Application of Matrix Iterative-Inversion in Solving Eigenvalue Problems in Structural Engineering

[1,]O. M. Ibearugbulem, [2,]L. O. Ettu, J. C. Ezeh, [3,]U. C. Anya
[1,2,3,4,]Civil Engineering Department, Federal University of Technology, Owerri, Nigeria

Abstrct:

There are many methods of solving eigenvalue problems, including Jacobi method, polynomial method, iterative methods, and Householder's method. Unfortunately, except the polynomial method, all of these methodsare limited to solving problems that have lump mass matrices. It is difficult to use them when solving problems that have consistent mass or stiffness matrix. The polynomial method also becomes very difficult to use when the size of the matrix exceeds 3 x 3. There is, therefore,a need for a method that can be used in solving all types of eigenvalue problems for allmatrix sizes. This work provides such a method by the application of matrix iterative-inversion, Iteration-Matrix Inversion (I-MI) method,consisting in substituting a trial eigenvalue, λ into $(A - \lambda B) = 0$, and checking if the determinant of the resultant matrix is zero. If the determinant is zero then the chosen eigenvalue is correct; but if not, another eigenvalue will be chosen and checked, and the procedure continued until a correct eigenvalue is obtained. A QBASIC program was written to simplify the use of the method. Five eigenvalue problems were used to test the efficiency of the method. The results show that the newly developed I-MI method is efficient in convergence to exact solutions of eigenvalues. The new I-MI method is not only efficient in convergence, but also capable of handling eigenvalue problems that use consistent mass or stiffness matrices. It can be used without any limit for problems whose matrices are of n X n order, where $2 \leq n \leq \infty$. It is therefore recommended for use in solving all the various eigenvalue problems in structural engineering.

Keywords: *Eigenvalue, Matrix, Consistent mass, Consistent Stiffness, Determinant*

I. INTRODUCTION

The dynamic equation in structural dynamicsis given by Fullard(1980) as in equation (1).

$\{F(t)\} = [M]\{X^{11}\} + [C]\{X^1\} + [K]\{X\}$ -------------------- (1)

where {F(t)} is the time-dependent loading vector; [M] is the mass matrix; $\{X^1\}$ and $\{X^{11}\}$ are the first and second time derivatives of the response vector,{X}; [C] is the damping matrix; and [K] is the stiffness matrix. For the case of free vibration in nature where {F(t)} = [c] = 0, the dynamic equation reduces to equation (2).

$[M]\{X11\} + [K]\{X\} = 0$ --------------------- (2)

When a static stability case is considered,the equation further reduces to equation (3).

$\{F\} = [K - Kg]\{X\}$ --------------------- (3)

Where K is the material stiffness and Kg is the geometric stiffness. {F} is the vector of bending forces (shear force and bending moment). The continuum (beam or plate) can bucklein cases of axial forces only with no bending forces; the equation becomes as written in equation (4).

$0 = [K - Kg]\{X\}$ --------------------- (4)

Equation (2) is used in finding the natural frequencies in structural dynamics, while equation (4) is used to determine the buckling loads in structural stability. The solutions in both cases are called eigenvalue or characteristic value solutions.

In structural mechanics, shape functions are usually assumed to approximate the deformed shape of the continuum. If the assumed shape function is the exact one, then the solution will converge to exact solution. The stiffness matrix [K] is formulated using the assumed shape function. In the same way, the mass matrix [M] and the geometric stiffness matrix [Kg] will be formulated. The mass matrix and the geometric stiffness matrix formulated in this way are called consistent mass matrix and consistent geometric stiffness matrix respectively (Paz, 1980 and Geradin, 1980).

Owen (1980) transformed equation (2) to into the form expressed in equations (5a) and (5b).
$[K]\{X\} = \omega^2[M]\{X\}$--------------------- (5a)
$([K] – \omega^2[M])\{X\} = 0$ --------------------- (5b)

