

Java Applets for Analysis of Trusses, Beams and Frames

by

Robert Schottler

Thesis submitted to the Faculty of
Virginia Polytechnic Institute and State University
in partial fulfillment of the degree of

MASTER OF SCIENCE

in

Civil and Environmental Engineering

Approved:

Kamal B. Rojiani (Chair)

Rakesh K. Kapania

Tommy Cousins

May, 2004

Blacksburg, Virginia

Java Applets for Analysis of Trusses, Beams and Frames

by

Robert Schottler

Committee Chairman: Dr. Kamal B. Rojiani

Charles E. Via, Jr. Department of Civil and Environmental Engineering
Virginia Polytechnic Institute and State University

Abstract

Java applets are developed to assist in the learning of basic structural analysis concepts. In order for these programs to be easily available over the Internet, they are written in the object-oriented Java programming language. The Java programs known as applets are embedded in HTML documents. The HTML documents, part of a series of instructional units, present the topics demonstrated by the applets. The applets include truss and frame determinacy applets; a three-hinged arch bridge applet; determinate and indeterminate truss analysis applets; determinate and indeterminate frame analysis applets and an influence line analysis applet. These programs are available to any student or instructor with Internet access. The applets provide good examples of the application of object-oriented programming and the development of software for a graphical user interface. They also serve as excellent tools that facilitate the understanding of structural engineering concepts utilizing a medium that allows independent learning at an individual pace.

Acknowledgements

I wish to express my sincere gratitude to my advisor Dr. Kamal Rojiani. Dr. Rojiani has freely offered his support and guidance throughout this process. For this I am truly grateful. I would also like to thank Dr. Rakesh Kapania and Dr. Tommy Cousins for serving as committee members. Your contributions are greatly appreciated. And thanks to the National Science Foundation for funding this project.

To my parents, my loving wife Sarah and daughter Caroline, I could not have completed this without your continued encouragement and fortitude.

Table of Contents

Acknowledgements	iii
Table of Contents	iv
List of Figures.....	vii
List of Tables	x
Chapter 1 Introduction.....	1
1.1 Introduction.....	1
1.2 Purpose.....	2
1.3 Organization.....	2
Chapter 2 An Overview of Object Oriented Programming.....	3
2.1 Introduction.....	3
2.2 Procedural Programming	3
2.3 Object Oriented Programming.....	3
2.3.1 Classes and Objects.....	4
2.3.2 Characteristics of Object Oriented Programming.....	4
2.3.3 Advantages and Disadvantages of Object Oriented Programming.....	5
2.4 Java Programming Language.....	6
2.5 Applications of Java Programming in Structural Engineering	6
Chapter 3 Determinacy Applets	9
3.1 Introduction	9
3.2 Truss Determinacy Applet.....	9
3.3 Frame Determinacy Applet.....	11
3.4 Three-Hinged Arch Bridge Applet	12
Chapter 4 Truss Analysis Applets.....	14

4.1	Introduction.....	14
4.2	Determinate Truss Analysis Applet.....	14
4.2.1	Analysis Procedure.....	14
4.2.2	Determinate Truss Analysis Applet Interface.....	16
4.2.3	Determinate Truss Analysis Applet Classes.....	17
4.2.4	Verification of the Determinate Truss Analysis Applet.....	19
4.3	Indeterminate Truss Analysis Applet.....	24
4.3.1	Stiffness Analysis.....	24
4.3.2	Analysis Procedure.....	27
4.3.3	Indeterminate Truss Analysis Applet Interface.....	29
4.3.4	Indeterminate Truss Analysis Applet Classes.....	29
4.3.5	Verification of the Indeterminate Truss Analysis Applet.....	31
Chapter 5	Frame Analysis Applets.....	36
5.1	Introduction.....	36
5.2	Determinate Frame Analysis Applet.....	36
5.2.1	Determinate Frame Analysis Applet Interface.....	38
5.2.2	Determinate Frame Analysis Applet Classes.....	42
5.2.3	Verification of the Determinate Frame Analysis Applet.....	44
5.3	Indeterminate Frame Analysis Applet.....	47
5.3.1	Indeterminate Frame Analysis Applet Interface.....	52
5.3.2	Indeterminate Frame Analysis Applet Classes.....	53
5.3.3	Verification of the Indeterminate Frame Analysis Applet.....	55
Chapter 6	Influence Line Analysis Applet.....	68
6.1	Introduction.....	68
6.2	Analysis Procedure.....	68
6.3	Influence Line Applet Interface.....	71
6.3.1	Menu Bar.....	72
6.4	Program Classes.....	73

6.4.1	Structural Classes.....	73
6.4.2	Interface Classes	75
6.5	Program Verification	75
6.5.1	Example 1	75
6.5.2	Example 2	77
6.5.3	Example 3	81
6.5.4	Example 4	85
Chapter 7	Summary and Conclusions	89
7.1	Summary	89
7.2	Conclusions.....	90
	References.....	92

List of Figures

Figure 3.1	Truss Determinacy Applet.....	10
Figure 3.2	Frame Determinacy Applet	11
Figure 3.3	Three-Hinged Arch Bridge Applet.....	13
Figure 4.1	Truss Element.....	15
Figure 4.2	Determinate Truss Applet.....	16
Figure 4.3	Set Options Class.....	17
Figure 4.4	Determinate Truss Example 1	20
Figure 4.5	Results from Determinate Truss Analysis Applet for Example 1	21
Figure 4.6	Determinate Truss Example 2	22
Figure 4.7	Results from Determinate Truss Analysis Applet for Example 2	23
Figure 4.8	Local Member Deformations	25
Figure 4.9	Global Member End Forces.....	26
Figure 4.10	Global Member Deformations.....	27
Figure 4.11	Indeterminate Truss Frame.....	29
Figure 4.12	Indeterminate Truss Example 1	32
Figure 4.13	Results from Indeterminate Truss Analysis Applet for Example 1.....	33
Figure 4.14	Geometry and Loads for the Indeterminate Truss of Example 2 from Fleming (97)	34
Figure 4.15	Results from Indeterminate Truss Analysis Applet for Example 2.....	35

Figure 5.1	Local Member End Loads	37
Figure 5.2	Determinate Frame Program Interface	39
Figure 5.3	Uniform Load Dialogbox	40
Figure 5.4	Partial Uniform Load Dialogbox.....	40
Figure 5.5	Intermediate Moment Dialogbox	41
Figure 5.6	Intermediate Moment Dialogbox	41
Figure 5.7	Concentrated Load Dialogbox.....	42
Figure 5.8	Change Load Dialogbox.....	42
Figure 5.9	Determinate Frame Example.....	45
Figure 5.10	Results for Determinate Frame Example from Applet	46
Figure 5.11	Local Member Forces for a Plane Frame Member.....	47
Figure 5.12	Local Member Deformations for a Plane Frame Member	48
Figure 5.13	Global Member End Loads for a Plane Frame Member	50
Figure 5.14	Global Member Deformations for a Plane Frame Member.....	51
Figure 5.15	Indeterminate Frame Analysis Applet Interface.....	53
Figure 5.16	Example 1 – Frame from Fleming (1997).....	55
Figure 5.17	Example 1 – Frame Applet Results	56
Figure 5.18	Example 2 – Frame from Hibbeler (1995)	58
Figure 5.19	Example 2 – Frame Applet Results	59
Figure 5.20	Example 3 –Beam from Holzer (1998)	61
Figure 5.21	Example 3 – Frame Applet Results	62
Figure 5.22	Indeterminate Frame Example 4	63
Figure 5.23	Example 4 – Applet Results	64

Figure 6.1	Forces on a Typical Beam Element.....	69
Figure 6.2	Fixed End Reactions for a Unit Concentrated Load.....	70
Figure 6.3	Influence Line Applet.....	71
Figure 6.4	Draw Influence Line DialogBox	72
Figure 6.5	Scale/Units DialogBox	73
Figure 6.6	Influence Line Output.....	73
Figure 6.7	Determinate Beam (from Hibbeler, 1985).....	75
Figure 6.8	Influence Line for Shear Force for the Determinate Beam of Example 1....	76
Figure 6.9	Influence Line Ordinates for Shear Force for the Determinate Beam Example 1	76
Figure 6.10	Influence Line for the Moment at 0.4 Times the Length of the First Span for the Continuous Beam With Three Equal Spans	77
Figure 6.11	Moment Influence Ordinates at 0.4 Times the Length of the First Span for the Continuous Beam of Example 2	78
Figure 6.12	Moment Influence Ordinates at 0.3 Times the Length of the Middle Span for the Continuous Beam of Example 2.....	79
Figure 6.13	Three Span Beam, with a 1.2 Span Ratio	81
Figure 6.14	Output from Influence Line Applet for Shear at 0.4 Times the Length of the First Span $V(104)$	83
Figure 6.15	Output from Influence Line Applet for Moment at 0.5 Times the Length of the Second Span $M(205)$	84
Figure 6.16	Four Span Beam	85
Figure 6.17	Results from Influence Line Applet for Reaction at the First Support for Four Span Beam Example.....	87
Figure 6.18	Results from Influence Line Applet for Reaction at the Fourth Support for Four Span Beam Example	88

List of Tables

Table 4.1	Structural Classes Contained in the Determinate Truss Analysis Applet.....	18
Table 4.2	Methods in the Analysis Class of the Determinate Truss Analysis Applet.....	18
Table 4.3	Example 1 – Member Forces from Hibbeler (1995).....	21
Table 4.4	Example 2 – Member Forces from Fleming (1997)	23
Table 4.5	Structure Classes Contained in the Indeterminate Truss Analysis Applet ...	30
Table 4.6	Methods in the Analysis Class of the Indeterminate Truss Analysis Applet.....	30
Table 4.7	Member Forces and Reactions from Holzer (1998) for the Truss of Example 1.....	33
Table 4.8	Analysis Results for Indeterminate Truss of Example 2 from Fleming (1997)	35
Table 5.1	Structural Classes Contained in the Determinate Frame Analysis Applet ...	43
Table 5.2	Methods in the Analysis Class of the Determinate Frame Analysis Applet.....	43
Table 5.3	Results for Determinate Frame Example from RISA-2D.....	46
Table 5.4	Structure Classes Contained in the Indeterminate Frame Analysis Applet.....	54
Table 5.5	Methods in the Analysis Class of the Indeterminate Frame Analysis Applet	54
Table 5.6	Example 1 - Frame Results from Fleming (1997).....	57
Table 5.7	Example 2 - Frame Results from Hibbeler (1995)	60
Table 5.8	Example 3 - Beam Results from Holzer (1998)	62
Table 5.9	Example 4 – Applet and RISA-2D Axial Force Results	65
Table 5.10	Example 4 - Applet and RISA-2D Shear Force Results.....	66

Table 5.11	Example 4 – Applet and RISA-2D Moment Results	67
Table 6.1	Structure Classes Contained in the Influence Line Applet	74
Table 6.2	Methods in the Analysis Class of the Influence Line Applet	74
Table 6.3	Moment Influence Ordinates for Continuous Beam with Three Equal Spans (Tonias, 1995)	80
Table 6.4	Shear and Moment Ordinates for Three Span Beam with a Span Ratio of 1.2 (Barker, 1997)	82
Table 6.5	Influence Line Ordinates for Support Reactions for Four Span Beam	86

Chapter 1

Introduction

1.1 Introduction

Procedural Programming (PP) has been the methodology of choice for the development of most engineering software. However, there has been a recent interest in developing engineering software using the Object Oriented Programming (OOP) methodology. This is due to the fact that OOP has significant advantages over procedural programming.

One advantage of OPP is that it uses single entities called objects, which combine data and functions. Objects represent real life objects. For instance, in the programs written for this project, objects such as a joint, a support, and a member were developed. In large engineering programs using PP, the program can become quite complex with separate entities. With OPP, the program is easier to understand and manage. Another advantage is that OPP possesses characteristics such as inheritance, abstraction, and encapsulation. These characteristics allow programs to be reusable and easier to maintain. Furthermore, most OPP languages provide class libraries that reduce the time and effort of developing applications.

In 1991, Sun Microsystems developed an object oriented language, Java, to be used for household electronic devices [Sun, 2003]. The primary reason for Java being developed, and now its main advantage over other programming languages, was platform independence. In other words, a Java program can be written on one computer and run on a different one without any modifications.

Java's platform independence made it desirable for use on the Internet. Thus, Java was modified for this purpose. Java programs can be classified into two types, applets and applications. Applets are Java programs that are embedded in HTML documents, and can

be downloaded over the Internet that run inside an Internet browser. Java applications are stand-alone programs written in Java that can run on a computer without any browser support.

1.2 Purpose

The purpose of this project was to develop several Java applets to assist in the teaching of basic structural analysis concepts. These applets include determinacy of structures, analysis of determinate and indeterminate trusses and frames, and influence line analysis for indeterminate beams. These applets were then placed on a web site so they are accessible to anyone with Internet capability.

1.3 Organization

In Chapter 2, an overview of object oriented programming methodology is presented. Chapter 3 describes three simple applets dealing with the determinacy of structures. The first applet is Truss Determinacy, the second applet is Frame Determinacy, and the third applet is a Three-Hinged Arch Bridge. The first two applets serve primarily as a tutorial for determining the determinacy of a structure. The Three-Hinged Arch Bridge Applet determines support reactions for an arch bridge. In Chapter 4, applets for analyzing determinate and indeterminate trusses are presented. In the next chapter, applets for analyzing determinate and indeterminate frames are described. The last applet written draws influence lines for determinate and indeterminate beams and is described in Chapter 6. A summary of the results and conclusions from the project are presented in Chapter 7.

Chapter 2

An Overview of Object Oriented Programming

2.1 Introduction

In this chapter, an overview of object oriented programming is presented. A brief description of the traditionally used procedural programming technique is given. This is followed by an explanation of object oriented programming and its advantages and disadvantages. Then, a description of the objected oriented Java programming language is presented. Finally, a review of the literature on the application of object oriented programming in structural engineering and structural engineering education is presented.

2.2 Procedural Programming

Procedural programming has been the most common programming method. Examples of procedural programming languages are PASCAL, FORTRAN, and C. In procedural programming the data and functions are separate entities within the program. Data variables have to be declared as global in order for functions in the program to have access to these variables. The use of global variables increases the possibility of functions accidentally changing the data. The data and functions of procedural programs do not model real life objects and program development is significantly more difficult. Furthermore, the maintaining of data and functions of a large complex program becomes a difficult task. For smaller programs these disadvantages are not as noticeable. Procedural programming is still widely used.

2.3 Object Oriented Programming

Object oriented programming models real world objects. These objects contain both data members and functions.

2.3.1 Classes and Objects

There are two main concepts in object oriented programming, classes and objects. A class consists of data members and functions. An object is an instance of a class. A class defines the data and the functions used by each object of the class. In other words, the class serves as a template for creating the object.

An example of a class from a structural engineering program that analyzes a truss would be the Member class. The Member class might consist of such data members as the start and end joints of the member, startJoint and endJoint; the cross sectional area of the member, A; the modulus of elasticity of the member, E; the directional cosines of the member, CosineX and CosineY; and the length of the member, L. An example of a function of the class Member is ComputeLength() function which calculates the length of the truss member.

Using the class Member, objects representing truss members could be created. Each object would have its own data members; startJoint, endJoint, A, E, cosineX, cosineY, and L.

Data members are usually declared as being private. Private data members can only be used by functions within the class in which they are defined. Data members, however, can also be declared as protected or public. When data members are declared as protected, functions in a derived class can also have access to the data members. Data members declared as public in a class are accessible to any class.

2.3.2 Characteristics of Object Oriented Programming

Data members and functions are contained within an object and cannot be directly accessed by other functions outside the object. This characteristic of an object hiding its internal workings from the rest of the program is called encapsulation.

An object in an object oriented program must interact with other objects in a clearly defined manner. This is called abstraction. Objects can receive a copy of private data of another object only by means of accessor functions.

In object oriented programming, an original class developed is called the base class. A new class can be created that uses the same features of the base class and extends the base class by adding new features. Deriving a new class from a base class is called inheritance.

Functions in a base class can be redefined by a derived class without changing the name of the function. A function called by a derived object in the program performs a different task depending on the type of object. This is called Polymorphism.

2.3.3 Advantages and Disadvantages of Object Oriented Programming

Having objects that relate to real world objects makes programming complex problems more manageable. The characteristics of object oriented programming provides several advantages. Abstraction reduces the chance of objects accidentally altering data. Inheritance allows for ease in development of programs by using existing classes to develop new classes.

The inheritance and polymorphism characteristics of object oriented programming allow for code reusability. Existing classes can be used as they are or can be modified to solve different problems. Libraries of classes are included in many of the object oriented programming languages. Reusing tested existing classes saves time, effort and increases program reliability.

A disadvantage of object oriented programming is the requirement for more memory and slower execution time. This disadvantage is due to message passing and dynamic linking. However, with the development of faster computers with increased memory at relatively lower costs, this disadvantage is becoming less critical. Instead, more emphasis is being placed on the ease of development i.e. lower labor costs of the programmers, than the cost of the hardware.

2.4 Java Programming Language

The Java programming language is an object oriented language developed by Sun Microsystems [Sun, 2003]. It is modeled after the C++ programming language. Java has all of the advantages of other objected oriented programming language such as reusability of code, extensibility, encapsulation and inheritance. It has most of the features of C++ but does not contain the more complex features, such as pointers, that have been the source of difficulty or frequent bugs in C++.

Java is the programming language of choice for the Internet because it is both platform and operating system independent. Java programs can be embedded in web pages (where they are called applets) and can run on any computer with a Java capable browser. Therefore, it is not necessary to create different versions of an application for different hardware platforms.

2.5 Applications of Java Programming in Structural Engineering

There have been many advances in information technology and educational institutions have worked to utilize these advances. Educational institutions have realized the importance of creating new methods for teaching engineering concepts and have turned to technology to aid in their development. When teaching complex engineering concepts and theories in standard lecture environments, students do not always easily grasp the information being presented. However, when these concepts and theories were presented in a virtual environment and there was interaction with instructive programs, Haque found that a student's understanding of the material was improved [Haque, 2001]. Hence, demonstrating the effectiveness and the need for interactive programs.

With the development of the World Wide Web (WWW), information can be easily accessed through the Internet. The Internet has become a useful tool which provides quick, easy, and relatively inexpensive access to interactive learning. Another advantage

to interactive learning through the Internet from web-based documents is that it allows students to learn at their own pace.

Haque conducted research to create an innovative structural design concept visualization methodology on a web-based interactive virtual environment [Haque, 2001]. He developed a web-based interactive virtual environment for the design of flexural and shear behavior of reinforced concrete beams using Java and Virtual Reality Modeling Languages (VRML). This visual environment used for reinforced beams can be applied to other design concepts to enhance a student's subject visualization and conceptual understanding.

Mishra developed applets to monitor a flagpole [Mishra, 2001]. This purpose of this research was to develop technology to monitor structures under duress, such as during an earthquake, by providing real time information. The applets created for the flagpole perform analysis and obtain information from archived data. One of the analysis applets includes real time stress/ strain limits of the flagpole. Another applet shows a visual representation of the flagpole and its real time deflections.

Jiang created three virtual laboratory modules which educate students on reinforced concrete structures [Jiang, 2002]. These modules are based on applets, which perform the calculations involved in the analysis of reinforced concrete sections. One module allows the user to explore the flexural design of rectangular singly reinforced concrete beams. Another module shows the axial force, moment, curvature relationships for rectangular beam and column sections. The last module explores the relationship of uniaxial stress-strain for confined and unconfined reinforced concrete.

Gao developed a Java-powered virtual laboratory for nonlinear structural dynamic analysis [Gao, 2003]. This visual environment allows users to understand structural dynamic concepts related to designing structures for seismic loads. The number of stories, the floor mass, stiffness, and damping coefficients of each story of the structure can be selected by the user. Features of the program include graphs of the dynamic analysis results and an animation of the virtual building.

Rojiani developed several web based instructional units using Java [Rojiani et al, 2000]. These instructional units, embedded in WWW pages where they are called applets, were developed to assist undergraduate students in the conceptualization of structural mechanics. The applets developed included shear, moment and deflection of beams; computation of section properties of sections built up from standard geometric shapes; and shear center for open and closed section thin-walled tubes. These interactive applets were made accessible to any student with a computer and Internet access.

An overview of object oriented programming was presented in this chapter. The most commonly used procedural programming paradigm was presented. The characteristics of object oriented programming including classes, objects, inheritance, encapsulation, abstraction and polymorphism were briefly discussed. The advantages and disadvantages of object oriented programming was also presented. A brief description of the Java programming language was also presented. In the last section, a review of the application of Java in structural engineering was presented.

Chapter 3

Determinacy Applets

3.1 Introduction

In order to cover the basic concepts of structural analysis, simple applets were written to assist in the understanding of the concepts of determinacy, indeterminacy and stability. The Truss and Frame Determinacy Applets test a student's understanding of these concepts. Each applet consists of a series of interactive examples of structures where the student must classify the structure as unstable, determinate or indeterminate. A third applet for analyzing three-hinged arches was also written to demonstrate how arches transmit loads. All of these applets described in this thesis are available on the Internet at the following website: <http://www.Rojiani.structures1.cce.vt.edu>.

3.2 Truss Determinacy Applet

The Truss Determinacy Applet tests the student's knowledge of the concepts of stability and determinacy of trusses. The truss determinacy applet displays a series of trusses and asks the user to enter the number of joints, members, reactions, and redundants and make a determination as to whether the truss is unstable, determinate, or indeterminate. For each truss, the user can check her answers or display the correct answers (Fig. 3.1).

All the members of a truss are pinned and loads are applied only at the joints. Thus, the members of a truss can only carry axial forces (tension and compression). In the analysis of a truss, the unknowns are the support reactions and the forces in the members (Eq. 3.1). At each joint, the summation of forces in the x and y directions must equal zero. Likewise, the summation of the member's axial forces along the local coordinate system of the member must also equal zero (Eq. 3.2). Therefore, a truss is statically determinate if the sum of the support reactions and the number of members is twice the number of joints in the truss. In other words, the number of unknowns is equal to the number of equilibrium equations in a given truss (Eq. 3.3). If there are more unknowns than equilibrium equations, the truss is statically indeterminate and the number of redundants is the difference between the number of unknowns and the number of

equilibrium equations. On the other hand, if there are more equilibrium equations than unknowns, the truss is unstable.

$$U = 2NM + NR \quad (\text{Eq. 3.1})$$

$$E = 2NJ + NM \quad (\text{Eq. 3.2})$$

$$2NJ = NM + NR \quad (\text{Eq. 3.3})$$

where, NM = number of members
 NR = number of restraints
 NJ = number of joints
 U = number of unknowns
 E = number of equations

If $U - E < 0$, then the truss is unstable
 $U - E = 0$, then the truss is determinate
 $U - E > 0$, then the truss is indeterminate

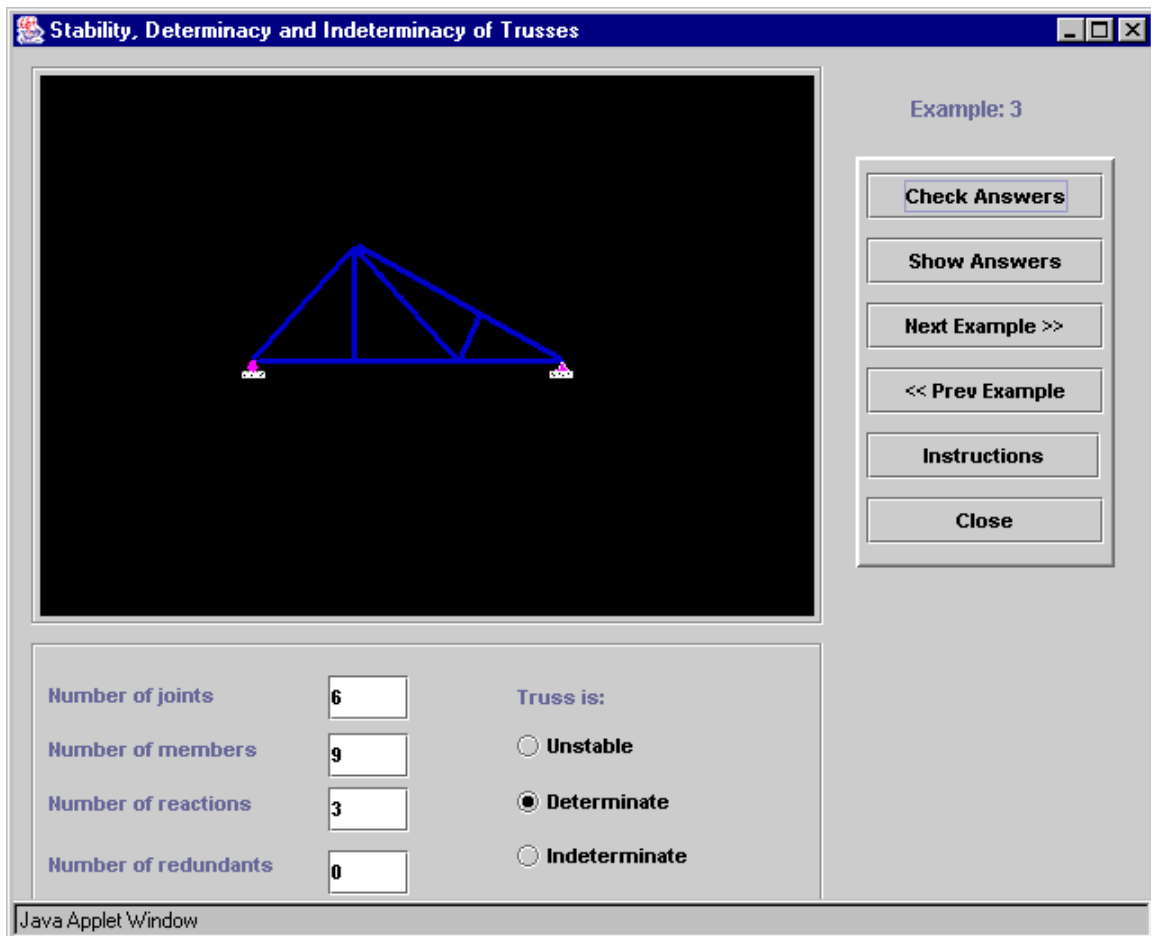


Figure 3.1 Truss Determinacy Applet.

3.3 Frame Determinacy Applet

The Frame Determinacy applet tests the student's knowledge of stability and determinacy of frames. The applet displays a series of frame examples. Users are then asked to enter the number of reactions, equations, members, degrees of freedom, as well as the degree of external, internal, and total indeterminacy (Fig. 3.2). For each of the examples, the user can check her answers or have the applet display the correct answers.

