Investigation of Rigid Frame by Integrated Force Method

Saeid Kamkar

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Civil Engineering

Eastern Mediterranean University
February 2010
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director (a)

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Civil Engineering.

Prof. Dr. Ali Gunyakti
Chair, Department of Civil Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Civil Engineering.

Asst. Prof. Dr. Erdinc Soyer
Supervisor

Examining Committee

1. Asst. Prof. Dr. Erdinc Soyer
2. Asst. Prof. Dr. Giray Ozay
3. Asst. Prof. Dr. Serhan Sensoy

ABSTRACT

When an indeterminate frame is acted on by external loads and couples internal stresses in the members and displacements at the joints will be produced. Calculation of the stresses in the members and joint displacements are needed in structural design. The two main methods for the analysis of frames are the classical force method and the displacement method. A third method called the Integrated Force Method (IFM) has been developed for analysis of structural mechanics problems. This method treats all independent internal forces as unknown variables that can be calculated by simultaneously coupling both equilibrium equations and compatibility conditions. The advantage of the Integrated Force Method is that compatibility conditions are generated by algebraic operations on the nodal equilibrium equations. These algebraic operations are Null Space of the equilibrium matrix and the Singular Value Decomposition of the equilibrium matrix. These algebraic operations are easily carried out by computer algebra systems like Mathematica7. An extension of the Integrated Force Method is the Dual Integrated Force Method (IFMD). The primary unknowns in the Dual Integrated Force Method are the joint displacements. To obtain the joint displacements the structure global stiffness matrix is generated by using the equilibrium matrix and the unconnected stiffness matrix of the structure. The advantage of the Dual Integrated Force Method is that the global stiffness matrix is generated easily by computer algebra systems like Mathematica7 by simple matrix multiplication commands.

In this thesis two analysis packages for analysis of indeterminate rigid frames by Integrated Force Method (IFM) have been developed. These packages are: Integrated Force Method via Null Space, Integrated Force Method via Singular Value Decomposition. One additional analysis package using Dual Integrated Force Method have been developed. All three analysis packages have been coded using Mathematica7.

In all three analysis packages all the calculation steps are presented, explained and they can be edited according to the needs of the user. The user can see the program code and its corresponding output at each calculation step. All of these specific characteristics make the analysis packages useful and practical.

Many problems have been analyzed by these packages and all the results have been compared with the results obtained from Mastan2 v3.2. All the results fully agree.

Keywords: Frame Analysis, Integrated Force Method (IFM), Dual Integrated Force Method (IFMD), Compatibility conditions.

ÖZ

Hiperstatik çerçeveler, yükler ve momentlere maruz kalınca elemanlarda iç kesit zorlamaları ve düğüm noktalarında deplasmanlar oluşur. Çerçeve tasarımı için elemanlardaki iç kesit zorlamaları ve düğüm noktalarındaki deplasmanlara gereksinim vardır. Çerçeve analizinde iki ana yöntem klasik kuvvet ve deplasman yöntemleridir. Üçüncü bir metod olarak Bileşik Kuvvet Metodu (IFM) geliştirilmiştir. Bu yöntemde serbest iç kuvvetler ana bilinmeyenler olarak kabul edilir ve denge denklemlerini uygunluk şartları ile birleştirerek hesaplanır. Bileşik Kuvvet Metodunun avantajı uygunluk şartlarının denge denklemlerine cebirsel operasyonlar yapılarak elde edilmesidir. Bu cebirsel operasyonlar denge matrisine Null Space veya Singular Value Decomposition yöntemleri uygulanmasıdır. Bu cebirsel yöntemler bilgisayar cebirsel sistemlerince, Mathematica7 gibi, kolayca yapılabilir. Çift Bileşik Kuvvet Metodu (IFMD), Bileşik Kuvvet Metodunun bir uzantısı olarak geliştirilmiştir. Çift Bileşik Kuvvet Metodunda ana bilinmeyenler düğüm noktalarındaki deplasmanlardır. Düğüm noktalarındaki deplasmanları hesaplamak için çerçevenin global rijitlik matrisi bulunmalıdır. Global rijitlik matrisi ise yalnızca denge matrisi ve çerçevenin bileşmemiş global rijitlik matrisi kullanılarak elde edilir. Çift Bileşik Kuvvet Metodunun avantajı çerçevenin global rijitlik matrisi bilgisayar cebirsel sistemlerince, Mathematica7 gibi, basit matris çarpımları kullanılarak elde edilmesidir.

Bu tezde Bileşik Kuvvet Metodunu kullanarak iki analiz paketi geliştirilmiştir. Bu iki analiz paketi sırası ile Null Space ve Singular Value Decomposition yöntemlerini

kullanarak uygunluk matrislerini elde eder. Bir ilave analiz paketi de Çift Bileşik Kuvvet Metodunu kullanarak geliştirilmiştir. Her üç analiz paketi de Mathematica7 proglamlama dilini kullanarak yazılmıştır. Üç analiz paketinde hesaplama adımları takdim edilmiş ve anlatılmıştır. Hesaplamalar kullanıcının ihtiyaçlarına göre değiştirilebir. Kullanıcılar programların tüm yazılımlarını görebilir ve her hesaplama basamağında cevaplar ekrana yansıtılır. Tüm bu özellikler bu üç analiz paketlenin kullanışlı ve pratik olduğunu gösterir.

Üç analiz paketinde hesaplama adımları takdim edilmiş ve anlatılmıştır. Hesaplamalar kullanıcının ihtiyaçlarına göre değiştirilebir. Kullanıcılar programların tüm yazılımlarını görebilir ve her hesaplama basamağında cevaplar ekrana yansıtılır. Tüm bu özellikler bu üç analiz paketlenin kullanışlı ve pratik olduğunu gösterir.

Bu üç analiz paketini kullanarak birçok problem analiz edilmiş ve sonuçlar Mastan2 v3.2 analiz programının sonuçları ile karşlaştırılmıştır. Tüm sonuçlar birbirleri ile tamamen uyuşma içerisidedir.

Anahtar kelimeler: Çerçeve analizi, Bileşik Kuvvet Metodu, Çift Bileşik Kuvvet Metodu, Ugunluk şartları.

To My Family

ACKNOWLEDGMENT

I would like to thanks my express deepest appreciation to my supervisor Asst. Prof. Dr. Erdinc Soyer for his great efforts in guiding and acquainting me throughout this work.

I would like to express my sincere gratitude to my family for their support and encouragement.

I wish to express my special thanks for all the members of the Civil Engineering Department at Eastern Mediterranean University.

I would like to thanks all my friends in North Cyprus, for their friendship and hospitality.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

A	Area of Member
$([B]^{T})^{pinv}$	Moore-Penrose Inverse of $[B]^{T}$
[B]	Equilibrium Equations Matrix
[C]	Compatibility Matrix
[CC]	Compatibility Conditions
E	Modulus of Elasticity
{F}	Internal Forces Vector
[G]	Unconnected Flexibility Matrix
[g]	Member Flexibility Matrix
[I]	Identity Matrix
I	Moment of Inertia
i	Starting Node of Element
IE	Internal Strain Energy
[J]	n Rows of $[[S]^{-1}]^{T}$
[K]	Global Stiffness Matrix
L	Length of Element
[M]	Singular Value Decomposition Matrix
$[M_u]$	Orthogonal Matrix
$[M_v]$	Orthogonal Matrix
$[M_б]$	Diagonal Matrix
m	Number of elements
n	Number of Nodes
[nls]	Null Space Matrix
{P}	External Loads

$\{P^*\}$	IFM Load Vector
Q	Basic Forces
q	Member End Forces
$\{X\}$	Displacements Vector
W	Work Done
$\{\beta\}$	Deformation Vector
r	Degree of Indeterminacy
$\{\delta R\}$	Initial Deformation Vector

LIST OF ABBREVIATIONS

dof	Displacement Degree of Freedom
DDR	Deformation Displacements Relations
EE	Equilibrium Equations
fof	Force Degree of Freedom
FDR	Force Deformation Relation
IFM	Integrated Force Method
IFMD	Dual Integrated Force Method
SFM	Standard Force Method
SM	Stiffness Method
SVD	Singular Value Decomposition

Chapter 1

INTRODUCTION

1.1 Introduction

When a structure is built, it should be in equilibrium. According to Newton's law all the particles (elements and joints) must be in equilibrium. For making sure of equilibrium the external loads and internal must be in balance. The equations of equilibrium of structure establish the relation between independent element forces and applied forces at the global degrees of freedom. Finding these equations in small structures is very easy but in large scale structures is manually very difficult and time consuming [1].

On type of structure is frame. A frame is made of straight or linear members. Indeterminate frame is obtained by extra restraints at the support of determinate frame or by increasing the number of frame members. A frame member carries the three independent internal forces: Axial Forces, Shear Force and a Bending Moment. Nodes of a frame have three displacements: translation, transverse displacement and rotation. A frame can be formulated by combining the action of the bar and beam members. The analysis of frame structure is considerably more difficult than that of beam or truss. The complexity is greater because of more algebraic equations required [2].

Two main methods for structural analysis of frame are:

- Stiffness method
- Force method

Stiffness method uses displacements as primary unknowns and force method uses forces as primary unknowns. An alternative formulation, termed the Integrated Force Method has been developed to analyze problems in structural mechanics [4].

In the Integrated Force Method (IFM) all independent forces, not just the redundants, are treated as unknown quantities that can be calculated by simultaneously coupling both equilibrium equations and compatibility conditions together [4]. The usage of all independent forces provides in the solution process provides a great advantage over the classical force method where only the redundant are solved for by using the compatibility equations. Selection of redundants and hence the generation of the compatibility equations in the classical force method require lengthy calculation processes using matrix algebra techniques. In the Integrated Force Method, on the other hand, the process of selection of redundants is not required and independent internal forces for the structural design process is determined for all the members is obtained in only one solution process. Thus the Integrated Force Method provides considerable advantage in the design of large scale structures like in the aeronautical industry and in the design of steel structures. IFM's advantages [3] over the Stiffness Method (SM) have been documented, including accurate stress results, a well-conditional system for finite element discrete analysis, fast convergence to correct solutions, and elegancy initial deformation problems. The Integrated Force Method has also been extended to nonlinear structural analysis [19] and optimization [21].

Generation of the compatibility conditions for the Integrated Force Method has been made easier by making use of the algebraic methods of Null Space and Singular Value Decomposition [12], [14], [15]. When applied to the equilibrium matrix the Null Space and Singular Value Decomposition yield directly the compatibility matrix. After generating the unconnected flexibility matrix the Null Space or Singular Value Decomposition is multiplied with the unconnected flexibility matrix to yield the compatibility conditions of the Integrated Force Method. The Null Space and Singular Value Decomposition are obtained by simple programming using computer algebra system Mathematica7. Therefore in the analysis of skeletal structures it is very advantageous to use the Integrated Force Method and generate the compatibility matrix via Null Space or via Singular Value Decomposition.

An extension of the Integrated Force Method is the Dual Integrated Force Method, IFMD, which is essentially a displacement method. In IFMD, only the equilibrium matrix and the unconnected stiffness matrix of the structure are used to generate the global stiffness matrix of the structure. Thus there is no need to write lengthy and complex computer programs to generate global stiffness matrix of the structure. The global stiffness matrix of the structure in IFMD is obtained by simple programming using computer algebra system Mathematica7. Therefore in the analysis of skeletal structures it is very advantageous to use the Dual Integrated Force Method

1.2 The Research Problem

The main purpose of this thesis is to write structural analysis packages to analyze coplanar rigid frames with Integrated Force Method via null space, Integrated Force Method via singular value decomposition and Dual Integrated Force Method. These programs are easy to use and useful for students, instructors, engineers and researchers to analyze the frame and better understand the procedure of above methods. The three packages are generated by computer codes automatically with Mathematica7 software. Various problems have been analyzed by these three methods and the results have been compared by Mastan2 v3.2. The results obtained by using the IFM and IFMD in full agreement with results obtained from Mastan2 v3.2.

This thesis also explains the theory for these three methods step by step. These explanations include the generation of the:

- equilibrium equations
- unconnected flexibility matrix
- compatibility conditions
- main IFM matrix
- nodal displacements
- member end forces
- support reactions

Also the axial force diagram, shear force diagram and bending moment diagram for each element are drawn.

1.3 Thesis Limitations

These programs do not include thermal, triangular, trapezoidal loading and support settlements.

1.4 Thesis organization

Chapter 2 presents background information about equilibrium equations, integrated force method via null space, integrated force method via singular value decomposition, and dual integrated force. This chapter also discusses theory of these three methods and explain their usage for the analysis of rigid frames.

Chapter 3 presents about the research problem, reason of research, desired characteristic of the program and show the solution approaches for each method.

Chapter 4 presents the way to assemble the equilibrium equations automatically. It gives algorithm to generate the equilibrium equations automatically.

Chapter 5 introduces the algorithms for following methods:

- Integrated force method via null space.
- Integrated force method via singular value decomposition.
- Dual integrated force method.

The aims of this chapter are:

Use algorithms to write computer codes, to be able to compare these methods and to be able to compare the matrix plot of the matrices obtained in these methods.

In Chapter 6 three illustrative examples are solved to present more details of the three methods and the results obtained by using the three packages are compared with results obtained by using Mastan2 v3.2.

Chapter 7 gives conclusions.

Chapter 2

BACKGROUND INFORMATION

2.1 Introduction

A novel formulation termed the "integrated force method" (IFM) has been developed for analyzing structures. In this method all the internal forces are taken as independent variables, and the system equilibrium equations (EE's) are integrated with the global compatibility conditions (CC's) to form the governing set of equations. In IFM no choices of redundant load systems have to be made, in contrast to the standard force method (SFM). This property of IFM allows the generation of the governing equation to be automated straightforwardly, as it is in the popular stiffness method (SM). Overall this new version of the force method produces more exact results than the stiffness method for comparable computational cost.

2.2 A Brief Review of Integrated Force Method

A discrete or discredited structure for analysis can be designated as structure (n,m) where "structure" denotes type of structure (truss, frame, plate, shell, or their combination discredited by finite elements) and *n, m* are force and displacement degrees of freedoms (fof, dof), respectively. The structure (n, m) has m equilibrium equations and r = (n-m) compatibility conditions. The m equilibrium equations

$$[B]\ \{F\} = \{P\} \tag{2.1}$$

and the r compatibility conditions

$$[C][G]\{F\} = \{\delta R\} \qquad 2.2$$

are coupled to obtain the governing equations of the IFM as

The basic equation of the integrated force method is:

$$\begin{bmatrix} [B] \\ [CC] \end{bmatrix} \{F\} = \begin{Bmatrix} P \\ \delta R \end{Bmatrix} \qquad 2.3$$

Or

$$[S]\{F\} = [P^*] \qquad 2.4$$

Where, matrix [B] is equilibrium equations matrix, [C][G] is compatibility conditions, vector {F} is internal forces, vector {P} is external mechanical loads and vector {δR} is initial deformation. The matrix [S] is the governing equations of the IFM. The column matrix $\{P^*\}$ contains applied loads as well as initial deformations, zeros will be added if there are no initial deformations [6].

According to reference [3], the procedure of analysis of a structure with Integrated Force Method is as follows:

1. Generate the Equilibrium Equations [B].
2. Generate the unconnected flexibility matrix [G].
3. Generate the Compatibility Matrix [C].
4. Generate the Compatibility Conditions [CC] = [C] [G].
5. Generate the IFM load vector $\{P\}^*$ of dimension *n*.
6. Couple compatibility conditions with equilibrium equations and find [S].
7. Solve for unknown internal forces {F}.

In the integrated force method, the forces are primary unknowns, and displacements {X} do not appear in this system of equation, but if it is needed it can be back calculated. In the integrated force method the displacements are calculated as:

$$\{X\} = [J][G][F] \qquad 2.5$$

Where, {X} is n component nodal displacement vector and [J] is n rows of $[[S]^{-1}]^{T}$ (n is number of equations or number of unrestrained degree of freedom).

According to Reference [3, 5].the compatibility conditions are obtained by eliminating displacements from deformations displacements (DDR) of the structure, therefore the compatibility conditions can be derived in two steps:

- Derive deformation displacements relation (DDR).
- Eliminating the displacement from the deformation displacement relationships.

The deformations displacements relations are derived on an energy bases. The internal energy IE which is sorted in structure is:

$$IE = \frac{1}{2}\{F\}^{T}\{\beta\} \tag{2.6}$$

The deformations (β_1, β_2,… β_m) are elongation in frame analysis correspond to the internal element forces (F_1, F_1, …,F_m), respectively.

Moreover, the work done W, by the external loads in structure is:

$$W = \frac{1}{2}\{P\}^{T}\{X\} \tag{2.7}$$

The n displacements of unrestrained joints (X_1, X_2… , Xn) correspond to n external loads (P_1, P_2, …, P_n), respectively.

According to the work-energy theorem:

$$IE = W \tag{2.8}$$

Therefore, using Equations 2.6, and 2.7

$$\frac{1}{2}\{F\}^{T}\{\beta\} = \frac{1}{2}\{P\}^{T}\{X\} \tag{2.9}$$

In Equation 2.9, there are m internal forces and deformation with n external forces and displacements, and by substituting governing equilibrium Equation 2.3 into Equation 2.9.

$$\{F\}^{T}(\{\beta\} - [B]^{T}\{X\}) = 0 \tag{2.10}$$

And since the internal forces {F}, is not null vector, therefore, deformation displacement relation will be obtain as:

$$\{\beta\} = [B]^{T}\{X\} \tag{2.11}$$

Equation 2.11 expresses m deformations in terms of n displacements and according to References [3, 5], by eliminating the displacements from the deformation displacement relation (Equation 2.10), to obtain $r = m - n$ compatibility condition as:

$$[C]\{\beta\} = \{0\} \tag{2.12}$$

The compatibility condition has $r = m - n$ rows with m columns and it is full row rank.

2.3 A Brief Review of Dual Integrated Force Method

Like the IFM, the dual method also has two sets of equations. The first set is used to calculate displacement, while the second set back calculates forces. The primal and dual methods produce identical solutions for force, and displacement, Patnaik [3, 4 and 9]. The IFMD governing equations are:

$$[\mathbf{K}]\{X\} = \{P\} \tag{2.13}$$

$$\{F\} = [\mathbf{G}]^{-1}[\mathbf{B}]^{T}\{X\} - [\mathbf{G}]^{-1}\{\beta\}^{0} \tag{2.14}$$

Where

$$[K] = [B][G]^{-1}[B]^{T} \tag{2.15}$$

According to Reference [9] the procedure of analysis of a structure with Dual Integrated Force Method is as follows:

1. Generate the Equilibrium Equations [B].

2. Generate the unconnected flexibility matrix [G].
3. Generate the $[G]^{-1}$.
4. Generate the Stiffness Matrix [K].
5. Generate the load vector {P}of dimension n.
6. Solve for unknown displacements {X}.
7. Solve for unknown internal forces {F}.

2.4 State of the Art in the Integrated Force Method

- The main approaches to the integrated force method are:

  Nonlinear analysis using the integrated force method [19] IFM method is used for analyzing nonlinear structures. General formulation of nonlinear structural analysis is given. Typically highly nonlinear bench-mark problems are considered. The characteristic matrices of the elements used in these problems are developed and later these structures are analyzed.

- Generation of the equilibrium matrix and the compatibility matrix automatically [18], [14], [15]. The compatibility matrix is obtained by using algebraic methods. In both of the References [14] and [15] the Nullspace and the Singular Value Decomposition of the equilibrium matrix is carried out to determine the compatibility matrix [CC].

2.4.1 Using Null Space to Obtain Compatibility Conditions

According to Reference [3], the null space of equilibrium equations is used to obtain compatibility condition. According to reference [5] for generating the compatibility conditions null space of the equilibrium equations should be coupled with unconnected flexibility matrix. Unconnected flexibility matrix [G] is a symmetric matrix which is a block diagonal matrix. For generating the unconnected flexibility matrix first the member flexibility matrix [g] should be obtained.

$$g = \begin{bmatrix} \frac{L}{AE} & 0 & 0 \\ 0 & \frac{L^3}{3EI} & \frac{L^2}{2EI} \\ 0 & \frac{L^2}{2EI} & \frac{L}{EI} \end{bmatrix} \qquad 2.16$$

Where, L is the length of member, I moment of inertia of member, A is area of member and E is modulus of elasticity. Then, placed each member flexibility matrix in the diagonal blocks of unconnected flexibility matrix [G].

$$[G] = \begin{bmatrix} \begin{bmatrix} \frac{L}{AE} & 0 & 0 \\ 0 & \frac{L^3}{3EI} & \frac{L^2}{2EI} \\ 0 & \frac{L^2}{2EI} & \frac{L}{EI} \end{bmatrix} & & 0 \\ & \begin{bmatrix} \frac{L}{AE} & 0 & 0 \\ 0 & \frac{L^3}{3EI} & \frac{L^2}{2EI} \\ 0 & \frac{L^2}{2EI} & \frac{L}{EI} \end{bmatrix} & \\ & & \ddots \\ 0 & & \begin{bmatrix} \frac{L}{AE} & 0 & 0 \\ 0 & \frac{L^3}{3EI} & \frac{L^2}{2EI} \\ 0 & \frac{L^2}{2EI} & \frac{L}{EI} \end{bmatrix} \end{bmatrix} \qquad 2.17$$

Or

$$G = \begin{bmatrix} g1 & & 0 & \\ 0 & g2 & & 0 \\ & & \ddots & \\ & 0 & & gn \end{bmatrix} \qquad 2.18$$

According to reference [2, 3, 4, 5] compatibility conditions obtained with combining the unconnected flexibility matrix (G) with null space of the equilibrium equations.

$$[CC] = [G]\,[C] \qquad 2.19$$

Where, matrix [C] is Compatibility Matrix obtained form null of equilibrium equations.

According to reference [2, 3, 4, 5] the compatibility conditions be coupled with equilibrium equations to solve for unknown internal forces.

2.4.2 Using Singular Value Decomposition to Obtain Compatibility Conditions

According to references [7, 8], there is another alternative to find the compatibility condition by using the singular value decomposition of the matrix [M] which is:

$$[M] = [I] - [B]^{T}\,([B]^{T})^{pinv} \qquad 2.20$$

Where, [I] is the identify matrix and the number of its rows and its columns are equal to the number of elements. The matrix $[B]^{T}$ is transpose of equilibrium equations, and $([B]^{T})^{pinv}$ is the Moore-Penrose pseudo inverse of $[B]^{T}$ as:

$$([B]^{T})^{pinv} = ([B]\,[B]^{T})^{-1}\,[B] \qquad 2.21$$

Singular value decomposition is applied to matrix [M] to obtain:

$$[M] = [M_u]\,[M_б]\,[M_v]^{T} \qquad 2.22$$

Where, $[M_u]$ and $[M_v]$ matrices are orthogonal matrix and number of its rows and its column are equal to number of frame elements, and the matrix $[M_б]$ is

$$[M_б] = \begin{bmatrix} \Lambda & 0 \\ 0 & 0 \end{bmatrix} \qquad 2.23$$

Where, it is square and number of its rows and columns are equal to the number of frame elements and:

$$\mathbf{\Lambda} = \text{diag}\,(\mathbf{\Lambda}_1, \mathbf{\Lambda}_2 \ldots \mathbf{\Lambda}_\rho) \qquad 2.24$$

And ρ is degree of indeterminacy and

$$\mathbf{\Lambda}_1 \geq \mathbf{\Lambda}_2 \geq \ldots \geq \mathbf{\Lambda}_\rho > 0 \qquad 2.25$$

According to Reference [7] it follows that:

$$[M] = [M_u] \begin{bmatrix} [C] \\ [0] \end{bmatrix} \qquad 2.26$$

$$[CC] = [C]\,[G] \qquad 2.27$$

Where, [CC] is compatibility condition matrix and will be computed by Equations 2.26 and 2.27.

The compatibility condition [CC] which is obtained by using singular value decomposition of $([B]^{T)})^{pinv}$ matrix, and also obtained from null space of equilibrium equation may not be banded whilst for small structure it will not cause any problems for large structure it may be numerical expensive.

2.5 Equilibrium Equations

2.5.1 Basic System

This chapter started discussion with a two dimensional frame, it has 2 elements and 3 nodes, as shown in Figure 2.1 this frame is describes in global reference system X-Y-Z. This example is taken from reference [1].

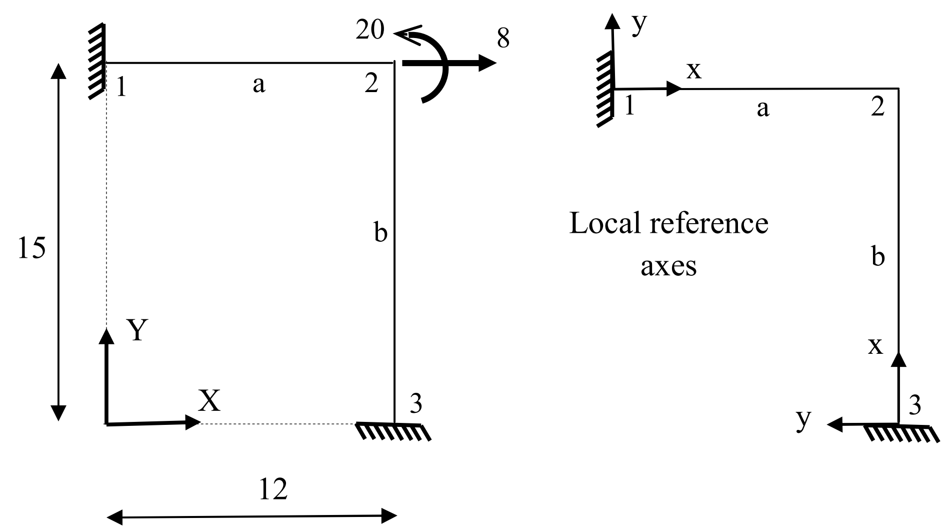


Figure 1: Two dimensional frame, definition of local reference axes

To obtain the equilibrium equations should be isolated joints from this model by imaginary cuts at the member ends and obtain 5 free bodies: 3 joints free bodies and 2 elements free bodies. So, three pair forces revealed at every cuts. A pair of forces parallels two to the element axis, a pair of forces normal to the element axis and pair of moments. It is important to decide on a consistent sign-conventional at the starting of this analysis. For this reason selecting a local X-Y-Z coordinate system for each element is necessary. To do this, should be started, numbering the two nodes of each element with (i) and (j).for start node of element put (i) and for end node of element put (j). start from node (i) and end with node (j) and decide to the x-axis in the positive local system for each element is oriented from node (i) to node (j). In local coordinate system node (i) for element 1 is node 1 and node (i) for element 2 is 3 and the forces at the element side of each cut is positive and the moment is positive when acting counterclockwise [1].

2.5.2 Formulation of Equilibrium Equations

The nodes and element free bodies as well as the forces at the cuts are shown in figure 2. The element end forces at this figure are denoted by q, this Figure is the complete set of element end forces in the local reference system.

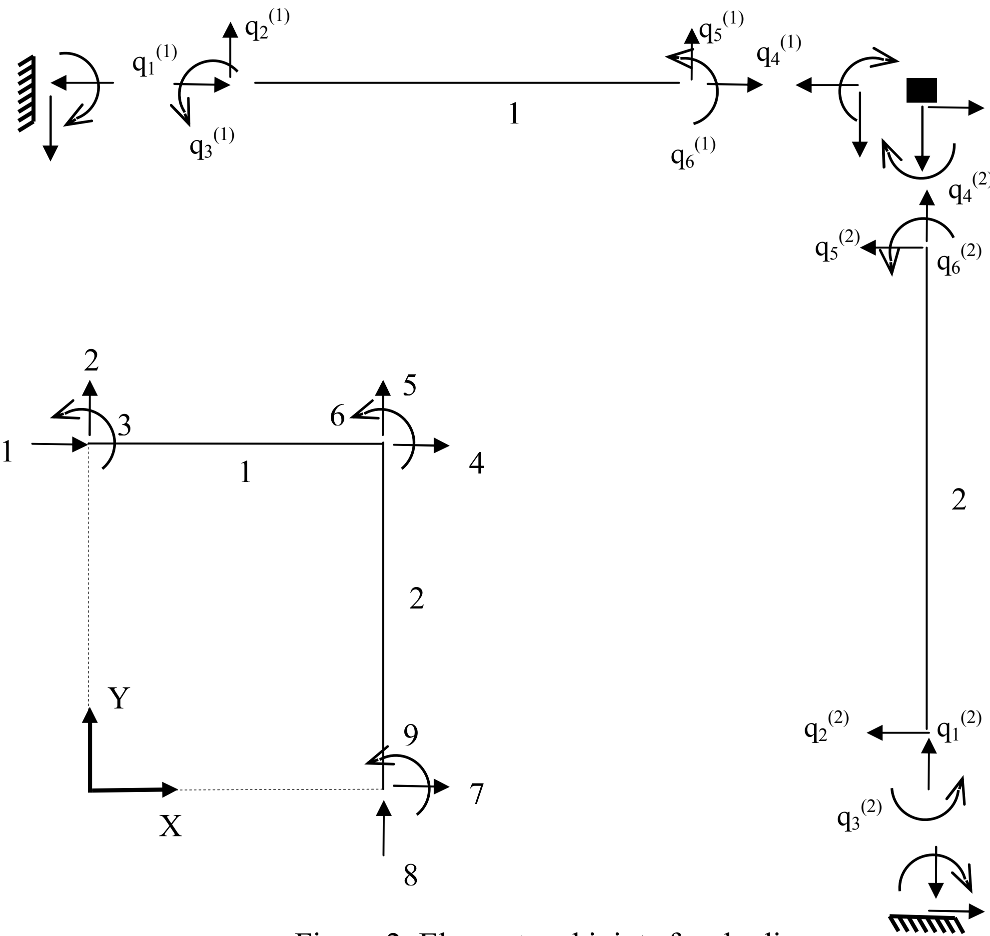


Figure 2: Element and joints free bodies

The components of the q vector are numbered as follow: Start from element node (i), the first degree of freedom in local X (first component), then the degree of freedom in local Y (second component), then the degree of freedom about local Z (third component) and then move with the same order to node (j) of the element.

From the equilibrium of the element free body should be concluded that the components of the complete end force vector q are not independent. Their dependence is provided by the equations of equilibrium of the element free body [1]. It can write the following three equations of equilibrium:

1. Sum of the forces in the local X-axis.
2. Sum of the forces in the local Y-axis.
3. Sum of the moments about end node (i) of the element.

In the absence of element loads these three equations yield:

$$q_1 = -q_4 \qquad 2.28$$

$$q_2 = -q_5 \qquad 2.29$$

$$q_3 = -q_6 - q_5 L \qquad 2.30$$

Where, L is the element length. These dependence relations suggest that the end forces q_4, q_5, q_6 can be selected as basic forces Q_1, Q_2, Q_3. This choice is illustrated in Figure 3.

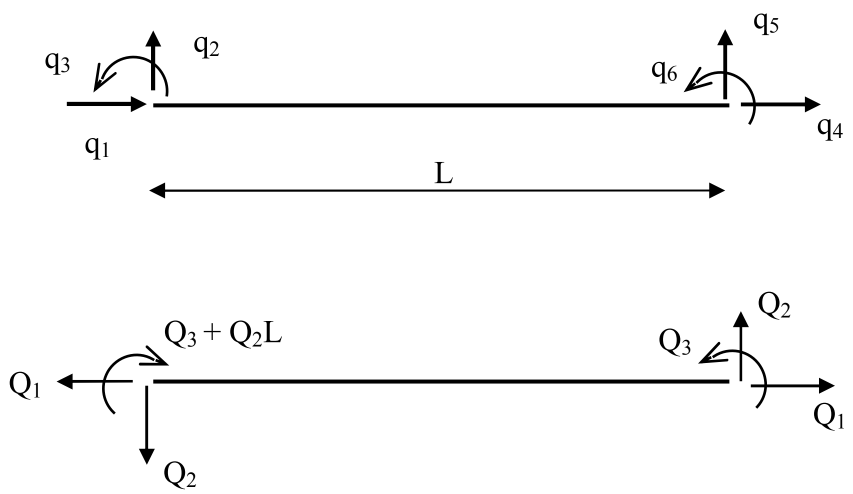


Figure 3: Element and joints free bodies

The relation between complete end force vector q and basic force vector Q can be derived directly from figure 3.

$$.q_1 = -Q_1 \qquad 2.31$$

$$q_2 = -Q_2 \qquad 2.32$$

$$q_3 = -Q_3 - Q_2\,L \qquad 2.33$$

$$q_4 = Q_1 \qquad 2.34$$

$$q_5 = Q_2 \qquad 2.35$$

$$q_6 = Q_3 \qquad 2.36$$

or, in matrix form

$$q = \begin{Bmatrix} q1 \\ q2 \\ q3 \\ q4 \\ q5 \\ q6 \\ q7 \\ q8 \\ q9 \end{Bmatrix} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -L & -1 \\ 1 & 0 & 0 \\ 0 & 1. & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad 2.37$$

this relation can reduced the complete set of element end forces to the basic set. The equilibrium equations that are remain to be satisfied at the joint free bodies [1].

To this end first all free dofs and then restrained dofs are numbered. To each global dof corresponds an equilibrium equation of generalized forces in global reference system. This example has three equations of equilibrium at the free global degrees of freedom:

$$P_1 = q_4^{(a)} - q_5^{(b)} \qquad 2.38$$

$$P_2 = q_5^{(a)} + q_4^{(b)} \qquad 2.39$$

$$P_3 = q_6^{(a)} + q_6^{(b)} \qquad 2.40$$

Where, a superscript in parentheses denotes the element, a subscript denotes the member end force and P_f is applied force at the free global dofs.

This example has three equilibrium equations and six unknown forces. It is therefore, the equilibrium equations matrix is not square so it is impossible to solve this system of equations for given applied forces P_f at the free degree of freedom, namely [1].

This system is called statically indeterminate. The degree of static indeterminacy is the difference between the number of columns and number of rows of the equilibrium matrix of the structure at the free dofs. .This implies that the degree of static indeterminacy is the difference between number of basic element forces and available equations of equilibrium at the free dofs [1].

2.6 Previous Work Done

Schottler R. has developed Java applets for analysis of trusses, beams and frames in reference [10]. The java programs known as applets are embedded in HTML document. They provide good examples of application of objected- oriented programming and development of software for graphical user interface.

Moh'd M. D. has developed a computer program in order to analyze two dimensional frame in Reference [11]. The nodal equilibrium equations, the force displacement relations and geometric compatibility relations are automatically generated by a FORTRAN computer program. In this work the mixed formulation and stiffness formulation are used. The computer program developed is capable of handling temperature and imperfections effects, support settlements and flexible support effects.

Sensoy S. has developed a computer program for two dimensional structural analysis by applying the Gauss-Jordan elimination procedure on the equilibrium equations, according to various pivot selection strategies. In this work, Integrated Force Method and Conventional Flexibility Method formulations are revised so that support reactions are also included among the unknowns. These formulations are further revised by including all the member end forces among the unknowns.

Under M. has developed a computer program for two dimensional frame analysis in Reference [13]. In this work rigid-joined frames are analyzed by neglecting the axial Deformations. This analysis is done by using Modified Mixed formulation, Modified Stiffness Formulations and Modified Flexibility Formulation.

Esmaeili R. has developed a computer program for indeterminate beam in Reference [15]. This work analyzed indeterminate beam problems with four different methods. These methods are included as Displacement Method, Integrated Force Method, LU Decomposition Method and Live Load Pattern Method. These methods can help users to understand better the theories and compared the methods with together.

Khosravi M. S. has developed a computer program for usage of equilibrium equations in truss analysis. This work has done by six deferent methods. These methods are sorted as three parts displacements method, classical force method and integrated force method. Each part has two approaches these approaches are:

Two approaches in Displacement Method are:

- Stiffness method
- Dual integrated force method

Two approaches in Classical Force Method are:

- Classical force method via QR decomposition
- Classical force method via LU decomposition

Two approaches in Integrated Force Method are:

- Integrated force method via null space.
- Integrated force method via singular value decomposition.

Chapter 3

PROBLEM STATEMENT AND SOLUTION APPROACH

3.1 Introduction

In this chapter the research problem and its characteristics are presented.

3.2 The Problem

This thesis intends to develop an algorithm that will enable the analysis of a planar rigid frame using the Integrated Force Method (IFM) and the Dual Integrated Force Method. The resulting computer code will yield,

- The independent member forces
- Member end forces
- Reactions at the supports
- Nodal displacements
- The Axial Force, Shear Force, and Bending Moment Diagrams

Programming will be done by using the computer algebra system Mathematica 7.

3.3 The Reason of Research

There are many computer programs about analysis of frame with stiffness method and classical force method but there are no computer programs about analysis of frame with integrated force method. On the other hand, most of these programs only show the results of frame analysis like: nodal displacements, member end forces, support reactions and axial force, shear force, bending moment diagrams and neglected to show all procedure of methods.

3.4 The Desired Characteristics of the Programs for IFM and IFMD

3.4.1 The Desired Characteristics of the Programs for IFM

In the survey of the state of the art, it is found that in the existing documents by Patnaik [2, 3, 4 and 9] the compatibility conditions are obtained by the following procedure:

- Generating the deformation displacement relations.
- and then eliminating the displacements.
- obtain the compatibility conditions.

In this thesis an alternative algebraic approach will be followed. After generating the equilibrium equations the following two algebraic techniques will be used.

- Nullspace of the equilibrium matrix combined with the unconnected flexibility matrix.
- Singular Value Decomposition of the equilibrium matrix combined with the unconnected flexibility matrix.

3.4.2 The Desired Characteristics of the Programs for IFMD

In the existing documents and computer software the main emphasis is to generate the global stiffness matrix, however in the Dual Integrated Force Method the global stiffness matrix is generated by using the easily generated equilibrium matrix and the matrix manipulation capabilities of the computer algebra system Mathematica 7. Using these programming capabilities the global stiffness matrix is generated in only one programming line. Therefore the usage of Mathematica 7 gives us programming advantage and computer usage time.

3.4.3 Other Attributes of the Proposed Analysis Packages for IFM and IFMD

a) Easy to Use:

There is no need to read any manual or documentation for first time users. The programs are easy to operate and learn.

b) Simple:

Compared to existing commercial structural analysis packages, the programs developed have few option and parameters which makes running of programs easy.

c) Transparent Theory:

The theory in the methods is displayed within the programs therefore it is easy to understand them and easy follow their procedures.

d) Chasing Variables:

If the user is suspicious of any results, he or she can pursue the value of any variable during the calculation procedure to find the source of probable mistake.

e) Flexible:

At each step of calculations the program code, which is doing process, is shown. For beginners this helps to learn more about programming techniques. Advanced users can change, or add parts to program code to change its utility.

f) Educational:

The programs are like tutorials. The theories, Integrated Force Method via Null Space, Integrated Force Method via Singular Value Decomposition and Dual Integrated Force Method, are introduced to the user working with the packages. At each step sufficient are presented.

g) Accessible:

There is no need to spend long hours searching the internet to find packages that have these desired characteristics. The packages are unrestricted and available for students, instructors and engineers.

3.5 An Overview of Solution Approach for IFM and SVD

In this section the solution approach for IFM is outlined in Figure 4:

- Generate the equilibrium equations [B].
- Assemble the unconnected flexibility matrix [G].
- Find compatibility conditions, [CC], by using the algebraic techniques Nullspace and Singular Value Decomposition.
- Solve for independent member forces.
- Find nodal displacements.
- Find the member end forces.
- Find support reaction.
- Plot Axial Force, Shear Force and Bending Moment Diagrams.

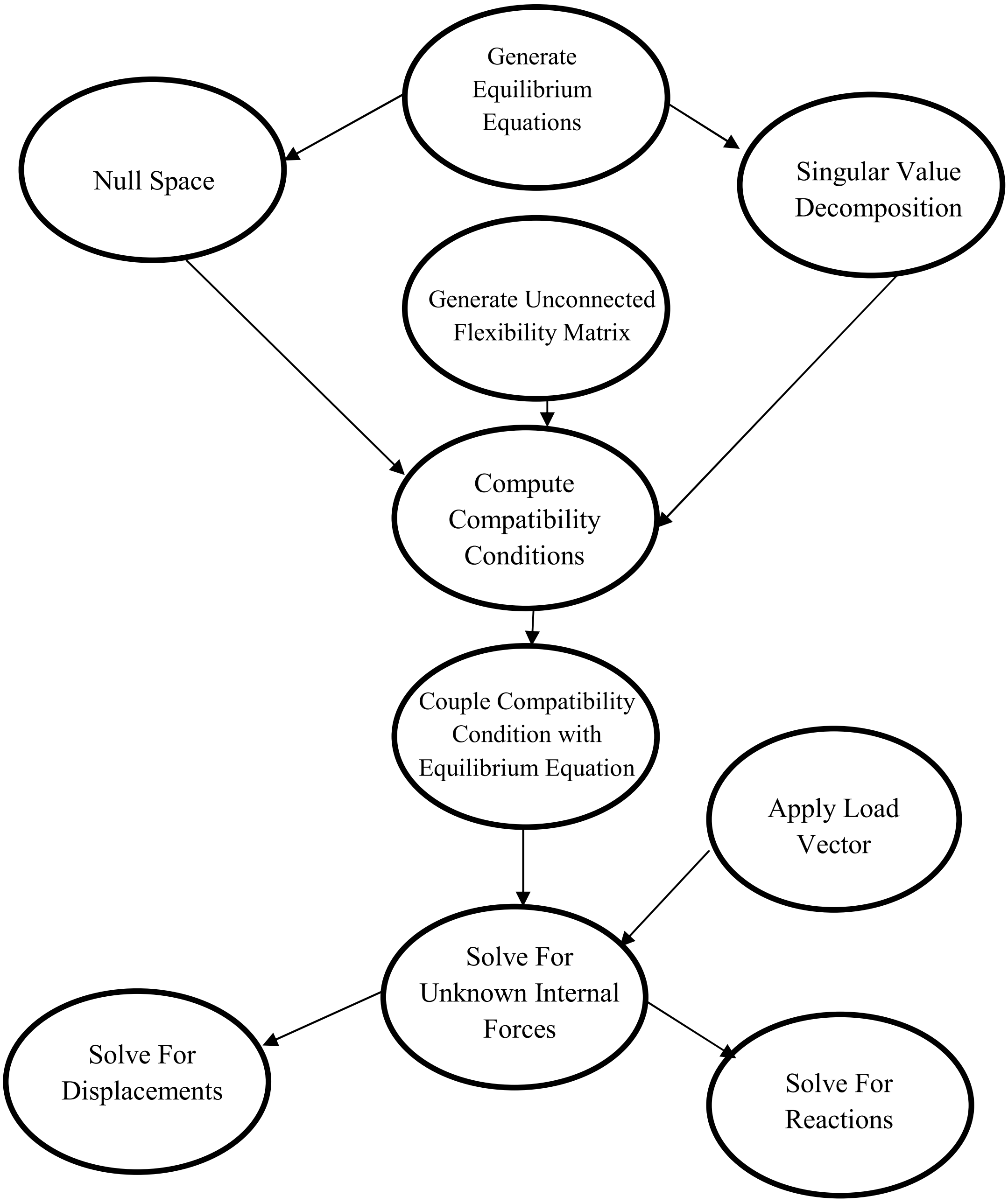


Figure 4: Overview of Integrated Force Method

3.6 An Overview of Solution Approach for IFMD

In this section the solution approach for IFM is outlined in Figure 5:

- Generate the equilibrium equations [B].
- Assemble the unconnected flexibility matrix [G].
- Find the inverse of unconnected flexibility matrix [G] to obtain the global stiffness matrix.
- Generate the Global Stiffness Matrix [K].
- Solve for nodal displacements [X].
- Find independent member forces.
- Find the member end forces.
- Find support reaction.
- Plot Axial Force, Shear Force and Bending Moment Diagrams.

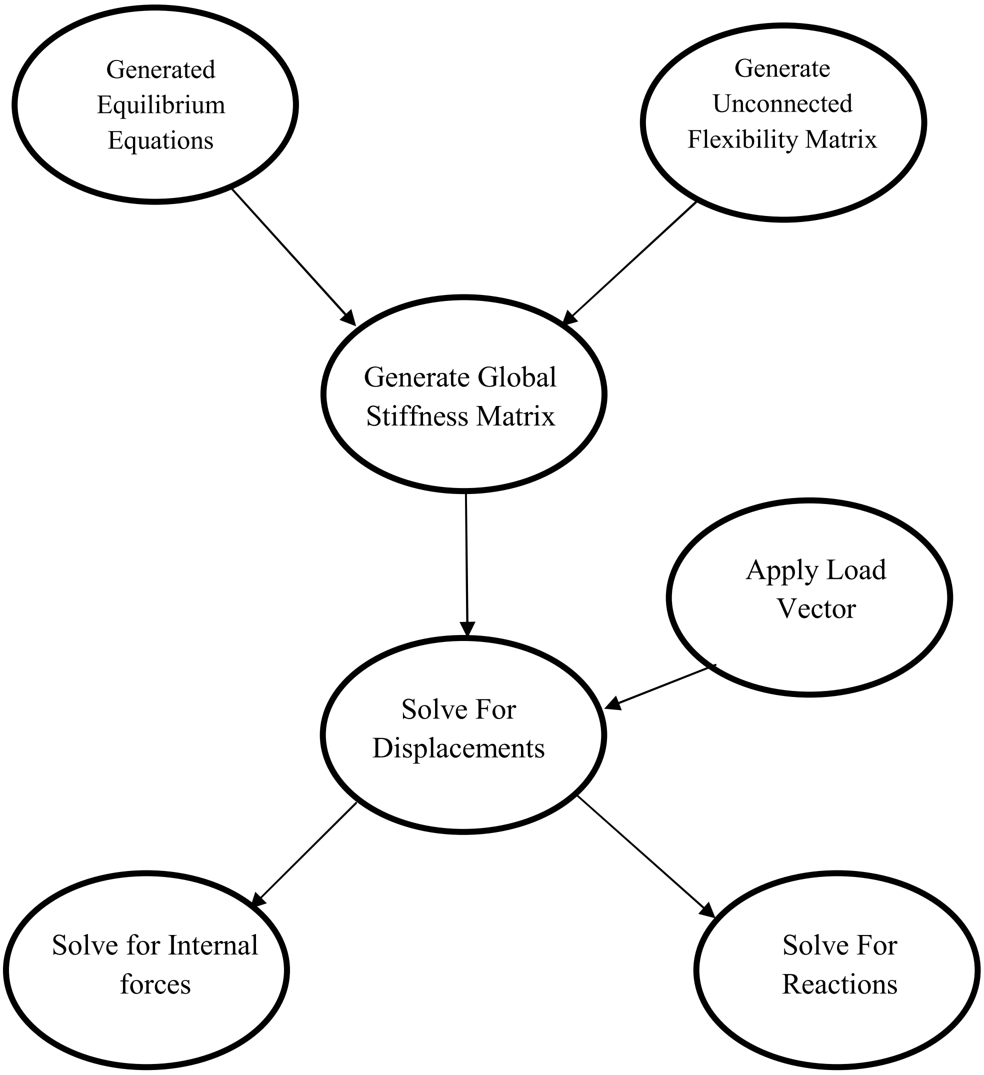


Figure 5: Overview of Dual Integrated Force Method

3.7 Mathematica Software as a Tool

To achieve the desired characteristics of the structural analysis package, the computer algebra system, Mathematica [16], is used.

The main reason for selecting Mathematica software is the following properties of it:

- User can do interactive calculation using notebooks.
- User can get started just like using calculator.
- User can chose from over a thousand built-in functions.
- User can do numerical calculation to any precision
- User can do symbolic calculation to get formulas.
- User can lists to present collections of things.
- User can create 2D and 3D graphics.
- User can solve equation symbolically or numerically.
- User can do integrals and derivatives.
- User can manipulate vectors and matrices.
- User can define his or her own functions.

Chapter 4

AUTOMATICALL ASSEMBELY OF EQUILIBRIUM EQUATIONS

4.1 Formulation of Equilibrium Equations

This section discusses a systematic way to setting up the equilibrium equations at the free degrees of freedom of the frame model. This way will help to write computer code to assemble the equilibrium equations automatically. To illustrate the technique, consider inclined plane frame with two elements and three nodes as shown in Figure 6. This example is from chapter two of (CE 220/ Filip C. Filippou), Reference [1].

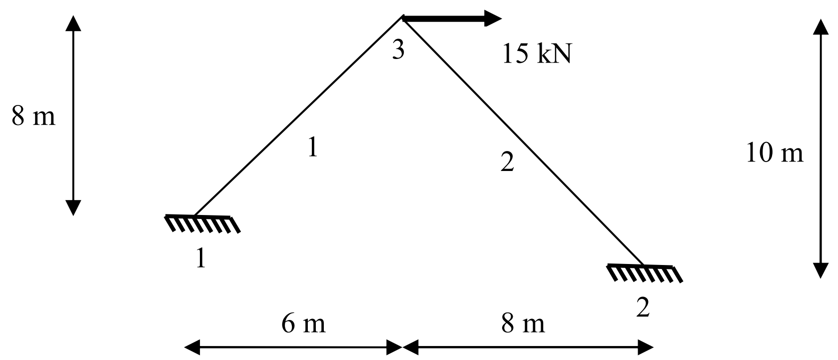


Figure 6: frame example with inclined members

According to section 2.5 the numbering of global degrees of freedom (dof) is done. Only free degrees of freedom (dof) exist at node number 3.

