

# **Efficiency of the Use of Computer Memories and Execution Time in Structural Analysis**

Binsar H. Hariandja\*

Department of Civil Engineering, President University, Cikarang, Indonesia

Received 16 February 2023; received in revised form 08 March 2023; accepted 15 March 2023

## **Abstract**

The use of computer programming provides accurate and speedy analyses of structural systems. The accuracy and the speed of the analysis are due to the novelty of the computer, i.e., the speed in storing and retrieving large size of data, and the speed and the accuracy in the numerical computation. However, in old days, due to the limited storage capacities of old generations of the computers, and the use of limited main frame computers, used by so many parties, the efficient use of computer memory became an important issue. Moreover, the use of rented base of computer time necessitated the efficient use of computer run time. The paper discusses the efforts to efficiently use the computer memory and computer run times. These issues seem to remain important issues, even though to date, the tendency is to use of personal computers and laptops instead the use of rented main frame computers.

**Keywords:** computer run time, memory, structural analysis

## **1. Introduction**

The structural analyses involve formulation of the problem, casting the analysis in a well-defined mathematical formulation that often needs some computational tool to carry out the numerical computation. If the size of the problem is relatively small, then the use of early generations of computational tools might still be appropriate. However, if the size of the analysis becomes quite large, and the absence of a more sophisticated computational tools, the practitioners may resort to the use of simple computational tools with coarse assumptions and simplifications to render the problem into a more simplified and small-scale analysis.

The introduction of computers in early 1950s, combined with the casting of the analysis in matrix format started in 1960s [1], that particularly in tune with the use of the computers, created significant impacts on the structural analysis, in terms of the speed and the accuracy of the computations. Since then, the application of the matrix method in the analysis of structures has been developed significantly [2-6]. The impacts also include the possibility of solving large scale of structural analyses.

Say for example, to deal with the analysis of a 3D multi story building with 1 story, 10 columns in one direction and 10 lines in perpendicular direction. In 3D system, the total nodal degrees of freedom become  $10 \times 10 \times 10 \times 6 = 6000$  that leads to a  $6000 \times 6000$  structural stiffness matrix. Even the use of a super-computer may not help the problem in storing such an extremely large matrix, not yet to mention the problem in the computational task. In this case, an efficient use of storage scheme of data, becomes very important issue. The storage scheme later will have impact in the formulation and the solution methods of the analysis.

---

\* Corresponding author. E-mail address: binsar\_hariandja@president.ac.id

Tel.: +62(0)21 89109763

## 2. The Storage Scheme in the Computer Memory

### 2.1 The stack of computer memory

So simple as it may sound, the system of storing data in the memory of computer, creates very significant effects on the analysis formulation and the method of solution. To give a clear picture, consider a two-dimensional matrix,

$$[A] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (1)$$

However, the two-dimensional matrix is only for the eyes of the practitioner. In memory, the computer will stack the matrix members in line, or vector, column by column, as shown below,

$$\{A\} = \{a_{11} \ a_{21} \ a_{31} \ a_{12} \ a_{22} \ a_{32} \ a_{13} \ a_{23} \ a_{33}\} \quad (2)$$

Hence, members in a row are apart from each other, and the members in a column are close to each other.

### 2.2 The effect of the storage scheme on analysis formulation

To demonstrate the effect of this novel property of the storage scheme of data in computers, let us consider a space frame. The structure consists of several bar elements, beam elements, and columns. The number of the nodal degrees of freedom of the element is denoted by NDNL, the total nodes of element is denoted by NOEL, and the total degrees of freedom of structural node is denoted by NDNT. So, the total degrees of freedom of an element is NDEL = NDNL x NOEL. The total elements is denoted by NELT, the total nodes of the structure is NODT, and the total degrees of the structure is NDOS.

