COMMUNICATIONS IN APPLIED NUMERICAL METHODS, Vol. 3, 63–69 (1987)

PENT: A PERIODIC PENTADIAGONAL SYSTEMS SOLVER

I. M. NAVON

Supercomputer Computations Research Institute, Florida State University, Tallahassee, FL 32306–4052, U.S.A.

INTRODUCTION

Systems of linear algebraic equations with a periodic pentadiagonal matrix often occur when solving partial differential equations in meteorology and oceanography, as well as in numerical analysis. Their appearance is typically connected with the approximation of fourth-order finite differencing of differential equations subject to periodic boundary conditions, data smoothing by cubic splines, use of generalized splines, etc. In our particular case, solution of the shallow-water equations by means of a finite element two-stage Numerov–Galerkin method requires the numerical solution of cyclic pentadiagonal systems when approximating the first derivative by a generalized spline.[4,14] In this application, we generalized the approach of Ahlberg *et al.*[1] for a cyclic pentadiagonal matrix.

While work on cyclic tridiagonal systems has been carried out by several researchers (see, for instance, References 2, 6–8, 10 and 15), cyclic pentadiagonal systems for non-symmetric matrices have been treated by Benson and Evans[18] for positive definite symmetric systems by Benson and Evans[19] following Cuthill and Varga.[20] Solutions for regular pentadiagonal systems have been proposed by Von Rosenberg[9] and have been also treated by general real band linear equations solvers of the NAG[15] scientific library (routine F04LDF) Miklosko[5] proposed a method of shooting for solving a pentadiagonal system of linear algebraic equations. For the same system Grund[13] proposed two methods—one direct and the other Gaussian—for finding the inverse of five-diagonal matrices. A parallel algorithm for inverting pentadiagonal matrices was proposed by Cei *et al.*[11]

In the present paper, we present the new algorithm employed for solving the cyclic pentadiagonal system following a suggestion of Temperton[8] and document briefly a FORTRAN program which implements the method. The new pentadiagonal algorithm is a generalization of the Ahlberg–Nilson–Walsh method for cyclic tridiagonal systems and is applicable to general cyclic pentadiagonal systems in which the matrix A is neither symmetric nor circulant. It seems to have a simpler form than Algorithm 80 of Benson and Evans[18] and is easily amenable to different applications. Our main intention in this short paper is to present the new algorithm and its applications without comparing it for accuracy and efficiency to existing periodic pentadiagonal algorithms. Two model problems are presented, the first involving fourth-order finite difference approximations and the second employing a high-order generalized spline of order $O(h^8)$ for the calculation of the first derivative to illustrate and verify the correctness of the FORTRAN routine given in Appendix I.

THE CYCLIC PENTADIAGONAL ALGORITHM

The general form of the cyclic pentadiagonal matrix system is

$$Ax = \begin{bmatrix} c & d & e & & & a & b \\ b & c & d & e & & & a \\ a & b & c & d & e & & 0 \\ & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & 0 & \ddots & \ddots & \ddots & e \\ e & & & & \ddots & \ddots & d \\ d & e & & & a & b & c \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ x_{N-2} \\ x_{N-1} \\ x_N \end{bmatrix} = \begin{bmatrix} d_1 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ d_{N-2} \\ d_{N-1} \\ d_N \end{bmatrix} \tag{1}$$

0748–8025/87/010063–07$05.00

Received June 1986

Following Ahlberg *et al.*,[1] we split matrix A into a pure $(n-2 \times n-2)$ pentadiagonal matrix E by deleting the last 2 rows and last 2 columns of A:

$$E = \begin{bmatrix} c & d & e & & & & \\ b & c & d & e & & & \\ a & b & c & d & e & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ & & \ddots & \ddots & \ddots & \ddots & e \\ & 0 & & \ddots & \ddots & c & d \\ & & & & a & b & c \end{bmatrix}_{(n-2) \times (n-2)} \tag{2}$$

and let $\hat{X}$ be the $(n-2)$ unknown vector

$$\hat{X} = \begin{bmatrix} x_1 \\ | \\ | \\ | \\ | \\ | \\ x_{n-2} \end{bmatrix}_{(n-2)} \qquad g_n = \begin{bmatrix} e & d \\ 0 & e \\ | & 0 \\ 0 & | \\ a & 0 \\ b & a \end{bmatrix}_{(n-2) \times 2} \tag{3}$$

$$\hat{d} = \begin{bmatrix} d_1 \\ | \\ | \\ | \\ | \\ | \\ d_{n-2} \end{bmatrix}_{(n-2)} \qquad f_n = \begin{bmatrix} a & b \\ 0 & a \\ 0 & 0 \\ 0 & | \\ e & 0 \\ d & e \end{bmatrix}_{(n-2) \times 2} \tag{4}$$

