

GRAPH-THEORETICAL FORCE METHOD OF FINITE ELEMENT MODELS WITH TRIANGULAR AND RECTANGULAR ELEMENTS

A. Kaveh* and E. Ebrahimi

Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of
Science and Technology, Narmak, Tehran-16, Iran

Received: 20 June 2011, **Accepted:** 10 January 2012

ABSTRACT

In this paper an efficient method is developed for the formation of null bases of finite element models (FEMs) composed of rectangular and triangular plane stress and plane strain elements, corresponding to highly sparse and banded flexibility matrices. This is achieved by associating special graphs to the finite element models and selecting appropriate subgraphs for the formation of localized self stress systems. The efficiency of the present method is illustrated through some examples.

Keywords: Finite elements; graph theory; null basis; flexibility matrix; sparsity; ordering; stiffness matrix

1. INTRODUCTION

The force method of structural analysis, in which the member forces are used as unknowns, is appealing to engineers, since the properties of members of a structure most often depend on the member forces rather than joint displacements. This method was used extensively until 1960. The advent of the digital computer and the amenability of the displacement method for computation attracted most researchers. As a result, the force method and some of the advantages it offers in non-linear analysis and optimization has been neglected.

Four different approaches are adopted for the force method of structural analysis classified as:

1. Topological force methods,
2. Algebraic force methods,
3. Mixed algebraic-combinatorial force methods,
4. Integrated force method.

* E-mail address of the corresponding author: alikaveh@iust.ac.ir (A. Kaveh)

Topological methods have been developed by Henderson [1] and Maunder [2] for rigid-jointed skeletal structures using manual selection of the cycle bases of their graph models. Methods suitable for computer programming are due to Kaveh [3-5]. These topological methods are generalized to cover different types of skeletal structures, such as rigid-jointed frames, pin-jointed planar trusses and ball-jointed space trusses [6,7]. Algebraic methods have been developed by Denke [8], Robinson [9], Topçu [10], Kaneko et al. [11], Soyer and Topçu [12] and mixed algebraic-topological methods have been used by Gilbert et al. [13], Coleman and Pothén [14-15], and Pothén [16]. The integrated force method has been developed by Patnaik [17-18], in which the equilibrium equations and the compatibility conditions are satisfied simultaneously in terms of the force variables. Graph theoretical methods for the formation of null basis for the model consisting of triangular finite elements [22] and rectangular finite elements are developed by Kaveh et al. [23].

In this paper an efficient algorithm is presented for the formation of null bases for the models consisting of rectangular and triangular plane stress and plane strain finite element models, corresponding to highly sparse and banded flexibility matrices. The bases obtained by this algorithm require low computational effort leading to highly sparse flexibility matrices with very small bandwidth. Thus using this algorithm, optimal flexibility analysis of such FEMs becomes feasible.

Here, first an interface graph is defined for 2D finite element models and then subgraphs corresponding to self stress systems are generated. By applying unit bi-actions and solving the corresponding statically determinate substructures, some null vectors are obtained. This is repeated for all the selected subgraphs to obtain the null basis. The efficiency of the present method is illustrated through simple examples.

2. ALGEBRAIC FORCE METHODS

Consider a discrete or discretized structure S , which is assumed to be statically indeterminate. Let r denote the m -dimensional vector of generalized independent element (member) forces, and p the n -vector of nodal loads. The equilibrium conditions of the structure can be expressed as

$$Ar = p. \quad (1)$$

where A is an $n \times m$ equilibrium matrix. The structure is assumed to be rigid, and therefore A has a full rank, i.e. $t = m \square n > 0$ and $\text{rank } A = n$.

The member forces can be written as

$$r = B_0 p + B_1 q, \quad (2)$$

where B_0 is an $m \times n$ matrix with AB_0 being an $n \times n$ identity matrix, and B_1 is an $m \times t$ matrix with AB_1 being an $n \times t$ zero matrix. B_0 and B_1 always exist for a structure, and in fact many of them can be found for a structure. B_1 is called a self-stress matrix as well as null basis matrix. Each column of B_1 is known as a null vector. Notice that the null space, null basis and null vectors correspond to complementary solution space, statical basis and S.E.Ss,

respectively, when S is taken as a general structure.

Minimizing the complementary potential energy requires that r minimize the quadratic form,

$$\frac{1}{2} r^T F_m r, \tag{3}$$

subjected to the constraint as in Eq. (1). F_m is an $m \times m$ block diagonal element flexibility matrix. Using Eq. (2), it can be seen that q must satisfy the following equation:

$$(B_1^T F_m B_1) q = -B_1^T F_m B_0 p, \tag{4}$$

Where $B_1^T F_m B_1 = G$ is the overall flexibility matrix of the structure. Computing the redundant forces q from Eq. (4), r can be found using Eq. (2), i.e.

$$r = [B_0 - B_1 (B_1^T F_m B_1)^{-1} B_1^T F_m B_0] p, \tag{5}$$

The structure of G is again important, and its sparsity, bandwidth and conditioning govern the efficiency of the force method. For the sparsity of G one can search for a sparse B_1 matrix, which is often referred to the sparse null basis problem.

Many algorithms exist for computing a null basis B_1 of a matrix A . For the moment, let A be partitioned so that,

$$AP = [A_1, A_2] \tag{6}$$

where A_1 is $n \times n$ and non-singular, and P is a column permutation matrix that may be required in order to ensure that A_1 is non-singular. One can write:

$$B_1 = P \begin{bmatrix} -A_1^{-1} A_2 \\ I \end{bmatrix}$$

$$AB_1 = [A_1 \ A_2] \begin{bmatrix} -A_1^{-1} A_2 \\ I \end{bmatrix} = 0 \tag{7}$$

Obviously, a permutation P that yields a non-singular A_1 , can be chosen purely symbolically, but this says nothing about the possible numerical conditioning of A_1 and the resulting B_1 .

In order to control the numerical conditioning, pivoting must be employed. There are many methods based on various matrix factorizations, including the Gauss-Jordan elimination, **QR**, **LU**, **LQ** and Turn-back method. The latter method is briefly described in the following:

Turn-Back LU Decomposition Method: Topçu developed a method, the so-called Turn-back LU procedure, which is based on factorization and often results in highly sparse

and banded B_1 matrices. Heath et al. [20] adopted this method for use with **QR** factorization. Due to the efficiency of this method, a brief description of their approach will be presented in the following.