Now, element equilibrium equation is in the form,

$$[k]\{u\} = \{p\} \quad (3)$$

in which, the size of stiffness matrix  $[k]$  of the element is NDEL x NDEL, the element displacement vector  $\{u\}$  is NDEL x 1, and the element load vector  $\{p\}$  is NDEL x 1. The rotation matrix  $[R]$  connecting element displacement vector  $\{u\}$  in local coordinates and the element displacement vector  $\{U\}$  in global coordinates,

$$\{u\} = [R]\{U\} \quad (4)$$

has the sizes of NROW x NCOL, in which NROW = NDNL x NOEL and NCOL = NDNT x NOEL. The sizes of elements matrices and vectors are shown in Table 1.

Table 1 Sizes of element stiffness matrix, vectors and rotation

Element	NDNL	NOEL	NDEL	NROW	NCOL
bar	1	2	2	2	12
beam	3	2	6	6	12
grid	3	2	6	6	12
column	5	2	10	10	12

The effort to closely modeling the real structure by discrete model, force us to use several types of elements with different sizes of stiffness matrix, displacement vector, force vector and especially rotation matrix. To cope with this situation, several options may be used. However, if one opts to make analysis formulation simpler, he will use the maximum sizes, i.e., NDEL = 10, NROW = 10 and NCOL = 12. The effect on this choice on column element is moderate, but more severe on bar element. For bar element, for example, the multiplication of  $[k][R]$  will give

$$[k][R] \quad (5)$$

in which the size of  $[k]$  is 10 x10 and the size of  $[R]$  is also 10 x 10. However, the two matrices are quite sparse and time consuming and costly to operate. It is seen that the matrix multiplication process is quite simple but multiplying two quite

sparse matrices. The effort to remedy this problematic matter will be shown in next chapter.

### 2.3 Sparseness of stiffness matrix

To demonstrate this matter, let us consider a space frame shown in Fig. 1. The total elements are  $NELS = 15$ , the total  $NODT=12$ . By assuming that the stiffness matrix of an element is densely populated, then, the contribution of the element stiffness to global stiffness is limited by the maximum differences of the smallest and the largest number of degrees of freedoms in the element, it is called the bandwidth. The bandwidths of the elements are tabulated in Table 2. The maximum bandwidth of the structures is then  $MBAN = 14$  given by element 7, 11 and 12. The outlook of the global stiffness of the space frame is shown in Fig. 2.