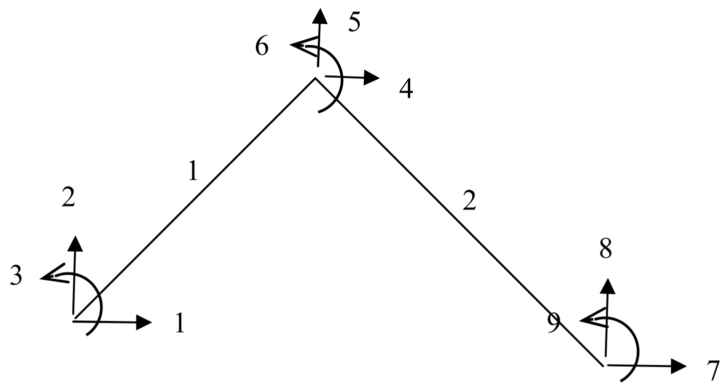


Figure 7: Numbering of global dofs

In order to write the equilibrium equations (B), free body diagram of node 3 is separated, as depicted in Figure 8.

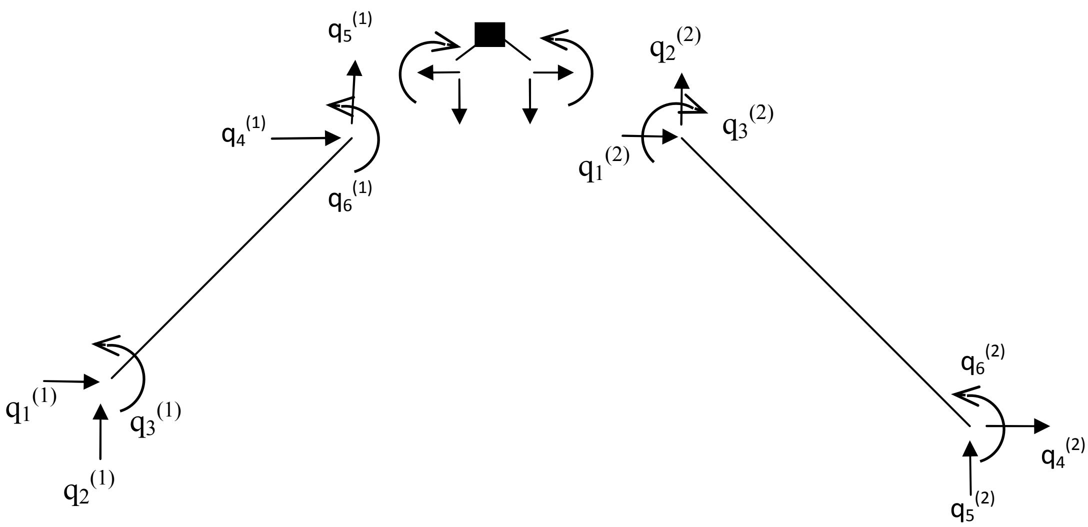


Figure 8: Node and element free bodies

Where superscript shows the number of element and subscripts shows the number of forces for each element. In Figure 8 the element end forces are oriented in the global reference system and denoted by q^1. The number of element forces corresponds to the global degrees of freedom numbering.

Therefore the length and direction cosines of elements are calculated with equations 4.1.

$$L = \sqrt{(xj - xi)^2 + (yj - yi)^2} \qquad 4.1\ (a)$$

$$cos\alpha = \frac{xj - xi}{L} \qquad 4.1\ (b)$$

$$sin\alpha = \frac{yj - yi}{L} \qquad 4.1\ (c)$$

According to section two, member equilibrium matrix calculated for each element. This matrix transformed the basic frame force to element equilibrium equations in local coordinate system.

$$b = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -L & -1 \\ 1 & 0 & 0 \\ 0 & 1. & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad 4.2$$

Where, L is length of member.

According to reference [1], to convert internal forces from local coordinate system to global coordinate system in each element, a transforming matrix t, is used:

$$t = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 & 0 & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & 0 & 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad 4.3$$

In additions, to form element equilibrium equations in global system, b_g are required which:

$$bg = b.t \tag{4.4}$$

$$\mathrm{bg} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 & 0 & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & 0 & 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -L & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -\cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & -\cos\alpha & 0 \\ 0 & -L & -1 \\ \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Note that the global numbering of degrees of freedom which is depicted in figure 6 is used in the row numbers of bg .therefore equilibrium equations can be assembled directly by transferring each entry form bg to overall equilibrium equations. This is carried out according to global degree of freedom as.

According to section 2.5.1, equilibrium equations only written for free degrees of freedom, it means the equilibrium matrix is not square and reaction forces are omitted. Therefore, two elements and 3 free degrees of freedom, the above system of equations is:

$$\begin{bmatrix} \sin\alpha & -\cos\alpha & 0 & -\cos\beta & -\sin\beta & 0 \\ \cos\alpha & \sin\alpha & 0 & \sin\beta & -\cos\beta & 0 \\ 0 & 0 & 1 & 0 & -L2 & -1 \end{bmatrix} \begin{bmatrix} f1 \\ f2 \\ f3 \\ f4 \\ f5 \\ f6 \\ f7 \\ f8 \\ f9 \end{bmatrix} = \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix}$$

Now the equilibrium equations are generated.

4.2 Algorithm for Automatic Assembly of Equilibrium Equations

This section explains how to computer codes to generated equilibrium equations automatically.

In order to generate equilibrium equations following procedure is used:

Step1: Get x and y coordinate to start node and end node according to member incidence.

Step2: Use equation 4.1 to find the length and cosine direction of members.

Step3: Use equations 4.2 to find member equilibrium matrix **(b)** for each element.

Step4: Use equation 4.3 to convert internal member end forces from local coordinate to global coordinate.

Step5: use equation 4.4 to create member equilibrium matrix in global coordinate system (**bg**) for each element.

Step6: Find the number of free degrees of freedom.

Step7: Establish (3m) × (3j-restrained degrees of freedom) zero matrix. (**m:** number of elements, **j:** number of nodes).

$$\begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}_{(3m) \times (3j\text{-restrained degrees of freedom})}$$

Step8: Place each member equilibrium matrix in global coordinate (**bg**) in to the columns of the zero matrix.

<u>Example:</u>

The following steps are done for the frame which is shown in Figure 4.

Step1:

x and y coordinate for element **1** are:

$$cos\alpha = \frac{x3 - x1}{L1} = \frac{8 - 0}{10} = 0.6$$

$$sin\alpha = \frac{y3 - y1}{L1} = \frac{6 - 0}{10} = 0.8$$

x and y coordinate for element **2** are:

$$cos\beta = \frac{x2 - x3}{L2} = \frac{14 - 6}{12.8} = 0.62$$

$$sin\beta = \frac{y2 - y3}{L1} = \frac{-2 - 8}{12.8} = -0.78$$

Step2:

Length of member **1** is:

$$\mathrm{L1} = \sqrt{(\mathrm{x3} - \mathrm{x1})^2 + (\mathrm{y3} - \mathrm{y1})^2} = \sqrt{(6 - 0)^2 + (8 - 0)^2} = 10$$

Length of member **2** is:

$$\mathrm{L1} = \sqrt{(\mathrm{x2} - \mathrm{x3})^2 + (\mathrm{y2} - \mathrm{y3})^2} = \sqrt{(14 - 6)^2 + (-2 - 8)^2} = 12.8$$

Step3:

Member equilibrium matrix for element **1** is:

$$b1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -10 & -1 \\ 1 & 0 & 0 \\ 0 & 1. & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Member equilibrium matrix for element **2** is:

$$b2 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & -12.8 & -1 \\ 1 & 0 & 0 \\ 0 & 1. & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step4:

Transformation matrix for element **1** is:

$$t1 = \begin{bmatrix} 0.6 & -0.8 & 0 & 0 & 0 & 0 \\ 0.8 & 0.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.6 & -0.8 & 0 \\ 0 & 0 & 0 & 0.8 & 0.6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Transformation matrix for element **2** is:

$$t2 = \begin{bmatrix} 0.62 & 0.78 & 0 & 0 & 0 & 0 \\ -0.78 & 0.62 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.62 & 0.78 & 0 \\ 0 & 0 & 0 & -0.78 & 0.62 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Step5:

Member equilibrium matrix in global coordinate for element **1** is:

$$bg1 = \begin{bmatrix} -0.6 & 0.8 & 0 \\ -0.8 & -0.6 & 0 \\ 0 & -10 & -1 \\ 0.6 & -0.8 & 0 \\ 0.8 & 0.6 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Member equilibrium matrix in global coordinate for element **2** is:

$$bg2 = \begin{bmatrix} -0.62 & -0.78 & 0 \\ 0.78 & -0.62 & 0 \\ 0 & -12.8 & -1 \\ 0.62 & 0.78 & 0 \\ -0.78 & 0.62 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Step6:

This example has totally nine degrees of freedom, six of them are restraint and of three of them are free.

Number of free degrees of freedom = {0, 0, 0, 1, 2, 3, 0, 0, 0}

Step7:

Zero matrix for this structure

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Step8:

in final step equilibrium equations matrix is ready

$$\begin{bmatrix} 0.8 & -0.6 & 0 & -0.62 & -0.78 & 0 \\ 0.6 & 0.8 & 0 & 0.78 & -0.62 & 0 \\ 0 & 0 & 1 & 0 & -12.8 & -1 \end{bmatrix}$$

Chapter 5

RIGID FRAME ANALYSIS PACKAGES

5.1 Introduction

Three packages are developed for the analysis of coplanar indeterminate rigid frames. Each package uses different theory which has been introduced using chapter2.

These frame analysis packages are:

Package1: Integrated Force Method via Null space.

Package2: Integrated Force Method via singular value decomposition.

Package3: Dual integrated force method.

Solving the same problem with *Packages 1* and *2* will present different approaches for finding the compatibility conditions, and using *Package 3* will present students and practicing engineers an efficient analysis for rigid frames. In academic environment students may learn more about advantages or disadvantages of the force and displacement methods.

For each method the frame of Figure 9 will be used as illustrative example. The frame has 28 elements and 20 nodes. The moment of inertia for each element is 0.0005 m^4,the area for each element is 0.002 m^2 and the modulus of elasticity is 2×10^8 kN/m^2. This frame has four horizontal applied joint loads at node 5,9,13 and

17. The value of these applied joint loads are same and they are equal to 150 kN. The members 17, 18,19,20,21,22,23,24,25,26,27,28 under the action of uniform distributed loads. Nine of these members (17, 18, 19, 20, 21, 22, 23, 24, 25) have 20 kN/m load and three of them (26, 27, 28) have 15kN/m load. This example will be solved with *Packages1, 2* and *3*. Researchers can learn more about advantages and disadvantages of these three methods.

The structures of the matrices generated for this frame will be demonstrated by using their matrix plot. Each matrix plot displays the nonzero entries of matrices in color, according to the values of the nonzero entries the matrix plot colors change.

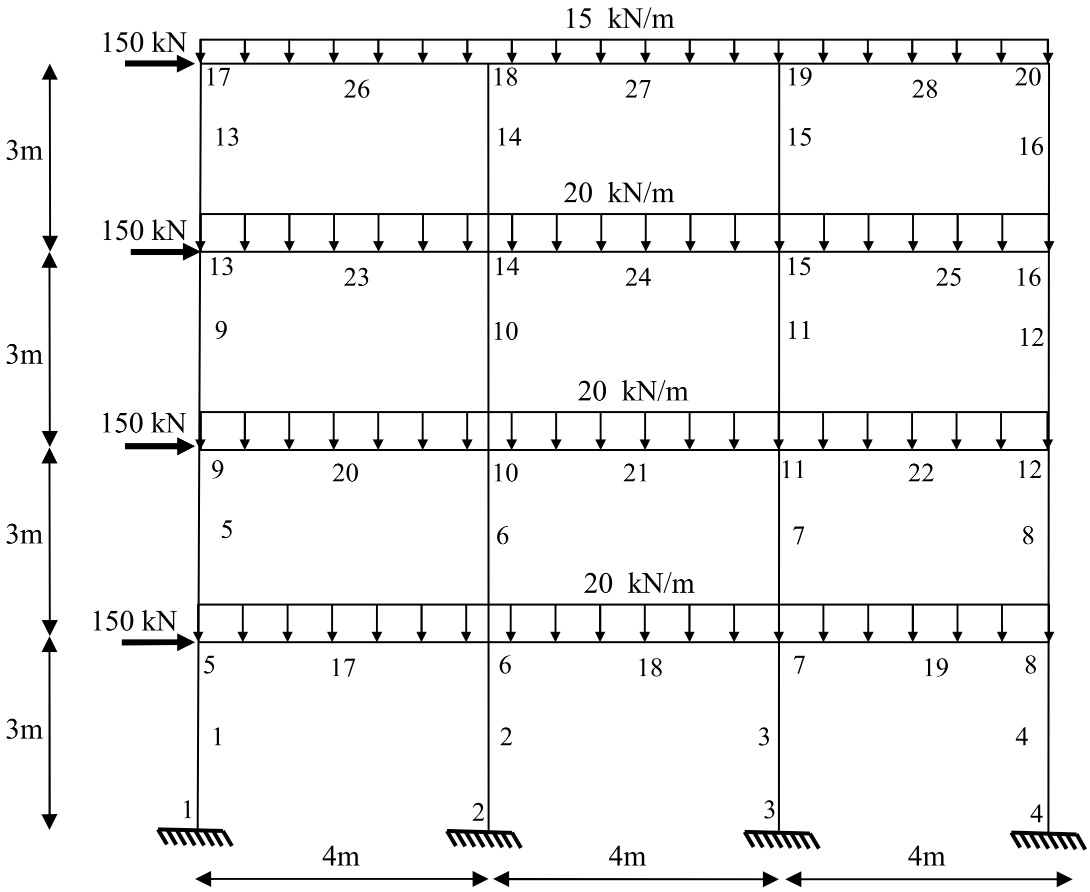


Figure 9: 28 Elements Frame

The packages consist of three main common phases:

- Data Input Phase
- Calculation Phase
- Reporting Phase

5.2 Data Input Phase

The phase defines the problem. Data input phase is designed to assist in the illustration of general input of frame (number of node and elements), geometry of frame (member incidence, coordinate of joints and freedom of joints), properties of elements and material (moment of inertia, area and modulus of elasticity) and loading case (joint load and fixed end load). A diagram of input phase skeleton is shown in Figure 10.

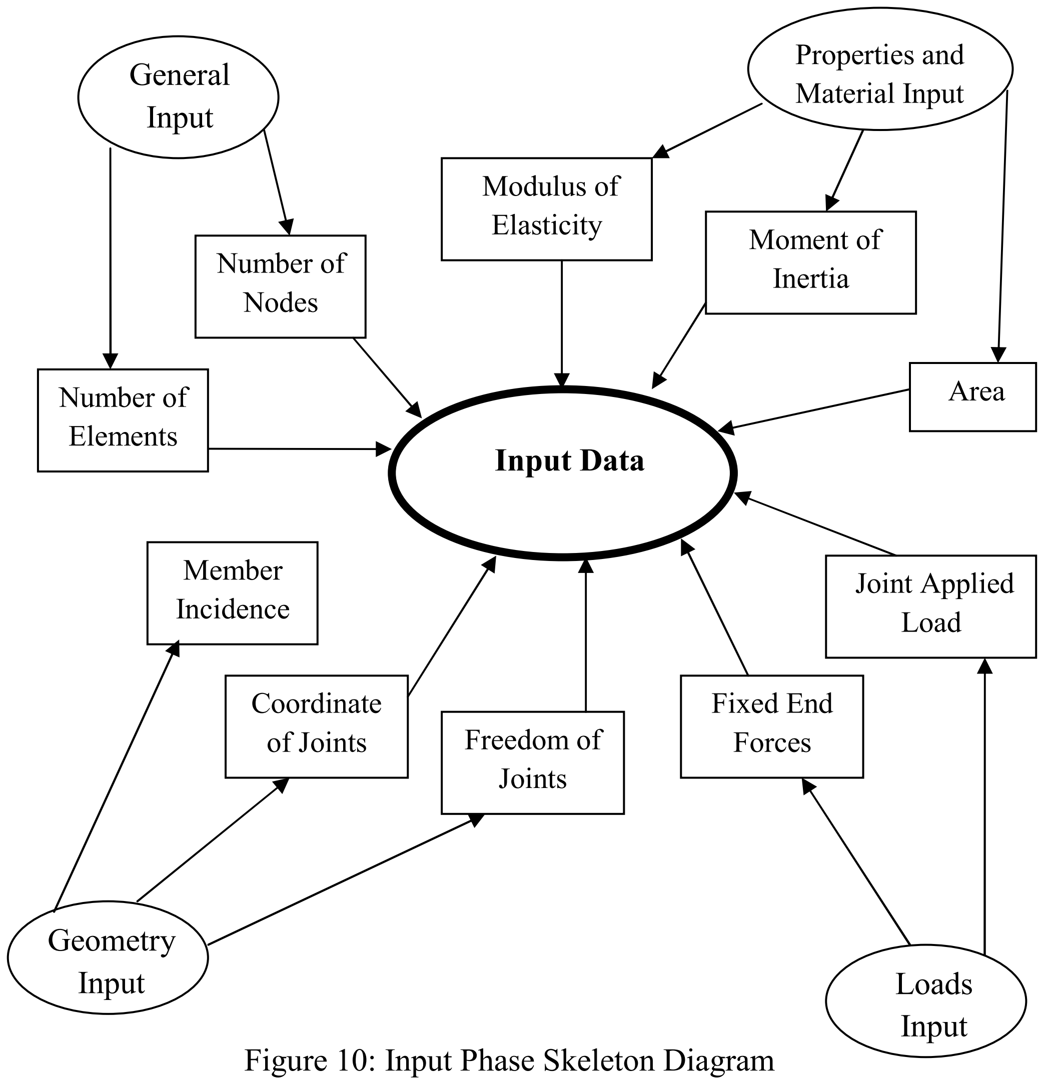


Figure 10: Input Phase Skeleton Diagram

5.2.1 User Interface of the Data Input Phase

The user interface of data input phase consists of four sections:

- General Data Input
- Geometry Data Input
- Properties and Material Data Input
- Loads Data Input

Once the user complete these four sections, data input phase is completed and the program can be run and user may go directly to last phase to see the results immediately.

5.2.1.1 General Data Input Phase

In this step of procedure the user can give the number of elements and number of nodes as shown in Figure 11. Different colors are used to assist in finding these variables. The user can change white color text.

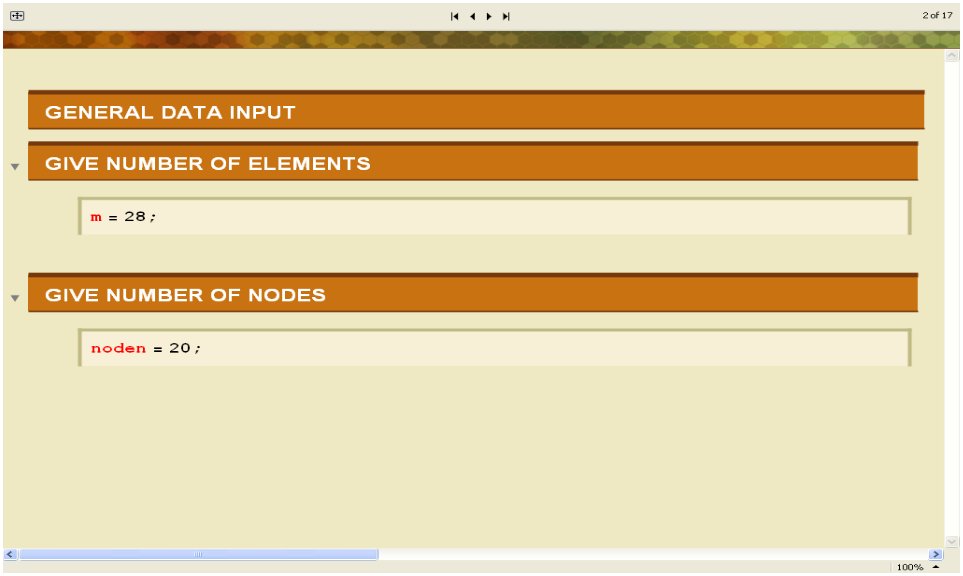


Figure 11: General Input Phase

5.2.1.2 Geometry Data Input Phase

In this step of procedure the user can give information to specify geometry of the frame and support conditions. This step has three parts (member incidence, coordinate of joints and freedom of joints). Part one is member incidence, this part includes m×3 matrix (m is equal to number of members). This matrix has three columns, first column shows the number of each member which are starting from 1 up to number of members. Second and third columns show the number of starting node and end node for each element respectively (Figure 12).

Figure 12: Geometry Input Phase (member incidence)

Part 2 is coordinate of joints, which includes n×3 matrix. The first column of this matrix shows the number of each join. Second and third columns show the coordinate of each joint according to x and y direction respectively (Figure 13).

Figure 13: Geometry Input Phase (coordinate of joints)

Part 3 is freedom of joints, which includes n×4 matrix. First column of this matrix shows the number of joints, second, third and fourth columns show the condition of supports and degrees of freedom to be free or restrained in each node (Figure 14). According to section two each node of frame has three degrees of freedom. For recognizing the degree of at the supports user can use either 1 or 0 numbers to demonstrate (0 for free and 1 is for restrained).

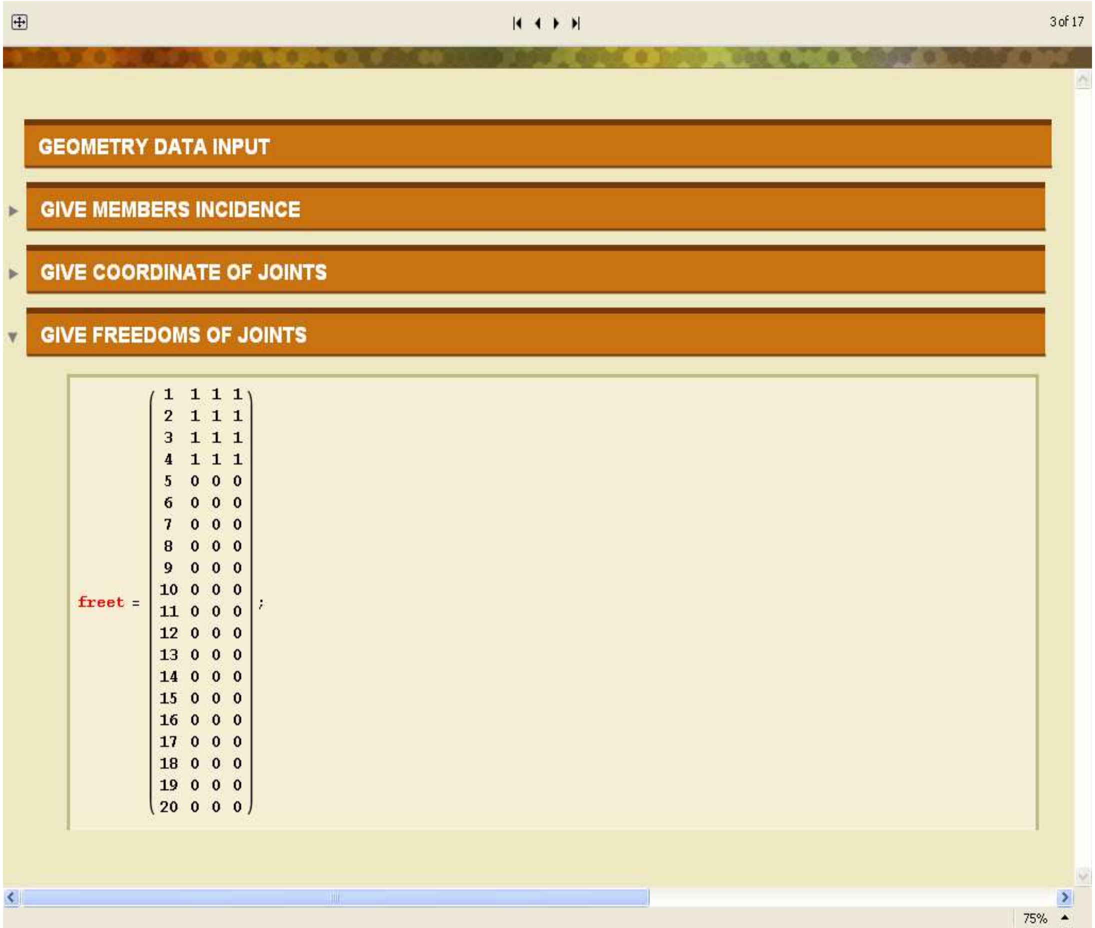


Figure 14: Geometry Input Phase (freedom of joints)

5.2.1.3 Properties Data Input Phase

In this step user can give necessary properties and material to program. These properties are included as: Moment of Inertia and Area. User should give these two properties for each element. Material is included as: Modulus of Elasticity. Figure 15 shows the properties and materials which are necessary for frame analysis.

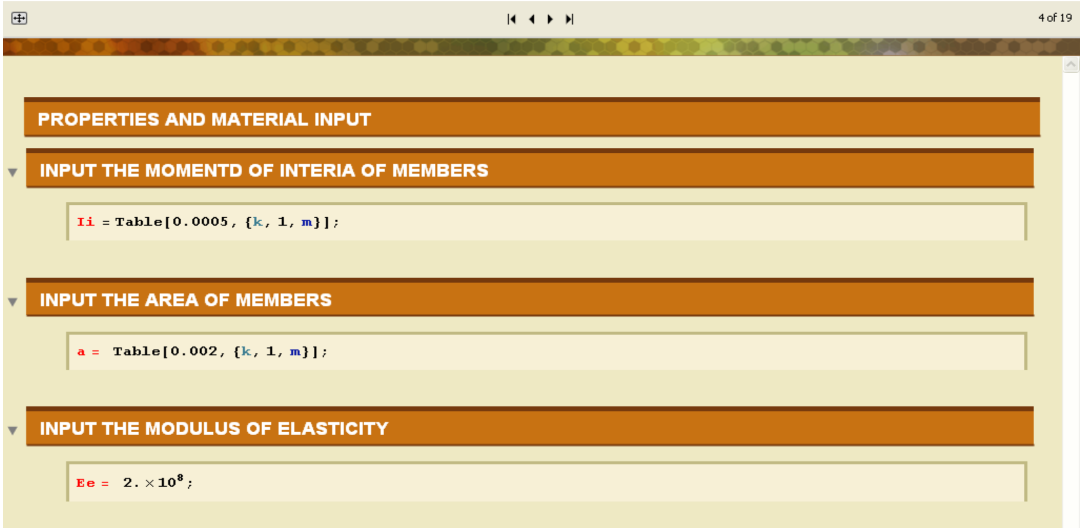


Figure 15: shows all necessary properties input.

5.2.1.4 Loading Data Input Phase

In this step user can define fixed end force, point horizontal load, point vertical load and external moments applied to the frame, by changing the value of four loading variables. Point horizontal load, point vertical load and external moment are applied at joint and fixed end force applied at member.

This step has two parts. Part one includes n×4 matrix (n is number of nodes) with four columns. Column one shows the number of joints, column two shows applied point horizontal load at each node, column three shows applied point vertical load at each joint and column four shows applied external moment at each node (Figure 16).

Part two is a vector applied fixed end force for each element (Figure 17).

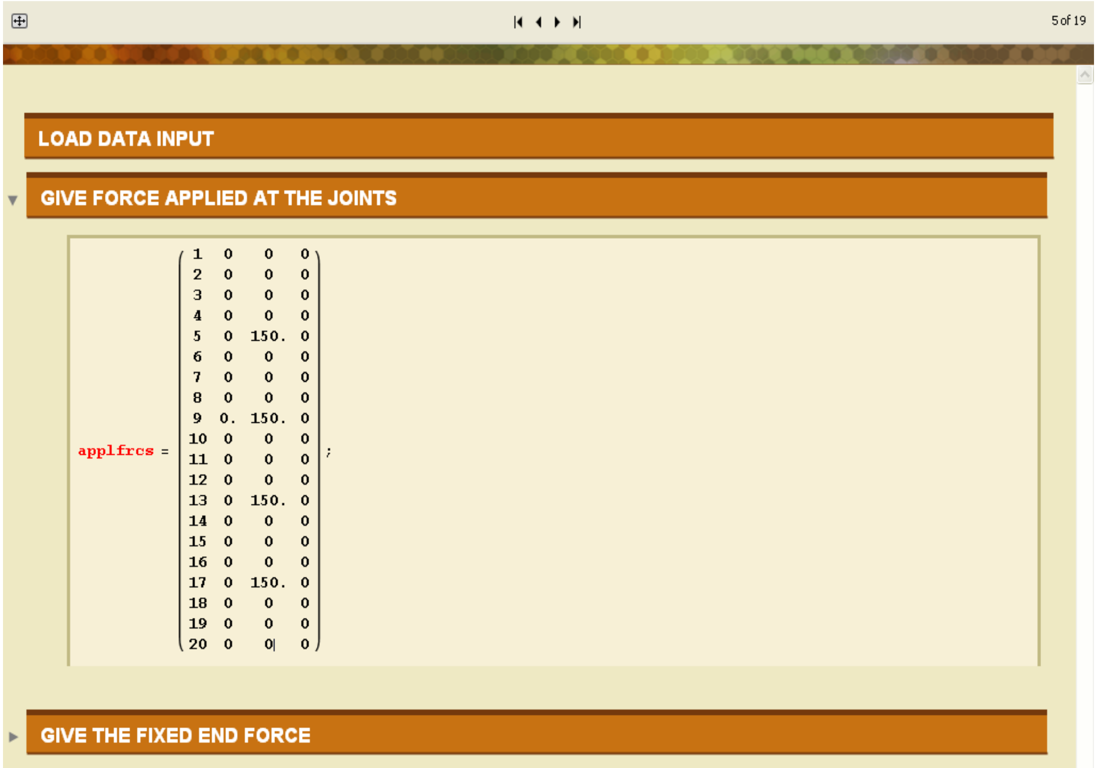


Figure 16: Loading Input Phase (Joint Load)

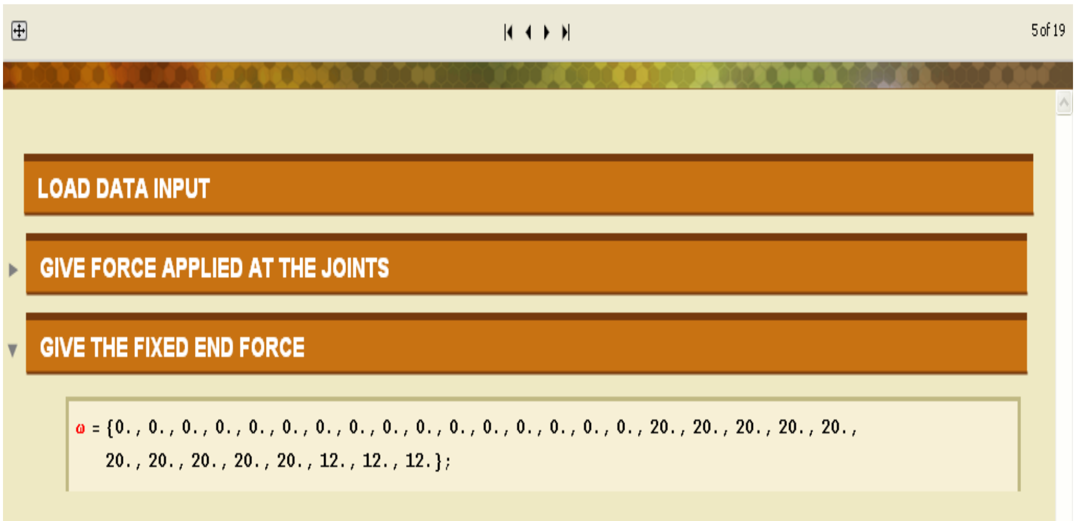


Figure 17: Loading Input Phase (Fixed End Force)

5.2.1.5 Subroutine For shape of Frame

The subroutines for graphical shape of frame are included as:

Frame Figure: creates a graphic of the frame.

Fixed Support: creates a graphic of affixed support.

Simple Support: creates a graphic of simple support.

Roller Support: creates a graphic roller support.

Element Numbering: create the number of each element.

Node Numbering: create the number of each node.

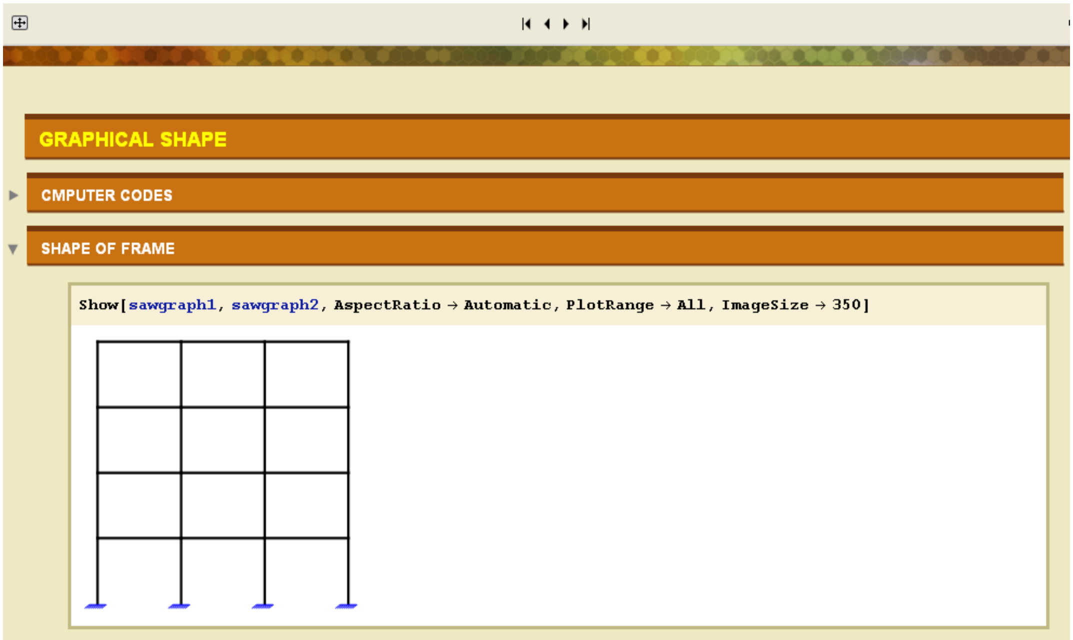


Figure 18: Shape of Frame

5.3 Calculation Phase

In this phase it was tried to make calculations similar to hand calculation procedures as far as possible. In this way, user can see each step of calculations like the way students do in exam papers, moreover, user can see its corresponding program code together with calculation phase consist of several steps depending on theory applied.

5.3.1 Integrated Force Method via Null Space

In integrated force method, according to Reference [3, 4] the null space of the equilibrium equations and the unconnected flexibility matrix is used to find the compatibility condition. Then the equilibrium equations are coupled with compatibility conditions to obtain a square matrix [4]. Finally the square matrix is used to solve independent forces.

Applying this method to the frame shown in Figure 9 and the procedure of integrated force method is demonstrated in Figure 19. In addition Matrix Plot of the Equilibrium Equations, Unconnected Flexibility Matrix, Compatibility Conditions, Coupled the Equilibrium Equations and Compatibility Conditions are shown in Figure 20.

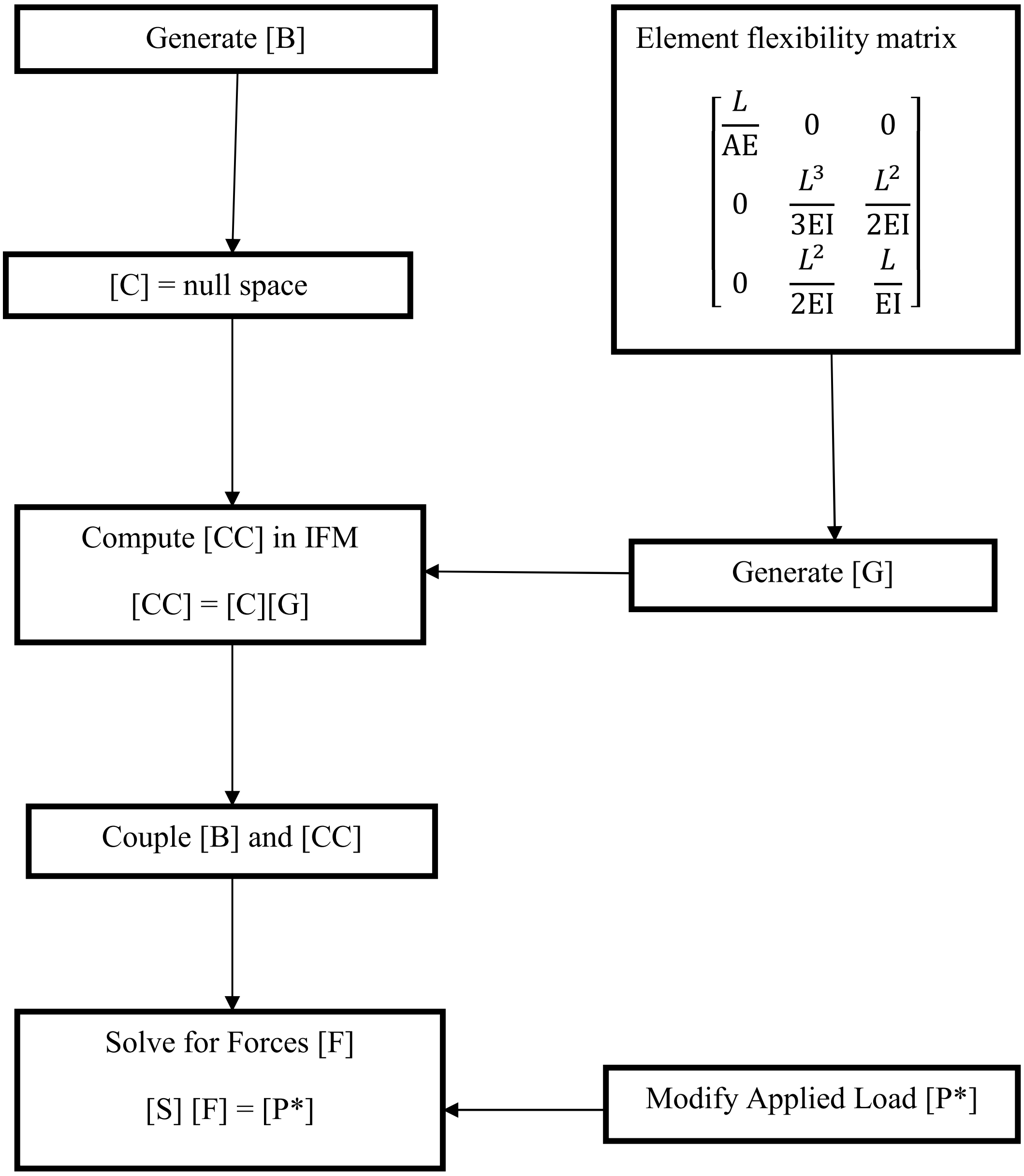
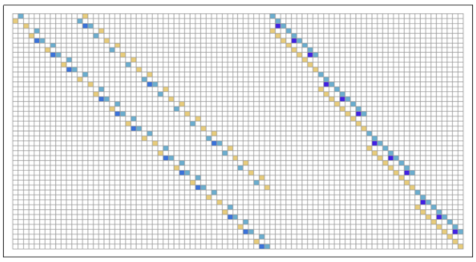
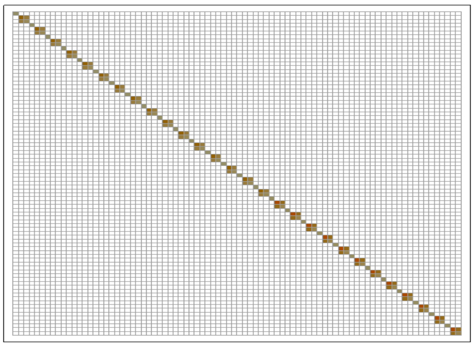


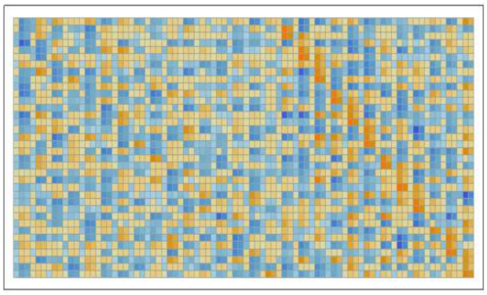
Figure 19: Algorithm of integrated force method via null space



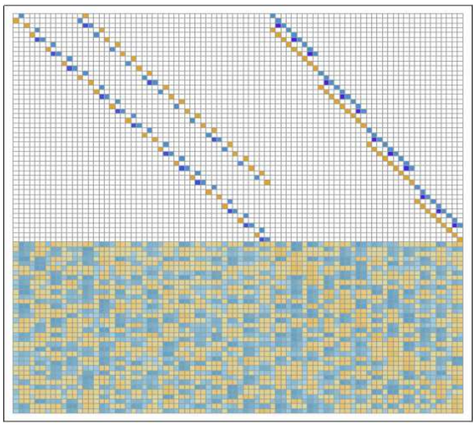
Equilibrium Equations [B] $_{84\times 48}$



Unconnected Flexibility Matrix [G] $_{84\times 84}$



Compatibility Conditions [CC] $_{84\times 36}$



Coupled Equilibrium Equations with

Compatibility Conditions matrix [S] $_{84\times 84}$

Figure 20: The matrix plot of the matrices generated with integrated force method

5.3.2 Integrated Force Method via Singular Value Decomposition

An alternative procedure for finding the compatibility conditions in the integrated force method is outlined in chapter two. After generating the equilibrium equations, Matrix $([B]^{T})^{pinv}$ is calculated by Equation 2.21. Then the matrix [M] is obtained by equation 2.20. Later, the singular value decomposition (SVD) of the matrix [M] carried out to obtain matrices $[M_u]$, $[M_v]$ and $[M_6]$. The compatibility conditions are obtained by using Equations 2.24 and 2.26. The procedure of Integrated Force Method via Singular Value Decomposition and applying this method to the frame are shown in Figures 21 and 9 respectively, The Matrix Plots of the Equilibrium Equations, Unconnected Flexibility Matrix are same as Figure 20.The [CC] matrix displayed in Figure 19 and 20 look almost similar but the numerical values of the compatibility conditions are different.

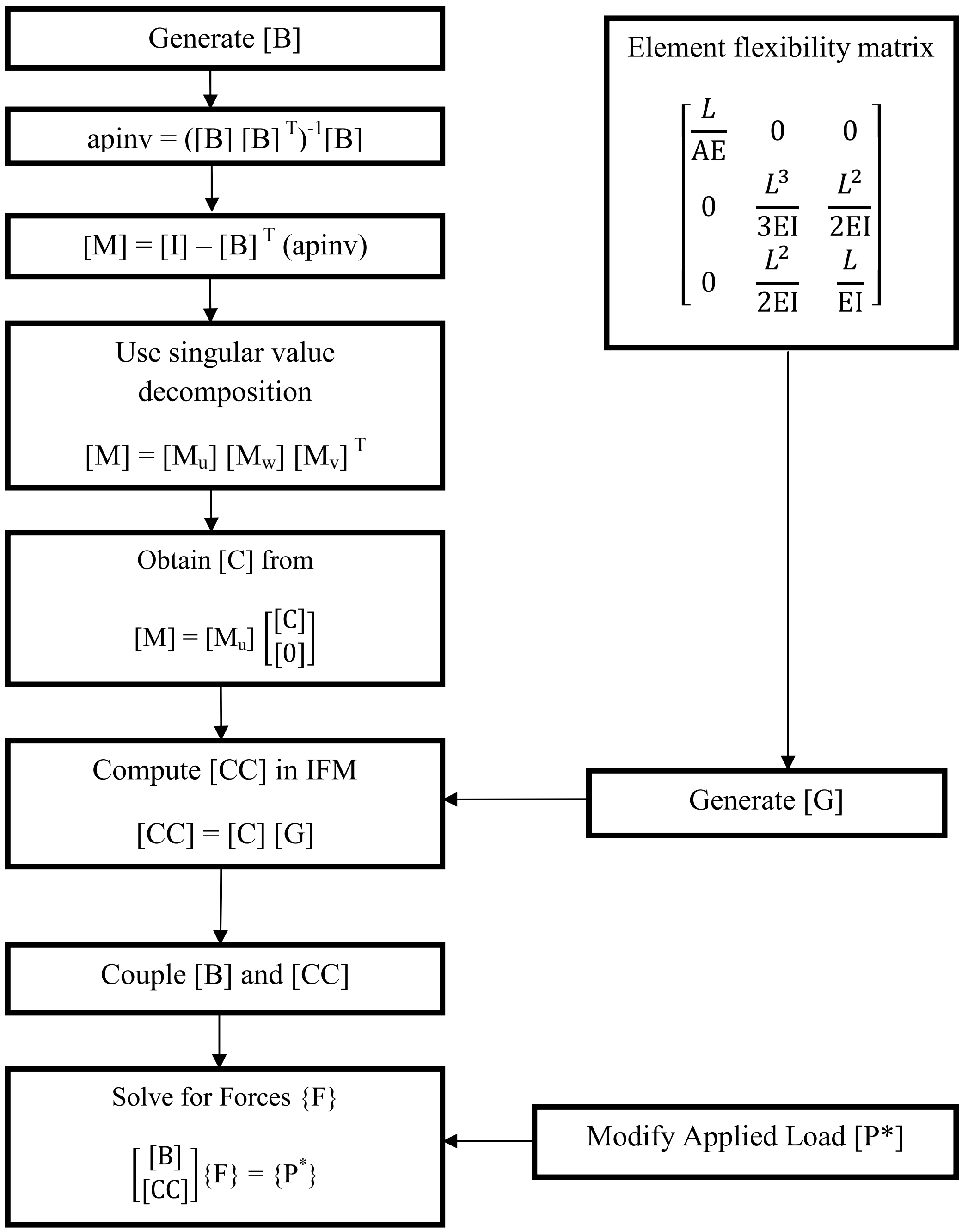


Figure 21: Algorithm of Integrated Force Method via Singular Value Decomposition

5.3.3 Dual Integrated Force Method

The procedure and equations of this method is shown in Figure 22. Equilibrium equations and unconnected flexibility matrix are generated.

According to Equation 2.15, the global stiffness matrix [K] is calculated. The inverse of the flexibility matrix is used to obtain the global stiffness matrix. In the next step the displacements vector is obtained by Equation 2.13.In the final step the independent forces are calculated from Equations 2.14.

In this method since the node and element numbering is the same, the Matrix Plots of the equilibrium equations, unconnected flexibility matrix, will remain unchanged. The Matrix Plot of the global stiffness matrix [K] is shown in Figure 23.

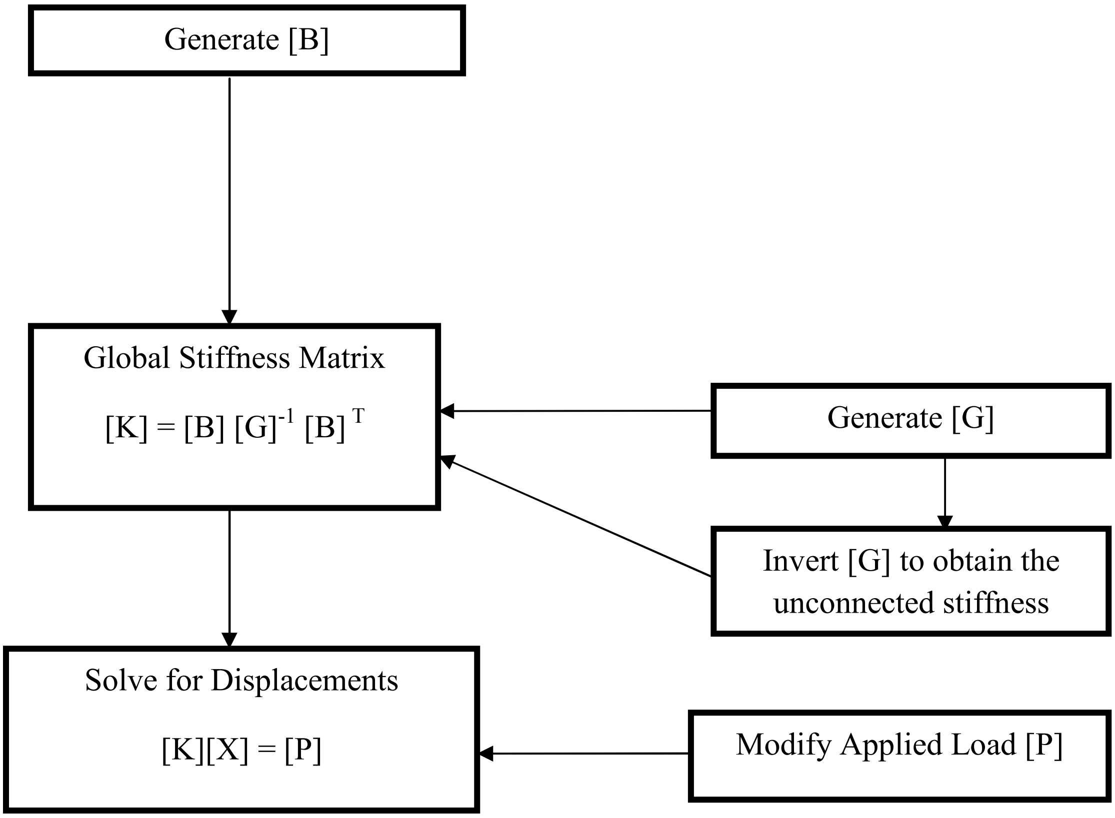


Figure 22: Algorithm of Dual Integrated Force Method

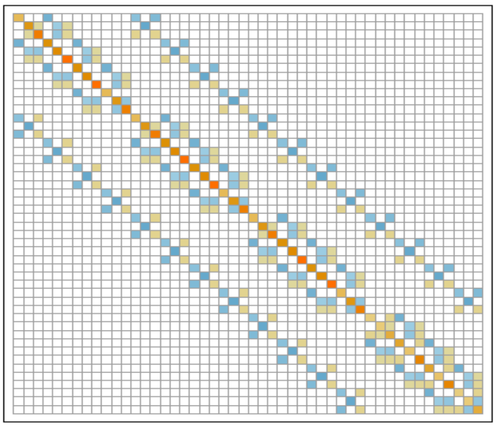


Figure 23: The matrix plot of the Global Stiffness Matrix $[K]_{48\times48}$, generated with Dual Integrated Force Method

5.4 Reporting Phase

In the reporting results phase the aim is to give the necessary information briefly and partially.

