

Lehigh University Lehigh Preserve

Fritz Laboratory Reports

Civil and Environmental Engineering

1969

Inversion of tri-submatrix banded matrices, June 1969

H. Jr. Bill

C. N. Kostem

Follow this and additional works at: <http://preserve.lehigh.edu/engr-civil-environmental-fritz-lab-reports>

Recommended Citation

Bill, H. Jr. and Kostem, C. N., "Inversion of tri-submatrix banded matrices, June 1969" (1969). *Fritz Laboratory Reports*. Paper 1636. <http://preserve.lehigh.edu/engr-civil-environmental-fritz-lab-reports/1636>

This Technical Report is brought to you for free and open access by the Civil and Environmental Engineering at Lehigh Preserve. It has been accepted for inclusion in Fritz Laboratory Reports by an authorized administrator of Lehigh Preserve. For more information, please contact preserve@lehigh.edu.

INVERSION OF TRI-SUBMATRIX
BANDED MATRICES

by

Herbert L. Bill, Jr.
Celal N. Kostem

Department of Civil Engineering
Fritz Engineering Laboratory
Lehigh University
Bethlehem, Pennsylvania

June 1969

Fritz Engineering Laboratory Report No. 237.60

TABLE OF CONTENTS

	<u>Page</u>
1. INTRODUCTION	1
2. MATHEMATICAL FORMULATION	3
3. LOGICAL FLOW CHART	7
4. LIMITATIONS	8
5. INPUT	9
6. OUTPUT	10
7. STORAGE LOCATIONS	11
8. NUMBER OF BASIC OPERATIONS	13
9. SUMMARY	15
10. PROGRAM LISTING	16
11. FIGURES	23
12. BIBLIOGRAPHY	32
13. ACKNOWLEDGMENTS	33

1. INTRODUCTION

This report presents a methodology and the relevant computer program for the inversion of "tri-diagonal" matrices by making use of its inherent properties.

A tri-diagonal matrix is one which has elements on the main diagonal, and which has elements once removed from the main diagonal. Each element may be a matrix itself, but each element must have the same dimensions as every other element.

The tri-diagonal matrix must be square and non-singular, and each sub-matrix must be square and non-singular (Fig. 1).

The application of tri-diagonal inversion scheme to complex structures was described by B. E. Gatewood and Norik Ohanian (Ref. 1). This article points out that a truss, or a frame, can be arranged in sections so that members from each section attach only in the section, or to the two adjacent sections; then it is possible to generate the stiffness matrix directly in tri-diagonal sub-matrix form from the generalized force-displacement equations for each member and the equilibrium equations for each joint of the structure. The size of the sub-matrices is determined by the number of joints in the section and the number of displacement components for each joint.

In Gatewood and Ohanian's article, each sub-matrix could be of any arbitrary size. However, as was pointed out earlier, this

subroutine handles only the case of each sub-matrix being the same size. It should be noted that the largest matrix that has to be inverted by standard procedures is a matrix having the same dimensions as the sub-matrix.

As a demonstration of the formulation of a tri-diagonal matrix, a simple frame will be set up (Fig. 2) using the "Direct Stiffness" method and including axial effects.

Although the example is valid for use in the tri-diagonal matrix subroutine, in general, the ordinary formulation of the direct stiffness method does not permit the use of this subroutine. That is, if more than two members frame into a joint or if axial deformations are neglected.

The purpose of this study is not merely to invert a specific type of matrix, but to invert it efficiently making use of the known properties of symmetry.

It is known that the matrix is symmetric (Fig. 1) and, therefore, its inverse is also symmetric. For the matrix to be symmetric, the sub-matrices on the main diagonal must be symmetric; and the off-diagonal sub-matrices must equal their transposed counterparts

$$(A_{ij} = A_{ji}^T).$$

2. MATHEMATICAL FORMULATION

$$\underline{A} \underline{A}^{-1} = \underline{I}$$

$$\begin{bmatrix} \underline{K}_{11} & \underline{K}_{12} & \underline{0} & \underline{0} & \underline{0} & \text{-----} & \underline{0} \\ \underline{K}_{21} & \underline{K}_{22} & \underline{K}_{23} & \underline{0} & \underline{0} & \text{-----} & \underline{0} \\ \underline{0} & \underline{K}_{32} & \underline{K}_{33} & \underline{K}_{34} & \underline{0} & \text{-----} & \underline{0} \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ \underline{0} & & & & & & \underline{K}_{n-1,n-2} \quad \underline{K}_{n-1,n-1} \quad \underline{K}_{n-1,n} \\ & & & & & & \underline{K}_{n,n-1} \quad \underline{K}_{nn} \end{bmatrix} \cdot \begin{bmatrix} \underline{F}_{11} & \underline{F}_{12} & \text{-----} & \underline{F}_{1n} \\ \underline{F}_{21} & \underline{F}_{22} & & \\ & & & \\ & & & \\ & & & \\ & & & \\ \underline{F}_{n1} & \underline{F}_{n2} & \text{-----} & \underline{F}_{nn} \end{bmatrix} =$$

$$= \begin{bmatrix} \underline{I} & \underline{0} & \underline{0} & \text{-----} & \underline{0} \\ \underline{0} & \underline{I} & \underline{0} & & \\ \underline{0} & \underline{0} & \underline{I} & & \\ & & & & \\ & & & & \underline{I} & \underline{0} \\ \underline{0} & \text{-----} & & & \underline{0} & \underline{I} \end{bmatrix}$$

Writing the equations for the first row of $[\underline{I}]$,

$$\underline{K}_{11} \underline{F}_{11} + \underline{K}_{12} \underline{F}_{21} = \underline{I} \quad (a)$$

$$\underline{K}_{11} \underline{F}_{12} + \underline{K}_{12} \underline{F}_{22} = \underline{0} \quad (b)$$

$$\underline{K}_{11} \underline{F}_{13} + \underline{K}_{12} \underline{F}_{23} = \underline{0} \quad (c)$$

$$\underline{K}_{11} \underline{F}_{14} + \underline{K}_{12} \underline{F}_{24} = \underline{0} \quad (d)$$

⋮

$$\underline{K}_{11} \underline{F}_{1n} + \underline{K}_{12} \underline{F}_{2n} = \underline{0} \quad (e)$$

From Equation (a)

$$\underline{F}_{11} = \underline{K}_{11}^{-1} (\underline{I} - \underline{K}_{12} \underline{F}_{21}) = \underline{K}_{11}^{-1} - \underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{21}.$$