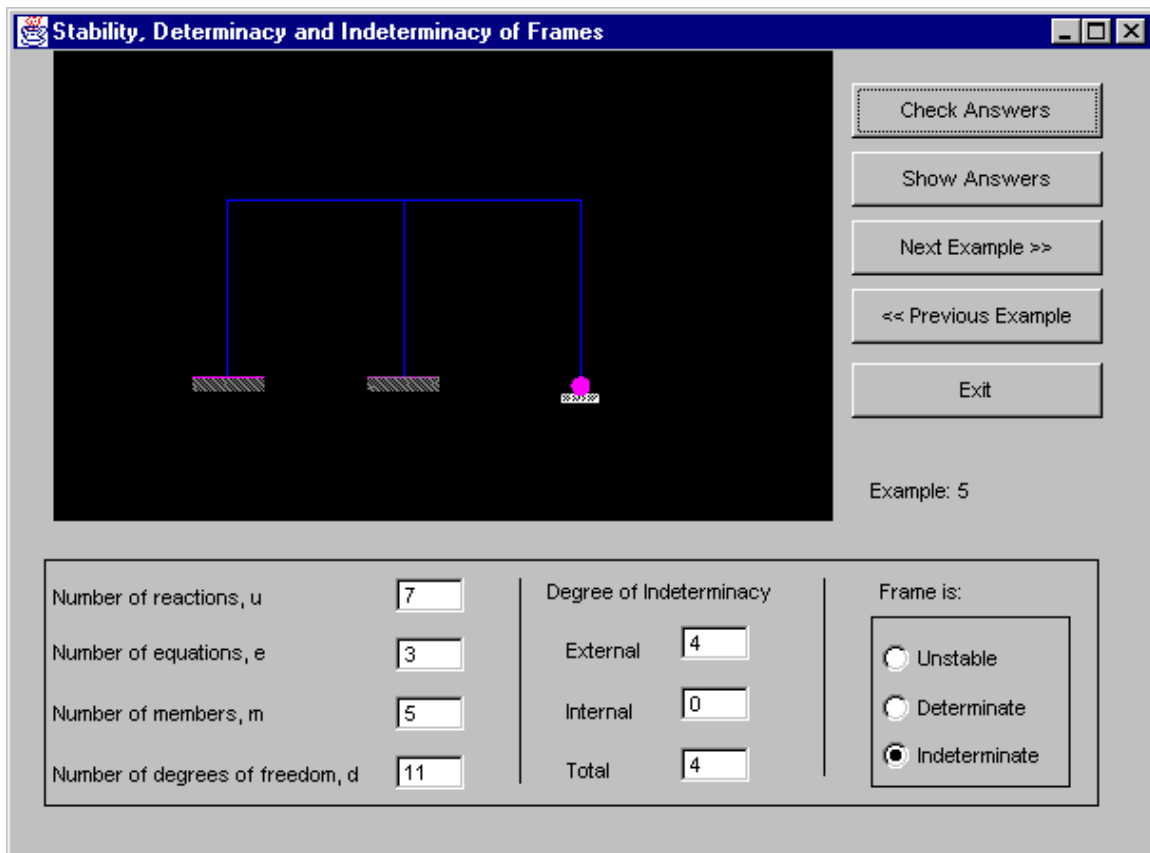


Figure 3.2 Frame Determinacy Applet.

The members of a frame are connected by fixed (moment resisting) joints. Each member of the frame can have three forces: an axial force, a shear force and a bending moment. A frame is statically determinate if the number of support reactions, u , is equal to the number of equilibrium equations, e . The equilibrium equations include the summation of forces in the x and y directions, the summation of moments, and any additional equilibrium equations due to releases (such as a hinge). The degree of external

indeterminacy, i_{ext} , of a frame is the difference between the number of support reactions and the number of equilibrium equations (Eq. 3.4).

$$i_{\text{ext}} = u - e \quad (\text{Eq. 3.4})$$

If $i_{\text{ext}} < 0$, the frame is unstable. If $i_{\text{ext}} = 0$, then the frame is externally determinate. If $i_{\text{ext}} > 0$, the frame is indeterminate.

The total degree of indeterminacy, i_{total} , of the frame is given by $i_{\text{total}} = 3m - d$ where m is the number of members and d is the number of degrees of freedom. A joint without any restraints will have three degrees of freedom. The degree of internal indeterminacy, i_{int} , of the frame is the difference between the degrees of total and external indeterminacy (Eq. 3.5).

$$i_{\text{int}} = i_{\text{total}} - i_{\text{ext}} \quad (\text{Eq. 3.5})$$

The Truss and Frame Determinacy Applets allow the student to experiment with identifying whether a structure is stable, determinate, or indeterminate. These are important concepts in structural analysis. Identifying the structure allows the student to select the appropriate analysis procedure needed to analyze the structure.

3.4 Three-Hinged Arch Bridge Applet

The Three-Hinged Arch applet demonstrates how arches carry loads. The applet computes the horizontal and vertical support reactions of an arch bridge with a hinge located at the crown of the arch (Fig. 3.3). The arch is subject to uniformly distributed load.

The Three-Hinged Arch Bridge Applet is a determinate structure. After the student realizes that the structure is determinate, the horizontal and vertical support reactions can be solved using equilibrium equations.

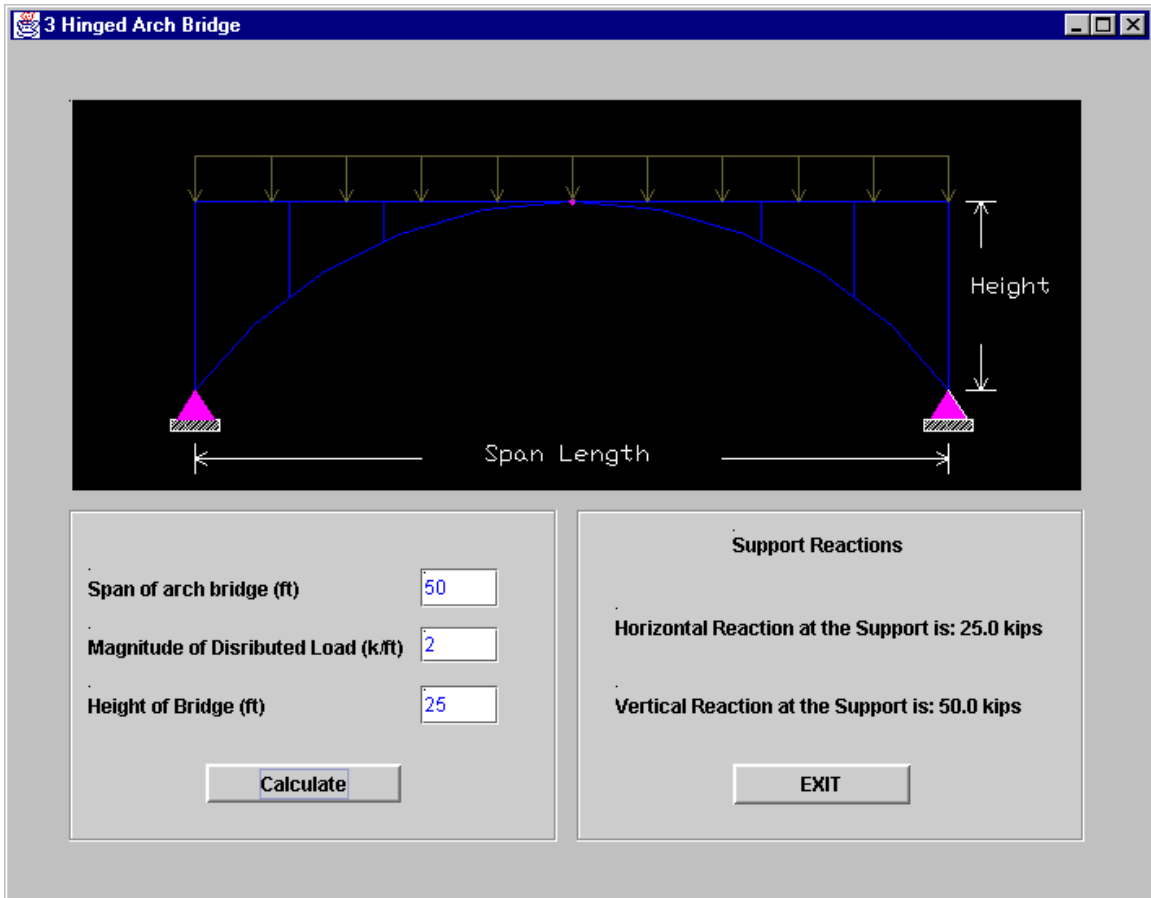


Figure 3.3 Three-Hinged Arch Bridge Applet.

Chapter 4

Truss Analysis Applets

4.1 Introduction

Two applets were developed for the analysis of plane trusses. The first applet computes member forces and support reactions for a statically determinate truss. The basic assumptions made in the analysis of the unknowns are: a) the truss consists of straight members in the same plane; b) the joints of the truss are pinned; c) there are only axial forces in the truss members; d) no moment is transferred between the members and joints; e) and the loads are only applied at the joints [Fleming, 1997]. The second applet is similar to the Determinate Truss Analysis Applet except that it can analyze indeterminate trusses.

4.2 Determinate Truss Analysis Applet

4.2.1 Analysis Procedure

The Determinate Truss Analysis Applet computes support reactions and member forces for a statically determinate truss. A typical truss member is shown in Figure 4.1. The coordinates of the start and end joints of the truss are (X_{n1}, Y_{n1}) and (X_{n2}, Y_{n2}) . The axial forces (S_n) at the ends of a truss member are converted into the global X and Y components, S_{nX} and S_{nY} , as shown in Figure 4.1.

Forces S_{nX} and S_{nY} are calculated by multiplying the axial force of the member, S_n , by the cosine of their corresponding orientation angle (θ_{nX} , θ_{nY}).

$$S_{nX} = S_n \cos \theta_{nX} = S_n C_{nX} \quad (\text{Eq. 4.1})$$

$$S_{nY} = S_n \sin \theta_{nY} = S_n S_{nY} \quad (\text{Eq. 4.2})$$

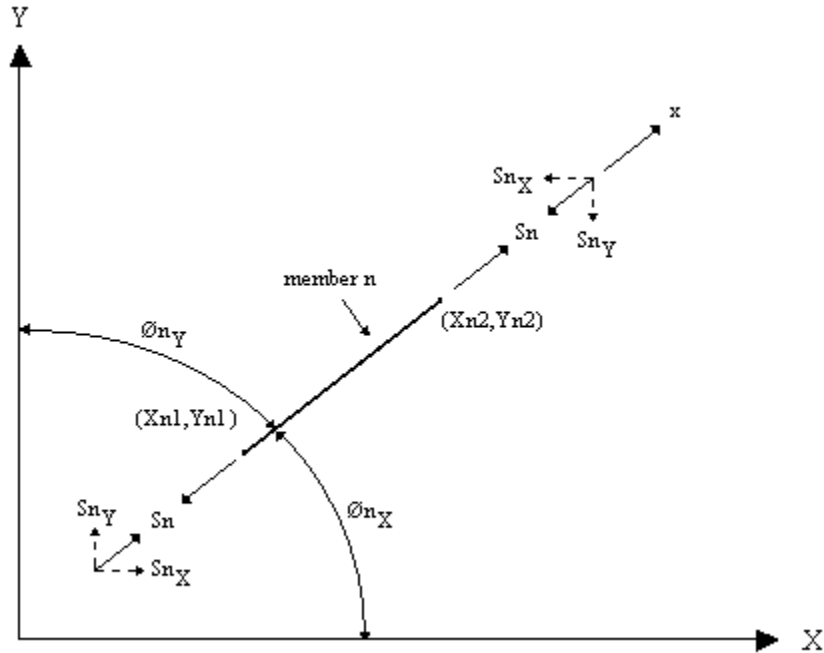


Figure 4.1 Truss Element.

The direction cosines, Cn_X and Cn_Y , are determined from the start and end coordinates:

$$Cn_X = \frac{Xn_2 - Xn_1}{\sqrt{(Xn_2 - Xn_1)^2 + (Yn_2 - Yn_1)^2}} \quad (\text{Eq. 4.3})$$

$$Cn_Y = \frac{Yn_2 - Yn_1}{\sqrt{(Xn_2 - Xn_1)^2 + (Yn_2 - Yn_1)^2}} \quad (\text{Eq. 4.4})$$

For each joint of the truss an equilibrium equation is set up in the joint's global unrestrained X and Y directions. For a joint that is not totally restrained, the directional cosines multiplied by the unknown member forces and the loads acting on the joint are summed. These equilibrium summation equations are then placed into a matrix to be solved.

The unknown member forces are solved using the Gauss-Jordan Elimination Method. After the member forces are determined, the support reactions are obtained by summing

the forces at the restrained joints. The joint loading, if any, is added to the now known member forces multiplied by their corresponding directional cosine.

4.2.2 Determinate Truss Analysis Applet Interface

The main frame of the determinate truss program contains a canvas area, a panel of buttons, and a menu bar as shown in Figure 4.2. As the truss geometry is entered, the truss is displayed in the canvas area.

As seen in Figure 4.2, the menu bar contains a File selection, Calculate selection, and an Options selection. The File Menu selection has two menu items, New and Exit. With the selection of the New item, the truss shown in the canvas is cleared. The selection of the

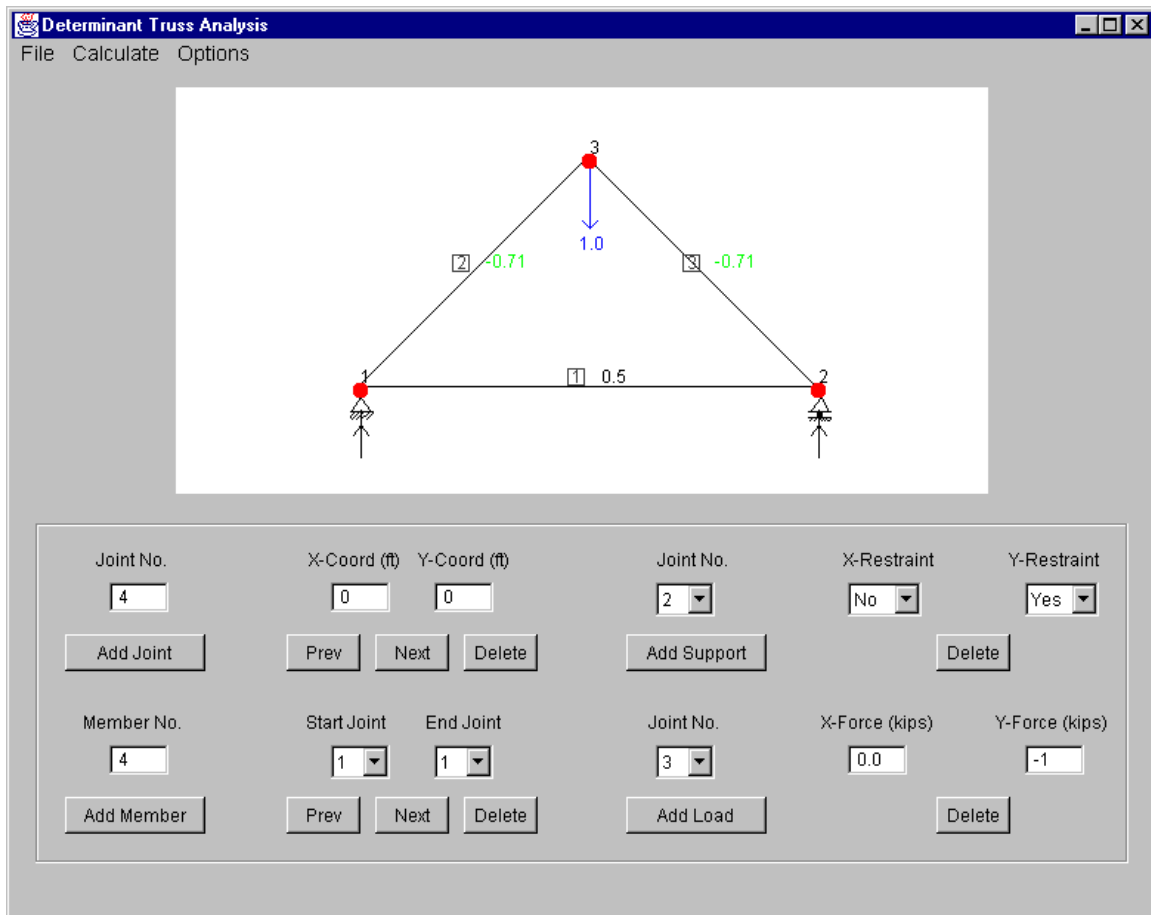


Figure 4.2 Determinate Truss Applet.

Exit menu item results in exiting the entire program. Before the truss geometry and loads are entered, the system of units needs to be indicated. The maximum X and Y coordinates of the problem also need to be set in order for the canvas to properly display the truss geometry. All of these are entered in the Set Options dialog box (see Figure 4.3). After the truss geometry and loading has been defined, the Analyze Truss selection can be chosen under the Calculate menu item to display the results of the analysis.



Figure 4.3 Set Options Class.

4.2.3 Determinate Truss Analysis Applet Classes

The Determinate Truss Analysis Applet contains two types of classes; those that model the structure, in this case a determinate truss, and those that represent the interface of the program. The structural classes that were developed for this applet are shown in Table 4.1. They include the Joint, the Support, the Member, the Load, and the Analysis classes. A list of the methods used in the Analysis class is shown in Table 4.2.

Table 4.1 Structural Classes Contained in the Determinate Truss Analysis Applet.

Class	Description
Analysis	All the analysis for a determinate truss is done here
Member	Represents members in the truss. This class contains data members for storing the start and end coordinates, length, direction cosines, and member forces.
Joint	Represents the joints of the truss. This class contains data members for storing the x and y coordinates of the joint.
Support	Represents the supports of the truss. This class contains data members for storing the directions that are restrained and the different reaction values.
Load	Represents the loads acting at the joints.

Table 4.2 Methods in the Analysis Class of the Determinate Truss Analysis Applet.

Function	Description
CheckStability()	Checks the stability of the determinate truss and displays an error message if the truss is unstable
CalculateTotalRestrains()	Computes the total number of restraints.
AssignEquation()	Sets up the equations for the member forces to be solved.
GenerateCoefficients()	Generates the coefficients in each of the equilibrium equations.
SetUpLoad()	Determines the load vector.
ComputeDirectionCosines()	Computes direction cosines for each member of the truss.
Gauss()	Solves member forces using Gauss elimination.
SolveJointEquations()	Stores member forces solved by Gauss elimination.
SumForces()	Sums the reaction forces from loads at supports with member forces acting on the joints.

The user interface classes consist of the Factor, the IntTextField, the DoubleTextField, the FloatTextField, the ManyLineLabel, the Dialogbox, the OptionParameter, the Option, the Metric, the DeterminatePlaneTrussApplet, and the DetPlaneTrussFrame classes.

As mentioned in the introduction, with object oriented programming, once a class has been developed it can be used when necessary in other programs. The Joint, Support, IntTextField, DoubleTextField, ManyLineLabel and, Dialogbox classes are reused in some of the other applets.

The DeterminatePlaneTrussApplet class is the main applet class, which is derived from the Java Swing JApplet class. The sole purpose of this class, as in the other programs, is to start up the program, and display the main frame of the program. The only differences between this class and other similar classes in other applets in later chapters are, of course, the name and title, and the frame, which it calls.

The DetPlaneTrussFrame class represents the main window frame and is derived from the JFrame class. This class contains all the code needed for the implementation of the menu bar and panel of buttons. It is also responsible for producing the graphics on the canvas.

The Dialogbox and the Option classes are dialog box classes derived from the JDialog class. The Option class contains the code for selecting program options such as units and limits on the truss geometry.

4.2.4 Verification of the Determinate Truss Analysis Applet

Several example trusses were analyzed and the member forces and support values obtained from the applet were verified with results obtained from other sources. In this section, two of these example problems are presented.

Determinate Truss Example 1

The determinate truss shown in Figure 4.4, taken from Hibbeler (1995), was analyzed. The results of the analysis are shown in Figure 4.5. The member forces from Hibbeler (1995) are shown in Table 4.3. When the two sets of results are compared, it can be seen that the member force results obtained from the applet are the same as those given by Hibbeler (1995).

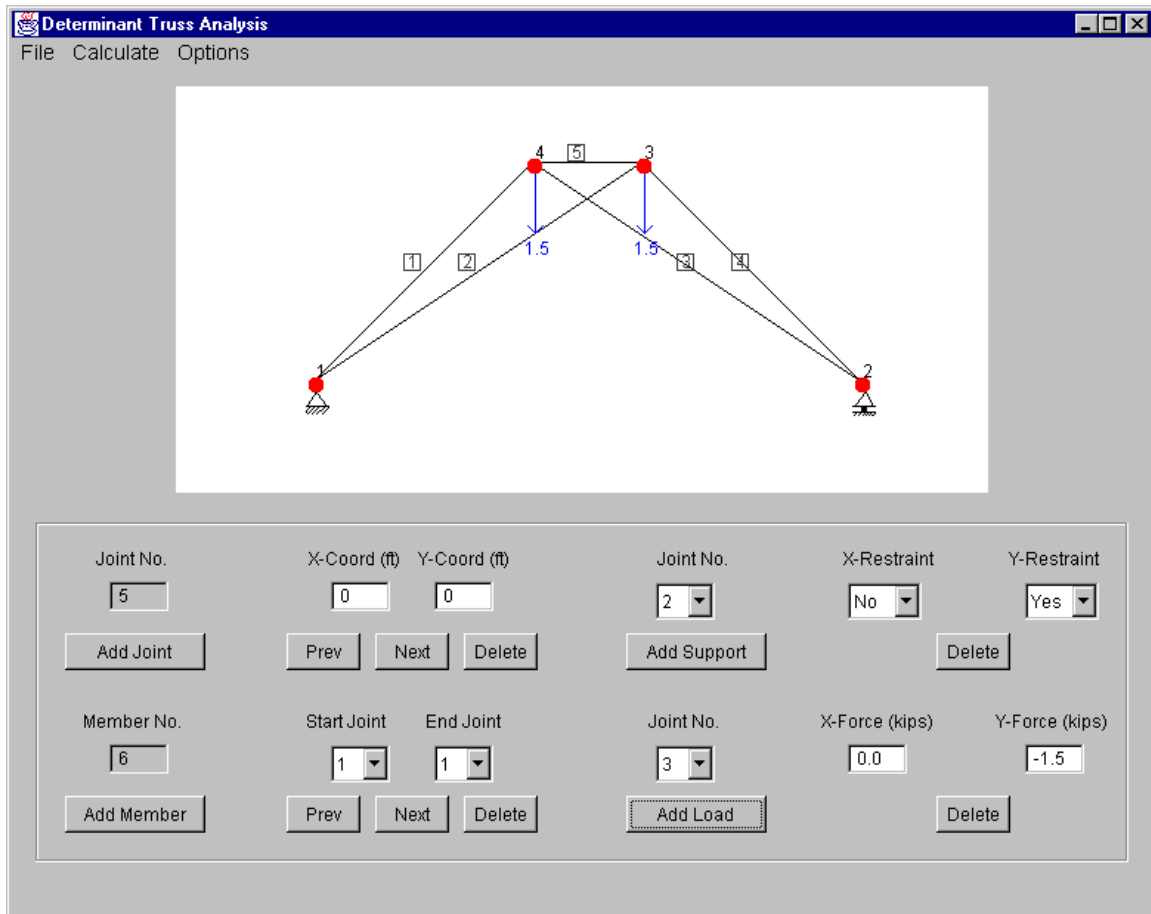


Figure 4.4 Determinate Truss Example 1.

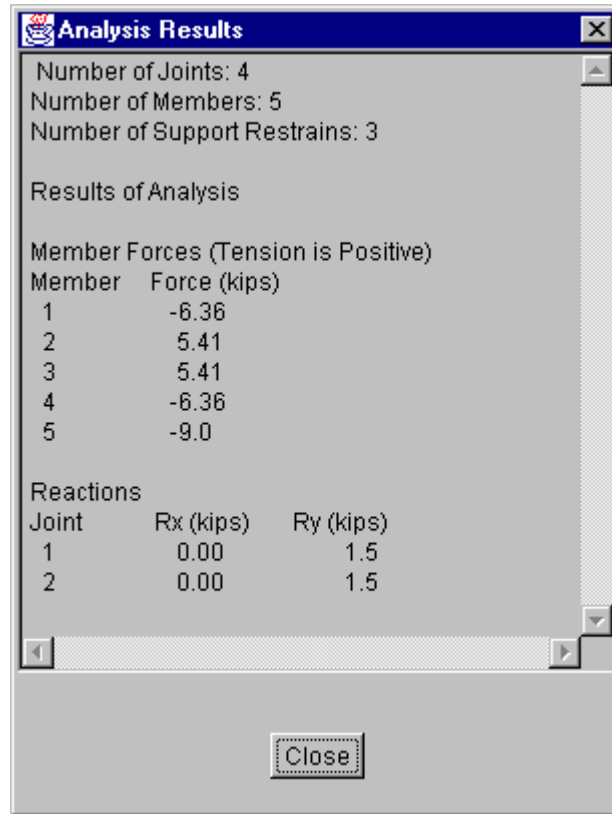


Figure 4.5 Results from Determinate Truss Analysis Applet for Example 1.

Table 4.3 Example 1 – Member Forces from Hibbeler (1995).

Member	Force
1-3	5.41 k (T)
1-4	6.36 k (C)
2-3	6.36 k (C)
2-4	5.41 k (T)
3-4	9.00 k (C)

Determinate Truss Example 2

The second example shown in Figure 4.6 was taken from Fleming (1997). The results obtained from the applet are shown in Figure 4.7. The member forces and support reactions from Fleming (1997) are shown in Table 4.4. From a comparison of Figure 4.7 and Table 4.4 it can be seen that member forces and support reaction computed using the Determinate Truss Analysis applet are the same as those presented in Fleming (1997).