Reporting Phase Skeleton diagram shows in Figure 24.

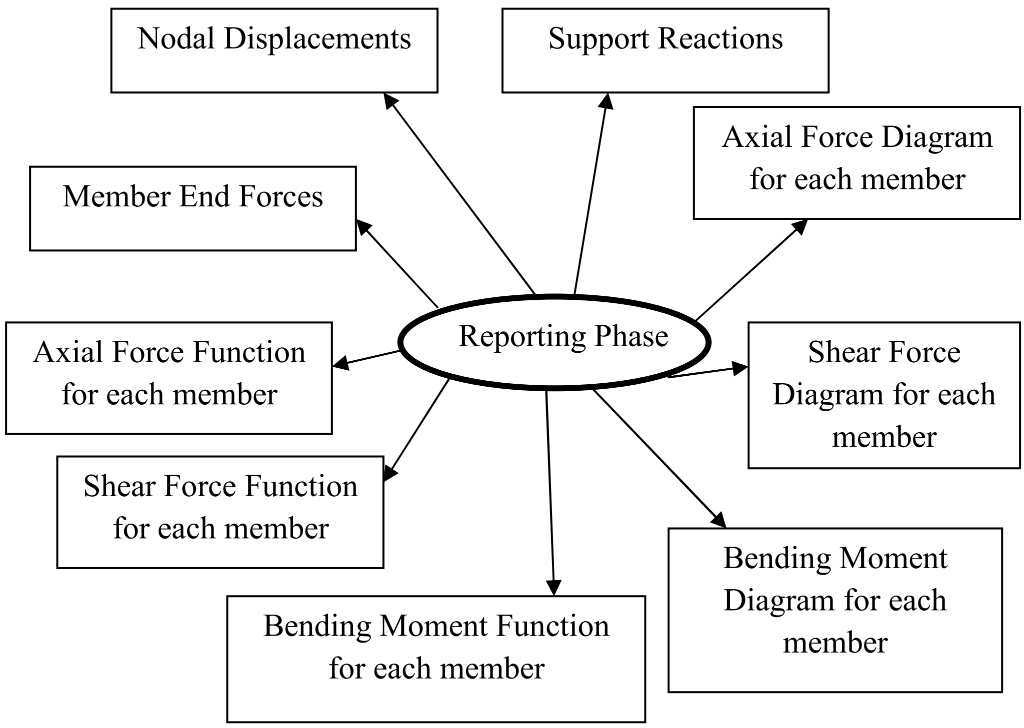


Figure 24: Reporting Results Phase, Skeleton Diagram

5.4.1 User Interface of the Reporting Input Phase.

The user interface of reporting phase consists of nine sections:

- Display the Nodal Displacements for each node.
- Display the Member End Forces for each element in local and global coordinate.
- Display Support Reactions results.
- Display Axial Force Function for each element.
- Display Shear Force Function for each element.
- Display Bending Moment Function for each element.
- Plot Axial Force Diagram for each element
- Plot Shear Force Diagram for each element
- Plot Bending Moment Diagram for each element

5.4.1.1 Display the Nodal Displacements for each node.

In this section, nodal displacements will be shown for each node. The three displacements for each node are reported as:

1) Displacement in the global X-direction.
2) Displacement in the global Y-direction.
3) Rotation in the global Z-direction.

The nodal displacements for the first six nodes of the example frame of Figure 5.1 are displayed in Figure 25.

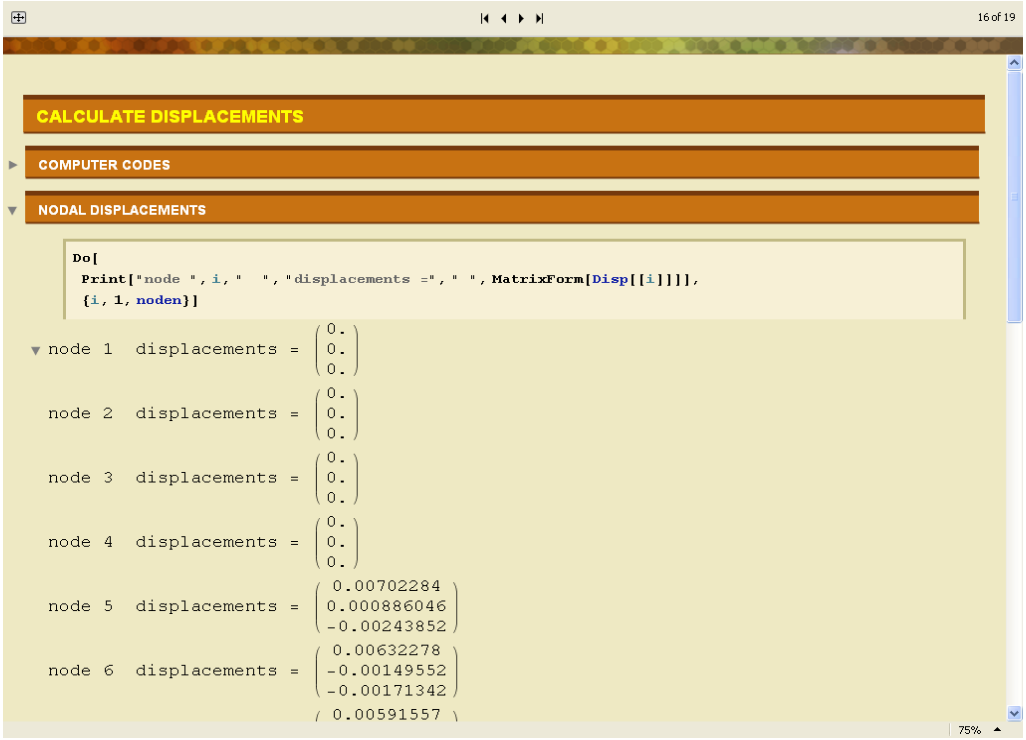


Figure 25: Reporting Results Phase, section 1, nodal displacements of node 1, 2, 3, 4, 5 and 6

5.4.1.2 Display the Member End Forces for each element in Local and Global coordinate.

In this section, Member End Forces for each element in local and global coordinate will be presented (Figure 26). The member end forces in local coordinate for the first three elements of the example frame of Figure 9 is displayed in Figure 26.

Figure 26: Reporting Results Phase, section 2, member end forces of member 1,2,3 in local coordinate

5.4.1.3 Display Support Reactions results.

In this section support reaction will be presented. The support reaction in global coordinate system for the supports of the example frame of Figure 9 is displayed in Figure 27.

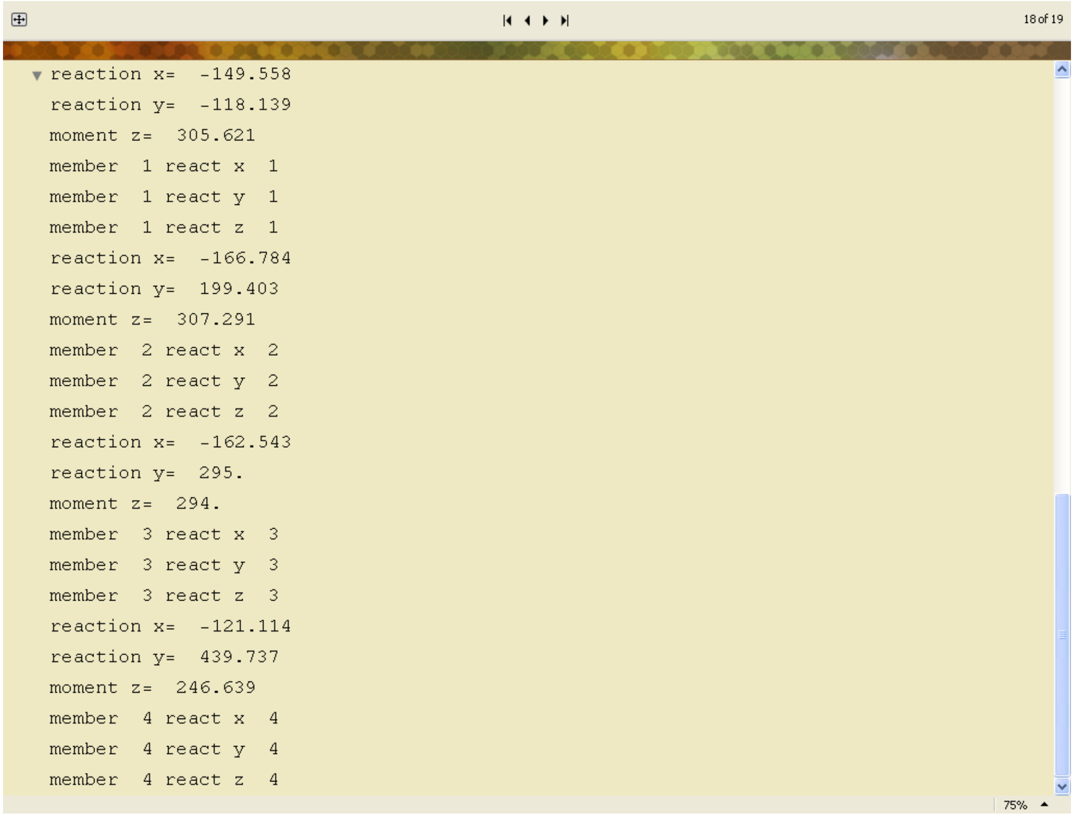


Figure 27: Reporting Results Phase, section 3, support reactions

5.4.1.4 Plot the Resulting Diagram for each element.

In this section the Axial Force, Shear Force and Bending Moment diagrams will be showed for users. Also the Axial Force, Shear Force and Bending Moment functions will be presented for user (Figure 28).

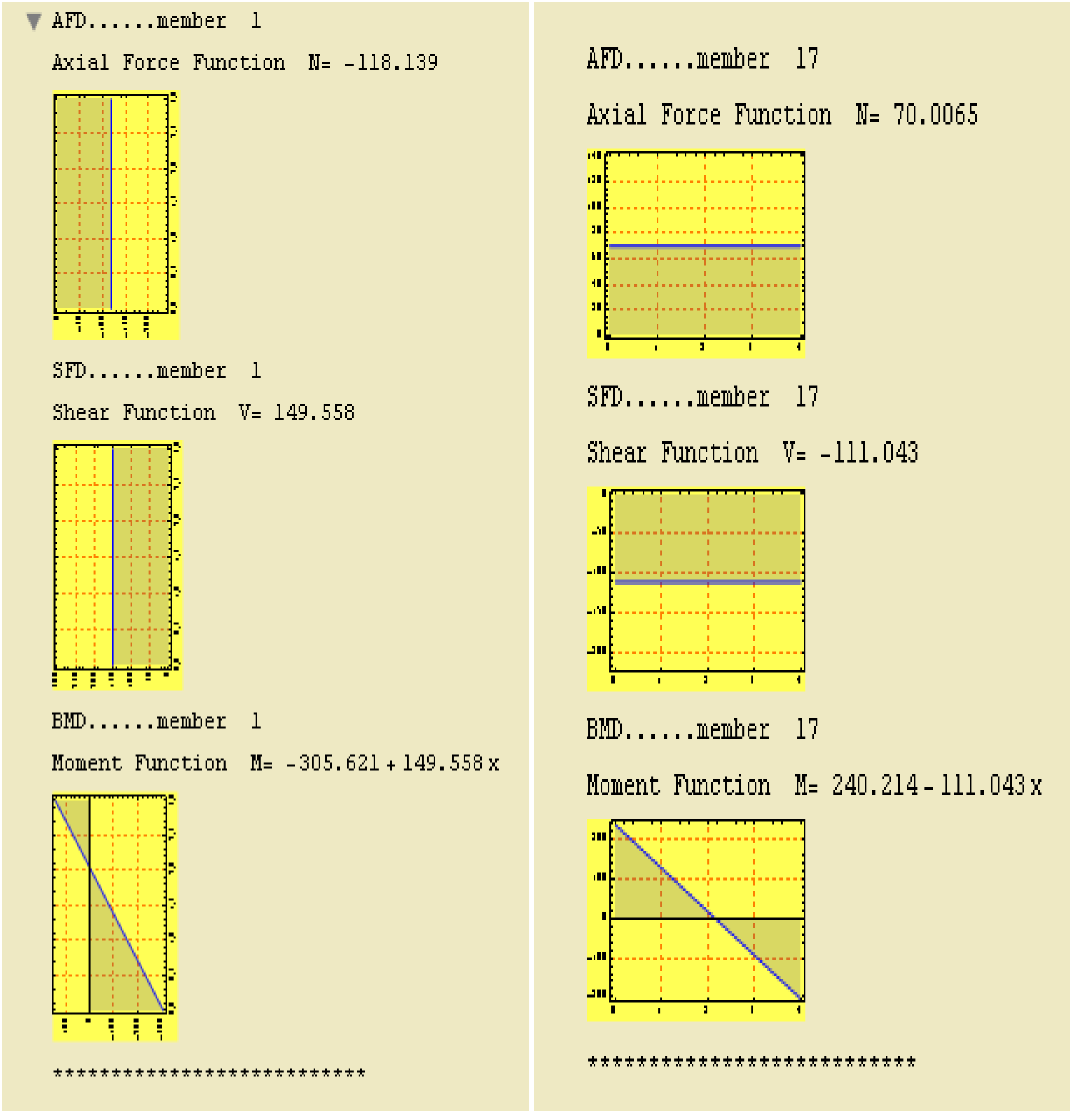


Figure 28: Reporting Results Phase, section 4, axial force, shear force and bending moment diagrams of member 1 and member 17 respectively.

5.5 Summary

In this chapter three analysis packages for indeterminate rigid frame has been introduced and in chapter 6 four illustrative examples are solved to help understanding more. The frame analysis packages were programmed by using Mathematica version 7.

The three packages and version 7 of Mathematica are provided on a CD which is placed at the back cover of this thesis.

Chapter 6

ILUSTRATIVE EXAMPLES

6.1 Introduction

In this chapter four examples are presented in order to illustrative the usage of analysis packages and the results are compared with Mastan2 V3.2.

6.2 Example for Integrated Force Method via Null Space (IFM)

Example 1: a frame is subjected for this example has 30 elements and 25 nodes. Nodes (6, 10, 13, 17, 20 and 24) are subjected to 100 kN shear joint load and nodes (8, 15, 22) are subjected to 20 kN/m moment. The example is analyzed by integrated force method via null space. The problem is solved for nodal displacements, member end forces in local coordinate, support reactions, the axial force diagram, shear force diagram and the bending moment diagram.

The moment of inertia and area of each member is:

$I = 5\times10^{-4}\ \ m^4$

$A = 2\times10^{-3}\ \ m^2$

The modulus of elasticity is:

$E = 2\times10^{8}\ \ kN/m^2$

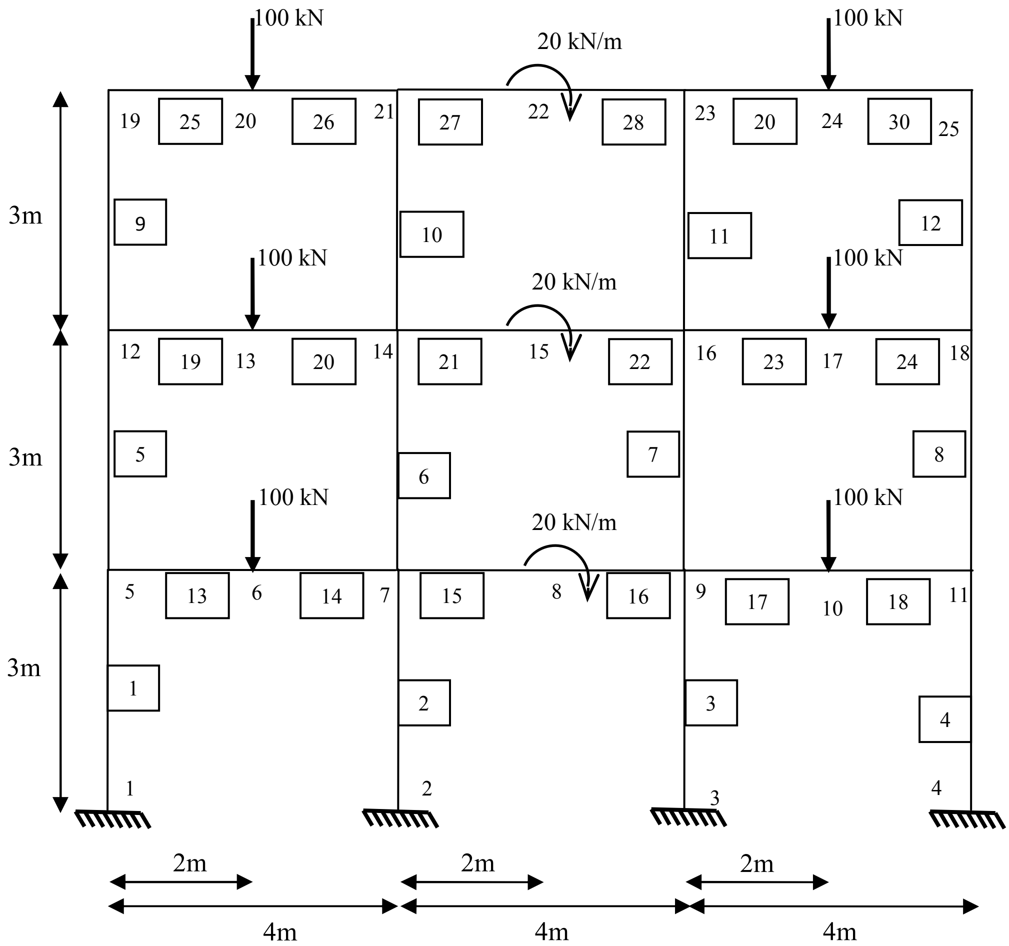


Figure 29: Example 1

To solve this problem first the Integrated Force Method Analysis Packages has been run and also the result from Mastan2 v3.2 is presented to compare the results.

The analysis procedure consists of the following phases:

a) Input Phase
    1. General Input
    2. Geometry Input
    3. Properties and Materials Input
    4. Load Data input
b) Calculation Phase
    1. Generating Equilibrium Equations and showing Matrix Plot of this matrix
    2. Generating Unconnected Flexibility Matrix and showing Matrix Plot of this Matrix
    3. Obtaining Compatibility Matrix from Null Space.
    4. Computing the Compatibility Conditions and showing Matrix Plot of this Matrix.
    5. Coupling the Compatibility Conditions with the Equilibrium Equations and showing Matrix Plot of this Matrix
    6. Forming Joint Load Vector
    7. Forming the Fixed End Forces Vector
    8. Combining Joint Load Vector with Fixed End Forces Vector
    9. Solving Independent Forces
c) Reporting Results Phase
    1. Displaying Nodal Displacements
    2. Displaying Member End Forces in Local Coordinate
    3. Displaying Support Reactions

4. Showing Diagrams of Axial Force, Shear Force and Bending Moment Diagrams.

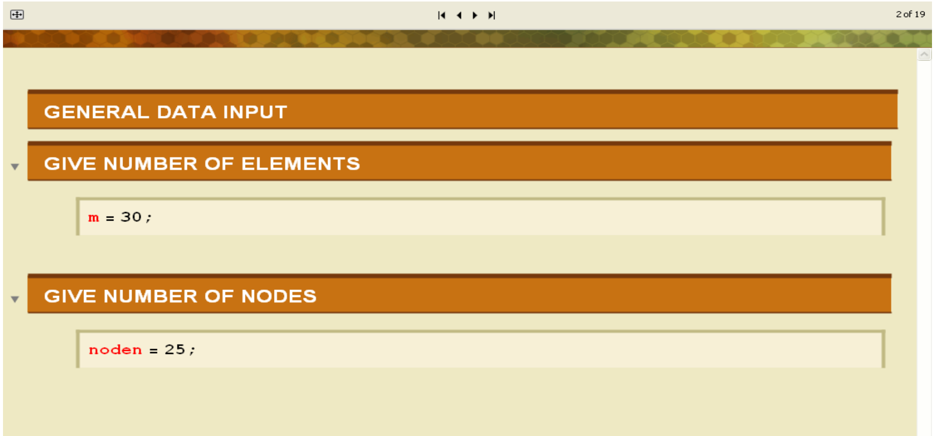


Figure 30: Input Phase, Step 1, General Data Input

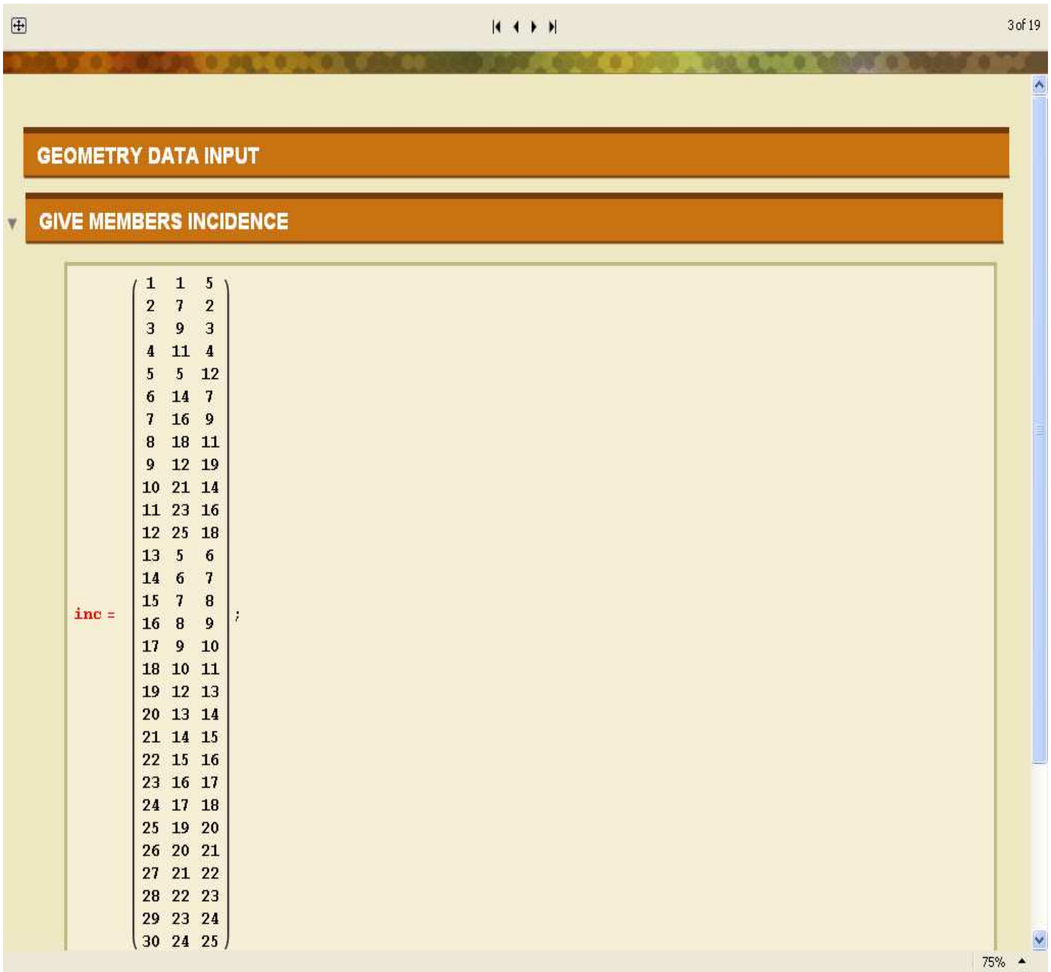


Figure 31: Input Phase, Step 2, Geometry Data Input (Member Incidence)

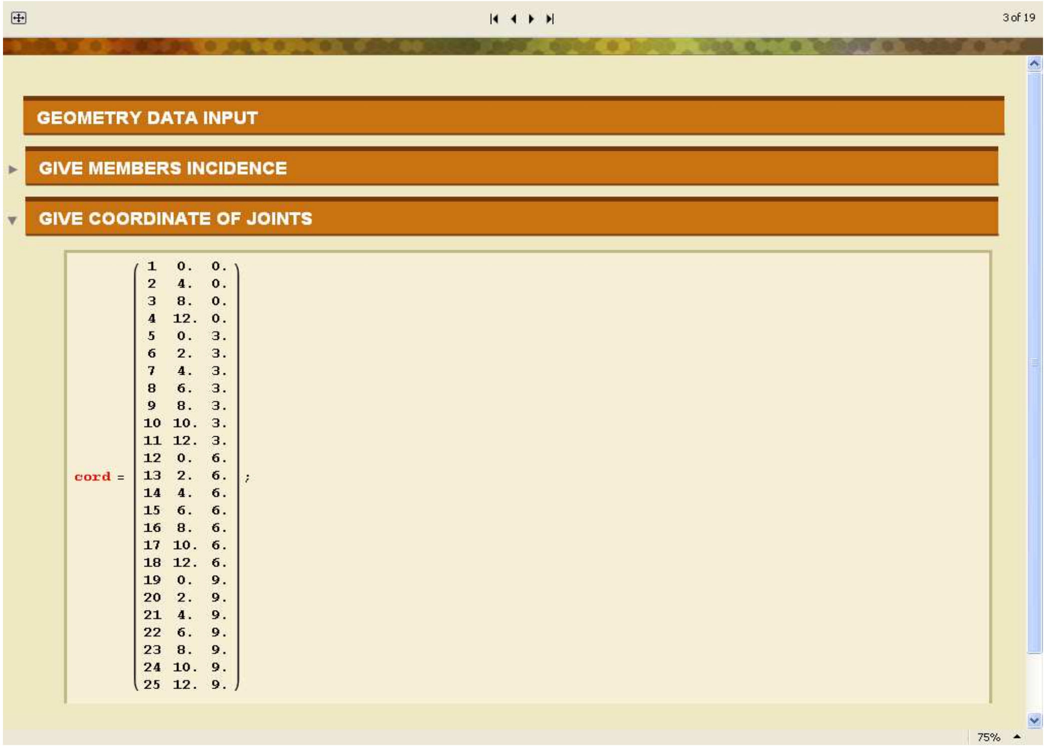


Figure 32: Input Phase, Step 2, Geometry Data Input (Coordinate of Joints)

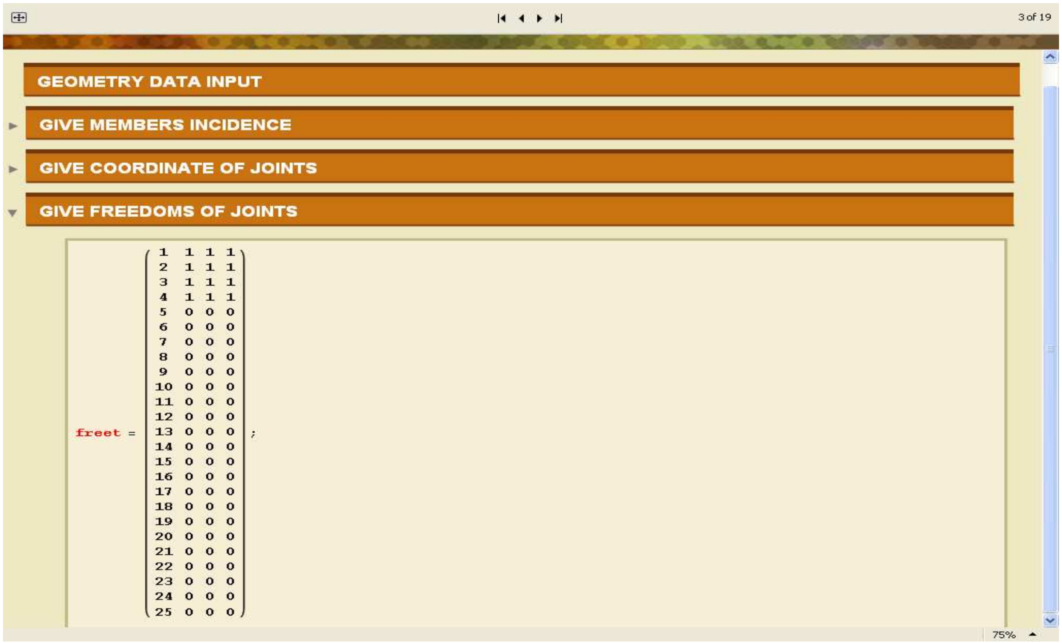


Figure 33: Input Phase, Step 2, Geometry Data Input (Freedom of Joints)

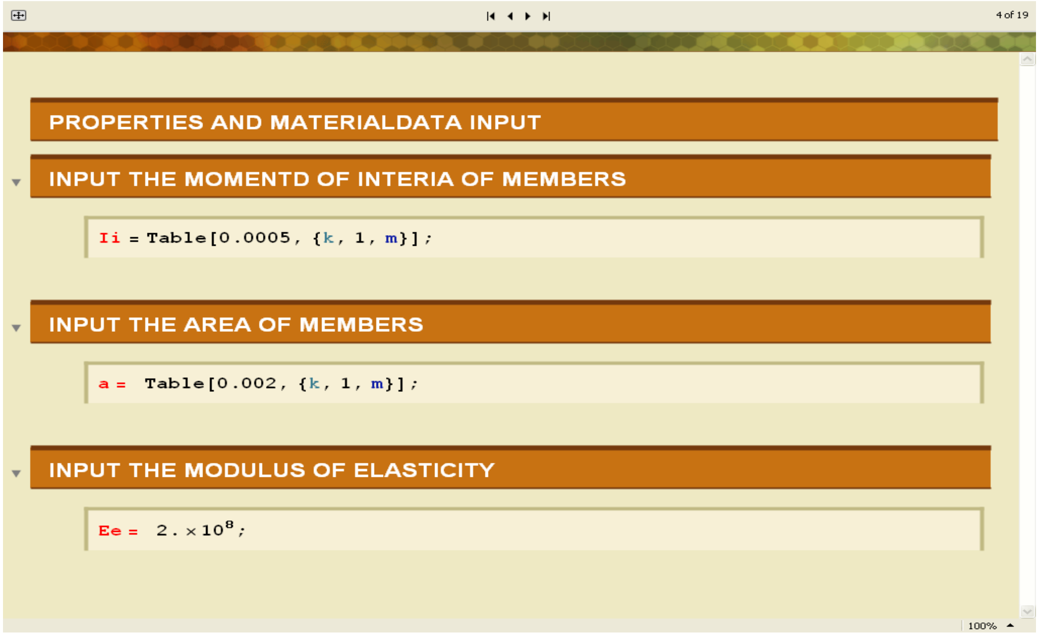


Figure 34: Input Phase, Step 3, Properties and Materials Input

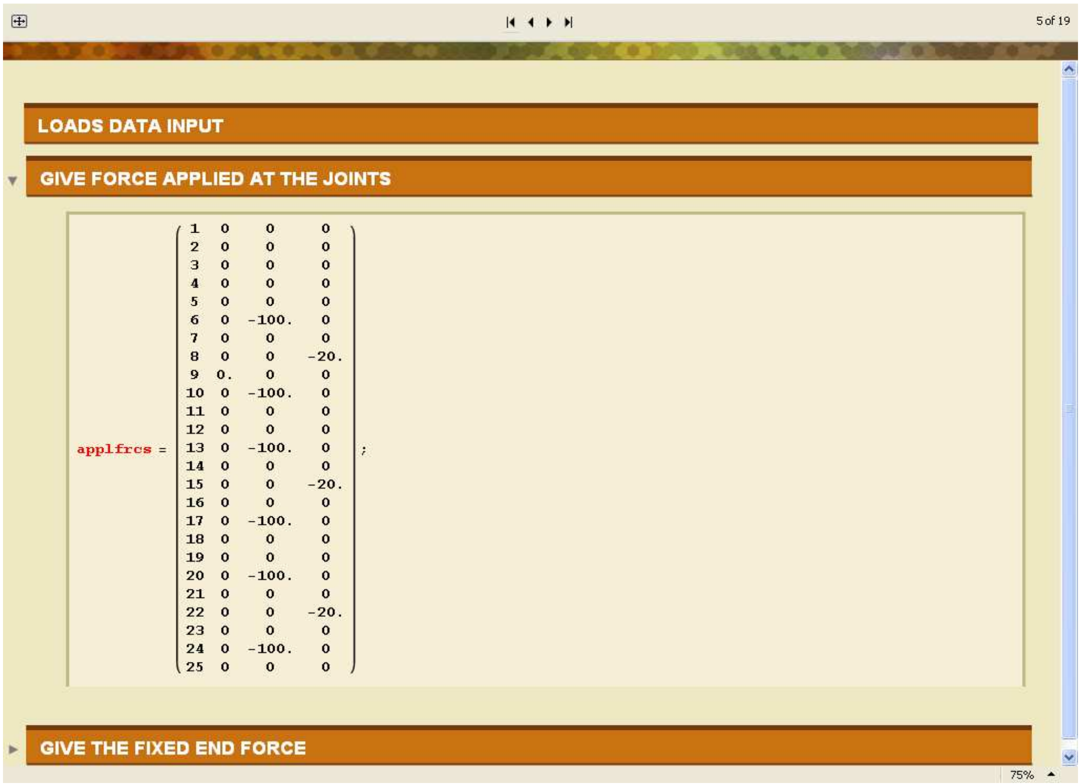


Figure 35: Input Phase, Step 4, Load Data Input (Joint Load)

Figure 36: Input Phase, Step 4, Load Data Input (Fixed End Forces)

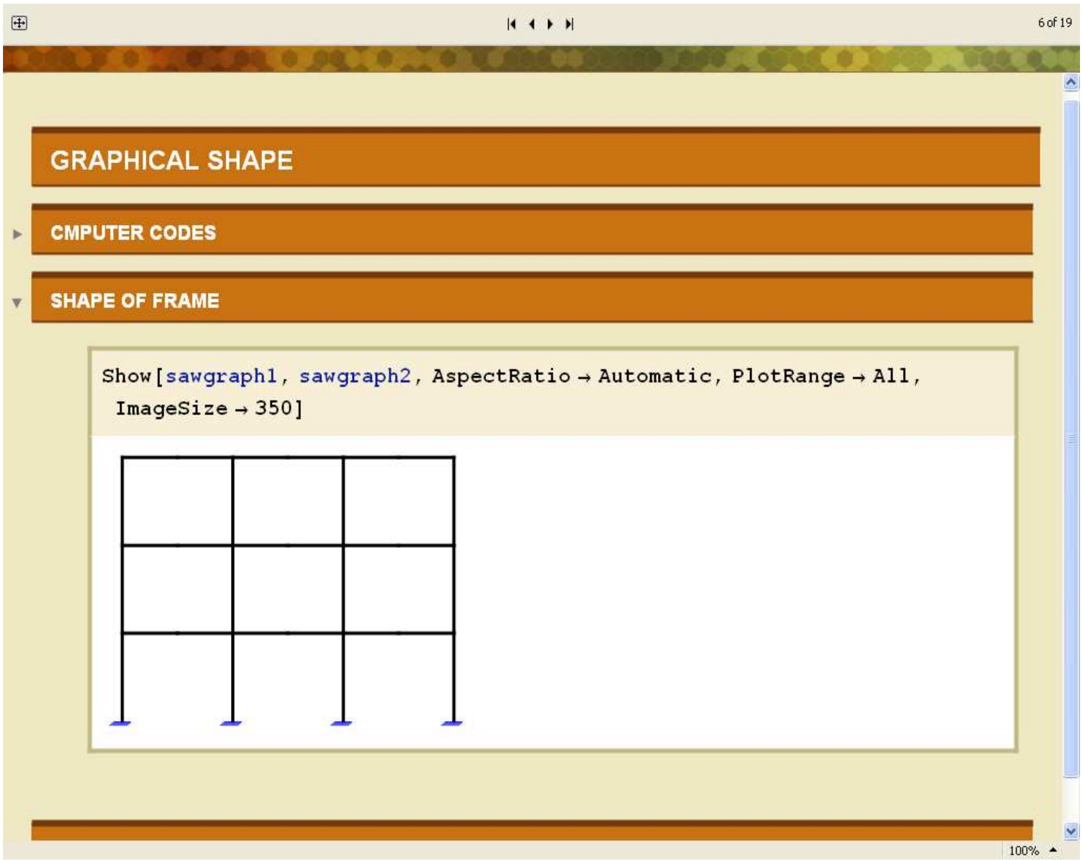


Figure 37: Shape of Frame

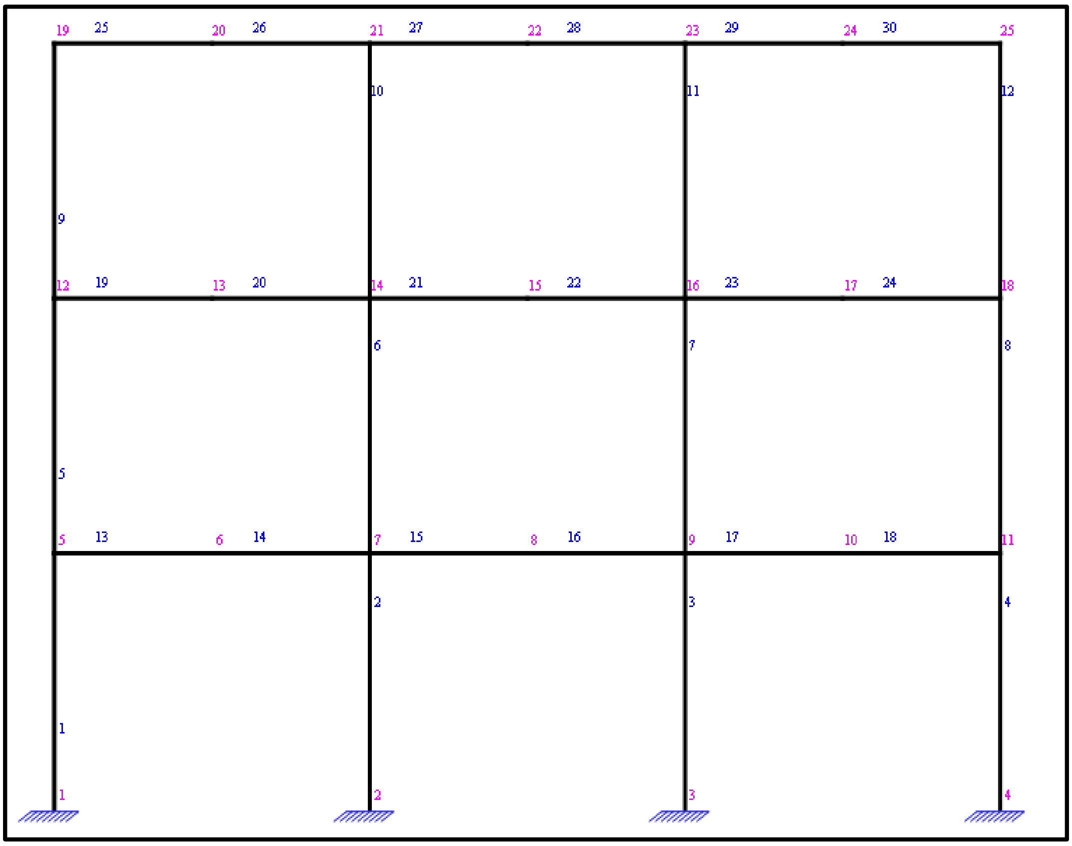


Figure 38: Element and Node Numbering

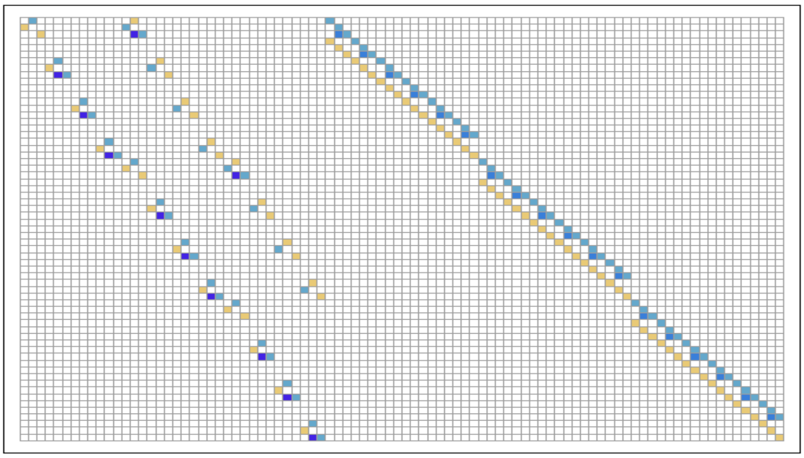


Figure 39: Calculation Phase, Step1, Matrix Plot of Equilibrium Equations [B] $_{63\times90}$

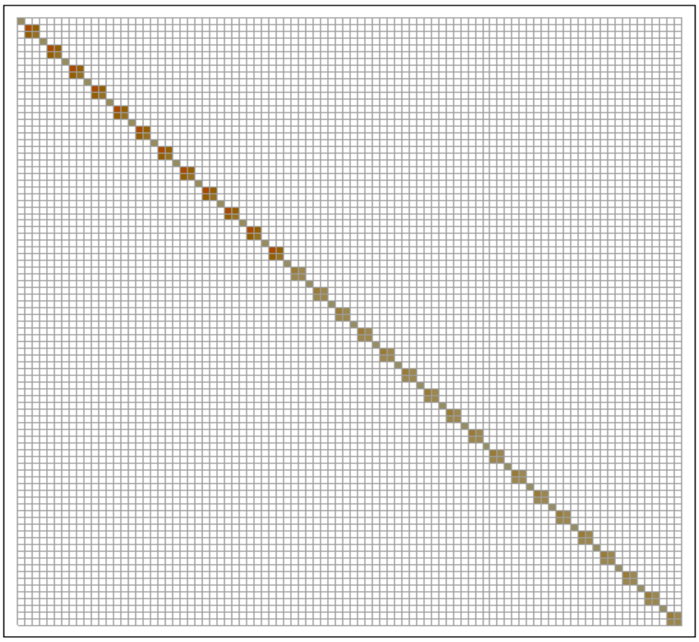


Figure 40: Calculation Phase, Step2, Matrix Plot of Unconnected Flexibility Matrix [G] $_{90\times90}$

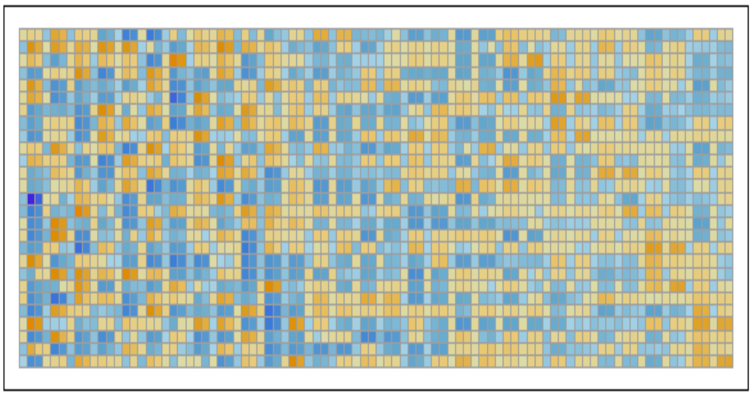


Figure 41 : Calculation Phase, Step4, Matrix Plot of Compatibility Conditions [C] $_{90\times27}$

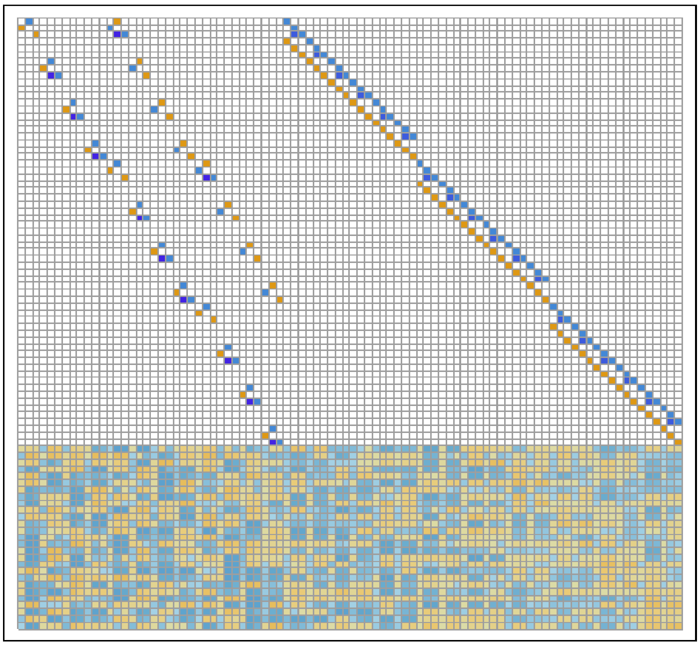


Figure 42: Calculation Phase, Step5, Matrix Plot of Coupled Equilibrium Equations with Compatibility Conditions [S] $_{90\times90}$

$$\text{node 1 displacements} = \begin{pmatrix} 0. \\ 0. \\ 0. \end{pmatrix}$$

$$\text{node 2 displacements} = \begin{pmatrix} 0. \\ 0. \\ 0. \end{pmatrix}$$

$$\text{node 3 displacements} = \begin{pmatrix} 0. \\ 0. \\ 0. \end{pmatrix}$$

$$\text{node 4 displacements} = \begin{pmatrix} 0. \\ 0. \\ 0. \end{pmatrix}$$

$$\text{node 5 displacements} = \begin{pmatrix} -0.0000551197 \\ -0.00109358 \\ -0.000122589 \end{pmatrix}$$

$$\text{node 6 displacements} = \begin{pmatrix} -0.0000324313 \\ -0.00154141 \\ 6.51231 \times 10^{-6} \end{pmatrix}$$

$$\text{node 7 displacements} = \begin{pmatrix} -9.74293 \times 10^{-6} \\ -0.00107387 \\ 0.000126115 \end{pmatrix}$$

$$\text{node 8 displacements} = \begin{pmatrix} -0.0000124682 \\ -0.00102146 \\ -0.000103515 \end{pmatrix}$$

$$\text{node 9 displacements} = \begin{pmatrix} -0.0000151934 \\ -0.00120714 \\ -0.000111971 \end{pmatrix}$$

$$\text{node 10 displacements} = \begin{pmatrix} 9.87893 \times 10^{-6} \\ -0.00162485 \\ 0.0000240148 \end{pmatrix}$$

$$\text{node 11 displacements} = \begin{pmatrix} 0.0000349512 \\ -0.00112541 \\ 0.000138516 \end{pmatrix}$$

$$\text{node 12 displacements} = \begin{pmatrix} -0.0000530933 \\ -0.00181394 \\ -0.000106163 \end{pmatrix}$$

$$\text{node 13 displacements} = \begin{pmatrix} -0.0000400641 \\ -0.00223626 \\ 9.77181 \times 10^{-6} \end{pmatrix}$$

Figure 43 : Reporting Results Phase, step1, Nodal Displacements of each node

$$\text{node 14 displacements} = \begin{pmatrix} -0.0000270349 \\ -0.00180448 \\ 0.0000812735 \end{pmatrix}$$

$$\text{node 15 displacements} = \begin{pmatrix} -0.0000360407 \\ -0.00182969 \\ -0.000128904 \end{pmatrix}$$

$$\text{node 16 displacements} = \begin{pmatrix} -0.0000450466 \\ -0.00200979 \\ -0.0000736194 \end{pmatrix}$$

$$\text{node 17 displacements} = \begin{pmatrix} -0.0000280723 \\ -0.00237554 \\ 0.0000378499 \end{pmatrix}$$

$$\text{node 18 displacements} = \begin{pmatrix} -0.000011098 \\ -0.00187179 \\ 0.000129217 \end{pmatrix}$$

$$\text{node 19 displacements} = \begin{pmatrix} 0.000124362 \\ -0.00216764 \\ -0.000278629 \end{pmatrix}$$

$$\text{node 20 displacements} = \begin{pmatrix} 0.0000355323 \\ -0.00274406 \\ 0.0000166012 \end{pmatrix}$$

$$\text{node 21 displacements} = \begin{pmatrix} -0.0000532971 \\ -0.00217699 \\ 0.0001982 \end{pmatrix}$$

$$\text{node 22 displacements} = \begin{pmatrix} -0.0000548047 \\ -0.00209623 \\ -0.000138593 \end{pmatrix}$$

$$\text{node 23 displacements} = \begin{pmatrix} -0.0000563123 \\ -0.00241597 \\ -0.000202304 \end{pmatrix}$$

$$\text{node 24 displacements} = \begin{pmatrix} -0.000152298 \\ -0.0029206 \\ 0.0000375753 \end{pmatrix}$$

$$\text{node 25 displacements} = \begin{pmatrix} -0.000248283 \\ -0.0022394 \\ 0.000316862 \end{pmatrix}$$

