by the method of inverse iteration, the advantages of which are discussed in [11]. We have

$$(\mathbf{A} - \lambda_m\mathbf{E})\,\mathscr{H}_m(l+1) = \mathscr{H}_m(l), \tag{25}$$

where l is the number of the iteration. The speed of convergence depends on the precision of λ_m. We have found that with λ_m determined to a tolerance of 1 in 10^8, five iterations are certainly sufficient if the initial guess prescribed by Wilkinson [11] is employed.

The vectors may of course be multiplied by an arbitrary constant. A suitable nor-

malization is found if the following properties of the vectors are considered. Let us define a diagonal matrix as follows.

$$D_{ii} = \delta x_i \,. \tag{26}$$

Then by inspection the matrix defined by

$$\mathbf{\Sigma} = \mathbf{DA} \tag{27}$$

is symmetric.

Now, instead of taking the scalar products of two vectors, weight each point by the appropriate grid interval, so that in effect an integration is performed. This may be written in matrix form.

$$H_{m;i}H_{n;i}\delta x_i \equiv \mathscr{H}_m{}^{\mathrm{T}}\mathbf{D}\mathscr{H}_n \,, \tag{28}$$

where the superscript T indicates transposition. Combining (20) and (27) we have

$$\mathbf{\Sigma}\mathscr{H}_m = \lambda_m \mathbf{D}_m \mathscr{H}_m \,, \tag{29}$$

which further gives

$$\mathscr{H}_n{}^{\mathrm{T}}\mathbf{\Sigma}\mathscr{H}_m = \lambda_m \mathscr{H}_n{}^{\mathrm{T}}\mathbf{D}\mathscr{H}_m \,. \tag{30}$$

Similarly we obtain

$$\mathscr{H}_m{}^{\mathrm{T}}\mathbf{\Sigma}\mathscr{H}_n = \lambda_n \mathscr{H}_m{}^{\mathrm{T}}\mathbf{D}\mathscr{H}_n \,. \tag{31}$$

Taking the transpose of both sides of (31) gives

$$\mathscr{H}_n{}^{\mathrm{T}}\mathbf{\Sigma}^{\mathrm{T}}\mathscr{H}_m = \lambda_n \mathscr{H}_n{}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathscr{H}_m \,, \tag{32}$$

which, since both $\mathbf{\Sigma}$ and $\mathbf{D}$ are symmetric, may be combined with (30) to give

$$(\lambda_m - \lambda_n)\, \mathscr{H}_n{}^{\mathrm{T}}\mathbf{D}\mathscr{H}_m = 0. \tag{33}$$

This means that if $m \neq n$, the quantities defined in (28) are zero. (Note that no degeneracies occur with the boundary conditions selected.)

Let us combine all the vectors into a matrix $\mathbf{H}$, of which each column is a vector. Then the product $\mathbf{H}^{\mathrm{T}}\mathbf{DH}$ is diagonal with elements defined by

$$G_{mm} = \mathscr{H}_m{}^{\mathrm{T}}\mathbf{D}\mathscr{H}_m \,. \tag{34}$$

It is desirable that the vectors are normalized such that all G_{mm} are unity, so that we can write

$$\mathbf{H}^{\mathrm{T}}\mathbf{DH} = \mathbf{E}. \tag{35}$$

If the set of reciprocal vectors used in Section 2 forms the rows of a matrix $\mathbf{h}$, then by definition we have

$$\mathbf{hH} = \mathbf{E}. \tag{36}$$

Comparison with Eq. (35) thus gives the following equation for the reciprocal vectors.

$$\mathbf{h} = \mathbf{H}^{\mathrm{T}}\mathbf{D}. \tag{37}$$

Finally, it is operationally convenient to use the matrix formulation and rewrite the transformations (13) and (17) as follows.

$$\mathbf{Q} = \mathbf{hS}; \qquad \mathbf{\Pi} = \mathbf{HP}. \tag{38}$$

Each column of the matrices $\mathbf{Q}, \mathbf{S}, \mathbf{\Pi}, \mathbf{P}$ is associated with a given k value, and the rows indicate i or m as appropriate.