Write the matrix $A = (a_1, a_2, \mathbf{K}, a_n)$ by columns. A start column is a column such that the ranks of $(a_1, a_2, \mathbf{K}, a_{s-1})$ and $(a_1, a_2, \mathbf{K}, a_s)$ are equal. Similarly, a_s will be a start column if it is linearly dependent on lower-numbered columns. The coefficients of this linear dependency form a null vector whose highest numbered non-zero is in position s . It is easy to see that the number of start columns is $m - n = t$, the dimension of the null space of A .

The start column can be found by performing a **QR** factorization on A , using orthogonal transformations to annihilate the subdiagonal non-zeros. Suppose that in the process of performing the **QR** factorization we do not perform column interchanges but simply skip over any columns which are already zero on and below the diagonal. The result will then be a factorization of the form

$$A = Q \begin{array}{|c|} \hline \begin{array}{c} \diagdown \\ \text{R} \\ \diagup \end{array} \\ \hline \end{array} \quad (8)$$

The start columns are those columns where the upper triangular structure jogs to the right; that is, a_s is a start column if the highest non-zero position in column s of R is no larger than the highest non-zero position in earlier columns of R .

The Turn-back method finds one null vector for each start column a_s by "turning back" from column s to find the smallest k for which columns $a_s, a_{s-1}, \mathbf{K}, a_{s-k}$ are linearly dependent. The null vector has a non-zero only in position $s-k$ through s . Thus, if k is small for most of the start columns, then the null basis will have a small profile. Notice that the Turn-back operates on A , and not on R . The initial **QR** factorization of A is used only to determine the start columns, and then discarded.

The null vector that Turn-back finds from start column a_s may not be non-zero in positions. Therefore Turn-back needs to have some way to guarantee that its null vectors are linearly independent. This can be accomplished by forbidding the left-most column of the dependency for each null vector from participating in any later dependencies. Thus, if the null vector for start column a_s has its first non-zero in position $s-k$, every null vector for a start column to the right of a_s will be zero in position $s-k$.

3. FORCE METHOD OF STRUCTURAL ANALYSIS

3.1 Natural associate graph and interface graph of a finite element model

In order to transfer the topological property of a finite element model to the connectivity of a graph, ten different graphs are introduced in Ref. [21]. Here, the natural associate graph is

used and an additional new graph is defined.

Natural associate graph:

The natural associate graph of a FEM is constructed as follows:

1. Associate one node of the associate graph with an element of the FEM.
2. Two nodes of the associate graph are connected by a member if the corresponding elements in the FEM have a common edge.

A FEM and the corresponding associate graph are shown in Figure 1(a).

Interface graph:

The Interface graph of a FEM is constructed by the following rules:

1. This graph contains all the nodes of the FEM.
2. For each edge of a finite element of the model, associate one independent member. Thus, a typical overlap of two elements in FEM is represented by double members in interface graph.
3. One diagonal member is associated with each rectangular element of the model.

Figure 1 shows a FEM and the corresponding interface graph.

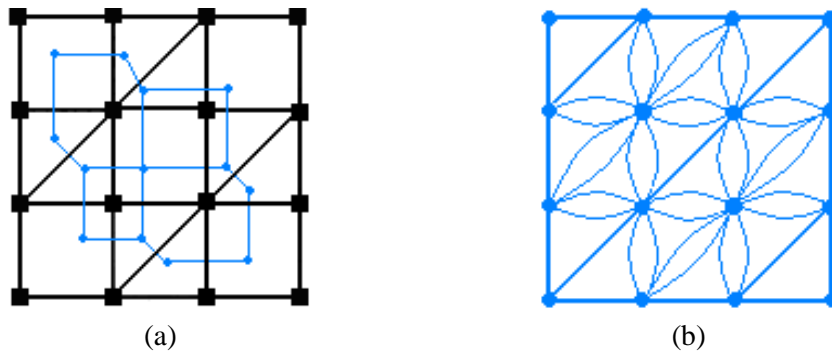


Figure 1. (a) A FEM and its corresponding natural associate graph; (b) The interface graph

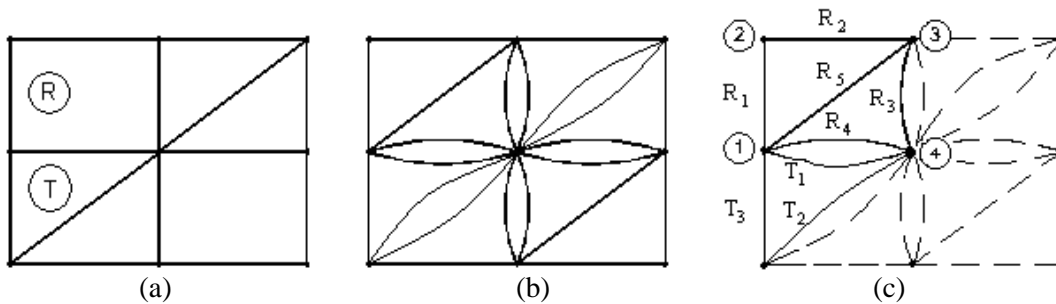


Figure 2. (a) A 2D finite element model; (b) The interface graph; (c) Numberings for a typical rectangular element R and a triangular element T

The member of the interface graph should be numbered according to the numbering of the FEM. A typical numbering is shown in Figure 2(c). For each rectangular element like R, five members of the interface graph and for each triangular element like T, three members of

interface graph should be numbered consequently. The numbering is performed according to the direction of the independent element forces (Figure 3(a) and Figure 3(b)).

3.2 The pattern corresponding to the self stress systems

The nodal forces and independent element forces of a rectangular and triangular element is defined as shown in Figure 3. This is the same convention used by Przemieniecki [25].