Figure 44: Reporting Results Phase, step1, Nodal Displacements of each node (continued)

member 1	endforces	$\begin{pmatrix} 145.811 \\ -10.6224 \\ -11.8472 \\ -145.811 \\ 10.6224 \\ -20.0198 \end{pmatrix}$
member 2	endforces	$\begin{pmatrix} 143.182 \\ 7.97463 \\ 7.75812 \\ -143.182 \\ -7.97463 \\ 16.1658 \end{pmatrix}$
member 3	endforces	$\begin{pmatrix} 160.953 \\ -8.14002 \\ -8.47765 \\ -160.953 \\ 8.14002 \\ -15.9424 \end{pmatrix}$
member 4	endforces	$\begin{pmatrix} 150.054 \\ 10.7878 \\ 11.5645 \\ -150.054 \\ -10.7878 \\ 20.7988 \end{pmatrix}$
member 5	endforces	$\begin{pmatrix} 96.0484 \\ -15.16 \\ -23.2876 \\ -96.0484 \\ 15.16 \\ -22.1925 \end{pmatrix}$
member 6	endforces	$\begin{pmatrix} 97.4152 \\ 13.0574 \\ 21.0807 \\ -97.4152 \\ -13.0574 \\ 18.0913 \end{pmatrix}$

Figure 45: Reporting Results Phase, step2, Member End Forces in Local Coordinate

member 7 endforces $\begin{pmatrix} 107.019 \\ -13.6995 \\ -21.8277 \\ -107.019 \\ 13.6995 \\ -19.2709 \end{pmatrix}$

member 8 endforces $\begin{pmatrix} 99.5173 \\ 15.8022 \\ 24.0133 \\ -99.5173 \\ -15.8022 \\ 23.3934 \end{pmatrix}$

member 9 endforces $\begin{pmatrix} 47.1592 \\ -17.7659 \\ -20.8999 \\ -47.1592 \\ 17.7659 \\ -32.3977 \end{pmatrix}$

member 10 endforces $\begin{pmatrix} 49.6679 \\ 17.4644 \\ 22.299 \\ -49.6679 \\ -17.4644 \\ 30.0941 \end{pmatrix}$

member 11 endforces $\begin{pmatrix} 54.1581 \\ -18.8956 \\ -24.0539 \\ -54.1581 \\ 18.8956 \\ -32.6328 \end{pmatrix}$

member 12 endforces $\begin{pmatrix} 49.0148 \\ 19.1971 \\ 22.5408 \\ -49.0148 \\ -19.1971 \\ 35.0505 \end{pmatrix}$

Figure 46: Reporting Results Phase, step2, Member End Forces in Local Coordinate (continued)

$$\text{member 13 endforces} \begin{pmatrix} -4.53768 \\ 49.7625 \\ 43.3075 \\ 4.53768 \\ -49.7625 \\ 56.2176 \end{pmatrix}$$

$$\text{member 14 endforces} \begin{pmatrix} -4.53768 \\ -50.2375 \\ -56.2176 \\ 4.53768 \\ 50.2375 \\ -44.2574 \end{pmatrix}$$

$$\text{member 15 endforces} \begin{pmatrix} 0.545045 \\ -4.47065 \\ 7.01085 \\ -0.545045 \\ 4.47065 \\ -15.9522 \end{pmatrix}$$

$$\text{member 16 endforces} \begin{pmatrix} 0.545045 \\ -4.47065 \\ -4.04785 \\ -0.545045 \\ 4.47065 \\ -4.89345 \end{pmatrix}$$

$$\text{member 17 endforces} \begin{pmatrix} -5.01446 \\ 49.4629 \\ 42.6636 \\ 5.01446 \\ -49.4629 \\ 56.2622 \end{pmatrix}$$

$$\text{member 18 endforces} \begin{pmatrix} -5.01446 \\ -50.5371 \\ -56.2622 \\ 5.01446 \\ 50.5371 \\ -44.8121 \end{pmatrix}$$

Figure 47: Reporting Results Phase, step2, Member End Forces in Local Coordinate (continued)

$$\text{member 19 endforces} \begin{pmatrix} -2.60584 \\ 48.8892 \\ 43.0925 \\ 2.60584 \\ -48.8892 \\ 54.6859 \end{pmatrix}$$

$$\text{member 20 endforces} \begin{pmatrix} -2.60584 \\ -51.1108 \\ -54.6859 \\ 2.60584 \\ 51.1108 \\ -47.5357 \end{pmatrix}$$

$$\text{member 21 endforces} \begin{pmatrix} 1.80117 \\ -3.36346 \\ 7.14541 \\ -1.80117 \\ 3.36346 \\ -13.8723 \end{pmatrix}$$

$$\text{member 22 endforces} \begin{pmatrix} 1.80117 \\ -3.36346 \\ -6.12768 \\ -1.80117 \\ 3.36346 \\ -0.599233 \end{pmatrix}$$

$$\text{member 23 endforces} \begin{pmatrix} -3.39486 \\ 49.4975 \\ 43.924 \\ 3.39486 \\ -49.4975 \\ 55.0709 \end{pmatrix}$$

$$\text{member 24 endforces} \begin{pmatrix} -3.39486 \\ -50.5025 \\ -55.0709 \\ 3.39486 \\ 50.5025 \\ -45.9342 \end{pmatrix}$$

Figure 48: Reporting Results Phase, step2, Member End Forces in Local Coordinate (continued)

$$\text{member 25 endforces} \begin{pmatrix} 17.7659 \\ 47.1592 \\ 32.3977 \\ -17.7659 \\ -47.1592 \\ 61.9207 \end{pmatrix}$$

$$\text{member 26 endforces} \begin{pmatrix} 17.7659 \\ -52.8408 \\ -61.9207 \\ -17.7659 \\ 52.8408 \\ -43.7608 \end{pmatrix}$$

$$\text{member 27 endforces} \begin{pmatrix} 0.301518 \\ -3.17292 \\ 13.6667 \\ -0.301518 \\ 3.17292 \\ -20.0126 \end{pmatrix}$$

$$\text{member 28 endforces} \begin{pmatrix} 0.301518 \\ -3.17292 \\ 0.012592 \\ -0.301518 \\ 3.17292 \\ -6.35844 \end{pmatrix}$$

$$\text{member 29 endforces} \begin{pmatrix} 19.1971 \\ 50.9852 \\ 38.9913 \\ -19.1971 \\ -50.9852 \\ 62.9791 \end{pmatrix}$$

$$\text{member 30 endforces} \begin{pmatrix} 19.1971 \\ -49.0148 \\ -62.9791 \\ -19.1971 \\ 49.0148 \\ -35.0505 \end{pmatrix}$$

Figure 49: Reporting Results Phase, step2, Member End Forces in Local Coordinate (continued)

```
reaction x=  10.6224
reaction y=  145.811
moment z=  -11.8472
member  1 react x  1
member  1 react y  1
member  1 react z  1
reaction x=  -7.97463
reaction y=  143.182
moment z=  7.75812
member  2 react x  2
member  2 react y  2
member  2 react z  2
reaction x=  8.14002
reaction y=  160.953
moment z=  -8.47765
member  3 react x  3
member  3 react y  3
member  3 react z  3
reaction x=  -10.7878
reaction y=  150.054
moment z=  11.5645
member  4 react x  4
member  4 react y  4
member  4 react z  4
```