It is easy to determine the eigenvalues of equations (4) and (5) when the size of the square matrix is not more than 3 x 3. Geradin (1980) noted that significant amount of computational effort is required for the eigenvalue problems using consistent mass matrix. Because of this difficulty, many analysts preferred using lump mass matrix to consistent mass matrix. The works of Key and Krieg (1972) and Key (1980) showed that the difference between the solutions from lump mass and consistent mass is very significant. Since the difference in the solutions is high, analystsneed not stick to the use of lump mass just because it is easy to solve.Sheik et al.(2004) recommended efficient mass lumping scheme to form a mass matrix having zero mass for the internal nodes. This, according to them, would help facilitate condensation of the structural matrix. The use of lump mass matrix will transform equation (5) into the form written in equation (6).
$([K] – \omega^2m[I])\{X\} = 0$ --------------------- (6)
where [I] is the identity matrix. Equation (6) can simply be written as shown in equation (7).
$(A – \lambda I)X = 0$ --------------------- (7)

where A is a square matrix, λ is a scalar number called eigenvalue or characteristic value of matrix A, I is identity matrix, and X is the eigenvector (Stroud, 1982 and James, Smith and Wolford, 1977). There are many methods of solvingequations (6) and (7). Some of the methods include Jacobi method, polynomial method, iterative methods, and Householder's method (Greenstadt, 1960; Ortega, 1967; and James, Smith and Wolford, 1977).Iterative methodsarebased on matrix–vector multiplication. Some of the iterative methods include power method, inverse iteration method (Wilkinson, 1965), Lanczos method (Lanczos, 1950), Arnoldi method (Arnoldi, 1951; Demmel, 1997; Bai et al., 2000; Chatelin, 1993; and Trefethen and Bau, 1997), Davidson method, Jacobi-Davidson method (Hochstenbach and Notay, 2004; and Sleijpen and van der Vorst, 1996), minimum residual method, generalized minimumresidual method(Barrett et al., 1994), multilevel preconditioned iterative eigensolvers (Arbenz and Geus, 2005), block inverse-free preconditioned Krylov subspace method (Quillen and Ye, 2010), Inner-outer iterative method (Freitag, 2007), and adaptive inverse iteration method (Chen, Xu and Zou, 2010). Unfortunately, except the polynomial method, all of these methodscan only be used for equations (6) and (7); they cannot handle equations (4) and (5); andas previously noted, Polynomial method also becomes very difficult to use when the size of the matrix exceeds 3 x 3.

There is, therefore,a need for a method that can be used in solving eigenvalue problems of equations (6) and (7) as well as equations (4) and (5) for any size of matrix.This work provides such a method by the application of matrix iterative-inversion, consisting in substituting a trial eigenvalue, λ into (A – λB) = 0, and checking if the determinant of the resultant matrix is zero. If the determinant is zero then the chosen eigenvalue is correct; but if not, another eigenvalue will be chosen and checked, and the procedure continued until a correct eigenvalue is obtained.

II. MATRIX ITERATIVE-INVERSION

Trivial solutions will exist for both equations (4) and (5)if and only if {X} = 0. To avoid trivial solutions, equations (4) and (5) will respectively satisfy equations (8) and (9).

$$|[K - Kg]| = 0 \quad ------------(8)$$

$$|[K] - \omega^2[M]| = 0 \quad ------------ \quad (9)$$

Equations (8) and (9) can simply be written as in equation (10).

$$|[A] - \lambda[B]| = 0 \quad -------------(10)$$

$$\text{Let } [C] = [[A] - \lambda[B]] -------------(11)$$

$$\text{That is to say } |C| = 0 \quad -------------(12)$$

The inverse of matrix [C] is denoted as $[C]^{-1}$. From elementary mathematics,

$$[C]^{-1} = \frac{[D]^T}{|C|} -------------(13)$$

Where [D] is the matrix of the cofactors of the elements of . The implication of equation (13) is that the inverse of matrix C, $[C]^{-1}$ will be infinity (and this does not exist) as long as its determinant is equal to zero. The approach used in this work is to deal with the inverse matrix because it is easier to evaluate the inverse of a matrix using row operation rather than the determinant of the same matrix. The iteration process can start withtaking the value of λ as zero and checking if the inverse $[C]^{-1}$ exists or not. If the inverse exists then zero is not the eigenvalue; λ will then be increased (say by 0.1) and used to test if the inverse of C matrix exists. If the inverse still exists, λ will again be increased and the process repeated until the inverse matrix ceases to exist as it becomes infinity. The value of λ at which the inverse matrix becomes infinity is the lowest eigenvalue.The next eigenvalue will be a slight increment of this lowest eigenvalue, say λ+0.1.

III. QBASIC PROGRAM FOR THE METHOD

A simple user-friendly and interactiveQBASIC program which requires no special training to be usedwas written in order tosimplify the use of this method (see appendix to this work).The program was used to test the following problems.