From Equation (b)

$$\underline{F}_{12} = - \underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{22} = \underline{F}_{21}^T$$

$$\underline{F}_{21} = (-\underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{22})^T.$$

From Equation (c)

$$\underline{F}_{13} = - \underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{23}.$$

From Equation (d)

$$\underline{F}_{14} = - \underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{24}.$$

And, in general, from Equation (e)

$$\underline{F}_{1n} = - \underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{2n}.$$

Define

$$\underline{T}_{12} = - \underline{K}_{11}^{-1} \underline{K}_{12}.$$

Then

$$\begin{aligned} \underline{F}_{11} &= \underline{K}_{11}^{-1} + \underline{T}_{12} \underline{F}_{21} = \underline{K}_{11}^{-1} + \underline{T}_{12} (-\underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{22})^T \\ &= \underline{K}_{11}^{-1} + \underline{T}_{12} (\underline{T}_{12} \underline{F}_{22})^T = \underline{K}_{11}^{-1} + \underline{T}_{12} \underline{F}_{22}^T \underline{T}_{12}^T \\ &= \underline{K}_{11}^{-1} + \underline{T}_{12} \underline{F}_{22}^T; \text{ and} \end{aligned}$$

$$\underline{F}_{12} = \underline{T}_{12} \underline{F}_{22}$$

$$\underline{F}_{13} = \underline{T}_{12} \underline{F}_{23}$$

$$\underline{F}_{14} = \underline{T}_{12} \underline{F}_{24}$$

$$\underline{F}_{1n} = \underline{T}_{12} \underline{F}_{2n}$$

1.

Writing the equations for the second row of $[I]$,

$$\underline{K}_{21} \underline{F}_{11} + \underline{K}_{22} \underline{F}_{21} + \underline{K}_{23} \underline{F}_{31} = \underline{0} \quad (a)$$

$$\underline{K}_{21} \underline{F}_{12} + \underline{K}_{22} \underline{F}_{23} + \underline{K}_{23} \underline{F}_{32} = \underline{I} \quad (b)$$

$$\underline{K}_{21} \underline{F}_{13} + \underline{K}_{22} \underline{F}_{23} + \underline{K}_{23} \underline{F}_{33} = \underline{0} \quad (c)$$

$$\underline{K}_{21} \underline{F}_{14} + \underline{K}_{22} \underline{F}_{24} + \underline{K}_{23} \underline{F}_{34} = \underline{0} \quad (d)$$

$$\underline{K}_{21} \underline{F}_{1n} + \underline{K}_{22} \underline{F}_{2n} + \underline{K}_{23} \underline{F}_{3n} = \underline{0} \quad (e)$$

Noting that

$$\underline{F}_{12} = -\underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{22}$$

as found from the solution of the first row, and substituting this values into Eq. (b) above

$$\underline{K}_{21} (-\underline{K}_{11}^{-1} \underline{K}_{12} \underline{F}_{22}) + \underline{K}_{22} \underline{F}_{22} + \underline{K}_{23} \underline{F}_{32} = \underline{I}$$

$$(-\underline{K}_{21} \underline{K}_{11}^{-1} \underline{K}_{12} + \underline{K}_{22}) \underline{F}_{22} = \underline{I} - \underline{K}_{23} \underline{F}_{32}$$

$$(\underline{K}_{21} \underline{T}_{12} + \underline{K}_{22}) \underline{F}_{22} = \underline{I} - \underline{K}_{23} \underline{F}_{32}$$

$$\underline{F}_{22} = (\underline{K}_{22} + \underline{K}_{21} \underline{T}_{12})^{-1} (\underline{I} - \underline{K}_{23} \underline{F}_{32})$$

$$\underline{F}_{23} = (\underline{K}_{23} + \underline{K}_{21} \underline{T}_{13})^{-1} - (\underline{K}_{23} + \underline{K}_{21} \underline{T}_{12})^{-1} \underline{K}_{23} \underline{F}_{32}$$

Noting that

$$\underline{F}_{13} = \underline{T}_{12} \underline{F}_{23}$$

as found from the solution of the first row, and substituting this value into Eq. (c) above

$$\underline{K}_{21} (\underline{T}_{12} \underline{F}_{23}) + \underline{K}_{22} \underline{F}_{23} + \underline{K}_{23} \underline{F}_{33} = \underline{0}$$

$$(\underline{K}_{21} \underline{T}_{12} + \underline{K}_{22}) \underline{F}_{23} = -\underline{K}_{23} \underline{F}_{33}$$

$$\underline{F}_{23} = -(\underline{K}_{21} \underline{T}_{12} + \underline{K}_{22})^{-1} \underline{K}_{23} \underline{F}_{33}$$

Define

$$\underline{T}_{23} = - (K_{22} + K_{21} \underline{T}_{12})^{-1} K_{23}$$

Then

$$\underline{F}_{22} = (K_{22} + K_{21} \underline{T}_{12})^{-1} + \underline{T}_{23} \underline{F}_{32} = (K_{22} + K_{21} \underline{T}_{12})^{-1} + \underline{T}_{23} \underline{F}_{23}^T$$

$$\underline{F}_{22} = (K_{22} + K_{21} \underline{T}_{12})^{-1} + \underline{T}_{23} \underline{F}_{33} \underline{T}_{23}^T; \text{ and}$$

$$\underline{F}_{23} = \underline{T}_{23} \underline{F}_{33}$$

$$\underline{F}_{24} = \underline{T}_{23} \underline{F}_{34}$$

$$\underline{F}_{2n} = \underline{T}_{23} \underline{F}_{3n}$$

2.