$\hat{d}$ is the corresponding $(n-2)$ right-hand-side vector (6), while g_n and f_n represent the effects of periodicity terms at the lower left-hand side and upper right-hand side of the original cyclic pentadiagonal matrix. We then obtain that the system (1) can be rewritten as

$$E\hat{X} + f_n \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} = \hat{d} \tag{5}$$

$$g_n^{\mathrm{T}}\hat{X} + \begin{bmatrix} c\,d \\ b\,c \end{bmatrix} \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} = \begin{bmatrix} d_{n-1} \\ d_n \end{bmatrix} \tag{6}$$

We have also

$$\hat{d} + \begin{bmatrix} d_{n-1} \\ d_n \end{bmatrix} = d \tag{7}$$

From (5) we obtain

$$\hat{X} = E^{-1}\hat{d} - E^{-1}f_n \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} \tag{8}$$

and if we substitute (8) into (6) we obtain

$$g_n^{\mathrm{T}}E^{-1}\hat{d} - g_n^{\mathrm{T}}E^{-1}f_n \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} + \begin{bmatrix} c\,d \\ b\,c \end{bmatrix} \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} = \begin{bmatrix} d_{n-1} \\ d_n \end{bmatrix} \tag{9}$$

or

$$\left\{\begin{bmatrix} c\,d \\ b\,c \end{bmatrix} - g_n^{\mathrm{T}} E^{-1} f_n\right\} \cdot \begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} = \begin{bmatrix} d_{n-1} \\ d_n \end{bmatrix} - g_n^{\mathrm{T}} E^{-1} \hat{d} \tag{10}$$

which finally permits us to obtain the two components of the unknown vector $\begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix}$ explicitly as

$$\begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix} = \left\{\begin{bmatrix} c\,d \\ b\,c \end{bmatrix} - g_n^{\mathrm{T}} E^{-1} f_n\right\}^{-1} \cdot \left\{\begin{bmatrix} d_{n-1} \\ d_n \end{bmatrix} - g_n^{\mathrm{T}} E^{-1} \hat{d}\right\} \tag{11}$$

Algorithmic implementation

To solve equation (10) for $\begin{bmatrix} X_{n-1} \\ X_n \end{bmatrix}$ we need a usual pentadiagonal solver of size $(n-2)\times(n-2)$ and we employ the pentadiagonal solver due to Von Rosenberg.[9] We then calculate first $E^{-1}f_n$ by calling the subroutine PENTDG, implementing the usual pentadiagonal solver. We then calculate $E^{-1}\hat{d}$ and finally the values of the right-hand-side bracketed expression in (11). We then proceed to calculate the explicit inverse of the first bracketed expression in (11), i.e.

$$\left\{\begin{bmatrix} c\,d \\ b\,c \end{bmatrix} - g_n^{\mathrm{T}} E^{-1} f_n\right\}^{-1} \tag{12}$$

by using Cramer's rules and having already calculated the expression for $g_n^{\mathrm{T}} E^{-1} f_n$.

The sought values of X_{n-1} and X_n can then be calculated. The algorithm developed is valid for a general pentadiagonal matrix, but it was implemented with a symmetric cyclic pentadiagonal matrix.

When the matrix E is neither positive definite nor diagonally dominant, one can use the general band solver performing an LU decomposition into triangular matrices—for instance routine F04LDF of NAG[15] scientific library. For further references see Wilkinson and Reinsch.[17]

NUMERICAL APPLICATIONS

The program PENT has been applied towards the numerical solutions of the simple following linear ordinary differential equation:[2]

$$f''(x) + f(x) = (1-4\pi^2)\sin(2\pi x) \quad 0\leqslant X\leqslant 1 \tag{13}$$

Subject to periodic boundary conditions

$$f(1) = f(0) \tag{14}$$
$$f(-h) = f(1-h)$$

where $h = \Delta x$ is the spacer grid size and

$$h = \frac{1}{N} \tag{15}$$

whose exact solution is

$$f(x) = \sin(2\pi x) \tag{16}$$

allowing therefore an easy accuracy comparison and an adequate test for first-time users.