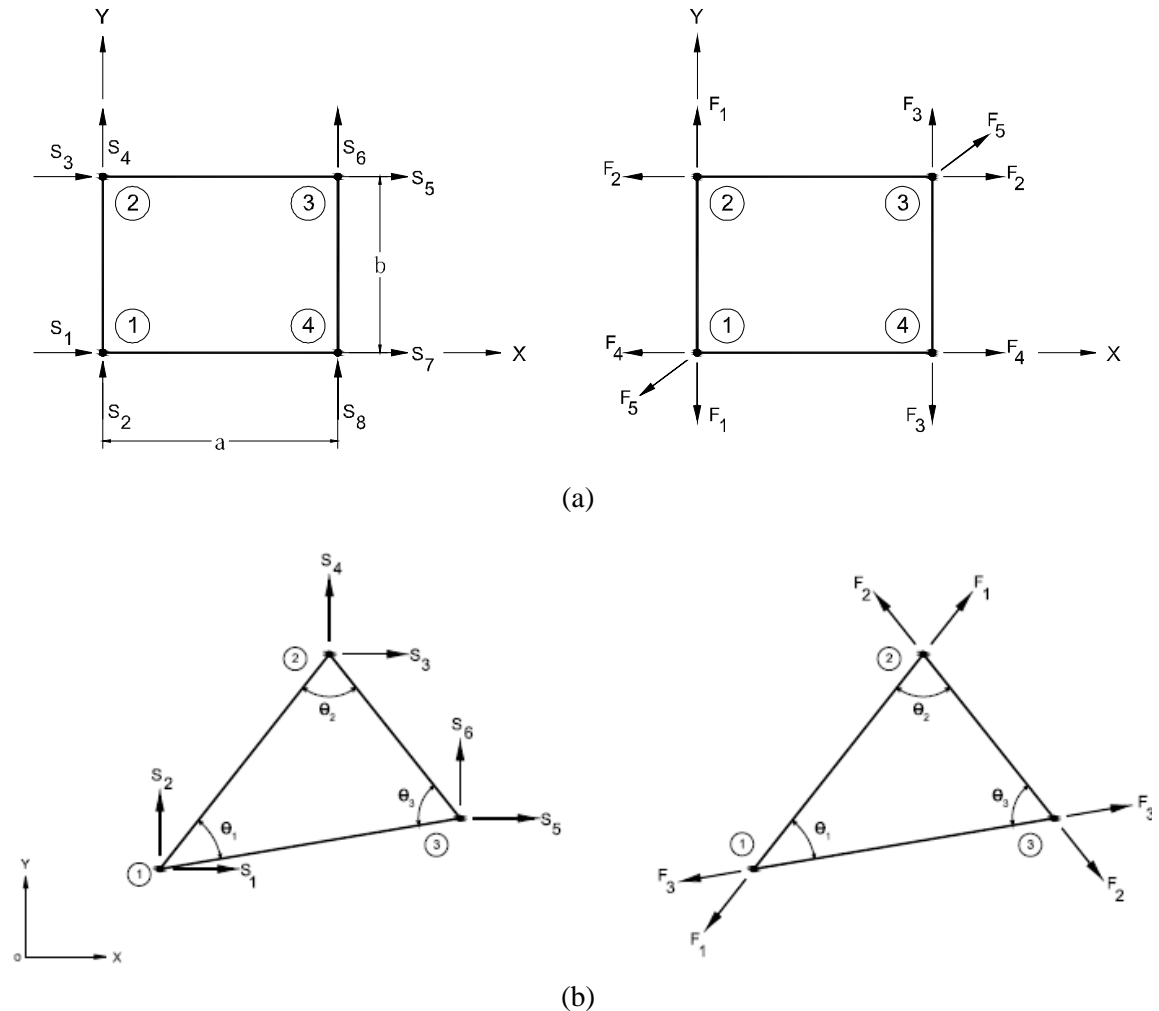


Figure 3. (a) The nodal and element forces for a rectangular element; (b) The nodal and element forces for a triangular element

Considering Figure 3(a), in order to find the patterns corresponding to the self stress systems, a rectangular is simulated as a planar truss formed as the 1-skeleton of the rectangular element together with a diagonal member. This is possible since the independent element forces F_1 to F_5 are applied to the nodes and are along the edges of the rectangular. Also, a FEM with plane strain and plane stress triangular elements can be viewed as a planar truss (Figure 3(b)). The statical indeterminacy of planar truss with m members and n nodes

is given as, $\gamma(S) = m - 2n + 3$.

The patterns of the underlying subgraphs of self stress systems (null vectors) are identified as follows:

3.3 Type I self stress system

Each double member of the interface graph is the underlying subgraph of a self stress system. In other words, a double member consisting of members numbered as i and j with ($i < j$), have two non-zero entries in the null basis matrix in the corresponding rows i and j in which the entry in row i is -1 and the entry in row j is $+1$. These double members are called type-I γ -cycles. The number of these double members is equal to the number of members of the natural associate graph, (see Kaveh [19, 23] for the definition of a γ -cycle).

Using these double members nearly %80 of the columns of a null basis matrix can easily be generated. For finding these double members one can use the adjacency matrix or the node-member incident matrix of the interface graph.

3.4 Type II self stress system

There are other types of self stress systems in the FEM which are topologically identical to the minimal self stress systems of corresponding planar trusses. The underlying subgraphs of these systems are known as type II γ -cycles, corresponding to the regional cycles of the natural associate graph bounding a single node of the FEM. In other words, if each multiple member from the interface graph is substituted by a member or generators of the Type I self stress systems are removed from S_1 , and then the remaining subgraph is a graph, denoted by S .

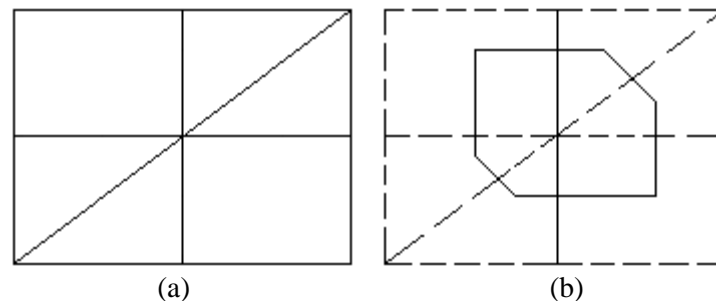


Figure 4. (a) A finite element model with 6 elements; (b) The corresponding associate graph

In general the self stress systems built on S are called Type II self stress systems. In fact these systems are γ -cycles which correspond to minimal cycles of associate graph of finite element model (see Refs. [26 or 27] for definition). A finite element model with 6 elements is shown in Figure 4(a), and its associate graph is depicted in Figure 4(b).