4. Treatment of Symmetric Meshes

If the mesh has been chosen such that the distribution shown in Fig. 1 has a line of symmetry midway between the boundaries (i.e., the right half is a reflection of the left) then some modifications to the scheme outlined above are necessary if the boundary conditions are also symmetric. For definiteness, we will assume that the number of points I is even so that the mirror line falls between points; the alternative case with the mirror line passing through the central gridpoint is only slightly more complicated. It is found that the most negative eigenvalues fall into pairs which are pathologically close. (This is not true degeneracy such as is found under periodic boundary conditions in the continuous or uniform-mesh problem, when both sine and cosine functions with the same eigenvalue are required; it occurs even when the boundary conditions only allow vectors of either sine or cosine type. Rather it is associated with the amplitude of the relevant vectors becoming very small where the grid intervals become largest, and this will normally be in the region of the mirror line in fluid-mechanical problems because gridpoints are concentrated near the boundaries. This means that the effect of the different boundary conditions (39), now about to be discussed, is hardly felt.)

When eigenvalues are very close, there are difficulties in separating the corresponding vectors [11]. However, the problem may be circumvented by realizing that the two vectors have different symmetry properties. All vectors must be either symmetric or antisymmetric about the mirror line, so that

$$H_{m;I/2} = \pm H_{m;I/2+1}\,, \tag{39}$$

where the sign depends on m: the vectors are split into two sets which we will call

"even" (positive sign) and "odd" (negative sign). Each set has $I/2$ members, which are all orthogonal to members of the other set. This means that each set may be treated independently, solving two equations of type (20) in which the matrices are only of order $I/2 \times I/2$. These matrices are both constructed from the first half of the set of operators using (21), but the second of the conditions (22) is replaced by the new conditions (39), one in each matrix. Only half of the components of the required vectors are obtained, and the other half is known to be either symmetric (even) or antisymmetric (odd). This means that the storage required for the vectors is reduced from I^2 elements to $I^2/2$ elements. The same applies to the storage of the reciprocal vectors; it is sufficient to find the inverse matrices to the two $I/2 \times I/2$ matrices containing the half-vectors, and then multiplying the reciprocal vectors by 0.5 to allow for the fact that the true eigenvectors have twice as many elements.

There is a further computational advantage in using symmetric meshes. The number of operations in transforms (38) is reduced by a factor of nearly 2. For example the matrix **S** (of order $I \times K$) is converted into a pair of matrices of order $I/2 \times K$, one of which is then used in association with the even vectors and the other with the odd vectors. The conversion algorithm is as follows.

$$\begin{aligned} {}^{E}S_{i,k} &= S_{i,k} + S_{I+1-i,k}\,, \\ {}^{O}S_{i,k} &= S_{i,k} - S_{I+1-i,k}\,. \end{aligned} \tag{40}$$

The two sets of equations (16) are solved independently. (The singular case $\lambda = 0$ is found to be the last of the even equations, if the ordering of eigenvalues suggested in Section 3 is adhered to.) After even and odd pressure fields are obtained using (38), the final solution is obtained as follows.

$$\begin{aligned} P_{i,k} &= {}^{E}P_{i,k} + {}^{O}P_{i,k}\,, \\ P_{I+1-i,k} &= {}^{E}P_{i,k} - {}^{O}P_{i,k}\,. \end{aligned} \tag{41}$$

5. Extensions for Other Boundary Conditions

This section will be concerned with the selection of the matrix **A** of Eq. (20). The determining factors are (a) the position of the boundary with respect to the gridpoints; (b) the boundary conditions; (c) whether the grid is symmetric; and if so, (d) the position of the mirror line with respect to the gridpoints. A Cartesian coordinate system will be assumed throughout this section.

In the absence of symmetry, use of periodicity as a boundary condition means that the matrix can be generated entirely using an appropriate modification of formulas (21), but is cyclic rather than purely tridiagonal. Therefore methods of solution other than those discussed in Section 3 are required, and this problem has not been investigated. This leaves eight cases, according as the boundary condition is zero-gradient or zero-value, is applied at the left or the right, and whether the boundary is