Figure 50: Reporting Results Phase, step3, Support Reactions

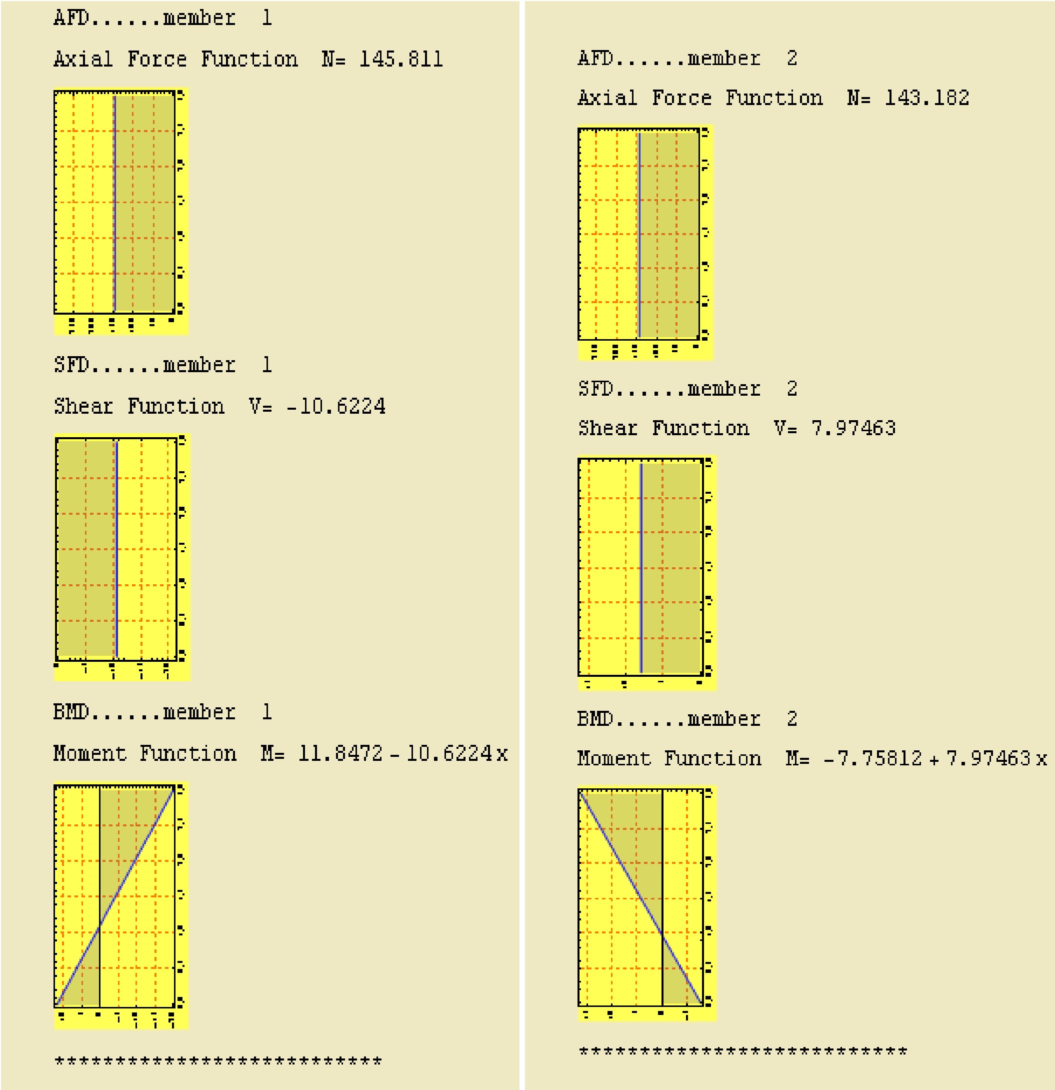


Figure 51: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 1, 2

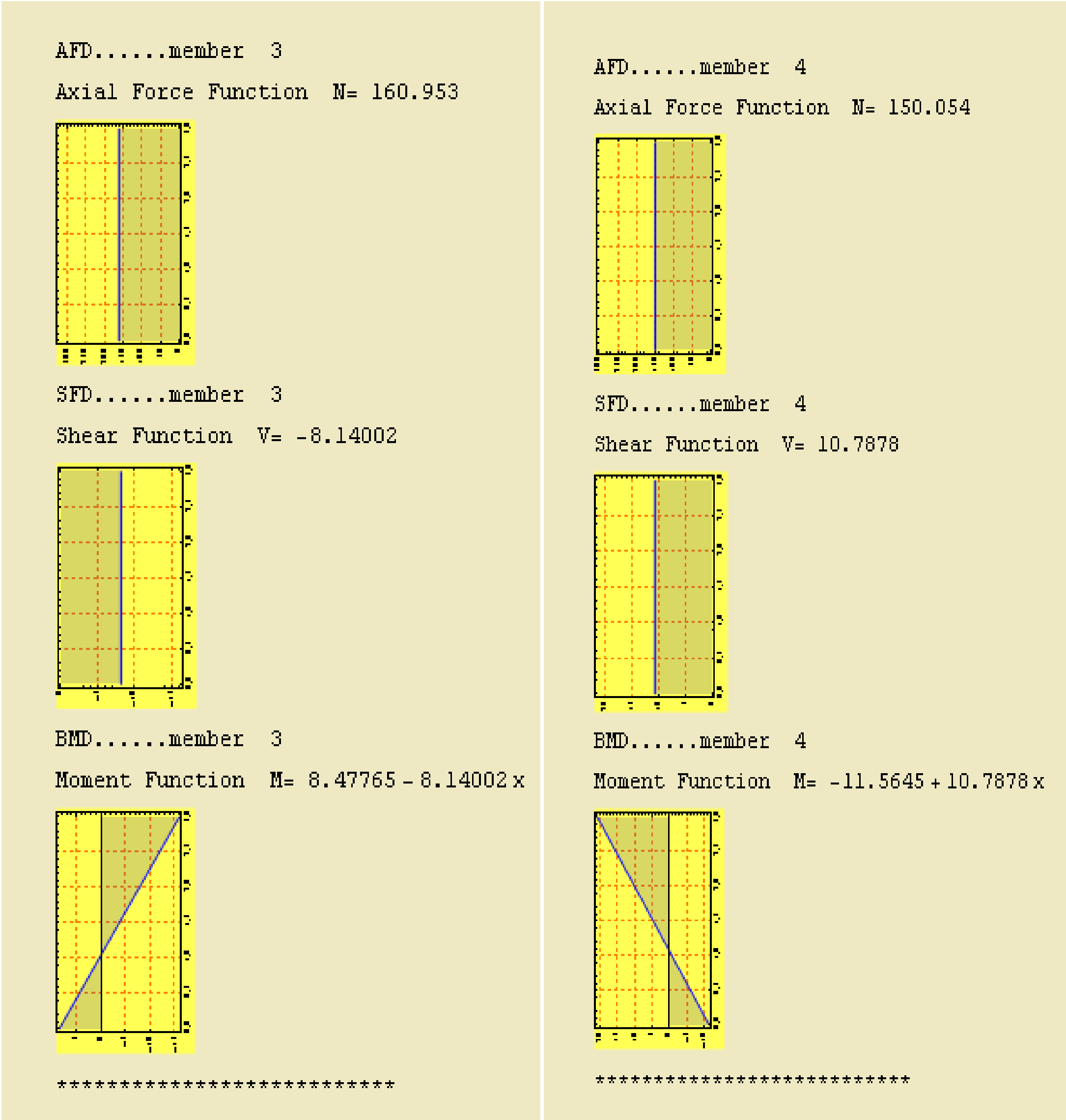


Figure 52: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 3, 4

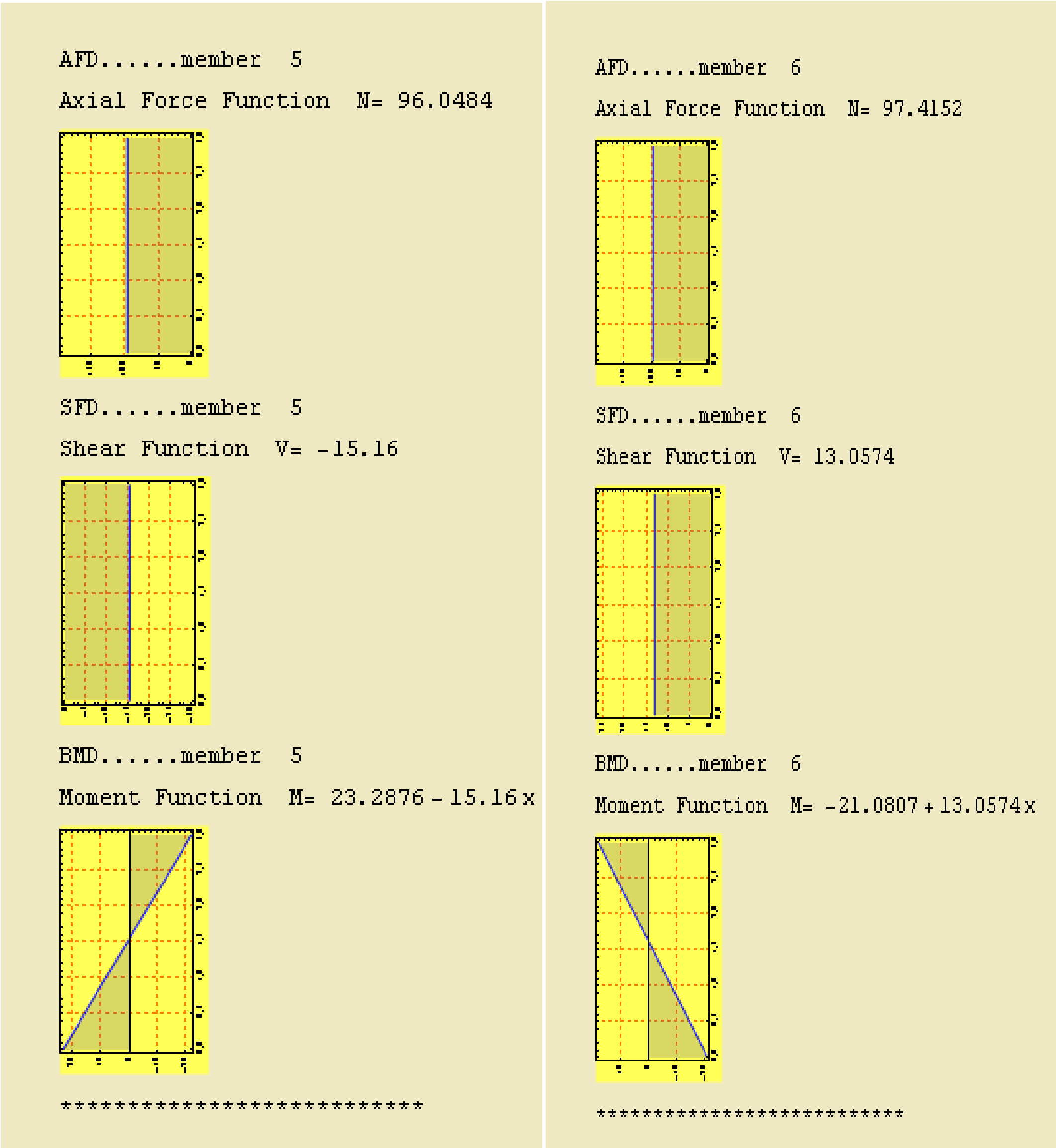


Figure 53: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 5, 6

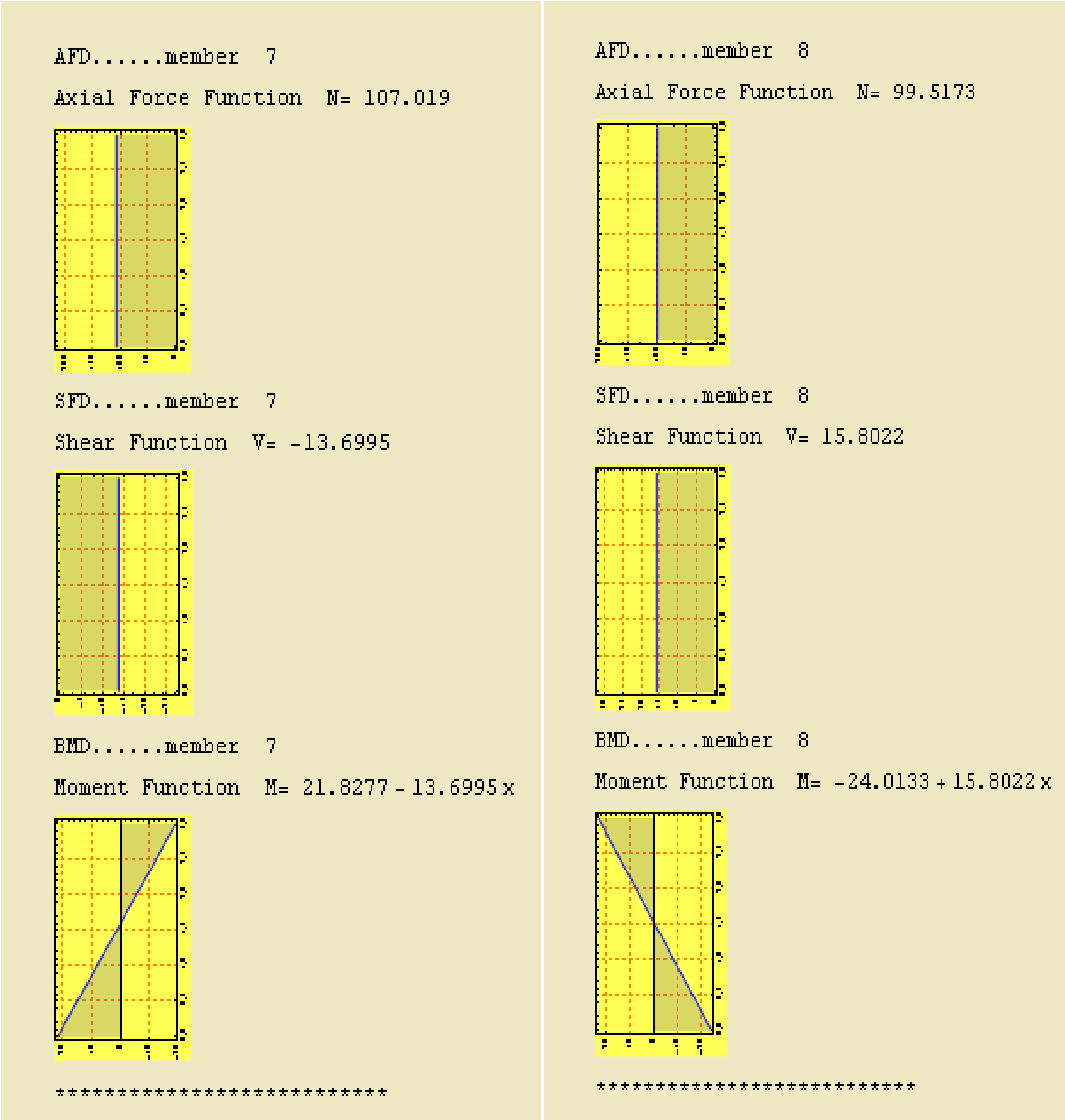


Figure 54: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 7, 8

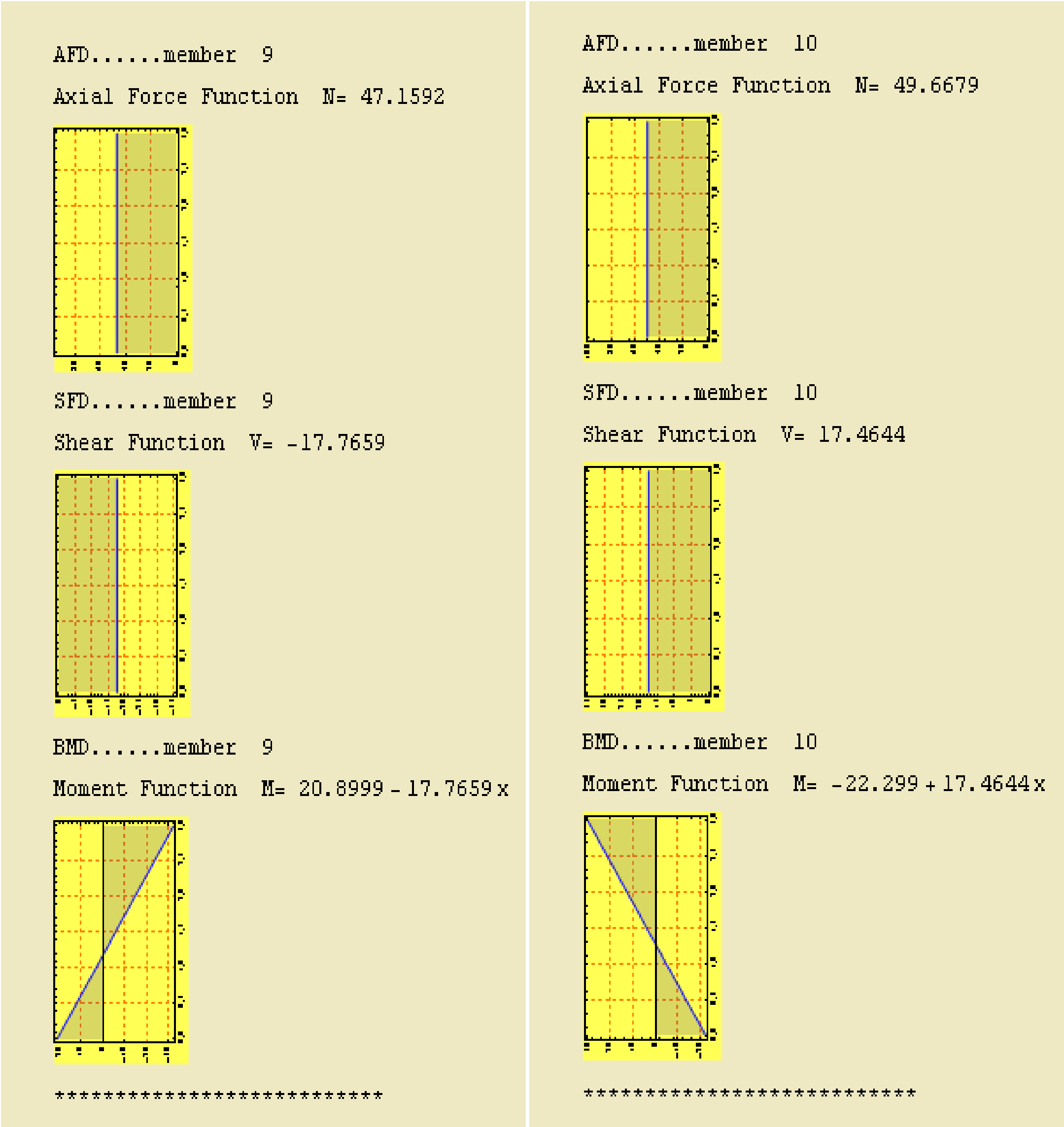


Figure 55: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 9, 10

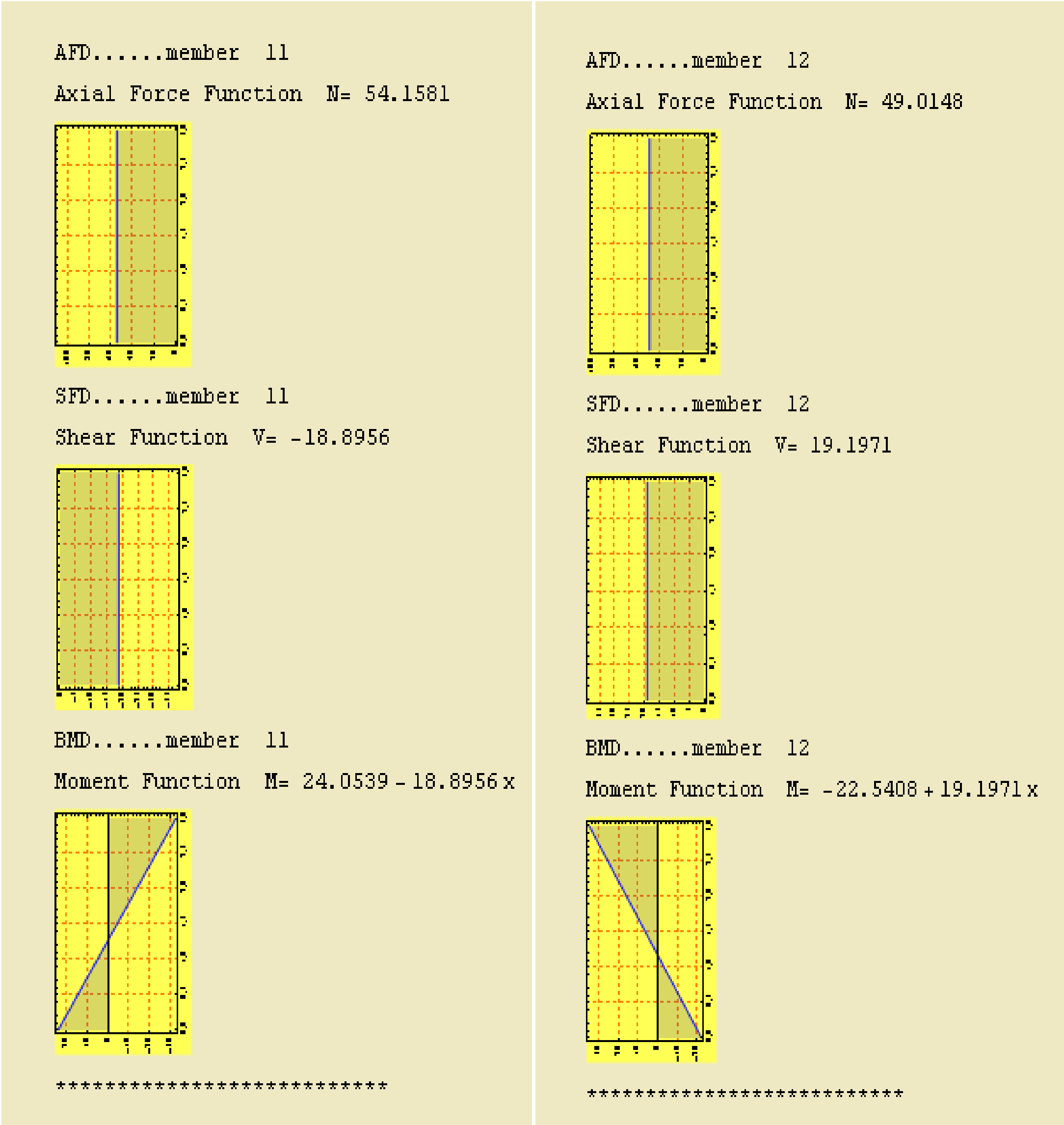


Figure 56: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member11, 12

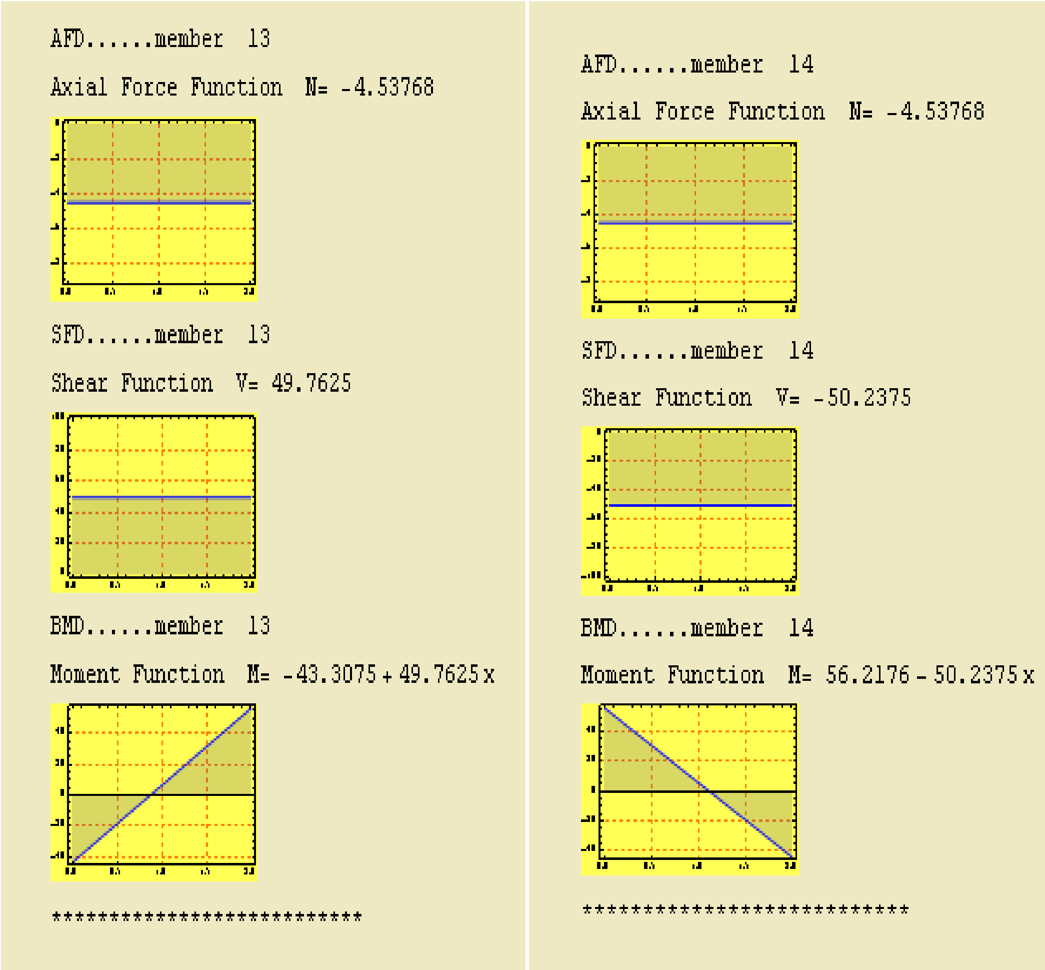


Figure 57: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member13, 14

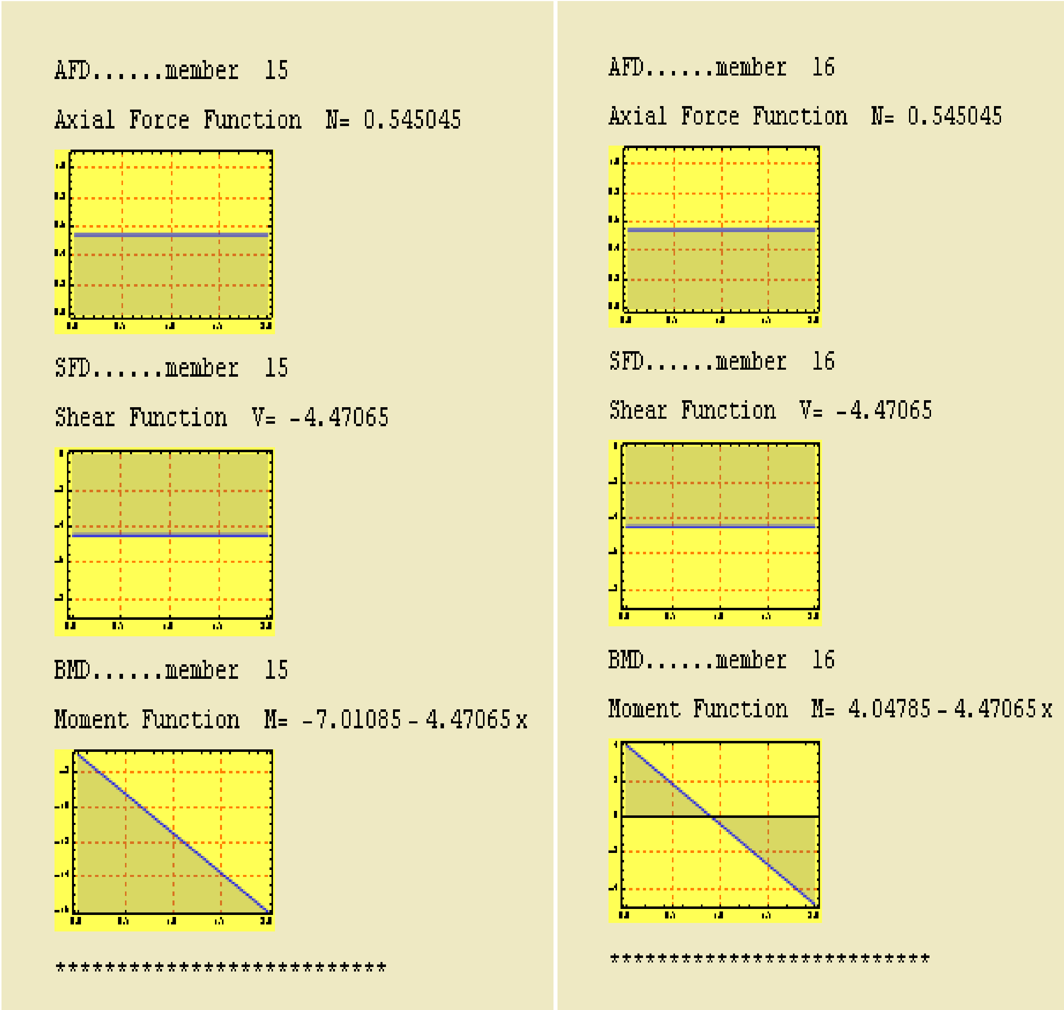


Figure 58: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member15, 16

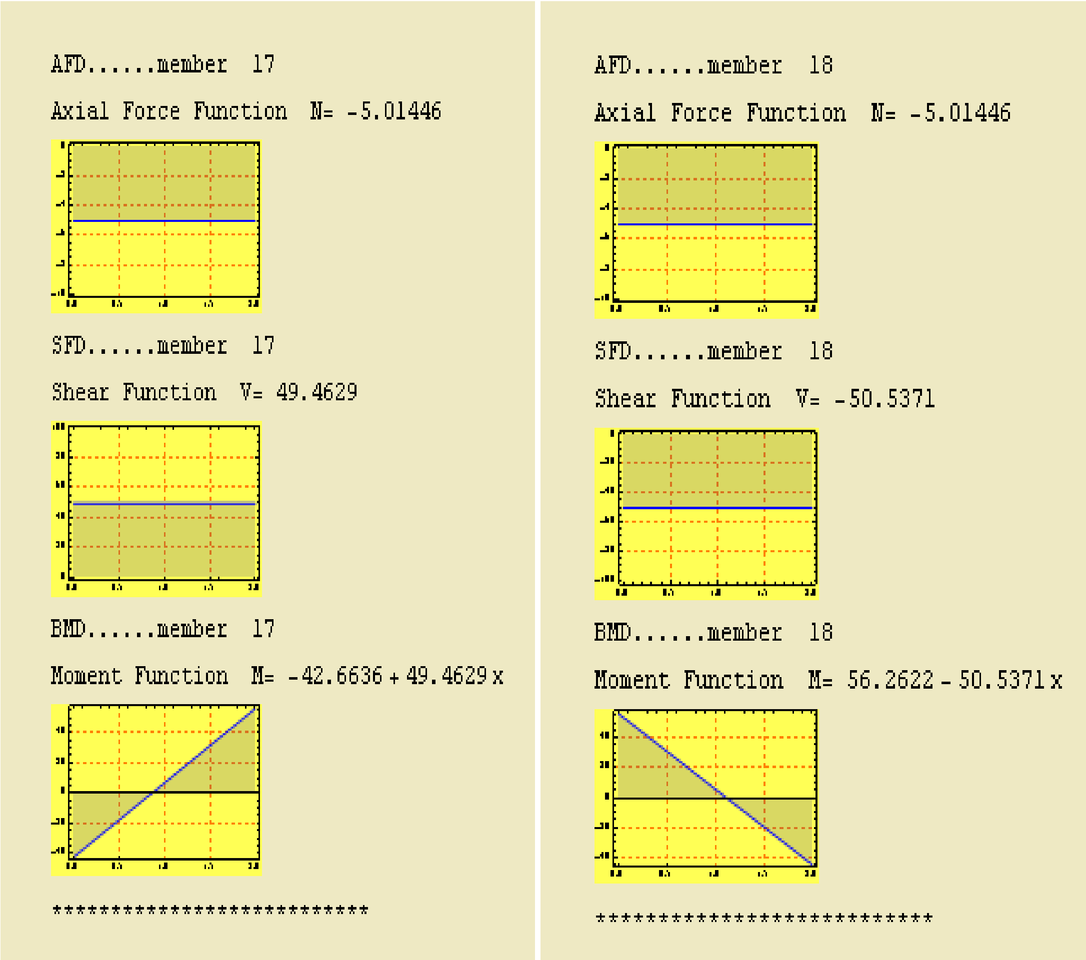


Figure 59: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member17, 18

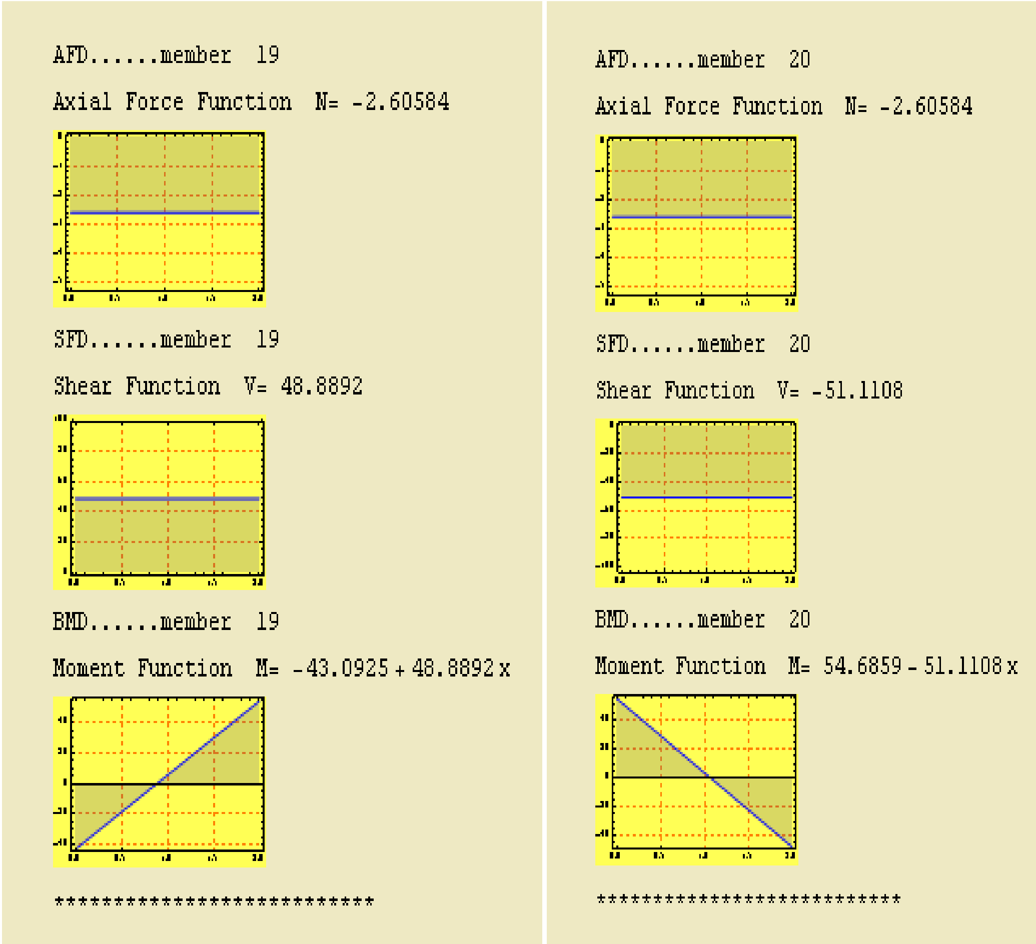


Figure 60: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member19, 20

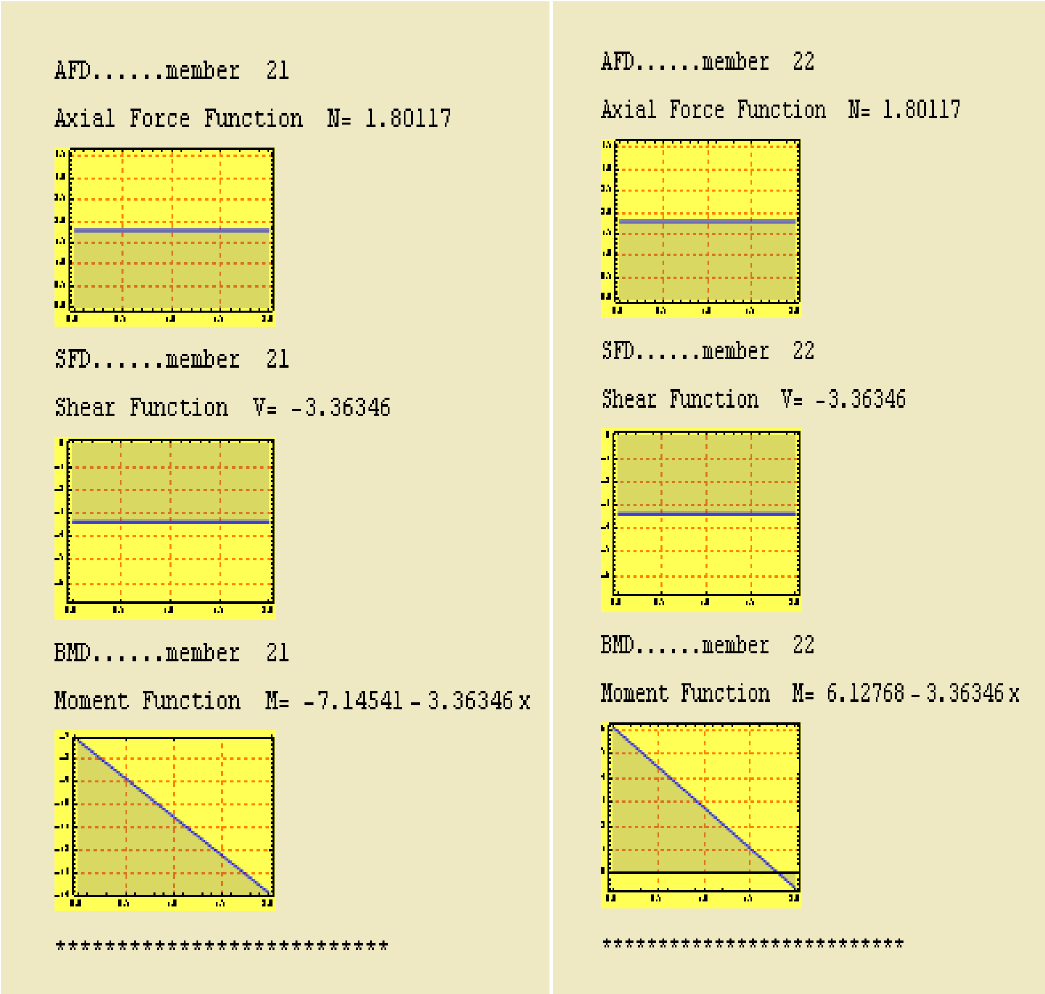


Figure 61: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member21, 22

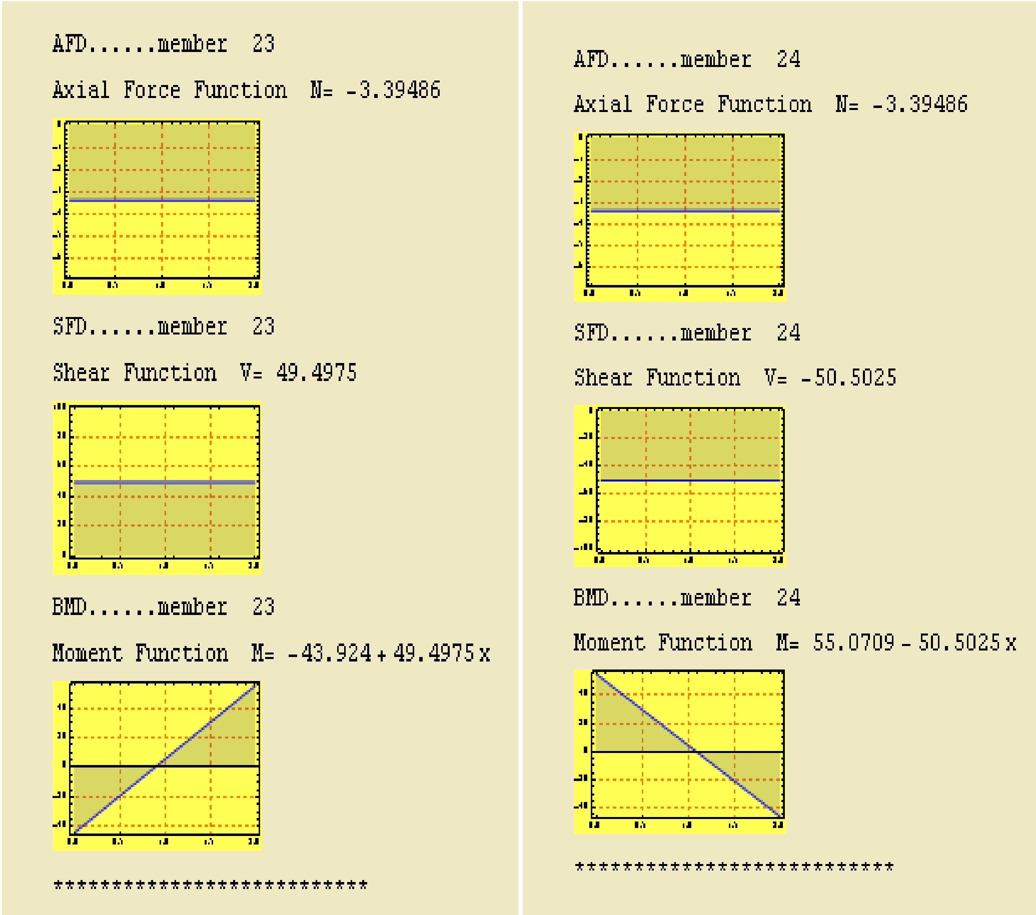


Figure 62: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member23, 24

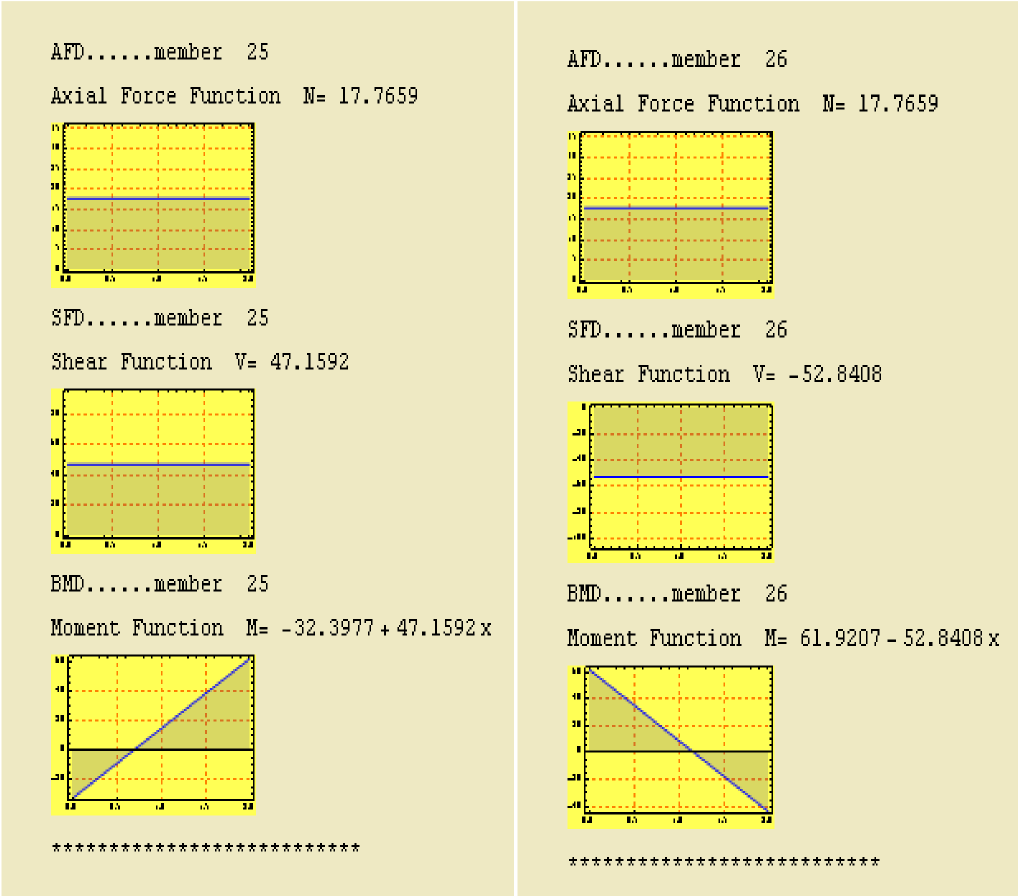


Figure 63: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member25, 26

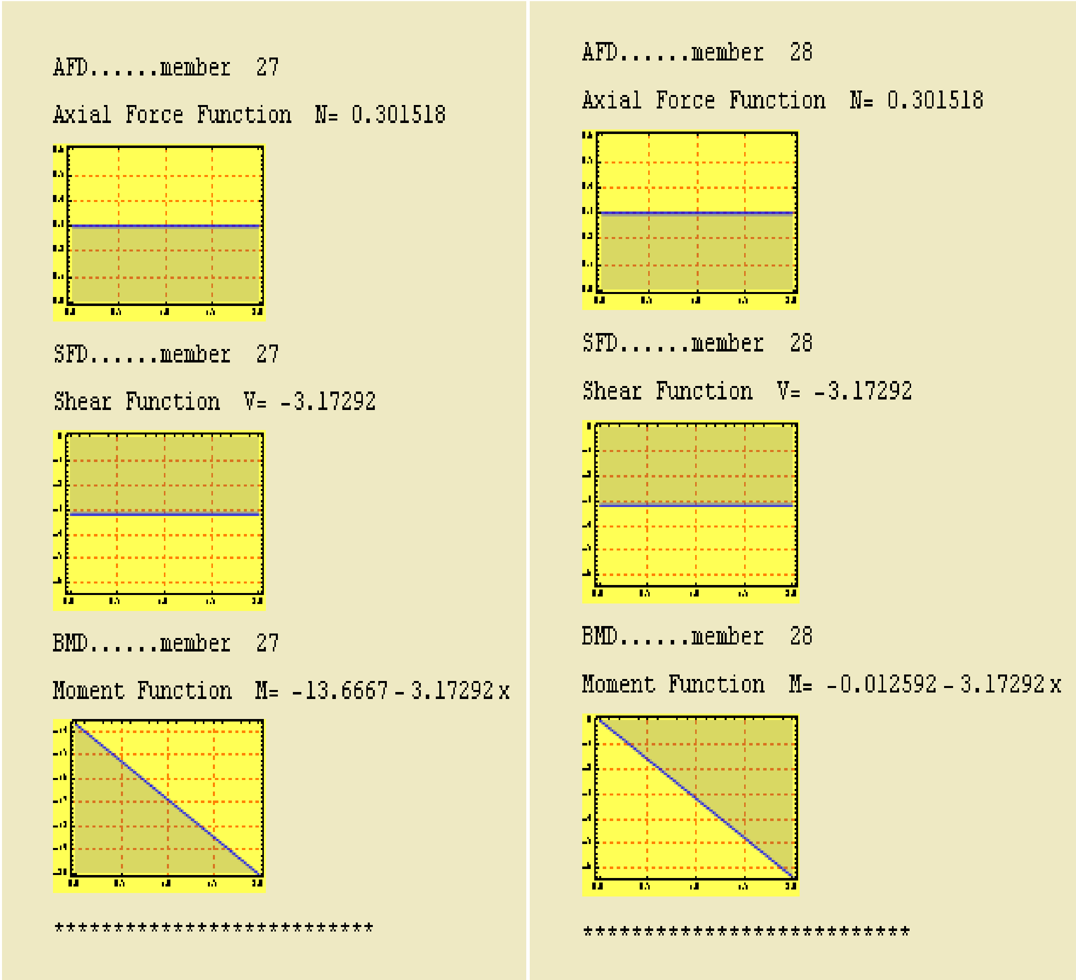


Figure 64: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member27, 28

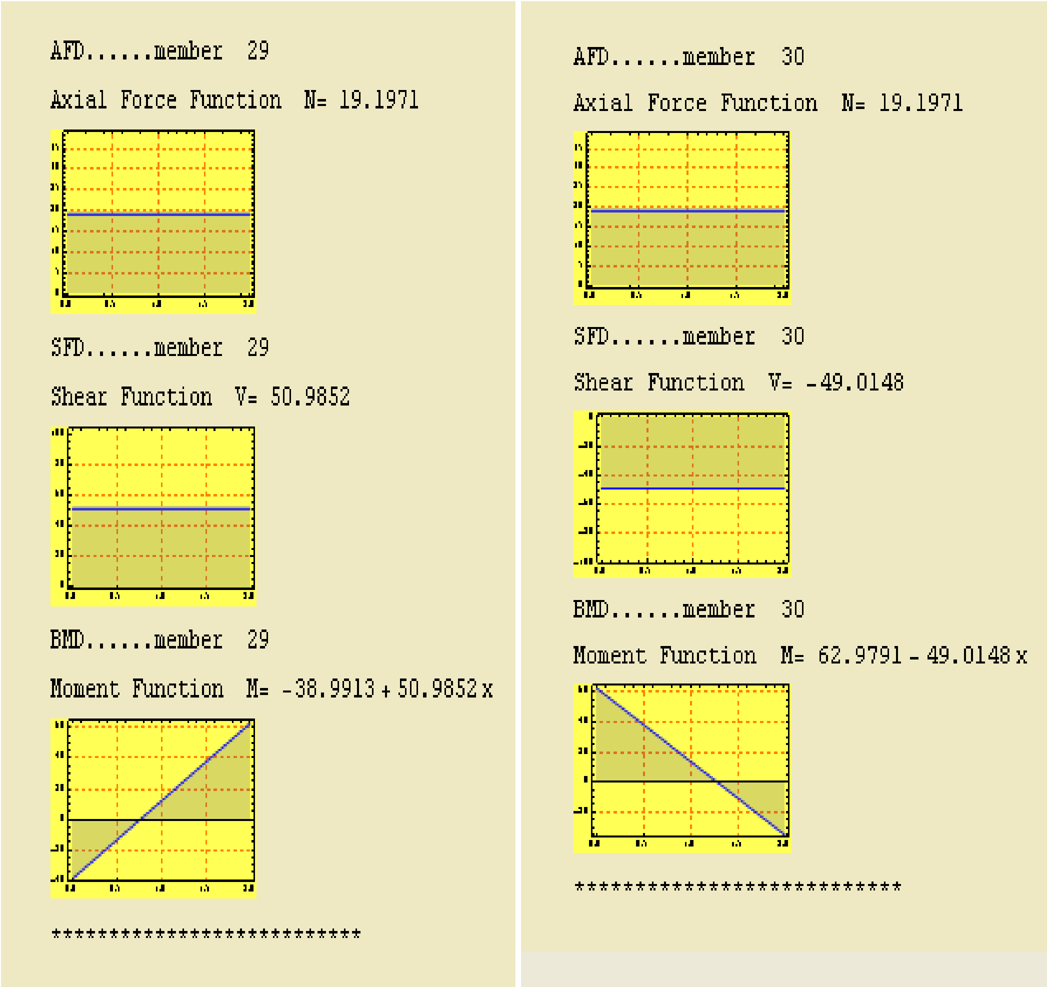


Figure 65: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member29, 30

```
###############################
Results of Structural Analysis
###############################

General Information:
      Structure Analyzed as:  Planar Frame
      Analysis Type:  First-Order Elastic

Analytical Results:

(i)   Displacements at Step # 1, Applied Load Ratio = 1.0000

  Deflections
     Node       X-disp          Y-disp          Z-disp
       1      0.0000e+000     0.0000e+000     0.0000e+000
       2      0.0000e+000     0.0000e+000     0.0000e+000
       3      0.0000e+000     0.0000e+000     0.0000e+000
       4      0.0000e+000     0.0000e+000     0.0000e+000
       5     -5.5120e-005    -1.0936e-003     0.0000e+000
       6     -3.2431e-005    -1.5414e-003     0.0000e+000
       7     -9.7429e-006    -1.0739e-003     0.0000e+000
       8     -1.2468e-005    -1.0215e-003     0.0000e+000
       9     -1.5193e-005    -1.2071e-003     0.0000e+000
      10      9.8789e-006    -1.6249e-003     0.0000e+000
      11      3.4951e-005    -1.1254e-003     0.0000e+000
      12     -5.3093e-005    -1.8139e-003     0.0000e+000
      13     -4.0064e-005    -2.2363e-003     0.0000e+000
      14     -2.7035e-005    -1.8045e-003     0.0000e+000
      15     -3.6041e-005    -1.8297e-003     0.0000e+000
      16     -4.5047e-005    -2.0098e-003     0.0000e+000
      17     -2.8072e-005    -2.3755e-003     0.0000e+000
      18     -1.1098e-005    -1.8718e-003     0.0000e+000
      19      1.2436e-004    -2.1676e-003     0.0000e+000
      20      3.5532e-005    -2.7441e-003     0.0000e+000
      21     -5.3297e-005    -2.1770e-003     0.0000e+000
      22     -5.4805e-005    -2.0962e-003     0.0000e+000
      23     -5.6312e-005    -2.4160e-003     0.0000e+000
      24     -1.5230e-004    -2.9206e-003     0.0000e+000
      25     -2.4828e-004    -2.2394e-003     0.0000e+000
```

Figure 66: Final results of Example 1 from Mastan2 v 3.2

```
 Rotations (radians)
    Node        X-rot          Y-rot          Z-rot
       1     0.0000e+000    0.0000e+000    0.0000e+000
       2     0.0000e+000    0.0000e+000    0.0000e+000
       3     0.0000e+000    0.0000e+000    0.0000e+000
       4     0.0000e+000    0.0000e+000    0.0000e+000
       5     0.0000e+000    0.0000e+000   -1.2259e-004
       6     0.0000e+000    0.0000e+000    6.5123e-006
       7     0.0000e+000    0.0000e+000    1.2611e-004
       8     0.0000e+000    0.0000e+000   -1.0352e-004
       9     0.0000e+000    0.0000e+000   -1.1197e-004
      10     0.0000e+000    0.0000e+000    2.4015e-005
      11     0.0000e+000    0.0000e+000    1.3852e-004
      12     0.0000e+000    0.0000e+000   -1.0616e-004
      13     0.0000e+000    0.0000e+000    9.7718e-006
      14     0.0000e+000    0.0000e+000    8.1274e-005
      15     0.0000e+000    0.0000e+000   -1.2890e-004
      16     0.0000e+000    0.0000e+000   -7.3619e-005
      17     0.0000e+000    0.0000e+000    3.7850e-005
      18     0.0000e+000    0.0000e+000    1.2922e-004
      19     0.0000e+000    0.0000e+000   -2.7863e-004
      20     0.0000e+000    0.0000e+000    1.6601e-005
      21     0.0000e+000    0.0000e+000    1.9820e-004
      22     0.0000e+000    0.0000e+000   -1.3859e-004
      23     0.0000e+000    0.0000e+000   -2.0230e-004
      24     0.0000e+000    0.0000e+000    3.7575e-005
      25     0.0000e+000    0.0000e+000    3.1686e-004

(ii)  Element Results at Step # 1, Applied Load Ratio = 1.0000

 Internal End Forces (Note: Refers to local coordinates)
   Element  Node          Fx             Fy             Fz
      1       1     1.4581e+002   -1.0622e+001    0.0000e+000
              5    -1.4581e+002    1.0622e+001    0.0000e+000
      2       2     1.4318e+002    7.9746e+000    0.0000e+000
              7    -1.4318e+002   -7.9746e+000    0.0000e+000
      3       3     1.6095e+002   -8.1400e+000    0.0000e+000
              9    -1.6095e+002    8.1400e+000    0.0000e+000
      4       4     1.5005e+002    1.0788e+001    0.0000e+000
             11    -1.5005e+002   -1.0788e+001    0.0000e+000
      5       5     9.6048e+001   -1.5160e+001    0.0000e+000
             12    -9.6048e+001    1.5160e+001    0.0000e+000
      6       7     9.7415e+001    1.3057e+001    0.0000e+000
             14    -9.7415e+001   -1.3057e+001    0.0000e+000
      7       9     1.0702e+002   -1.3700e+001    0.0000e+000
             16    -1.0702e+002    1.3700e+001    0.0000e+000
      8      11     9.9517e+001    1.5802e+001    0.0000e+000
             18    -9.9517e+001   -1.5802e+001    0.0000e+000
```

Figure 67: Final results of Example 1 from Mastan2 v 3.2 (continued)

9	12	4.7159e+001	-1.7766e+001	0.0000e+000
	19	-4.7159e+001	1.7766e+001	0.0000e+000
10	14	4.9668e+001	1.7464e+001	0.0000e+000
	21	-4.9668e+001	-1.7464e+001	0.0000e+000
11	16	5.4158e+001	-1.8896e+001	0.0000e+000
	23	-5.4158e+001	1.8896e+001	0.0000e+000
12	18	4.9015e+001	1.9197e+001	0.0000e+000
	25	-4.9015e+001	-1.9197e+001	0.0000e+000
13	5	-4.5377e+000	4.9763e+001	0.0000e+000
	6	4.5377e+000	-4.9763e+001	0.0000e+000
14	6	-4.5377e+000	-5.0237e+001	0.0000e+000
	7	4.5377e+000	5.0237e+001	0.0000e+000
15	7	5.4505e-001	-4.4706e+000	0.0000e+000
	8	-5.4505e-001	4.4706e+000	0.0000e+000
16	8	5.4505e-001	-4.4706e+000	0.0000e+000
	9	-5.4505e-001	4.4706e+000	0.0000e+000
17	9	-5.0145e+000	4.9463e+001	0.0000e+000
	10	5.0145e+000	-4.9463e+001	0.0000e+000
18	10	-5.0145e+000	-5.0537e+001	0.0000e+000
	11	5.0145e+000	5.0537e+001	0.0000e+000
19	12	-2.6058e+000	4.8889e+001	0.0000e+000
	13	2.6058e+000	-4.8889e+001	0.0000e+000
20	13	-2.6058e+000	-5.1111e+001	0.0000e+000
	14	2.6058e+000	5.1111e+001	0.0000e+000
21	14	1.8012e+000	-3.3635e+000	0.0000e+000
	15	-1.8012e+000	3.3635e+000	0.0000e+000
22	15	1.8012e+000	-3.3635e+000	0.0000e+000
	16	-1.8012e+000	3.3635e+000	0.0000e+000
23	16	-3.3949e+000	4.9497e+001	0.0000e+000
	17	3.3949e+000	-4.9497e+001	0.0000e+000
24	17	-3.3949e+000	-5.0503e+001	0.0000e+000
	18	3.3949e+000	5.0503e+001	0.0000e+000
25	19	1.7766e+001	4.7159e+001	0.0000e+000
	20	-1.7766e+001	-4.7159e+001	0.0000e+000
26	20	1.7766e+001	-5.2841e+001	0.0000e+000
	21	-1.7766e+001	5.2841e+001	0.0000e+000
27	21	3.0152e-001	-3.1729e+000	0.0000e+000
	22	-3.0152e-001	3.1729e+000	0.0000e+000
28	22	3.0152e-001	-3.1729e+000	0.0000e+000
	23	-3.0152e-001	3.1729e+000	0.0000e+000
29	23	1.9197e+001	5.0985e+001	0.0000e+000
	24	-1.9197e+001	-5.0985e+001	0.0000e+000
30	24	1.9197e+001	-4.9015e+001	0.0000e+000
	25	-1.9197e+001	4.9015e+001	0.0000e+000

Figure 68: Final results of Example 1 from Mastan2 v 3.2 (continued)

Internal End Moments (Note: Refers to local coordinates)

Element	Node	Mx	My	Mz	B
1	1	0.0000e+000	0.0000e+000	-1.1847e+001	0.0000e+000
	5	0.0000e+000	0.0000e+000	-2.0020e+001	0.0000e+000
2	2	0.0000e+000	0.0000e+000	7.7581e+000	0.0000e+000
	7	0.0000e+000	0.0000e+000	1.6166e+001	0.0000e+000
3	3	0.0000e+000	0.0000e+000	-8.4777e+000	0.0000e+000
	9	0.0000e+000	0.0000e+000	-1.5942e+001	0.0000e+000
4	4	0.0000e+000	0.0000e+000	1.1564e+001	0.0000e+000
	11	0.0000e+000	0.0000e+000	2.0799e+001	0.0000e+000
5	5	0.0000e+000	0.0000e+000	-2.3288e+001	0.0000e+000
	12	0.0000e+000	0.0000e+000	-2.2193e+001	0.0000e+000
6	7	0.0000e+000	0.0000e+000	2.1081e+001	0.0000e+000
	14	0.0000e+000	0.0000e+000	1.8091e+001	0.0000e+000
7	9	0.0000e+000	0.0000e+000	-2.1828e+001	0.0000e+000
	16	0.0000e+000	0.0000e+000	-1.9271e+001	0.0000e+000
8	11	0.0000e+000	0.0000e+000	2.4013e+001	0.0000e+000
	18	0.0000e+000	0.0000e+000	2.3393e+001	0.0000e+000
9	12	0.0000e+000	0.0000e+000	-2.0900e+001	0.0000e+000
	19	0.0000e+000	0.0000e+000	-3.2398e+001	0.0000e+000
10	14	0.0000e+000	0.0000e+000	2.2299e+001	0.0000e+000
	21	0.0000e+000	0.0000e+000	3.0094e+001	0.0000e+000
11	16	0.0000e+000	0.0000e+000	-2.4054e+001	0.0000e+000
	23	0.0000e+000	0.0000e+000	-3.2633e+001	0.0000e+000
12	18	0.0000e+000	0.0000e+000	2.2541e+001	0.0000e+000
	25	0.0000e+000	0.0000e+000	3.5050e+001	0.0000e+000
13	5	0.0000e+000	0.0000e+000	4.3307e+001	0.0000e+000
	6	0.0000e+000	0.0000e+000	5.6218e+001	0.0000e+000
14	6	0.0000e+000	0.0000e+000	-5.6218e+001	0.0000e+000
	7	0.0000e+000	0.0000e+000	-4.4257e+001	0.0000e+000
15	7	0.0000e+000	0.0000e+000	7.0109e+000	0.0000e+000
	8	0.0000e+000	0.0000e+000	-1.5952e+001	0.0000e+000
16	8	0.0000e+000	0.0000e+000	-4.0478e+000	0.0000e+000
	9	0.0000e+000	0.0000e+000	-4.8935e+000	0.0000e+000
17	9	0.0000e+000	0.0000e+000	4.2664e+001	0.0000e+000
	10	0.0000e+000	0.0000e+000	5.6262e+001	0.0000e+000
18	10	0.0000e+000	0.0000e+000	-5.6262e+001	0.0000e+000
	11	0.0000e+000	0.0000e+000	-4.4812e+001	0.0000e+000
19	12	0.0000e+000	0.0000e+000	4.3092e+001	0.0000e+000
	13	0.0000e+000	0.0000e+000	5.4686e+001	0.0000e+000
20	13	0.0000e+000	0.0000e+000	-5.4686e+001	0.0000e+000
	14	0.0000e+000	0.0000e+000	-4.7536e+001	0.0000e+000
21	14	0.0000e+000	0.0000e+000	7.1454e+000	0.0000e+000
	15	0.0000e+000	0.0000e+000	-1.3872e+001	0.0000e+000
22	15	0.0000e+000	0.0000e+000	-6.1277e+000	0.0000e+000
	16	0.0000e+000	0.0000e+000	-5.9923e-001	0.0000e+000

Figure 69: Final results of Example 1 from Mastan2 v 3.2 (continued)

23	16	0.0000e+000	0.0000e+000	4.3924e+001	0.0000e+000
	17	0.0000e+000	0.0000e+000	5.5071e+001	0.0000e+000
24	17	0.0000e+000	0.0000e+000	-5.5071e+001	0.0000e+000
	18	0.0000e+000	0.0000e+000	-4.5934e+001	0.0000e+000
25	19	0.0000e+000	0.0000e+000	3.2398e+001	0.0000e+000
	20	0.0000e+000	0.0000e+000	6.1921e+001	0.0000e+000
26	20	0.0000e+000	0.0000e+000	-6.1921e+001	0.0000e+000
	21	0.0000e+000	0.0000e+000	-4.3761e+001	0.0000e+000
27	21	0.0000e+000	0.0000e+000	1.3667e+001	0.0000e+000
	22	0.0000e+000	0.0000e+000	-2.0013e+001	0.0000e+000
28	22	0.0000e+000	0.0000e+000	1.2592e-002	0.0000e+000
	23	0.0000e+000	0.0000e+000	-6.3584e+000	0.0000e+000
29	23	0.0000e+000	0.0000e+000	3.8991e+001	0.0000e+000
	24	0.0000e+000	0.0000e+000	6.2979e+001	0.0000e+000
30	24	0.0000e+000	0.0000e+000	-6.2979e+001	0.0000e+000
	25	0.0000e+000	0.0000e+000	-3.5050e+001	0.0000e+000

(iii) Reactions at Step # 1, Applied Load Ratio = 1.0000

Forces

Node	Rx	Ry	Rz
1	1.0622e+001	1.4581e+002	FREE
2	-7.9746e+000	1.4318e+002	FREE
3	8.1400e+000	1.6095e+002	FREE
4	-1.0788e+001	1.5005e+002	FREE

Moments

Node	Mx	My	Mz
1	FREE	FREE	-1.1847e+001
2	FREE	FREE	7.7581e+000
3	FREE	FREE	-8.4777e+000
4	FREE	FREE	1.1564e+001

#####################################
End of Results of Structural Analysis
#####################################

Figure 70: Final results of Example 1 from Mastan2 v 3.2 (continued)

6.2.1 Comparison Results:

The results obtained from Integrated Force Method Analysis Package and Mastan2 v 3.2 [17] are compared and all the diagrams and the numerical results are agree fully as shown in Figures (66, 67, 68, 69, 70).

6.3 Example for Dual Integrated Force Method (IFMD)

Example 2: This Example is from the book, Structural Analysis, Reference [20]. A rigid frame as shown in Figure71 is analyzed with Dual Integrated Force Method. The problem is solved for nodal displacements, member end forces in local coordinate, support reactions, axial force, shear force and bending moment diagrams.

$I = 1\ m^4$ $\qquad$ $A = 1.5\times10^{-4}\ m^2$ $\qquad$ $E = 2\times10^{8}\ kN/m$

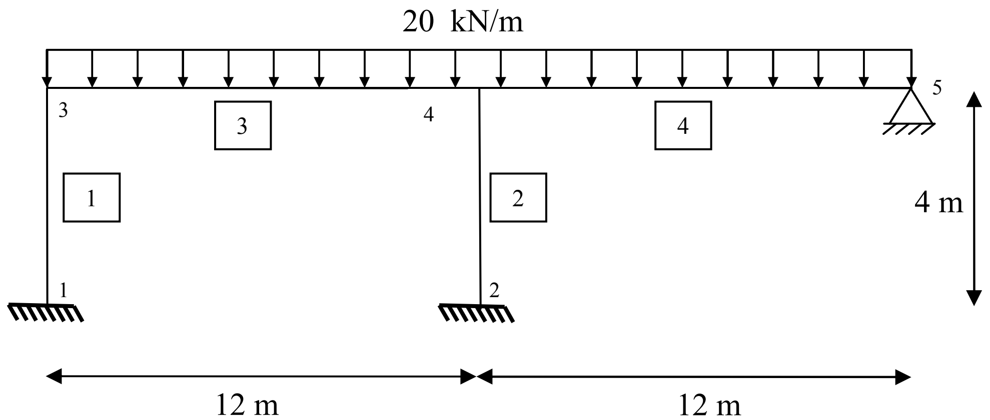


Figure 71: Example 2

To solve this problem first the Dual Integrated Force Method Analysis Package has been run and also the result from Mastan2 v3.2 is presented to compare the results.

The analysis procedure consists of the following phases:

a) Input Phase
    1. General Input
    2. Geometry Input
    3. Properties and Materials Input
    4. Load Data input

b) Calculation Phase

1. Generating Equilibrium Equations and showing Matrix Plot of this matrix
2. Generating Unconnected Flexibility Matrix and showing Matrix Plot of this matrix
3. Inverting Unconnected Flexibility Matrix
4. Generating Global Stiffness Matrix and Showing Matrix Plot of this matrix
5. Forming Joint Load Vector
6. Forming the Fixed End Forces Vector
7. Combining Joint Load Vector with Fixed End Forces Vector
8. Solving Displacements

c) Reporting Results Phase

1. Displaying Nodal Displacements
2. Displaying Member End Forces in Local Coordinate
3. Displaying Support Reactions
4. Showing Diagrams of Axial Force, Shear Force and Bending Moment Diagrams.

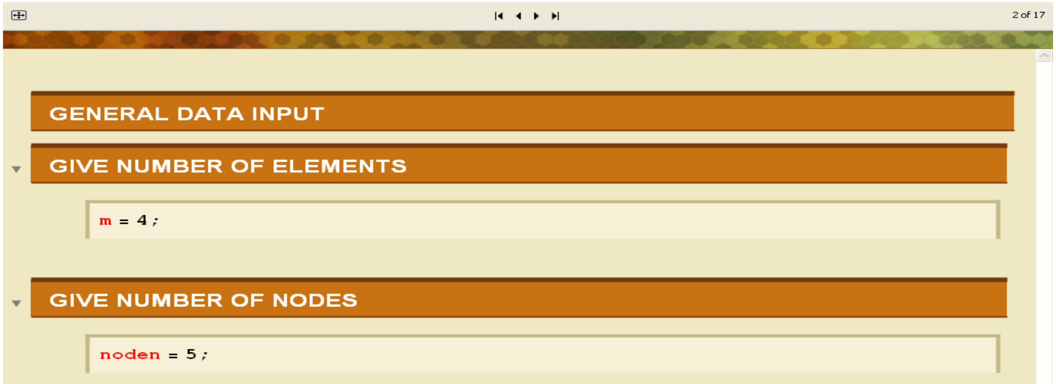


Figure 72: Input Phase, Step 1, General Input

Figure 73: Input Phase, Step 2, Geometry Input

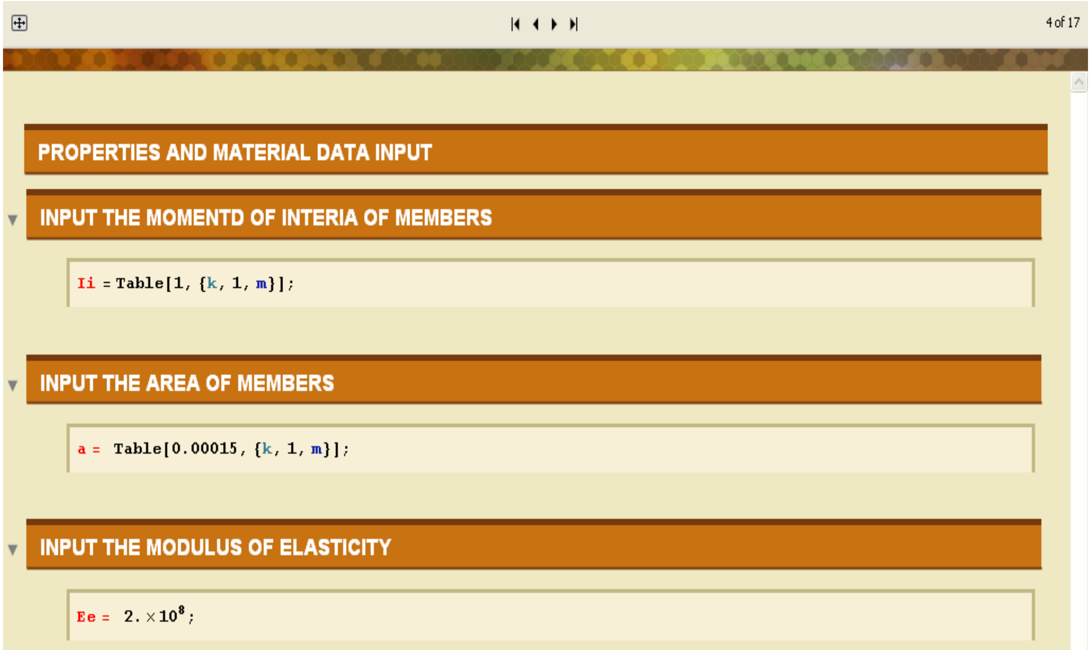


Figure 74: Input Phase, Step 3, Properties and Materials Input

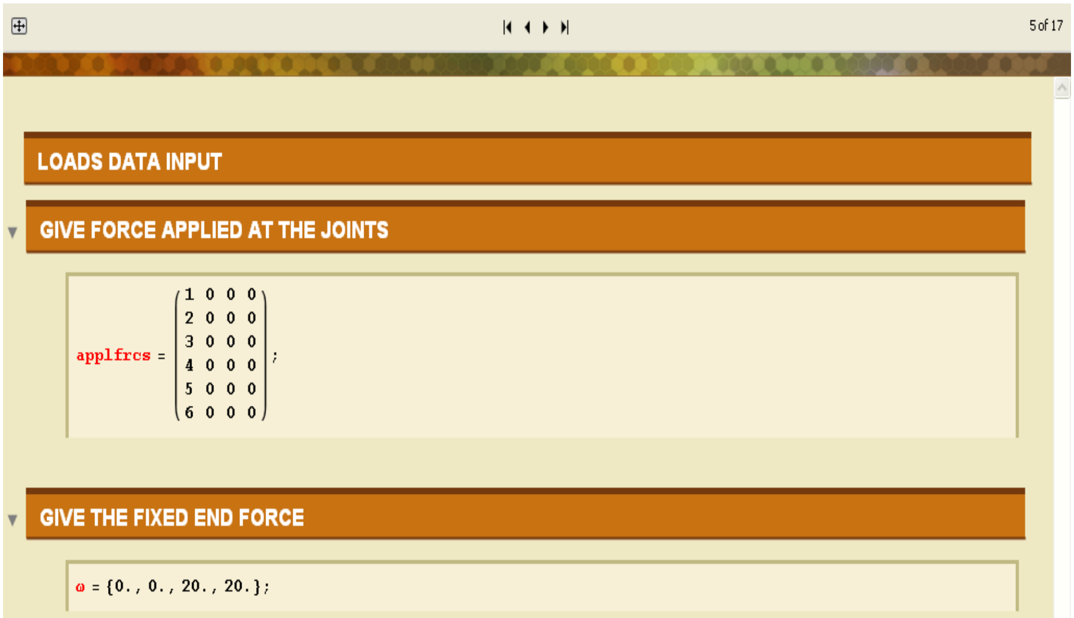


Figure 75: Input Phase, Step 4, Load Data Input

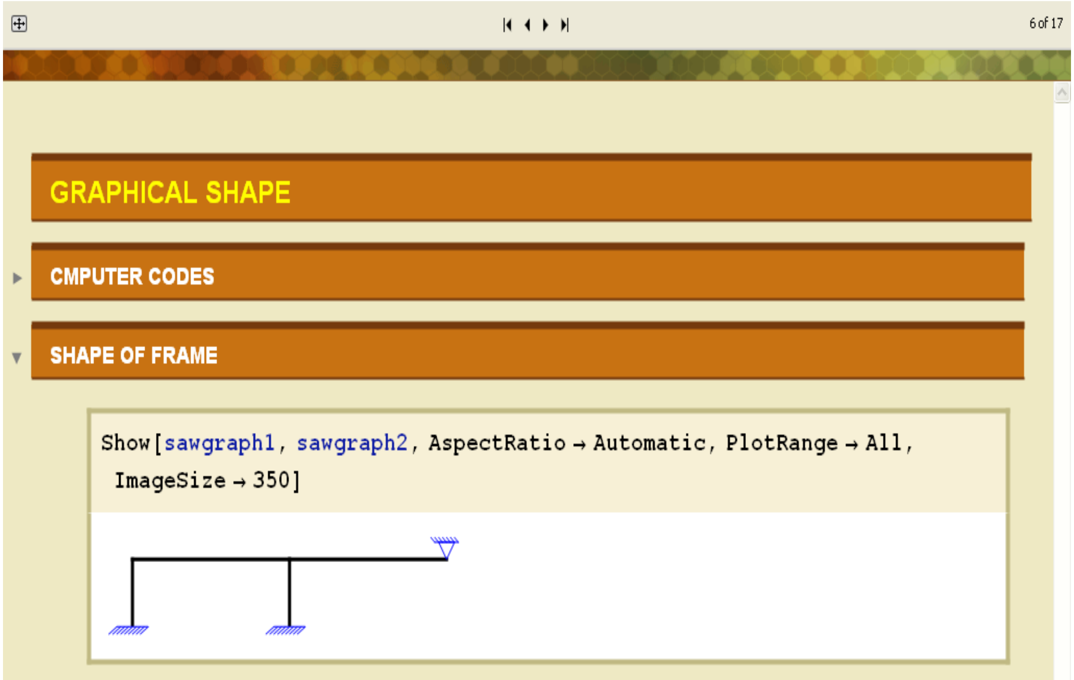


Figure 76: Shape of Frame

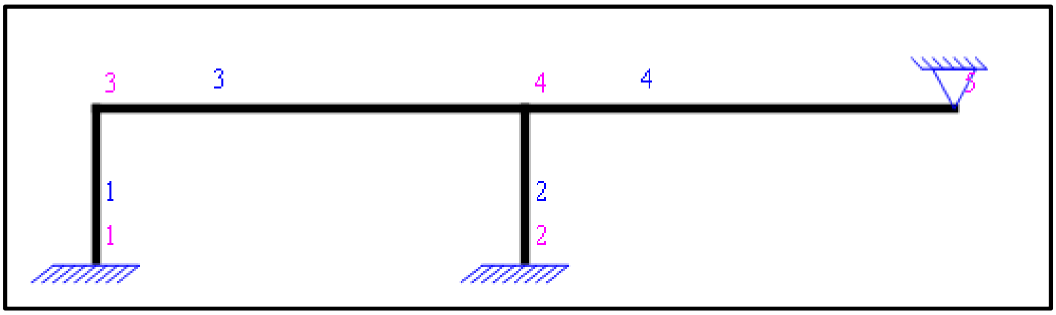


Figure 77: Element and Node Numbering

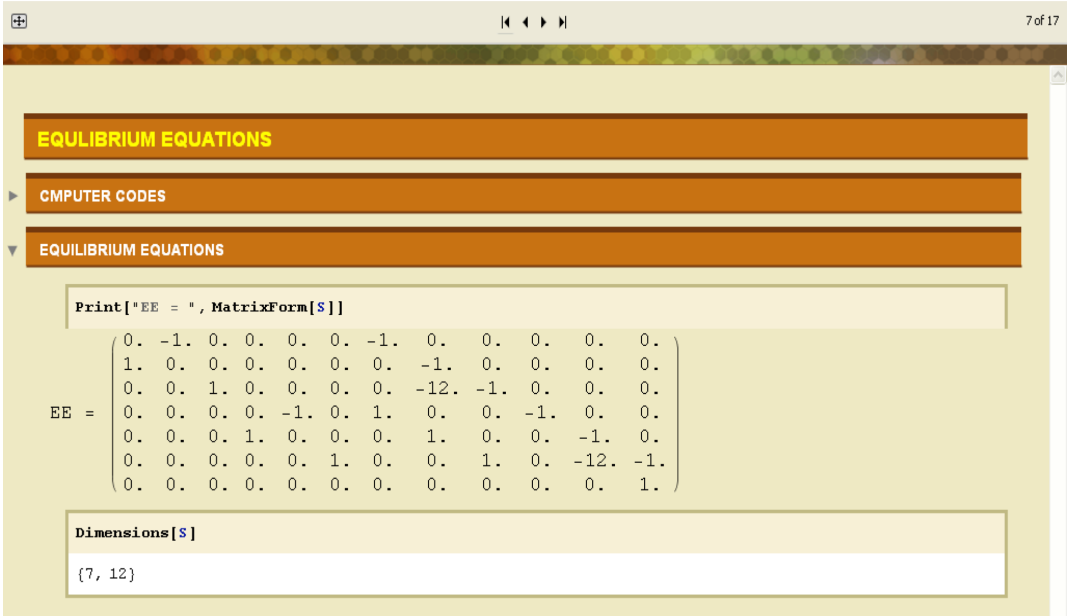


Figure 78: Calculation Phase, Step1, Generate Equilibrium Equations

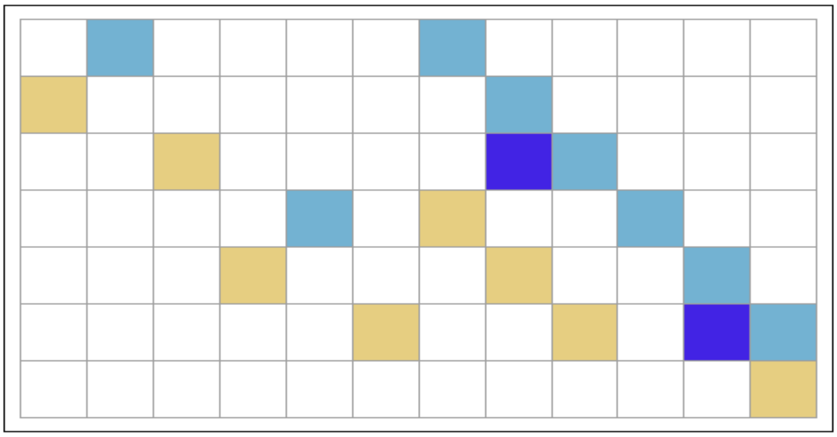


Figure 79: Calculation Phase, Step1, Matrix Plot of Equilibrium Equations [B] $_{15\times10}$

Figure 80: Calculation Phase, Step2, Generate Unconnected Flexibility Matrix

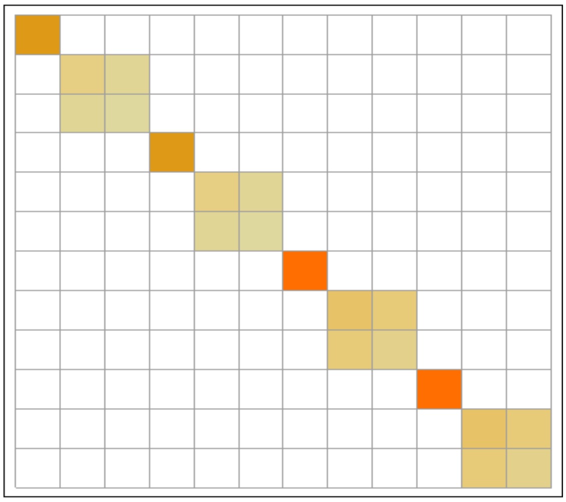


Figure 81: Calculation Phase, Step2, Matrix Plot of Unconnected Flexibility Matrix $[G]_{15\times15}$

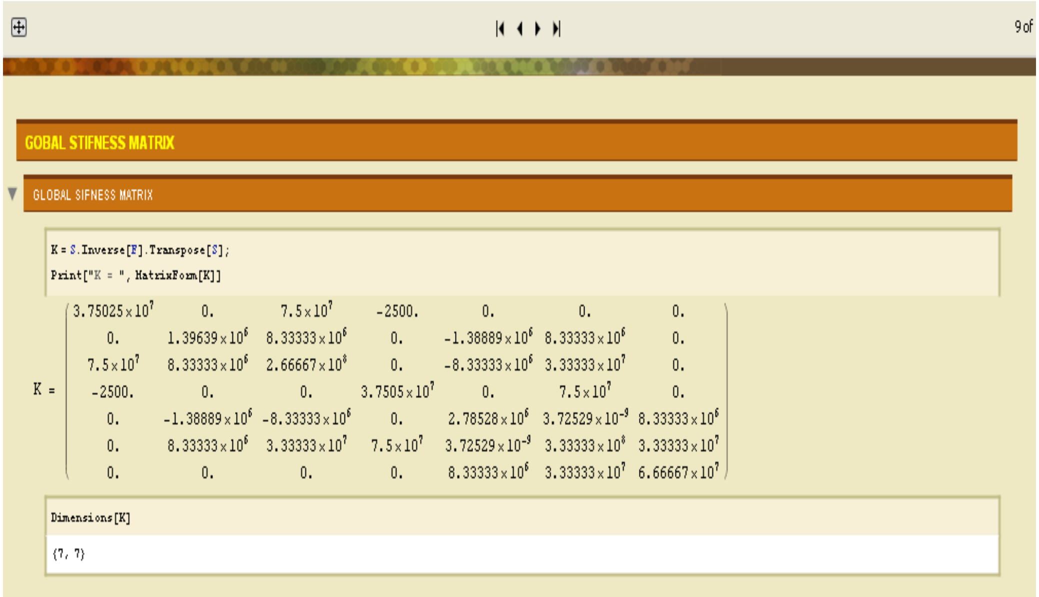


Figure 82: Calculation Phase, Step3, 4, Invert Unconnected Flexibility Matrix and Generate Global Stiffness Matrix

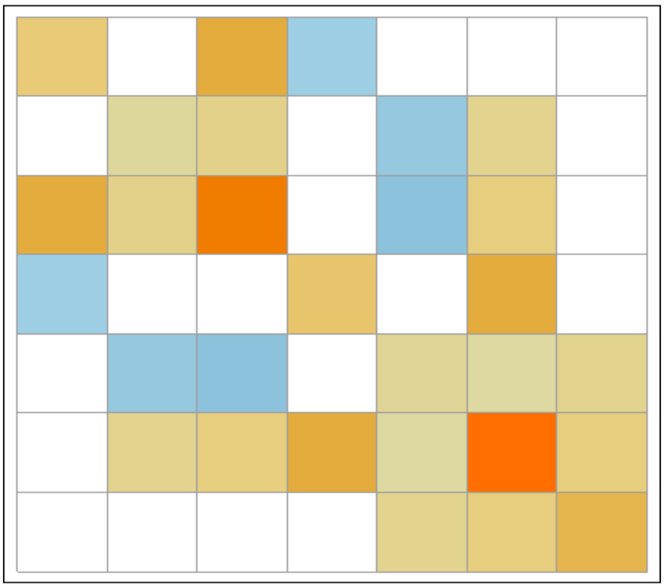


Figure 83: Calculation Phase, Step4, Matrix Plot of Global Stiffness Matrix [K] $_{10\times10}$

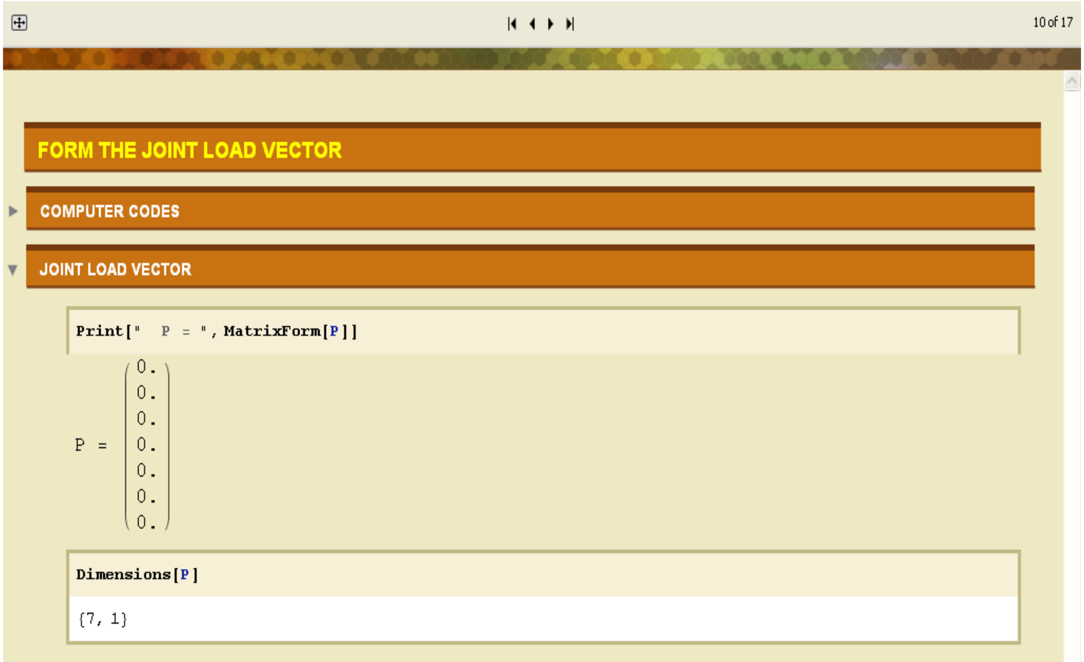


Figure 84: Calculation Phase, Step5, Form Joint Load Vector

Figure 85: Calculation Phase, Step6, Form Fixed End Forces

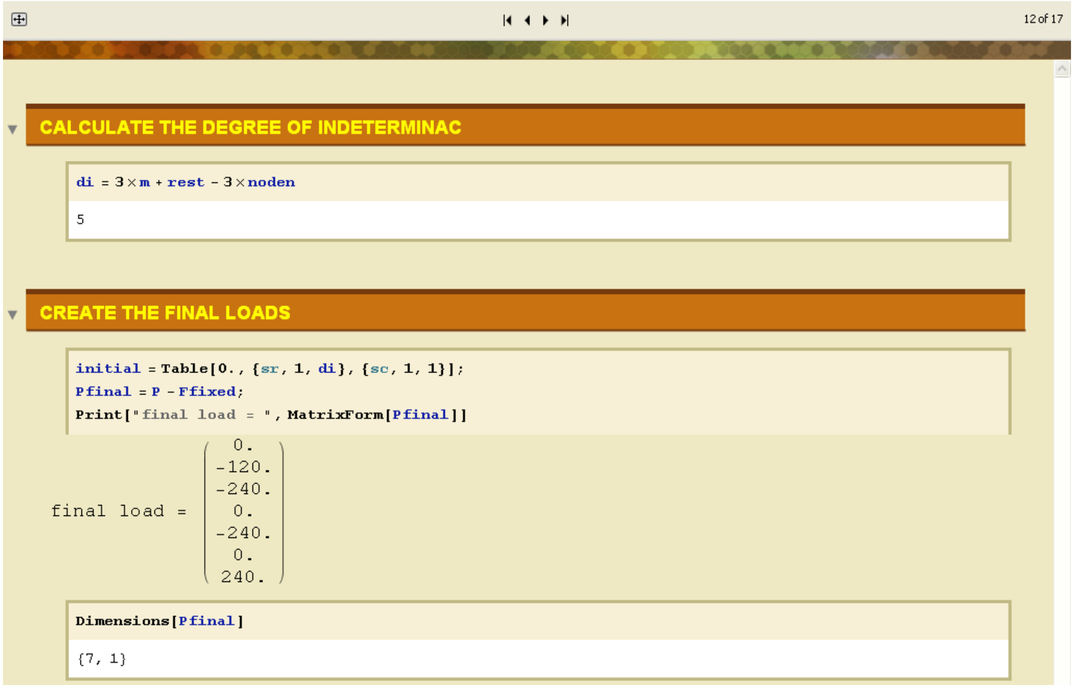


Figure 86: Calculation Phase, Step 7,Combined Joint Load Vector with Fixed End Forces