3.5 Type III self stress system

Each regional cycle bounding a cut-out in the FEM corresponds to a regional cycle of the natural associate graph with 3 degrees of statical indeterminacy, correspond to 3 self stress systems.

For a FEM with n_c cut-outs, apart from the self stress systems corresponding to double members of the interface graph, b_1 (natural associate graph) $- n_c + 3n_c$ additional self stress systems should be generated. This is obvious since for each γ -cycle of S corresponding to

non-cut out cycle of natural associate graph one self stress system, and for each general subgraph corresponding to a cut out cycle of natural associate graph, three independent self stress systems can be generated. Such a general subgraph consists of three independent γ -cycles. In the above relations, b_1 (natural associate graph) is the first Betti number of the natural associate graph of the FEM.

4. SELECTION OF OPTIMAL γ -CYCLES CORRESPONDING TO TYPE II SELF STRESS SYSTEMS

Thus far, it is found out that each γ -cycle corresponds to a cycle of the associate graph. Also each cycle with n nodes from $A(S)$ such as c passes through n elements. The subgraph s_1^c ($s_1^c \subset S_1$) which is relevant to these n elements and cycle c , is a base for the selection of an optimal γ -cycle. Such a subgraph may contain simple and multiple members, where each multiple member with k members corresponds to the overlap of k elements, and each simple member corresponds to the edge of a boundary element. By imposing a special condition on such subgraphs s_1^c , the lists corresponding to optimal γ -cycles can be obtained.

A finite element model with six elements and its associate graph are shown in Figure 5(a). The corresponding s_1^c which contains multiple and simple members is illustrated in Figure 5(b), and the corresponding γ -cycle is depicted in Figure 5(c).

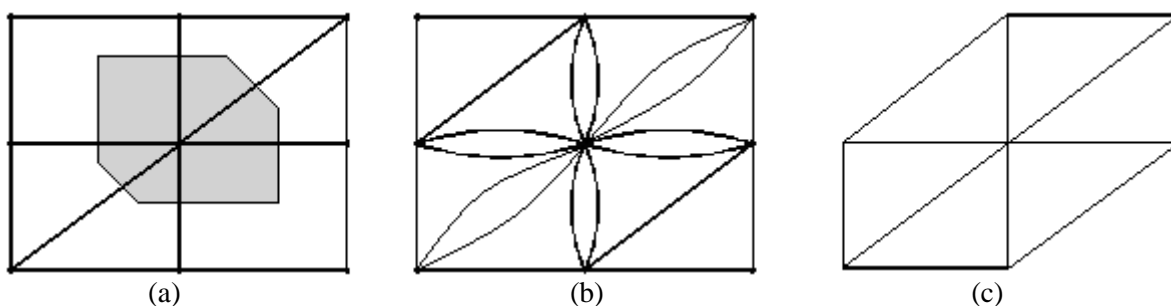


Figure 5. (a) A finite element model with its associate graph; (b) The corresponding s_1^c ; (c) The corresponding γ -cycle.

In general, from each s_1^c many γ -cycles (self stress systems) can be extracted, since each simple member of a multiple member can be present in the final graph, while the presence of a simple member in the final graph is obvious.

Thus for obtaining an optimal self stress system, on each s_1^c , two basic selections should be performed which are as follows:

1. Selection of the generator or the last member of a self stress system, which is required for independency of null vectors.
2. Selection of a list of members from the subgraph s_1^c with maximum possible number for the first member. This selection reduces the bandwidth of the null basis

matrix considerably.

The mathematical representation of this selection can be expressed as

$$\text{Minimize } (j - i) \tag{9}$$

Where j is the generator's member number and i is the least member number of current γ -cycle. In the following a simple and fast method is presented for these selections.

4.1 Selection of optimal lists

First for each subgraph s_i^c we have to delete all the nodes with degree 2 and the members connected to such nodes (Figure 5(c)) since the numeric values of these members in the corresponding null vector are zero, and therefore they play no role in the formation of a γ -cycle. It should be mentioned that if these members are not omitted then the process of finding a generator will be disturbed. Since each double member corresponds a self stress system which has already been selected, then for independency of a new null vector, each member of a double member which has greater member number cannot be selected as the last member. Then between the lower numbers of double members, the maximum number should be chosen as the last member of the current γ -cycle.

For finding a typical optimal list of members of a γ -cycle two row matrices T_1 and T_2 are considered as follows:

$$T_1 = \{d_1^1, d_2^1, \mathbf{K}, s_1, s_2, \mathbf{K}, d_i^1, s_j, \mathbf{K}, d_k^1\}$$

$$T_2 = \{d_1^2, d_2^2, \mathbf{K}, d_i^2, \mathbf{K}, d_k^2\}$$

In the above matrices, d_i^2, d_i^1 ($i=1, \mathbf{K}, k$) are lower and higher member numbers of the double members, in which $d_i^1 < d_i^2$ ($i=1, \mathbf{K}, k$) and s_j ($j=1, \mathbf{K}, t$) are simple members of the selected subgraph. The entries of row matrix, $T_2, (d_i^2)$ show the last members of previous self stress systems. Obviously, the maximum number of the entries of matrix, $T_1, \{d_i^1, s_j\}$ ($i=1, \mathbf{K}, k ; j=1, \mathbf{K}, t$), must be selected as the last member (d_l^1) of this self stress system.

In order to increase the first member number and minimizing the difference between the first member number and the generator of the current self stress system, the following condition can be used.

$$\text{if } ((d_i^2 > d_i^1) \text{ and } (d_i^1 < d_l^1)) \text{ then } d_i^1 = d_i^2 \quad (i = 1, \dots, k) \tag{10}$$

After using the above condition, the desired optimal list will be obtained as $\{d_i^1, s_j\}$ ($i=1, \mathbf{K}, k ; j=1, \mathbf{K}, t$).