TABLE I

Selection of Basic Boundary Conditions on an Asymmetric Mesh

Position of boundary	Boundary conditions	Formulation of boundary conditions		Formulas for corner elements	
		Left	Right	Upper left	Lower right
Between points	Zero-value	$H_0 + H_1 = 0$	$H_I + H_{I-1} = 0$	$-R_1 - 2L_1$	$-L_I - 2R_I$
	Zero-gradient	$H_0 = H_1$	$H_I = H_{I-1}$	$-R_1$	$-L_I$
Through points	Zero-value	$H_0 = 0$	$H_I = 0$	$-R_1 - L_1$	$-L_I - R_I$
	Zero-gradient	$H_1 = H_{-1}$	$H_I = H_{I+2}$	$-R_0 - L_0$[a]	$-R_{I+1} - L_{I+1}$[b]

[a] Modify A_{12} to $R_0 + L_0$, instead of using (21).
[b] Modify A_{I+2I+1} to $R_{I+1} + L_{I+1}$, instead of using (21).

positioned as shown in Fig. 1 or passes instead through a gridpoint, which will be taken as $i = 0$ or $i = I + 1$ as appropriate. This choice retains the indexing 1 to I for elements of **A** when zero-value boundary points are required, but under zero-gradient conditions the matrix is of order $(I + 2) \times (I + 2)$, with the elements corresponding to all gridpoints $i = 0$ to $i = I + 1$; leftward and rightward coefficients must therefore be defined in this range, which necessitates the use of dummy points outside the

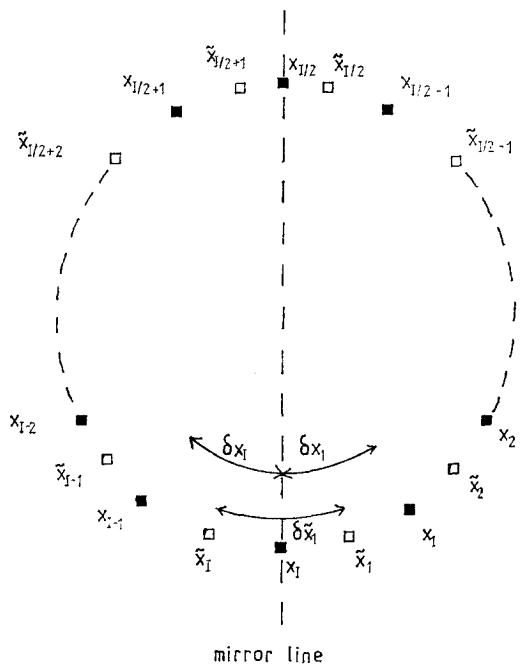

FIG. 2. Schematic representation of a symmetric periodic mesh.

boundaries indexed $i = -1$ and $i = I + 2$. The formulas for the corner elements of **A** are listed in Table I. (This includes formulas (22) and (23).)

If the mesh and the boundary conditions are symmetric, than the left boundary condition is retained, but two pseudo-boundary conditions are chosen on the right as discussed in Section 4. These will be one zero-gradient and one zero-value, and may be selected from Table I according to the position of the mirror line, which acts as a pseudo-boundary.

A further case to be considered is when periodicity is a boundary condition and the gridpoints are symmetrically distributed as illustrated in Fig. 2. The vectors must be symmetric or antisymmetric with respect to the mirror line, so that half-vectors may be determined as discussed in Section 4 by use of the following pseudo-boundary conditions.

$$\begin{aligned} &\text{Even} \quad H_1 = H_I \,(\equiv H_0) \quad \text{and} \quad H_{I/2} = H_{I/2+1}\,, \\ &\text{Odd} \quad -H_1 = H_I \,(\equiv H_0) \quad \text{and} \quad H_{I/2} = -H_{I/2+1}\,. \end{aligned} \tag{42}$$

This means that the original cyclic matrix is reduced to two tridiagonal matrices, thus avoiding the problems mentioned previously. Obviously this can be done for gridpoint distributions other than that of Fig. 2 if this is necessary.

A final point in connection with boundary conditions is that a zero eigenvalue is only found with zero-gradient on both boundaries. It arises because the columns of **A** are not linearly independent in these circumstances.

6. Extensions to Other Problems

(a) The methods outlined above can be readily extended to three or more dimensions since each is treated independently. In particular it is possible to combine them with standard fast Fourier transforms if the mesh is uniform in any direction. No problems arise with respect to computer storage because the source function is continually overwritten until the pressure field is obtained. Only one-dimensional work arrays are required, whose size is determined by the maximal number of gridpoints in any direction. A possible difficulty considered in [10] is the optimization of input–output operations where it is not possible to hold the entire source function in main core. This is not considered here because in the applications known to the author the problem has not arisen.