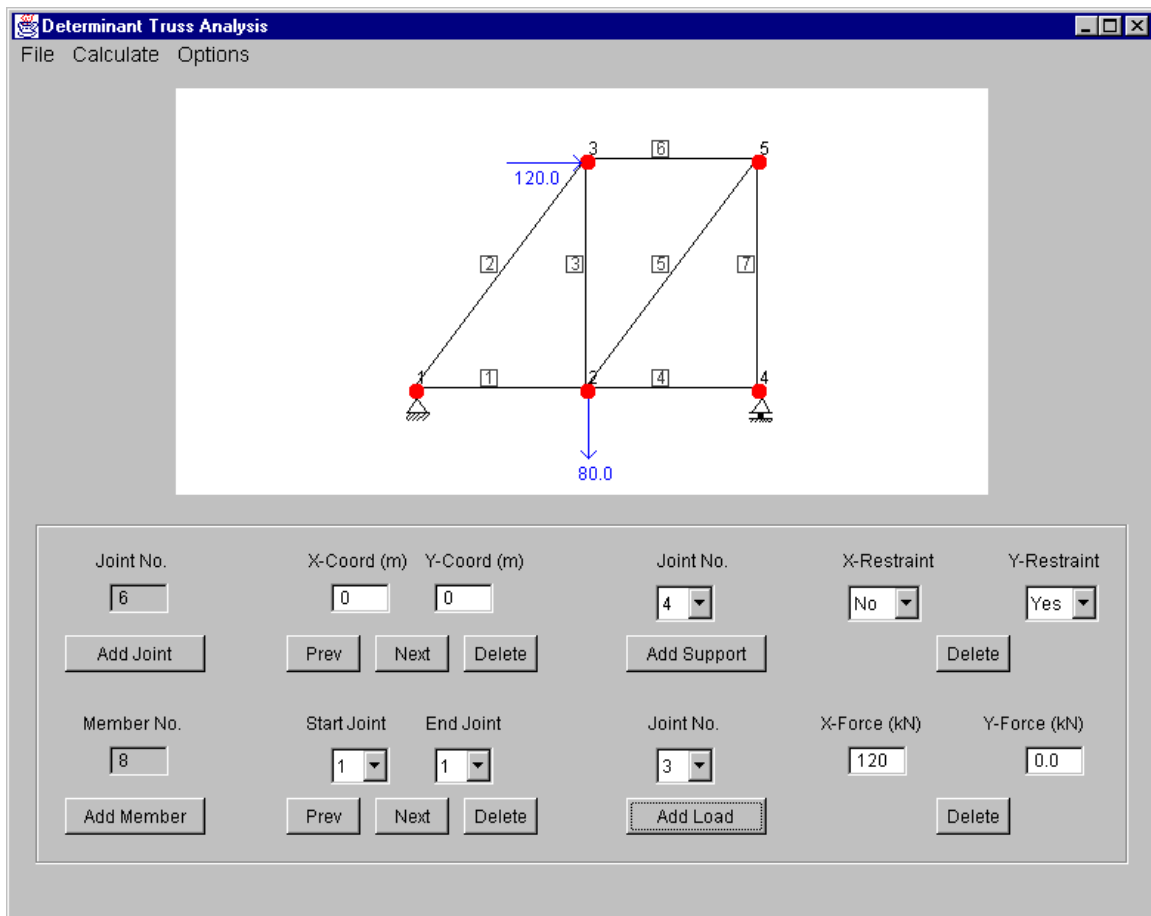


Figure 4.6 Determinate Truss Example 2.

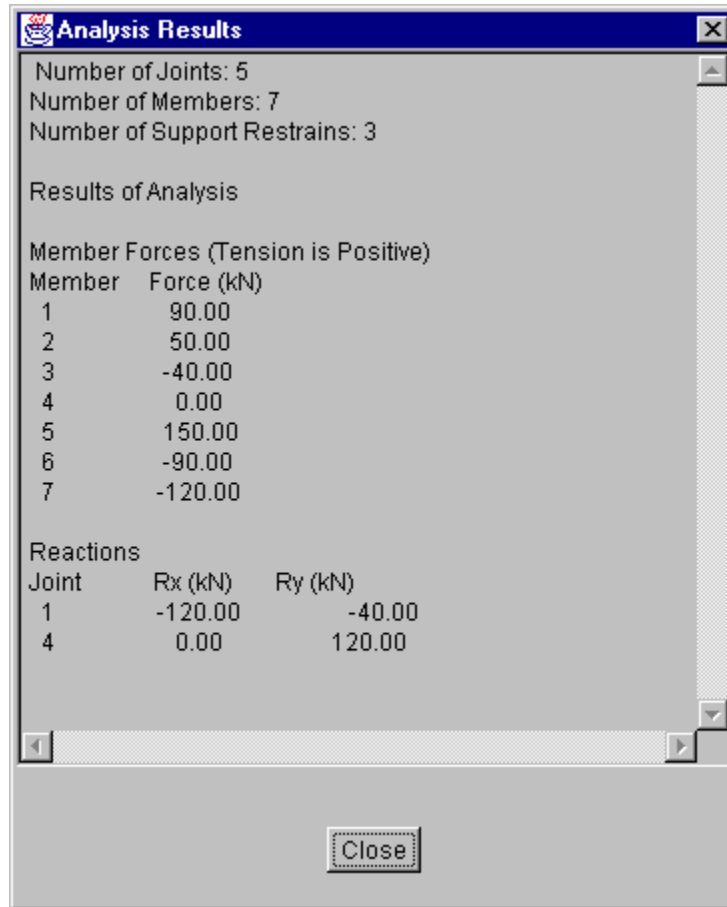


Figure 4.7 Results from Determinate Truss Analysis Applet for Example 2.

Table 4.4 Example 2 – Member Forces from Fleming (1997).

Member	Force
1	90 kN (T)
2	50 kN (T)
3	40 kN (C)
4	0
5	150 kN (T)
6	90 kN (C)
7	120 kN (C)

Reaction	Value
R1X	-120 kN
R1Y	-40 kN
R4Y	120 kN

4.3 Indeterminate Truss Analysis Applet

4.3.1 Stiffness Analysis

The indeterminate truss analysis applet uses the matrix stiffness approach to analysis an indeterminate truss. In the stiffness method, a structure is analyzed by calculating the joint displacements.

As described in section 4.2.1, a member in a truss has an axial force at each end. The local deformations of a truss member, n , are shown in Figure 4.8. The force S_n in a typical member is equal to the member's local deformations $[U]_n$ multiplied by the stiffness matrix $[K_m]_n$ of the member in the local coordinate system.

$$[S]_n = [K_m]_n[U]_n \quad (\text{Eq. 4.5})$$

$$\text{where, } [S]_n = \begin{pmatrix} s(1) \\ s(2) \end{pmatrix}_n \quad (\text{Eq. 4.6})$$

$$[K_m]_n = \begin{pmatrix} \frac{A_x E}{L} & -\frac{A_x E}{L} \\ -\frac{A_x E}{L} & \frac{A_x E}{L} \end{pmatrix}_n \quad (\text{Eq. 4.7})$$

$$\text{and } [U]_n = \begin{pmatrix} u(1) \\ u(2) \end{pmatrix}_n \quad (\text{Eq. 4.8})$$

Here $s(1)$ and $s(2)$ are the forces at the start and end of the member, $u(1)$ and $u(2)$ are the corresponding displacements, and $[K]_m$ is the stiffness matrix of the member.

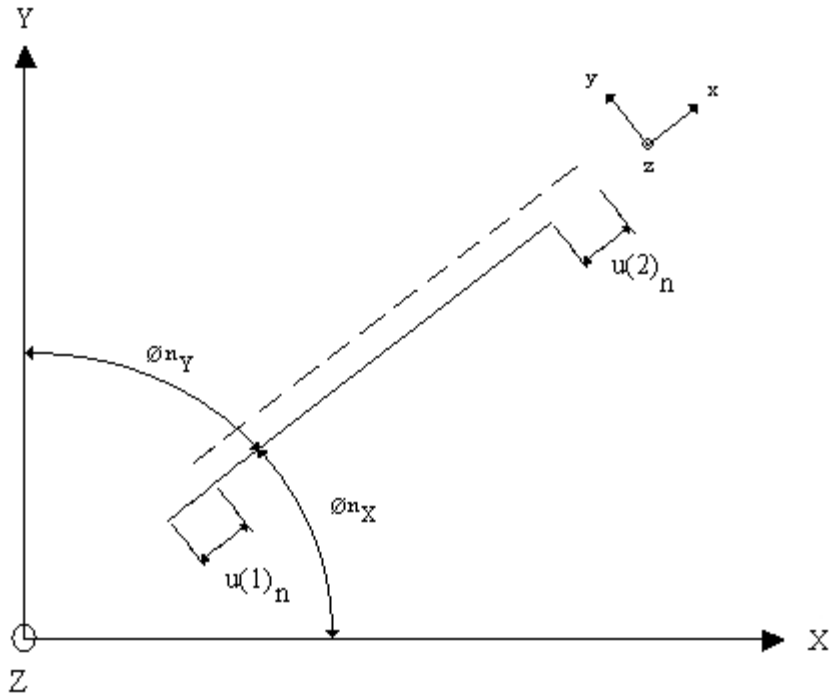


Figure 4.8 – Local Member Deformations.

Member end forces $[Q]_n$ in the global coordinate system for member n are equal to the stiffness matrix $[K]_n$ of the member multiplied by its global member deformations $[D]_n$. For a plane truss, the global load vector has four elements as shown in Figure 4.9. The global member deformations matrix also consists of four elements. Member deformations in the global coordinate system are in shown in Figure 4.10. Thus,

$$[Q]_n = [K]_n [D]_n \quad (\text{Eq. 4.9})$$

where

$$[Q]_n = \begin{pmatrix} q(1) \\ q(2) \\ q(3) \\ q(4) \end{pmatrix}_n \quad (\text{Eq. 4.10})$$

and

$$[D]_n = \begin{pmatrix} d(1) \\ d(2) \\ d(3) \\ d(4) \end{pmatrix}_n \quad (\text{Eq. 4.11})$$

The stiffness matrix of the truss member in the global coordinate system is given by

$$[K]_n = \left(\frac{A_x E}{L} \right)_n \begin{pmatrix} C_X^2 & C_X C_Y & -C_X^2 & -C_X C_Y \\ C_X C_Y & C_Y^2 & -C_X C_Y & -C_Y^2 \\ -C_X^2 & -C_X C_Y & C_X^2 & C_X C_Y \\ -C_X C_Y & -C_Y^2 & C_X C_Y & C_Y^2 \end{pmatrix} \quad (\text{Eq. 4.12})$$

where n = member number, A_x = area of member, E = modulus of elasticity, L = length of member, $C_x = \cos(\theta_{nX})$ and $C_Y = \cos(\theta_{nY})$.

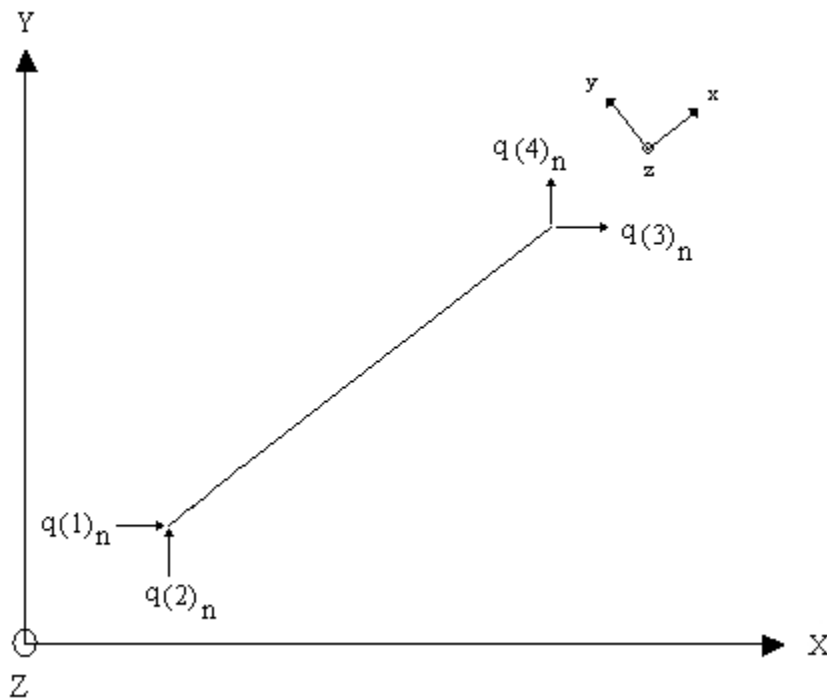


Figure 4.9 – Global Member End Forces.

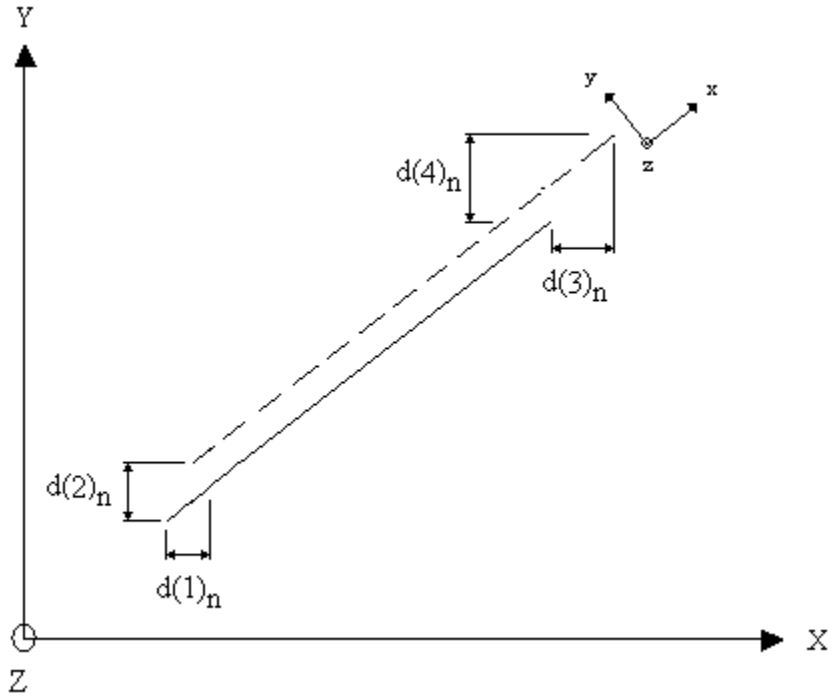


Figure 4.10 – Global Member Deformations.

4.3.2 Analysis Procedure

The first step in the analysis is to assign degrees of freedom. The joints of the truss are checked in ascending order to determine if they are restrained in the global X and Y directions. If the joint is unrestrained in a direction, a degree of freedom number is assigned. The degree of freedom numbers are assigned consecutively for each unrestrained degree of freedom. If the joint is restrained, a zero value is assigned to the corresponding direction.

The global load matrix [Q] for the structure corresponding to the number of degrees of freedom for the truss is obtained. The global stiffness matrix [K] for the structure corresponding to the degrees of freedom is given by

$$[K] = \begin{pmatrix} k(1,1) & k(1,2) & \cdots & k(1,\text{NDOF}) \\ k(2,1) & k(2,2) & \cdots & k(2,\text{NDOF}) \\ \vdots & \vdots & \ddots & \vdots \\ k(\text{NDOF},1) & k(\text{NDOF},2) & \cdots & k(\text{NDOF},\text{NDOF}) \end{pmatrix} \quad (\text{Eq. 4.13})$$

The global stiffness matrix is obtained by adding the contributions of the members in a structure.

$$[K] = \sum_{n=1}^{NM} [K]_n \quad (\text{Eq 4.14})$$

The assembly process involves placing the elements of the global member stiffness matrices into the correct locations of the global stiffness matrix.

The system model for the truss is

$$[Q] = [K] [D] \quad (\text{Eq 4.15})$$

that is, the global stiffness matrix of the structure multiplied by the joint displacements of the structure is equal to the joint loads of the structure. The joint displacements $[D]$ are solved using the Gauss-Jordan Elimination Method.

After the global displacements are determined, member end forces for each truss member are calculated using Equations 4.16.

$$\begin{aligned} s(1)_n &= -A_x E/L ([d(3)_n - d(1)_n]C_x + [d(4)_n - d(2)_n]C_y) \\ s(2)_n &= A_x E/L ([d(3)_n - d(1)_n]C_x + [d(4)_n - d(2)_n]C_y) \end{aligned} \quad (\text{Eqs. 4.16})$$

These equations were obtained by transforming the global displacements to local displacements.

$$\begin{aligned} u(1)_n &= d(1)_n \cos\theta_x + d(2)_n \cos\theta_y \\ u(2)_n &= d(3)_n \cos\theta_x + d(4)_n \cos\theta_y \end{aligned} \quad (\text{Eqs. 4.17})$$

The reactions at the supports are then found by summing forces at joints.

4.3.3 Indeterminate Truss Analysis Applet Interface

The interface for the Indeterminate Truss Analysis Applet is similar to that of the Determinate Truss Analysis Applet described in section 4.2. However, since the analysis of an indeterminate truss requires member properties such as the modulus of elasticity and area, an additional panel is provided for entering these properties. The applet interface is presented in Figure 4.11.

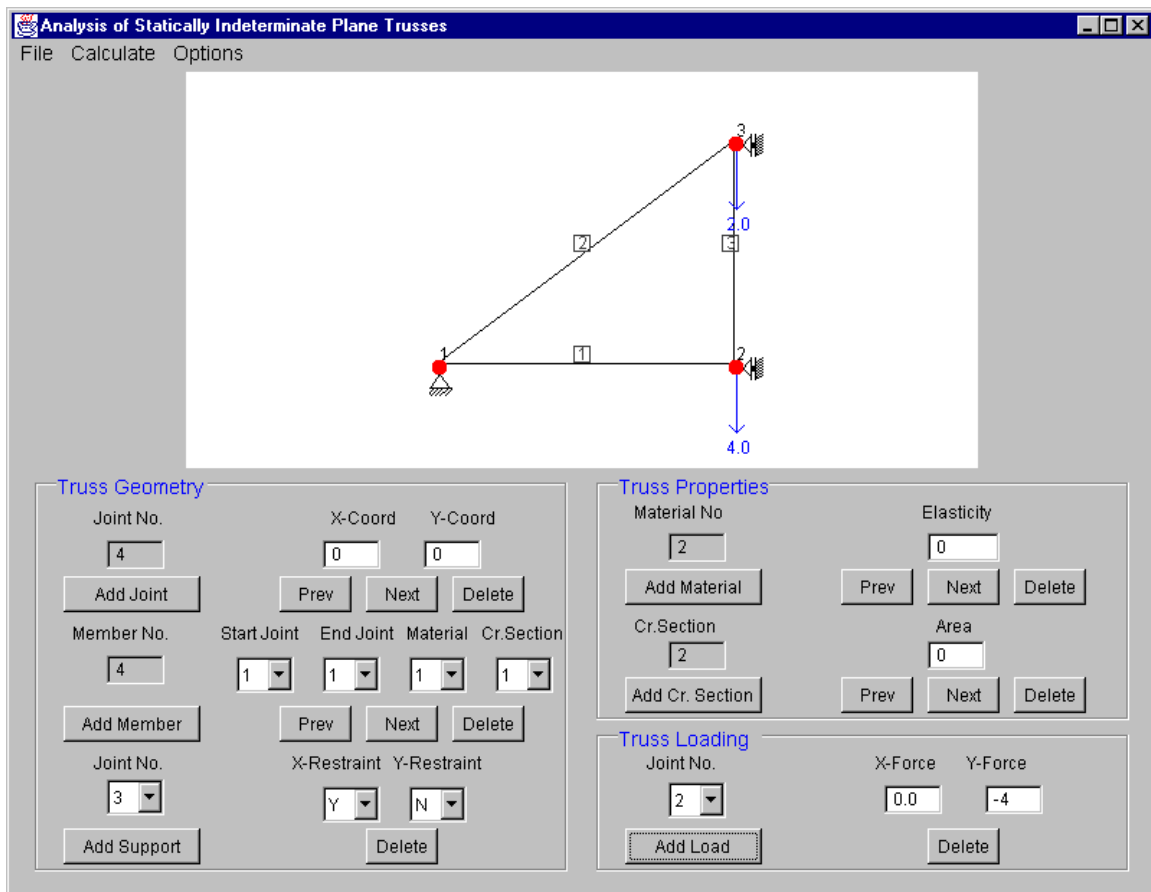


Figure 4.11 Indeterminate Truss Frame.

4.3.4 Indeterminate Truss Analysis Applet Classes

As with the Determinate Truss Analysis Applet, the classes in the Indeterminate Truss Analysis Applet can be divided into structural classes and interface classes.

The structural classes included are the Joint, the Load, the Support and the Member classes. Table 4.5 provides a brief description of the structural classes. An Analysis class was developed specifically for this program using the analysis procedure described earlier. The methods contained in the Analysis class are shown in Table 4.6.

Table 4.5 Structural Classes Contained in the Indeterminate Truss Analysis Applet.

Class	Description
Analysis	The analysis for an indeterminate truss is done in this class.
Member	Represents members in the truss. This class contains data members for storing information about the start and end coordinates, length, direction cosines, modulus of elasticity, area of the member, and member forces.
Joint	Represents joints in a truss. This class contains data members for storing the x and y coordinates of the joint.
Support	Represents supports in a truss. This class contains data members for storing the directions that are restrained, the reaction values, and joint displacements.
Load	Represents loads acting at the joints.

Table 4.6 Methods in the Analysis Class of the Indeterminate Truss Analysis Applet.

Function	Description
CheckStability()	Checks the stability of the truss and displays an error message if the truss is unstable.
CalculateTotalRestrains()	Computes the total number of restraints.
AssignDegreesOfFreedom()	Assigns degrees of freedom to the unrestrained joint displacements.
initializeReactions()	Initializes support reactions. Joint loads at the supports are added to the support reactions.
SetUpLoad(),	Sets up the loads in a matrix.
initializestiffMatrix()	Initializes the global stiffness matrix.
GlobalStiffnessMatrix()	Computes the stiffness matrix of the truss.
Gauss()	Solves for member forces using Gauss elimination.
storeDisplacements()	Stores the joint displacements computed using Gauss elimination.
ComputeMemberForces()	Computes member forces using the solved joint displacements. Also computes support reactions by summing forces at the joint.

The interface classes contained in the applet are the Factor, the IntTextField, the DoubleTextField, the FloatTextField, the ManyLineLabel, the Dialogbox, the OptionParameter, the Option, and the Metric classes. Interface classes were specifically developed for this program are the IndeterminatePlaneTrussApplet and the IndetPlaneTrussFrame classes.

The IndeterminatePlaneTrussApplet class is the main applet class and is instantiated from the web page. This class in turn creates an instance of the IndetPlaneTrussFrame class.

The IndetPlaneTrussFrame class represents the window of the program. This class contains the code needed for the interaction of the menu bar and panel of buttons and for displaying the graphics on the canvas. Code was added to change the interface for entering an indeterminate truss. A panel of buttons for truss properties was added and drop boxes were added for entering the number of the cross section and area of each member.

4.3.5 Verification of the Indeterminate Truss Analysis Applet

Indeterminate Truss Example 1

The simple indeterminate truss shown in Figure 4.12, taken from Holzer (1998), was analyzed in the applet. The truss geometry and loads are shown in Figure 4.12. The results from the applet and Holzer (1998) are shown in Figure 4.13 and Table 4.3, respectively. A comparison of the results shows that the results are identical.

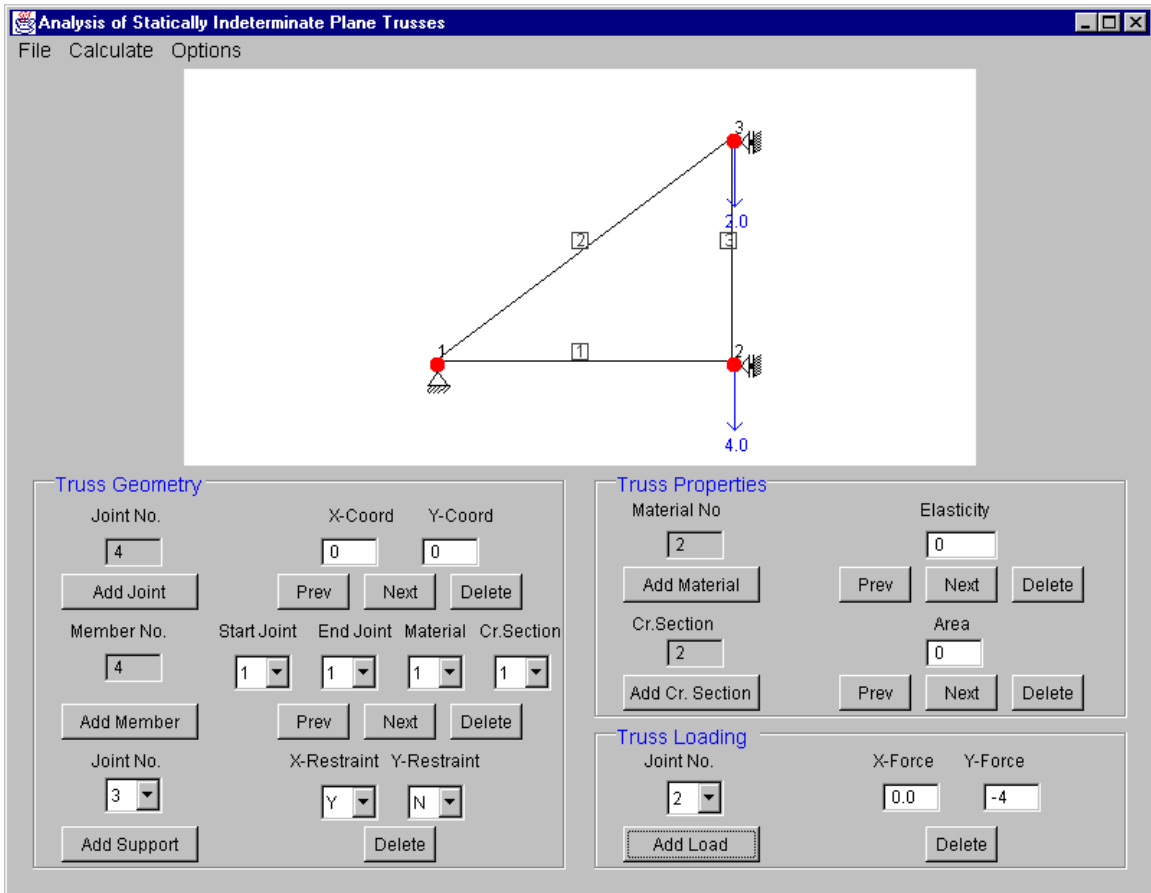


Figure 4.12 Indeterminate Truss Example 1.

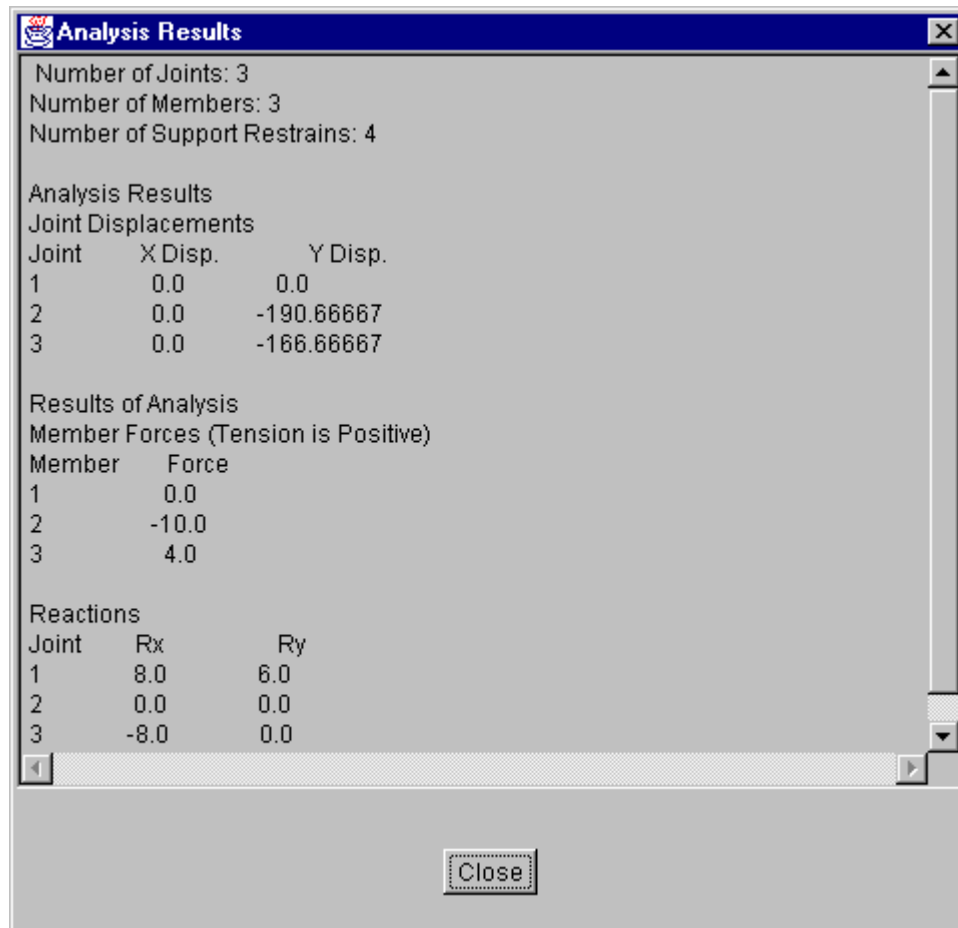


Figure 4.13 Results from Indeterminate Truss Analysis Applet for Example 1.