After finding optimal lists corresponding to type II γ -cycles, using relevant equilibrium submatrix, numerical values for each null vector are calculated.

The list corresponding to the remaining subgraph will have DSI equal to 3. Three null vectors corresponding to such cycles will be obtained directly from the equilibrium submatrix which leads to suboptimal basis.

For type III γ -cycles, finding an optimal list is a time consuming process and considering the fact that the number of cut outs is low in the real structures, the use of this process is not economical for improvement of the final null basis. Thus for each cycle of this type, graph s_1^c is decomposed and all members corresponding to type I and Type II self stress systems and all the nodes of degree 2 are removed [22].

5. ALGORITHM

Step 1: Generate the associate graph of finite element model and use an efficient method for its node numbering, Kaveh [27]. It is obvious that a good numbering of this graph corresponds to a good numbering of the elements of a finite element model. This numbering leads to a banded adjacency matrix of the graph and correspondingly to a banded flexibility matrix. Since numbering the members of the interface graphs correspond to the element numbering of the finite elements, therefore such a numbering is the only parameter for controlling the bandwidth of the flexibility matrix.

Step 2: Setup the equilibrium matrix of the finite elements model.

Step 3: Generate the interface graph and perform its numbering. The numbering of this graph should be performed according to the element numbering of the considered finite elements model. After this numbering, the interface graph can easily be formed and its members can be numbered.

Step 4: Find the Type I self stress systems. All multiple members of interface graph are identified and the values -1 and 1 are assigned to appropriate rows (corresponding to the member numbers) and the corresponding null vectors are created.

Step 5: Find the Type II self stress systems. Using the Type I and Type II minimal cycles of the associate graph, the subgraphs s_1^c relevant subgraphs are identified and their corresponding optimal lists are found.

Step 6: Calculate numerical values of the optimal lists. Using optimal lists selected in Step 5, null vectors corresponding to the Type I and Type II cycles are calculated from the relevant equilibrium submatrix. For each generator, unit load is applied at a cut in the generator and the internal forces are calculated to form a null vector.

Step 7: Order the null vectors. At this step the constructed null vectors should be ordered such that their generators form a list with an ascending order.

In the following the efficiency of this algorithm is demonstrated using numerical

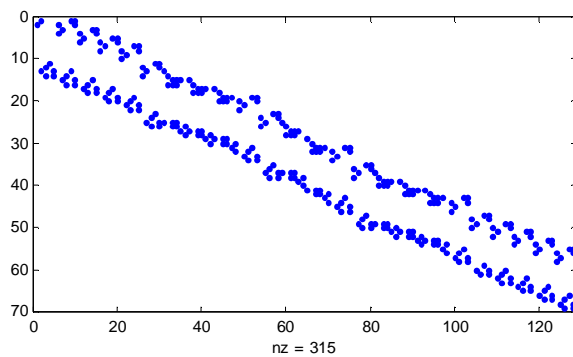


Figure 8. Pattern of the equilibrium matrix

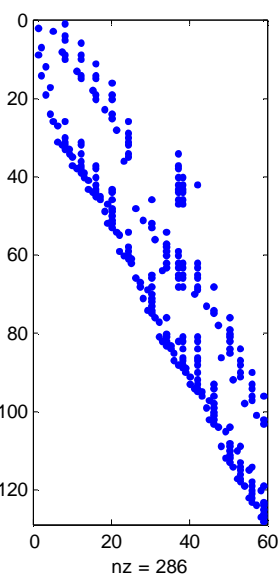


Figure 9. Pattern of the null bases matrix using the present method

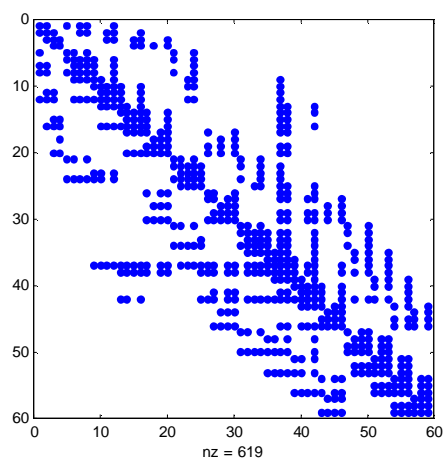


Figure 10. Pattern of the flexibility matrix utilizing the present method

Table 1: Comparison of the displacement method and the present force method

Element stresses						
Method	Displacement method			The present force method		
Element	σ_{xx}	σ_{yy}	σ_{xy}	σ_{xx}	σ_{yy}	σ_{xy}
	kN/m ² xE+4	kN/m ² xE+4	kN/m ² xE+4	kN/m ² xE+4	kN/m ² xE+4	kN/m ² xE+4
1	0.2447	0.0664	0.0938	0.2447	0.0664	0.0938
5	-2.2291	0.0126	0.0756	-2.2291	0.0126	0.0756
8	0.0187	-0.0227	-0.0115	0.0187	-0.0227	-0.0115
10	0.0021	-0.0104	0.0470	0.0021	-0.0104	0.0470
12	-0.1141	-0.0008	0.0009	-0.1141	-0.0008	0.0009
16	0.0432	0.0175	0.0245	0.0432	0.0175	0.0245
18	-0.0905	-0.0112	0.0309	-0.0905	-0.0112	0.0309
22	0.0151	0.0114	0.0165	0.0151	0.0114	0.0165
24	-0.0313	-0.0009	0.0358	-0.0313	-0.0009	0.0358
26	-0.0493	-0.0043	0.0305	-0.0493	-0.0043	0.0305
28	0.0349	-0.9580	0.0220	0.0349	-0.9580	0.0220
32	0.0133	-0.0448	0.0066	0.0133	-0.0448	0.0066

Example 2: A finite element model which is supported in a statically determinate fashion is depicted in Figure 11. This structure is also discretized using quadrilateral and triangular finite elements. The properties of the model are:

- Number of quadrilateral elements = 66
- Number of Triangular elements = 44
- Number of type I self stress systems = 172
- Number of type II self stress systems = 63
- $E = 2e+8 \text{ kN/m}^2$
- $\nu = 0.3$
- $t=0.1\text{m}$
- Number of nodes = 115
- $DSI_T = 235 = (172 + 63)$