(b) There is no fundamental difficulty in using orthogonal coordinate systems other than Cartesian, for example cylindrical and spherical polars. However, incorporation of curvature may preclude the use of symmetry even though the gridpoints are correctly distributed: the important point is that the leftward and rightward coefficients should be symmetric. Furthermore elimination of derivatives with respect to angles will leave terms which contain both radius and angular eigenvalues. This would mean that a distinct set of radial eigenvectors would be required for each

angular eigenfunction, which is evidently a poor use of computer storage. For this reason it is in general desirable to leave radius as the last dimension to be treated, and solve in this direction by Gaussian elimination in which the variation in angular eigenvalue creates no serious degradation of computer use. Finally it will be necessary to check that the matrix $\mathbf{\Sigma}$ of Eq. (27) is symmetric, else Eq. (37) cannot be used for determination of the reciprocal vectors; the standard Gauss–Jordan method can be used instead.

(c) More general elliptic equations can be treated by rewriting Eq. (19), after defining appropriate "central" coefficients C_i, as follows.

$$R_i H_{m;i-1} + C_i H_{m;i} + L_i H_{m;i-1} = \lambda_m H_{m;i} \tag{43}$$

Wherever this is possible the matrix $\mathbf{A}$ can be constructed by obvious modifications to (21) and Table I. Also, higher-order finite-difference schemes than the simple three-point ones discussed here can be employed, but this would require more general methods of eigenvector determination since $\mathbf{A}$ would no longer be tridiagonal.

The following three-dimensional problems have actually been used operationally.

EXAMPLE 1 [1]. A Poisson equation in cylindrical polar coordinates

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \Pi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 \Pi}{\partial \theta^2} + \frac{\partial^2 \Pi}{\partial z^2} = S \tag{44}$$

is solved with the boundary conditions being zero normal gradients (r and z) and azimuthal periodicity (θ). The radial and vertical meshes are nonuniform but symmetric, the azimuthal mesh is regular. The finite-difference equations to be solved are thus

$$\frac{r_i}{\tilde{r}_i \delta r_i \delta \tilde{r}_{i+1}}(\Pi_{i+1jk} - \Pi_{ijk}) + \frac{r_{i-1}}{\tilde{r}_i \delta r_i \delta \tilde{r}_i}(\Pi_{i-1jk} - \Pi_{ijk}) + \frac{1}{\delta z_k \delta \tilde{z}_{k+1}}(\Pi_{ijk+1} - \Pi_{ijk}) \cdots$$
$$+ \frac{1}{\delta z_k \delta \tilde{z}_k}(\Pi_{ijk-1} - \Pi_{ijk}) + \frac{1}{\tilde{r}^2 \delta \theta^2}(\Pi_{ij+1k} + \Pi_{ij-1k} - 2\Pi_{ijk}) = S_{ijk}, \tag{45}$$

where $1 \leqslant i \leqslant I$, $1 \leqslant k \leqslant K$, $1 \leqslant j \leqslant J$.

The steps in the solution are as follows.

(a) Expansion of pressure and source function as a sum over azimuthal eigenvectors defined by

$$H_{m;j} = \exp(\mathrm{i}m\theta_j), \qquad -\frac{J}{2} < m \leqslant \frac{J}{2} \tag{46}$$

where j indexes the azimuthal gridpoints. The corresponding eigenvalues are

$$\lambda_m = \frac{2}{\delta \theta^2}[\cos(m\delta\theta) - 1] \tag{47}$$

(see [12]).