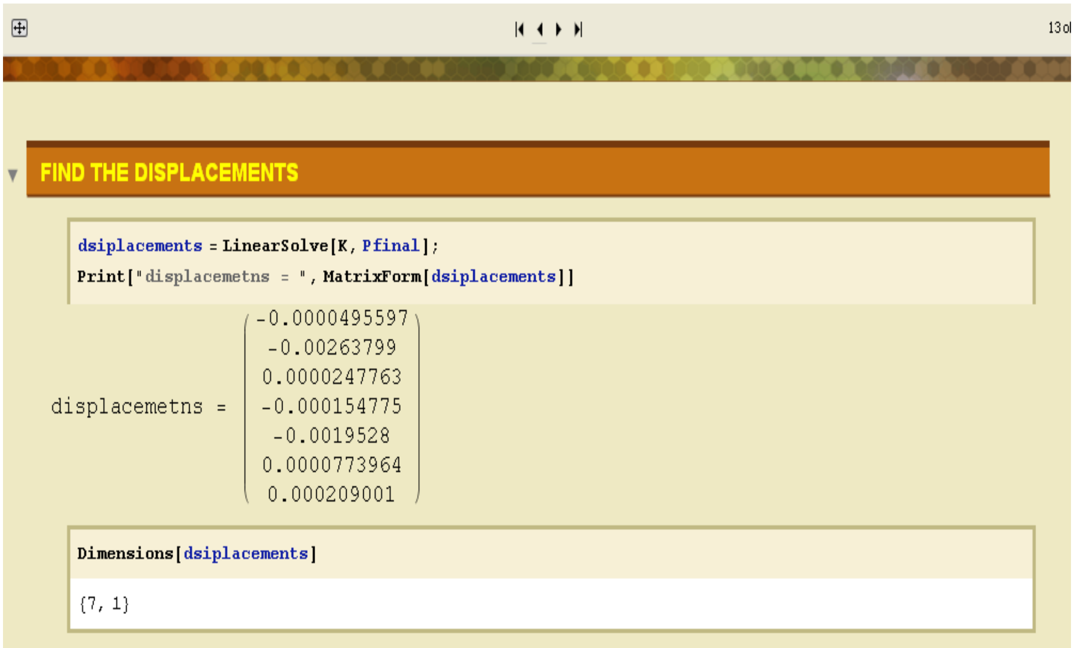


Figure 87: Calculation Phase, Step8, Solve for Displacements

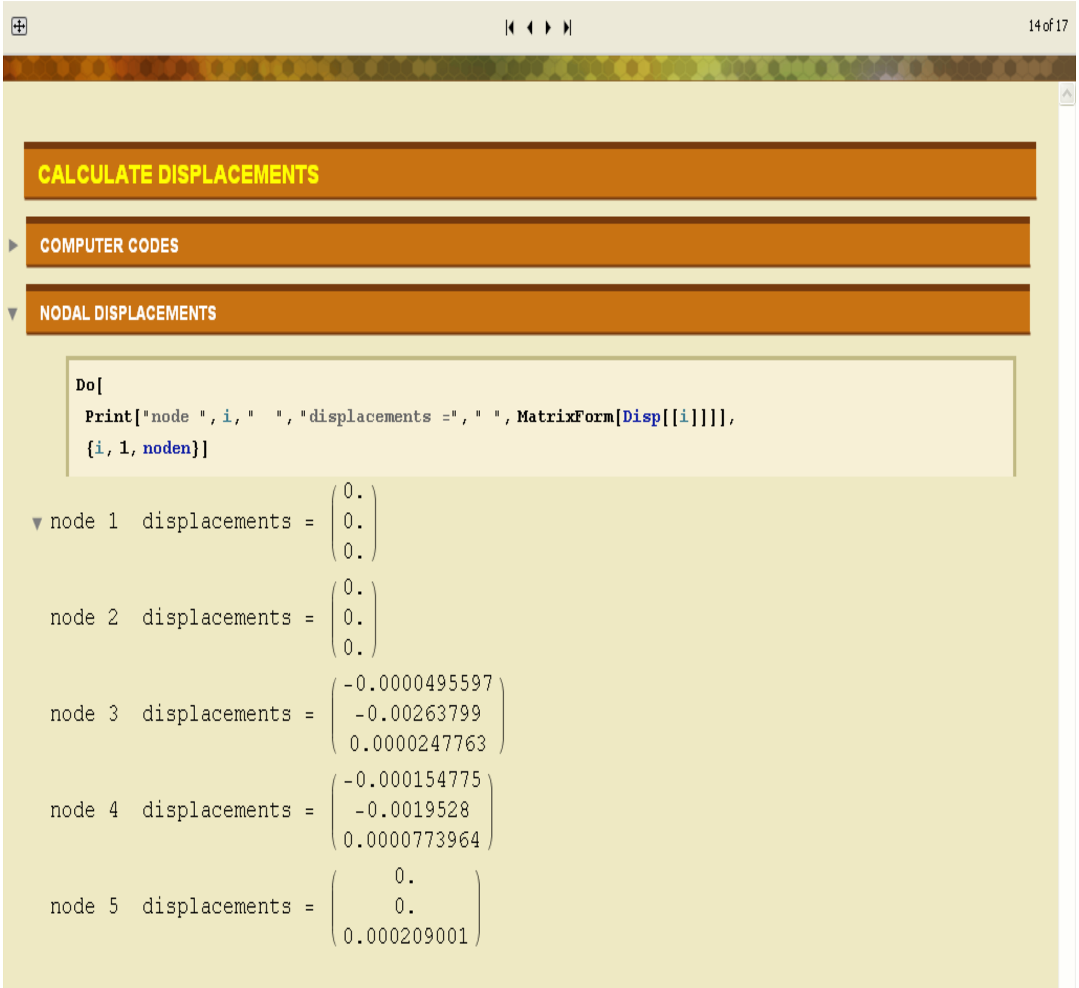


Figure 88: Reporting Results Phase, step1, Nodal Displacements of each node

member 1 endforces $\begin{pmatrix} 19.7849 \\ -0.263039 \\ -1239.34 \\ -19.7849 \\ 0.263039 \\ 1238.29 \end{pmatrix}$

member 2 endforces $\begin{pmatrix} 14.646 \\ 0.649978 \\ -3868.52 \\ -14.646 \\ -0.649978 \\ 3871.12 \end{pmatrix}$

member 3 endforces $\begin{pmatrix} 0.263039 \\ 19.7849 \\ -1238.29 \\ -0.263039 \\ 220.215 \\ 35.7101 \end{pmatrix}$

member 4 endforces $\begin{pmatrix} -0.386939 \\ -205.569 \\ -3906.83 \\ 0.386939 \\ 445.569 \\ 0 \end{pmatrix}$

Figure 89: Reporting Results Phase, step2, Member End Forces in Local Coordinate

```
reaction x=  0.263039
reaction y=  19.7849
moment z=  -1239.34
member  1 react x  1
member  1 react y  1
member  1 react z  1
reaction x=  -0.649978
reaction y=  14.646
moment z=  -3868.52
member  2 react x  2
member  2 react y  2
member  2 react z  2
reaction x=  0.386939
reaction y=  445.569
member  4 react x  5
member  4 react y  5
```

Figure 90: Reporting Results Phase, step3, Support Reactions

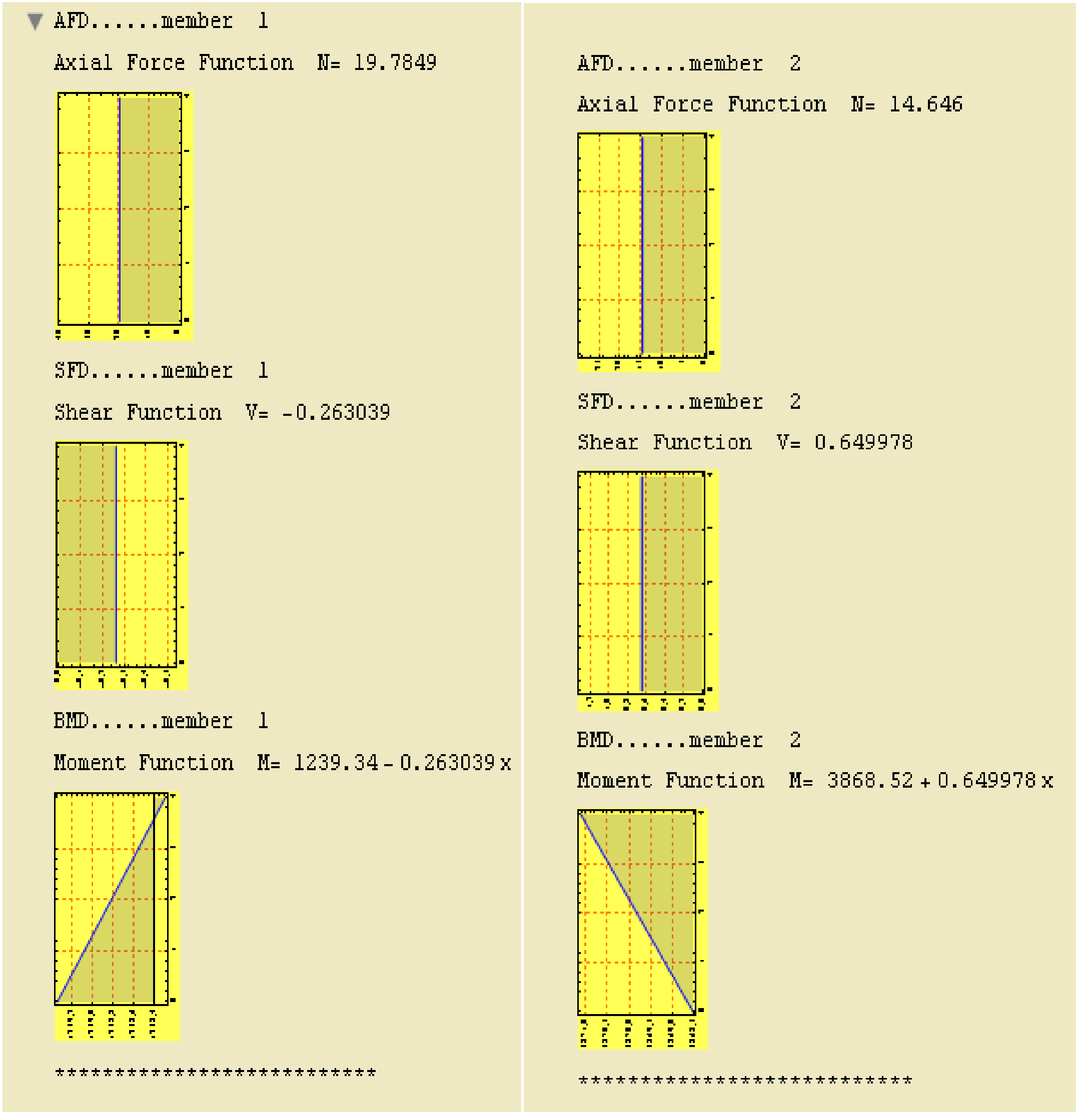


Figure 91: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member1, 2

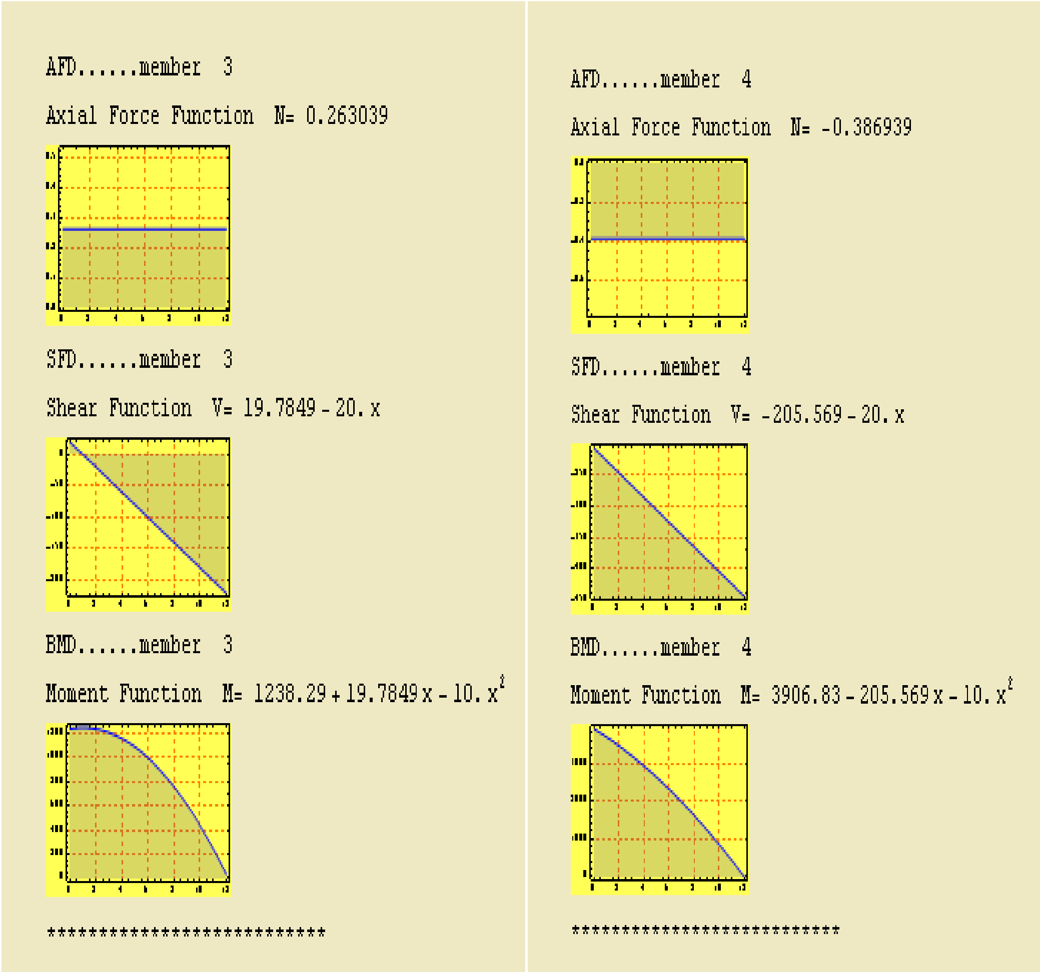


Figure 92: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member3, 4

6.3.1 Comparison Results:

The results obtained from Dual Integrated Force Method Package Analysis and Mastan2 v 3.2 [17] are compared and all the diagrams and numerical results are agree fully as shown in Figures (93 and 94)

```
###############################
Results of Structural Analysis
###############################

General Information:
      Structure Analyzed as:  Planar Frame
      Analysis Type:  First-Order Elastic

Analytical Results:

(i)   Displacements at Step # 1, Applied Load Ratio = 1.0000

  Deflections
     Node         X-disp           Y-disp           Z-disp
       1      0.0000e+000      0.0000e+000      0.0000e+000
       2      0.0000e+000      0.0000e+000      0.0000e+000
       3     -4.9560e-005     -2.6380e-003      0.0000e+000
       4     -1.5478e-004     -1.9528e-003      0.0000e+000
       5      0.0000e+000      0.0000e+000      0.0000e+000

  Rotations (radians)
     Node         X-rot            Y-rot            Z-rot
       1      0.0000e+000      0.0000e+000      0.0000e+000
       2      0.0000e+000      0.0000e+000      0.0000e+000
       3      0.0000e+000      0.0000e+000      2.4776e-005
       4      0.0000e+000      0.0000e+000      7.7396e-005
       5      0.0000e+000      0.0000e+000      2.0900e-004
```

Figure 93: Final results of Example 2 from Mastan2 v 3.2 (Displacements)

```
(ii)  Element Results at Step # 1, Applied Load Ratio = 1.0000

  Internal End Forces (Note: Refers to local coordinates)
    Element  Node        Fx             Fy             Fz
      1       1     1.9785e+001   -2.6304e-001    0.0000e+000
              3    -1.9785e+001    2.6304e-001    0.0000e+000
      2       2     1.4646e+001    6.4998e-001    0.0000e+000
              4    -1.4646e+001   -6.4998e-001    0.0000e+000
      3       3     2.6304e-001    1.9785e+001    0.0000e+000
              4    -2.6304e-001    2.2022e+002    0.0000e+000
      4       4    -3.8694e-001   -2.0557e+002    0.0000e+000
              5     3.8694e-001    4.4557e+002    0.0000e+000

  Internal End Moments (Note: Refers to local coordinates)
    Element  Node        Mx             My             Mz              B
      1       1     0.0000e+000    0.0000e+000   -1.2393e+003    0.0000e+000
              3     0.0000e+000    0.0000e+000    1.2383e+003    0.0000e+000
      2       2     0.0000e+000    0.0000e+000   -3.8685e+003    0.0000e+000
              4     0.0000e+000    0.0000e+000    3.8711e+003    0.0000e+000
      3       3     0.0000e+000    0.0000e+000   -1.2383e+003    0.0000e+000
              4     0.0000e+000    0.0000e+000    3.5710e+001    0.0000e+000
      4       4     0.0000e+000    0.0000e+000   -3.9068e+003    0.0000e+000
              5     0.0000e+000    0.0000e+000   -1.8190e-012    0.0000e+000

(iii) Reactions at Step # 1, Applied Load Ratio = 1.0000

  Forces
     Node         Rx              Ry             Rz
       1     2.6304e-001     1.9785e+001        FREE
       2    -6.4998e-001     1.4646e+001        FREE
       5     3.8694e-001     4.4557e+002        FREE

  Moments
     Node         Mx              My             Mz
       1         FREE            FREE       -1.2393e+003
       2         FREE            FREE       -3.8685e+003

######################################
End of Results of Structural Analysis
######################################
```

Figure 94: Final results of Example 2 from Mastan2 v 3.2 (Element Results and Reactions)

6.4 Example for Integrated Force Method via Singular Value Decompositions (SVD)

Example 3: A rigid frame is subjected for this example has 7 elements and 7 nodes. Nodes 4 and 6 have subjected to 50 kN and 75 kN axial point load respectively. Node 5 is subjected to 6 kN/m bending moment and element 7 has subjected to 12 kN/m fixed end force. This example is analyzed by Integrated Force Method via Singular Value Decomposition. This problem is solved for nodal displacements, member end forces in local coordinate, support reactions, axial force, shear force and bending moment diagrams.

$I = 4\times10^{-4}\ \ m^4$

$A = 2\times10^{-3}\ \ m^2$

$E = 2\times10^{8}\ \ kN/m$

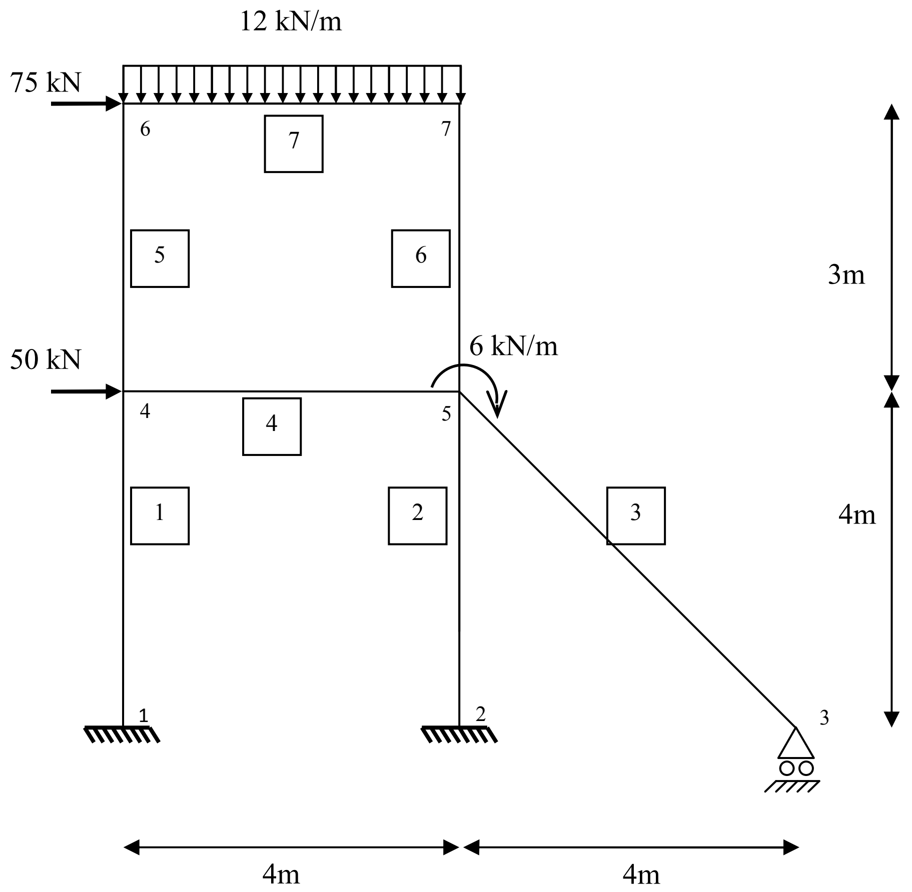


Figure 95: Example 3

The analysis procedure consists of the following phases:

a) Input Phase

   1. General Input
   2. Geometry Input
   3. Properties and Materials Input
   4. Load Data input

b) Calculation Phase

1. Generating Equilibrium Equations and showing Matrix Plot of this matrix.
2. Generating Unconnected Flexibility Matrix and showing Matrix Plot of this matrix.
3. Creating Singular Value Decomposition.

4. Obtaining Compatibility Matrix form Singular Value Decomposition.
5. Computing the Compatibility Conditions and showing Matrix Plot of this matrix.
6. Coupling the Compatibility Conditions with the Equilibrium Equations and showing Matrix Plot of this matrix.
7. Forming Joint Load Vector.
8. Forming the Fixed End Forces Vector.
9. Combining Joint Load Vector with Fixed End Forces Vector.
10. Solving Independent Forces.

c) Reporting Results Phase

1. Displaying Nodal Displacements
2. Displaying Member End Forces in Local Coordinate
3. Displaying Support Reactions
4. Showing Axial Force, Shear Force and Bending Moment Diagrams.

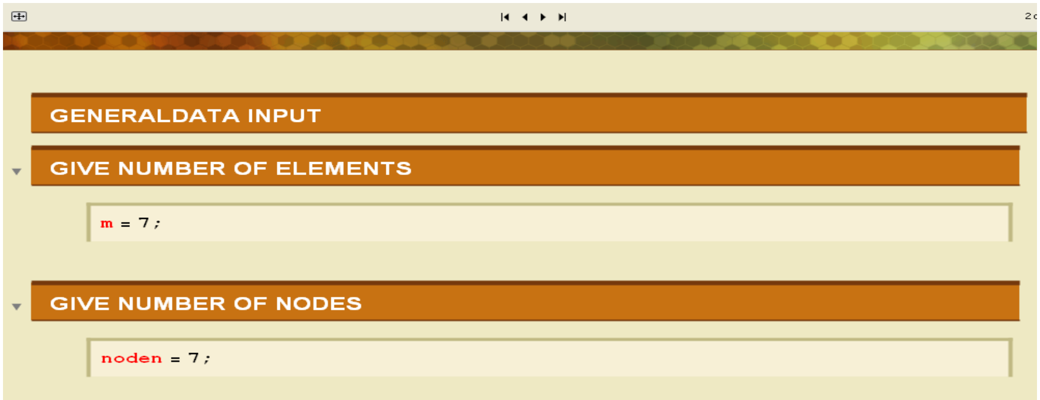


Figure 96: Input Phase, Step 1, General Data Input

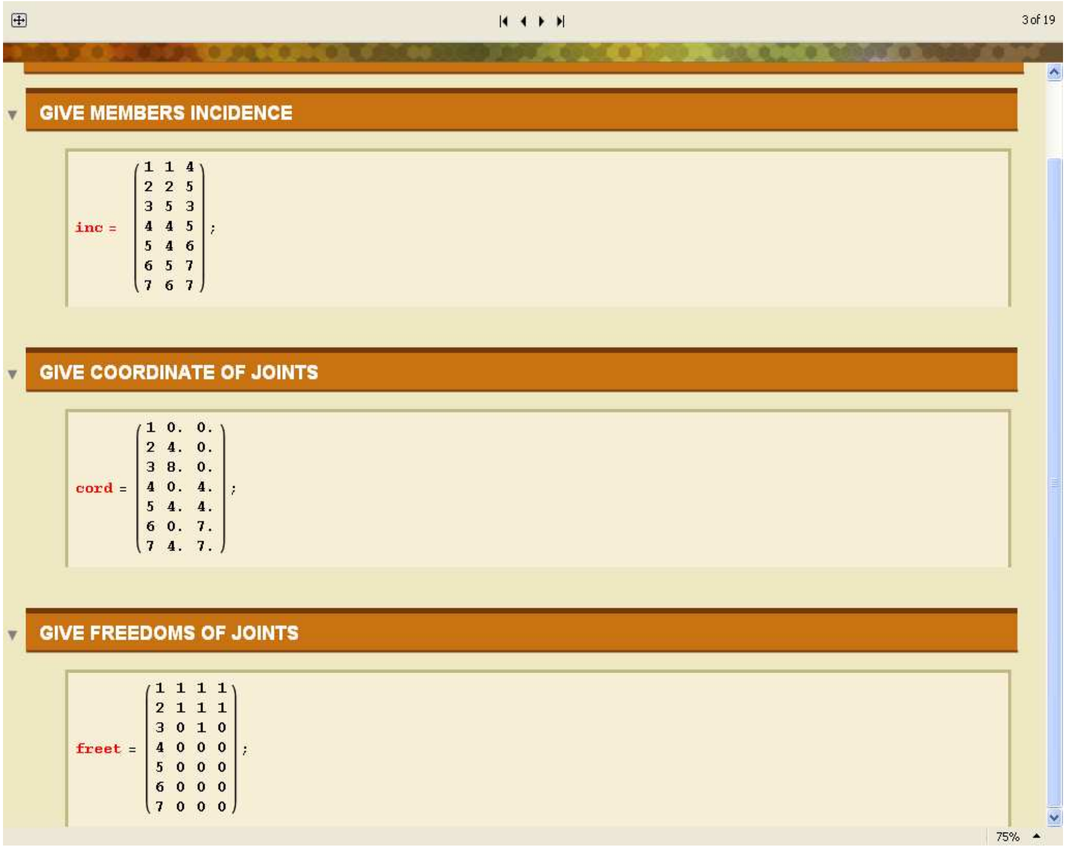


Figure 97: Input Phase, Step 2, Geometry Data Input

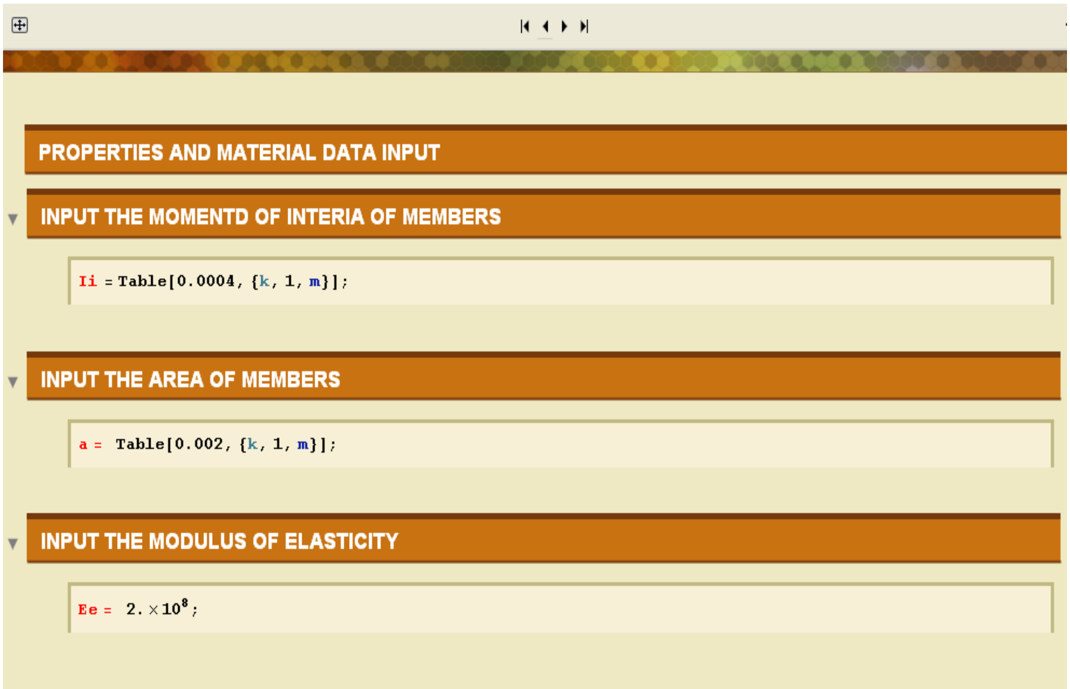


Figure 98: Input Phase, Step 3, Properties and Materials Input

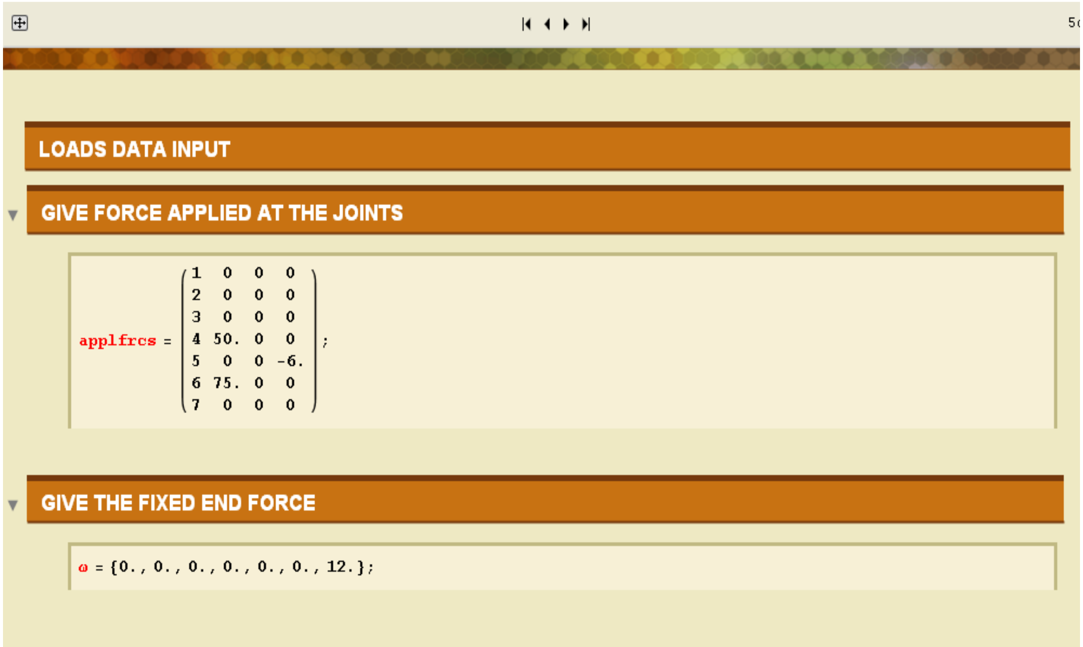


Figure 99: Input Phase, Step 4, Load Data Input

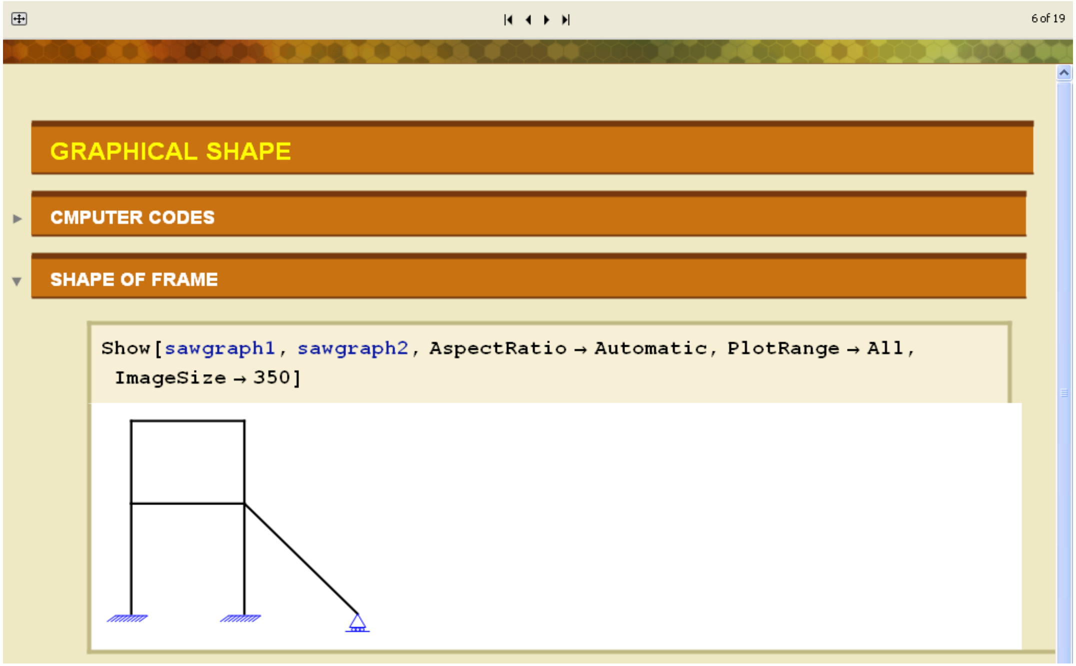


Figure 100: Shape of Frame

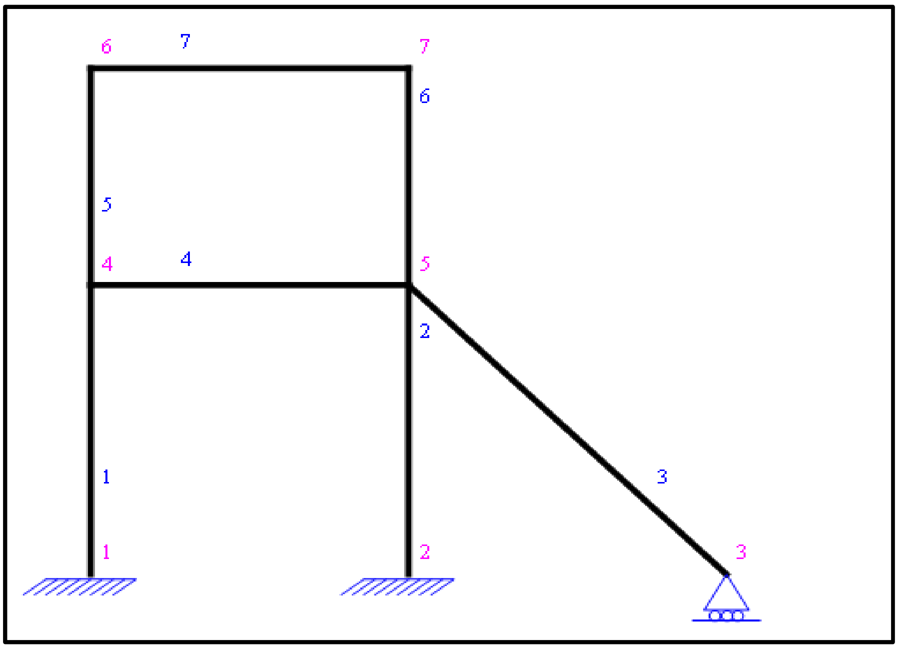


Figure 101: Element and Node Numbering

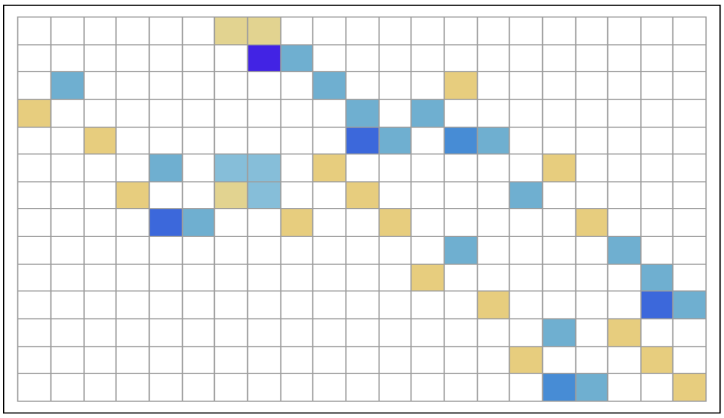


Figure 102: Calculation Phase, Step1, Matrix Plot of Equilibrium Equations [B] $_{21\times14}$

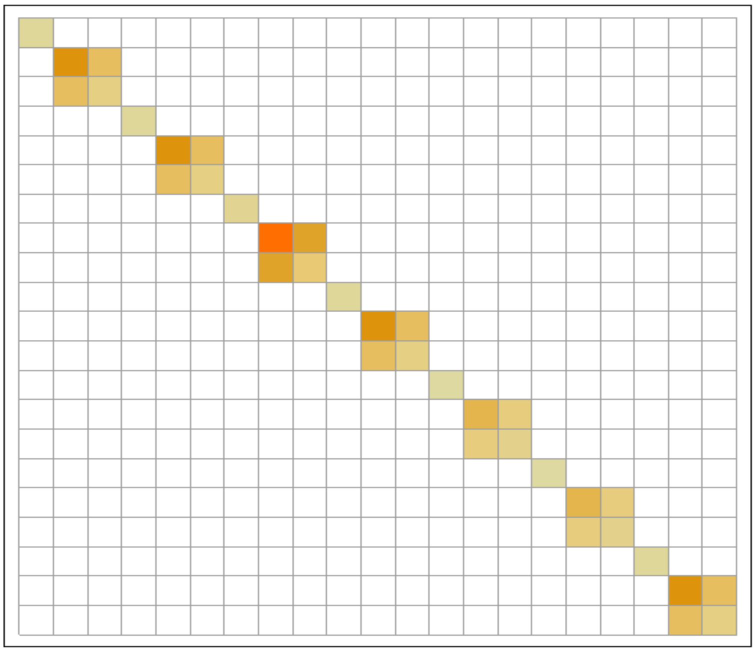


Figure 103: Calculation Phase, Step2, Matrix Plot of Unconnected Flexibility Matrix [G] $_{21\times21}$

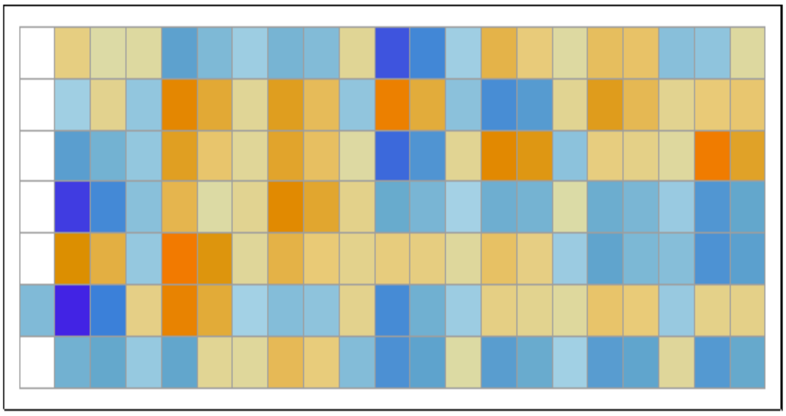


Figure 104: Calculation Phase, Step5, Matrix Plot of Compatibility Conditions [C] $_{21\times 7}$

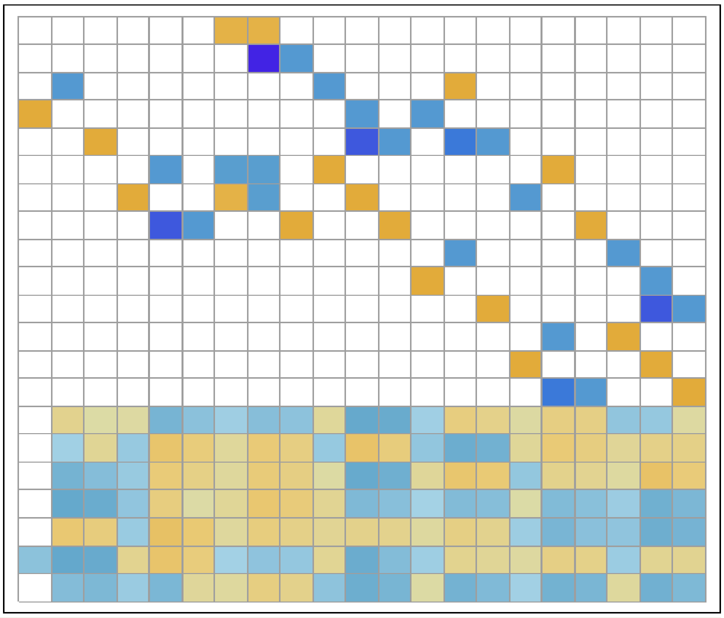


Figure 105: Calculation Phase, Step6, Matrix Plot of Coupled Equilibrium Equations with Compatibility Conditions [S] $_{21\times 21}$

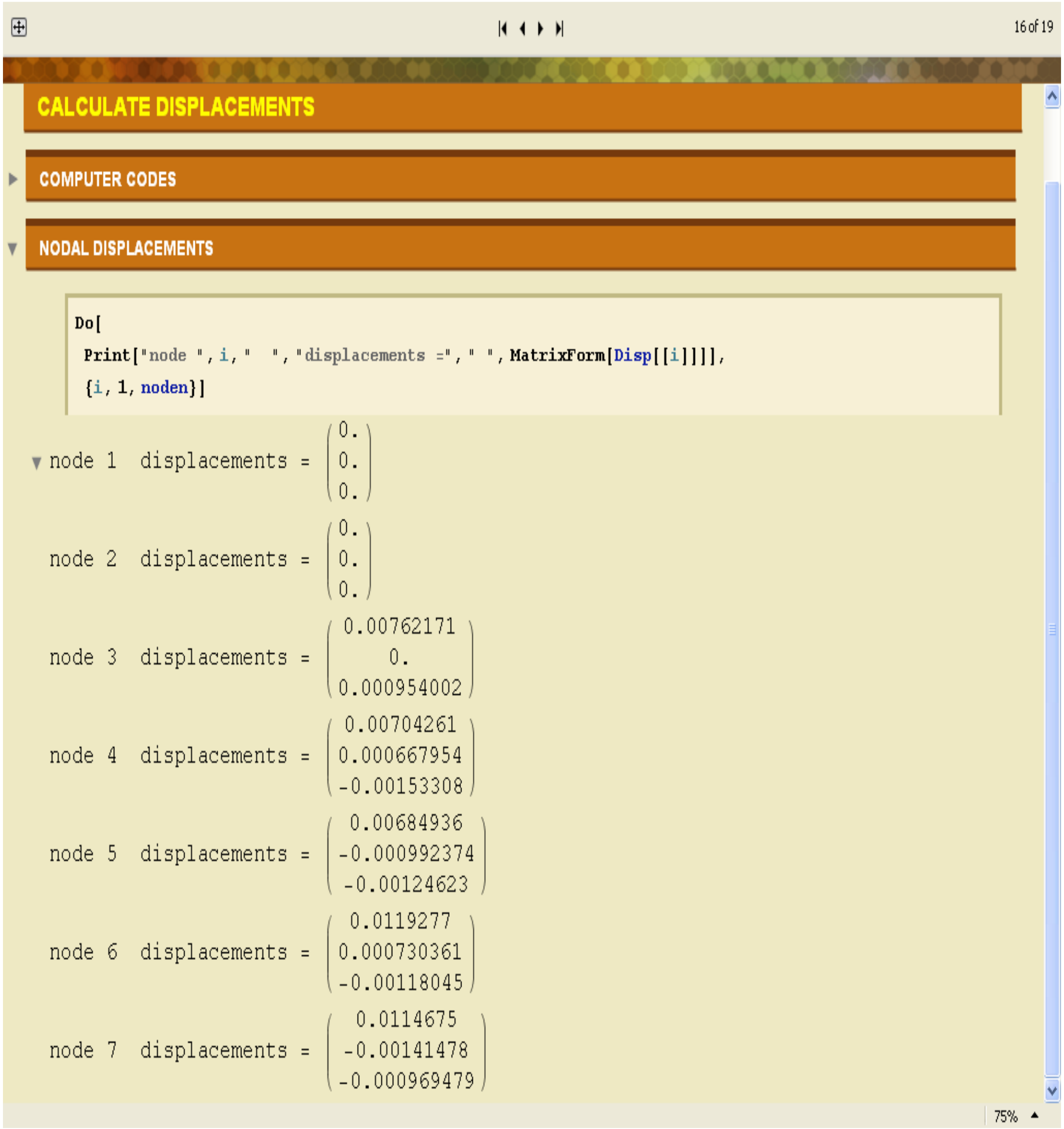


Figure 106: Reporting Results Phase, step1, Nodal Displacements of each node

member 1 endforces $\begin{pmatrix} -66.7954 \\ 59.6467 \\ 149.955 \\ 66.7954 \\ -59.6467 \\ 88.6317 \end{pmatrix}$

member 2 endforces $\begin{pmatrix} 99.2374 \\ 65.3533 \\ 155.631 \\ -99.2374 \\ -65.3533 \\ 105.782 \end{pmatrix}$

member 3 endforces $\begin{pmatrix} 11.0012 \\ -11.0012 \\ -62.232 \\ -11.0012 \\ 11.0012 \\ 0. \end{pmatrix}$

member 4 endforces $\begin{pmatrix} 19.3252 \\ -58.4745 \\ -122.686 \\ -19.3252 \\ 58.4745 \\ -111.212 \end{pmatrix}$

member 5 endforces $\begin{pmatrix} -8.32088 \\ 28.9719 \\ 34.0544 \\ 8.32088 \\ -28.9719 \\ 52.8612 \end{pmatrix}$

member 6 endforces $\begin{pmatrix} 56.3209 \\ 46.0281 \\ 61.6621 \\ -56.3209 \\ -46.0281 \\ 76.4223 \end{pmatrix}$

member 7 endforces $\begin{pmatrix} 46.0281 \\ -8.32088 \\ -52.8612 \\ -46.0281 \\ 56.3209 \\ -76.4223 \end{pmatrix}$

Figure 107: Reporting Results Phase, step2, Member End Forces in Local Coordinate

```
▼ reaction x=  -59.6467
  reaction y=  -66.7954
  moment z=  149.955
  member  1 react x  1
  member  1 react y  1
  member  1 react z  1
  reaction x=  -65.3533
  reaction y=  99.2374
  moment z=  155.631
  member  2 react x  2
  member  2 react y  2
  member  2 react z  2
  reaction y=  15.558
  member  3 react y  3
```