[1] $$\left\| \begin{bmatrix} 2 & 0 & 1 \\ -1 & 4 & -1 \\ -1 & 2 & 0 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \right.$$ (Stroud, 1982)

[2] $$\left\| \begin{bmatrix} 0.1 & 0.1 & 0.1 \\ 0.1 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.3 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \right.$$ (James, Smith and Wolford, 1977)

[3] $$\left\| \begin{bmatrix} 5.143 & 3.863 & 3.543 \\ 3.863 & 6.05 & 6.857 \\ 3.543 & 6.857 & 8.341 \end{bmatrix} - \lambda \begin{bmatrix} 0.324 & 0.194 & 0.1 \\ 0.194 & 0.139 & 0.1 \\ 0.162 & 0.121 & 0.1 \end{bmatrix} \right.$$

[4] $$\left\| \begin{bmatrix} 1.8844 & 4.7212 & 4.7212 \\ 4.7212 & 11.8316 & 11.80294 \\ 4.7212 & 11.80294 & 11.91948 \end{bmatrix} - \lambda \begin{bmatrix} 0.0478 & 0.1195 & 0.11 \\ 0.1195 & 0.29904 & 0.2 \\ 0.1195 & 0.299 & 0.3 \end{bmatrix} \right.$$

[5] $$\left\| \begin{bmatrix} 0.60953 & 0.3962 & 0.34287 & 0.30477 \\ 0.3962 & 0.4039 & 0.40964 & 0.41738 \\ 0.34287 & 0.40964 & 0.44353 & 0.4777 \\ 0.30477 & 0.41738 & 0.4777 & 0.53884 \end{bmatrix} - \lambda \begin{bmatrix} 0.0127 & 0.00762 & 0.00635 & 0.005 \\ 0.00762 & 0.00544 & 0.00476 & 0.004 \\ 0.00635 & 0.00476 & 0.00423 & 0.003 \\ 0.00544 & 0.00423 & 0.00381 & 0.003 \end{bmatrix} \right.$$

IV. RESULTS, DISCUSSION, AND CONCLUSION

The resulting lowest eigenvalues obtained for the above five problemsby use of the developed Q BASIC program are as shown in Table 1. When these lowest eigenvalues are substituted into their respective problems, and the determinants of the problems calculated, the resulting values of the determinants are as shown in Table 2. It is a common knowledge that the determinant of an eigenvalue matrix is zero when the exact eigenvalue is substituted into it. Hence, if the eigenvalues in table 1 were exact or approximate eigenvalues of matrices 1, 2, 3, 4 and 5, the determinants would be exactly or approximately equal to zeroupon substituting the eigenvalues into the matrices.Table 2 shows that the determinants from the Iteration-Matrix Inversion (I-MI) method areapproximately zero.It can also be seen from table 2 that the determinant for matrix 2 from power method (James, Smith and Wolford, 1977) is far from being zero. The results show that the newly developed I-MI method is efficient in convergence to exact solutions of eigenvalues. The new I-MI method is not only efficient in convergence, but also capable of handling eigenvalue problems that use consistent mass or stiffness matrices.It can be used without any limit for problems whose matrices are of n X n order, where $2 \leq n \leq \infty$. It is therefore recommended for use in solving all the various eigenvalue problems in structural engineering.

Table 1: Results of Eigenvalue Problems

Problem	Eigenvalues from Matrix Iterative-Inversion method				1st Eigenvalues from Reference
	1st eigenvalue	2nd eigenvalue	3rd eigenvalue	4th eigenvalue	
1	1	2	3		1[1]
2	0.031	0.2618	0.5049		1.98[2]
3	15.113	15.8732	63.2421		No reference
4	10.089	10.189	10.289		No reference
5	47.2399	47.9867	116.9181	117.0181	No reference

1:(Stroud, 1982); 2: (James, Smith and Wolford, 1977)

Table 2: Determinants of Eigenvalue Problems

Problem	Determinant from Matrix Iterative-Inversion method				Determinant from Reference
	From 1st eigenvalue	From 2nd eigenvalue	From 3rd eigenvalue	From 4th eigenvalue	
1	0	0	0		0[1]
2	-3.2E-06	0.01109	-0.0000017		-5.50815[2]
3	5.67E-06	-0.18079	-0.00285		No reference
4	-0.011196	-0.011067	-0.01093784		No reference
5	3.967E-13	-1.91E-08	-9.701E-09	-5.9579E-09	No reference

1:(Stroud, 1982), 2: (James, Smith and Wolford, 1977)