**Table 4.7 Member Forces and Reactions from Holzer (1998)
for the Truss of Example 1.**

Member	Force
1	0
2	10 k (C)
3	4 k (T)

Support	Reaction
R1X	8 k
R1Y	6 k
R2X	0
R3X	8 k

Indeterminate Truss Example 2

The second example used to verify the analysis results of this program was taken from Fleming (1997) The truss geometry and loads are shown in Figure 4.14. The member force results obtained from this program when compared to the results of Fleming (1997) are very close. The support reactions are identical. (See Fig. 4.15 and Table 4.8).

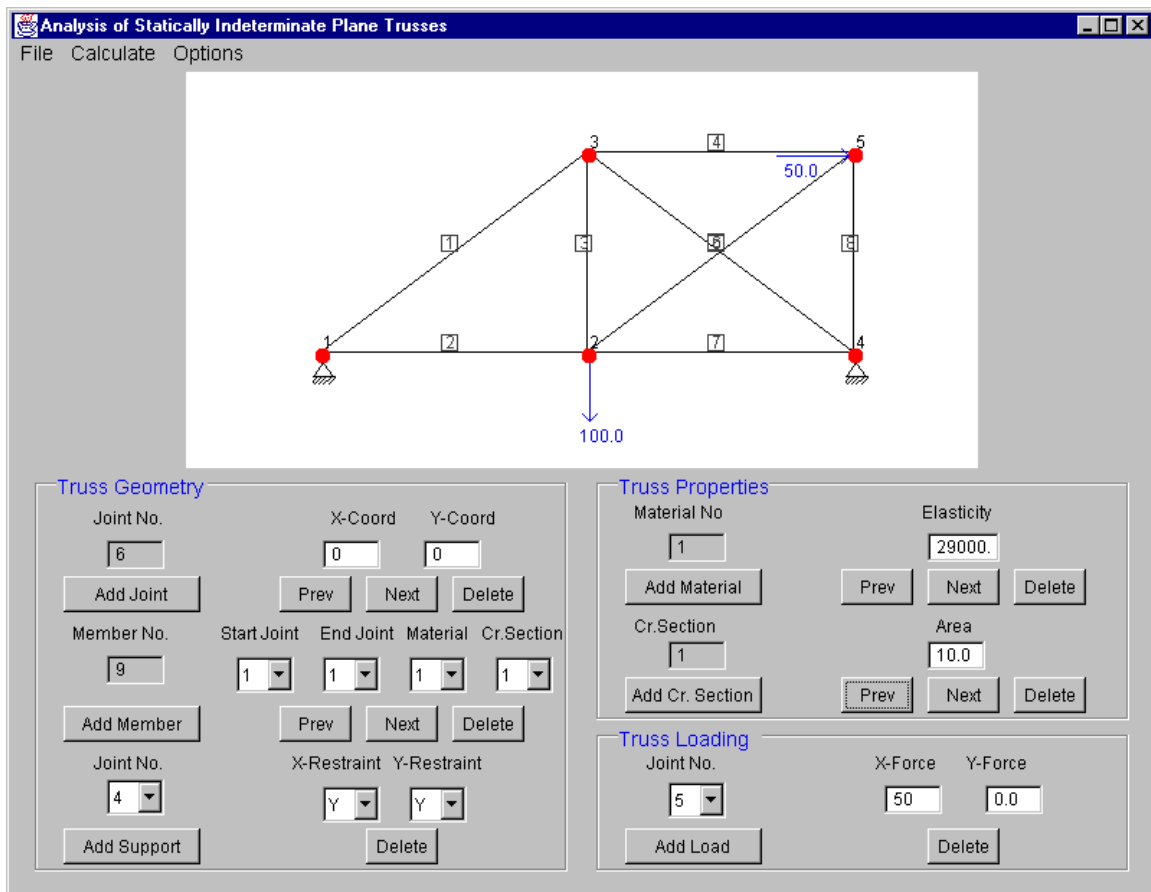


Figure 4.14 Geometry and Loads for the Indeterminate Truss of Example 2 from Fleming (1997).

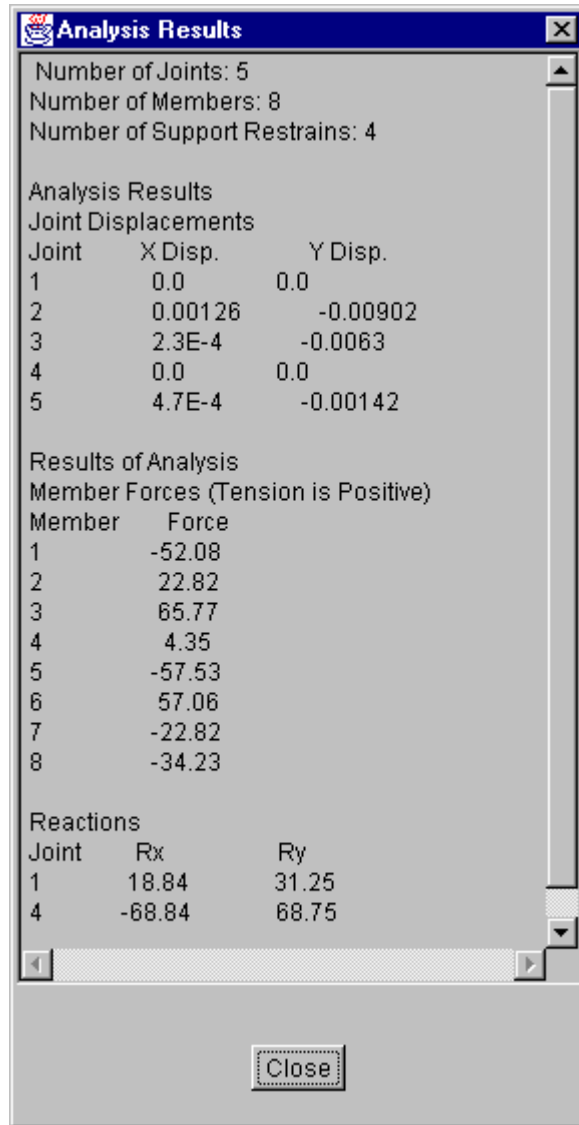


Figure 4.15 Results from Indeterminate Truss Analysis Applet for Example 2.

Table 4.8 Analysis Results for Indeterminate Truss of Example 2 from Fleming (1997).

Member	Force
1	52.103 k (C)
2	22.807 k (T)
3	65.653 k (T)
4	4.308 k (T)
5	57.517 k (C)
6	57.033 k (T)
7	22.807 k (C)
8	34.236 k (C)

Support	Reaction
R1X	18.844 k
R1Y	31.250 k
R4X	-68.844 k
R4Y	68.750 k

Chapter 5

Frame Analysis Applets

5.1 Introduction

In this chapter, two applets for the analysis of plane frames are presented. The Determinate Frame Analysis Applet analyzes frames that are statically determinate using a procedure that involves solving a series of simultaneous equations representing the equilibrium of the forces at the joints. The Indeterminate Frame Analysis Applet analyzes statically indeterminate plane frames using a matrix analysis approach.

5.2 Determinate Frame Analysis Applet

This applet analyzes statically determinate plane frames. The frame is analyzed by solving a set of simultaneous equations. The analysis is similar to that for statically determinate trusses described in the previous chapter. The forces on each frame member are converted to joint loads and member end loads. A set of equilibrium equations is written for each joint in the frame. These equilibrium equations are solved to obtain the member forces.

A typical frame member is shown in Figure 5.1. The local member end forces represented by $s(1)_n$, $s(2)_n$, $s(3)_n$, $s(4)_n$, $s(5)_n$ and $s(6)_n$. $s(2)_n$, $s(3)_n$, $s(4)_n$, and $s(6)_n$ are the unknowns. The other unknowns, the support reactions and $s(1)_n$ and $s(5)_n$, are later solved by the program by summing forces. The total number of unknowns is given by

$$\text{NUN} = 6\text{NM} + \text{NR} \quad (\text{Eq. 5.1})$$

where NUN = number of unknowns, NM = number of members and NR = number of restraints. The equilibrium equations are obtained by summing forces at each joint, summing local forces for each member, and summing forces at pinned connections. The

number of equilibrium equations is

$$NEQ = 3NJ + 3NM + NP \quad (\text{Eq. 5.2})$$

where NEQ = number of equations, NJ = number of joints and NP = number of pinned connections. Setting the total number of equilibrium to the total number of unknowns, gives

$$3NJ = 3NM + NR - NP \quad (\text{Eq. 5.3})$$

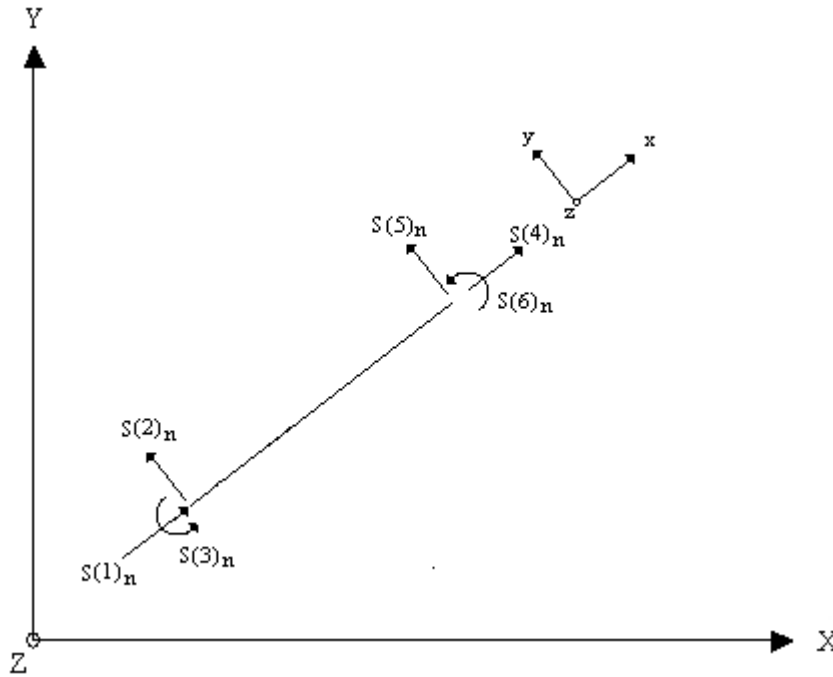


Figure 5.1 – Local Member End Loads.

As discussed in Chapter 3, Section 3.3, if the number of equations is equal to the number of unknowns, the frame is statically determinate. If the number of equations is greater than the number of unknowns the frame is unstable ($3NJ > 3NM + NR - NP$). On the other hand, if the number of equations is less than the number of unknowns the frame is unstable ($3NJ < 3NM + NR - NP$).

The program first sets up equations for the sum of the forces at the unrestrained joints of the frame in the global X and Y directions and the sum of the moments in the

unrestrained Z directions (3NJ-NR equations). Next, the program sets up equations for the sum of the moments about the start of each member in the local directions (NM). If any pinned connections are present, the corresponding member end loads are set to zero (NP). Therefore, the total number of equilibrium equations that are set up is equal to four times the number of members in the frame (4NM).

As was done in the analysis of determinate trusses, the directional cosines are determined from the joint coordinates. Also, member loads are converted to joint loads. This is done by computing the equivalent fixed end reactions for the member loads. If the joint is unrestrained the joint load is added to the joint load vector of the structure. The matrix of the coefficients of the unknowns is then obtained. The coefficients for the unknown forces summed in the global unrestrained X and Y directions include the directional cosines.

The unknown member forces are solve using the Gauss-Jordan Elimination Method. After member forces are determined, $S_n(1)$ and $S_n(5)$ are computed by summing forces in the local directions of each member. Next, the support reactions are found by summing forces at the restrained joints.

5.2.1 Determinate Frame Analysis Applet Interface

The main frame of the determinate truss program contains two panels of buttons, a group of buttons for the main program, a menu bar, and a canvas (see Figure 5.2). The main menu bar controls the operation of the program.

The Frame Geometry panel allows the user to enter the geometry of the frame. Within the Frame Geometry panel there are input fields for entering joint coordinates, member incidences and joint restraints. Other features include support for adding and deleting joints, member and supports.

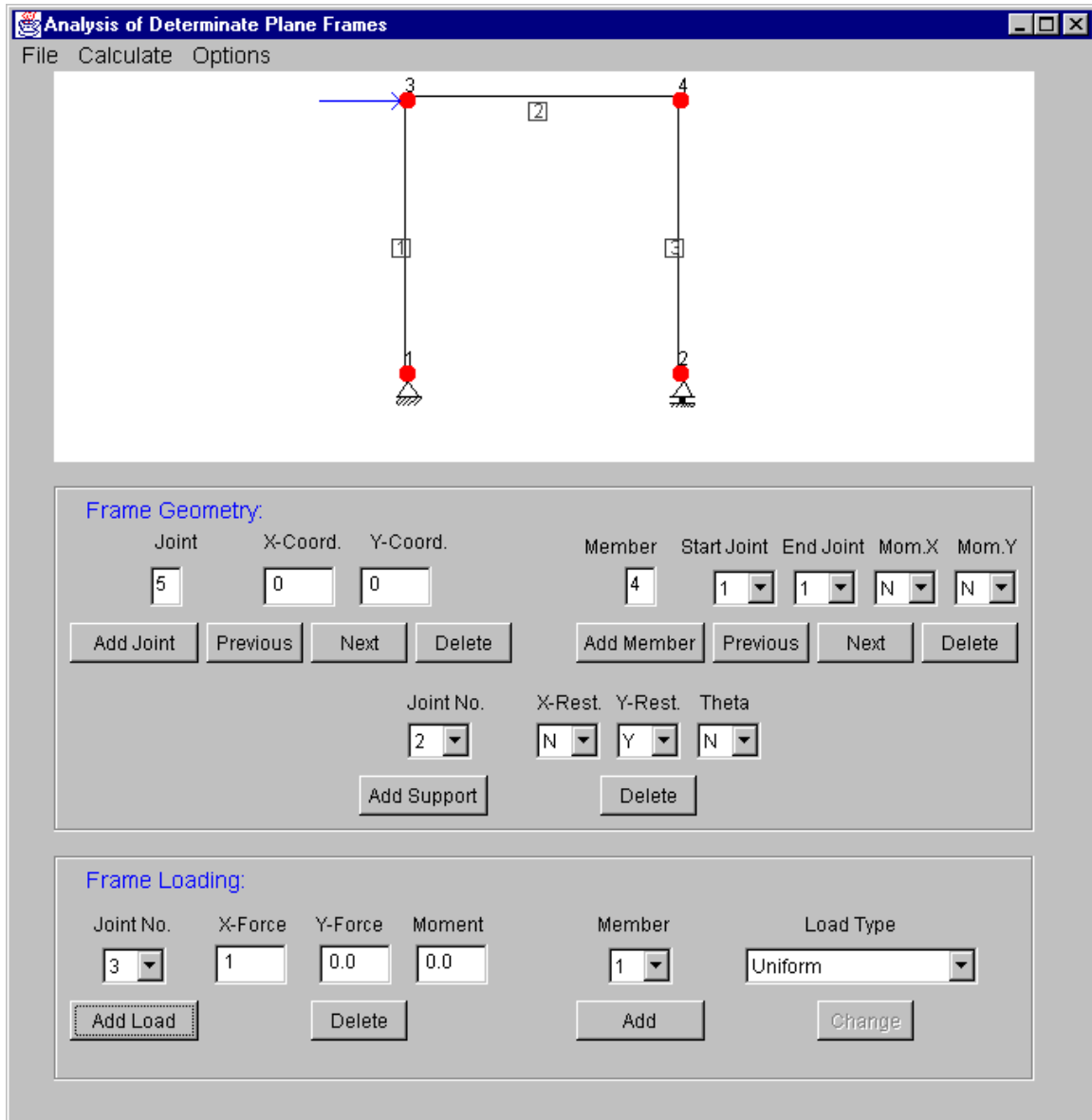


Figure 5.2 Determinate Frame Program Interface.

Loads acting on the frame are entered through the Frame Loading panel. Joint loads can be entered using the first group of textfields and buttons in the “Frame Loading” panel. The second group of buttons and drop boxes on the panel allow member loads such as uniform, partial uniform, linear, intermediate moment, and concentrated loads to be placed on the members of the frame. The dialog boxes that appear when different load types are chosen are shown in Figures 5.3 through 5.7. Loads can be edited by clicking on the “Change” button and by using the “Change Load” dialog box that appears (see Figure 5.8).

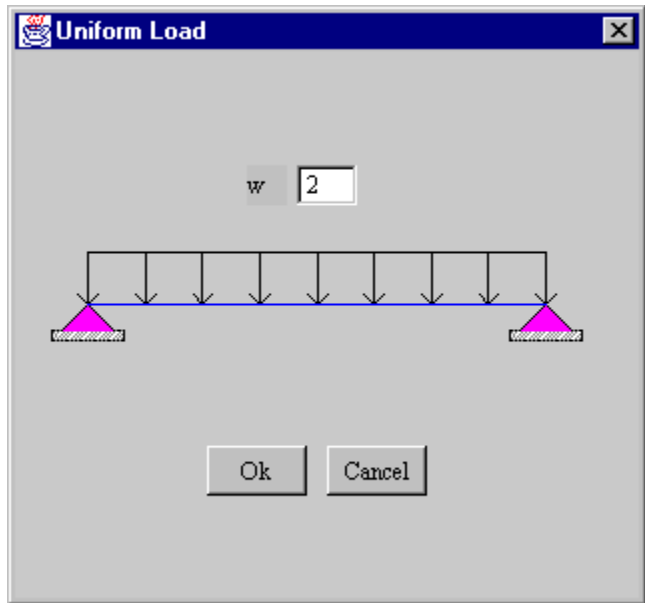


Figure 5.3 Uniform Load Dialogbox.

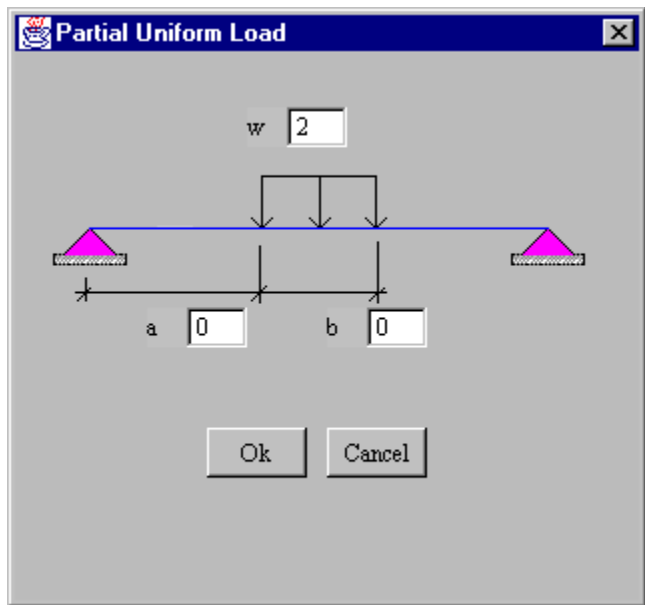


Figure 5.4 Partial Uniform Load Dialogbox.

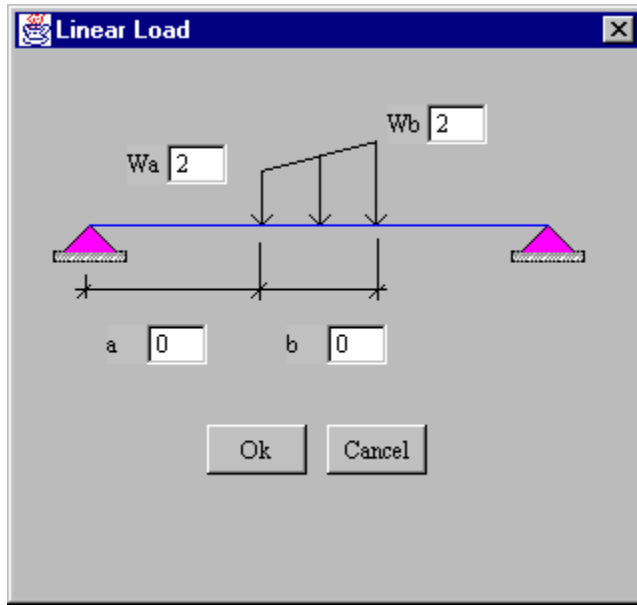


Figure 5.5 Intermediate Moment Dialogbox.

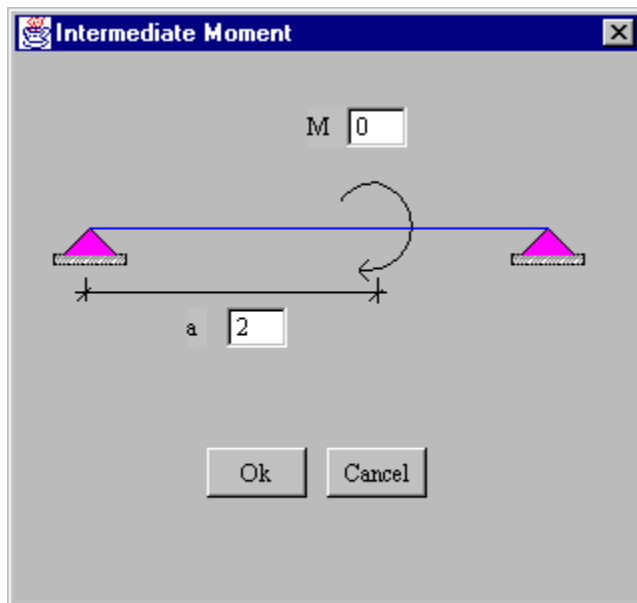


Figure 5.6 Intermediate Moment Dialogbox.

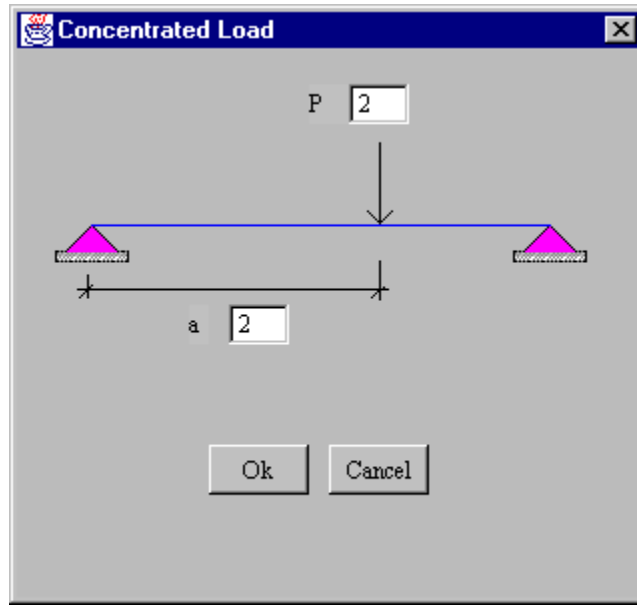


Figure 5.7 Concentrated Load Dialogbox.

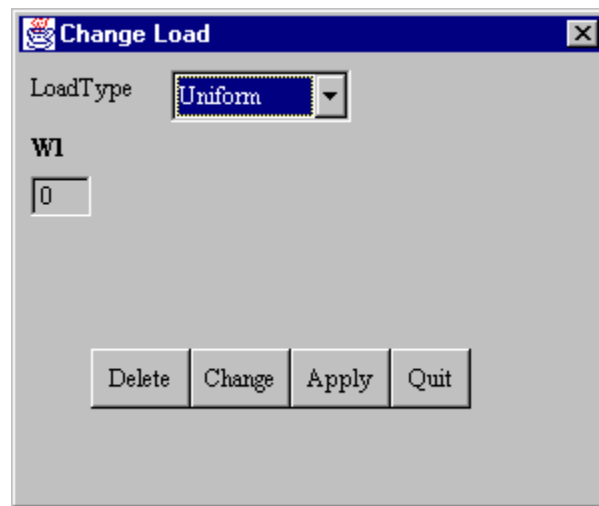


Figure 5.8 Change Load Dialogbox.

5.2.2 Determinate Frame Analysis Applet Classes

The structural classes included in this applet are the Joint, Load, Support, Member and Analysis classes. A description of these classes is given in Table 5.1. An Analysis class was developed specifically for this program using the analysis approach described in Section 5.2.1. A list of the methods used in the Analysis Class is given in Table 5.2.

Table 5.1 Structural Classes Contained in the Determinate Frame Analysis Applet.

Class	Description
Analysis	Performs the analysis for a determinate frame.
Member	Represents a member. This class contains data members for storing information about the start and end coordinates, length, direction cosines, and member forces.
Joint	Represents a joint. This class contains data members for storing the x and y coordinates of the joint.
Support	Represents a support. This class contains data members for storing the degrees of freedom and corresponding reactions.
Load	Represents a load acting at the joint. The member loads are converted to joint loads.

Table 5.2 Methods in the Analysis Class of the Determinate Frame Analysis Applet.

Function	Description
CheckStability()	Checks stability of the determinate frame and displays an error message if the frame is unstable
ComputeDirectionCosines()	Computes direction cosines for each member of the frame.
convertMemberLoadsToJointLoads()	Converts loads acting on the members to joint loads.
AssignEquationNo()	Assigns degrees of freedom to unrestrained directions
GenerateCoefficients()	Generates coefficient matrix.
SetUpLoad()	Generates load matrix.
Guass()	Solves member forces using Guass elimination.
computeReactions()	Computes reaction forces from loads at a support.
sumUpMemberLoad()	Sums joint loads and member forces at the supports.

The interface classes that were used were the Factor, the IntTextField, the DoubleTextField, the FloatTextField, the ManyLineLabel, the Dialogbox , the OptionParameter, the Option, and the Metric classes. These classes are the same as those used in the applets described earlier. Other interface classes solely created for this particular applet were the IndeterminatePlaneFrameApplet and the IndetPlaneTrussFrame classes. Dialog box classes for the different member loads were also created. These different load classes are the UniformLoad, PartialUniformLoad, ConcentratedLoad, IntermediateMoment, and LinearLoad, and ChangeLoad classes.