Following the same procedure for the other rows of [I],

$$\underline{F}_{m-1,m-1} = (K_{m-1,m-1} + K_{m-1,m-2} \underline{T}_{m-2,m-1})^{-1} + \underline{T}_{m-1,m} \underline{F}_{mm} \underline{T}_{m-1,m}^T$$

$$\underline{F}_{m-1,n} = \underline{T}_{m-1,m} \underline{F}_{m,n}$$

$$\underline{T}_{m-1,m} = - (K_{m-1,m-1} + K_{m-1,m-2} \underline{T}_{m-2,m-1})^{-1} K_{m-1,m}$$

which are the recurrence formulae for all but the first and last rows. For the first row (see Eq. (I)),

$$\underline{F}_{11} = K_{11}^{-1} + \underline{T}_{12} \underline{F}_{22} \underline{T}_{12}^T$$

$$\underline{F}_{1n} = \underline{T}_{12} \underline{F}_{2n}$$

$$\underline{T}_{12} = - K_{11}^{-1} K_{12}$$

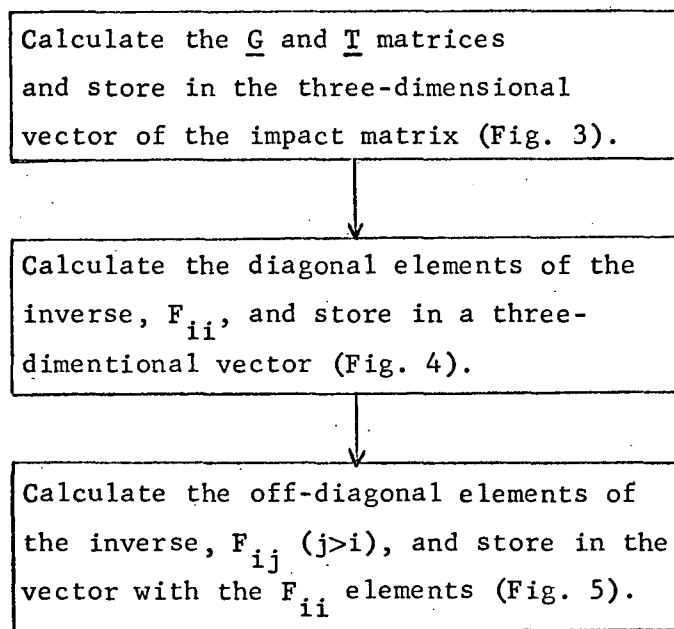
And, for the last row,

$$\underline{F}_{n,n} = (K_{n,n} + K_{n,n-1} \underline{T}_{n-1,n})^{-1}$$

3. LOGICAL FLOW CHART

From an examination of the recurrence formulae obtained from the derivation, it is obvious that two "sweeps" of the matrix are needed. The first sweep progresses down the three-dimensional vector calculating the T and G matrices. The second sweep progresses up the three-dimensional vector calculating the F matrices.

Below is shown a logical flow chart for the program. It will be noted that the second sweep has been broken down into two steps.



4. LIMITATIONS

The limitations of the program, from a mathematical point of view, are those of symmetry and banding. The limitations of the program, from a practical point of view, are the capacity of the machine and the size of the sub-matrices.

The program has been written for a maximum sub-matrix size of nine elements. This dimension can be changed by changing the DIMENSION statement in the subroutine and by changing the value of the variable JOB. It is not advisable to increase the value of JOB much beyond the value needed for a particular job (run) because this only serves to tie up core storage with zeros.

The limitation of the machine capacity (core storage) can be overcome by using disk storage.

It should also be noted that the form, in which the inverted matrix is returned to the calling program, is such that the resulting inverted matrix is not suitable for further mathematical operation- it is return as a three-dimensional vector rather than a two-dimensional array. The form of the three-dimensional vector is illustrated by the form of the output from the subroutine.

5. INPUT

Input to the subroutine consists of the banded matrix K stored as a three-dimensional vector with the K_{ii} 's stored in the odd locations and the K_{ij} 's (where $j>i$) stored in the even locations.

$$\begin{bmatrix} K_{11} \\ K_{12} \\ K_{22} \\ K_{23} \\ \vdots \\ K_{nn} \end{bmatrix} ;$$

N , the number of K_{ii} 's in the banded matrix; M , the actual size of each sub-matrix; MN , M times N ; $2N-1$, 2 time N minus 1; IO , number of output unit; and $JKKJ, \sum_{i=1}^N i$. (See Fig. 6 for the input matrix used for the test run of this program).

6. OUTPUT

The output is only a listing of the upper triangle of the inverted matrix. It is only a listing of the three-dimensional vector. The individual sub-matrices are labeled to show their position in the assembled two-dimensional inverted matrix. (See Fig. 7 for the output from the test run of this program).

7. STORAGE LOCATIONS

The storage locations required to store the total, unaltered banded matrix is $M^2 \times N^2$. The storage required for this subroutine can be broken down as follows.

The original matrix can be stored as a three-dimensional vector composed of only the upper triangular portion of the original banded matrix. This will require N storage locations for the K_{ii} 's and $N-1$ storage locations for the K_{ij} 's ($j>i$). Therefore, the total storage locations are used to store the \underline{T} and \underline{G} matrices calculated in the subroutine.

The inverted matrix requires $0.50 \times M^2 \times N^2$ storage locations if stored element by element. However, this subroutine requires $M^2 \times \sum_{i=1}^N$ storage locations for the inverted matrix.

Therefore the total storage required (not including "scratch" matrices) is

$$S = (2N - 1) M^2 + \frac{1}{2} N (N + 1) M^2 = \frac{M^2}{2} (N^2 + 5N - 2)$$

as composed to $M^2 N^2$ for storing just the total original banded matrix.

For comparative purposes with M = 2,

<u>N</u>	<u>S</u>	<u>M² N²</u>	<u>Ratio</u>
1	8	4	2.00
2	24	16	1.50
3	44	36	1.22
4	68	64	1.06
5	96	100	0.96
6	128	144	0.89
7	164	196	0.84
8	204	256	0.80
9	248	324	0.77
10	296	400	0.74

And with M = 4

<u>N</u>	<u>S</u>	<u>M² N²</u>	<u>Ratio</u>
1	32	16	2.00
2	96	64	1.50
3	176	144	1.22
4	272	256	1.06
5	384	400	0.96
⋮	⋮	⋮	⋮
100	83,984	160,000	0.52

8. NUMBER OF BASIC OPERATIONS

Basic operations, as considered here, are matrix addition, multiplication, inversion and transposition. Basic operations does not include the transfer back and forth from a two-dimensional array to part of a three-dimensional vector.

Strictly speaking the matrix inversion is not a basic operation because it involves some of the other basic operations. However, in order to arrive at an approximate figure, matrix inversion will be considered a basic operation.

The number of basic operations will be computed for each block of calculation as shown in the Logical Flow Chart.

1. For calculating the T and G matrices,

(a) each G matrix, except the first one, requires

1 - multiplication
1 - addition
1 - inversion

(the first one requires only one inversion) and

(b) each T matrix requires

2 - multiplications

Therefore, for this block of operation,

$$\begin{aligned} OP_1 &= (N-1) (3) + N + (N-1) (2) = 3N - 3 + N + 2N - 2 \\ &= 6N - 5 \end{aligned}$$

2. For calculating the diagonal sub-matrices of the inverted matrix,

(a) each F_{ii} matrix, except the last one, requires

1 - transposition
2 - multiplications
1 - addition

(the last one, F_{n,n}, doesn't require any basic operations).