Using a fourth-order finite difference approximation for the second derivative, i.e.

$$f'' = \frac{\mathrm{d}^2 f}{\mathrm{d}x^2} = (f_{i-2} + 16f_{i-1} - 30f_i + 16f_{i+1} - f_{i+2})/12h^2 \tag{17}$$

(see, for instance, Reference 3) we obtain the following cyclic (periodic) pentadiagonal system of order N:

$$
\frac{1}{12}
\begin{bmatrix}
-30+12h^2 & & & 16 & -1 & & -1 & 16 \\
16 & -30+12h^2 & & & 16 & -1 & & -1 \\
-1 & 16 & -30+12h^2 & & & 16 & -1 & \\
\ddots & \ddots & \ddots & & & \ddots & & 0 \\
& & \ddots & \ddots & \ddots & & \ddots & -1 \\
0 & & & \ddots & \ddots & \ddots & & 16 \\
-1 & & & & \ddots & \ddots & \ddots & \\
16 & -1 & & & -1 & 16 & -30+12h^2 &
\end{bmatrix}
\cdot
\begin{bmatrix} f_1 \\ \vdots \\ \vdots \\ f_n \end{bmatrix}
= h^2
\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ (1-\pi^2)\sin(2\pi(I-1)h) \\ \vdots \\ d_n \end{bmatrix}
\tag{18}
$$

The solution vector provides the grid-point values for equation (1). One can calculate in this case the average truncation errors for the numerical solution (see Reference 2) of f, by

$$
\epsilon = \left[\sum_{i=1}^{N} |f_i - \sin(2\pi(I-1)h| \right] \Big/ N \tag{19}
$$

The errors for finer and finer meshes as N increases should vanish proportionally to h^4. The values are displayed in Table I, and indeed they appear to decrease proportionally to h^4 thus verifying the correctness of our routine PENT. In another application,[4] a generalized spline application giving a high-order approximation of order $O(h^8)$ to the first derivative $\partial u/\partial x$ is[1]

$$
\begin{aligned}
&\frac{1}{70}\left[\left(\frac{\partial u}{\partial x}\right)_{i+2} + 16\left(\frac{\partial u}{\partial x}\right)_{i+1} + 36\left(\frac{\partial u}{\partial x}\right)_{i} + 16\left(\frac{\partial u}{\partial x}\right)_{i-1} + \left(\frac{\partial u}{\partial x}\right)_{i-2}\right] \\
&= \frac{1}{84h}\left[-5u_{i-2} - 32u_{i-1} + 32u_{i+1} + 5u_{i+2}\right] + O(h^8)
\end{aligned}
\tag{20}
$$

This is an implicit finite difference approximation to $\partial/\partial x$ having a better resolving power than its spline counterpart of the same bandwidth.[14]

Using again $u=\sin(2\pi x)$, $0\leqslant x\leqslant 1$, we obtain the following cyclic pentadiagonal matrix:

$$
\frac{1}{70}
\begin{bmatrix}
36 & 16 & 1 & & & 1 & 16 \\
16 & 36 & 16 & 1 & & & 1 \\
1 & 16 & 36 & 16 & 1 & & \\
& \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
& 0 & \ddots & \ddots & \ddots & \ddots & 1 \\
1 & & & 1 & 16 & 36 & 16 \\
16 & 1 & & & 1 & 16 & 36
\end{bmatrix}
\begin{bmatrix}
\left(\frac{\partial u}{\partial x}\right)_1 \\
\left(\frac{\partial u}{\partial x}\right)_2 \\
\vdots \\
\left(\frac{\partial u}{\partial x}\right)_N
\end{bmatrix}
= \tag{21}
$$

$$
\frac{1}{84h}
\begin{bmatrix}
-5u_3 - 32u_2 + 32u_n + 5u_{n-1} \\
-5u_4 - 32u_3 + 32u_1 + 5u_n \\
\vdots \\
-5u_{i+2} - 32u_{i+1} + 32u_{i-1} + 5u_{i-2} \\
\vdots \\
-5u_1 - 32u_n + 32u_{n-2} + 5u_{n-3} \\
-5u_2 - 32u_1 + 32u_{n-1} + 5u_{n-2}
\end{bmatrix},
\qquad u_i = \sin 2\pi(i-1)h \tag{22}
$$