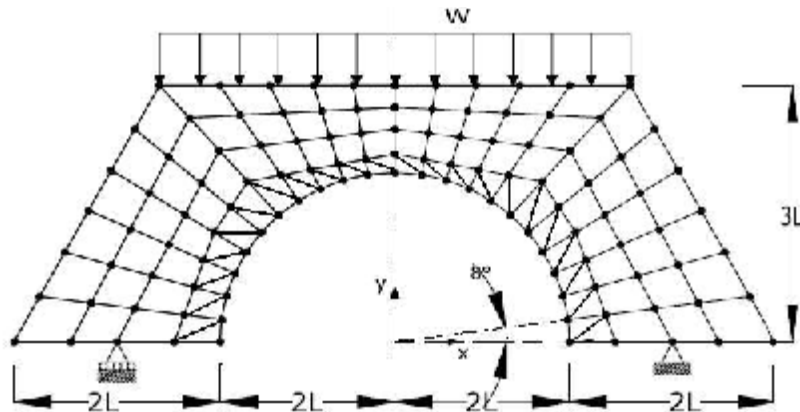


Figure 11. A finite element model with quadrilateral and triangular elements

Pattern of the null basis matrix with 1100 entries for the present method is shown in Figure 12. Pattern of the flexibility matrix using the present algorithms is illustrated in Figure 13.

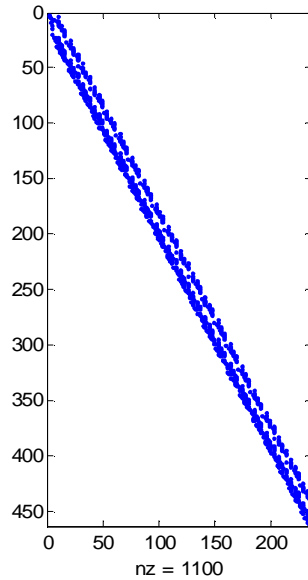


Figure 12. Pattern of the null bases matrix

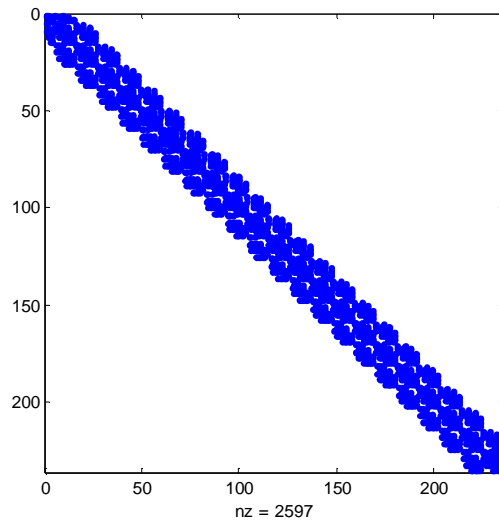


Figure 13. Pattern of the flexibility matrix

Example 3: A circular plate which is supported in a statically determinate fashion is depicted in Figure 14. The properties of the model are:

Number of quadrilateral elements = 48

Number of Triangular elements = 216

Number of type I self stress systems = 408

Number of type II self stress systems = 145

$E = 2e+8 \text{ kN/m}^2$ $\nu = 0.3$

$t=0.05\text{m}$

Number of nodes = 169

$DSI_T = 553$

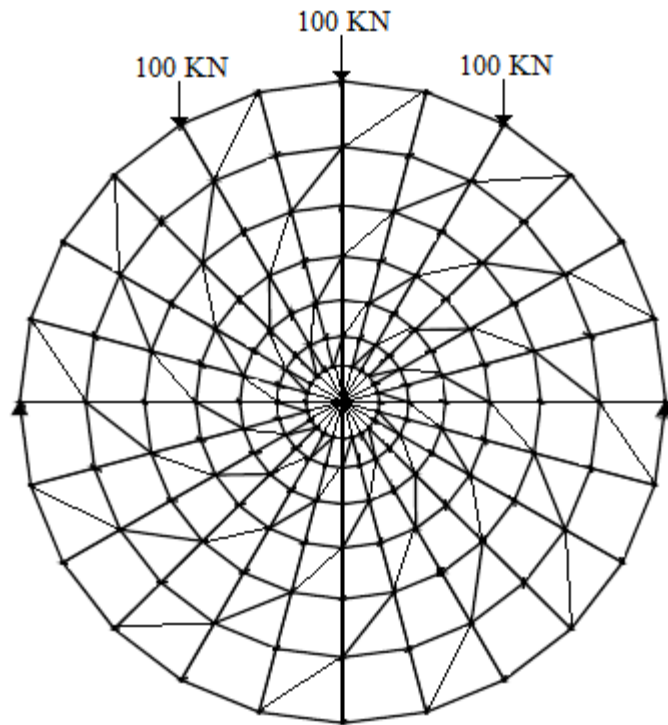


Figure 14. A circular disk and the corresponding loading

Pattern of the null basis matrix with 2584 entries for the present method is shown in Figure 15. Pattern of the corresponding flexibility matrix is illustrated in Figure 16.

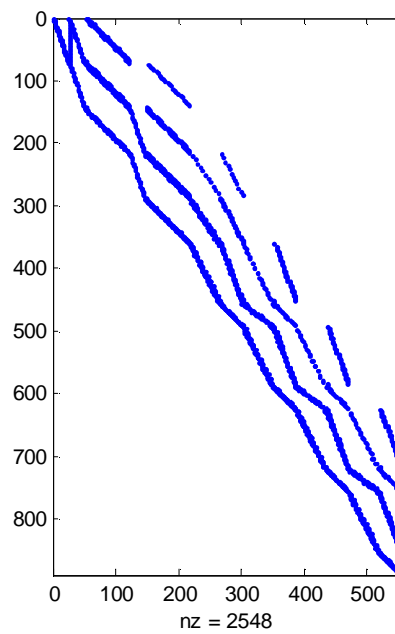


Figure 15. Pattern of the null bases matrix

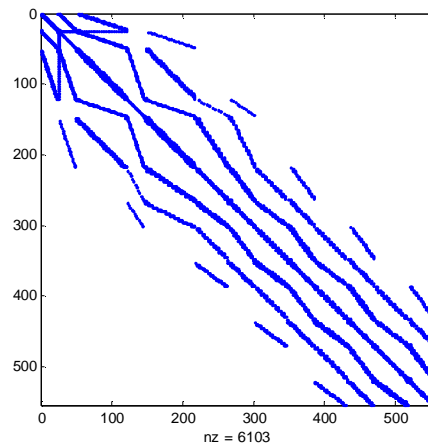


Figure 16. Pattern of the flexibility matrix

Example 4: The Finite element model a dam which is supported in a statically determinate fashion is depicted in Figure 17. The properties of the model are:

Number of Triangular elements = 386	$E = 2e+8 \text{ kN/m}^2$	$\nu = 0.3$
Number of rectangular elements = 265	$t=1\text{m}$	
Number of type I self stress systems = 1067	Number of nodes = 501	
Number of type II self stress systems = 417	$DSI_T = 1484$	

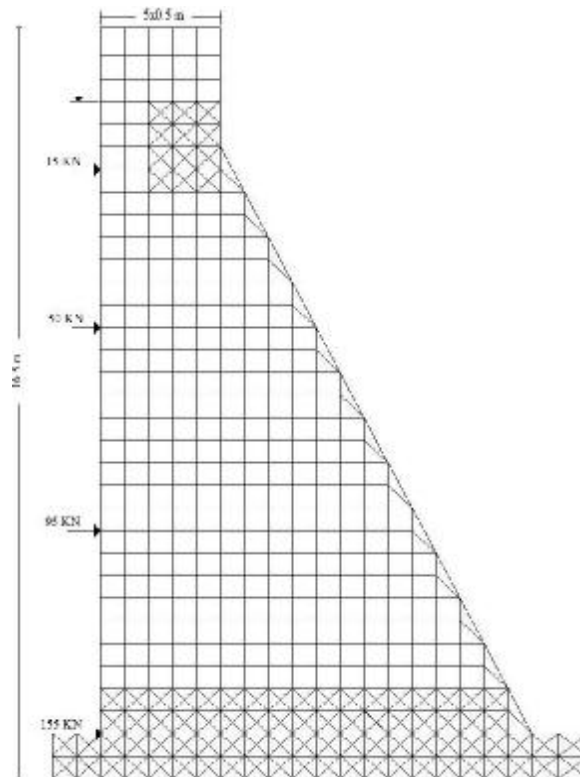


Figure 14 . A dam and the corresponding loading

Pattern of the null basis matrix with 7020 entries for the present method is shown in Figure 18. Pattern of the flexibility matrix using the present algorithms is illustrated in Figure 19. These matrices are highly sparse and banded.

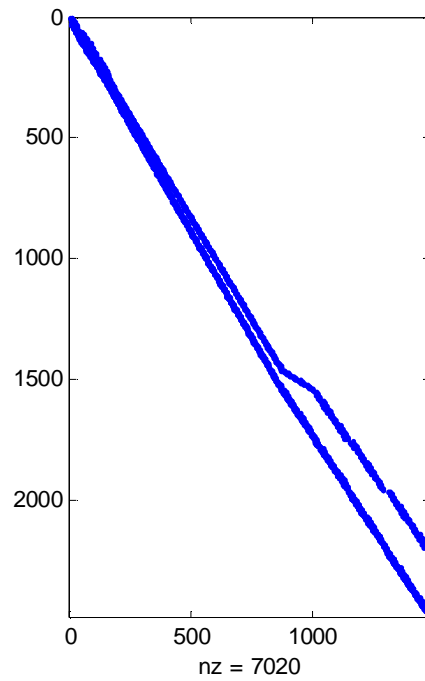


Figure 18. Pattern of the null bases matrix

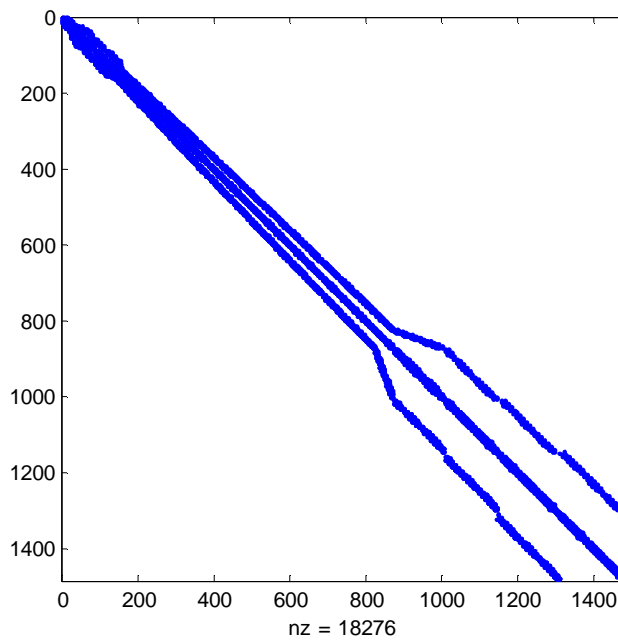


Figure 19. Pattern of the flexibility matrix

7. CONCLUDING REMARKS

Many problems are solved by the present method and the comparison of the results with those of standard FE programs indicate the high accuracy of the method. The flexibility matrices based on the selected self-stress systems are highly sparse and narrowly banded, compared to other algebraic force methods. This is due to the use of regional cycles of the natural associate graphs and appropriate ordering of the selected self-stress systems.

The comparative study of the present algorithm and other existing approaches such as LU factorization, Turn-back method, and REDUC shows the superiority of the developed graph-theoretical method. In the present algorithm, due to a considerable reduction of the floating point operations, the results have higher accuracy than the other pure algebraic methods, since nearly 80% of the null vectors are selected with algebraic operations and the remaining operations are performed on limited and definite vectors.

In the present method, numbering the nodes of a finite element model is less important and only a suitable ordering of the members of the natural associate graph is required for reducing the bandwidth of the flexibility matrices.