Therefore, for this block of operation,

$$OP_2 = (N-1) (4) = 4N - 4.$$

3. For calculating the off-diagonal sub-matrices (F_{ij} for $j>i$) of the inverted matrix,

(a) each F_{ij} (for $j>i$) matrix requires
1 - multiplication

Therefore for this block of operation,

$$OP_3 = \left(\sum_{i=1}^N i-N \right) (1) = \frac{N}{2} (N + 1) - N = \frac{N}{2} (N-1).$$

The total number of basic operations is

$$OP = OP_1 + OP_2 + OP_3 = 6N - 5 + 4N - 4 + \frac{N(N-1)}{2}$$

$$OP = 1/2 (N^2 + 19N - 18).$$

(It should be noted that each of these basic operations is performed on a sub-matrix of size $M \times M$.)

For comparative purposes,

<u>N</u>	<u>N²</u>	<u>19N</u>	<u>OP</u>
1	1	19	1
2	4	38	12
3	9	57	24
4	16	76	37
5	25	95	51
6	36	114	66
7	49	133	82
8	64	152	99
9	81	171	117
10	100	190	136

9. SUMMARY

The purpose of the program is to efficiently invert a large matrix which can be partitioned into a symmetric, triple-banded matrix. For a trial run with $M = 2$ and $N = 5$, it was found that the standard inversion routine is faster. It is believed that a break-even point, as far as the execution time is concerned, would be reached for $MN = 40$. As far as a storage is concerned, the break even point occurs for $N = 5$.

As for the form of the output from the subroutine, it should be remembered that ideally a very large matrix will be inverted which can not be stored completely in the core of the machine. Therefore, the form of the output should be either a punched deck containing the inverse of the input matrix which would be used as input with another program or assembled and stored on magnetic tape. The choice is up to the individual user. The user will have to write an "assembler" subroutine to handle the problem as well as develop techniques for manipulating such a large matrix on magnetic tape so that it can be used in other operations leading to the desired solution.

10. PROGRAM LISTING

```

SUBROUTINE TRIBAND (AK,F,N,M,JKKJ,MN,IO)
DIMENSION AK(M,M,MN),F(M,M,JKKJ),A(9,9),B(9,9),C(9,9),D(9,9),E(9,9),G(9),H(9)

```

```

*****
*
K MATRIX MUST BE SYMMETRIC - THAT IS, EACH K-SUBMATRIX ON THE *
MAIN DIAGONAL MUST BE SYMMETRIC (K(I,I)=TRANPOSE OF K(I,I)) AND *
EACH K-SUBMATRIX OFF THE MAIN DIAGONAL MUST BE SUCH THAT K(I,J)= *
TRANPOSE OF K(J,I). *
*
*****

```

```

JOB=9
DO 55 I=1,JOB
DO 55 J=1,JOB
A(I,J)=0.0
B(I,J)=0.0
C(I,J)=0.0
D(I,J)=0.0
55 E(I,J)=0.0
DO 5002 II=1,M

```

```

*****
*
CALCULATE T AND G MATRICES AND STORE THEM IN THE AK MATRIX. *
THIS DESTROYOS THE ORIGINAL AK MATRIX. *
*
NOTE - G MATRICES ARE K ODD AND T MATRICES ARE K EVEN. *
*
*****

```

```

DO 100 K=1,MN,2
KKK=K+1
KK=K-1
IF(KK.EQ.0) GO TO 3
DO 1 I=1,M
DO 1 J=1,M
1 A(I,J)=AK(I,J,KK)
DO 5001 II=1,M
DO 5001 JJ=1,M
DUM=A(II,JJ)
A(II,JJ)=A(JJ,II)
5001 A(JJ,II)=DUM
DO 5003 IK=1,M
DO 5003 IJ=1,M
SUM=0.0
DO 5003 II=1,M
SUM=SUM+A(IK,II)*B(II,IJ)
5003 C(IK,IJ)=SUM
3 DO 2 I=1,M
DO 2 J=1,M
2 A(I,J)=AK(I,J,K)
DO 5004 IK=1,M
DO 5004 IJ=1,M
5004 E(IK,IJ)=A(IK,IJ)+C(IK,IJ)

```

```

*****
THE FOLLOWING INVERSION SCHEME WAS MODIFIED TO HANDLE THE
INVERSION OF A SINGLE ELEMENT ARRAY - E(1,1) - BY
HERBERT L. BILL, JR.
*****

```

```

*****
THE FOLLOWING INVERSION SCHEME IS BASED ON THE SUBROUTINE MINV
AUTHORED BY SAMPATH IYENGAR.
*****

```

```

REFERENCE - GENERAL INFORMATION MANUAL, AN INTRODUCTION TO
ENGINEERING ANALYSIS FOR COMPUTERS (IBM).
*****

```

```

*****
IF(M.EQ.1) GO TO 6052
NM1=M-1
DO 5005 IK=1,M
*****

```

```

*****
SEARCH FOR THE LARGEST ABSOLUTE-VALUED ELEMENT IN THE FIRST
COLUMN. EXCHANGE THE FIRST ROW WITH THE CORRESPONDING ROW.
*****

```

```

*****
BIG=0.0
LIMIT=M-IK+1
NR=1
DO5006 IKK=1,LIMIT
ABSA=ABS(E(IKK,1))
IF(BIG.GE.ABSA) GO TO 5007
BIG=ABSA
NR=IKK
5007 CONTINUE
5006 CONTINUE
H(IK)=IK
IF(NR.EQ.1) GO TO 5010
T=E(1,J)
E(1,J)=E(NR,J)
5008 E(NR,J)=T
H(IK)=IK+NR-1
*****

```

```

*****
CREATE PIVOT ROW ELEMENTS.
*****

```

```

*****
5010 DO 5009 J=1,NM1
JP1=J+1
5009 G(J)=E(1,JP1)/E(1,1)
G(M)=1.0/E(1,1)
*****

```

```

*****
COMPUTE NEW VALUES FOR THE MATRIX ELEMENTS WHILE SIMULTANEOUSLY
RENUMBERING THE ROWS.
*****

```

```

DO 5011 I=2,M
  IM1=I-1
  E(IM1,M)=-E(I,1)*G(M)
  DO 5011 J=1,NM1
    JP1=J+1
5011 E(IM1,J)=E(I,JP1)-E(I,1)*G(J)
    DO 5012 J=1,M
5012 E(M,J)=G(J)
5005 CONTINUE