Here the solution is

$$u_x = \frac{\partial u}{\partial x} = 2\pi\cos(2\pi x), f'_i = \left(\frac{\partial u}{\partial x}\right)_i \tag{23}$$

and the error is given as

$$\epsilon' = \left[\sum_{i=1}^{N} |f'_i - 2\pi\cos(2\pi(i-1)h)|\right] \Big/ N \tag{24}$$

A test was carried out and indeed the errors decrease as h^8 for finer and finer meshes as N increases, and indeed we obtain a decrease of $2^8 = 256$ in the error. As the accuracy on the CDC-205 started to deteriorate in single precision around 10^{-13}–10^{-14}, a double precision routine PENT1 was written for the calculation of $\partial u/\partial x$ with the $O(h^8)$ generalized spline.

Table I

N	20	40	80	160	320
ϵ_{h^4}	$0{\cdot}695E-4$	$0{\cdot}434E-5$	$0{\cdot}276E-6$	$0{\cdot}172E-7$	$0{\cdot}107E-8$
$\epsilon_{h^8 GS}$	$0{\cdot}87013E-8$	$0{\cdot}33711E-10$	$0{\cdot}13141E-12$	$0{\cdot}51307E-15$	$0{\cdot}20039E-17$

Not only is this very high-order implicit centred difference scheme highly accurate with a truncation error $O(h^{4m})$ and bandwidth $2m+1$, but its phase error is

$$\epsilon_m(N,m) = 2\pi\{1-b_m(\theta)/[i\theta a_m(\theta)]\} \tag{25}$$

where

$$\theta = 2\pi wh = 2\pi/N \tag{26}$$

N being the nmber of intervals per wavelength given by

$$N = (wh)^{-1} \tag{27}$$

and

$$b_m(\theta)/a_m(\theta) = i\sin\theta g_m\,[\sin^2(\theta/2)] \tag{28}$$

where g_m is the truncation error of a continued fraction approximation of arc $\sin(\gamma)/\gamma\sqrt{1-\gamma^2}$).

So, as a by-product of the periodic pentadiagonal solver a double-precision routine for high-accuracy calculation of the first derivative to $O(h^8)$ is also provided. The complete computer programs and results are available upon request.

ACKNOWLEDGEMENTS

Mrs. R. de Villiers of NRIMS/Council of Scientific and Industrial Research, Pretoria, South Africa, expertly programmed the first version of the periodic pentadiagonal solver. The expert help of Dr. G. Beckwith, CDC consultant, is acknowledged.