REFERENCES

1. Henderson J.C. de C. Topological aspects of structural analysis, *Aircraft Engineering* **32**(1960) 137-41.
2. Maunder EWA. Topological and linear analysis of skeletal structures, Ph.D. thesis, London University, Imperial College, March 1971.
3. Kaveh A. Application of topology and matroid theory to the analysis of structures, Ph.D. thesis, London University, Imperial College, March 1974.
4. Kaveh A. Improved cycle bases for the flexibility analysis of structures, *Computer Methods in Applied Mechanics and Engineering*, **9**(1976) 267-72.
5. Kaveh A. An efficient program for generating subminimal cycle bases for the flexibility analysis of structures, *Communications in Numerical Methods in Engineering*, **2**(1986) 339-44.
6. Kaveh A. A combinatorial optimization problem; Optimal generalized cycle bases, *Computer Methods in Applied Mechanics and Engineering*, **20**(1979) 39-52.
7. Cassell AC. An alternative method for finite element analysis; A combinatorial approach to the flexibility method, *Proceedings of the Royal Society of London*, **A352**(1976) 73-89.
8. Denke PH. A general digital computer analysis of statically indeterminate structures, NASATD-D-1666, 1962.
9. Robinson J. *Integrated Theory of Finite Element Methods*, John Wiley, New York, 1973.
10. Topçu A. A contribution to the systematic analysis of finite element structures using the force method (in German), Doctoral dissertation, Essen University, 1979.
11. Kaneko I, Lawo M, Thierauf G. On computational procedures for the force methods, *International Journal for Numerical Methods in Engineering*, **18**(1982) 1469-95.
12. Soyer E, Topçu A. Sparse self-stress matrices for the finite element force method,

- International Journal for Numerical Methods in Engineering*, **50**(2001) 2175-94.
13. Gilbert JR, Heath MT. Computing a sparse basis for the null space, *SIAM Journal of Algebraic and Discrete Methods*, **8**(1987) 446-59.
 14. Coleman TF, Pothen A. The null space problem I; Complexity, *SIAM Journal of Algebraic Discrete Methods* **7**(1986)527-37.
 15. Coleman TF, Pothen A. The null space problem II; Algorithms, *SIAM Journal on Algebraic and Discrete Methods*, **8**(1987) 544-61.
 16. Pothen A. Sparse null basis computation in structural optimization, *Numerische Mathematik*, **55**(1989) 501-19.
 17. Patnaik SN. Integrated force method versus the standard force method, *Computers and Structures*, **22**(1986) 151-64.
 18. Patnaik SN. The variational formulation of the integrated force method, *AIAA Journal*, **24**(1986) 129-37.
 19. Kaveh A. A combinatorial optimization problem; Optimal generalized cycle bases, *Computer Methods in Applied Mechanics and Engineering*, **20**(1979) 39-52.
 20. Heath MT, Plemmons RJ, Ward RC. Sparse orthogonal schemes for structural optimization using the force method, *SIAM Journal on Scientific Statistical. Computing*, **5**(1984) 514-32.
 21. Kaveh A, Roosta GR. Comparative study of finite element nodal ordering methods, *Engineering Structures*, Nos. 1&2, **20**(1998) 86-96.
 22. Prezemieniecki JS. *Theory of Matrix Structural Analysis*, McGraw-Hill, New York, 1968.
 23. Kaveh A. *Structural Mechanics: Graph and Matrix Methods*, Research Studies Press (Wiley), 3rd edition, Somerset, UK, 2004.
 24. Watkins DS. *Fundamentals of Matrix Computations*, John Wiley and Sons Ltd., 2nd edition, USA, 2002.
 25. Kaveh A, Koohestani K, Taghizadieh N. Efficient finite element analysis by graph-theoretical force method, *Finite Elements in Analysis and Design*, **43**(2007) 543-54.
 26. Kaveh A, Koohestani K. Efficient graph-theoretical force method for two dimensional rectangular finite element, *Communications in Numerical Methods in Engineering*, No. 11, **24**(2008) 1533-51.

APPENDIX: BASIC DEFINITIONS FROM THEORY OF GRAPHS

A graph S consists of a set $N(S)$ of elements called nodes (vertices or points) and a set $M(S)$ of elements called members (edges or arcs) together with a relation of incidence which associated with each member a pair of nodes, called its ends. The connectivity properties of a skeletal structure can simply be transformed into that of a graph S ; the joints and the members of the structure correspond to the nodes and the edges of S , respectively.

Two or more members joining the same pair of nodes are known as multiple members, and a member joining a node to itself is called a loop. A graph with no loops and multiple members is called a simple graph. If $N(S)$ and $M(S)$ are countable sets, then the corresponding graph S is finite. A graph is called planar if it can be drawn on a plane and its members intersect each other only at the nodes.

A graph S_i is a subgraph of S if $N(S_i) \subseteq N(S)$, $M(S_i) \subseteq M(S)$ and each member of S_i has

the same end nodes as in S .

A walk of S is a finite sequence $P_i = \{n_0, m_1, n_1, \mathbf{K}, m_p, n_p\}$ whose terms are alternately nodes n_i and members m_i of S for $1 \leq i \leq p$, and n_{i-1} and n_i are the two ends of m_i . A walk with distinct nodes and members is called a path. The length of a path P_i denoted by $L(P_i)$ is taken as the number of its members. P_i is called the shortest path between the two end nodes n_0 and n_p , if for any other path P_j between these nodes $L(P_i) \leq L(P_j)$. The distance between two nodes n_1 and n_2 in a graph S is taken as the length of the shortest path between them provided that the members of the path are contained in S and is denoted by $d_s(n_1, n_2)$ or $d_s(n_2, n_1)$.

Two nodes n_i and n_j are said to be connected in S if there exists a path between these nodes. A cycle is a path $(n_0, m_1, n_1, \mathbf{K}, m_p, n_p)$ for which $n_0 = n_p$ and $p \geq 3$; i.e. a cycle is a closed path. A tree T of S is a connected subgraph which contains no cycle. If a tree contains all the nodes of S it is called a spanning tree of S . A shortest route tree (SRT) rooted at a specified n_i node n_0 of S is a spanning tree for which the distance between every node of T and n_0 is minimum.