If we let

$$\Pi_{ijk} = P_{m;ik}H_{m;j} \qquad \text{and} \qquad S_{ijk} = T_{m;ik}H_{m;j} \tag{48}$$

then the only arithmetic operation to be performed is

$$T_{m;ik} = H_{-m;j}S_{ijk}/J. \tag{49}$$

Eqs. (45) now become

$$\frac{r_i}{\tilde{r}_i\delta r_i\delta\tilde{r}_{i-1}}(P_{m;i+1k} - P_{m;ik}) + \frac{r_{i-1}}{\tilde{r}_i\delta r_i\delta\tilde{r}_i}(P_{m;i-1k} - P_{m;ik}) + \frac{1}{\delta z_k\delta\tilde{z}_{k+1}}(P_{m;ik+1} - P_{m;ik})$$
$$+ \frac{1}{\delta z_k\delta\tilde{z}_k}(P_{m;ik-1} - P_{m,ik}) + \frac{\lambda_m P_{m;ik}}{\tilde{r}_i^2} = T_{m;ik}, \tag{50}$$

where m has the range given in (46).

(b) Expansion as a sum over vertical eigenvectors. These are generated numerically as discussed in Sections 3 and 4 and their components will be denoted $Z_{p;k}$, where p identifies the vector. The eigenvalues are μ_p and the set of reciprocal vectors $\zeta_{p;k}$. If the expansions are

$$P_{m;ik} = Q_{mp;i}Z_{p;k} \qquad \text{and} \qquad T_{m;ik} = U_{mp;i}Z_{p;k} \tag{51}$$

then the necessary transforms are

$$U_{mp;i} = T_{m;ik}\zeta_{p;k}. \tag{52}$$

Each of Eqs. (50) now becomes

$$\frac{r_i}{\tilde{r}_i\delta r_i\delta\tilde{r}_{i+1}}(Q_{mp;i+1} - Q_{mp;i}) + \frac{r_{i-1}}{\tilde{r}_i\delta r_i\delta\tilde{r}_i}(Q_{mp;i-1} - Q_{mp;i}) \cdots$$
$$+ \mu_p Q_{mp;i} + \frac{\lambda_m}{\tilde{r}_i^2}Q_{mp;i} = U_{m;ik}, \tag{53}$$

where $1 \leqslant p \leqslant K$.

(c) The set of simultaneous equations, selected from (53) by having the same m and p, is now solved by standard techniques to obtain the $Q_{mp;i}$. The vertical eigenvector expansion was preferred to a radial expansion because the operators in (53) are asymmetric, and because of the term $(\lambda_m/\tilde{r}_i^2)\,Q_{mp;i}$.

(d) The summations given in (51) and (48) are now performed to obtain all Π_{ijk} from the $Q_{mp;i}$. In practice operations (48) and (49) are not executed as written down because fast transform techniques [2] can be substituted.

EXAMPLE 2 [13]. Poisson's equation in Cartesian coordinates is solved with horizontal periodicity (both x and y) and zero-gradient boundary conditions vertically (z). The horizontal meshes are both symmetrically nonuniform, the vertical mesh is also nonuniform but asymmetric. The finite-difference equations to be solved are

$$\frac{1}{\delta x_i \delta \tilde{x}_{i+1}}(\Pi_{i+ijk} - \Pi_{ijk}) + \frac{1}{\delta x_i \delta \tilde{x}_i}(\Pi_{i-1jk} - \Pi_{ijk}) + \frac{1}{\delta y_j \delta \tilde{y}_{j+1}}(\Pi_{ij+1k} - \Pi_{ijk}) \cdots$$
$$+ \frac{1}{\delta y_j \delta \tilde{y}_j}(\Pi_{ij-1k} - \Pi_{ijk}) + \frac{1}{\delta z_k \delta \tilde{z}_{k+1}}(\Pi_{ijk+1} - \Pi_{ijk}) \cdots$$
$$+ \frac{1}{\delta z_k \delta \tilde{z}_k}(\Pi_{ijk-1} - \Pi_{ijk}) = S_{ijk}\,, \tag{54}$$

where appropriate assumptions are made with regard to horizontal indexing to ensure periodicity.

The procedure adopted is to make transformations in terms of two sets of numerically generated eigenvectors which are functions of either x or y, and leave a set of one-dimensional problems in terms of the vertical coordinate. There is in fact no great overhead in using two different sets of vectors for the two directions, but if all the grid-intervals in one direction are related to those in the other by the same scaling factor, i.e.,

$$\delta x_i = \alpha \delta y\,; \qquad \delta \tilde{x}_i = \alpha \delta \tilde{y}_i \tag{55}$$

for all i (which implies that the numbers of gridpoints in the two directions are the same), then the vectors are identical and the eigenvalues are simply related:

$$\lambda_m(x) = \lambda_m(y)\,\alpha^{-2}. \tag{56}$$

7. EFFECTIVENESS OF THE METHOD

The assumption is made that a large number of equations are to be solved on the same mesh, as in the time-integration of a fluid-mechanical problem, so that the generation of the numerical eigenvectors is a negligible task compared to their use. Since the presence of symmetry only enhances the effectiveness, comparisons will be done (against ADI) in terms of an asymmetric grid.