5.2.3 Verification of the Determinate Frame Analysis Applet

Several determinate plane frames were analyzed using the applet and the results obtained were verified with results from other sources. In this section, one of these examples used is presented.

Determinate Frame Example:

A one bay, one story frame 10 feet high spanning 10 feet with a one kip force acting at joint 2 is shown in Figure 5.8. This frame was analyzed using the applet. The results given by the applet were compared with those obtained using the RISA-2D commercial software analysis program (RISA 2001). Results obtained from the applet are shown in Table 5.3. The results from RISA-2D are shown in Table 5.4. The results are identical for both cases.

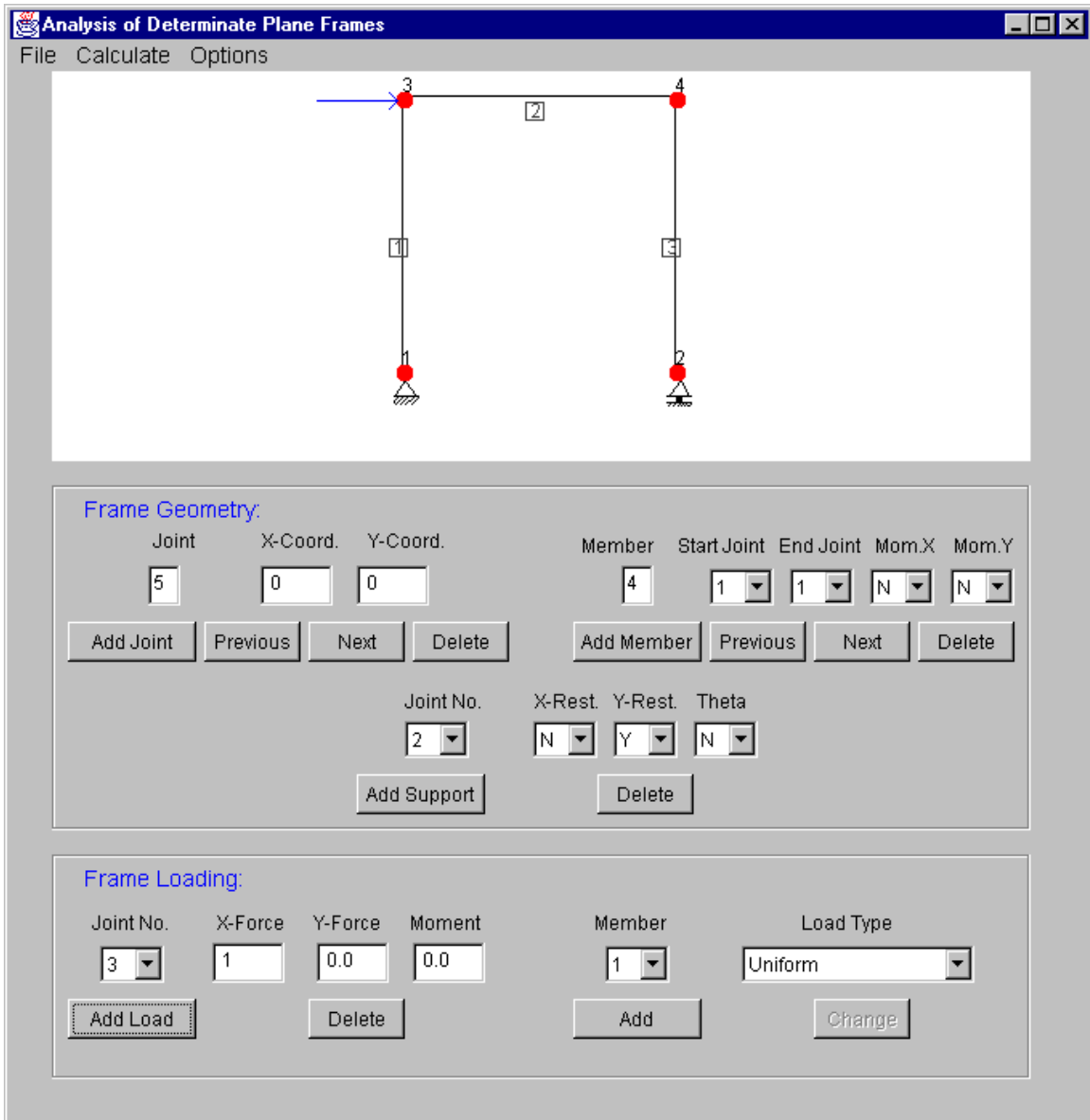


Figure 5.9 – Determinate Frame Example.

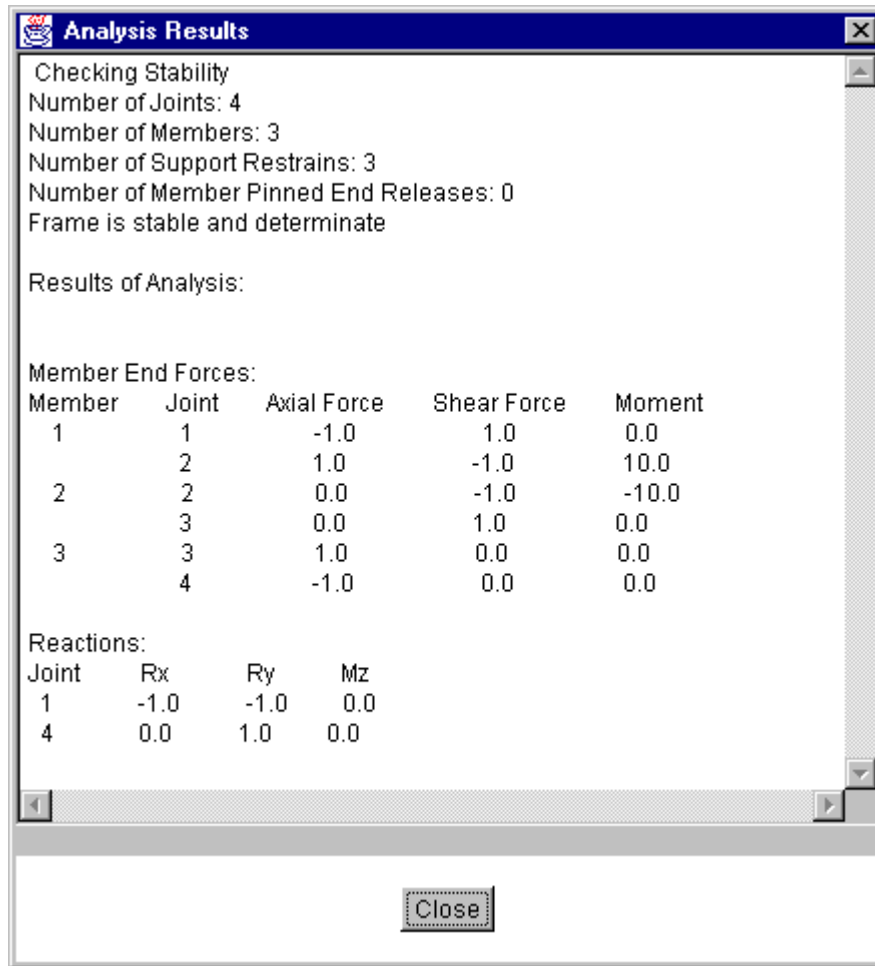


Figure 5.10 Results for Determinate Frame Example from Applet.

Table 5.3 Results for Determinate Frame Example from RISA-2D.

Member End Loads				
Member	Joint	Sx (k)	Vy (k)	Mz (k-ft)
1	1	-1	1	0
	2	1	-1	10
2	2	0	-1	-10
	3	0	1	0
3	3	1	0	0
	4	-1	0	0

Reactions			
Joint	RX (k)	RY (k)	RZ (k-ft)
1	-1	-1	0
2	0	1	0

5.3 Indeterminate Frame Analysis Applet

The Indeterminate Frame Analysis Applet analyzes statically indeterminate plane frames. The analysis procedure used is similar to that described for analysis of indeterminate trusses.

A member in a plane frame has an axial force, a shear force and a moment at each end (see Figure 5.11). The member end displacements of a typical frame member, n , are shown in Figure 5.12.

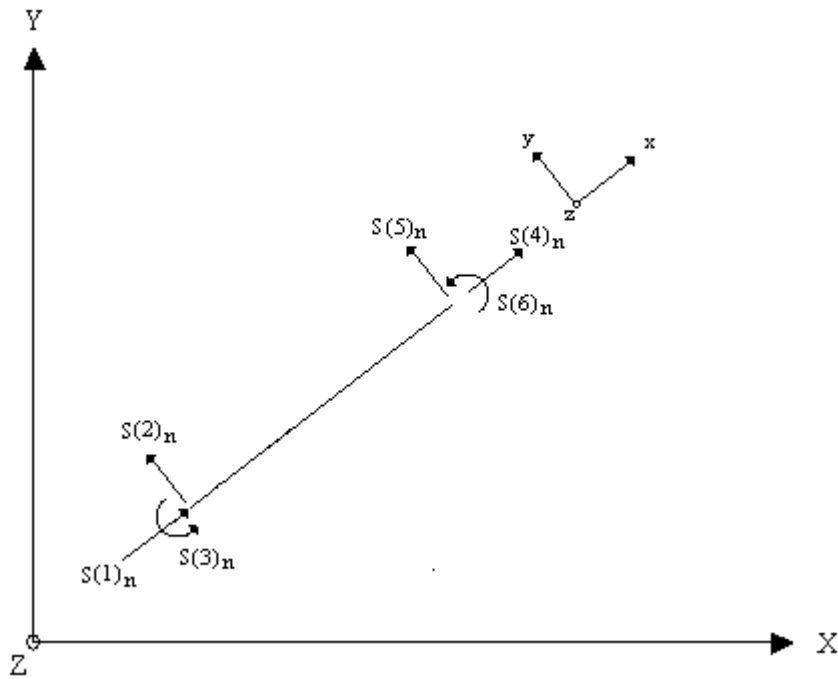


Figure 5.11 – Local Member Forces for a Plane Frame Member.

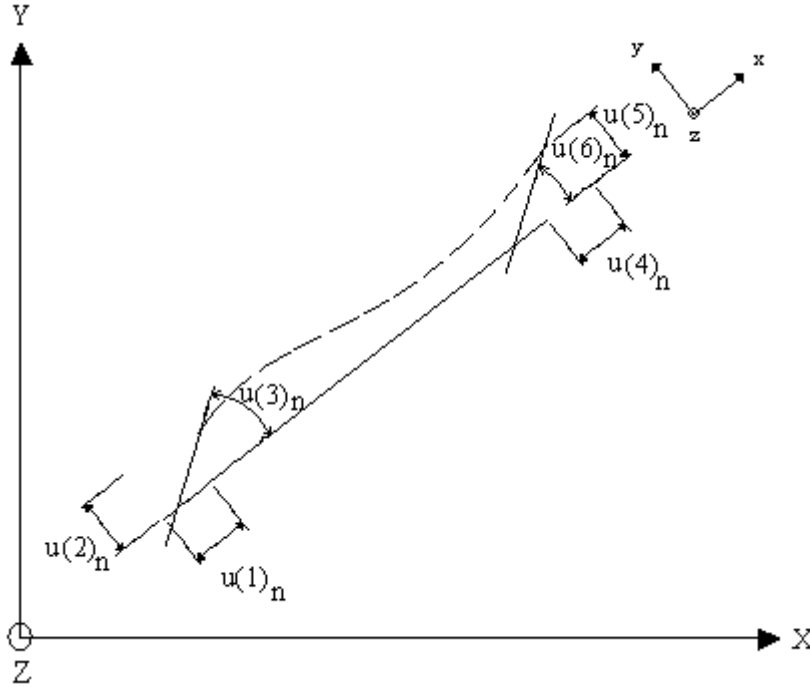


Figure 5.12 – Local Member Deformations for a Plane Frame Member.

The member forces S_n are equal to the local deformations $[U]_n$ multiplied by the local stiffness matrix $[K_m]_n$.

$$[S]_n = [K_m]_n [U]_n \quad (\text{Eq. 5.4})$$

$$[S]_n = \begin{pmatrix} s(1) \\ s(2) \\ s(3) \\ s(4) \\ s(5) \\ s(6) \end{pmatrix}_n \quad (\text{Eq. 5.5})$$

$$[K_m]_n = \begin{pmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ \frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{pmatrix}_n \quad (\text{Eq. 5.6})$$

where A = area of the member, E = modulus of elasticity, I = moment of inertia and L = length of the member.

$$[U]_n = \begin{pmatrix} u(1) \\ u(2) \\ u(3) \\ u(4) \\ u(5) \\ u(6) \end{pmatrix}_n \quad (\text{Eq. 5.7})$$

The member end forces in the global coordinate system, $[Q]_n$ for member n , are equal to the stiffness matrix $[K]_n$ of the member in the global coordinate system multiplied by its global member deformations $[D]_n$ (Eq. 5.8). For a plane frame the global load vector consists of six elements as shown in Figure 5.13. The global member deformations matrix also consist of six elements as shown in Figure 5.14. Thus, the relationship between member loads and member displacements in the global coordinate system can be written as

$$[Q]_n = [K]_n [D]_n \quad (\text{Eq. 5.8})$$

$$[Q]_n = \begin{pmatrix} q(1) \\ q(2) \\ q(3) \\ q(4) \\ q(5) \\ q(6) \end{pmatrix}_n \quad (\text{Eq. 5.9})$$

$$[K]_n = \begin{pmatrix} g1 & g2 & g4 & -g1 & -g2 & g4 \\ & g3 & g5 & -g2 & -g3 & g5 \\ & & g6 & -g4 & -g5 & g7 \\ & & & g1 & g2 & -g4 \\ & & & & g3 & -g5 \\ \text{sym.} & & & & & g6 \end{pmatrix}_n \quad (\text{Eq. 5.10})$$

$$[D]_n = \begin{pmatrix} d(1) \\ d(2) \\ d(3) \\ d(4) \\ d(5) \\ d(6) \end{pmatrix}_n \quad (\text{Eq. 5.11})$$

The elements of the member stiffness matrix g_1 through g_7 are given by (Holzer 98)

$$\begin{aligned} g_1 &= \alpha(\beta c_1^2 + 12c_2^2) \\ g_2 &= \alpha c_1 c_2 (\beta - 12) \\ g_3 &= \alpha(\beta c_2^2 + 12c_1^2) \\ g_4 &= -\alpha 6c_2 \\ g_5 &= \alpha 6c_1 \\ g_6 &= \alpha 4L^2 \\ g_7 &= \alpha 2L^2 \end{aligned} \quad (\text{Eq. 5.12})$$

where $c_1 = \cos \theta_x$, $c_2 = \cos \theta_y$, $\alpha = \frac{EI}{L^3}$ and $\beta = \frac{AL^2}{I}$.

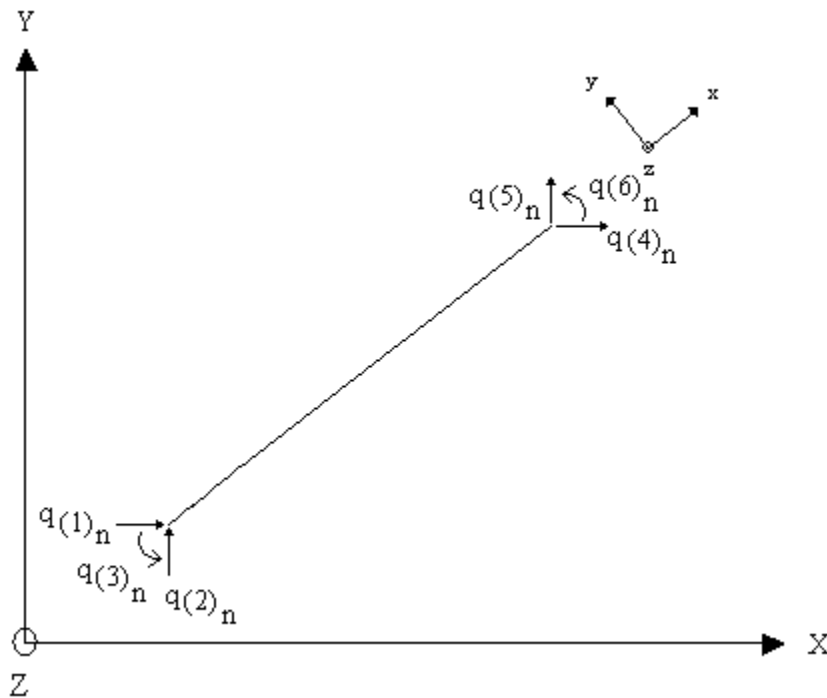


Figure 5.13 Global Member End Loads for a Plane Frame Member.

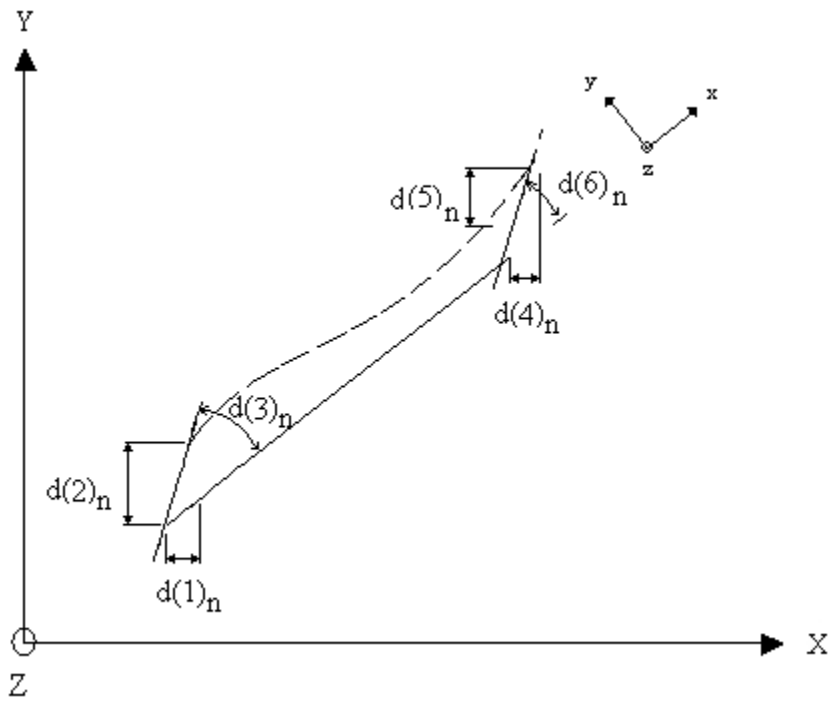


Figure 5.14 – Global Member Deformations for a Plane Frame Member.

The analysis begins with the consecutive assignment of degrees of freedom. The joints in the frame are checked in ascending order for restraints against rotation and translation in the global X and Y directions. A degree of freedom number is assigned to each unrestrained direction. If the joint is restrained, a zero value is assigned to the degree of freedom for the corresponding direction. A global load vector $[Q]$ for the structure corresponding to the number of degrees of freedom for the frame is defined.

The global stiffness matrix $[K]$ for the structure, also corresponding to the number of degrees of freedom for the frame is then initialized. The global stiffness matrix is the sum of the global stiffness matrices for all of the members in a structure. The assembly process or the process of placing the elements of the global member matrices into the correct locations of the global stiffness matrix is similar to that for indeterminate truss analysis.

The joint displacements are obtained by solving the system model using the Gauss-Jordan Method. The member end forces of each member are then calculated by solving Equations 5.13.

$$\begin{aligned}
 s(1)_n &= -AE/L ([d(4)_n - d(1)_n]C_X + [d(5)_n - d(2)_n]C_Y) & \text{(Eqs. 5.13)} \\
 s(2)_n &= -12EI/L^3 ([d(5)_n - d(2)_n]C_X + [d(4)_n - d(1)_n]C_Y) + 6EI/L^2([d(3)_n - d(6)_n]) \\
 s(3)_n &= 6EI/L ([d(5)_n - d(2)_n]C_X + [d(4)_n - d(1)_n]C_Y) + EI/L([d(3)_n + d(6)_n]) \\
 s(4)_n &= AE/L ([d(4)_n - d(1)_n]C_X + [d(5)_n - d(2)_n]C_Y) \\
 s(5)_n &= 12EI/L^3 ([d(5)_n - d(2)_n]C_X + [d(4)_n - d(1)_n]C_Y) - 6EI/L^2([d(3)_n + d(6)_n]) \\
 s(6)_n &= 6EI/L ([d(5)_n - d(2)_n]C_X - [d(4)_n - d(1)_n]C_Y) + EI/L([d(3)_n + d(6)_n])
 \end{aligned}$$

These equations were arrived at by transforming the global displacements to local displacements. The member end displacements are obtained from

$$\begin{aligned}
 u(1)_n &= d(1)_n \cos \theta_x + d(2)_n \cos \theta_y & \text{(Eqs. 5.14)} \\
 u(2)_n &= d(3)_n \cos \theta_x + d(4)_n \cos \theta_y \\
 u(3)_n &= d(3)_n \\
 u(4)_n &= d(4)_n \cos \theta_x + d(5)_n \cos \theta_y \\
 u(5)_n &= -d(4)_n \cos \theta_y + d(5)_n \cos \theta_x \\
 u(6)_n &= d(6)_n
 \end{aligned}$$

The reactions at the supports are then found by summing forces at the joints.

5.3.1 Indeterminate Frame Analysis Applet Interface

The interface for the indeterminate frame applet is similar to that of the determinate frame applet described in Section 5.2.2. The main frame of the applet is shown in Figure 5.14. Since the analysis of an indeterminate frame requires properties of the members, additional input fields and buttons are provided for entering the cross-sectional area, A , and I_x and I_y for each member of the frame.

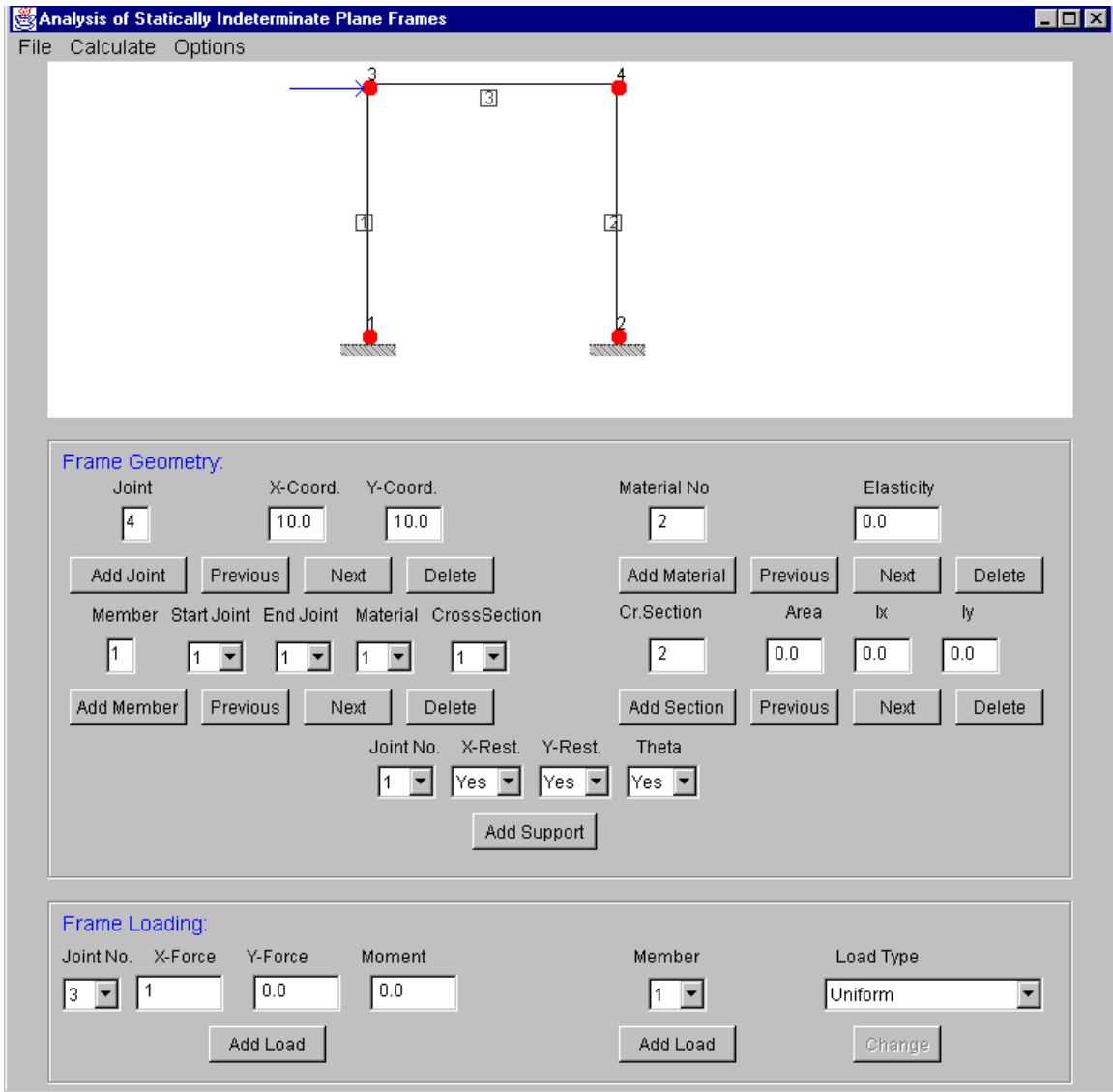


Figure 5.15 Indeterminate Frame Analysis Applet Interface.

5.3.2 Indeterminate Frame Analysis Applet Classes

This program contains many of the same classes as the determinate plane frame program. The structural classes included in this program are listed in Table 5.4. An Analysis class was developed specifically for this program using the analysis procedure described earlier. Brief descriptions of the methods used in the Analysis class are given in Table 5.5.

**Table 5.4 Structural Classes Contained in the Indeterminate
Frame Analysis Applet.**

Class	Description
Analysis	Performs the analysis for an indeterminate frame.
Member	Represents a member in the frame. This class contains data members for storing information about the start and end coordinates and member forces of a member.
Material	Represents the modulus of elasticity of each member.
CrossSection	Represents the cross sectional properties of each member. The data members include area and the moment of inertia about the x and y axes.
Joint	Represents a joint in the frame. This class contains data members for storing the x and y coordinates of the joint.
Support	Represents a support in the frame. This class contains data members for storing the directions that are restrained and the different reaction values.
JointLoad	Represents a load acting at the joints. Note that member loads are converted to joint loads.

**Table 5.5 Methods in the Analysis Class of the Indeterminate
Frame Analysis Applet.**

Function	Description
CheckStability()	Checks the stability of the frame and displays an error message if the frame is unstable.
AssignDegreesOf Freedom()	Checks if joints are restrained and assigns degrees of freedom.
SetUpLoad()	Sets up the load vector.
setUpGlobalStiff ()	Sets up the global stiffness of the structure.
Gauss()	Solves for joint displacements using Gauss elimination.
storeDisplacements()	Stores joint displacements at each joint.
ComputeMemberForces()	Computes member forces and reactions.