```

```

*****
MATCH COLUMN EXCHANGES WITH THE PREVIOUS ROW EXCHANGES.
*****

```

```

DO 5013 IKK=1,M
  IK=M+1-IKK
  IF(H(IK).EQ.IK) GO TO 5014
  NC=H(IK)
  DO 5013 I=1,M
    T=E(I,IK)
    E(I,IK)=E(I,NC)
    E(I,NC)=T
5014 CONTINUE
5013 CONTINUE
GO TO 6053
6052 E(1,1)=1.0/E(1,1)
6053 DO 4 I=1,M
  DO 4 J=1,M
    4 AK(I,J,K)=E(I,J)
    DO 5020 IK=1,M
      DO 5020 IJ=1,M
        SUM=0.0
        DO 5020 II=1,M
          SUM=SUM+D(IK,II)+E(II,IJ)
5020 A(IK,IJ)=SUM
        DO 5 I=1,M
          DO 5 J=1,M
            5 E(I,J)=AK(I,J,KKK)
            IF(KK.EQ.0) GO TO 9
            DO 6 I=1,M
              DO 6 J=1,M
                6 AK(I,J,KK)=B(I,J)
            9 CONTINUE

```

```

DO 5021 IK=1,M
DO 5021 IJ=1,M
SUM=0.0
DO 5021 II=1,M
SUM=SUM+A (IK,II)*E(II,IJ)
5021 B(IK,IJ)=SUM
100 CONTINUE
DO 7 I=1,JOB
DO 7 J=1,JOB
B(I,J)=0.0
7 E(I,J)=0.0

```

ASSEMBLE THE INVERSE OF AK - ESTABLISH THE F MATRIX. *

ASSEMBLE THE MAIN DIAGONAL OF F. *

```

K=MN
NN=N+2
DO 57 I=1,JOB
DO 57 J=1,JOB
57 D(I,J)=0.0
DO 114 JJ=1,N
KK=K+1
NN=NN-1
IF(NN.GT.N) GO TO 58
DO 45 I=1,M
DO 45 J=1,M
45 A(I,J)=AK(I,J,KK)
DO 5030 I=1,M
DO 5030 J=1,M
5030 B(J,I)=A(I,J)
DO 5031 IK=1,M
DO 5031 IJ=1,M
SUM=0.0
DO 5031 II=1,M
SUM=SUM+E(IK,II)*B(II,IJ)
5031 C(IK,IJ)=SUM
DO 5032 IK=1,M
DO 5032 IJ=1,M
SUM=0.0
DO 5032 II=1,M
SUM=SUM+A(IK,II)*C(II,IJ)
5032 D(IK,IJ)=SUM
DO 46 I=1,M
DO 46 J=1,M
46 F(I,J,NN)=E(I,J)
58 DO 47 I=1,M
DO 47 J=1,M

```

```

47  A(I,J)=AK(I,J,K)
    DO 5033 I=1,M
    DO 5033 J=1,M
5033 E(I,J)=A(I,J)+D(I,J)
114  K=K-2
    NN=1
    DO 48 I=1,M
    DO 48 J=1,M
48  F(I,J,NN)=E(I,J)

```

```

*****
ASSEMBLE THE OFF DIAGONAL ELEMENTS OF F.
*****

```

```

    KJ=2
    KJJ=JKKJ
    MNM=N-1
    NMN=2*N-2
    NNMMN=N
    JKMKJN=JKKJ
    JK=2
    DO 215 II=1,MNM
    DO 205 I=1,M
    DO 205 J=1,M
205  A(I,J)=AK(I,J,NMN)
    DO 206 I=1,M
    DO 206 J=1,M
206  B(I,J)=F(I,J,NNMMN)
    DO 5040 IK=1,M
    DO 5040 IJ=1,M
    SUM=0.0
    DO 5040 INIK=1,M
    SUM=SUM+A(IK,INIK)*B(INIK,IJ)
5040 C(IK,IJ)=SUM
    DO 207 I=1,M
    DO 207 J=1,M
207  F(I,J,JKMKJN)=C(I,J)
    JKMKJN=JKMKJN-JK
    IF(II.EQ.1) GO TO 235
    JIK=II-1
    JKJ=KJJ
    DO 216 JKJKKJ=1,JIK
    JJK=JKJ-JIK
    DO 208 I=1,M
    DO 208 J=1,M
208  D(I,J)=F(I,J,JKJ)
    DO 5041 IK=1,M
    DO 5041 IJ=1,M
    SUM=0.0
    DO 5041 INIK=1,M
    SUM=SUM+A(IK,INIK)*D(INIK,IJ)

```



```

5041 E(IK,IJ)=SUM
      DO 209 I=1,M
      DO 209 J=1,M
209  F(I,J,JJK)=E(I,J)
      JKJ=JKJ+1
216  CONTINUE
      KJJ=KJJ-KJ
      KJ=KJ+1
235  JK=JK+1
      NMN=NMN-2
      NNMN=NNMN-1
215  CONTINUE

```

```
*****
```

```

      THE INVERSION IS COMPLETE.

```

```

      THE SUBMATRICES OF THE INVERSE ARE STORED AS A COLUMN VECTOR
      WITH THE FIRST BLOCK OF THE VECTOR BEING THE SUBMATRICES OF
      THE MAIN DIAGONAL OF THE INVERSE, AND WITH THE SECOND BLOCK
      OF THE VECTOR BEING THE OFF-DIAGONAL SUBMATRICES OF THE
      UPPER TRIANGLE OF THE INVERSE ARRANGED BY ROWS.