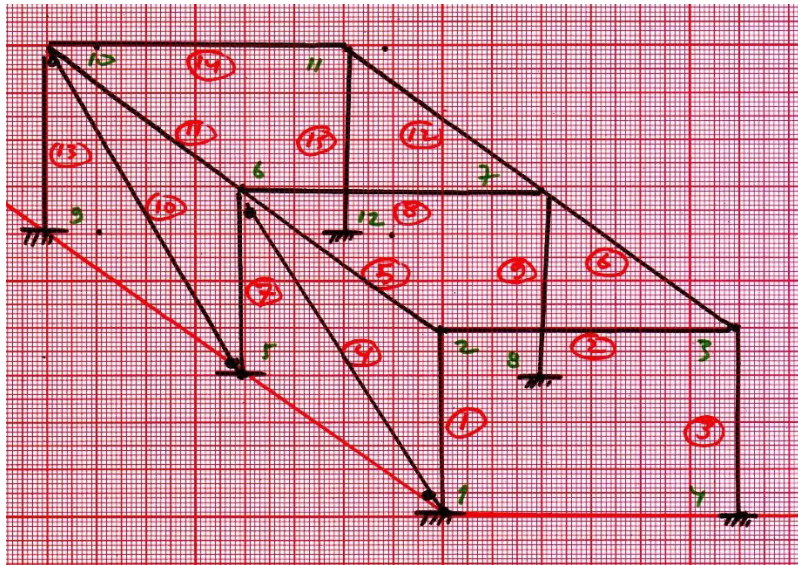


Fig. 1 Space frame

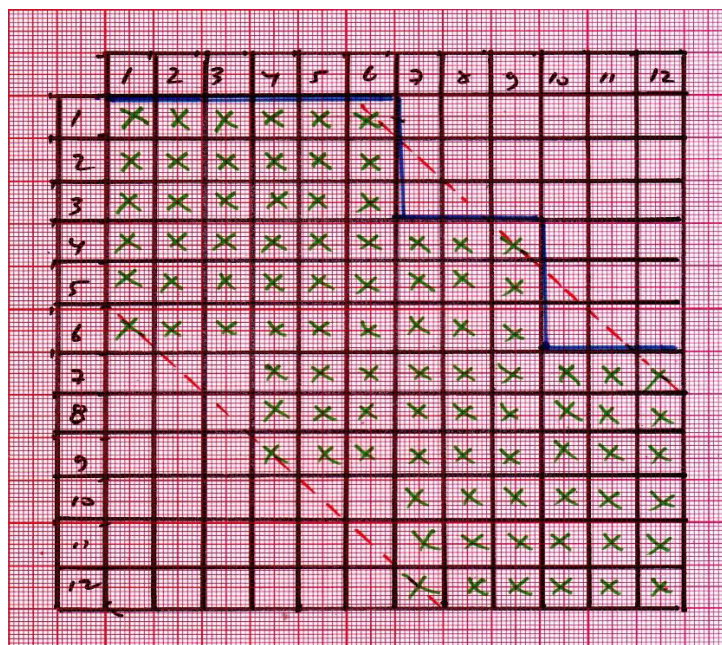


Fig. 2 Profile of stiffness matrix

It is clear that the stiffness matrix members beyond the bandwidth line are all zeroes, but they are still stored and manipulated. The remedy to this inefficiency is given in the next chapter.

Table 2 Bandwidth scheme of the global stiffness matrix

Element	Min dof	Max dof	Bandwidth
1	1	5	4
2	1	10	9
3	6	10	4
4	6	10	4
5	11	16	4
6	10	20	9
7	6	20	14
8	11	15	4
9	16	20	4
10	21	25	4
11	11	25	14
12	16	30	14
13	21	25	4
14	21	30	9
15	26	30	4

### 3. Efficiency in Structural Analysis

The effort to achieve the efficiency in using the aid of computer in structural analysis is provided in two cases, i.e., the use of variable dimension of the matrices and the storage scheme of the global stiffness of the structure.

#### 3.1 Variable dimension of arrays

Using variable dimensions of the elements, condense the matrix and minimize computational efforts. The variable dimensions of the arrays are shown in Table 3.

Table 3 Variable dimensions of arrays

Element	Type (I)	NDEL(I)	NROW(I)	NCOL(I)
bar	1	2	2	10
beam	2	6	6	10
grid	3	6	6	10
column	4	10	10	10

For example, the post multiplication of element stiffness of bar with rotation matrix becomes in a more condensed feature like in the following.

$$[k][R] = \begin{bmatrix} k_{11} & k_{22} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & & & & & & & \\ & & & r_{26} & r_{27} & r_{28} & & & & \end{bmatrix} \quad (6)$$

For demonstration, suppose a main program named SPAFRAM assigns a subroutine named ELSTIFF to compute the above post multiplication. The code is shown in Fig. 3.

```

PROGRAM SPAFRAM
.
CALL ELSTIFF(...,STE,ROT, NDEL(LTYP(I)),NROWLTYP(I)), NCOL(LTYP(I)),....)
.
SUBROUTINE ELSTIFF(...,STE,ROT,NDE,NRO,NCO,....)
REAL*8 STE(NDE,*),ROT(NDE,*),...
.
DO 30 I=1,NDE
DO 20 K=1,NCO
SUM=0.0
DO 10 J=1,NDE
SUM=SUM+STE(I,J)*ROT(J,K)
10 CONTINUE
OPR(I,J)= SUM
20 CONTINUE
30 CONTINUE

```

Fig. 3 Variable dimension of array

### 3.2 Partial storage of global stiffness matrix

In the preceding discussion it is pointed out that using a discrete model in representing a real structure in the analysis, the total degrees of freedom of structure may become large. Storing all members of the stiffness matrix may consume a large portion of computer memory. The process run may become relatively time consuming. So, the tendency is to try to store only zero members, in the starting time and along the time of execution.