The interface classes used were the Factor, the IntTextField, the DoubleTextField, the FloatTextField, the ManyLineLabel, the Dialogbox, the OptionParameter, the Option, and the Metric classes. Other interface classes specifically created for this applet were the IndeterminatePlaneFrameApplet and the IndetPlaneFrame classes.

5.3.3 Verification of the Indeterminate Frame Analysis Applet

In this section, the results given by the applet for several example problems are compared with those obtained from other sources.

Indeterminate Frame Example 1:

The first example is a one bay, one story frame. The story height is 10 feet high and the bay width is 10 feet. The loads acting on the frame consist of a 100 k horizontal force at joint 2 and an applied moment of 5000 kip-in at joint 3 (see Figure 5.16). The results

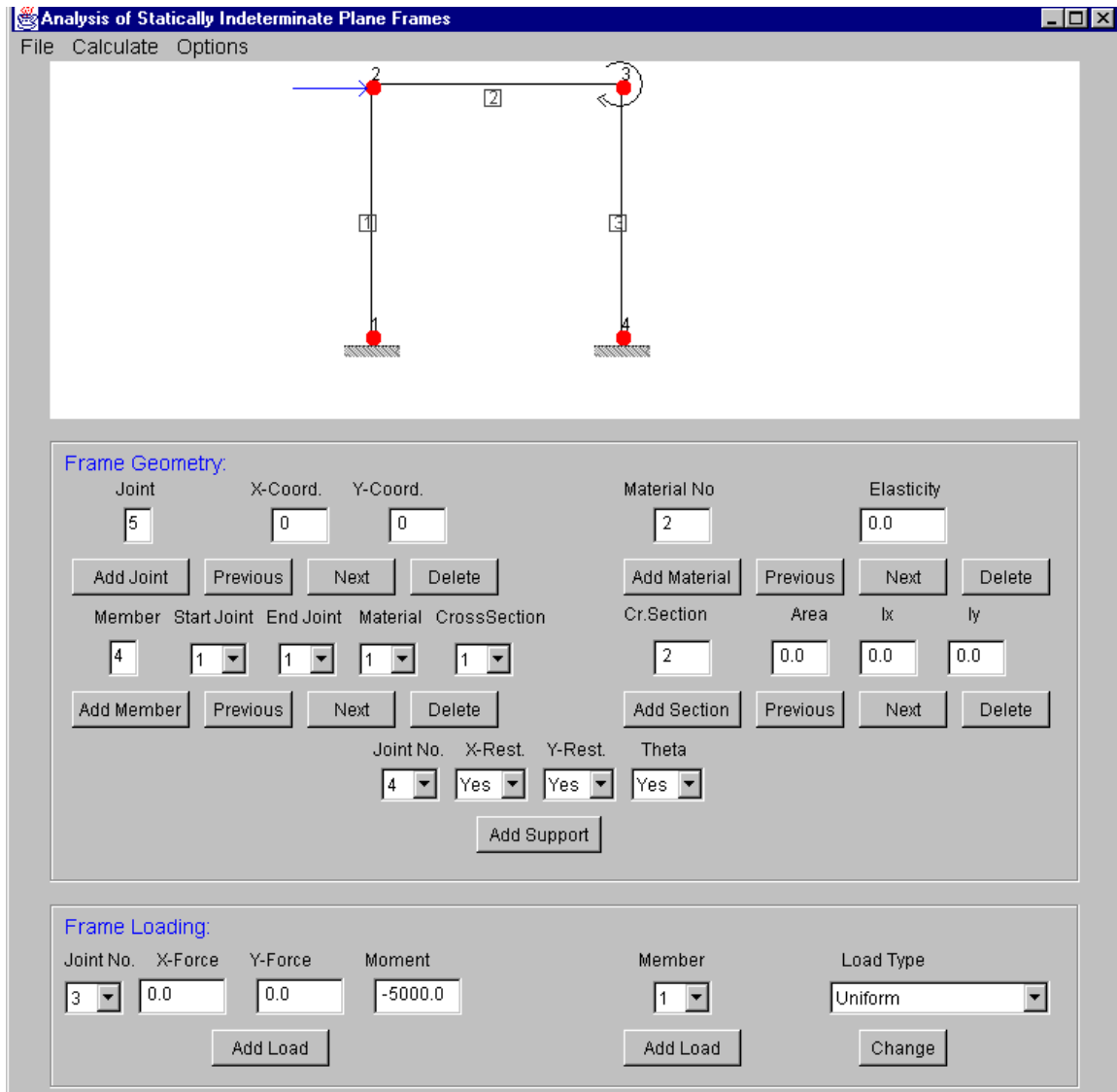


Figure 5.16 Example 1 - Frame Example From Fleming (1997).

obtained from the applet are shown in Figure 5.17. The corresponding values obtained from Fleming (1997) are shown in Table 5.6. As can be seen, the results for the applet are identical to those obtained by Fleming (1997).

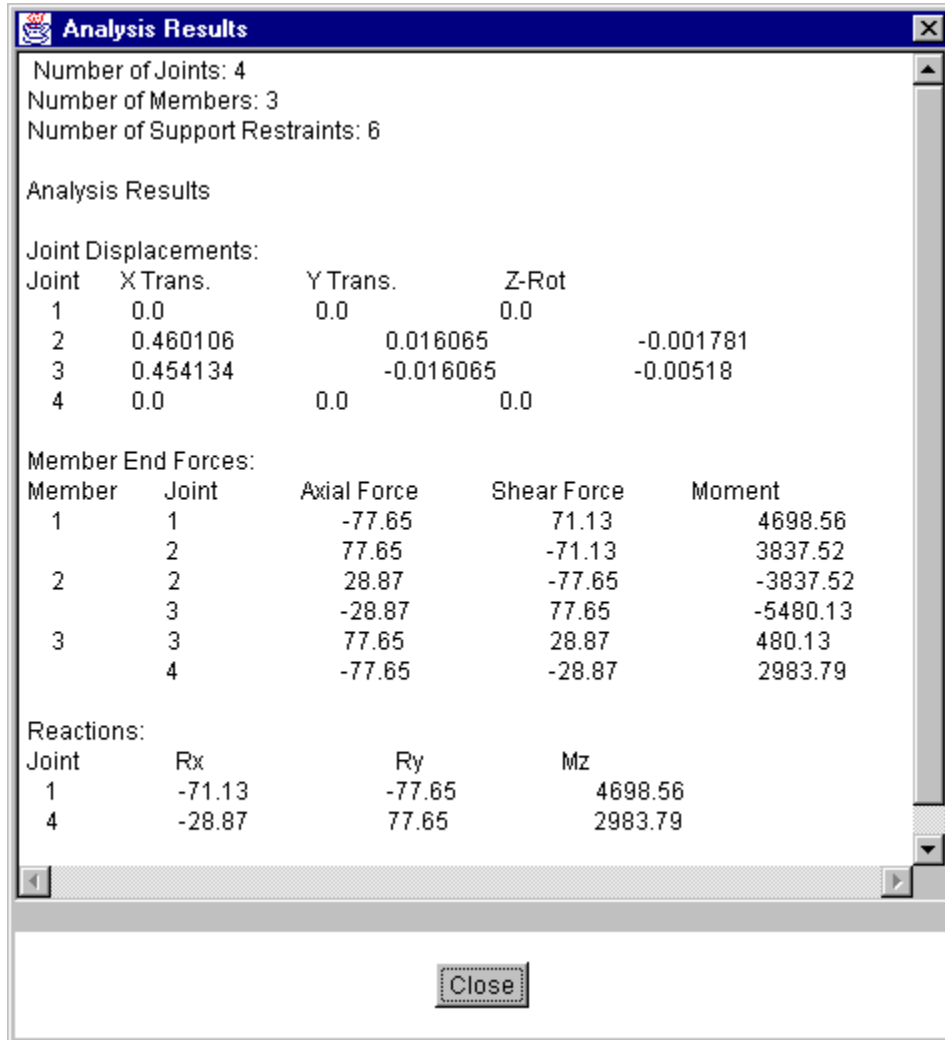


Figure 5.17 Example 1 - Frame Applet Results.

Table 5.6 Example 1 - Frame Results From Fleming (1997).

Joint Displacements			
Joint	X-Tran (in)	Y-Tran (in)	Z-Rot (rad)
1	0.000	0.000	0.00000
2	0.460	0.016	-0.00178
3	0.454	-0.016	-0.00518
4	0.000	0.000	0.00000

Member End Loads				
Member	Joint	Sx (k)	Vy (k)	Mz (k-in)
1	1	-77.647	71.134	4698.57
	2	77.647	-71.134	3837.52
2	2	28.866	-77.647	-3837.52
	3	-28.866	77.647	-5480.13
3	3	77.647	28.866	480.13
	4	-77.647	-28.866	2983.80

Reactions			
Joint	RX (k)	RY (k)	RZ (k-in)
1	-71.134	-77.647	4698.57
2	-28.866	77.647	2983.80

Indeterminate Frame Example 2:

The second example consists of a two member frame 20 feet high and 20 feet wide. There is a 5 k load acting on it at joint 2. (See Figure 5.18). The results from the applet are shown in Figure 5.19. The results of the program match the answers from Hibbeler (1995) shown in Table 5.7.

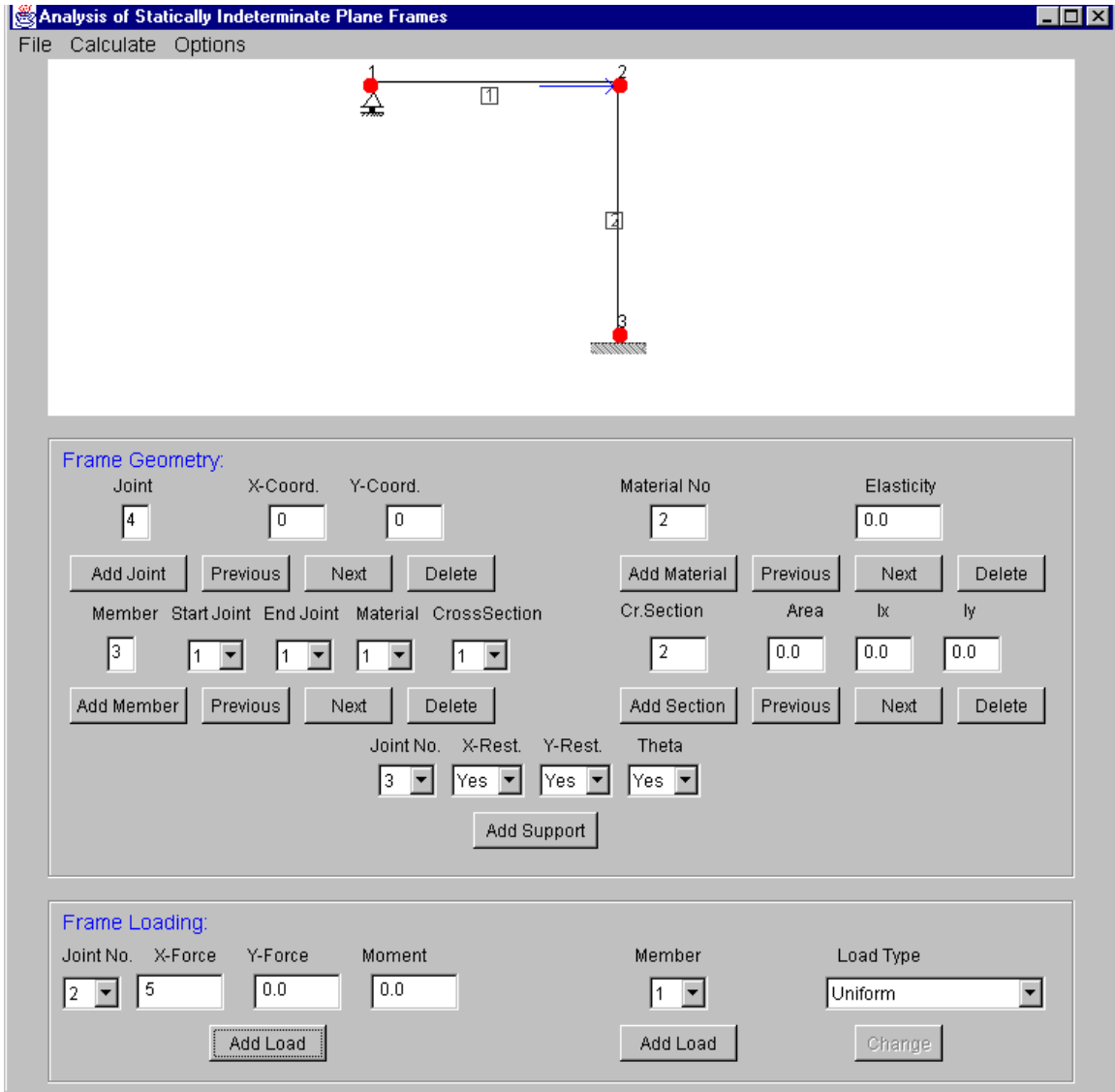


Figure 5.18 Example 2 - Frame From Hibbeler (1995).

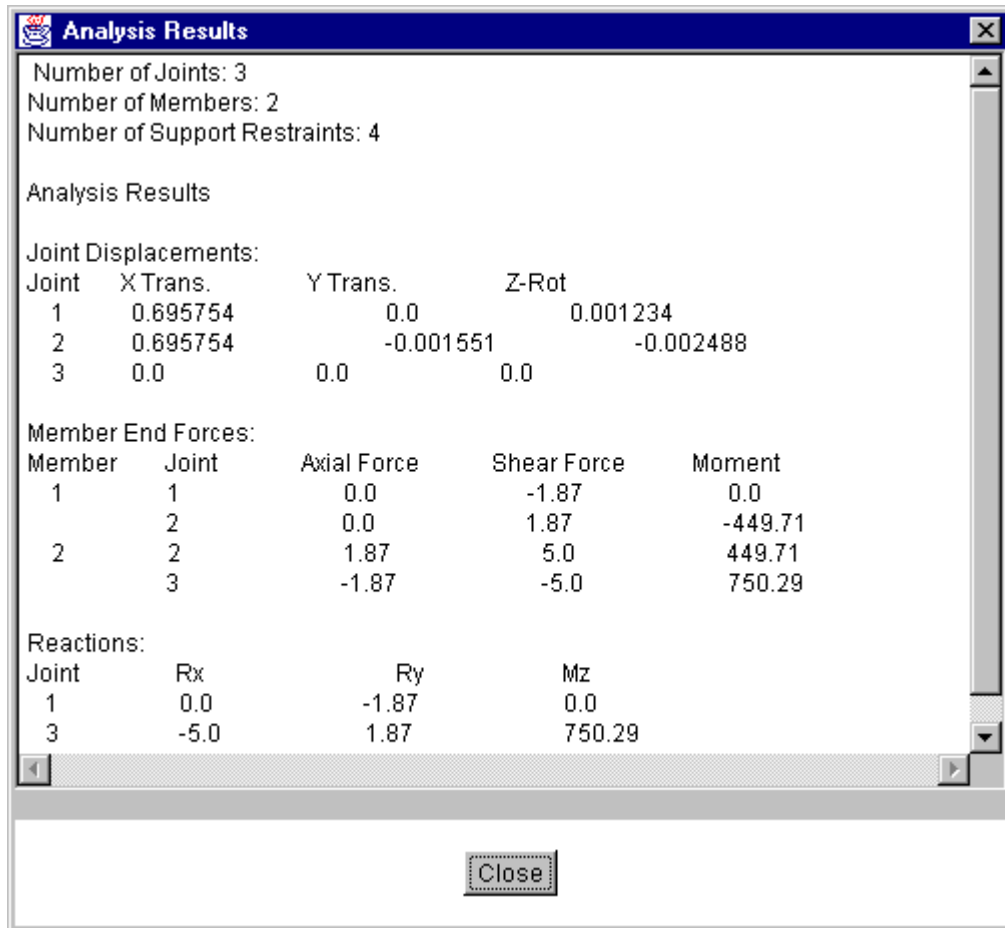


Figure 5.19 Example 2 - Frame Applet Results.

Table 5.7 Example 2 - Frame Results From Hibbeler (1995).

Joint Displacements			
Joint	X-Tran (in)	Y-Tran (in)	Z-Rot (rad)
1	0.696	0	0.001234
2	0.696	-0.00155	-0.002488
3	0	0	0

Member End Loads				
Member	Joint	Sx (k)	Vy (k)	Mz (k-in)
1	1	0	-1.87	0
	2	77.647	1.87	-450.0
2	2	-1.87	5.0	450.0
	3	1.87	-5.0	750.0

Reactions			
Joint	RX (k)	RY (k)	RZ (k-in)
1	0	-1.87	0
3	-5	1.87	750.0

Indeterminate Frame Example 3:

The third example is a two span continuous beam. The beam in Figure 5.20 was taken from Holzer (1998). The load acting on the beam consists of a one kip per foot uniformly distributed load on member 1. As can be seen from a comparison of Figure 5.21 and Table 5.8 the member end loads and reactions from the applet match the results obtained by Holzer (1998).

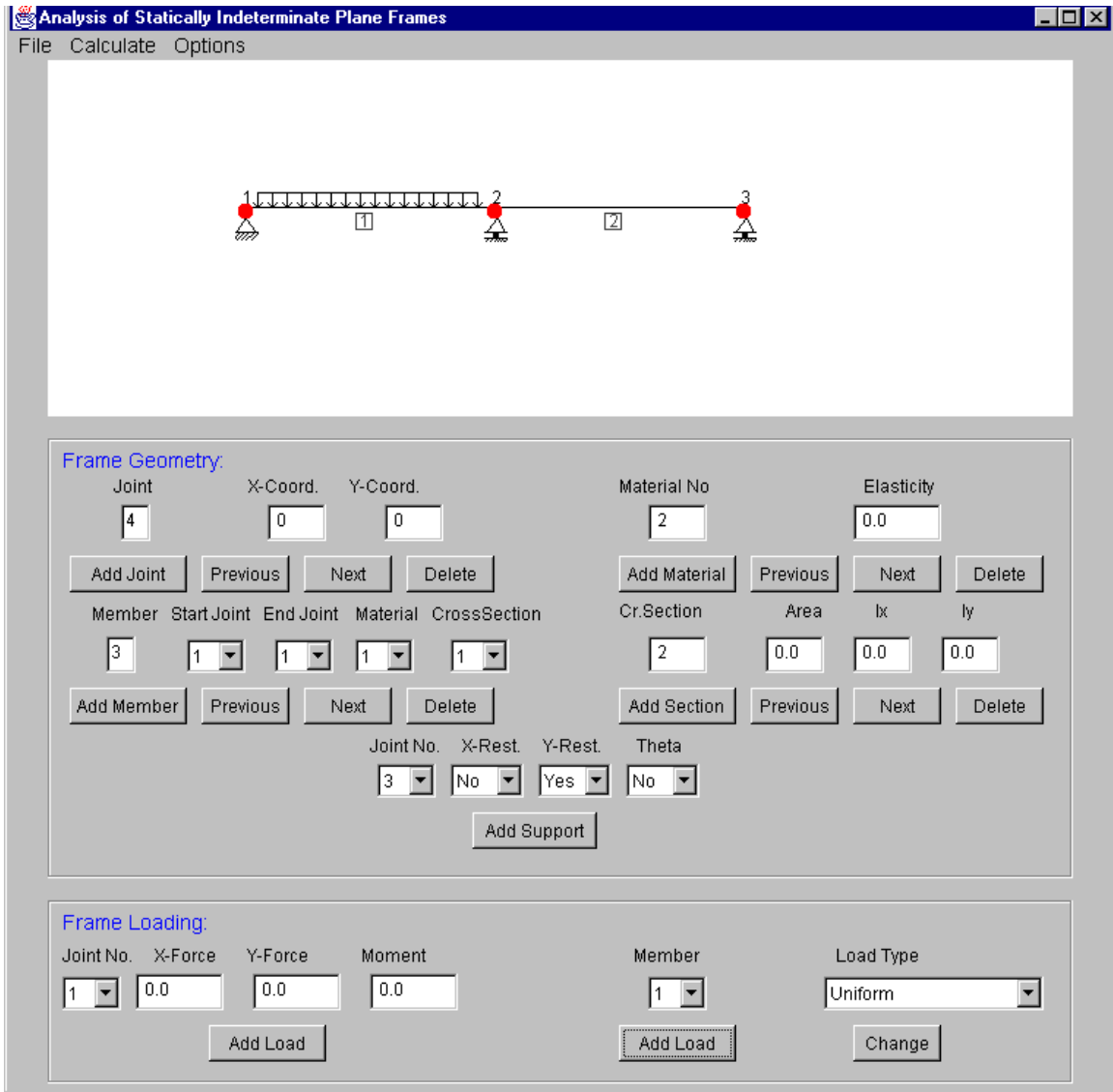


Figure 5.20 Example 3 - Beam From Holzer (1998).

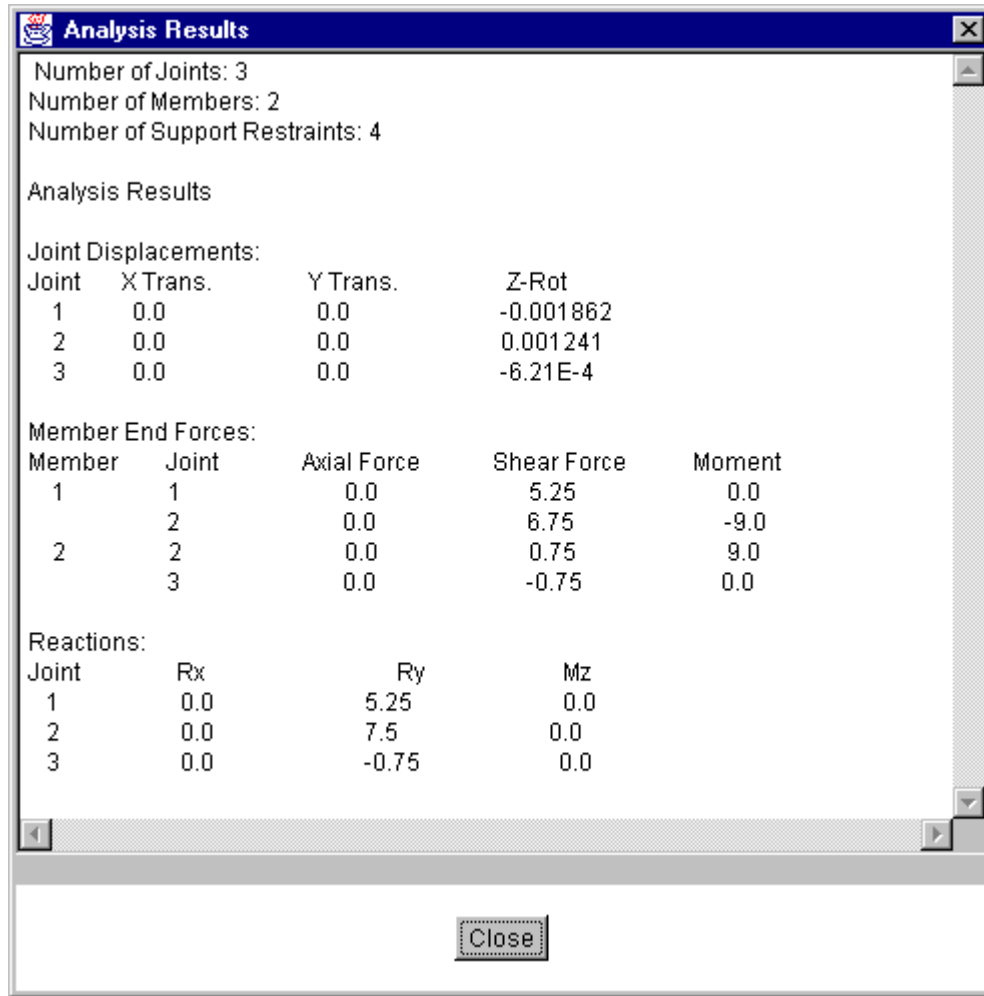


Figure 5.21 Example 3 - Frame Applet Results.

Table 5.8 Example 3 - Beam Results From Holzer (1998).

Member End Loads				
Member	Joint	Sx (k)	Vy (k)	Mz (k-ft)
1	1	0	5.25	0
	2	0	6.75	-9.0
2	2	0	0.75	9.0
	3	0	-0.75	0

Reactions			
Joint	RX (k)	RY (k)	RZ (k-ft)
1	0	5.25	0
2	0	7.50	0
3	0	-0.75	0

Indeterminate Frame Example 4:

The fourth example is a two story frame with story heights of 12 feet and bays widths of 20 feet is shown in Figure 5.22. There is a five kip load acting at joint 7 and a ten kip load acting at joint 4. This frame was analyzed using the applet and the commercial analysis program RISA-2D. Results obtained from the applet are shown in Figure 5.23. The results from RISA-2D and the applet and a percent comparison are shown in Tables 5.9-5.11.

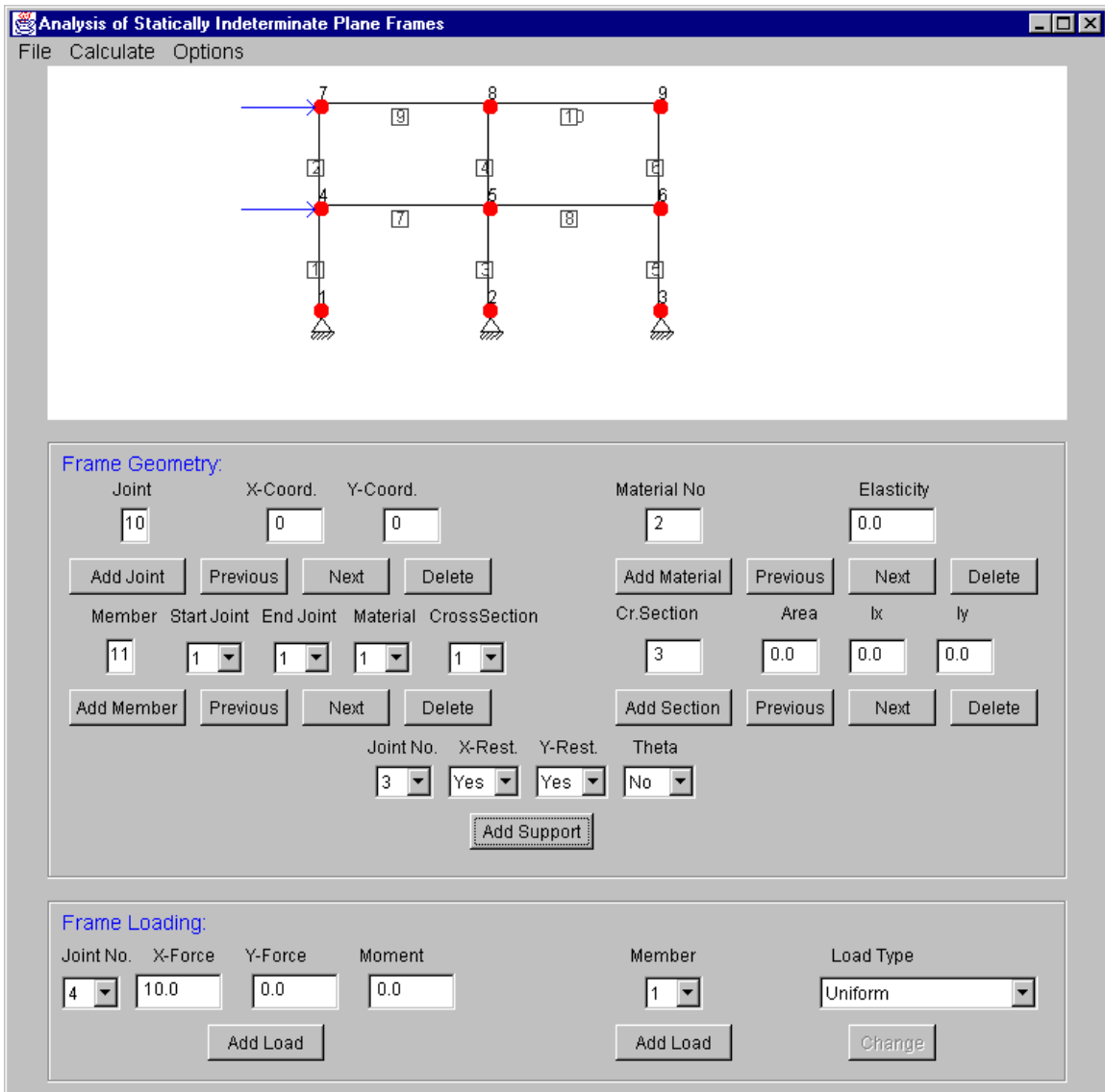


Figure 5.22 Indeterminate Frame Example 4.