V. APPENDIX (VISUAL BASIC PROGRAM)

```
Private Sub STARTMNU_Click()
ReDim AA(40, 100, 100), AANS(100, 100), ANS(100, 100)
ReDim MROW(100), MCOLUMN(100), MM(40, 100, 100), MMANS(100, 100), EMMANS(100, 100)
ReDim INVM(100, 200), INVAM(200, 200), INVRM(200, 200), INVABM(100, 200), A(200, 200), B(200,
200)
Dim VROW As Variant, VCOLUMN As Variant
Cls
FontSize = 11: OWUS = 0
2220    OWUS = 0: OW = 1
' THIS AREA IS FOR MATRIX INVERSION
'HERE IS THE INPUT FOR INVERSION
10      VROW = InputBox("WHAT'S THE NO. OF ROWS OF THIS MATRIX ?"): NR = 1 * VROW
If VROW = 0 Then Notice = InputBox("IT IS NOT POSSIBLE", "ROW OF MATRIX CAN'T BE ZERO",
"Click O.K. for me"): GoTo 10
20      VCOLUMN = InputBox("WHAT'S THE NO OF COLUMNS OF THIS MATRIX?")
If VCOLUMN = 0 Then Notice = InputBox("IT ISNOT POSSIBLE", "COLUMN OF MATRIX CAN'T BE
ZERO", "Click O.K. for me"): GoTo 20
If VROW <> VCOLUMN Then MsgBox (IMPOSSIBLE), , "IMPOSSIBLE" Else GoTo 2221
2221    For X = 1 To VROW
For Y = 1 To VROW
A(X, Y) = InputBox([Y], [X], "ENTER A")
Next Y
Next X
For X = 1 To VROW
For Y = 1 To VROW
B(X, Y) = InputBox([Y], [X], "ENTER B")
Next Y
Next X
T = 0
22555   For I = 1 To VROW
For J = 1 To VCOLUMN
INVM(I, J) = A(I, J) - T * B(I, J)
Next J
Next I
'THE INVERSE IS CARRIED OUT HERE
' THE PREAMBLE OF INVERSION
For I = 1 To VROW
For J = 1 To 2 * VCOLUMN
INVAM(I, J) = 0
Next J
Next I
For I = 1 To VROW
```

```
For J = 1 To 2 * VCOLUMN
INVAM(I, J) = INVAM(I, J) + INVM(I, J)
Next J
Next I
For I = 1 To VROW
INVAM(I, I + VCOLUMN) = INVAM(I, I + VCOLUMN) + 1
Next I
For I = 1 To VROW
For J = 1 To VCOLUMN
INVABM(I, J) = INVAM(I, J)
Next J
Next I
ZZ = 1
'THIS IS THE PLACE FOR INVERSION PROPER
For I = 1 To VROW
OWUSS = INVAM(I, I)
3333    If OWUSS > -0.0001 And OWUSS < 0.0001 Then GoTo 2222
For J = 1 To 2 * VCOLUMN
INVAM(I, J) = INVAM(I, J) / OWUSS
Next J
For J = 1 To VROW
If (J = I) Then GoTo 77777
OWUSS = INVAM(J, I)
For K = 1 To 2 * VCOLUMN
INVAM(J, K) = INVAM(J, K) - OWUSS * INVAM(I, K)
Next K
77777   Next J
Next I
'  If T > 40 Then GoTo 1111111
T = T + 0.0001
GoTo 22555
2222    ' HERE IS THE PLACE INTERCHANGE OF ROWS
If I + ZZ = 3 * VROW Then GoTo 1111111: 'MsgBox (IMPOSSIBLE), , "THIS MATRIX HAS NO
INVERSE":
For W = 1 To 2 * VCOLUMN
INVRM(I, W) = INVAM(I, W): INVRM(I + ZZ, W) = INVAM(I + ZZ, W)
INVAM(I, W) = INVRM(X + ZZ, W): INVAM(I + ZZ, W) = INVRM(I, W)
Next W
OWUSS = INVAM(I, I)
If OWUSS = 0 Then ZZ = ZZ + 1: GoTo 6666
Z = 1: GoTo 3333
6666    For W = 1 To 2 * VCOLUMN
INVRM(I, W) = INVAM(I, W): INVRM(I + ZZ, W) = INVAM(I + ZZ, W)
INVAM(I, W) = INVRM(I + ZZ, W): INVAM(I + ZZ, W) = INVAM(I, W)
Next W
If I + ZZ = 3 * VROW Then: GoTo 1111111: 'MsgBox (IMPOSSIBLE), , "THIS MATRIX HAS NO
INVERSE":
ZZ = ZZ + 1: GoTo 2222
'this is the end of inversion
1111111
Print "RESULT"
Print "   H  =  "; Format(T, "0.###0");
If OW = NR Then GoTo 1111112
OW = OW + 1: T = T + 0.1
GoTo 22555
1111112
WWW = InputBox(" Press OK or Cancle to Stop")
End Sub
```

REFERENCES

[1] Arnoldi, W. E. (1951). The principle of minimized iteration in the solution of the matrix eigenvalue problem.Quarterly of Applied Mathematics, 9: 17– 29.

