Paper No. 216, Bull. ISET, Vol. 18, No. 3, Sept. 1981, pp. 114–124

ECONOMICAL SOLUTION OF BOUNDARY VALUE PROBLEMS

V.H. JOSHI[1] AND J.J. EMERY[2]

INTRODUCTION

Safe and economical aseismic design of important life line structures is very important in providing post-disaster relief to earthquake affected communities. As such, efficient storage schemes and econmical solution techniques for solving such problems are of great engineering interest, especially for problems which are too large to fit into present day computers.

The skyline method of storage of symmetric banded matrices is discussed else where (Bathe and Wilson, 1976). However, such methods are not available for matrices that are not square. Besides, full advantage of boundary value problems can be realized only if the matrices can be explicitly partitioned into submatrices corresponding to interior and boundary displacement freedoms. More economical storage schemes for square and nonsquare matrices are desirable.

For boundary value dynamic problems the current day practice obtains the net dynamic response for interior displacement freedoms as the algebraic sum of component responses, namely, the pseudostatic and the balancing dynamic response (Clough, 1971; Nair, 1974; Wolf, 1977). Since component responses are individually of not much interest a method directly obtaining the net dynamic response (used in computing stresses and strains of engineering interest) is desirable.

For problems with ill codditioned matrices, Householder factorization is desirable. But it increases the demand for central memory of the computer. Use of square roots in Choleski factorization results into unacceptable errors in computed results. Gaussian elimination avoids square roots. But it alters the force vector resulting in wastage of computational effort in certain analyses (Joshi, 1980). A procedure combining advantages (and free from disadvantages) of Choleski and Gaussian methods is desirable.

Through out this presentation, symbols explained in the Appendix-A are used. As such, they are not always explained as and when they appear in the presentation.

SKYLINE METHOD OF STORAGE FOR SYMMETRIC MATRICES

For a given problem individual bandwidths for columns of the stiffness matrix can be obtained. Because of the symmetry, it is enough to store the elements on the leading diagonal and above. If leading zeros in each column are excluded, the resulting profile is called

1. Earthquake Engineering Department, University of Roorkee, Roorkee, U.P., India
2. Civil Engineering Department, McMaster University, Hamilton, Canada, L8S4L7.

the skyline. All elements under this profile are stored in a single dimension array. For finding addresses of elements two auxiliary arrays are used. The first array LT stores the addresses of diagonal elements. The second array LA stores the row number of the first nonzero element for each column. Further details about this method are presented else where (Bathe and Wilson, 1976). The advantage of this method is the use of same auxiliary arrays for finding addresses of the parent matrix and the $[L]^T$ matrix of its factorization by Choleski or Gaussain method or that proposed by Joshi (1981).

BASE LINE METHOD OF STORAGE FOR SYMMETRIC MATRICES

If elements of the leading diagonal and below of the stiffness matrix are stored, excluding zero elements below the last nonzero element in each column, the resulting profile may be called as the base line. All elements on the leading diagonal, the base line and those in between are stored in single dimension array. Addresses of these elements are obtained by using two auxiliary ar[illegible]s similar to those explained before. In this scheme if any zero element below corresponding element [illegible] [L] matrix of Choleaski or Guassian factorization will be a nonzero element. This lead[illegible]ifferent base line profile for the [L] matrix obtained after factorization. Additional pair of auxiliary arrays have to be formed for finding addresses of elements of the [L] matrix.

In solving a set of equations by the forward and backward substitution process, elements of the [L] matrix are once scanned row wise and once column wise. The auxiliary arrays explained before require two additions/subtractions to obtain the address of an element for rowwise scanning If the first array LT is formed in such a way that LT (I) + J gives the address of the element on I^{th} row and J^{th} column, only one addition is needed for finding this address. Such auxiliary arrays can be formulated for the parent matrix and that obtained after its factorization for the skline as well as base line method of storage.