```

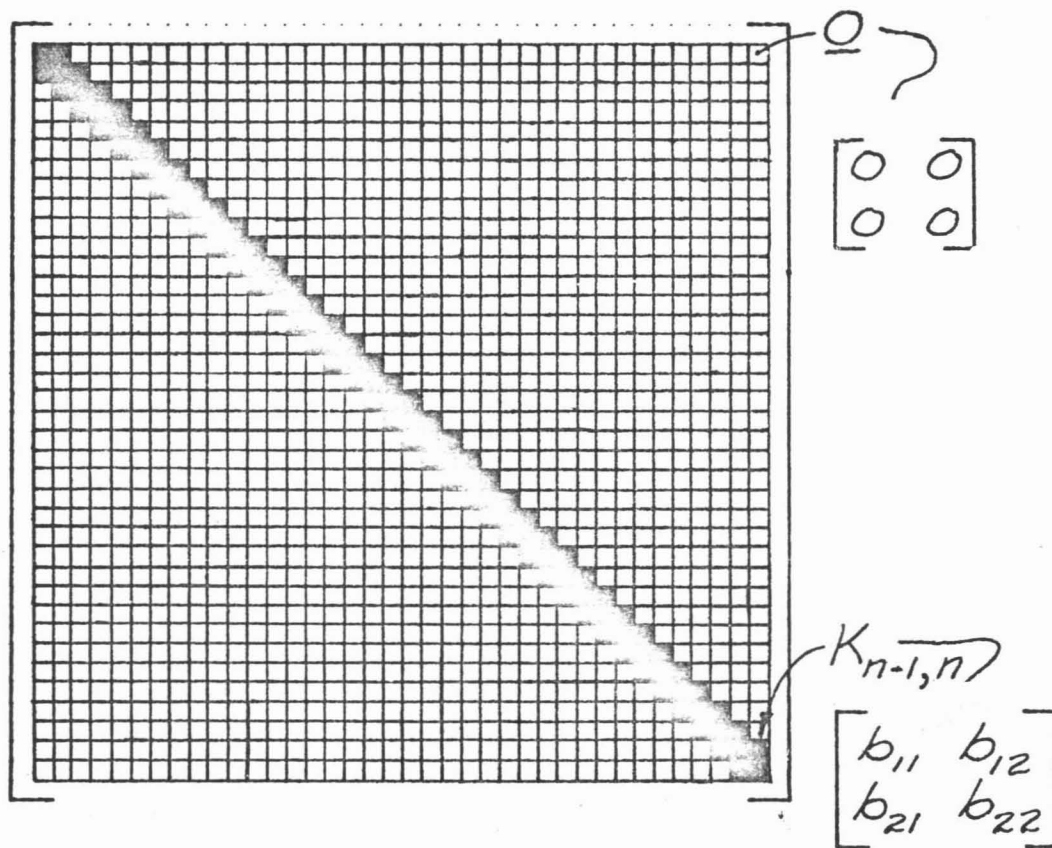
```
*****
```

```

      WRITE(10,2000)
      DO 6002 K=1,N
6002  WRITE(10,1000) K,K,((F(I,J,K),J=1,M),I=1,M)
      LL=N+1
      JK=N-1
      NN=2
      DO 6001 I=1,JK
      DO 6000 J=NN,N
      WRITE(10,1000) I,J,((F(K,L LL),L=1,M),K=1,M)
6000  LL=LL+1
      NN=NN+1
6001  CONTINUE
6000  FORMAT(1H0,12X,*F(*,12,*,*,12,*)*//,2(2X,F10,3,8X,F10.3/)////)
6000  FORMAT(1H1)
      RETURN
      END

```

11. FIGURES



d_{11}	d_{12}	f_{11}	f_{12}	b_{11}	b_{12}
d_{21}	d_{22}	f_{12}	f_{22}	b_{21}	b_{22}
0	0	b_{11}	b_{21}	c_{11}	c_{12}
0		b_{12}	b_{22}	c_{12}	c_{22}

$$= \begin{array}{|c|c|c|} \hline K_{jk} & K_{jj} & K_{ij}^T \\ \hline 0 & K_{ij} & K_{ii} \\ \hline \end{array}$$