APPENDIX I

```
      PROGRAM  PENT  (INPUT,OUTPUT,TAPE3=OUTPUT)
C****THE FOLLOWING PROGRAM SOLVES A PERIODIC PENTADIAGONAL SYSTEM OF **
C****LINEAR EQUATIONS OF DIMENSION NX.
C****THE COEFFICIENTS ARE A(I-2),B(I-1),C(I)AND IN OUR EXAMPLE THE  **
C****PENTADIAGONAL MATRIX IS SYMMETRIC,BUT THE PROGRAM CAN SOLVE A GEN**
C****NERAL CYCLIC PENTADIAGONAL SYSTEM.
C****THE R.H.S. IS GIVEN IN THE D ARRAY.THE SOLUTION ARRAY IS RETURNED**
C****IN THE Z(NX) ARRAY.THE METHOD USED IS A GENERALIZATION OF THE
C**** AHLBERG,NILSON AND WALSH ALGORITHM.WE USE VON ROSENBERG'S
C****METHOD FOR THE (N-2)X(N-2) PENTADIAGONAL MATRIX.
      PARAMETER(NX=160,NXP1=161)
      DIMENSION D(NXP1),Z(NX),TMP(NX),V(NX),W(NX,2),FN(NX)
      PI=2.*ASIN(1.0)
      DX=1./FLOAT(NX)
```

```
C****COEFFICIENTS OF SPECIFIC PROBLEM-FOURTH ORDER FINITE-DIFFERENCE
C****APPROXIMATION TO THE SECOND DERIVATIVE.
      A=-1./12.
      B=16./12.
      C=(-30./12.+DX**2)
      NX1=NX-1
      NX2=NX-2
      NX3=NX-3
      NX4=NX-4
C        CALCULATE W=E(INVERSE)FN
C        FIRST COLUMN
      DO 1 I=2,NX4
    1 FN(I)=0.
      FN(1)=A
      FN(NX3)=A
      FN(NX2)=B
      CALL PENTDG(TMP,FN,NX2,DX)
      DO 2 I=1,NX2
      W(I,1)=TMP(I)
C        SECOND COLUMN
    2 W(I,2)=TMP(NX1-I)
C        CALCULATE V=E(INVERSE)D
      NP1=NX+1
      DO 101 I=1,NP1
      D(I)=DX**2*(1.-4.*PI**2)*SIN(2.*PI*(I-1)*DX)
 101  CONTINUE
      CALL PENTDG(V,D,NX2,DX)
      GW11=A*W(1,1)+A*W(NX3,1)+B*W(NX2,1)
      GW12=A*W(1,2)+A*W(NX3,2)+B*W(NX2,2)
      GW21=B*W(1,1)+A*W(2,1)+A*W(NX2,1)
      GW22=B*W(1,2)+A*W(2,2)+A*W(NX2,2)
      GV1=A*V(1)+A*V(NX3)+B*V(NX2)
      GV2=B*V(1)+A*V(2)+A*V(NX2)
      DMGV1=D(NX1)-GV1
      DMGV2=D(NX)-GV2
      C11=C-GW11
      C12=B-GW12
      C21=B-GW21
      C22=C-GW22
      CDET=C11*C22-C12*C21
      CI11=C22/CDET
      CI12=(-C12)/CDET
      CI21=(-C21)/CDET
      CI22=C11/CDET
      Z(NX1)=CI11*DMGV1+CI12*DMGV2
      Z(NX)=CI21*DMGV1+CI22*DMGV2
      DO 5 I=1,NX2
      WZ=W(I,1)*Z(NX1)+W(I,2)*Z(NX)
    5 Z(I)=V(I)-WZ
      SUM=0.0
C****CALCULATION OF TRUNCATION ERROR AS DIFFERENCE BETWEEN NUMERICAL
C****AND ANALYTIC SOLUTIONS.
      DO 102 I=1,NX

      SUM=SUM+ABS(Z(I)-SIN(2.*PI*(I-1)*DX))
  102 CONTINUE
C***AVERAGE TRUNCATION ERROR.
      SUM=SUM/NX
      PRINT 998,SUM
  998 FORMAT(5X,'EPS=',E12.5)
      PRINT 999,(Z(I),I=1,NX)
 999  FORMAT(1X,5E12.5/)
      RETURN
      END

      SUBROUTINE PENTDG(U,F,NX,DX)
      DIMENSION U(NX),F(NX)
      REAL DEL(160),LAM(160),GAM(160),MU
C       SUBROUTINE PENTDG SOLVES THE EQUATIONS
C       A*U(I-2)+B*U(I-1)+C*(I)+D*(I+1)+E*(I+2)=F(I)
C       FOR 1.LE.I.LE.NX
C       WITH A=0 FOR I=1 AND I=2
C             B=0 FOR I=1
C             D=0 FOR I=NX
C             E=0 FOR I=(NX-1) AND I=NX
C
C****COEFFICIENTS FOR FOUTRH ORDER FINITE DIFFERNCE APPROXIMATION TO
C**** SECOND ORDER DERIVATIVE.
      A=-1./12.
      B=16./12.
      C=(-30./12.+DX**2)
      D=B
      E=A
```

```
C
      NX1=NX-1
      NX2=NX-2
      NX3=NX-3
C       I=1
      DEL(1)=D/C
      LAM(1)=E/C
      GAM(1)=F(1)/C
C       I=2
      MU=C-B*DEL(1)
      DEL(2)=(D-B*LAM(1))/MU
      LAM(2)=E/MU
      GAM(2)=(F(2)-B*GAM(1))/MU
C       3.LE.I.LE.(NX-2)
      DO 1 I=3,NX2
      BETA=B-A*DEL(I-2)
      MU=C-BETA*DEL(I-1)-A*LAM(I-2)
      DEL(I)=(D-BETA*LAM(I-1))/MU
      LAM(I)=E/MU
      GAM(I)=(F(I)-BETA*GAM(I-1)-A*GAM(I-2))/MU
    1 CONTINUE
C       I=NX-1
      BETA=B-A*DEL(NX3)
      MU=C-BETA*DEL(NX2)-A*LAM(NX3)
      DEL(NX1)=(D-BETA*LAM(NX2))/MU
      GAM(NX1)=(F(NX1)-BETA*GAM(NX2)-A*GAM(NX3))/MU
C       I=NX
      BETA=B-A*DEL(NX2)
      MU=C-BETA*DEL(NX1)-A*LAM(NX2)
      GAM(NX)=(F(NX)-BETA*GAM(NX1)-A*GAM(NX2))/MU
C       BACK SOLUTION
      U(NX)=GAM(NX)
      U(NX1)=GAM(NX1)-DEL(NX1)*U(NX)
      DO 2 J=1,NX2
      I=NX1-J
      U(I)=GAM(I)-DEL(I)*U(I+1)-LAM(I)*U(I+2)
    2 CONTINUE
      RETURN
      END
```