With the system of two auxiliary arrays explained so far, the number of elements to be scanned in each row for the rowwise scanning is equal to the number of elements in the largest half bandwidth for that matrix. This is a wasteful procedure if many columns of large half bandwidth are followed by a number of columns of relatively smaller half bandwidths. This is avoided if a third auxiliary array LN is formed which stores the individual bandwidth for each row. Similar scheme can be deviced for the base line method of storage scheme also. This together with the scheme explained in the previsous paragraph can help to economize the computational effort particularly for dynamic analysis in time domain.

CHOICE OF THE MOST ECONOMICAL STORAGE SCHEME

For most matrices under consideration, skyline and base line methods of storages are much more economical compared to that for the method which adopts the largest half bandwidth for all the columns. Choice of the storage scheme is governed by the size of the [L] matrix and the stiffness matrix. Storage for the auxiliary arrays is considered to be negligible which is reasonable when the number of displacement freedoms is large.

Consider a parent matrix with many columns of large half band width and each of them followed by many columns of much smaller half band widths. Examine the elements on the leading diagonal, the base line and those in between. If these long columns have many zero elements (particularly near the lower end) the skyline method is more economical. If these columns contain very few zero elements, the base line methods is more economical. Similar observations may be made for the [L] or [L]^T matrices also. But the matrices are seldom available for visual inspection. The best way is to workout storage by the two methods and choose the method that is more economical. Such a check is quite inexpensive.

From the matrices given in eq. 1 and 2 it may be observed that for the dynamic problem where parent matrix as well as the matrix derived form it after factorization are required to be stored in the central memory, the skyline method is more economical for the matrices cited in eq. 1 where as the base line method is more economical for the matrices cited id eq. 2.

$$[k]=\begin{bmatrix} x & x & x & x & x & 0 \\ x & x & 0 & 0 & 0 & 0 \\ x & 0 & x & 0 & 0 & 0 \\ x & 0 & 0 & x & x & 0 \\ x & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix} \qquad [L]=\begin{bmatrix} x & 0 & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ x & x & x & 0 & 0 & 0 \\ x & x & x & x & 0 & 0 \\ x & x & x & x & x & 0 \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix} \qquad (1)$$

Skyline profile — Base line profile

Storage requirement is 17 locations by skyline method and 12 by base line method

Storage requirement is 17 locations by base line method

In eq. (1) and (2), x stands for a nonzero element.

$$[k]=\begin{bmatrix} x & x & 0 & 0 & x & 0 \\ x & x & x & 0 & 0 & 0 \\ 0 & x & x & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 \\ x & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad [L]=\begin{bmatrix} x & 0 & 0 & 0 & 0 & 0 \\ x & x & 0 & 0 & 0 & 0 \\ 0 & x & x & 0 & 0 & 0 \\ 0 & 0 & 0 & x & 0 & 0 \\ x & x & x & x & x & 0 \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Storage requirement is 13 locations by skyline method and 12 locations by base line method

Storage requirement is 17 locations by base line method.

BOUNDARY VALUE PROBLEMS IN STATIC ANALYSES

The equation of equilibrium is given by:

$$\left[\begin{array}{c|c} k_{11} & k_{12} \\ \hline k_{21} & k_{22} \end{array}\right] \left\{\begin{array}{c} u_i \\ \hline u_b \end{array}\right\} = \left\{\begin{array}{c} R_1 \\ \hline R_2 \end{array}\right\} \quad \ldots(3)$$

i.e. $$[k_{11}]\{u_i\} = \{R_1\} - [k_{12}]\{u_b\} \quad \ldots(4)$$

and $$[k_{12}]\{u_i\} + [k_{22}]\{u_b\} = \{R_2\} \quad \ldots(5)$$

Here the stiffness matrix, the force vector, {R}, and the displacement vector, $\{u_b\}$, are assumed to be known. Therefore, eq (5) can be solved by using forward and backward substitution process to obtain the unknown displacement vector, $\{u_i\}$. When the scheme of numbering the displacement freedoms intermixes the interior and boundary displacements, the explicit partitioning cited in eq. (3) is not possible. Hence, full advantage of the partitioning can not be realized in reducing the storage requirement. When all interior displacement freedoms are numbered in one sequence and all the boundary displacement next to them, explicit partitioning cited in eq. (3) is possible. In such a case, it is enough to store the $[k_{11}]$ and $[k_{12}]$ submatrices. This results into significant economy in storage requirment, particularly if the boundary displacement freedoms are large in number. The $[k_{22}]$ matrix is stored only if vector $\{R_2\}$ is also to be computed. In problems such as soil structure interaction, the boundary is far away from the structure and the reaction $\{R_2\}$ may not be of interest. For such cases only $[k_{11}]$ and $[k_{12}]$ matrices need be stored. $[k_{12}]$ is not a square matrix.