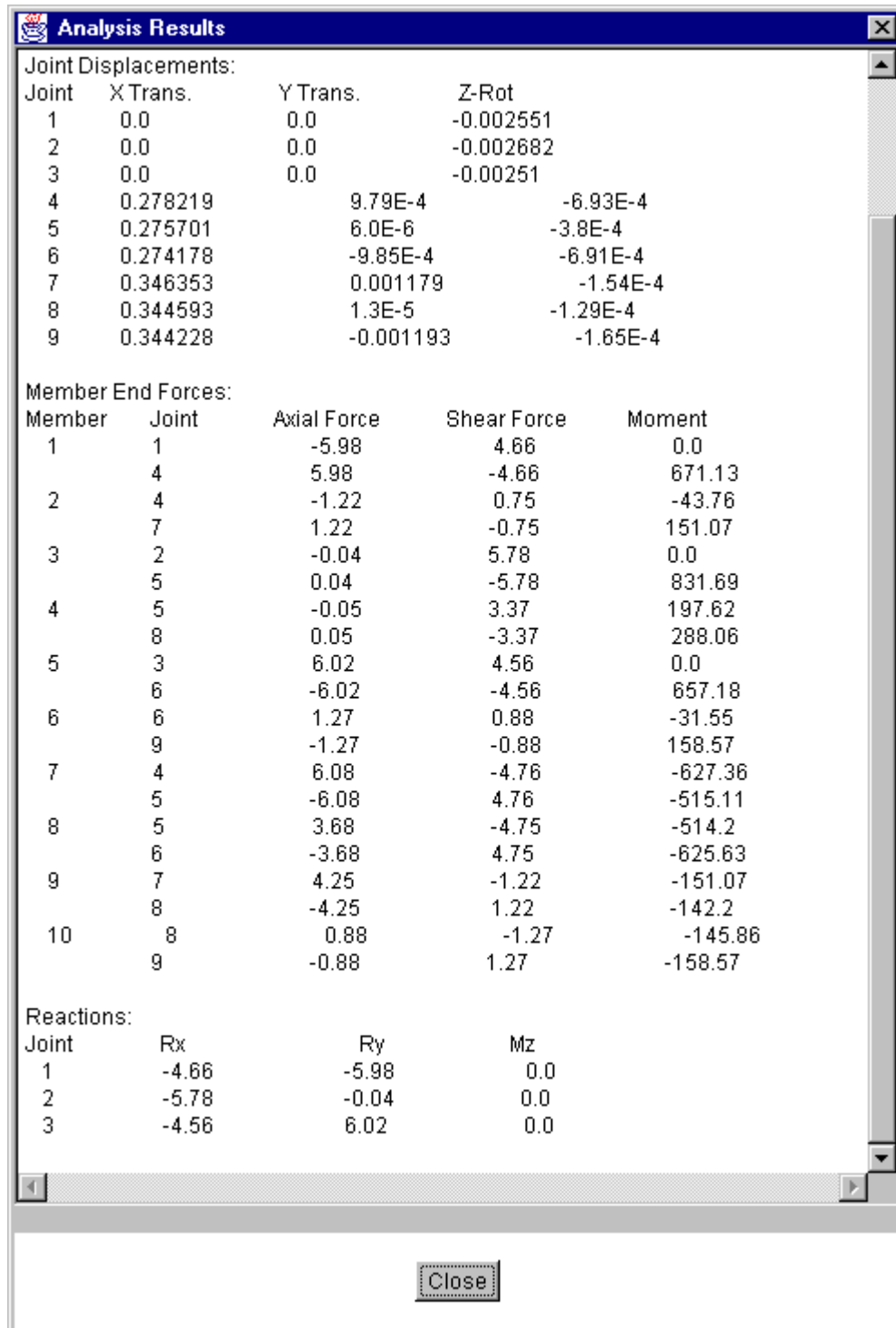


Figure 5.23 Example 4 – Applet Results.

Table 5.9 Example 4 – Applet and RISA-2D Axial Force Results.

Member	Joint	Axial Force (k) from Applet	Axial Force (k) from RISA 2D	% Difference
1	1	-5.98	-5.985	0
	4	5.98	5.985	0
2	4	-1.22	-1.26	3
	7	1.22	1.26	3
3	2	-0.04	-0.03	33
	5	0.04	0.03	33
4	5	-0.05	-0.042	19
	8	0.05	0.042	19
5	3	6.02	6.015	0
	6	-6.02	-6.015	0
6	6	1.27	1.301	2
	9	-1.27	-1.301	2
7	4	6.08	6.115	0.6
	5	-6.08	-6.115	0.6
8	5	3.68	3.663	0.5
	6	-3.68	-3.663	0.5
9	7	4.25	4.225	0.6
	8	-4.25	-4.225	0.6
10	8	0.88	0.904	2
	9	-0.88	-0.904	2

Table 5.10 Example 4 – Applet and RISA-2D Shear Force Results.

Member	Joint	Shear Force (k) from Applet	Shear Force (k) from RISA 2D	% Difference
1	1	4.66	4.66	0
	4	-4.66	-4.66	0
2	4	0.75	0.775	2
	7	-0.75	-0.775	2
3	2	5.78	5.773	0.2
	5	-5.78	-5.773	0.2
4	5	3.37	3.321	1
	8	-3.37	-3.321	1
5	3	4.56	4.567	0
	6	-4.56	-4.567	0
6	6	0.88	0.904	2
	9	-0.88	-0.904	2
7	4	-4.76	-4.725	0.8
	5	4.76	4.725	0.8
8	5	-4.75	-4.714	0.8
	6	4.75	4.714	0.8
9	7	-1.22	-1.26	3
	8	1.22	1.26	3
10	8	-1.27	-1.301	2
	9	1.27	1.301	2

Table 5.11 Example 4 – Applet and RISA-2D Moment Results.

Member	Joint	Shear Force (k) from Applet	Shear Force (k) from RISA 2D	% Difference
1	1	0	0	0
	4	55.93	55.923	0
2	4	-3.65	-3.894	0
	7	12.59	13.199	4
3	2	0	0	0
	5	69.31	69.275	0
4	5	14.97	15.595	4
	8	24.01	24.251	1
5	3	0	0	0
	6	54.77	54.802	0
6	6	-2.63	-2.924	1
	9	13.21	13.772	4
7	4	-52.28	-52.029	0.5
	5	-42.93	-42.471	1
8	5	-42.85	-42.399	1
	6	-52.14	-51.878	0.5
9	7	-12.59	-13.199	0.5
	8	-11.85	-11.997	1
10	8	-12.16	-12.255	0.7
	9	-13.22	-12.772	4

Chapter 6

Influence Line Applet

6.1 Introduction

An influence line is a useful tool developed by Professor E. Winkler, a German engineer, in 1867 for analyzing structures subjected to moving loads [Kinney, 1957]. Examples of structures subjected to moving loads include highway and railway bridges, industrial buildings with overhead travelling cranes and frames supporting conveyers. The internal forces in these structures vary not only with the magnitude of the loads but also with the position of the loads. Structures with moving loads are designed for the maximum forces developed as the loads move across the structure. The analysis of these structures involves determining the positions of the loads that produce the maximum response. Responses of interest include support reactions, shear forces, bending moments and displacements. An influence line shows graphically how the movement of a unit load across a structure influences the response of the structure. The shape of the influence line is helpful in finding the position of the load that produces the maximum.

The Influence Line Applet draws influence lines for determinate and indeterminate beams. The program considers a number of equally spaced points in each span of the beam. A load is placed at these points as it moves along the beam. At each point that the unit load is placed, the beam is analyzed using the matrix displacement method. The support reactions are found and stored in an array. The program then computes the chosen response at a given location by summing forces. The value that is obtained is placed into another array. After the program has completed all of its analyses at the different interval points, the influence line of the beam is drawn.

6.2 Analysis Procedure

The beam is analyzed using the matrix displacement method. A brief description of the analysis procedure follows. The forces acting at the ends of a typical beam element and

the corresponding displacements are shown in Figure 6.1. The element forces, f^i on the beam are obtained by multiplying the element stiffness matrix by the displacements at the ends of the element (Eq. 6.1).



Figure 6.1 Forces on a Typical Beam Element.

$$f^i = kd^i \quad (\text{Eq. 6.1})$$

$$\begin{pmatrix} f1 \\ f2 \\ f3 \\ f4 \end{pmatrix} = \alpha \begin{pmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{pmatrix} \begin{pmatrix} d1 \\ d2 \\ d3 \\ d4 \end{pmatrix}, \quad \alpha = EI/L^3$$

where E = modulus of elasticity, I = moment of inertia, L = length of the beam span, k = beam stiffness matrix, $f1$ - $f4$ = element forces and $d1$ - $d4$ = displacements.

The stiffness matrix of the system was determined by first converting the element stiffness matrices from the local system to the global system, and then by imposing compatibility conditions (Eq. 6.2).

$$Kq_k = Q_k \quad (\text{Eq. 6.2})$$

The member code matrix, which relates the presence of joint displacements at the ends of each element was developed as described in Chapter 4. The structure stiffness matrix was

produced by the summation of each element's converted global stiffness matrix (Eq. 6.3).

$$K = \sum K^{(i)} \quad (\text{Eq. 6.3})$$

where q_k are the unknown joint displacements, Q_k are the known external loads and K is the global stiffness matrix of the system.

The known external loads (Q_k) are calculated from the fixed end reactions for a concentrated load along a span at a given distance as shown in Figure 6.2.

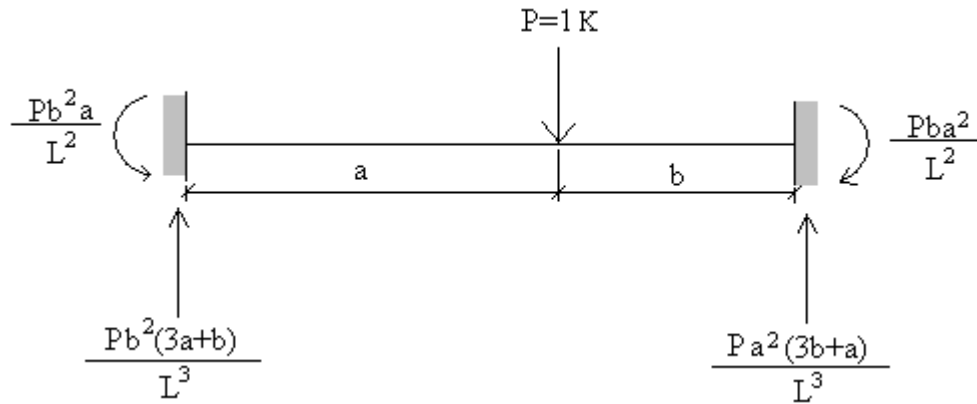


Figure 6.2 Fixed End Reactions for a Unit Concentrated Load.

For this program, the concentrated load is a unit load and the distance, a , varies as the load is moved along the span.

The joint displacements are then computed using Gauss elimination to solve the system of equations (Eq. 6.2). Once the joint displacements are known, they are substituted into the element model (Eq. 6.1) and element forces are calculated. After element forces are computed, the forces at each joint of the system are determined from the joint equilibrium equations.

The matrix displacement calculations described in the previous sections are performed at each point. The response at the desired location for each position of the unit load is

determined from the equilibrium equations. These results are stored in an array and are then plotted by the program, producing the influence lines.

6.3 Influence Line Applet Interface

The main frame of the influence line program consists of two canvas areas, a panel of buttons, and a menu bar as shown in Figure 6.3. The beam geometry is displayed in the upper canvas area. The influence line is drawn in the lower canvas area. The beam geometry, which includes joint coordinates and joint restraints, is entered via the input panel. The user can add or delete joints and supports and enter joint coordinates using the various buttons, text input fields and list boxes provided in the input panel.

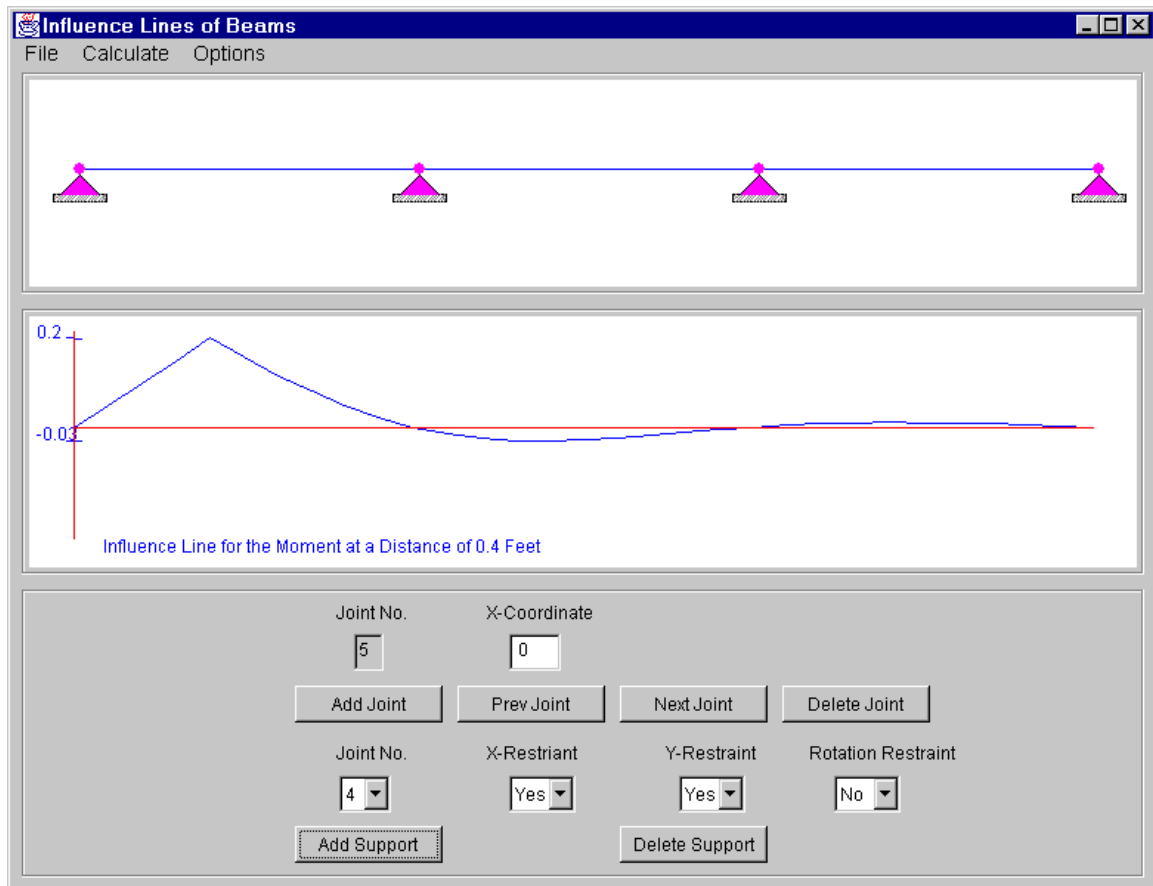


Figure 6.3 Influence Line Applet.

6.3.1 Menu Bar

As seen in Figure 6.3, the menu bar of the program contains a File menu, a Calculate menu, and an Options menu. The File menu has two menu items, New and Exit. With the selection of the New item, the present beam and data are cleared. Selecting the Exit menu item selection causes the program to terminate.

The Calculate menu contains the Draw Influence Line menu item. When this menu item is selected, the Draw Influence Line DialogBox shown in Figure 6.4 is displayed. The desired response and the location of the response can be entered in this dialog box.

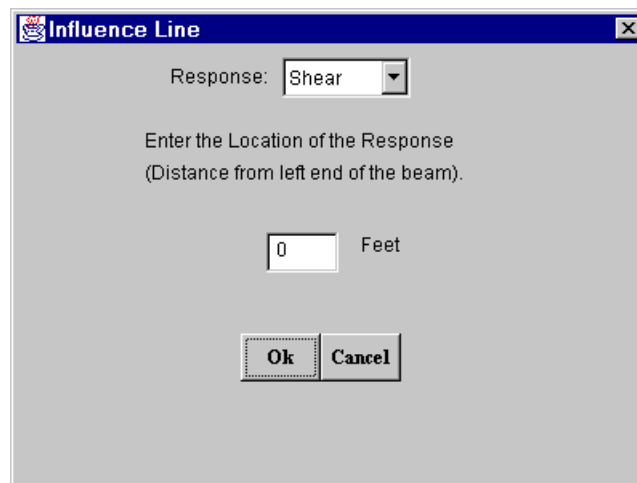


Figure 6.4 Draw Influence Line DialogBox.

The Options menu contains two menu items. The first item is Scale/Units. When this menu item is selected, theScale/Units DialogBox shown in Figure 6.5 is displayed. The maximum x coordinate, the number of interval positions of the unit load between spans, and the system of units may be changed using this dialog box.

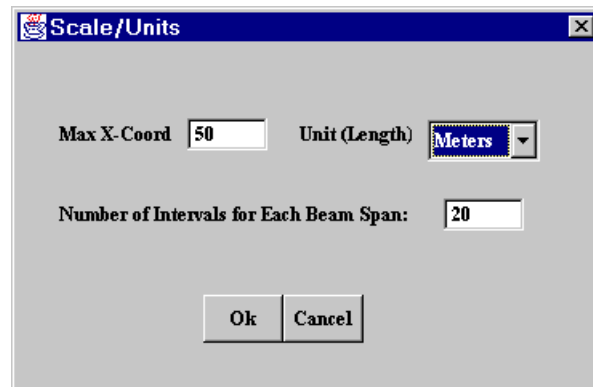


Figure 6.5 Scale/Units DialogBox.

The second menu item under the Options Selection is Show Output. When this menu item is selected, an output window showing influence line values at each interval point is displayed. The output window is shown in Figure 6.6.

The image shows a window titled "Influence Line Values" containing a table with two columns: "Distance (Feet)" and "Shear". The data points are as follows:

Distance (Feet)	Shear
0.00	1.00000
0.50	0.90000
1.00	0.80000
1.50	0.70000
2.00	0.60000
2.50	0.50000
3.00	0.40000
3.50	0.30000
4.00	0.20000
4.50	0.10000
5.00	0.00000

Figure 6.6 Influence Line Output.

6.4 Program Classes

6.4.1 Structural Classes

The same structural classes as those developed in the indeterminate frame program were used in this applet. These include the Support, the Joint, and the Member classes.

Additional structural classes developed for this applet are the Value and the Analysis

classes. Brief descriptions of these classes are given in Table 6.1. A description of the different methods used in the Analysis class is given in Table 6.2.

Table 6.1 Structural Classes Contained in the Influence Line Applet.

Class	Description
Analysis	All the analysis for the beam is done here.
Member	Represents members in the structure. This class contains data members for storing information about the start and end coordinates and member forces of a member.
Joint	Represents a joint in the beam. This class contains data members for storing the x and y coordinates of the joint.
Support	Represents a support. This class contains data members for storing the directions that are restrained and the different reaction values.
Value	Represents a load at the joint. The loads not acting at a joint are converted to joint loads.

Table 6.2 Methods in the Analysis Class of the Influence Line Applet.

Function	Description
CheckMaxCoordinate()	Computes the maximum x-coordinate. This value is needed for scaling the influence line diagrams.
setUpMemberVector()	Initializes member and joint vectors. These vectors are used to store joint and member data.
CalculateTotalRestrains()	Determines the total number of restraints on the beam.
CheckStability()	Checks the stability of the structure and displays an error message if the beam is unstable.
Stiffness()	Computes the global stiffness matrix for the beam.
elementStiffness()	Generates element stiffness matrices.
GenerateMCode()	Generates a matrix that relates member displacements to structure displacements.
Loading()	Checks for the presence of a unit load on the span and computes fixed end forces.
Guass()	Solves for joint displacements of the structure using Guass elimination.
storeDisplacements()	Saves the joint displacements obtained from solving the global system model.
memberEndForces()	Computes member end forces.
shearInfluence()	Determines the value of the ordinate for the influence diagram for shear.
momentInfluence()	Determines the value of the ordinate for the influence diagram for moment.

6.4.2 Interface Classes

The interface classes that were reused from the previous applets were the ManyLineLabel, Dialogbox, OutputFrame, IntTextfield, and DoubleTextField classes. The interface classes that were created for this applet are the Option, the InfluenceLineApplet, the InfluenceLineFrame, and the OutputFrame classes.

6.5 Program Verification

In this section, the results obtained from the Influence Line Applet for several example problems are compared with those obtained from other sources.

6.5.1 Example 1:

The first example taken from Hibbeler (1985) consists of a determinate beam with a unit load. The program was used to draw the influence line for the shear at point C (see Figure 6.7). As can be seen from Figures 6.8 and 6.9, the results obtained from the program are in agreement with those given by Hibbeler (1985).

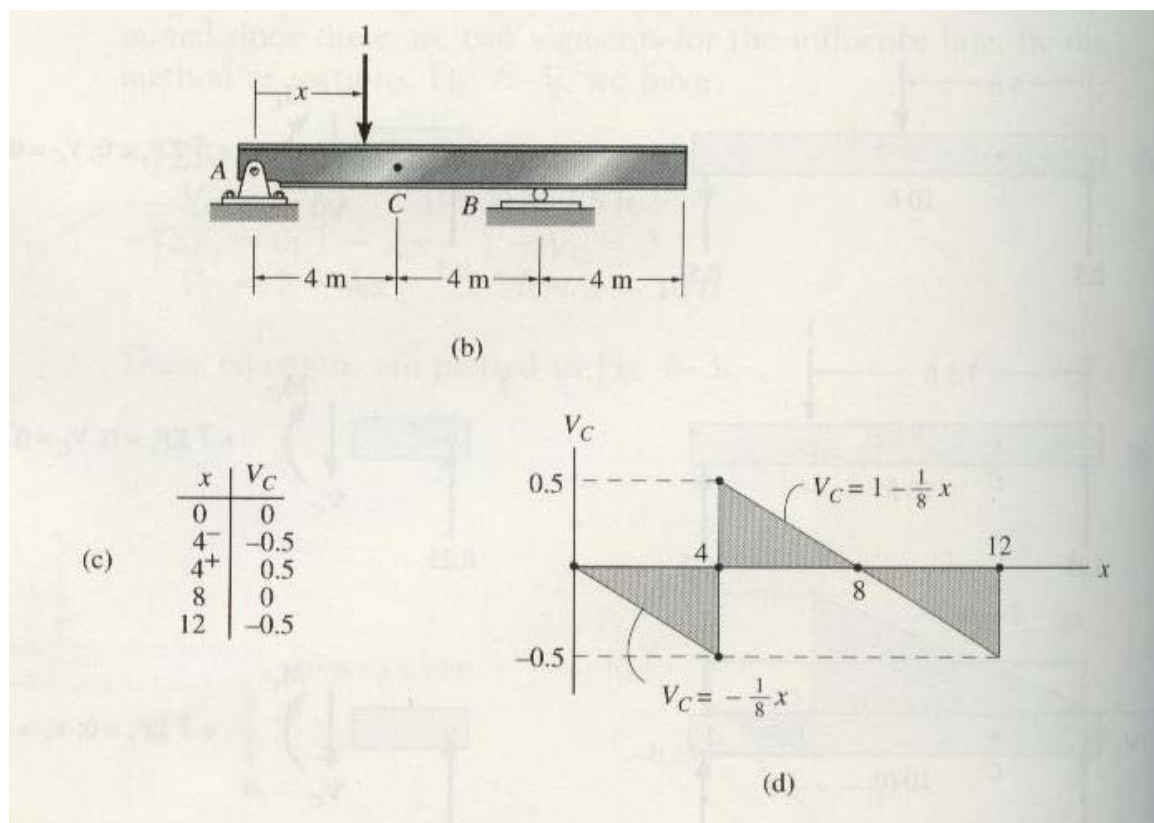


Figure 6.7 Determinate Beam (from Hibbeler, 1985).

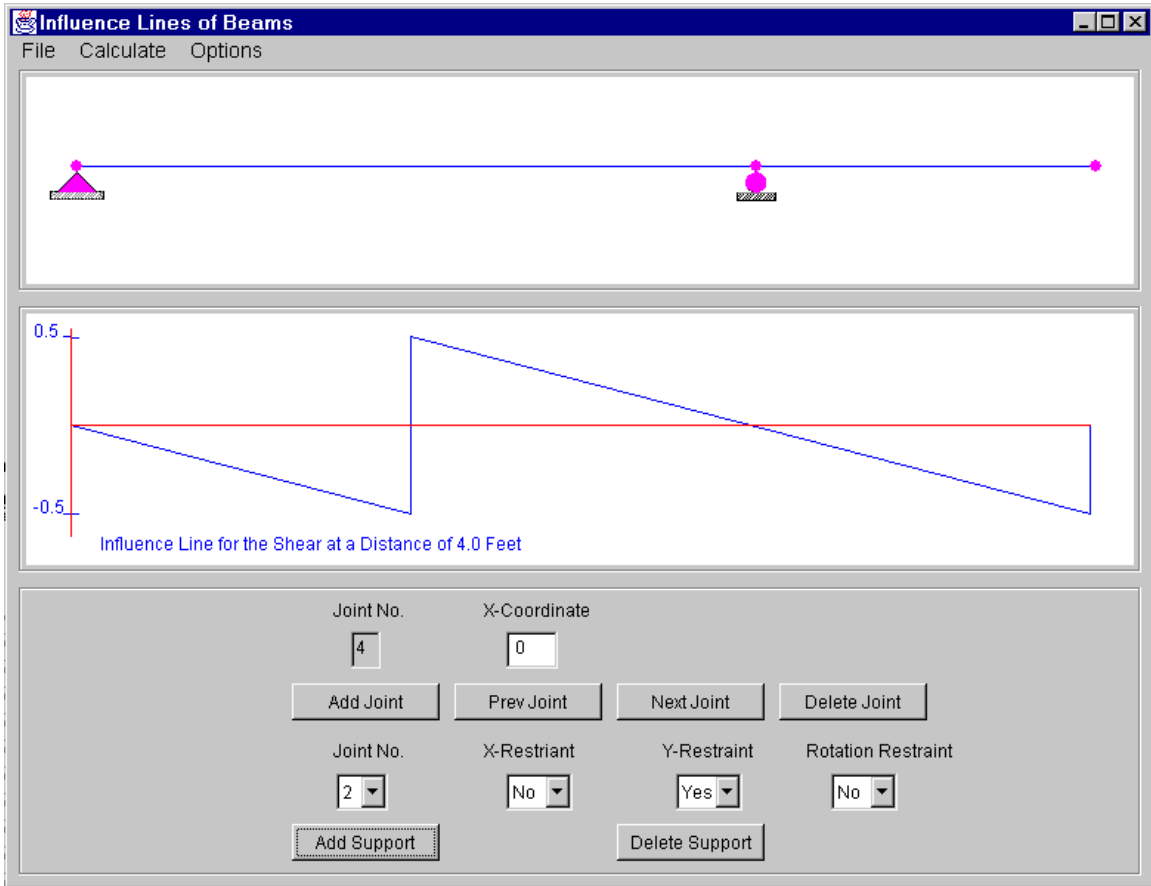


Figure 6.8 Influence Line for Shear Force for the Determinate Beam of Example 1.