Fig. 1 The Tri-Diagonal or Triple-Band Matrix with 2x2 Sub-Matrices

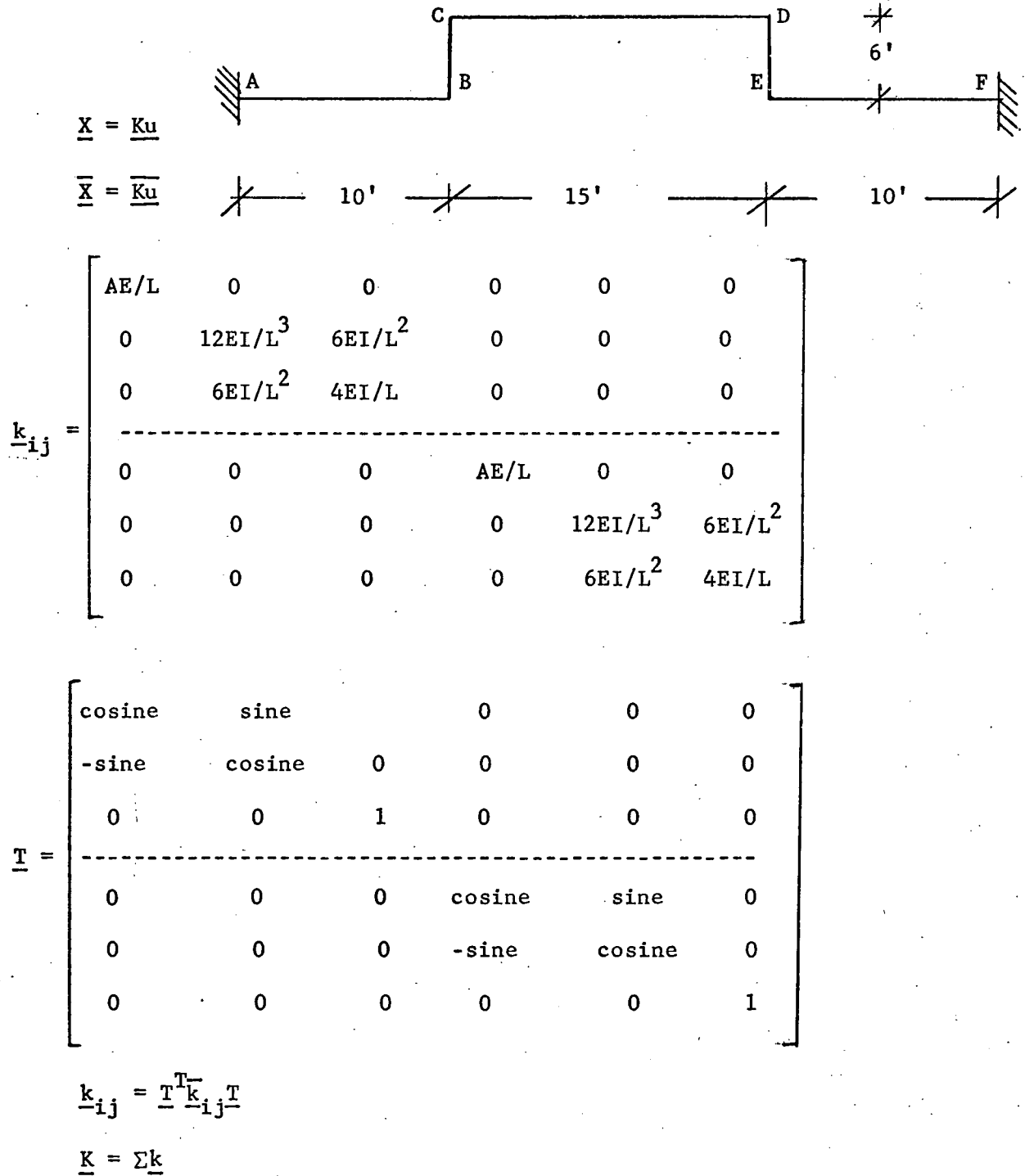


Fig. 2 Schematic Formulation of the Direct Stiffness Method for the Sample Frame

$$\underline{K} = \begin{array}{c} \text{JOINT} \\ \text{LOAD} \end{array} \begin{array}{c} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \\ \text{E} \\ \text{F} \end{array} \begin{array}{c} \text{A} \\ \text{B} \\ \text{C} \\ \text{D} \\ \text{E} \\ \text{F} \end{array} \begin{array}{c} \underline{k}_{AA}^{AB} \quad \underline{k}_{AB}^{AB} \quad \underline{0} \quad \underline{0} \quad \underline{0} \quad \underline{0} \\ \underline{k}_{BA}^{AB} \quad \underline{k}_{BB}^{AB} + \underline{k}_{BB}^{BC} \quad \underline{k}_{BC}^{BC} \quad \underline{0} \quad \underline{0} \quad \underline{0} \\ \underline{0} \quad \underline{k}_{CB}^{BC} \quad \underline{k}_{CC}^{BC} + \underline{k}_{CC}^{CD} \quad \underline{k}_{CD}^{CD} \quad \underline{0} \quad \underline{0} \\ \underline{0} \quad \underline{0} \quad \underline{k}_{DC}^{CD} \quad \underline{k}_{DD}^{CD} + \underline{k}_{DD}^{DE} \quad \underline{k}_{DE}^{DE} \quad \underline{0} \\ \underline{0} \quad \underline{0} \quad \underline{0} \quad \underline{k}_{ED}^{DE} \quad \underline{k}_{EE}^{DE} + \underline{k}_{EE}^{EF} \quad \underline{k}_{EF}^{EF} \\ \underline{0} \quad \underline{0} \quad \underline{0} \quad \underline{0} \quad \underline{k}_{FE}^{EF} \quad \underline{k}_{FF}^{EF} \end{array}$$

Fig. 2 (Continued)

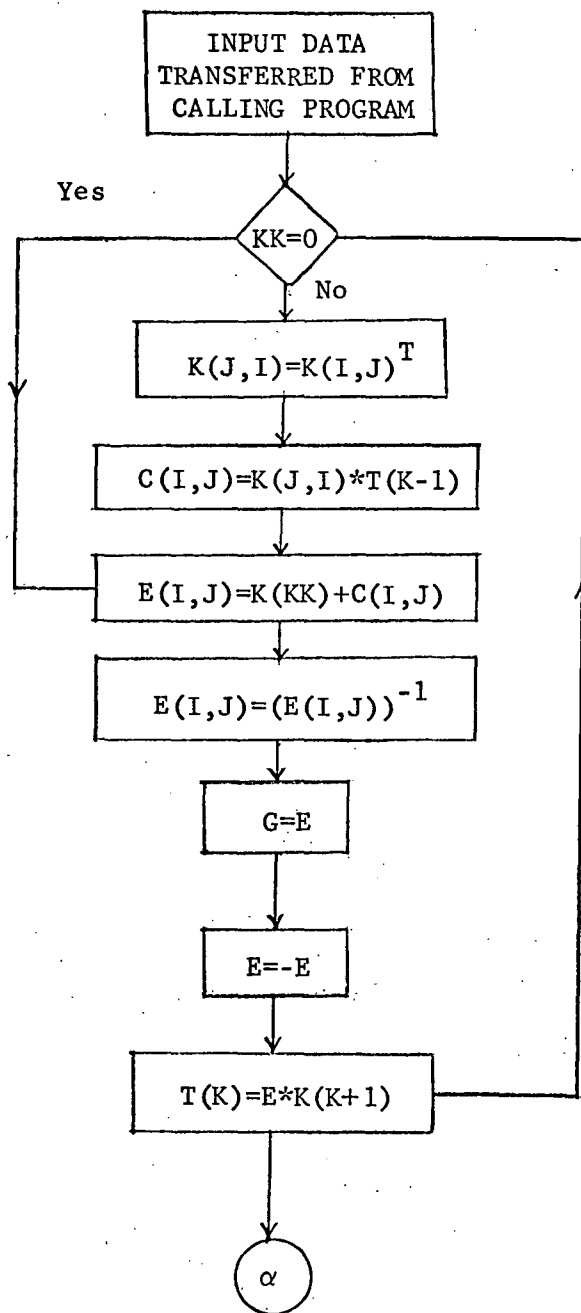


Fig. 3 Glow Chart for Calculating the \underline{T} and \underline{G} Matrices, and for Storing the \underline{T} and \underline{G} Matrices in the Three-Dimensional Vector of the Input Matrix

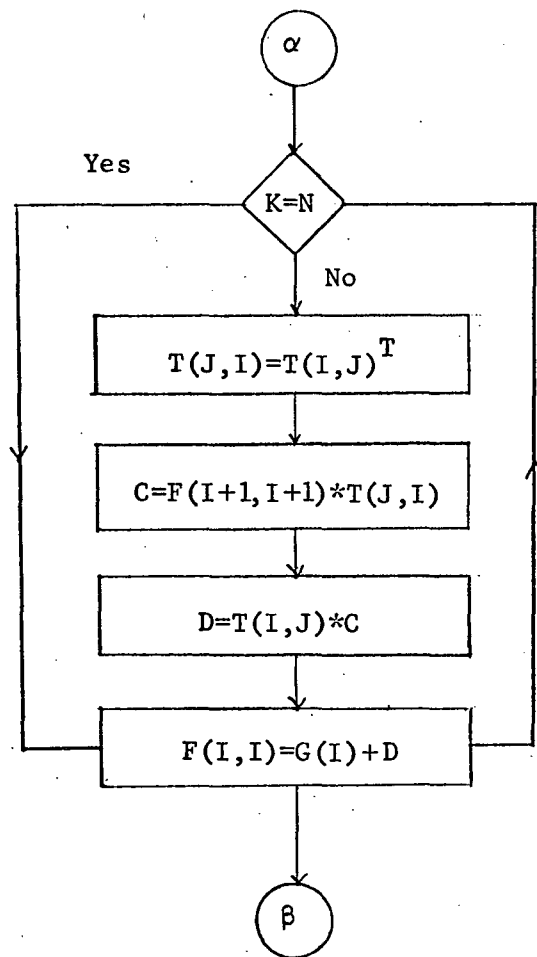


Fig. 4 Flow Chart for Calculating and Storing the Diagonal Elements of the Inverse