Storage for the $[k_{11}]$ matrix can be further economized by minimizing the bandwidth for interior displacement freedoms only. The principal minor of a positive definite matrix is also a positive definite matrix. So it is possible to factorize the $[k_{11}]$ matrix by the Gaussian or Choleski method.

In the proposed displacement freedom numbering scheme, individual bandwidths for boundary displacement freedoms are greatly incressed by choice. These columns are contained in the $[k_{12}]$ matrix which is scanned row wise in solving eq. (4). In most problems very few interior displacement freedoms are connected to boundary displacement freedoms. Therefore most rows of $[k_{12}]$ matrix are filled with zero elements. Such rows may be eliminated from the storage scheme. An oxillary array NII may be formed to identify those displacement freedoms which are connected to one or more of the boundary displacement freedoms. For each such row, the column number with the first nonzero element is stored in the NFC array, the column number for the last nonzero element is stored in the NLC array and the address of the first nonzero element is stored on the NHM array. All zero elements between the first and the last nonzero elements of such rows are also included in the storage. Experience with such problems indicates that the storage requirement for the auxiliary arrays NII, NFC, NLC, NHM and for the submatrix $[k_{12}]$ itself is quite insignificant compared to the size of $[k_{11}]$ matrix. This scheme results into considerable simplification of the logic of the computer program and significant saving in the cost of computation.

STATE OF ART FOR SOLVING DYNAMIC BOUNDARY VALUE PROBLEMS

The equation of motion in partitioned form is given by:

$$\left[\begin{array}{c|c} m_{11} & m_{12} \\ \hline m_{21} & m_{22} \end{array}\right] \left\{\begin{array}{c} \ddot{u}_i^t \\ \hline \ddot{u}_b^t \end{array}\right\} + \left[\begin{array}{c|c} c_{11} & c_{12} \\ \hline c_{21} & c_{22} \end{array}\right] \left\{\begin{array}{c} \dot{u}_i^t \\ \hline \dot{u}_b^t \end{array}\right\} + \left[\begin{array}{c|c} k_{11} & k_{12} \\ \hline k_{21} & k_{22} \end{array}\right] \left\{\begin{array}{c} u_i^t \\ \hline u_b^t \end{array}\right\} = \left\{\begin{array}{c} 0 \\ 0 \end{array}\right\} \qquad \ldots(6)$$

Here, external forces are assumed to be absent. When present, they appear on the right hand side (RHS) of eq. (6). According to the method suggested by Clough (1971) the net dynamic displacement $\{u_i^t\}$ is the algebraic sum of component responses, namely, the pseudo static response $\{u_i^s\}$ (that keeps the system in static equilibrium under the imposed net boundary response) and the balancing dynamic response $\{u_i^d\}$. The boundary response is assumed to be known. As such, only equations above the horizontal partition need be solved. From principles of the static equilibrium $\{u_i^s\}$ can be obtained as:

$$\{u_i^s\} = -[k_{11}]^{-1}[k_{12}]\{u_b^t\} = [H]\{u_b^t\} \qquad \ldots(7)$$

where

$$[H] = -[k_{11}]^{-1}[k_{12}] \qquad \ldots(8)$$

It may be further assumed that the system has Rayleigh damping, expressed as:

$$\left[\begin{array}{c|c} c_{11} & c_{12} \\ \hline c_{21} & c_{22} \end{array}\right] = \alpha \left[\begin{array}{c|c} m_{11} & m_{12} \\ \hline m_{21} & m_{22} \end{array}\right] + \beta \left[\begin{array}{c|c} k_{11} & k_{12} \\ \hline k_{21} & k_{22} \end{array}\right] \qquad \ldots(9)$$

where α and β are constants of proportionality. On substituting Rayleight damping and pseudostatic respanse in eq. (6) and simplifying, the equations above the horizontal partition line may be expressed as

$$[m_{11}]\{\ddot{u}_i^d\} + [c_{11}]\{\dot{u}_i^d\} + [k_{11}]\{u_i^d\}$$

$$= -[m_{11}][H]\{\ddot{u}_b^t\} - \alpha[m_{11}][H]\{\dot{u}_b^t\} \qquad \ldots(10)$$