REFERENCES

1. H. H. Ahlberg, E. N. Nilson and J. L. Walsh, *The Theory of Splines and their Application*, Academic Press, New York, 1967.
2. M. Napolitano, 'A Fortran subroutine for the solution of periodic block-tridiagonal systems', *Commun. appl. numer. methods,* **1**, 11–15 (1985).
3. I. M. Navon and Z. Alperson, 'Application of fourth-order finite differences to a baroclinic model of the atmosphere', *Arch. Met. Geoph. Biokl.*, ser. A., **27**, 1–19 (1978).
4. I. M. Navon, 'A Numerov–Galerkin technique applied to a finite-element shallow-water equations model with enforced conservation of integral invariants and selective lumping', *J. Comp. Phys.*, **52**, 313–339 (1983).
5. J. Miklosko, 'Method of shooting for a pentadiagonal systems of linear algebraic equations', *ZAMM*, **56**, 273–276 (1976).
6. D. J. Evans and S. O. Okolie, 'A generalized sparse factorization method for the solution of periodic tridiagonal systems', *Comp. Maths Appls.*, **5**, 211–216 (1979).
7. W. N. Pickering, 'On the solution of a quasi-tridiagonal systems of linear equations', *Int. J. Comp. Math,* **15**, 181–191 (1984).
8. C. Temperton, 'Algorithms for the solutions of cyclic tridiagonal systems', *J. Comp. Phys.*, **19**, 317–323 (1975).
9. D. U. Von Rosenberg, *Methods for the Numerical Solution of Partial Differential Equations*, G. L. Farrar Associates, Tulsa, OK, U.S.A., 128 pp., 1984.
10. D. J. Evans and L. V. Atkinson, 'An algorithm for the solution of general three-term linear-systems', *Comp. J.*, **13**, 323–324 (1970).
11. S. Cei, M. Galli, M. Soccio and P. Zellini, *Calcolo*, **18**, 303–319 (1984).
12. D. J. Evans and M. J. Biggins, 'Peripheral block over relaxation methods for triangular grids', *Comp. Meth. Appl. Mech Eng.*, **27**, 63–80 (1981).
13. F. Grund, 'Inversion of five-diagonal matrices', *Zh. Vy chisl. Math. mat.*, Fiz 17, **5**, 1123–1131 (1977).
14. B. K. Swartz and B. Wendroff, 'The relative efficiency of finite difference and finite-element methods. I. Hyperbolic problems and splines', *SIAM J. Numer. Anal.*, **11**, 979–993 (1974).
15. NAG, *Numerical Algorithms Group Fortran Library Manuals*, vols. 1–6, NAG, Banbury Road, Oxford, England, or NAG Inc, Downers Grove, Ill., U.S.A., 1982.
16. I. M. Navon, 'Algorithms for the solution of scalar and block cyclic tridiagonal systems', *CSIR Spec. Rep. WISK 265*, Pretoria, South Africa, 25 pp. (1977).
17. D. M. Wilkinson and C. Reinsch, *Handbook for Automatic Computation,* vol. II, *Linear Algebra*. Springer-Verlag, Berlin, 1971.
18. A. Benson and D. J. Evans, 'Algorithm 80: an algorithm for the solution of periodic quindiagonal systems of linear equations', *Comp. J.*, **16**(3), 278–279 (1973).
19. A. Benson and D. J. Evans, 'Algorithm 512: a normalized algorithm for the solution of positive definite symmetric quindiagonal systems of linear equations', *ACM Trans. Math. Soft.*, **3**(1), 96–103 (1977).
20. E. H. Cuthill and R. S. Varga, 'A method of normalized block iterative', *J. ACM*, **6**(2), 236–244 (1959).