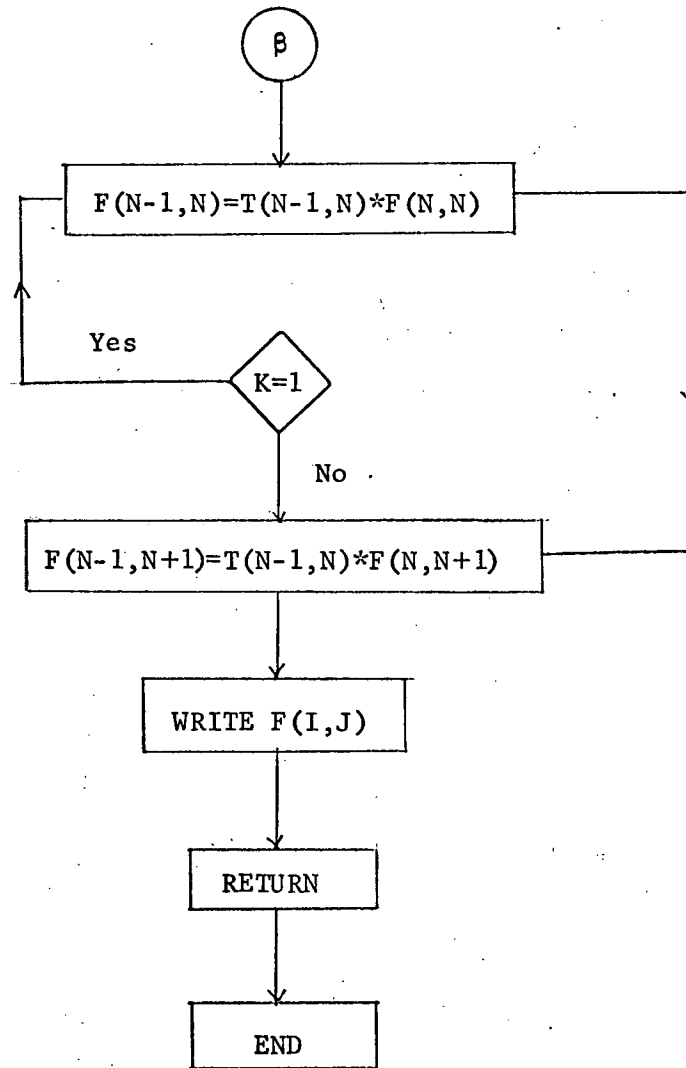


Fig. 5 Flow Chart for Calculating and Sorting the Off-Diagonal Elements of the Inverse

INPUT MATRIX IN NORMAL NOTATION:

2.00	1.00	1.00	0.50	0.00	0.00	0.00	0.00	0.00	0.00
1.00	2.00	0.50	1.00	0.00	0.00	0.00	0.00	0.00	0.00
1.00	0.50	1.00	0.50	0.50	0.25	0.00	0.00	0.00	0.00
0.50	1.00	0.50	1.00	0.25	0.50	0.00	0.00	0.00	0.00
0.00	0.00	0.50	0.25	2.00	1.00	1.00	0.50	0.00	0.00
0.00	0.00	0.25	0.50	1.00	2.00	0.50	1.00	0.00	0.00
0.00	0.00	0.00	0.00	1.00	0.50	1.00	0.50	0.50	0.25
0.00	0.00	0.00	0.00	0.50	1.00	0.50	1.00	0.25	0.50
0.00	0.00	0.00	0.00	0.00	0.00	0.50	0.25	3.00	1.50
0.00	0.00	0.00	0.00	0.00	0.00	0.25	0.50	1.50	3.00

INPUT MATRIX AS A THREE-DIMENSIONAL VECTOR:

2.00	1.00
1.00	2.00
1.00	0.50
0.50	1.00
1.00	0.50
0.50	1.00
0.50	0.25
0.25	0.50
2.00	1.00
1.00	2.00
1.00	0.50
0.50	1.00
1.00	0.50
0.50	1.00
0.50	0.25
0.25	0.50
3.00	1.50
1.50	3.00

Fig. 6 Input Matrix for Test Run of the Program

OUTPUT AS OBTAINED FROM THE SUBROUTINE:

F(1,1)

2.148	-1.074
-1.074	2.148

F(2,2)

5.926	-2.963
-2.963	5.926

F(3,3)

3.259	--1.630
-1.630	3.259

F(4,4)

5.333	--2.667
-2.667	5.333

F(5,5)

0.593	-0.296
-0.296	0.593

F(1,2)

-2.963	1.481
1.481	-2.963

F(1,3)

1.630	-0.815
-0.815	1.630

F(1,4)

-1.778	0.889
0.889	-1.778

Fig. 7 Upper Triangle of the Inverse of the Input Matrix for the Test Run

F(1,5)

0.296	-0.148
-0.148	0.296

F(2,3)

-3.259	1.630
1.630	-3.259

F(2,4)

3.556	-1.778
-1.778	3.556

F(2,5)

-0.593	0.296
0.296	-0.593

F(3,4)

-3.556	1.778
1.778	-3.556

F(3,5)

0.593	-0.296
-0.296	0.593

F(4,5)

-0.889	0.444
0.444	-0.889

Fig. 7 (Continued)

12. BIBLIOGRAPHY

1. Gatewood, B. E. and Ohanian, N.
"Tri-Diagonal Matrix Method for Complex Structures",
Journal of the Structural Division, Proceedings of
the American Society of Civil Engineers, Vol. 91,
ST2, April 1965, pp. 27-41.
2. Martin, H. C.
Introduction to Matrix Methods of Structural Analysis,
McGraw-Hill Book Co., New York, 1966, pp. 69-168.
3. Przemieniecki, J. S.
Theory of Matrix Structural Analysis, McGraw-Hill Book
Co., New York, 1968, pp. 427-428.

13. ACKNOWLEDGMENTS

Thanks and appreciation are due to Miss Karen Philbin and Mrs. Jane Lenner for typing and proofreading this report.