On the right hand side the damping terms being insignificant compared to inertia terms may be neglected. This equation may be solved for the balancing dynamic response $\{\ddot{u}_i^d\}$. The details of this method are documented else where (Clough, 1971; Nair, 1974; Wolf, 1977 and Joshi, 1980).

PROPOSED METHOD

Equation (6) above the horizontal partition may be expressed as :

$$[m_{11}]\{\ddot{u}_i^t\} + [c_{11}]\{\dot{u}_i^t\} + [k_{11}]\{u_i^t\} = -[m_{12}]\{\ddot{u}_b^t\} - [c_{12}]\{\dot{u}_b^t\} - [k_{12}]\{u_b^t\} \qquad \ldots(11)$$

All the terms on the RHS being known, eq. (11) can be solved for obtaining net dynamic response directly.

TRIAL PROBLEM

A concrete lined horizontal circular tunnel of 6.1m diameter and 19.5m length situated within a single homogenous and isotropic soil layer of depth 53.4m overlying the base rock which is at a distance of 26m from the axis of tunnel. The base excitation was a sinusoidal acceleration function of frequency 2Hz and a peak amplitude of $\pm 0.3 m/s^2$ in the axial as well as transverse direction. Rayleigh waves were not considered. Vertical propagation of shear waves was assumed. The dynamic analysis in the time domain was carried out using a time step of 0.01 s. Along transverse boundaries of the problem free field motions at corresponding positions with out the presence of the tunnel were assumed. Further details regarding this problem are discussed else where (Joshi, 1980). The finite element idealization of the problem cited Fig. 1 is as shown in Fig. 2.