First, each set of vectors, reciprocal vectors and eigenvalues requires the storage of $(2I + 1)\,I$ numbers, where I is the number of gridpoints in the auxiliary mesh, excluding dummy points. This compares with perhaps 32 iteration parameters in ADI, so it is evident that for large I the latter method is preferable.

Operation counts for the model problem discussed in Sections 2 and 3 are given in Tables II and III. These are obtained by assuming that all necessary coefficients are evaluated once from the grid-intervals and stored, but eigenvalues and iteration parameters are added in as required rather than multiplying the number of coeffi-

TABLE II

Operation Counts for ADI

	$+ -$	$\times \div$
(a) In two dimensions		
Rectangular mesh, per iteration	$10ik + 3(i + k)$	$12ik + 2(i + k)$
Square mesh, per iteration	$10n^2 + 6n$	$12n^2 + 4n$
Square mesh, eight iterations	$80n^2 + 48n$	$96n^2 + 32n$
(b) In three dimensions		
Cuboidal mesh, per iteration	$21ijk + ij + jk + ik + 3(i + j + k)$	$24ijk + 2(i + j + k)$
Cubic mesh, per iteration	$21n^3 + 3n^2 + 9n$	$24n^3 + 6n$
Cubic mesh, eight iterations	$168n^3 + 24n^2 + 72n$	$192n^3 + 48n$

cients to be stored. Moreover any special treatment required at the boundaries is neglected.

It may be seen that, given a minimum of eight ADI iterations, a crossover point occurs for an $n \times n$ mesh when n is about 40. For fewer gridpoints ADI is not only less precise but also slower than the direct solution.

The tables also give counts for the three-dimensional Cartesian problem with all meshes asymmetrically nonuniform. In this case the crossover point for an $n \times n \times n$ mesh is at about 45 points.

The following test was devised for investigation of the Poisson equation solvers, being chosen to simulate the problems we are actually faced with in the overall numerical integration [1]. A two-dimensional array of gridpoints (cylindrical polar coordinates with no azimuthal variation) was defined, with both grids being symmetrically distributed to concentrate points near the boundaries $r = a, b$ and $z = 0, d$.

TABLE III

Operation Counts for Present Method

	$+ -$	$\times \div$
(a) In two dimensions		
Rectangular mesh	$2i^2k + 2ik$	$2i^2k + 5ik$
Square mesh	$2n^3 + 2n^2$	$2n^3 + 5n^2$
(b) In three dimensions		
Cuboidal mesh	$ijk(2i + 2j + 1)$	$ijk(2i + 2j + 5)$
Cubic mesh	$4n^4 + n^3$	$4n^4 + 5n^3$

A pressure field was defined by

$$\begin{aligned} \Pi &= M(r)\, N(z), \\ M(r) &= 1 - \exp((r - b)/0.3)) - \exp((a - r)/0.3)), \\ N(z) &= 1 - \exp((z - d)/0.3)) - \exp(-z/0.3), \end{aligned} \tag{57}$$

with $a = 2$, $b = 5$, and $d = 3$, and a source function was derived using the finite-difference equation (45) with zero-gradient boundary conditions. The Poisson equation was then solved for this source function, using the following initial guess for ADI.

$$\Pi_g = \sin(\pi(r - a)/(b - a)) \sin(\pi z/d). \tag{58}$$

The accuracy was checked by comparing the gradients $\partial\Pi/\partial r$, $\partial\Pi/\partial z$ of the solution against the original, since these are the important quantities in fluid-mechanical problems.

Comparison was made as follows. (a) ADI with 8, 16, or 32 iterations against the present method. (b) Small grid (14×14 points within the boundaries) against larger grid (30×30 points). (c) Single precision (24 bits) against double precision (56 bits) arithmetic. In this connection the accuracy of the eigenvalues in the direct solution is 1 in 10^6 and 1 in 10^{10}, respectively.