Figure 108: Reporting Results Phase, step3, Support Reactions

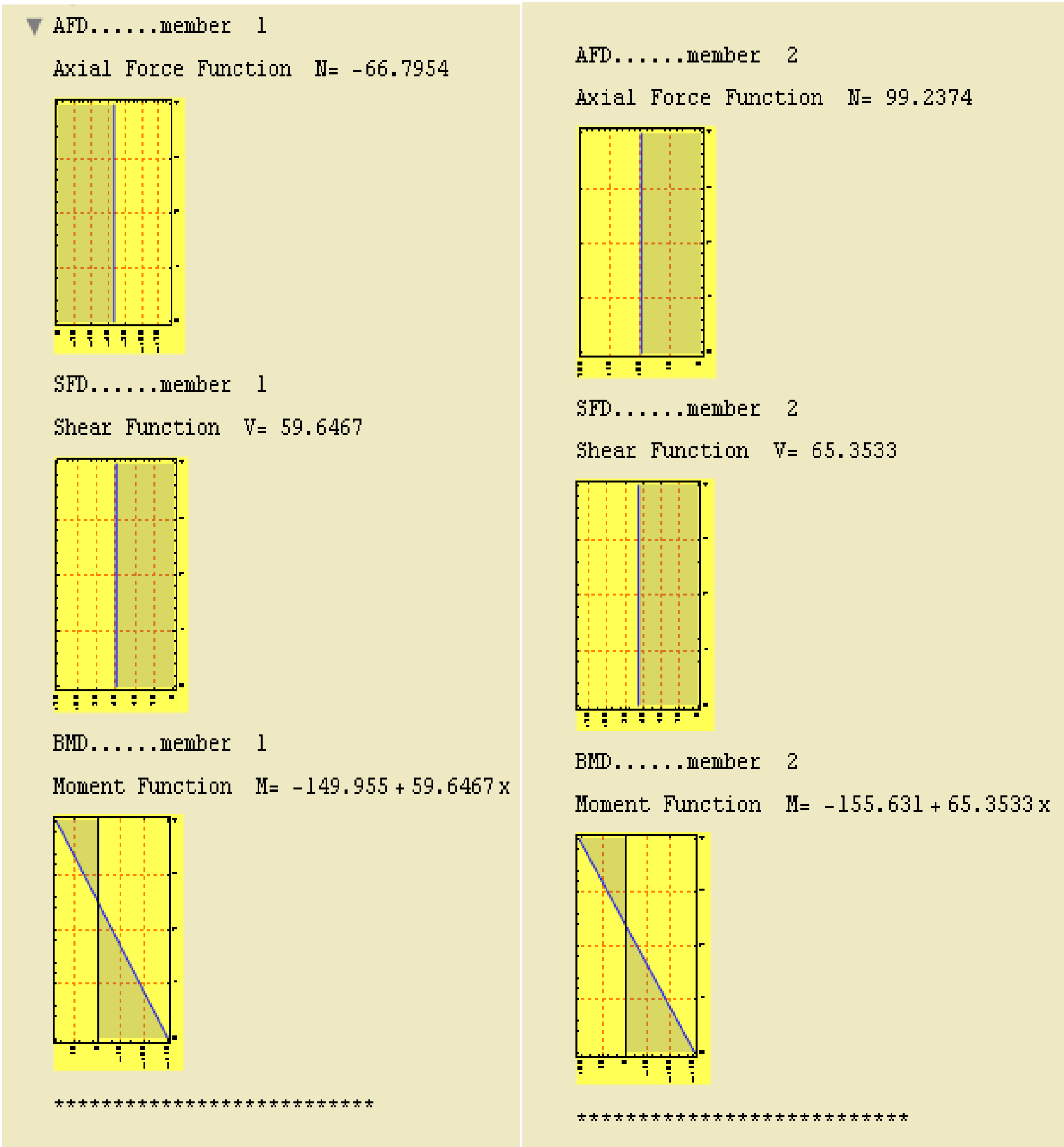


Figure 109: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 1, 2

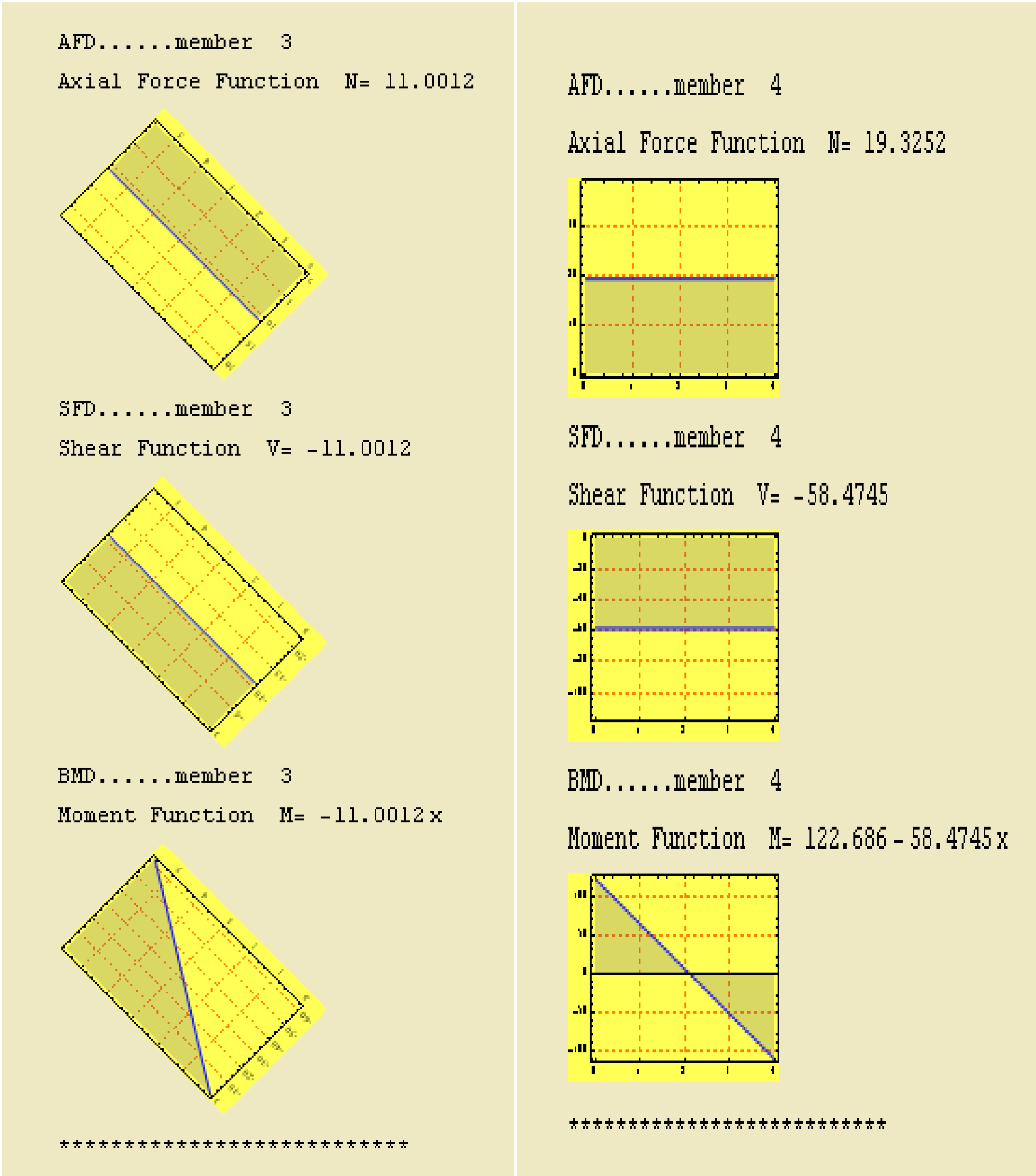


Figure 110: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 3, 4

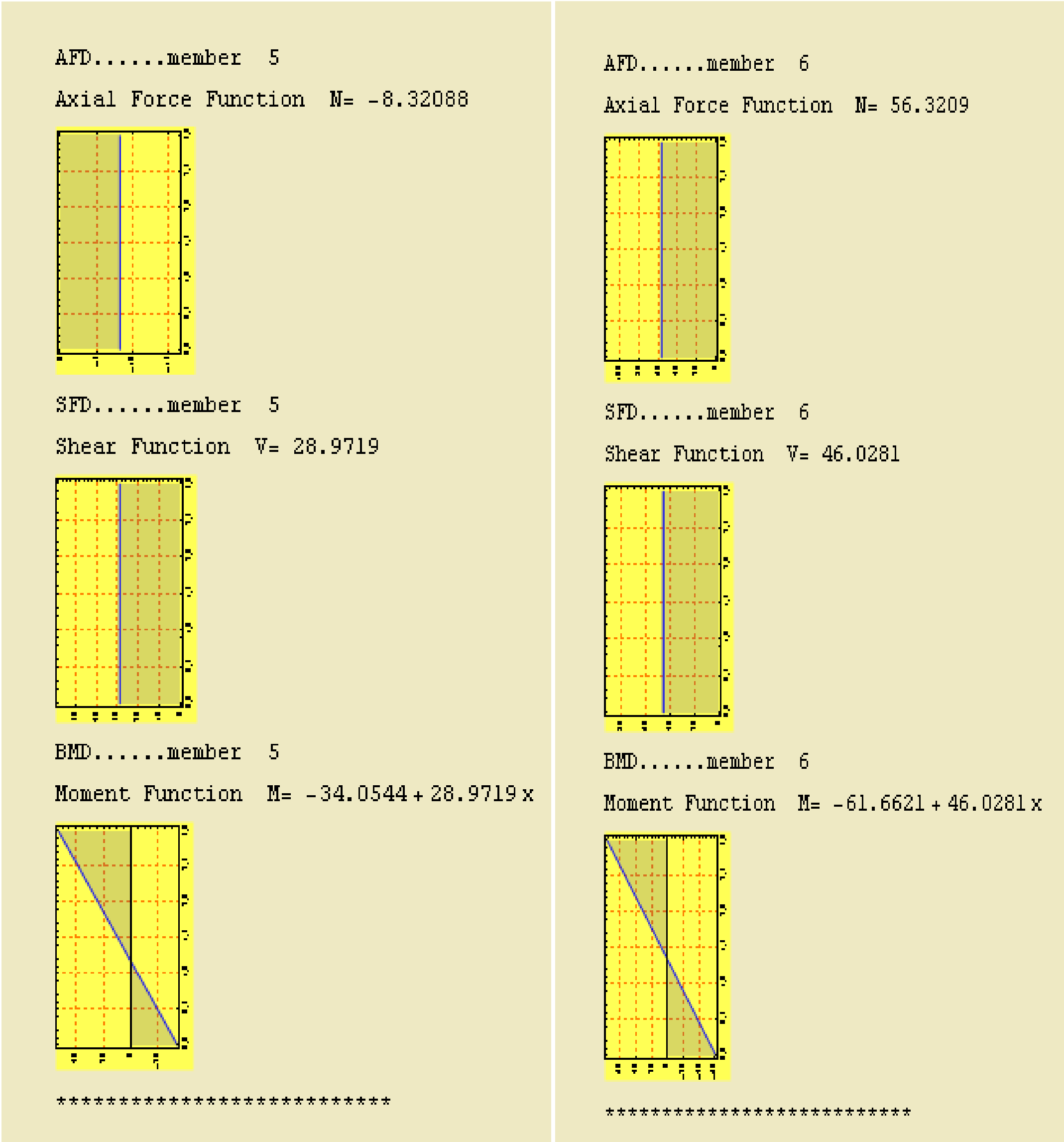
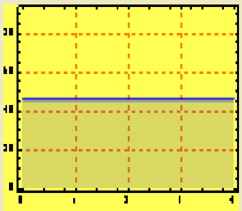


Figure 111: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 5, 6

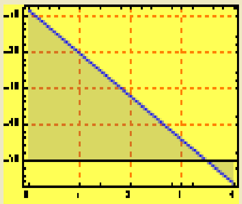
AFD......member 7

Axial Force Function N= 46.0281



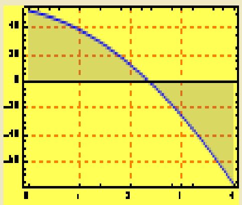
SFD......member 7

Shear Function V= -8.32088 - 12. x



BMD......member 7

Moment Function M= 52.8612 - 8.32088 x - 6. x^2



****

Figure 112: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 7

6.4.1 Comparison of Results

The results obtained from Integrated Force Method via Singular Value Decomposition Analysis Package and Mastan2 v 3.2 [17] are compared and all the diagrams agree fully and the numerical results agree up (Figures 114, 115, 116).

```
###############################
Results of Structural Analysis
###############################

General Information:
      Structure Analyzed as:  Planar Frame
      Analysis Type:  First-Order Elastic

Analytical Results:

(i)   Displacements at Step # 1, Applied Load Ratio = 1.0000

  Deflections
     Node        X-disp          Y-disp          Z-disp
       1      0.0000e+000     0.0000e+000     0.0000e+000
       2      0.0000e+000     0.0000e+000     0.0000e+000
       3      7.6217e-003     0.0000e+000     0.0000e+000
       4      7.0426e-003     6.6795e-004     0.0000e+000
       5      6.8494e-003    -9.9237e-004     0.0000e+000
       6      1.1928e-002     7.3036e-004     0.0000e+000
       7      1.1467e-002    -1.4148e-003     0.0000e+000

  Rotations (radians)
     Node        X-rot           Y-rot           Z-rot
       1      0.0000e+000     0.0000e+000     0.0000e+000
       2      0.0000e+000     0.0000e+000     0.0000e+000
       3      0.0000e+000     0.0000e+000     9.5400e-004
       4      0.0000e+000     0.0000e+000    -1.5331e-003
       5      0.0000e+000     0.0000e+000    -1.2462e-003
       6      0.0000e+000     0.0000e+000    -1.1805e-003
       7      0.0000e+000     0.0000e+000    -9.6948e-004
```

Figure 113: Final results of Example 3 from Mastan2 v 3.2 (Displacements)

(ii) Element Results at Step # 1, Applied Load Ratio = 1.0000

Internal End Forces (Note: Refers to local coordinates)

Element	Node	Fx	Fy	Fz
1	1	-6.6795e+001	5.9647e+001	0.0000e+000
	4	6.6795e+001	-5.9647e+001	0.0000e+000
2	2	9.9237e+001	6.5353e+001	0.0000e+000
	5	-9.9237e+001	-6.5353e+001	0.0000e+000
3	5	1.1001e+001	-1.1001e+001	0.0000e+000
	3	-1.1001e+001	1.1001e+001	0.0000e+000
4	4	1.9325e+001	-5.8475e+001	0.0000e+000
	5	-1.9325e+001	5.8475e+001	0.0000e+000
5	4	-8.3209e+000	2.8972e+001	0.0000e+000
	6	8.3209e+000	-2.8972e+001	0.0000e+000
6	5	5.6321e+001	4.6028e+001	0.0000e+000
	7	-5.6321e+001	-4.6028e+001	0.0000e+000
7	6	4.6028e+001	-8.3209e+000	0.0000e+000
	7	-4.6028e+001	5.6321e+001	0.0000e+000

Internal End Moments (Note: Refers to local coordinates)

Element	Node	Mx	My	Mz	B
1	1	0.0000e+000	0.0000e+000	1.4995e+002	0.0000e+000
	4	0.0000e+000	0.0000e+000	8.8632e+001	0.0000e+000
2	2	0.0000e+000	0.0000e+000	1.5563e+002	0.0000e+000
	5	0.0000e+000	0.0000e+000	1.0578e+002	0.0000e+000
3	5	0.0000e+000	0.0000e+000	-6.2232e+001	0.0000e+000
	3	0.0000e+000	0.0000e+000	2.1316e-014	0.0000e+000
4	4	0.0000e+000	0.0000e+000	-1.2269e+002	0.0000e+000
	5	0.0000e+000	0.0000e+000	-1.1121e+002	0.0000e+000
5	4	0.0000e+000	0.0000e+000	3.4054e+001	0.0000e+000
	6	0.0000e+000	0.0000e+000	5.2861e+001	0.0000e+000
6	5	0.0000e+000	0.0000e+000	6.1662e+001	0.0000e+000
	7	0.0000e+000	0.0000e+000	7.6422e+001	0.0000e+000
7	6	0.0000e+000	0.0000e+000	-5.2861e+001	0.0000e+000
	7	0.0000e+000	0.0000e+000	-7.6422e+001	0.0000e+000

Figure 114: Final results of Example 3 from Mastan2 v 3.2 (Element Results)

(iii) Reactions at Step # 1, Applied Load Ratio = 1.0000

Forces

Node	Rx	Ry	Rz
1	-5.9647e+001	-6.6795e+001	FREE
2	-6.5353e+001	9.9237e+001	FREE
3	FREE	1.5558e+001	FREE

Moments

Node	Mx	My	Mz
1	FREE	FREE	1.4995e+002
2	FREE	FREE	1.5563e+002

######################################
End of Results of Structural Analysis
######################################

Figure 115: Final results of Example 3 from Mastan2 v 3.2 (Reactions)

6.5 Example for Integrated Force Method via Null Space (IFM)

Example 4: A frame is subjected for this example has taken form a master thesis of completed at EMU (reference [12]). This example has 3 elements and 4 nodes. Node 3 is subjected to 1 kN shear joint load and node 4 are subjected to 0.5 kN/m axial joint load. The example is analyzed by integrated force method via null space. The problem is solved for nodal displacements, member end forces in local coordinate, support reactions, the axial force diagram, shear force diagram and the bending moment diagram.

$I = 1\ m^4$ $\quad\quad$ $A = 1000\ m^2$ $\quad\quad$ $E = 1\ kN/m^2$

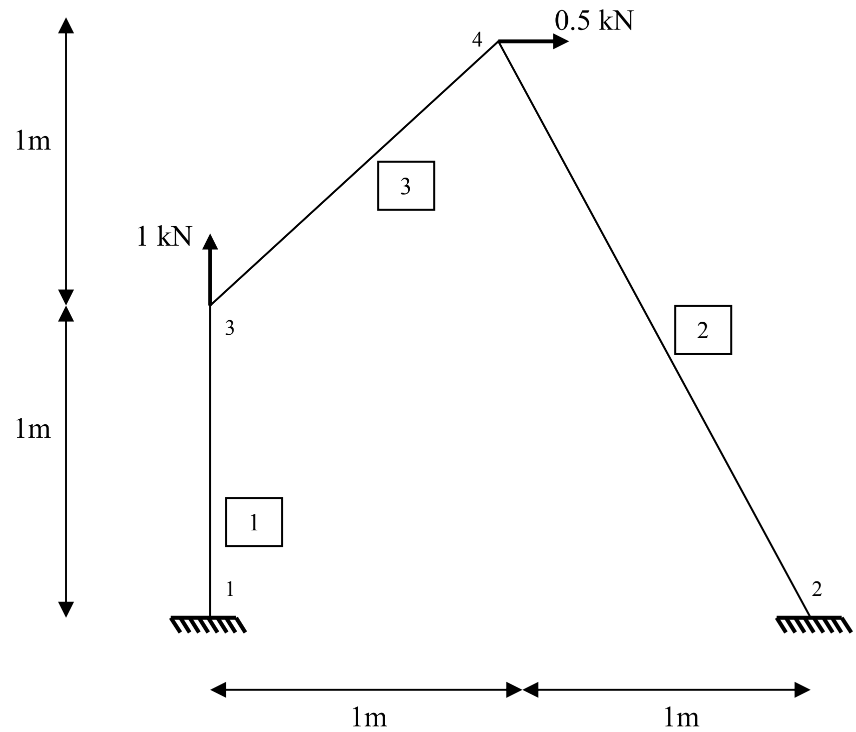


Figure 116: Example 4

To solve this problem first the Integrated Force Method Analysis Packages has been run and also the result from Mastan2 v3.2 is presented to compare the results.

The analysis procedure consists of the following phases:

a) Input Phase

1. General Input
2. Geometry Input
3. Properties and Materials Input
4. Load Data input

b) Calculation Phase

1. Generating Equilibrium Equations and showing Matrix Plot of this matrix
2. Generating Unconnected Flexibility Matrix and showing Matrix Plot of this Matrix
3. Obtaining Compatibility Matrix from Null Space.
4. Computing the Compatibility Conditions and showing Matrix Plot of this Matrix.
5. Coupling the Compatibility Conditions with the Equilibrium Equations and showing Matrix Plot of this Matrix
6. Forming Joint Load Vector
7. Forming the Fixed End Forces Vector
8. Combining Joint Load Vector with Fixed End Forces Vector
9. Solving Independent Forces

c) Reporting Results Phase

1. Displaying Nodal Displacements
2. Displaying Member End Forces in Local Coordinate
3. Displaying Support Reactions
4. Showing Diagrams of Axial Force, Shear Force and Bending Moment Diagrams.

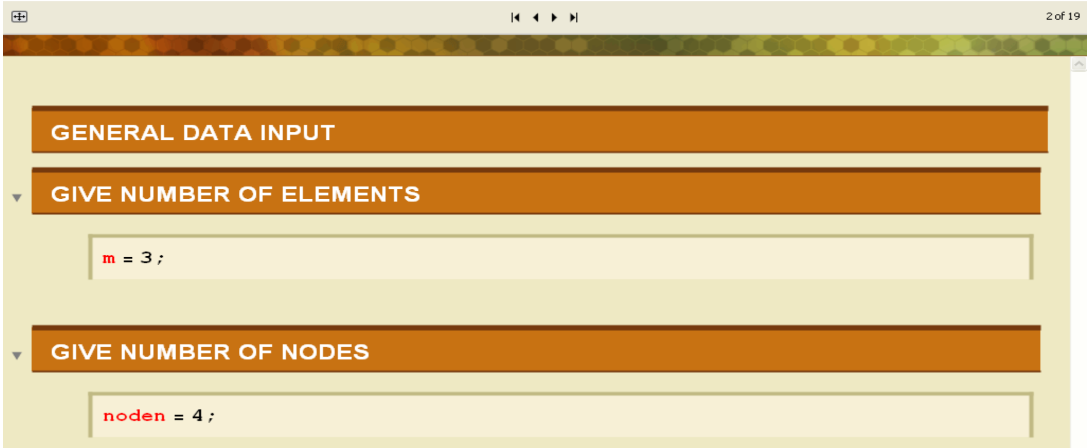


Figure 117: Input Phase, Step 1, General Input Phase

Figure 118: Input Phase, Step 2, Geometry Data Input

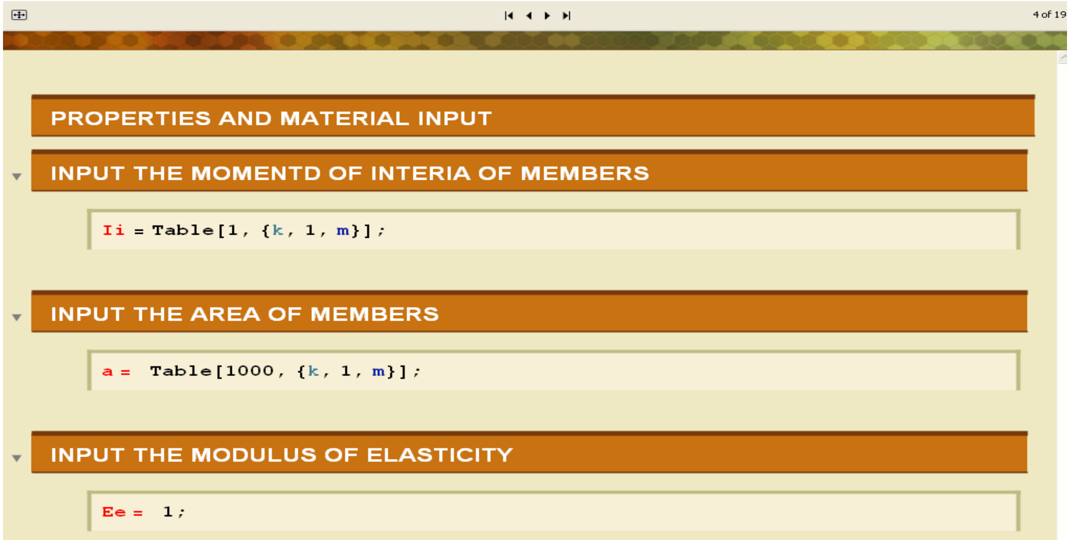


Figure 119: Input Phase, Step 3, Properties and Materials Input

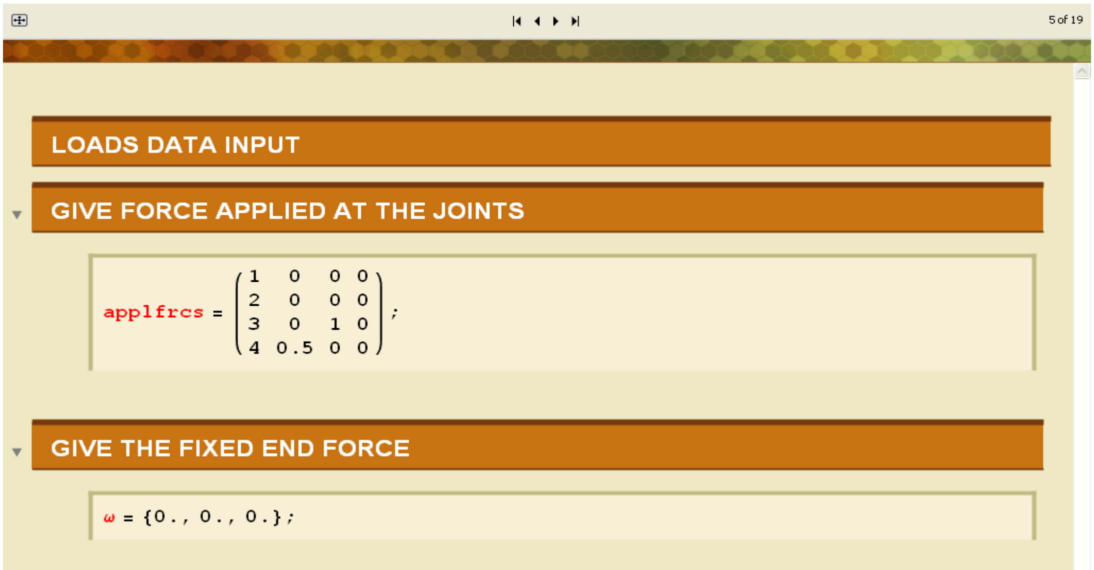


Figure 120: Input Phase, Step 4, Load Data Input

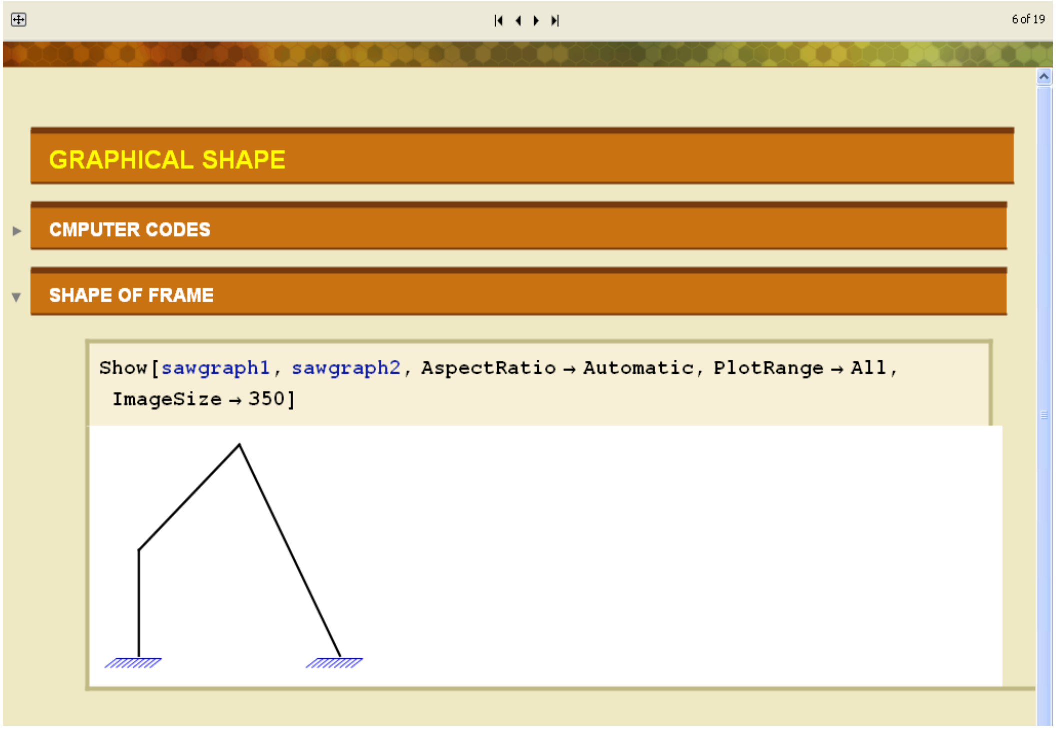


Figure 121: Shape of Frame

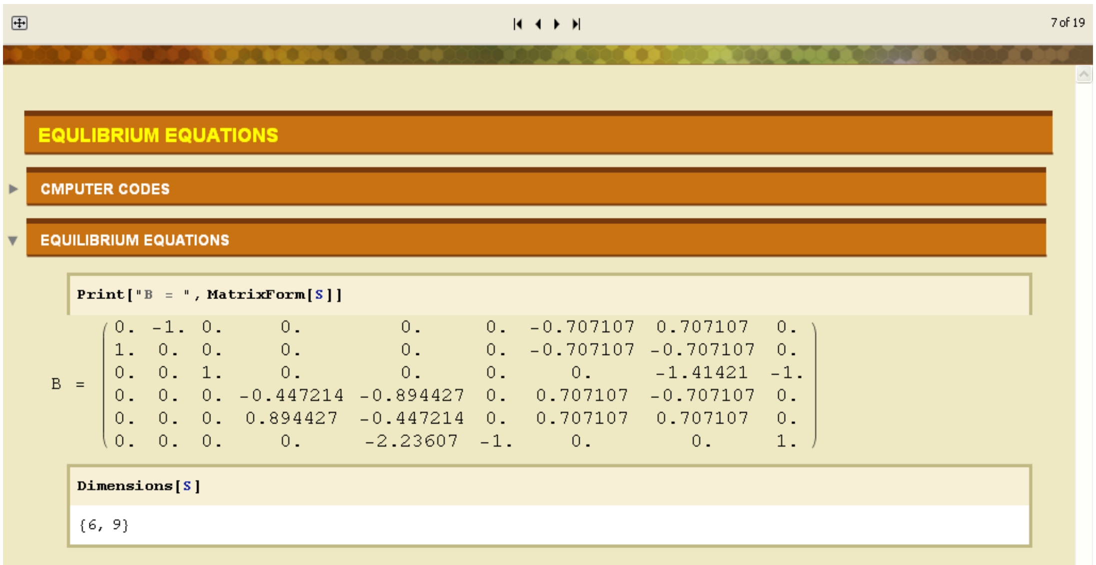


Figure 122: Calculation Phase, Step1, Equilibrium Equations

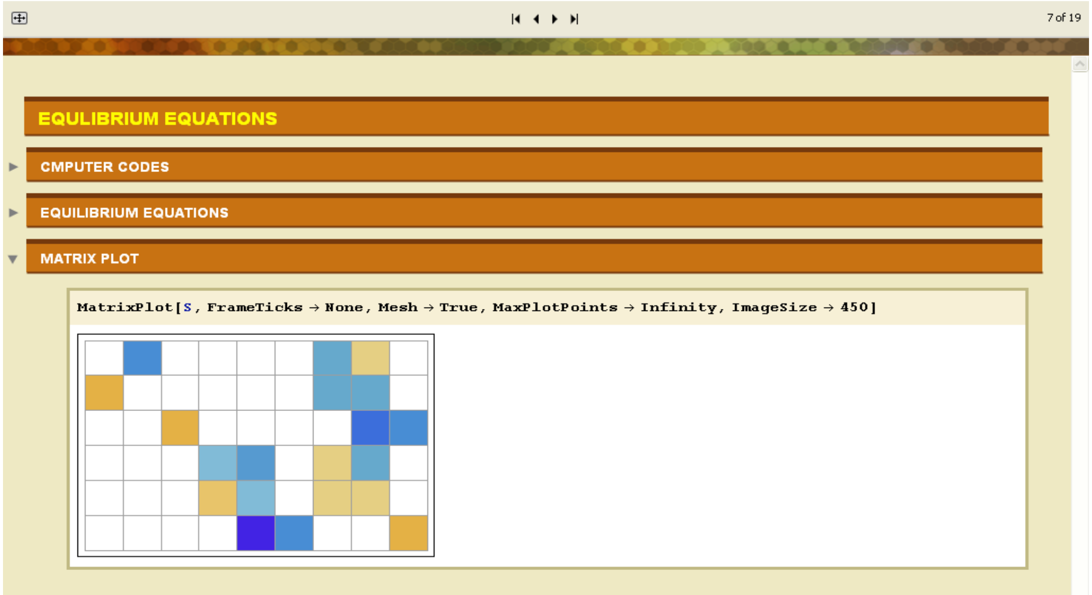


Figure 123: Calculation Phase, Step1, Matrix Plot of Equilibrium Equations [B] $_{6\times9}$

8 of 19

UNCONNECTED FLEXIBILTY MATRIX

COMPUTER CODES

FLEXIBILTY MATRIX

```
Print["G = ", MatrixForm[F]]
```

$$G = \begin{pmatrix} 0.001 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0.333333 & 0.5 & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0.5 & 1. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0.00223607 & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 3.72678 & 2.5 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 2.5 & 2.23607 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0.00141421 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.942809 & 1. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1. & 1.41421 \end{pmatrix}$$

```
Dimensions[F]
{9, 9}
```

Figure 124: Calculation Phase, Step2, Unconnected Flexibility Matrix

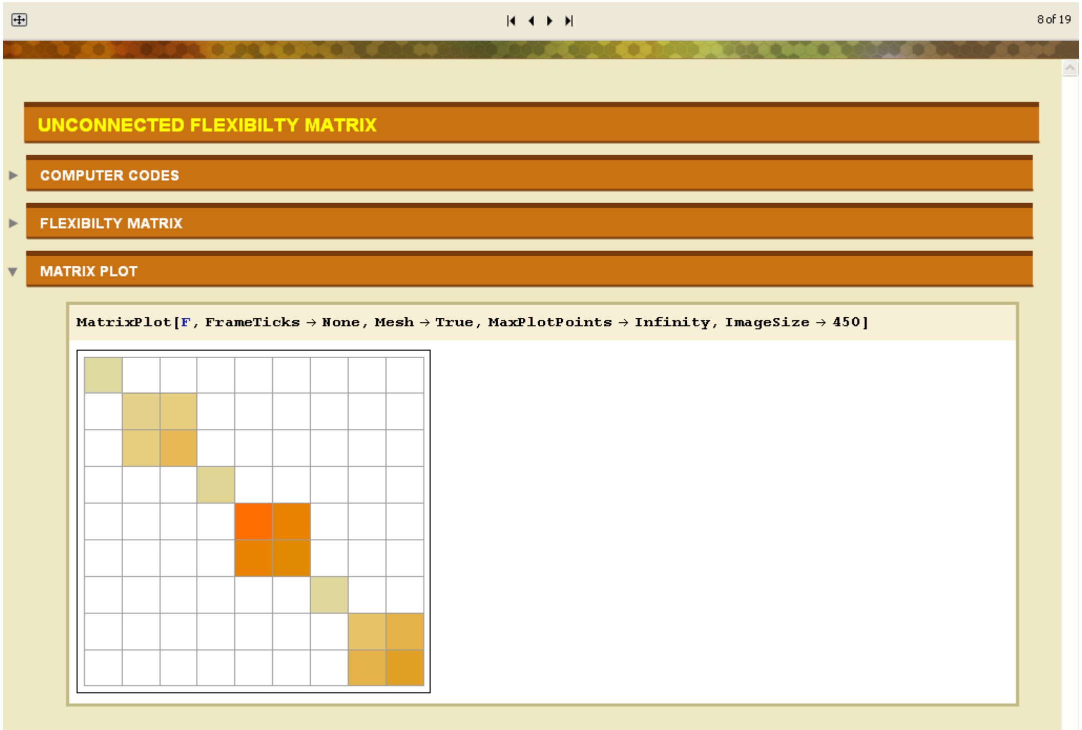


Figure 125: Calculation Phase, Step2, Matrix Plot of Unconnected Flexibility Matrix [G] $_{9\times9}$

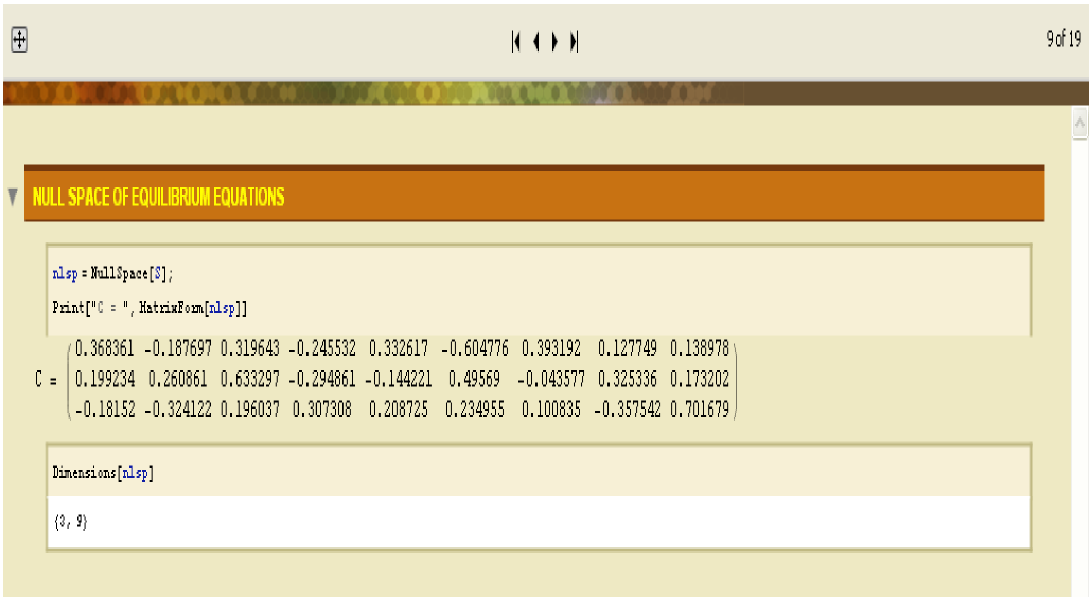


Figure 126: Calculation Phase, Step3, Compatibility Matrix

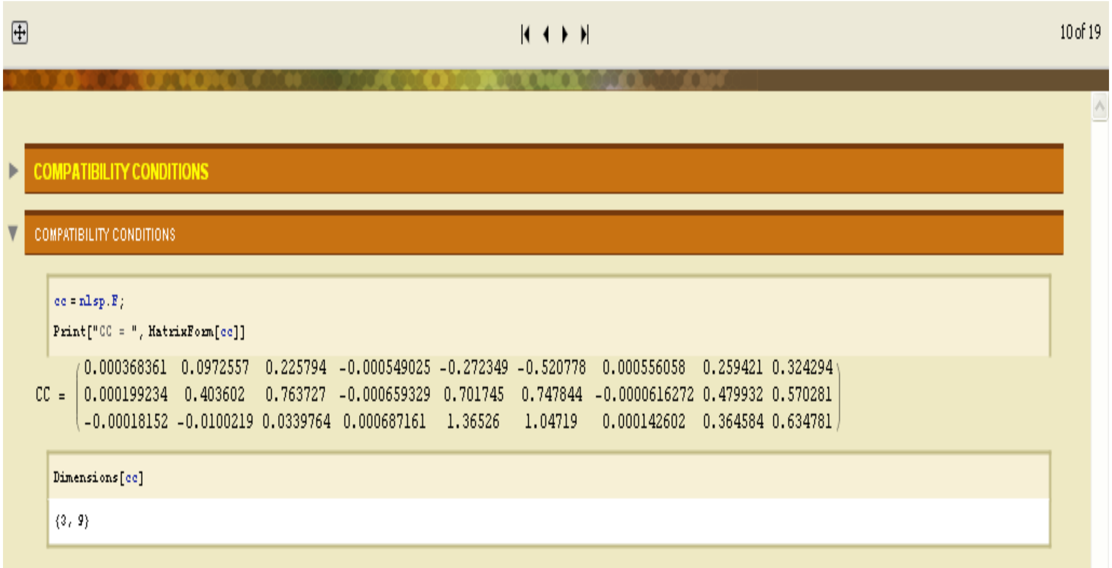


Figure 127: Calculation Phase, Step4, Compatibility Matrix

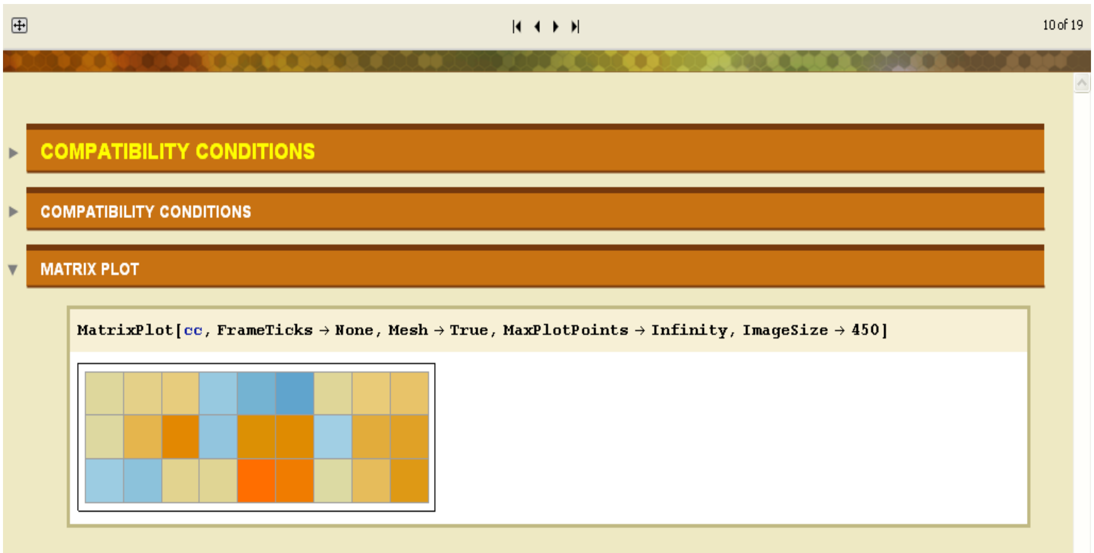
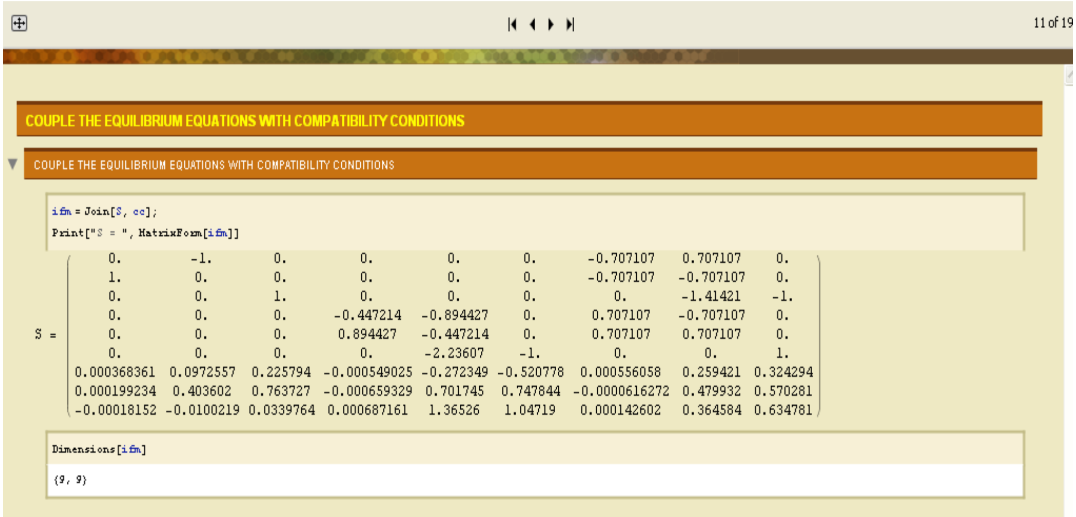


Figure 128: Calculation Phase, Step4, Matrix Plot of Compatibility Conditions [CC] 3×9



11 of 19

COUPLE THE EQUILIBRIUM EQUATIONS WITH COMPATIBILITY CONDITIONS

COUPLE THE EQUILIBRIUM EQUATIONS WITH COMPATIBILITY CONDITIONS

```
ifm = Join[S, cc];
Print["S = ", MatrixForm[ifm]]
```

S =

0.	-1.	0.	0.	0.	0.	-0.707107	0.707107	0.
1.	0.	0.	0.	0.	0.	-0.707107	-0.707107	0.
0.	0.	1.	0.	0.	0.	0.	-1.41421	-1.
0.	0.	0.	-0.447214	-0.894427	0.	0.707107	-0.707107	0.
0.	0.	0.	0.894427	-0.447214	0.	0.707107	0.707107	0.
0.	0.	0.	0.	-2.23607	-1.	0.	0.	1.
0.000368361	0.0972557	0.225794	-0.000549025	-0.272349	-0.520778	0.000556058	0.259421	0.324294
0.000199234	0.403602	0.763727	-0.000659329	0.701745	0.747844	-0.0000616272	0.479932	0.570281
-0.00018152	-0.0100219	0.0339764	0.000687161	1.36526	1.04719	0.000142602	0.364584	0.634781

```
Dimensions[ifm]
```

{9, 9}

Figure 129: Calculation Phase, Step5, Coupled Equilibrium Equations with Compatibility Conditions

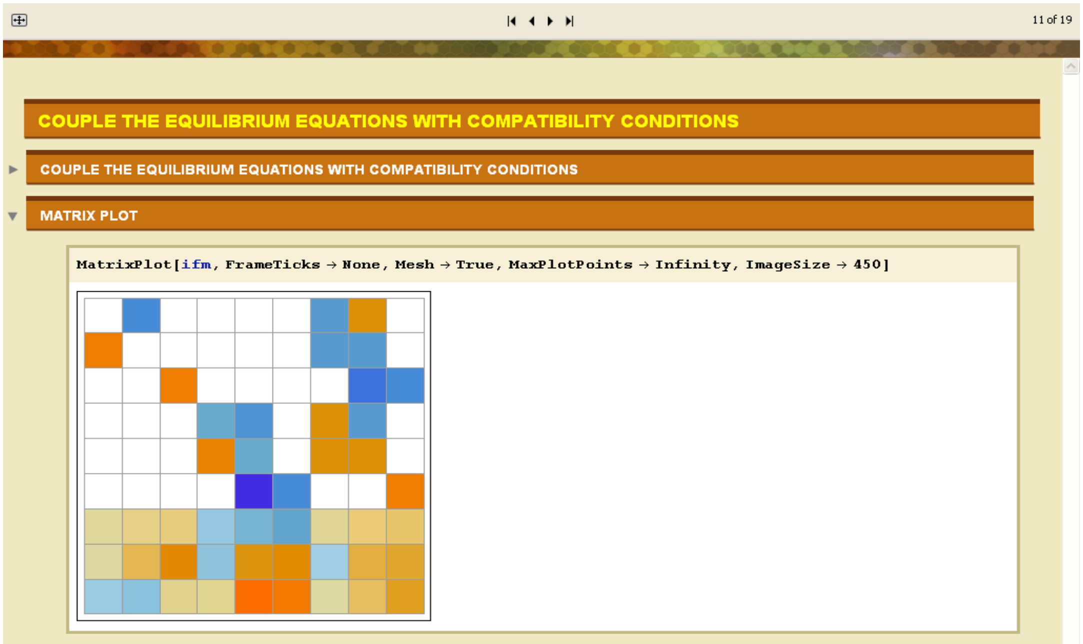


Figure 130: Calculation Phase, Step5, Matrix Plot of Coupled Equilibrium Equations with Compatibility Conditions [S] $_{9\times9}$

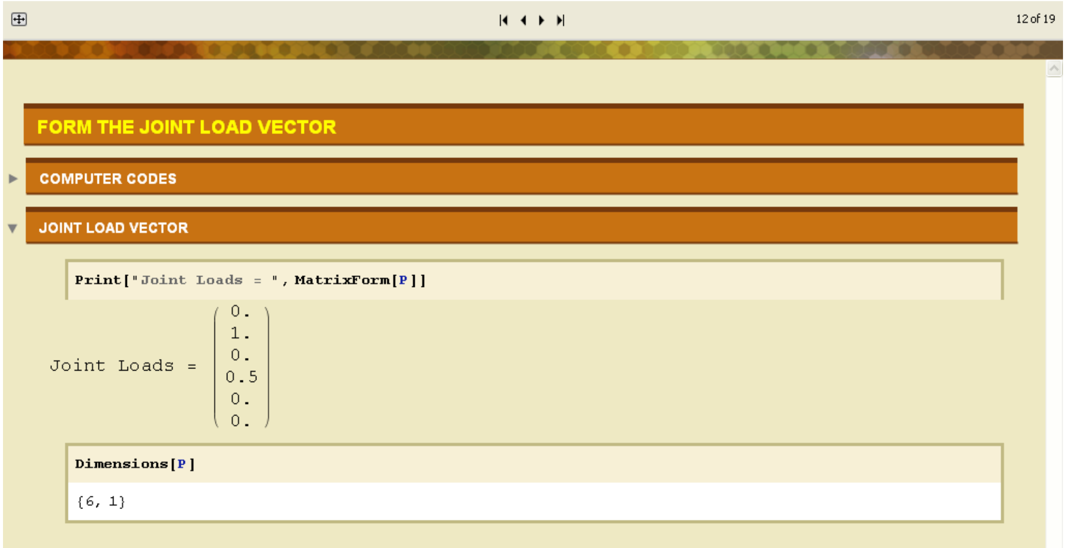


Figure 131: Calculation Phase, Step6, Form Joint Load Vector

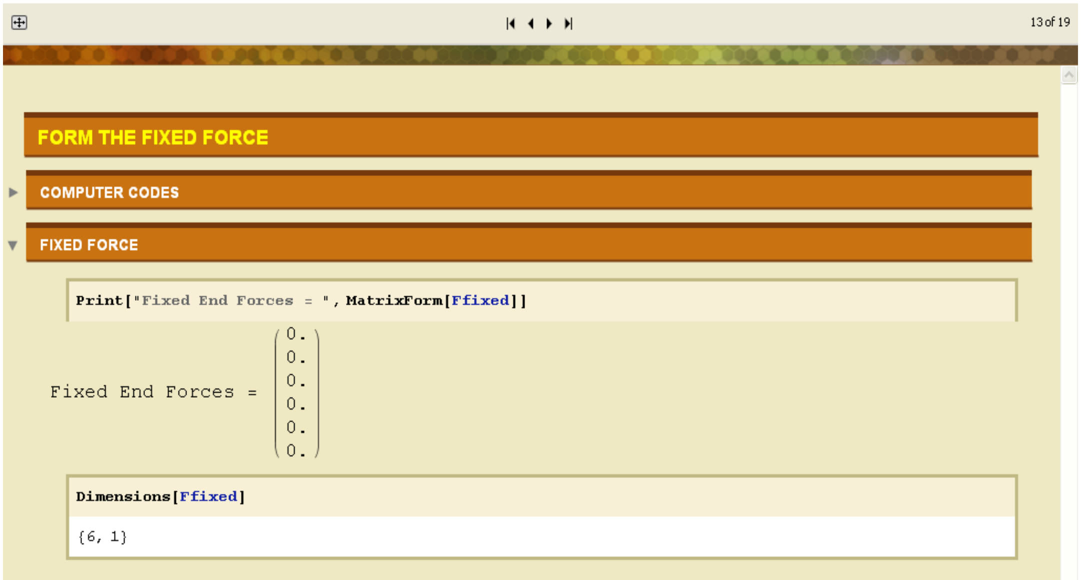


Figure 132: Calculation Phase, Step7, Form Fixed End Forces

Figure 133: Calculation Phase, Step 8, Combined Joint Load Vector with Fixed End Forces