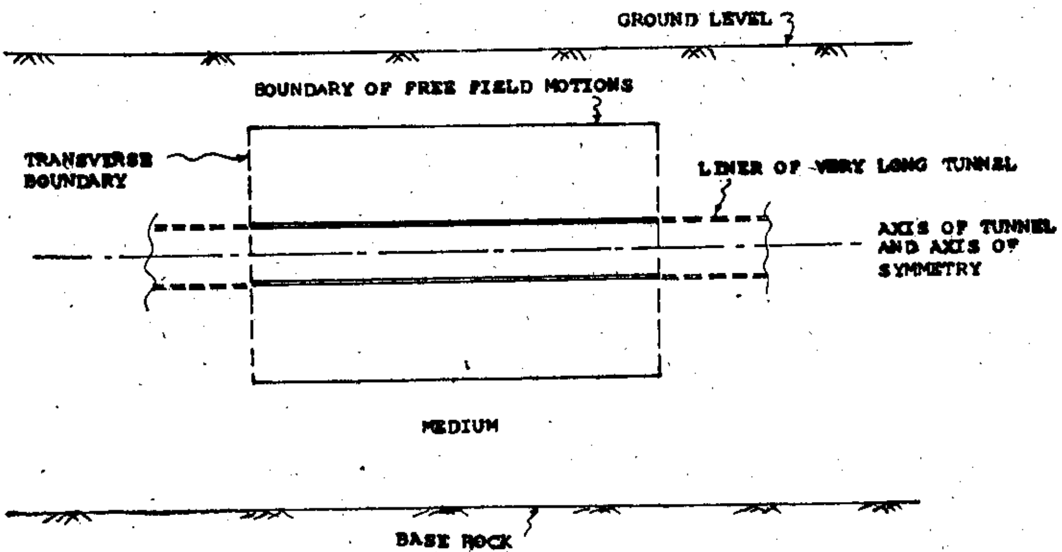
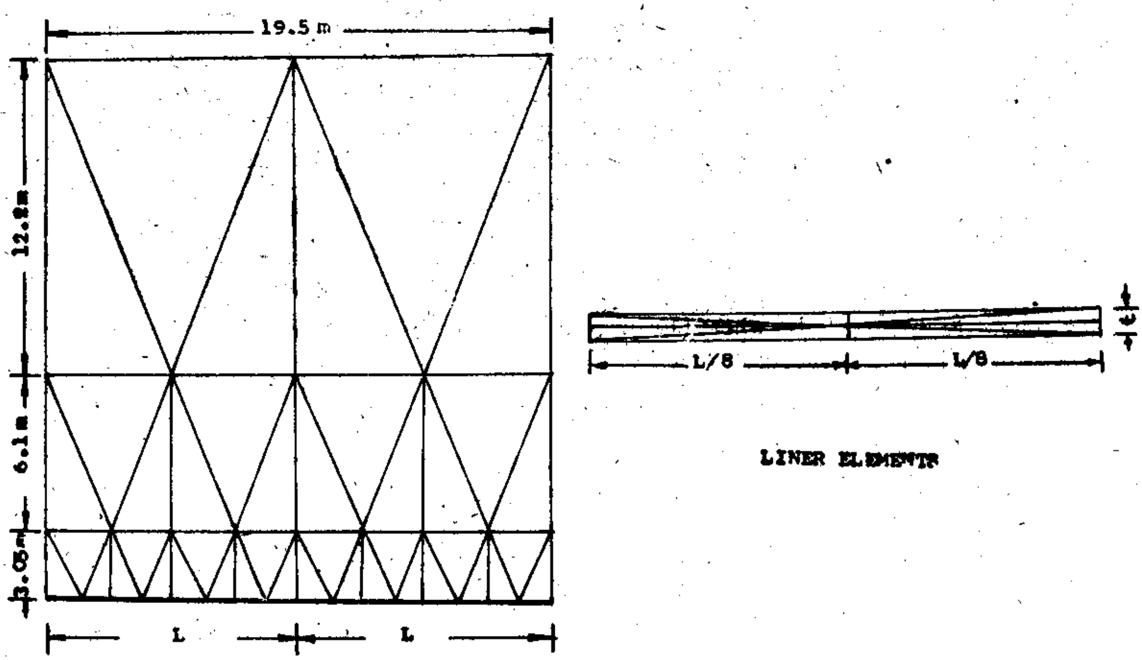


Fig. 1. Finite Portion of a Long Axisymmetric Interactive System

The results of the analyses by the first method (proposed by Clough, 1971) and by the proposed method (the second method) were identical for the first six digits for all significant responses. Discrepancies were noticeable only when the response was insignificantly small compared to significant responses which is considered to be adequate for all engineering purposes. Besides, some discre, ancy is expected because damping terms on the RHS are neglected in the first method which is not the case in the second method. These details are discussed in greater details else whese (Joshi, 1980; Joshi and Emery, 1981).

ADVANTAGES OF THE PROPOSED METHOD

For the first method, the size of the [H] matrix required is comparable to that of the $[k_{11}]$ matrix in many problems and contains few zero elements. In comparison, the proposed



ELEMENTS FOR THE MEDIUM

t = Thickness of Liner = 0.254m

Fig. 2. Finite Element mesh For Tunnel-Soil System

method needs $[k_{12}]$, $[c_{12}]$ and $[m_{12}]$ which are very small matrices. They can be further reduced with assumption of Rayleigh damping and lumped mass matrix.

All the effort required in computing pseudostatic response and for summation of component responses needed for the first method are avoided in the proposed method which leads to a significant saving in computational effort. The reduction in the compilation time for the dynamic analysis represents the significant simplification of the programme logic as indicated in the Table 1. For both these methods, explicit partitioning of the stiffness matrix was employed as discussed before. If the half bandwidth type of conventional storage scheme is to be employed and the interior and boundary displacements are intermixed in minimizing the bandwidth of the stiffness matrix, the storage requirements and the (estimated) computational effort for this case (the third method) are considerably more than the corresponding requirements for the proposed method as shown in Table 1.