Concerning the case stated above, utilize the properties of the stiffness matrix. First, in the analysis of the linearly elastic structures, the stiffness matrix is symmetric. Secondly, there are the limits beyond which all members of the stiffness matrix are zero before, along and after the execution time. Zero members are excluded out of the computer storage.

#### 3.2.1 Bandwidth scheme of the storing

Return to the space frame already given in Fig. 1 with the profile of the stiffness matrix shown in Fig. 2. Zero members beyond the upper band limit and below the lower band limit are not stored in the computer memory, leaving a skew stiffness matrix. See Fig. 2 as an explanation. The remaining skew stiffness matrix is made rectangular by translating the members,

$$k(i, j) \rightarrow k'(j - i + 1) \quad (6)$$

#### 3.2.2 Sky – line scheme

Return back to the stiffness profile in Fig. 2. In fact, not all of the bandwidth of all columns are the same but vary so as to make a sky-line limit as shown in the figure. The column height is called COLH, and the location of the main diagonal members are called LOCA. All of the members are stored in a row. The location of the main diagonal members are given by

$$LOCA(I + 1) = LOCA(I) + COLH(I); LOCA(1) = 1 \quad (7)$$

and the location of the member is then given by

$$k(i, j) \rightarrow k'(loca(i), i - j) \quad (8)$$

The storage and the execution of the stiffness matrix is more cumbersome in the case of sky-line scheme that the case of bandwidth scheme. Hence, the bandwidth scheme is more attractable and more often applied by the practitioners.

### 3.3 Numbering sequence of the nodes

If applying the bandwidth scheme in the storage system of the stiffness matrix, should pay attention so as to minimize the bandwidth of the matrix. To manage to do this by minimizing the difference between the largest and the smallest numbers of the element degrees of freedom. This can be done minimizing the numbering of the nodes of the element. Since the numbering of the nodal degrees of freedom follows the numbering of the nodes, this can be done by numbering the nodes in the direction of the shorter plan of the structure.

## 4. Example of the Application

As the examples of the application, in this chapter several structural cases are presented. One example shows the efficiency effect of applying the variable dimension of the arrays. One representing the efficiency effect due to the application of the bandwidth storing scheme. The last example demonstrated the effect of the numbering sequencing of the nodes. All of the examples are incorporated in one structure, that already given in the preceding chapter. A computer package program, written in Fortran high level language, has been developed for the use in that analysis of space framed structural systems.

#### 4.1 Effect of variable arrays

For an example, introduce the use of variable dimension of arrays in the case of multiplication of stiffness matrix and rotation matrix of a bar element,

$$[k][R] \quad (9)$$

If using the fixed maximum dimension of stiffness and rotation arrays, then NDEL = 10, NROW = 10 and NCOL = 12. The size of bar stiffness matrix is 10 x 10, and the size of the rotation matrix is also 10 x 10,

$$[k] = \begin{bmatrix} k_{11} & & k_{16} & & & \\ & & & & & \\ & & & & & \\ k_{61} & & & & k_{66} & \\ & & & & & \\ & & & & & \end{bmatrix} \quad (10)$$

and

$$[R] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & r_{66} & r_{67} & r_{68} \\ & & & & & & \end{bmatrix} \quad (11)$$

in which unfilled member locations are zero. The multiplication process involves  $10 \times 10 \times 10 = 1000$  calculations.

On the other hand, if using variable dimension, the size of the stiffness matrix is  $2 \times 2$  and the size of the rotation matrix is  $2 \times 10$ , and the multiplication of the two matrices becomes

$$[k][R] = \begin{bmatrix} k_{11} & k_{22} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & r_{26} & r_{27} & r_{28} \\ & & & & & & \end{bmatrix} \quad (12)$$

The multiplication process involves only  $2 \times 2 \times 10 = 40$  calculations. Therefore, just from a element alone, the saving due to the use of variable dimension of arrays reduces the computation run time and cost significantly.

#### 4.2 Effect of the use of bandwidth scheme in the storing of the stiffness matrix

In the example already given in the preceding discussion, the total degrees of the structure is NDOS = 30, and the bandwidth MBAN = 14. The process involving concerning the stiffness matrix consists of lower triangulation and diagonalization processes.