The results are presented in Table IV. For each case three numbers are given, first the time required to solve the equation, then the time used to set up eigenvectors or

TABLE IV

Comparison of Speed and Accuracy of Poisson-Equation Solvers

	ADI(8)	ADI(16)	ADI(32)	Direct
14×14	13.44	26.46	52.49	5.32
(s.p.)	(1.77)	(2.29)	(10.00)[a]	(16.87)
	10^{-2}	10^{-4}	3×10^{-5}	5×10^{-6}
14×14	13.22	26.36	52.71	5.31
(d.p.)	(2.08)	(2.82)	(3.96)	(20.52)
	10^{-2}	2×10^{-4}	3×10^{-8}	3×10^{-11}
30×30	53.65	108.13	215.52	32.61
(s.p.)	(1.67)	(2.29)	(4.06)	(61.04)
	2×10^{-2}	6×10^{-4}	3×10^{-4}	10^{-5}
30×30	54.37	106.88	223.86	32.40
(d.p.)	(2.29)	(2.71)	(4.37)	(75.31)
	2×10^{-2}	3×10^{-4}	2×10^{-6}	5×10^{-11}

[a] Precision problems arose in this case during evaluation of iteration parameters, which throws some doubt on this calculation.

iteration parameters, both times being in milliseconds on an IBM 360/195. The third figure indicates the accuracy, obtained by dividing the root mean square error by the r.m.s. gradient. It should be noted that in the course of a time-integration it is generally possible to generate a much better initial guess than (58), with corresponding improvements in the final answer. Nevertheless for a "one-off" problem such a guess is not likely to be available, so that the present method is to be preferred even when the evaluation of eigenvectors has to be taken into account.

REFERENCES

1. I. N. JAMES, P. R. JONAS, AND L. FARNELL, *J. Fluid Mech.*, submitted for publication.
2. W. M. GENTLEMAN AND G. SANDE, "Fast Fourier Transforms for Fun and Profit," Fall Joint Computer Conference, AFIPS Proceedings, Vol. 29, p. 563, Washington, D.C., 1966.
3. R. W. HOCKNEY, "Methods of Computational Physics" (B. Alder, S. Fernbach, and M. Rotenberg Eds.), Vol. 9, p. 136, Academic Press, New York, 1969.
4. E. KALNAY DE RIVAS, *J. Comput. Phys.* **10** (1972), 202.
5. D. W. PEACEMAN AND H. H. RACHFORD. *J. Soc. Ind. Appl. Math.* **3** (1958), 28.
6. C. QUON, *J. Comput. Phys.* **20** (1976), 442.
7. G. O. ROBERTS, *in* "Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics" (M. Holt, Ed.), p. 171, Springer-Verlag, Berlin/Heidelberg/New York, 1971.
8. P. N. SWARZTRAUBER, *SIAM J. Numer. Anal.* **11** (1974), 1136.
9. E. L. WACHSPRESS, "Iterative Solutions of Elliptic Systems," p. 173, Prentice-Hall, Englewood Cliffs, N.J., 1966.
10. R. B. WILHELMSON AND J. M. ERICKSON, *J. Comput. Phys.* **25** (1977), 319.
11. J. H. WILKINSON, "Algebraic Eigenvalue Problem," p. 335, Oxford Univ. Press (Clarendon), Oxford, 1965.
12. G. P. WILLIAMS, *J. Fluid Mech.* **37** (1969), 727.
13. P. J. MASON AND R. I. SYKES, *J. Fluid Mech.*, **91** (1979), 433.
14. A. BRANDT, *Math. Comput.* **31** (1977), 333.
15. R. NICOLAIDES, *J. Comput. Phys.* **19** (1975), 418.
16. B. L. BUZBEE, G. H. GOLUB, AND C. W. NIELSON, *SIAM J. Numer. Anal.* **7** (1970), 627.
17. R. E. BANK AND D. J. ROSE, *SIAM J. Numer. Anal.* **14** (1977), 792.
18. S. A. PIACSEK AND G. P. WILLIAMS, *J. Comput. Phys.* **6** (1970), 392.

581/35/3-10