Though actual savings in storage and computational effort vary from problem to problem, the trial analysis indicates that significant savings could be achieved by proposed (i.e. the second) method. The larger the ratio of number of boundary degree of freedoms to interior degrees of freedom, the greater are such savings.

Table 1. Storage and Computation Requirements for the Trial Problem.

Item	Method 1	Method 2	Method 3	Percentage Savings of (3) with respect to (2)	Percentage Savings of (3) with respect to (4)
(1)	(2)	(3)	(4)	(5)	(6)
Half Bandwidth	45	45	51	—	13
$[k_{11}]$ matrix	2733	2733	10404	—	280
[F] matrix	3777	3777	10404	—	175
$[k_{11}]^{-1}$ $[k_{12}]$ matrix	6435	—	7956	936	7856
$[k_{12}]$ matrix	—	621			
Total size of the program	24576	19770	44544	26	128
Compilation time	6.33 sec	1.16 sec	—	—	—
Execution time	94–96 sec	57.08 sec	273 sec (estimated)	66.36	378.27

$$[F]=\left(\left[m_{11}\right]+\frac{\Delta t}{2}\left[c_{11}\right]\quad\frac{\Delta t}{6}\left[k_{11}\right]\right)^{-1} \text{ for method 1 and method 2}$$

$$=\left(\left[\begin{array}{c|c} m_{11} & m_{12} \\ \hline m_{22} & m_{22}\end{array}\right]+\frac{\Delta t}{2}\left[\begin{array}{c|c} c_{11} & c_{12} \\ \hline c_{22} & c_{22}\end{array}\right]-\frac{\Delta t}{6}\left[\begin{array}{c|c} k_{11} & k_{12} \\ \hline k_{22} & k_{22}\end{array}\right]\right)^{-1} \text{ for method 3.}$$

Δt=time interval for the step by step method dynamic analysis in time domain.

FACTORIZATION OF POSITIVE DEFINITE MATRICES

In many static and dynamic analyses it is required to solve equations of the type

$$[k]\{u\}=\{R\} \qquad \ldots(12)$$

For solving eq. (12) the [k] matrix is factorized into upper and lower triangular matrices so that the equation may be solved by the forward and backward substitution. When the [k] matrix is ill conditioned Householder method is ideal but results into increased storage requirements. The Choleski factorization involves the use of square roots which results into unacceptably large errors in computed results. Gaussian factorization avoids square roots, but alters the load vector {R} which results into waste of computational effort in certain analyses such as dynamic analyses in time domain.

However, advantages of the Choleski and Gaussian methods can be combined by the factorization suggested by Joshi (1981). The elements of the matrices of $[L][D][L]^T$ factorization by this method may be obtained from the following expressions which subscripts i and j refer to row/column numbers of these elements.

$$
\begin{aligned}
d_{11} &= k_{11} \\
l_{ii} &= 1 \\
l_{i1} &= k_{i1}/d_{11} \\
d_{ii} &= k_{ii} - \sum_{m=1}^{i-1} l_{im}^2 \, d_{mm} \\
l_{ij} &= \left(k_{ij} - \sum_{m=1}^{j-1} l_{im} \, l_{jm} \, d_{mm}\right) \Big/ d_{jj}
\end{aligned}
\qquad \ldots (13)
$$

Because it avoids use of square roots the errors associated with the same are avoided. The factorization leaves the force vector in tact. So, the wastage of the computation effort in reducing this force vector associated with Gaussian method is also eliminated. This greater degree of accuracy helps to reduce the time step required for the dynamic analyses in time domain when compared with that if Choleski factorization is employed. This results into economy in execution time required. Further details regarding the same are discussed else where (Joshi, 1981).

CONCLUSIONS

Explicit partitioning of matrices corresponding to interior and boundary displacement degrees of freedom can be of use in reducing the size of computer central memory storage and the cost of computation for certain types of boundary value problems. If all displacement freedoms for which the displacements are not known are numbered in one sequence and all boundary displacement freedoms with known displacements are numbered after wards, such explicit partitioning is possible.