For fully stored stiffness matrix, the triangulation process involves NDOS x [NDOS x (NDOS+1)/2] x [NDOS x (NDOS+1)/2] = 6,486,750 calculations, whereas the use of bandwidth scheme involves only NDOS x [NDOS x (NDOS+1)/2] x MBA = 195,300 calculations. Therefore, the use of bandwidth scheme, reduces the calculations as much as [(6,486,750 – 195,300) : 6,486,750] x 100% = 97%.

#### 4.3 Effect of the numbering sequence of the nodes

It has been pointed out that the size of the bandwidth depends on the numbering sequence of the nodes, it does not depend on the numbering of the elements. For the example, again recall the preceding structure, numbering the nodes in two ways. One is the numbering in X – direction, and another in Y – direction. See Table 4 and 5 as an explanation.

Table 4 Bandwidth, numbering in X-direction

Element	Min dof	Max dof	Bandwidth
1	1	5	4
2	1	10	9
3	6	10	4
4	6	10	4
5	11	16	4
6	10	20	9
7	6	20	14
8	11	15	4
9	16	20	4
10	21	25	4
11	11	25	14
12	16	30	14
13	21	25	4
14	21	30	9
15	26	30	4

Table 5 Bandwidth, numbering in Y-direction

Element	Min dof	Max dof	Bandwidth
1	1	5	4
2	1	20	19
3	16	20	4
4	6	10	4
5	1	10	9
6	16	25	9
7	6	10	4
8	6	25	19
9	21	25	4
10	11	15	4
11	6	15	9
12	21	30	9
13	11	15	4
14	11	30	19
15	26	30	4

The first numbering scheme results in  $MBA = 14$  and the second scheme results in  $MBA = 19$ . The first scheme involves  $NDOS \times [NDOS \times (NDOS+1)/2] \times MBA = 195,300$  calculations, while the second scheme involves  $NDOS \times [NDOS \times (NDOS+1)/2] \times 19 = 2,650,500$  calculations. The reduction is  $[(265,050 - 195,300)/265,050] \times 100\% = 26\%$ . The most efficient scheme is the numbering in the direction of the shortest direction of the structural plan.

## 5. Conclusions

This paper has been discussed about the efficiency of the use of computer storage and the run time of the computer program in the analysis of the structures. Based on the above discussion and examples of the application, several conclusions are derived as follows. The storage scheme of the arrays data in the computer memory, affects the efficiency of the computer execution by the computer.

- The use of variable dimension of the arrays in the computer program significantly reduces the memory allocation and the run time of computer.
- The use of bandwidth and sky-line scheme in storing stiffness matrix of the structure also significantly reduces the memory allocation and the run time of computer.
- To make the bandwidth and hence to make the storage memory smaller and the execution time shorter, the numbering of the structural nodes may be carried out in the shortest direction of the structural plan.

## **Acknowledgments**

The author wishes to his appreciation to the Head of the Department of Civil Engineering by supporting the subscription to the Fortran Application. The excellent drawings are carried out by Mrs. Okke Tumbuan, to which the author also extends his sincere appreciation.

## **References**

- [1] J. M. Argyris, and S. Kelsey, *Energy Theorems in Structural Analysis*, London: Butterworth, 1960.
- [2] B. H. Hariandja, *Analisis Struktur Berbentuk Rangka Dalam Formulasi Matriks*, Bandung: Penerbit Aksara Hutasada, 1997.
- [3] H. C. Martin, *Introduction to Matrix Method of Structural Analysis*, New York: McGraw-Hill Company, 1966.
- [4] M. D. Vanderbielt, *Matrix Structural Analysis*, New York: Quantum Publishers, Inc., 1975.
- [5] W. Weaver Jr. and J. M. Gere, *Matrix Analysis of Framed Structures*, New York: Van Nostrand Company, 1980.
- [6] N. Willems and W. M. Lucas Jr., *Matrix Analysis for Structural Engineers*, Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1968.