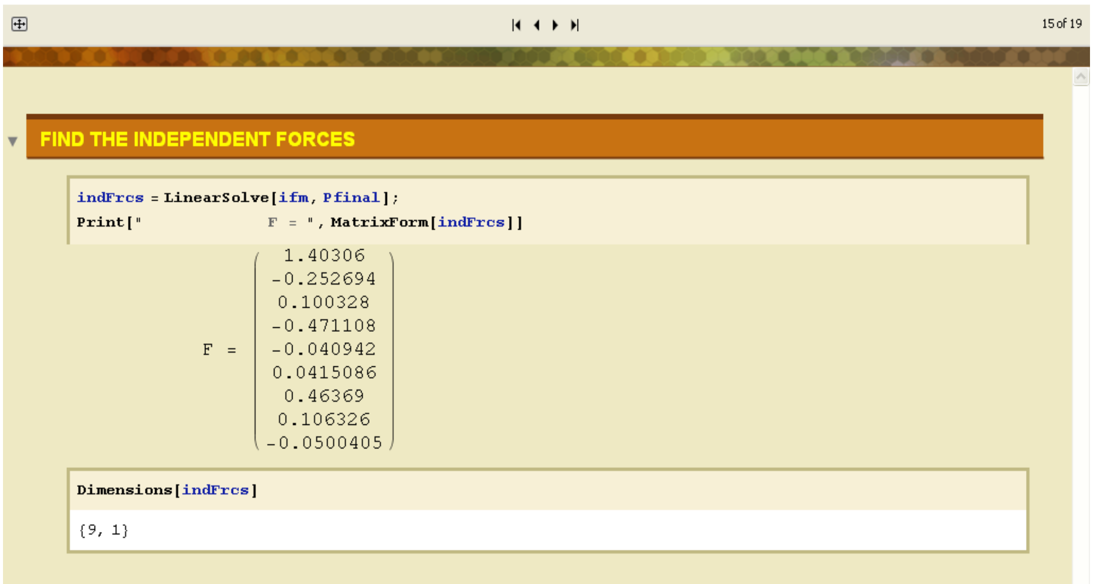


Figure 134: Calculation Phase, Step 9, Solve for Independent Forces

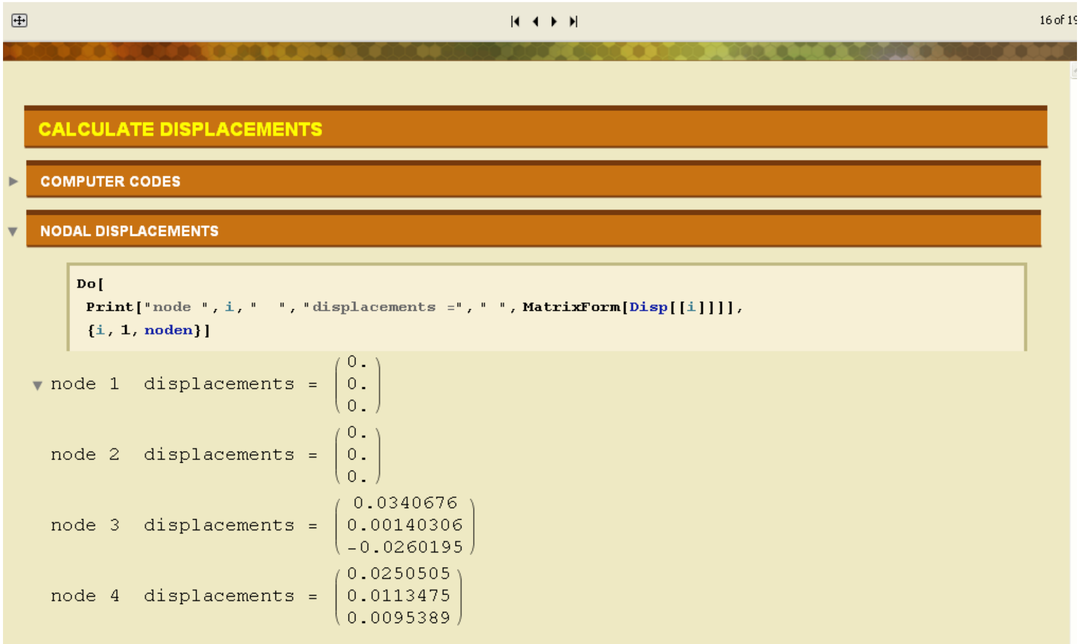


Figure 135: Reporting Results Phase, step1, Nodal Displacements of each node

Figure 136: Reporting Results Phase, step2, Member End Forces in Local Coordinate

```
reaction x=  -0.252694
reaction y=  -1.40306
moment z=  0.152367
member  1 react x  1
member  1 react y  1
member  1 react z  1
reaction x=  -0.247306
reaction y=  0.403062
moment z=  0.0415086
member  2 react x  2
member  2 react y  2
member  2 react z  2
```
Figure 137: Reporting Results Phase, step3, Support Reactions

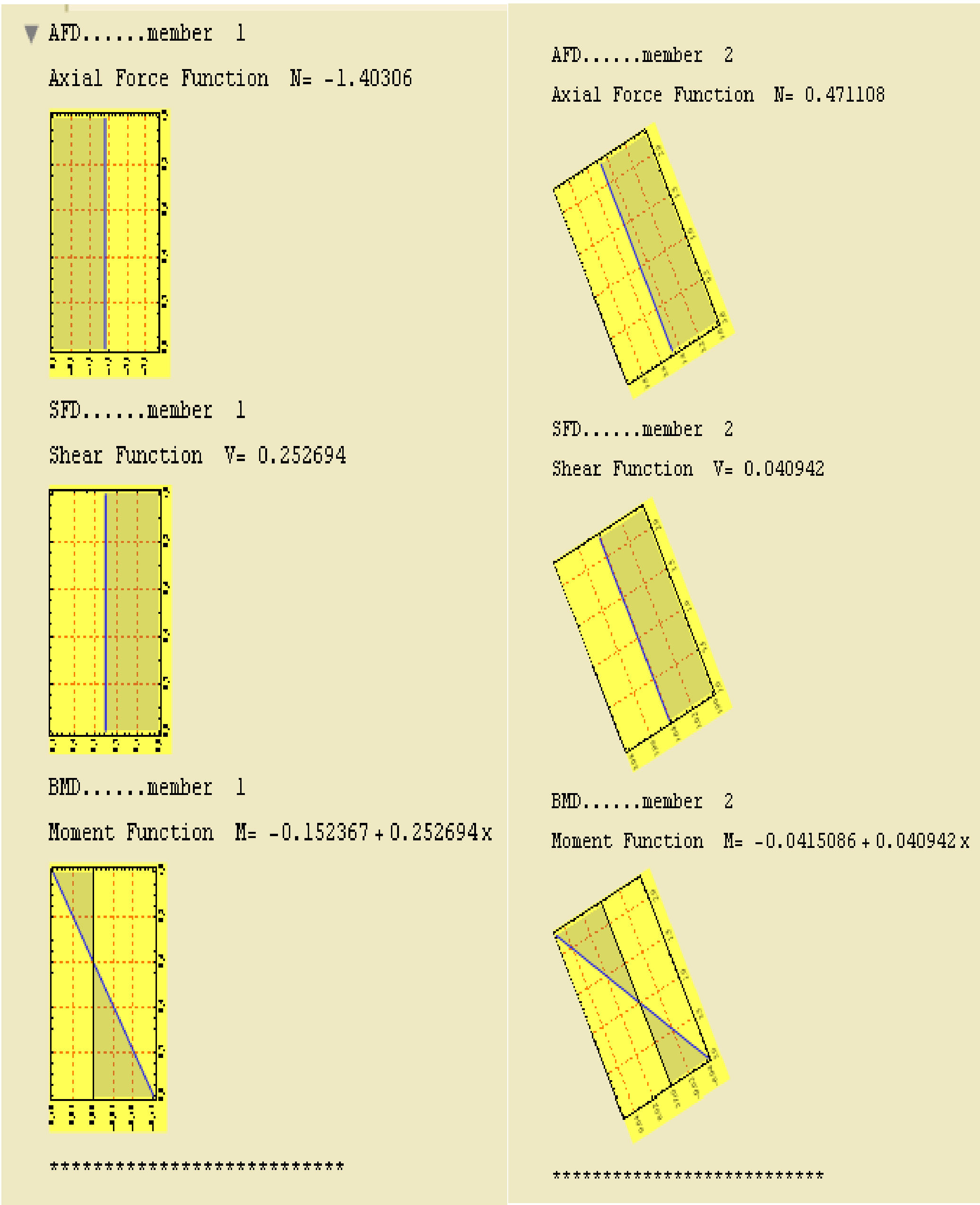


Figure 138: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 1, 2

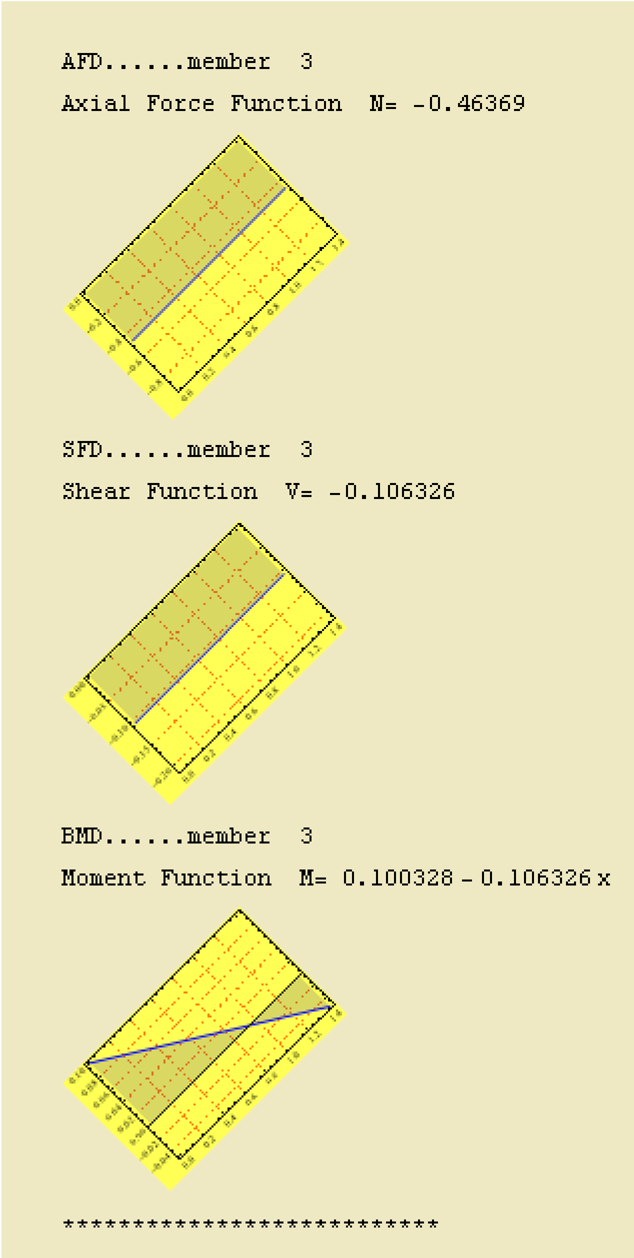


Figure 139: Reporting Results Phase, step4, Axial Force, Shear Force and Bending Moment Diagrams of member 3

```
##############################
Results of Structural Analysis
##############################

General Information:
      Structure Analyzed as:  Planar Frame
      Analysis Type:  First-Order Elastic

Analytical Results:

(i)   Displacements at Step # 1, Applied Load Ratio = 1.0000

  Deflections
     Node        X-disp            Y-disp            Z-disp
       1      0.0000e+000      0.0000e+000      0.0000e+000
       2      0.0000e+000      0.0000e+000      0.0000e+000
       3      3.4068e-002      1.4031e-003      0.0000e+000
       4      2.5051e-002      1.1348e-002      0.0000e+000

  Rotations (radians)
     Node        X-rot             Y-rot             Z-rot
       1      0.0000e+000      0.0000e+000      0.0000e+000
       2      0.0000e+000      0.0000e+000      0.0000e+000
       3      0.0000e+000      0.0000e+000     -2.6019e-002
       4      0.0000e+000      0.0000e+000      9.5389e-003
```

Figure 140: Final results of Example 4 from Mastan2 v 3.2 (Displacements)

```
(ii)  Element Results at Step # 1, Applied Load Ratio = 1.0000

  Internal End Forces (Note: Refers to local coordinates)
    Element  Node        Fx               Fy               Fz
       1       1    -1.4031e+000     2.5269e-001     0.0000e+000
               3     1.4031e+000    -2.5269e-001     0.0000e+000
       2       4     4.7111e-001     4.0942e-002     0.0000e+000
               2    -4.7111e-001    -4.0942e-002     0.0000e+000
       3       3    -4.6369e-001    -1.0633e-001     0.0000e+000
               4     4.6369e-001     1.0633e-001     0.0000e+000

  Internal End Moments (Note: Refers to local coordinates)
    Element  Node        Mx               My               Mz               B
       1       1     0.0000e+000     0.0000e+000     1.5237e-001     0.0000e
               3     0.0000e+000     0.0000e+000     1.0033e-001     0.0000e
       2       4     0.0000e+000     0.0000e+000     5.0040e-002     0.0000e
               2     0.0000e+000     0.0000e+000     4.1509e-002     0.0000e
       3       3     0.0000e+000     0.0000e+000    -1.0033e-001     0.0000e
               4     0.0000e+000     0.0000e+000    -5.0040e-002     0.0000e

(iii) Reactions at Step # 1, Applied Load Ratio = 1.0000

  Forces
     Node          Rx              Ry              Rz
       1      -2.5269e-001    -1.4031e+000        FREE
       2      -2.4731e-001     4.0306e-001        FREE

  Moments
     Node          Mx              My              Mz
       1          FREE            FREE         1.5237e-001
       2          FREE            FREE         4.1509e-002

#####################################
End of Results of Structural Analysis
#####################################
```

Figure 141: Final results of Example 4 from Mastan2 v 3.2 (Element Results and Reactions)

6.5.1 Comparison Results:

The results obtained from Integrated Force Method Analysis Package and Mastan2 v 3.2 [17] and the result obtained from reference [12] are compared and all the diagrams and the numerical results are agree fully.

6.6 Summary

In this chapter four examples have been presented in order to illustrate the usage of Analysis Packages and the results are compared with Mastan2 v 3.2 [17] results.

Chapter 7

CONCLUSION

7.1 Conclusion

Coplanar rigid frames have been analyzed using the Integrated Force Method, where, the process of selection of redundants is not required and independent internal forces for the structural design process is determined for all the members in only one solution process. Thus the Integrated Force Method provides considerable advantage in the design of large scale structures like in the aeronautical industry and in the design of steel structures. IFM's advantages [3] over the Stiffness Method (SM) have been documented, including accurate stress results, a well-conditional system for finite element discrete analysis, fast convergence to correct solutions The Integrated Force Method can also be extended to nonlinear structural analysis [19] and optimization problems [21].

The second method used to analyze coplanar rigid frames is the Dual Integrated Force Method, IFMD, which is essentially a displacement method. In IFMD, only the equilibrium matrix and the unconnected stiffness matrix of the structure are used to generate the global stiffness matrix of the structure. Thus there is no need to write lengthy and complex computer programs to generate global stiffness matrix of the structure. The global stiffness matrix of the structure in IFMD is obtained by simple programming using computer algebra system Mathematica7. Therefore in the analysis of skeletal structures it is very advantageous to use the Dual Integrated Force Method.

In this thesis three analysis packages for indeterminate rigid frame have been developed. The two main methods used are

1) Integrated Force Method
2) Dual Integrated Force Method

Two of the packages use the IFM and the compatibility matrices were obtained by using the Null space of the equilibrium matrix, and Singular Value Decomposition of the equilibrium matrix. The third package employed the Dual Integrated Force Method. Various problems have been analyzed by these three packages and the results have been compared with the results form Mastan2 v3.2. [17] .all the results are fully agree.

7.2 Summary of Contributions

Three analysis packages have been developed and strategy of this development produced the following specific characteristics:

1) **Easy to use**: there is no need to read any manual.
2) **Simple:** analysis packages are easy to run.
3) **Transparent Theory:** the theory is explained at each step.
4) **Chasing Variables:** Step by Step calculations make it easy to follow the values of variables.
5) **Flexible:** It is possible to change its utility.
6) **Educational:** It is an opportunity for students to learn at their own pace.
7) **Accessible:** It is available for students, instructors, researchers and engineers

without any restriction.

7.4 Future Research

For future work the following items are recommended:

a. Develop analysis for space frames and space trusses.
b. Develop analysis considering nonlinear theories.
c. Make programs running on the web pages and make more accessible.
d. Include support settlements, initial deformations and temperature differentials.
e. Incorporate different types of distributed loading into the analysis packages.
f. Investigation of the conditioning of the main IFM matrix. As the number of members gets larger the condition number of the IFM matrix may worsen leading to inaccurate results.

REFERENCES

[1] Fillip C. Fillppou . CE 220 Structural Analysis Theory, Lecture Notes, Chapter 2 , University of California, Berkley, (2001).

[2] S.N. Patnaik, D.A. Hpkins, Strength of Materials: A Unified for the 21st Centry, pp.274-431, Butterworth-Heinemann, 2004.

[3] S.N. Ptnaik, Integrated Force Method Solution to Indeterminate Structural Mechanics Problems, NASA/TP_ 2004-207430 (2004).

[4] S.N. Patnaik, L. Berke,R.H. Gallagher, Integrated Force Method versus Displacements Method for Finite Element Analysis, Comput. Struct., vol.38, no. 4, pp. 37_407, 1991.

[5] S.N. Patnaik, Compatibility Conditions in Structural Mechanics, NASA/TM _ 1999-209175.

[6] I. Kaljevic, S.N. Patnaik, D.A. Hopkins, Treatment of Initial Deformation in the Integrated Force Method, Comput, Methods Appl. Mech. Engrg. 140 (1997) 281 _ 289, 1996.

[7] R. Sedaghati, A. Suleman, Force Method Revisited, AIAA Journal, vol. 41, No. 5, May 2003.

[8] R. Sedaghati, Benchmark Case Studies in Structural Design Optimization Using the Force Method, International Journal of solids and structures 42 (2005). 5848_5871.

[9] S.N. Patnsik, and D. A. Hopkins, Recent Advances in the Method of Force Method of Structural Analysis, Advances Engrg. Software, Vol.29, no.3_6, pp. 463_474, 1998.

[10] R. Schottler, Java Applets for Analysis of Trusses, Beam and Frames, M.S. Thesis, Virginia Polytechnic Institute and State University (2004).

[11] M.D. Moh'D, Matrix Analysis of two Dimensional Frames, M.S. Thesis, EMU, (1994).

[12] S. Sensoy, Two Dimensional Structural Analysis by Flexibility Method, M.S. Thesis, EMU, (1995).

[13] M.Under, Structural Analysis by Neglecting Axial Deformations, M.S. Thesis, EMU, (1995).

[14] S.K. Movaghar, Usage of Equilibrium Equations in Truss Analysis, M.S. Thesis, EMU, (2005).

[15] R. Esmaeili, Analysis Packages for Indeterminate Beam Problems, M.S. Thesis, EMU, (2005).

[16] S.Wolfram, , 1959 – Mathematica book / Stephen Wolfram. — 5th ed. p. cm. Includes index. ISBN 1–57955–022–3.

[17] Mastan2 v3: Developed by R.D. Ziemian and W. McGuire, Distributed by John Wiley and Sons, Inc. (2000).

[18] Dhananjaya, HR and Nagabhushanam, J and Pandey, Automatic generation of equilibrium and flexibility matrices for plate bending elements using Integrated Force Method, Structural Engineering and Mechanics, 30 (4). pp. 387-402, (2008).

[19] N R B Krishnam Raju and J Nagabhushanam, Nonlinear structural analysis using integrated force method, *Sadhana* Vol. 25, Part 4, pp. 353-365,(2004).

[20] A Chajes, Structural Analysis, (Prentice-Hall international series in civil engineering and engineering mechanics), ISBN 0-13-855073-5

APPENDIX

Computer Codes

This chapter presents the Mathematica Computer Codes to generated the methods which

were explained in chapter 2.

a) Computer Codes for Integrated Force Method are included as follows:

```
lines = {}; length = {};
Do[
  node1 = inc[[i, 2]];
  node2 = inc[[i, 3]];
  px = cord[[node1, 2]];
  py = cord[[node1, 3]];
  qx = cord[[node2, 2]];
  qy = cord[[node2, 3]];
   len = Sqrt[(qx - px)^2 + (qy - py)^2];
  sawline = Line[{{px, py}, {qx, qy}}];
  lines = AppendTo[lines, sawline];
  length = AppendTo[length, len];
  , {i, 1, m}];
sawgraph1 = Graphics[{AbsoluteThickness[3], lines}];
GrSt = Max[length] / 40;
Fixed1[x_, y_] := {
   Table[Line[{{x + (Hsh - 2) * GrSt, y - 2 * GrSt}, {x + Hsh * GrSt, y - 0.5 * GrSt}}], {Hsh, -4, 4}],
   Line[{{x - 4 * GrSt, y - 0.5 * GrSt}, {x + 4 * GrSt, y - .5 * GrSt}}]
  };
Fixed2[x_, y_] := {
   Table[Line[{{x + (Hsh - 2) * GrSt, y + 2 * GrSt}, {x + Hsh * GrSt, y + 0.5 * GrSt}}], {Hsh, -4, 4}],
   Line[{{x - 4 * GrSt, y + 1 * GrSt}, {x + 4 * GrSt, y + .5 * GrSt}}]
  };
Simple1[x_, y_] := {
   Line[{{x - 2 * GrSt, y - 3.5 * GrSt}, {x, y}, {x + 2 * GrSt, y - 3.5 * GrSt}, {x - 2 * GrSt, y - 3.5 * GrSt}}],
   Table[Line[{{x + (Hsh - 1) * GrSt, y - 4.5 * GrSt}, {x + Hsh * GrSt, y - 3.5 * GrSt}}], {Hsh, -3, 3}],
   Line[{{x - 3 * GrSt, y - 3.5 * GrSt}, {x + 3 * GrSt, y - 3.5 * GrSt}}]
  };
```

Figure 142: Computer Codes for Graphical Shape of Frame

```
Simple2[x_, y_] := {
   Line[{{x - 2 * GrSt, y + 3.5 * GrSt}, {x, y}, {x + 2 * GrSt, y + 3.5 * GrSt}, {x - 2 * GrSt, y + 3.5 * GrSt}}],
   Table[Line[{{x + (Hsh - 1) * GrSt, y + 4.5 * GrSt}, {x + Hsh * GrSt, y + 3.5 * GrSt}}], {Hsh, -3, 3}],
   Line[{{x - 3 * GrSt, y + 3.5 * GrSt}, {x + 3 * GrSt, y + 3.5 * GrSt}}]
  };
Roller1[x_, y_] := {
   Line[{{x - 2 * GrSt, y - 3.5 * GrSt}, {x, y}, {x + 2 * GrSt, y - 3.5 * GrSt}, {x - 2 * GrSt, y - 3.5 * GrSt}}],
   Table[Circle[{x + Hsh * GrSt, y - 4 * GrSt}, .5 * GrSt], {Hsh, -1, 1}],
   Line[{{x - 3 * GrSt, y - 4.5 * GrSt}, {x + 3 * GrSt, y - 4.5 * GrSt}}]
  };
Roller2[x_, y_] := {
   Line[{{x - 2 * GrSt, y + 3.5 * GrSt}, {x, y}, {x + 2 * GrSt, y + 3.5 * GrSt}, {x - 2 * GrSt, y + 3.5 * GrSt}}],
   Table[Circle[{x + Hsh * GrSt, y + 4 * GrSt}, .5 * GrSt], {Hsh, -1, 1}],
   Line[{{x - 3 * GrSt, y + 4.5 * GrSt}, {x + 3 * GrSt, y + 4.5 * GrSt}}]
  };
sawsuport = {};
Do[
  If[
   freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Fixed1[cord[[i, 2]], cord[[i, 3]]]]
  ];
  If[
   freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Fixed2[cord[[i, 2]], cord[[i, 3]]]]
  ];
  If[
   freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Simple1[cord[[i, 2]], cord[[i, 3]]]]
```

Figure 143: Computer Codes for Graphical Shape of Frame (continued)

```
  freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Simple1[cord[[i, 2]], cord[[i, 3]]]]
 ];
f[
  freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Simple1[cord[[i, 2]], cord[[i, 3]]]]
 ];
 If[
  freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Simple2[cord[[i, 2]], cord[[i, 3]]]]
 ];
f[
  freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Simple2[cord[[i, 2]], cord[[i, 3]]]]
 ];
 If[
  freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Simple2[cord[[i, 2]], cord[[i, 3]]]]
 ];
 If[
  freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Roller1[cord[[i, 2]], cord[[i, 3]]]]
 ];
f[
  freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
  sawsuport = AppendTo[sawsuport, Roller1[cord[[i, 2]], cord[[i, 3]]]]
 ];
```

Figure 144: Computer Codes for Graphical Shape of Frame (continued)

```
If[
   freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Min[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Roller1[cord[[i, 2]], cord[[i, 3]]]]
  ];
  If[
   freet[[i, 2]] == 1 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Roller2[cord[[i, 2]], cord[[i, 3]]]]
  ];
If[
   freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 1 ∧ freet[[i, 4]] == 0 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Roller2[cord[[i, 2]], cord[[i, 3]]]]
  ];
If[
   freet[[i, 2]] == 0 ∧ freet[[i, 3]] == 0 ∧ freet[[i, 4]] == 1 ∧ cord[[i, 3]] == Max[cord[[All, 3]]],
   sawsuport = AppendTo[sawsuport, Roller2[cord[[i, 2]], cord[[i, 3]]]]
  ],
  {i, 1, noden}];
sawgraph2 = Graphics[{AbsoluteThickness[1], RGBColor[0, 0, 1], sawsuport}];
Grno = Max[length] / 300;
Show[sawgraph1, sawgraph2, AspectRatio → Automatic, PlotRange → All, ImageSize → 350]
```

Figure 145: Computer Codes for Graphical Shape of Frame (continued)

```
Grno = Max[length] / 300;
Show[sawgraph1, sawgraph2, AspectRatio → Automatic, PlotRange → All, ImageSize → 350]

elnumbering = {};
Do[
  node1 = inc[[i, 2]];
  node2 = inc[[i, 3]];
  px = cord[[node1, 2]];
  py = cord[[node1, 3]];
  qx = cord[[node2, 2]];
  qy = cord[[node2, 3]];
  elnum = Text[ToString[i], {(3 px + qx) / 4 + 12 * Grno, (3 py + qy) / 4 + 18 * Grno}];
elnumbering = AppendTo[elnumbering, elnum]
  , {i, 1, m}
 ];
sawgraph3 = Graphics[{RGBColor[0, 0, 1], elnumbering}];
Grnod = Max[length] / 200;

nodenumbering = {};
Do[
  nodenum = Text[ToString[p], {(cord[[p, 2]]) + 8 * Grnod, cord[[p, 3]] + 10 * Grnod}];
  nodenumbering = AppendTo[nodenumbering, nodenum]
  , {p, 1, noden}];
sawgraph4 = Graphics[{AbsoluteThickness[8], RGBColor[1, 0, 1], nodenumbering}];

Show[sawgraph1, sawgraph2, sawgraph3, sawgraph4, AspectRatio → Automatic, PlotRange → All, ImageSize → 350]
```

Figure 146: Computer Codes for Graphical Shape of Frame (continued)

```
Off[General::spell]
Lmem = {};
bgmem = {};
bmem = {};
tmem = {};
Qfmem = {};
Ffmem = {};
memθ = {};
Off[General::spell]
Do[

  mincb = inc[[i, 2]];
  mince = inc[[i, 3]];
  dtabl = {3 * mincb - 2, 3 * mincb - 1, 3 * mincb,
           3 * mince - 2, 3 * mince - 1, 3 * mince};
  xd = cord[[mince, 2]] - cord[[mincb, 2]];
  yd = cord[[mince, 3]] - cord[[mincb, 3]];
  If[freet[[mincb, 2]] == 1, ReplacePart[dtabl, 0, 1]];
  If[freet[[mincb, 3]] == 1, ReplacePart[dtabl, 0, 2]];
  If[freet[[mincb, 4]] == 1, ReplacePart[dtabl, 0, 3]];
  If[freet[[mince, 2]] == 1, ReplacePart[dtabl, 0, 4]];
  If[freet[[mince, 3]] == 1, ReplacePart[dtabl, 0, 5]];
  If[freet[[mince, 4]] == 1, ReplacePart[dtabl, 0, 6]];
  Lm = Sqrt[xd^2 + yd^2];
  CC = xd/Lm;
  SS = yd/Lm;
  θ = ArcCos[CC];
  AppendTo[memθ, θ];
```
Figure 147: Computer Codes for generating Equilibrium Equations [B]

$$b = \begin{pmatrix} -1. & 0 & 0 \\ 0 & -1. & 0 \\ 0 & -Lm & -1. \\ 1. & 0 & 0 \\ 0 & 1. & 0 \\ 0 & 0 & 1. \end{pmatrix};$$

$$t = \begin{pmatrix} CC & -SS & 0. & 0. & 0. & 0. \\ SS & CC & 0. & 0. & 0. & 0. \\ 0. & 0. & 1 & 0. & 0. & 0. \\ 0. & 0. & 0. & CC & -SS & 0. \\ 0. & 0. & 0. & SS & CC & 0. \\ 0. & 0. & 0. & 0. & 0. & 1 \end{pmatrix};$$

```
bg = t.b;
AppendTo[bmem, b];
AppendTo[Lmem, Lm];
AppendTo[tmem, t];
AppendTo[bgmem, bg];
```

$$Qf = \begin{pmatrix} 0. \\ \frac{\omega[[i]]*Lm}{2.} \\ \frac{\omega[[i]]*Lm^2}{12.} \\ 0. \\ \frac{\omega[[i]]*Lm}{2.} \\ -\frac{\omega[[i]]*Lm^2}{12.} \end{pmatrix};$$

```
Ff = t.Qf;
AppendTo[Ffmem, Ff];
AppendTo[Qfmem, Qf];
, {i, 1, m}];
```

Figure 148: Computer Codes for generating Equilibrium Equations (continued)

```
rest = 0;
Do[If[freet[[i, 2]] == 1, rest = rest + 1];

 If[freet[[i, 3]] == 1, rest = rest + 1];
 If[freet[[i, 4]] == 1, rest = rest + 1],
 {i, 1, noden}]
dof = Table[jj, {jj, 1, 3 * noden}];

Clear[kk]
kk = 0;
Do[
 cj1 = 3 * i - 2;
 cj2 = 3 * i - 1;
 cj3 = 3 * i;
 If[freet[[i, 2]] == 1, dof[[3 * i - 2]] = 0];
 If[freet[[i, 2]] == 0, kk = kk + 1];
 If[freet[[i, 2]] == 0, dof[[3 * i - 2]] = kk];
 If[freet[[i, 3]] == 1, dof[[3 * i - 1]] = 0];
 If[freet[[i, 3]] == 0, kk = kk + 1];
 If[freet[[i, 3]] == 0, dof[[3 * i - 1]] = kk];
 If[freet[[i, 4]] == 1, dof[[3 * i]] = 0];
 If[freet[[i, 4]] == 0, kk = kk + 1];
 If[freet[[i, 4]] == 0, dof[[3 * i]] = kk],
```

Figure 149: Computer Codes for generating Equilibrium Equations (continued)

```
S = Table[0., {sr, 1, 3 * noden - rest}, {sc, 1, 3 * m}];
mi = 0;
Do[
  node1 = inc[[i, 2]];
  node2 = inc[[i, 3]];
  k1 = 3 * node1 - 2;
  k2 = 3 * node1 - 1;
  k3 = 3 * node1;
  mi = mi + 1;
  c1 = 3 * mi - 2;
  c2 = 3 * mi - 1;
  c3 = 3 * mi;
  k4 = 3 * node2 - 2;
  k5 = 3 * node2 - 1;
  k6 = 3 * node2;
  kc1 = dof[[k1]];
  kc2 = dof[[k2]];
  kc3 = dof[[k3]];
  kc4 = dof[[k4]];
  kc5 = dof[[k5]];
  kc6 = dof[[k6]];
```

Figure 150: Computer Codes for generating Equilibrium Equations (continued)

```
  If[kc1 ≠ 0, S[[kc1, c1]] = S[[kc1, c1]] + bgmem[[i]][[1, 1]]];
  If[kc1 ≠ 0, S[[kc1, c2]] = S[[kc1, c2]] + bgmem[[i]][[1, 2]]];
  If[kc1 ≠ 0, S[[kc1, c3]] = S[[kc1, c3]] + bgmem[[i]][[1, 3]]];
  If[kc2 ≠ 0, S[[kc2, c1]] = S[[kc2, c1]] + bgmem[[i]][[2, 1]]];
  If[kc2 ≠ 0, S[[kc2, c2]] = S[[kc2, c2]] + bgmem[[i]][[2, 2]]];
  If[kc2 ≠ 0, S[[kc2, c3]] = S[[kc2, c3]] + bgmem[[i]][[2, 3]]];
  If[kc3 ≠ 0, S[[kc3, c1]] = S[[kc3, c1]] + bgmem[[i]][[3, 1]]];
  If[kc3 ≠ 0, S[[kc3, c2]] = S[[kc3, c2]] + bgmem[[i]][[3, 2]]];
  If[kc3 ≠ 0, S[[kc3, c3]] = S[[kc3, c3]] + bgmem[[i]][[3, 3]]];
  If[kc4 ≠ 0, S[[kc4, c1]] = S[[kc4, c1]] + bgmem[[i]][[4, 1]]];
  If[kc4 ≠ 0, S[[kc4, c2]] = S[[kc4, c2]] + bgmem[[i]][[4, 2]]];
  If[kc4 ≠ 0, S[[kc4, c3]] = S[[kc4, c3]] + bgmem[[i]][[4, 3]]];
  If[kc5 ≠ 0, S[[kc5, c1]] = S[[kc5, c1]] + bgmem[[i]][[5, 1]]];
  If[kc5 ≠ 0, S[[kc5, c2]] = S[[kc5, c2]] + bgmem[[i]][[5, 2]]];
  If[kc5 ≠ 0, S[[kc5, c3]] = S[[kc5, c3]] + bgmem[[i]][[5, 3]]];
  If[kc6 ≠ 0, S[[kc6, c1]] = S[[kc6, c1]] + bgmem[[i]][[6, 1]]];
  If[kc6 ≠ 0, S[[kc6, c2]] = S[[kc6, c2]] + bgmem[[i]][[6, 2]]];
  If[kc6 ≠ 0, S[[kc6, c3]] = S[[kc6, c3]] + bgmem[[i]][[6, 3]]],
  {i, 1, m}];
Print["EE = ", MatrixForm[S]]
MatrixPlot[S, FrameTicks → None, Mesh → True, MaxPlotPoints → Infinity, ImageSize → 450]
```

Figure 151: Computer Codes for generating Equilibrium Equations (continued)

```
F = Table[0., {sr, 1, 3*m}, {sc, 1, 3*m}];
mi = 0;
Do[
  node1 = inc[[i, 2]];
  node2 = inc[[i, 3]];
  k1 = 3*node1 - 2;
  k2 = 3*node1 - 1;
  k3 = 3*node1;
  mi = mi + 1;
  c1 = 3*mi - 2;
  c2 = 3*mi - 1;
  c3 = 3*mi;
  k4 = 3*node2 - 2;
  k5 = 3*node2 - 1;
  k6 = 3*node2;
  kc1 = dof[[k1]];
  kc2 = dof[[k2]];
  kc3 = dof[[k3]];
  kc4 = dof[[k4]];
  kc5 = dof[[k5]];
  kc6 = dof[[k6]];
  flex1 = Lmem[[i]] / (a[[i]] × Ee);
  flex2 = (Lmem[[i]])^3 / (3. × Ee × Ii[[i]]);
  flex3 = (Lmem[[i]]) / (Ee × Ii[[i]]);
  flex4 = (Lmem[[i]])^2 / (2. × Ee × Ii[[i]]);
```

Figure 152: Computer Codes for generating Unconnected Flexibility Matrix [G]

```
fm = ( flex1   0      0
       0       flex2  flex4
       0       flex4  flex3 );
F[[c1, c1]] = F[[c1, c1]] + fm[[1, 1]];
F[[c2, c2]] = F[[c2, c2]] + fm[[2, 2]];
F[[c3, c3]] = F[[c3, c3]] + fm[[3, 3]];
F[[c2, c2 + 1]] = F[[c2, c2 + 1]] + fm[[2, 3]];
F[[c3, c3 - 1]] = F[[c3, c3 - 1]] + fm[[3, 2]];
,
{i, 1, m}];
Print["G = ", MatrixForm[F]]
MatrixPlot[F, FrameTicks → None, Mesh → True, MaxPlotPoints → Infinity, ImageSize → 450]
```

Figure 153: Computer Codes for generating Unconnected Flexibility Matrix (continued)

```
nlsp = NullSpace[S];
Print["null space  = ", MatrixForm[nlsp]]
```

Figure 154: Computer Codes for generating Compatibility Matrix [C]

```
cc = nlsp.F;
Print["cc = ", MatrixForm[cc]]
MatrixPlot[cc, FrameTicks → None, Mesh → True, MaxPlotPoints → Infinity, ImageSize → 450]
```

Figure 155: Computer Codes for generating Compatibility Conditions [CC]

```
ifm = Join[S, cc];
Print["ifm = ", MatrixForm[ifm]]
MatrixPlot[ifm, FrameTicks → None, Mesh → True, MaxPlotPoints → Infinity, ImageSize → 450]
```

Figure 156: Computer Codes for Coupling [B] and [CC]

```
Ffixed = Table[0., {sr, 1, 3 * noden - rest}, {sc, 1, 1}];
mi = 0;
Do[
 node1 = inc[[i, 2]];
 node2 = inc[[i, 3]];
 k1 = 3 * node1 - 2;
 k2 = 3 * node1 - 1;
 k3 = 3 * node1;
 mi = mi + 1;
 c1 = 3 * mi - 2;
 c2 = 3 * mi - 1;
 c3 = 3 * mi;
 k4 = 3 * node2 - 2;
 k5 = 3 * node2 - 1;
 k6 = 3 * node2;
 kc1 = dof[[k1]];
 kc2 = dof[[k2]];
 kc3 = dof[[k3]];
 kc4 = dof[[k4]];
 kc5 = dof[[k5]];
 kc6 = dof[[k6]];
 If[kc1 ≠ 0, Ffixed[[kc1, 1]] = Ffixed[[kc1, 1]] + Ffmem[[i]][[1, 1]]];
 If[kc2 ≠ 0, Ffixed[[kc2, 1]] = Ffixed[[kc2, 1]] + Ffmem[[i]][[2, 1]]];
 If[kc3 ≠ 0, Ffixed[[kc3, 1]] = Ffixed[[kc3, 1]] + Ffmem[[i]][[3, 1]]];
 If[kc4 ≠ 0, Ffixed[[kc4, 1]] = Ffixed[[kc4, 1]] + Ffmem[[i]][[4, 1]]];
 If[kc5 ≠ 0, Ffixed[[kc5, 1]] = Ffixed[[kc5, 1]] + Ffmem[[i]][[5, 1]]];
 If[kc6 ≠ 0, Ffixed[[kc6, 1]] = Ffixed[[kc6, 1]] + Ffmem[[i]][[6, 1]]];
 ,
 {i, 1, m}]
Print["Ffixed = ", MatrixForm[Ffixed]]
```
Figure 157: Computer Codes for Forming the Fixed End Forces

```
P = Table[0., {sr, 1, 3 * noden - rest}, {sc, 1, 1}];
Do[k1 = 3 * i - 2;
 k2 = 3 * i - 1;
 k3 = 3 * i;
 kc1 = dof[[k1]];
 kc2 = dof[[k2]];
 kc3 = dof[[k3]];
 If[kc1 ≠ 0, P[[kc1, 1]] = P[[kc1, 1]] + applfrcs[[i, 2]]];
 If[kc2 ≠ 0, P[[kc2, 1]] = P[[kc2, 1]] + applfrcs[[i, 3]]];
 If[kc3 ≠ 0, P[[kc3, 1]] = P[[kc3, 1]] + applfrcs[[i, 4]]],
 {i, 1, noden}]
Print["  P = ", MatrixForm[P]]
```

Figure 158: Computer Codes for Forming the Point Joint Load

```
di = 3 × m + rest - 3 × noden
```

Figure 159: Computer Codes for finding Degree of Indeterminacy

```
initial = Table[0., {sr, 1, di}, {sc, 1, 1}];
Pact = Join[P, initial];
Fact = Join[Ffixed, initial];
Pfinal = Pact - Fact;
Print["final load = ", MatrixForm[Pfinal]]
```

Figure 160: Computer Codes for combining Point Joint Load and Fixed End Forces

```
indFrcs = LinearSolve[ifm, Pfinal];
Print["independent forces = ", MatrixForm[indFrcs]]
```

Figure 161: Computer Codes for Solving Internal Forces [F]

```
invIFM = Inverse[ifm];
tinvIFM = Transpose[invIFM];
jd = Take[tinvIFM, 3 * noden - rest];
tdis = jd.F.indFrcs;
freet2 = Table[{0}, {i, 1, 3 * noden}];
Do[
 freet2[[3 i - 2, 1]] = freet[[i, 2]];
 freet2[[3 i - 1, 1]] = freet[[i, 3]];
 freet2[[3 i, 1]] = freet[[i, 4]];
 , {i, 1, noden}]
UVdis = Table[0., {i, 1, 3 * noden}];
Do[
 {k = 0;
  Do[
   If[freet2[[j, 1]] == 0, {k = k + 1, If[k == i, UVdis[[j]] = tdis[[i]]]}]
         , {j, 1, 3 * noden}];
 }
 , {i, 1, 3 * noden - rest}]
displa = Partition[Flatten[UVdis], 3];
Disp = {};
Disp = {};
Do[
  Dis = (displa[[i, 1]]
         displa[[i, 2]]
         displa[[i, 3]]);
  AppendTo[Disp, Dis],
  {i, 1, noden}];
Do[
 Print["node ", i, "  ", "displacements =", " ", MatrixForm[Disp[[i]]]],
 {i, 1, noden}]
```

Figure 162: Computer Codes for Solving Nodal Displacements

```
mend = 0;
endfrcs = {};
Do[
  mend = mend + 1;
  cm1 = 3 * mend - 2;
  cm2 = 3 * mend - 1;
  cm3 = 3 * mend;
  endfrc = bmem[[i]].(indFrcs[[cm1, 1]]
                      indFrcs[[cm2, 1]]
                      indFrcs[[cm3, 1]]) + Qfmem[[i]];
  AppendTo[endfrcs, endfrc],
  {i, 1, m}];
rmend = 0;
glbendfrcs = {};
Do[
  rmend = rmend + 1;
  cm1 = 3 * rmend - 2;
  cm2 = 3 * rmend - 1;
  cm3 = 3 * rmend;
  glbendfrc = bgmem[[i]].(indFrcs[[cm1, 1]]
                          indFrcs[[cm2, 1]]
                          indFrcs[[cm3, 1]]) + Ffmem[[i]];
  AppendTo[glbendfrcs, glbendfrc],
  {i, 1, m}];
Do[
 Print["member ", i, "   ", "endforces", "   ", MatrixForm[endfrcs[[i]]]],
 {i, 1, m}]
Do[
 Print["member ", i, "   ", "glbendforces", "   ", MatrixForm[glbendfrcs[[i]]]],
 {i, 1, m}]
```

Figure 163: Computer Codes for finding Member End Forces

```
rmmi = 0;
Do[
 node1 = inc[[i, 2]];
 node2 = inc[[i, 3]];
 k1 = 3 * node1 - 2;
 k2 = 3 * node1 - 1;
 k3 = 3 * node1;
 rmmi = rmmi + 1;
 c1 = 3 * rmmi - 2;
 c2 = 3 * rmmi - 1;
 c3 = 3 * rmmi;
 k4 = 3 * node2 - 2;
 k5 = 3 * node2 - 1;
 k6 = 3 * node2;
 kc1 = dof[[k1]];
 kc2 = dof[[k2]];
 kc3 = dof[[k3]];
 kc4 = dof[[k4]];
 kc5 = dof[[k5]];
 kc6 = dof[[k6]];
```

Figure 164: Computer Codes for finding Reactions

```
If[kc1 == 0, Print["reaction x=  ", glbendfrcs[[i]][[1, 1]]]];
If[kc2 == 0, Print["reaction y=  ", glbendfrcs[[i]][[2, 1]]]];
If[kc3 == 0, Print["moment  z=  ", glbendfrcs[[i]][[3, 1]]]];
If[kc4 == 0, Print["reaction x=  ", glbendfrcs[[i]][[4, 1]]]];
If[kc5 == 0, Print["reaction y=  ", glbendfrcs[[i]][[5, 1]]]];
If[kc6 == 0, Print["moment  z=  ", glbendfrcs[[i]][[6, 1]]]];
If[kc1 == 0, Print["member  ", i, " react x  ", node1]];
If[kc2 == 0, Print["member  ", i, " react y  ", node1]];
If[kc3 == 0, Print["member  ", i, " react z  ", node1]];
If[kc4 == 0, Print["member  ", i, " react x  ", node2]];
If[kc5 == 0, Print["member  ", i, " react y  ", node2]];
If[kc6 == 0, Print["member  ", i, " react z  ", node2]],
{i, 1, m}]
```

Figure 165: Computer Codes for finding Reactions (continued)

```
vvfunc = {};
mmfunc = {};
mexpL = {};
ndf = Flatten[endfrcs];
endfrcsT = Partition[ndf, 6];
Do[q1 = endfrcsT[[i, 1]];
 nn = q1;
 vv = -ω[[i]] × x + endfrcsT[[i, 2]];
 axx = endfrcsT[[i, 1]];
 AppendTo[vvfunc, vv];
 mm = -ω[[i]] × x^2/2 + endfrcsT[[i, 2]] × x - endfrcsT[[i, 3]];
 AppendTo[mmfunc, mm];
 mexp = Exponent[mm, x];
 AppendTo[mexpL, mexp];
 axx = Plot[nn, {x, 0, Lmem[[i]]}, ImageSize → 250, Filling → Axis, PlotStyle → Directive[Blue, Thick], Frame → True,
   GridLines → Automatic, GridLinesStyle → Directive[Orange, Dashed], Background → Lighter[Yellow]];
 If[memθ[[i]] ≠ 0., axxr = Rotate[axx, memθ[[i]]]];
 Print["AFD......member  ", i];
 Print["Axial Force Function  N= ", Chop[nn]];
 If[memθ[[i]] ≠ 0., Print[axxr]];
 If[memθ[[i]] == 0., Print[axx]];
 sh = Plot[vv, {x, 0, Lmem[[i]]}, ImageSize → 250, Filling → Axis, PlotStyle → Directive[Blue, Thick], Frame → True,
   GridLines → Automatic, GridLinesStyle → Directive[Orange, Dashed], Background → Lighter[Yellow]];
```

Figure 166: Computer Codes for Plotting Diagrams

```
If[memθ[[i]] ≠ 0., shr = Rotate[sh, memθ[[i]]]];
Print["SFD......member  ", i];
Print["Shear Function  V= ", Chop[vv]];
If[memθ[[i]] ≠ 0., Print[shr]];
If[memθ[[i]] == 0., Print[sh]];
mh = Plot[mm, {x, 0, Lmem[[i]]}, ImageSize → 250, Filling → Axis, PlotStyle → Directive[Blue, Thick], Frame → True,
  GridLines → Automatic, GridLinesStyle → Directive[Orange, Dashed], Background → Lighter[Yellow]];
If[memθ[[i]] ≠ 0., mhr = Rotate[mh, memθ[[i]]]];
Print["BMD......member  ", i];
Print["Moment Function  M= ", Chop[mm]];
If[memθ[[i]] ≠ 0., Print[mhr]];
If[memθ[[i]] == 0., Print[mh]];
Print["***************************"];
Print["                            "],
{i, 1, m}]
```

Figure 167: Computer Codes for Plotting Diagrams (continued)

b) Computer Codes for Singular Value Decomposition Method:

```
apinv = (Inverse[(S.Transpose[S])]).S;
sa = IdentityMatrix[3 × m] - (Transpose[S].apinv);
{u, w, v} = SingularValueDecomposition[sa];
Print["  u =   "  MatrixForm[u]]
Print["  w =   "  MatrixForm[w]]
Print["  v =   "  MatrixForm[v]]
```

Figure 168: Computer Codes for finding Singular Value Decomposition

```
c2 = Chop[Inverse[u].sa];
{row, col} = Dimensions[S];
c1 = Take[c2, col - row, col];
cc = c1.F;
Print["cc = ", MatrixForm[cc]]
```

Figure 169: Computer Codes for finding Compatibility Conditions [CC]

The computer codes for other steps are similar as Integrated Force Method via Null Space.

c) Computer Codes for Dual Integrated Force Method:

```
K = S.Inverse[F].Transpose[S];
Print["K = ", MatrixForm[K]]
MatrixPlot[K, FrameTicks → None, Mesh → True, MaxPlotPoints → Infinity, ImageSize → 450]
```

Figure 170: Computer Codes for finding Global Stiffness Matrix [K]

```
dsiplacements = LinearSolve[K, Pfinal];
Print["displacemetns = ", MatrixForm[dsiplacements]]
```

Figure 171: Computer Codes for finding Displacements

The computer codes for other steps are similar as Integrated Force Method via Null Space and Integrated Force Method via Singular Value Decomposition.