It is possible to device economical storage schemes for symmetric matrices by the skyline and the base line storage methods. Auxiliary arrays are suggested to further economize the computational effort required for finding addresses of elements. A method of finding the most economical scheme of storage for different problems is also indicated.

A method of solving the boundary value problem that is more economical compared to the present day practice is explained. The use of the suggested method of factorization is helpful in increasing the accuracy of the computed results.

BIBLIOGRAPHY

1. Bathe, K.J. and Wilson, E.L. (1976), "Numerical Methods in Finite Element Analysis", Prentice-Hall Inc., New Jersey.

2. Clough, R. W. (1971), "Analysis of Structural Vibrations and Dynamic Response", Recent Advances in Matrix Methods of Structural Analysis and Design, University of Alabama Press, Huntsville, pp 441–486.

3. Joshi, V.H. (1960), "Seismic Analysis of Under Ground Openings", Ph.D. Thesis, McMaster University, Hamilton, Ont., Canada.

4. Joshi, V.H. (1981), "Factorization of Symmetric Positive Definite Matrices", sent for publication to International conf. on Numerical Methods for Coupled Problems, to be held in Swansea, U.K. in Sept. 1981.

5. Joshi, V.H. and Emery, J.J. (1881), "Solving Equations of Motion for Boundary Value Problems", Sent for publication to International Conf. on Numerical Methods for Coupled Problems, to be held at Swansea, U.K. in Sept. 1981.

6. Nair, G.P. (1974), "Response of Soil Pile Systems to Seismic Waves", Ph.D. Thesis, McMaster University, Hamilton, Ont., Canada.

7. Wolf, J.P. (1977), "Seismic Response due to Travelling Shear Wave Including Soil-Structure Interaction with Base Mat Uplift", Journal of Earthquake Engineering and Structural Dynamics, Vol. 5, pp 337–363.

APPENDIX A

SYMBOLS USED IN THE PRESENTATION

b	Number of boundary displcement freedoms
i	Number of interior displacement freedoms
$n=i+b$	Total number of displacement freedom
$[m]=\left[\begin{array}{c\|c} m_{11} & m_{12} \\ \hline m_{21} & m_{22} \end{array}\right]$	Mass matrix of the system
$[c]=\left[\begin{array}{c\|c} c_{11} & c_{12} \\ \hline c_{21} & c_{22} \end{array}\right]$	Damping matrix of the system
$[k]=\left[\begin{array}{c\|c} k_{11} & k_{12} \\ \hline k_{21} & k_{22} \end{array}\right]$	Stiffness matrix of the system
$[m_{11}], [c_{11}], [k_{11}]$	Square matrices of i rows and i columns
$[H]=-[k_{11}]^{-1}[k_{12}]$	A matrix used in the solujon of the equation of motion
$[m_{12}], [c_{12}], [k_{12}]$	Matrices of i rows and b columns
$\{\ddot{u}_i'\}, \{\dot{u}_i'\}, \{u_i'\}$	Net acceleration, velocity and displacement vectors for interior displacement freedoms

$\{\ddot{u}_i^s\}, \{\dot{u}_i^s\}, \{u_i^s\}$	Corresponding pseudostatic vectors
$\{\ddot{u}_i^d\}, \{\dot{u}_i^d\}, \{u_i^d\}$	Corresponding balancing dynamic response vectors
$\{\ddot{u}_b^t\}, \{\dot{u}_b^t\}, \{u_b^t\}$	Net acceleration, velocity and acceleration vectors for the boundary displacement freedoms
$\{\bar{R}\}$	Force Vector
$\{R_1\}$	Force vector for the interior freedoms
$\{R_2\}$	Force vector for the boundary displacement freedoms
[D]	Diagonal matrix
d_{ii}	Elements of the [D] matrix
[L]	Lower triangular matrix obtained after factorization
$[L]^T$	Transpose of [L] matrix
k_{ij} , l_{ij}	Element on i^{th} row and j^{th} column of [k] matrix and [L] matirx respectively
α, β	Proportionality constants used in Rayleigh Damping