[2] Arbenz, P., and Geus, R. (2005). Multilevel preconditioned iterative eigensolvers for maxwell eigenvalue problems. Applied numerical mathematics, 54(2): 107 – 121.

[3] Bai, Z. D. J., Dongarra, J. R. A., and Van-der Vorst, H. (2000). Templates for the Solution of Algebraic Eigenvalue Problems - A Practical Guide.Philadelphia, PA:SIAM.

[4] Barrett, R.; Berry, M.; Chan, T. F.; Demmel, J.; Donato, J.; Dongarra, J.; Eijkhout, V.; Pozo, R.; Romine, C.; and Van-der Vorst, H. A. (1994). Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd ed.Philadelphia, PA:SIAM.

[5] Chatelin, F. (1993). Eigenvalues of matrices. Chichester,West Sussex: JohnWiley&Sons Ltd.

[6] Demmel, J. W. (1997). Applied Numerical Linear Algebra. Philadelphia, PA:SIAM.

[7] Freitag, M. (2007). Inner-outer Iterative Methods for Eigenvalue Problems - Convergence and Preconditioning (Unpublished doctoral dissertation). University of Bath.

[8] Fullard, K. (1980). The Modal Method for Transient Response and its Application to Seismic Analysis. In Donea, J. (ed.), Advanced Structural Dynamics (pp.43-70).

[9] Geradin, M. (1980). Variational Methods of Structural Dynamics and their Finite Element Implementation.In Donea, J. (ed.), Advanced Structural Dynamics(pp.1-42)

[10] Greenstadt, J. (1960). Mathematical Methods for Digital Computers. Vol. 1(ed. A. Ralston and H. S. Wilf). New York: John Wiley & sons. Pp. 84-91.

[11] Hochstenbach, M. E. and Notay, Y. (2004). The Jacobi-Davidson method, GAMM Mitt.

[12] James, M. L., Smith, G. M. and Wolford, J. C. (1977). Applied Numerical Methods for Digital Computation: withFORTRAN and CSMP. New York: Harper and Row Publishers.

[13] Key, S. W. (1980). Transient Response by Time Integration: Review of Implicit and Explicit Operators. In Donea, J. (ed.), Advanced Structural Dynamics.

[14] Key, S. W. and Krieg, R. D. (1972) Transient Shell Response by numerical Time Integration. 2nd US-JAPAN Seminar on Matrix Methods in Structural Analysis. Berkeley, California.

[15] Lanczos, C. (1950). An iterative method for the solution of the eigenvalue problem of linear differential and integral operators.Journal of Research of the National Bureau of Standards, 45: 255–282.

[16] Ortega, J. (1967). Mathematical Methods for Digital Computers. Vol. 2(ed. A. Ralston and H. S. Wilf). New York: John Wiley & sons. Pp. 94-115.

[17] Owen, D. R. J. (1980). Implicit Finite Element Methods for the Dynamic Transient Analysis of Solids with Particular Reference to Non-linear Situations. In Donea, J. (ed.), Advanced Structural Dynamics (pp. 123-152).

[18] Paz, M. (1980). Structural Dynamics: Theory and Computation. New York: Litton Education Publishers.

[19] Quillen, P. and Ye, Q. (2010). A block inverse-free preconditioned Krylov subspace method for symmetric generalized eigenvalue problems. Journal of Computational and Applied Mathematics, 233 (5): 1298-1313

[20] Sheikh, A. H.; Haldar, S.; and Sengupta, D. (2004). Free flexural vibration of composite plates in different situations using a high precision triangular element. Journal of Vibration and Control, 10 (3): 371-386.

[21] Sleijpen, G. L. G., and Van-der Vorst, H. A. (1996). A Jacobi-Davidson iteration method for linear eigenvalue problems.SIAM J. Matrix Anal. Appl., 17: 401–425.

[22] Stroud, K. A.(1982). Engineering Mathematics, 2nd ed. London: MacMillan Press.

[23] Trefethen, L. N., and Bau, D. I. (1997). Numerical Linear Algebra.Philadelphia, PA: SIAM.

[24] Wilkinson, J. H. (1965).The Algebraic Eigenvalue Problem.Oxford, UK: Oxford University Press.