Distance (Meters)	Shear
0.00	0.00000
4.00	-0.50000
4.00	0.50000
8.00	0.00000
8.00	0.00000
10.00	-0.25000
12.00	-0.50000

Figure 6.9 Influence Line Ordinates for Shear Force for the Determinate Beam of Example 1.

6.5.2 Example 2:

The second example consists of a three span continuous beam. All spans are equal length. The influence lines for the moment at four-tenths of the first span and at three-tenths of the middle span were computed. The beam geometry is shown in Figure 6.10. The output values for both cases are shown in Figures 6.11 and 6.12. The results were compared with those presented by Toniais (1995) (see Table 6.3). It is seen that the values are the same when rounded to the same number of significant digits. Also, it can be seen from Figure 6.10 that the influence line for the moment at four-tenths ($0.4L$) of the first span is drawn correctly by the applet.

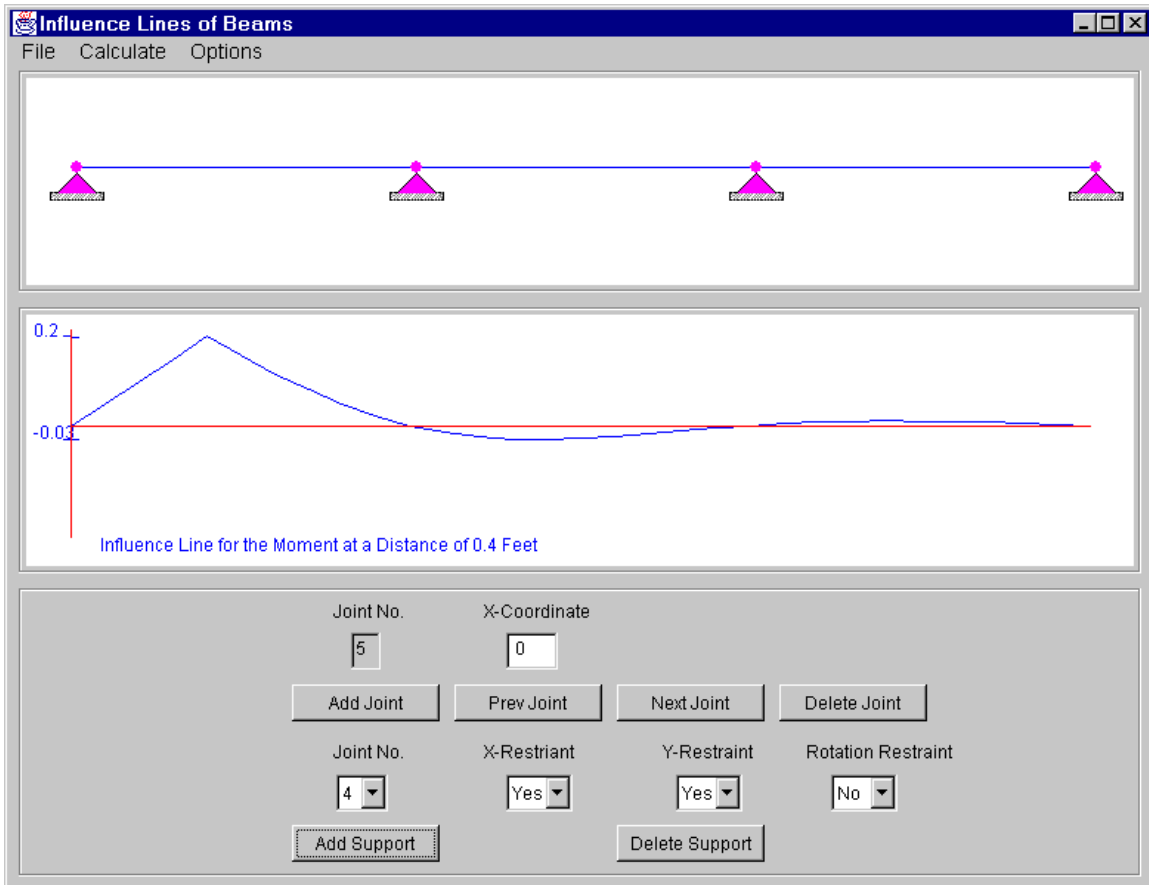


Figure 6.10 Influence Line for the Moment at 0.4 Times the Length of the First Span for the Continuous Beam With Three Equal Spans.

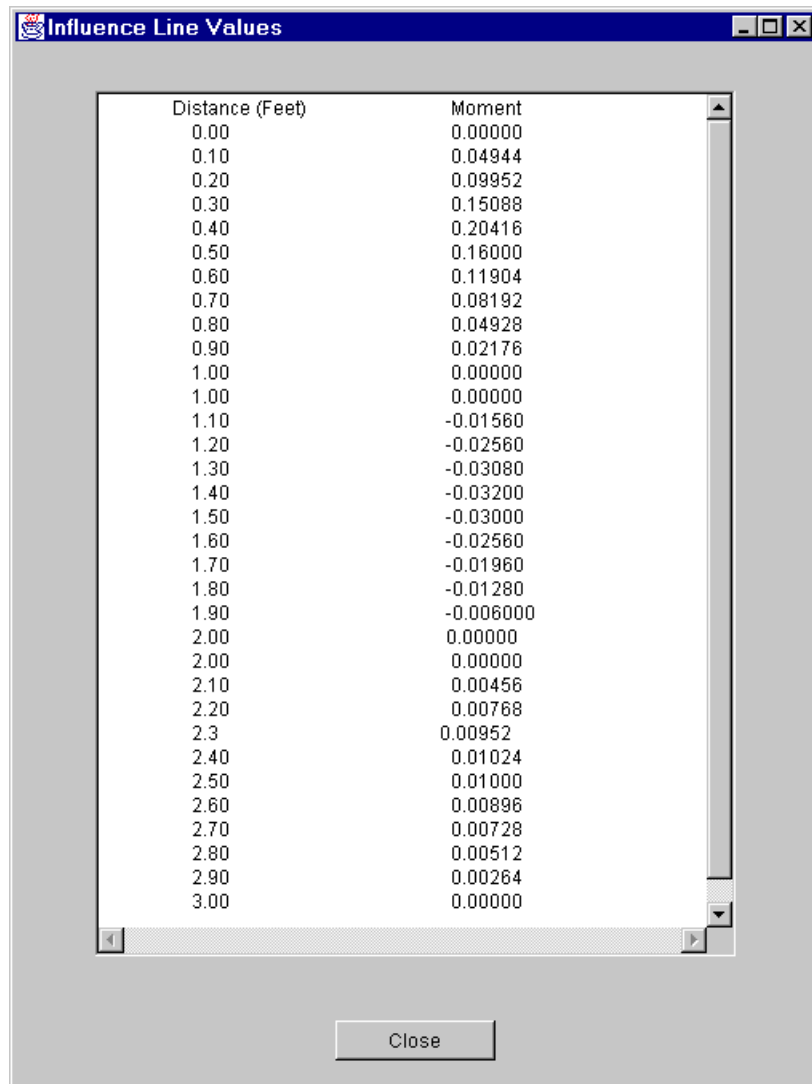


Figure 6.11 Moment Influence Ordinates at 0.4 Times the Length of the First Span for the Continuous Beam of Example 2.

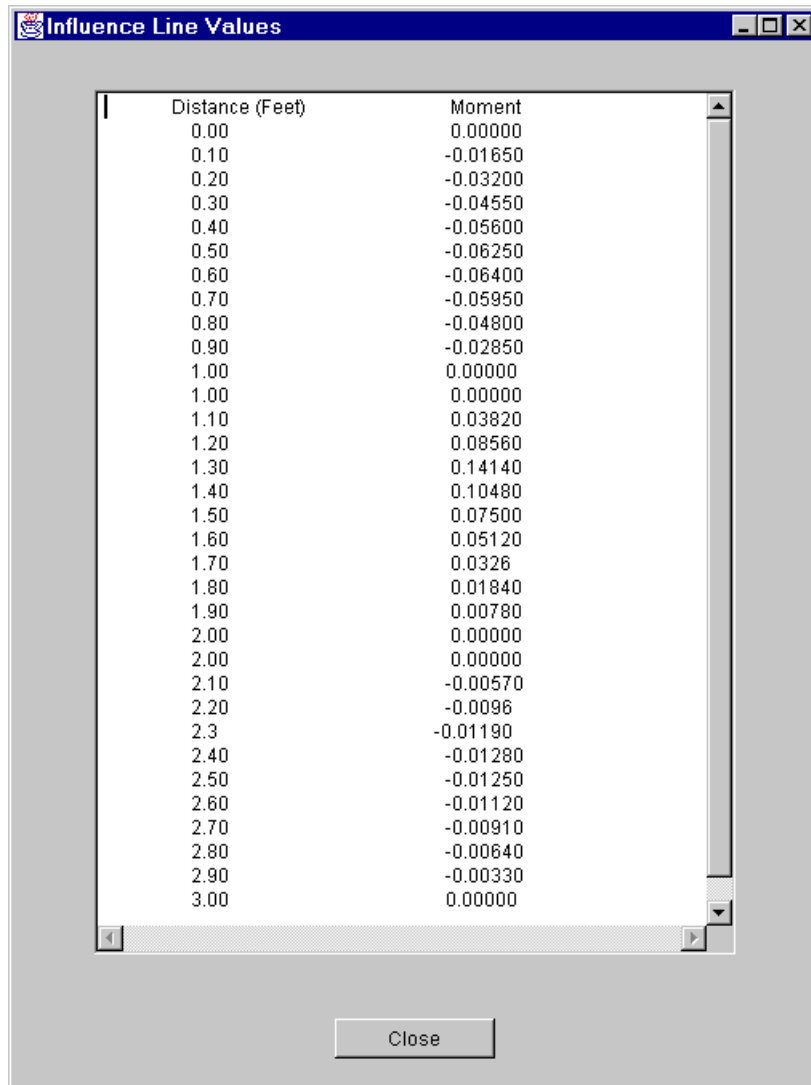


Figure 6.12 Moment Influence Ordinates at 0.3 Times the Length of the Middle Span for the Continuous Beam of Example 2.

Table 6.3 Moment Influence Ordinates for Continuous Beam With Three Equal Spans (Tonias, 1995)

Unit Load at	SPAN 1									SPAN 2											
	A	.1	.2	.3	.4	.5	.6	.7	.8	.9	B	.1	.2	.3	.4	.5	.6	.7	.8	.9	C
SPAN 1	A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	.1	0	.087	.075	.062	.049	.037	.024	.012	-.001	-.014	-.026	-.023	-.020	-.017	-.013	-.010	-.007	-.003	.000	.003
	.2	0	.075	.150	.125	.100	.074	.049	.024	-.001	-.026	-.051	-.045	-.038	-.032	-.026	-.019	-.013	-.006	.000	.006
	.3	0	.063	.125	.188	.151	.114	.076	.039	.002	-.036	-.073	-.064	-.055	-.046	-.035	-.027	-.018	-.009	.000	.009
	.4	0	.051	.102	.153	.204	.115	.105	.057	.008	-.041	-.090	-.078	-.067	-.056	-.045	-.034	-.022	-.011	.000	.011
	.5	0	.040	.080	.120	.160	.200	.140	.080	.020	-.040	-.100	-.088	-.075	-.063	-.050	-.038	-.025	-.013	.000	.013
	.6	0	.030	.060	.089	.119	.149	.179	.108	.038	-.032	-.102	-.090	-.077	-.064	-.051	-.038	-.026	-.013	.000	.013
	.7	0	.021	.041	.061	.082	.102	.123	.143	.064	-.016	-.095	-.083	-.071	-.060	-.048	-.036	-.024	-.012	.000	.012
	.8	0	.012	.025	.037	.049	.062	.074	.086	.099	.011	-.077	-.067	-.058	-.048	-.038	-.029	-.019	-.010	.000	.010
	.9	0	.003	.006	.009	.022	.027	.033	.038	.044	.049	-.046	-.040	-.034	-.029	-.023	-.017	-.011	-.006	.000	.006
	B	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SPAN 2	.1	0	-.004	-.008	-.012	-.016	-.020	-.023	-.027	-.031	-.035	-.039	.053	.046	.038	.031	.023	.015	.007	.000	-.007
	.2	0	-.006	-.013	-.019	-.026	-.032	-.038	-.045	-.051	-.058	-.064	.019	.102	.086	.069	.052	.035	.018	.002	-.015
	.3	0	-.008	-.015	-.023	-.031	-.039	-.046	-.054	-.062	-.064	-.077	-.004	.069	.141	.114	.087	.050	.033	.005	-.022
	.4	0	-.008	-.016	-.024	-.032	-.040	-.048	-.056	-.064	-.072	-.080	-.018	.043	.105	.166	.128	.090	.051	.013	-.026
	.5	0	-.008	-.015	-.023	-.030	-.038	-.045	-.053	-.060	-.068	-.075	-.025	.025	.075	.125	.175	.125	.075	.025	-.025
	.6	0	-.006	-.013	-.019	-.026	-.032	-.038	-.045	-.051	-.058	-.064	-.026	.013	.051	.090	.128	.166	.105	.043	-.018
	.7	0	-.005	-.010	-.015	-.020	-.025	-.029	-.034	-.039	-.044	-.049	-.022	.005	.033	.050	.087	.114	.141	.069	-.004
	.8	0	-.003	-.006	-.010	-.013	-.016	-.019	-.022	-.026	-.029	-.032	-.015	.002	.018	.035	.052	.069	.086	.102	-.019
	.9	0	-.002	-.003	-.005	-.006	-.008	-.009	-.011	-.012	-.014	-.015	-.007	.000	.008	.015	.023	.031	.038	.046	-.053
SPAN 3	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	.1	0	.001	.002	.003	.005	.006	.007	.008	.009	.010	.011	.006	.000	-.006	-.011	-.017	-.023	-.029	-.034	-.040
	.2	0	.002	.004	.006	.008	.010	.012	.013	.015	.017	.019	.010	.000	-.010	-.019	-.029	-.038	-.048	-.058	-.067
	.3	0	.002	.005	.007	.010	.012	.014	.017	.019	.021	.024	.012	.000	-.012	-.024	-.036	-.048	-.060	-.071	-.083
	.4	0	.003	.005	.008	.010	.013	.015	.018	.021	.023	.026	.013	.000	-.013	-.026	-.036	-.051	-.064	-.077	-.090
	.5	0	.003	.005	.008	.010	.013	.015	.018	.020	.023	.025	.013	.000	-.013	-.025	-.036	-.050	-.063	-.075	-.088
	.6	0	.002	.005	.007	.009	.011	.013	.016	.018	.020	.022	.011	.000	-.011	-.022	-.034	-.045	-.056	-.067	-.076
	.7	0	.002	.004	.006	.007	.009	.011	.013	.015	.016	.018	.009	.000	-.010	-.018	-.027	-.036	-.046	-.055	-.064
	.8	0	.001	.003	.004	.005	.006	.008	.009	.010	.012	.013	.006	.000	-.006	-.013	-.019	-.026	-.032	-.038	-.045
	.9	0	.001	.001	.002	.003	.003	.004	.005	.005	.006	.007	.003	.000	-.003	-.007	-.010	-.013	-.017	-.020	-.023
	D	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.5.3 Example 3

The third example considered is also a three span continuous beam. However, in this example (see Figure 6.13) the spans are unequal. The end spans are equal and the middle span has a length equal to 1.2 times that of the end spans. For this example the influence lines for the shear and reaction responses were examined. Table 6.4 shows the shear and moment ordinate values at various locations obtained from Barker (1997). The values generated by the computer program are exactly the same. The output from the program for the two cases is shown to demonstrate the accuracy of the results obtained from the program. The output from the program for shear at four-tenths of the first span of the beam, $V(104)$ is shown in Figure 6.14. The results for the moment at five-tenths of the second span, $M(205)$ are shown in Figure 6.15.

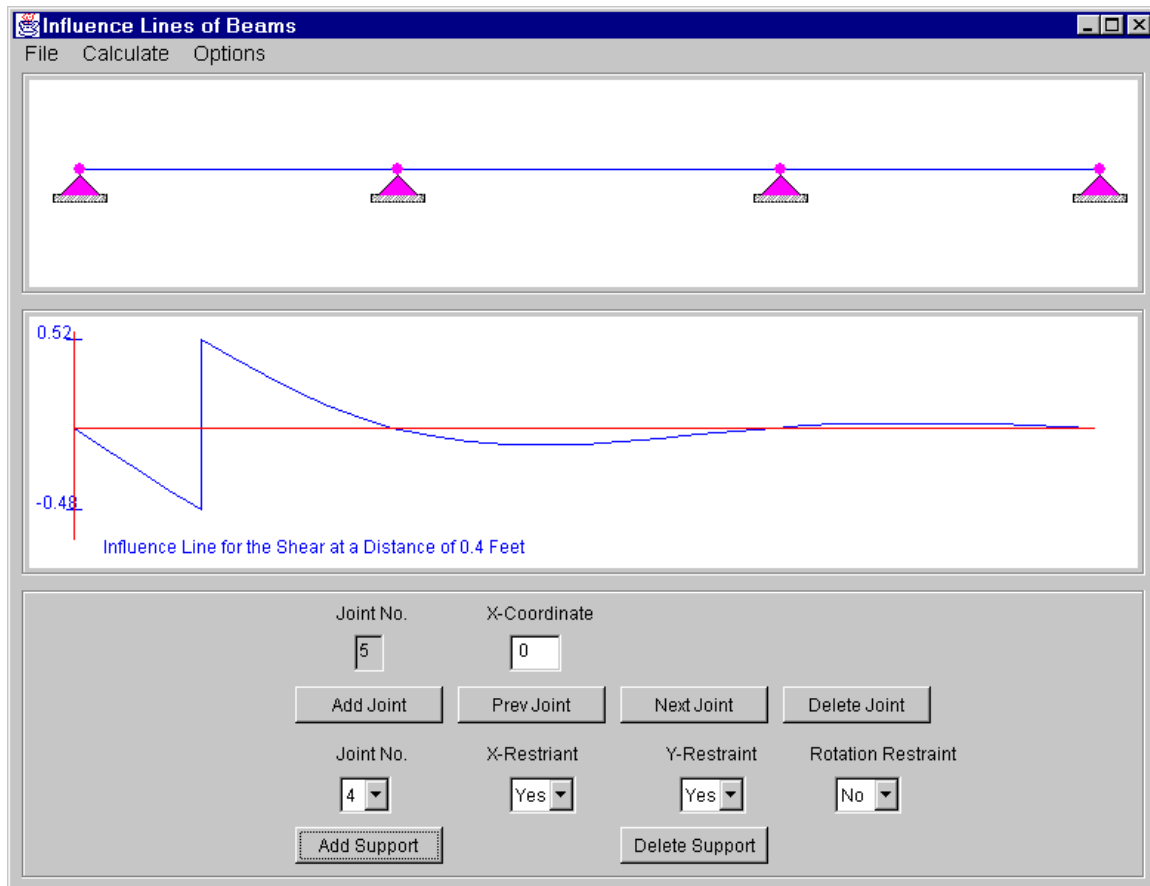


Figure 6.13 Three Span Beam, with a 1.2 Span Ratio.

**Table 6.4 Shear and Moment Ordinates for Three Span Beam with a
Span Ratio of 1.2 (Barker, 1997).**

Location	M(104)	M(200)	M(205)	V(100)	V(104)	V(110)	V(200)	V(205)
100	0.00000	0.00000	0.00000	1.00000	0.00000	0.00000	0.00000	0.00000
101	0.05028	-0.02431	-0.00884	0.87569	-0.12431	-0.12431	0.02578	0.02578
102	0.10114	-0.04714	-0.01714	0.75286	-0.24714	-0.24714	0.05000	0.05000
103	0.15319	-0.06703	-0.02437	0.63297	-0.36703	-0.36703	0.07109	0.07109
104	0.20700	-0.08250	-0.03000	0.51750	-0.48250	-0.48250	0.08750	0.08750
					0.51750			
105	0.16317	-0.09208	-0.03348	0.40792	0.40792	-0.59208	0.09766	0.09766
106	0.12229	-0.09429	-0.03429	0.30571	0.30571	-0.69429	0.10000	0.10000
107	0.08494	-0.08766	-0.03187	0.21234	0.21234	-0.78766	0.09297	0.09297
108	0.05171	-0.07071	-0.02571	0.12929	0.12929	-0.87071	0.07500	0.07500
109	0.02321	-0.04199	-0.01527	0.05801	0.05801	-0.94199	0.04453	0.04453
110	0.00000	0.00000	0.00000	0.00000	0.00000	-1.00000	0.00000	0.00000
200	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	1.00000	0.00000
201	-0.02037	-0.05091	0.02529	-0.05091	-0.05091	-0.05091	0.92700	-0.07300
202	-0.03333	-0.08331	0.05829	-0.08331	-0.08331	-0.08331	0.83600	-0.16400
203	-0.03996	-0.09990	0.09900	-0.09990	-0.09990	-0.09990	0.73150	-0.26850
204	-0.04135	-0.10337	0.14743	-0.10337	-0.10337	-0.10337	0.61800	-0.38200
205	-0.03857	-0.09643	0.20357	-0.09643	-0.09643	-0.09643	0.50000	-0.50000
								0.50000
206	-0.03271	-0.08177	0.14743	-0.08177	-0.08177	-0.08177	0.38200	0.38200
207	-0.02484	-0.06210	0.09900	-0.06210	-0.06210	-0.06210	0.26850	0.26850
208	-0.01650	-0.04011	0.05829	-0.04011	-0.04011	-0.04011	0.16400	0.16400
209	-0.00741	-0.01851	0.02529	-0.01851	-0.01851	-0.01851	0.07300	0.07300
210	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
300	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
301	0.00458	0.01145	-0.01527	0.01145	0.01145	0.01145	-0.04453	-0.04453
302	0.00771	0.01929	-0.02571	0.01929	0.01929	0.01929	-0.07500	-0.07500
303	0.00956	0.02391	-0.03188	0.02391	0.02391	0.02391	-0.09297	-0.09297
304	0.01029	0.02571	-0.03429	0.02571	0.02571	0.02571	-0.10000	-0.10000
305	0.01004	0.02511	-0.03348	0.02511	0.02511	0.02511	-0.09766	-0.09766
306	0.00900	0.02250	-0.03000	0.02250	0.02250	0.02250	-0.08750	-0.08750
307	0.00731	0.01828	-0.02437	0.01828	0.01828	0.01828	-0.07109	-0.07109
308	0.00514	0.01286	-0.01714	0.01286	0.01286	0.01286	-0.05000	-0.05000
309	0.00265	0.00663	-0.00884	0.00663	0.00663	0.00663	-0.02578	-0.25780
310	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

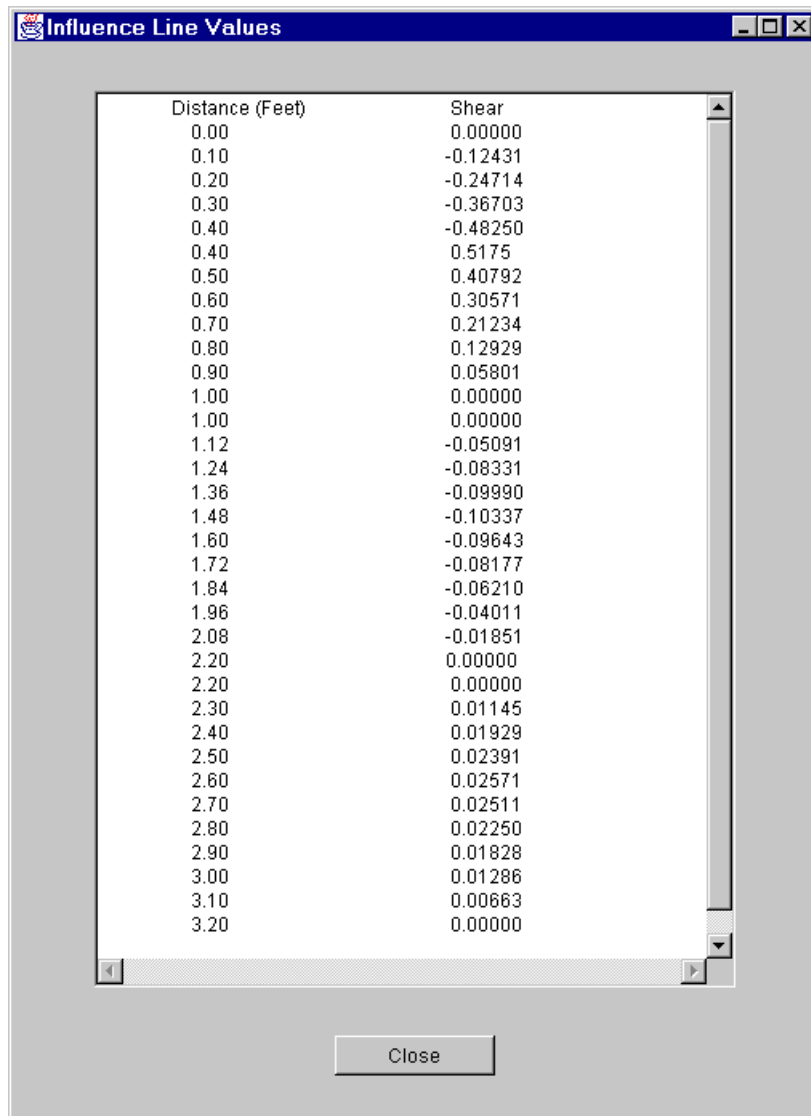


Figure 6.14 Output from Influence Line Applet for Shear at 0.4 Times the Length of the First Span V(104).

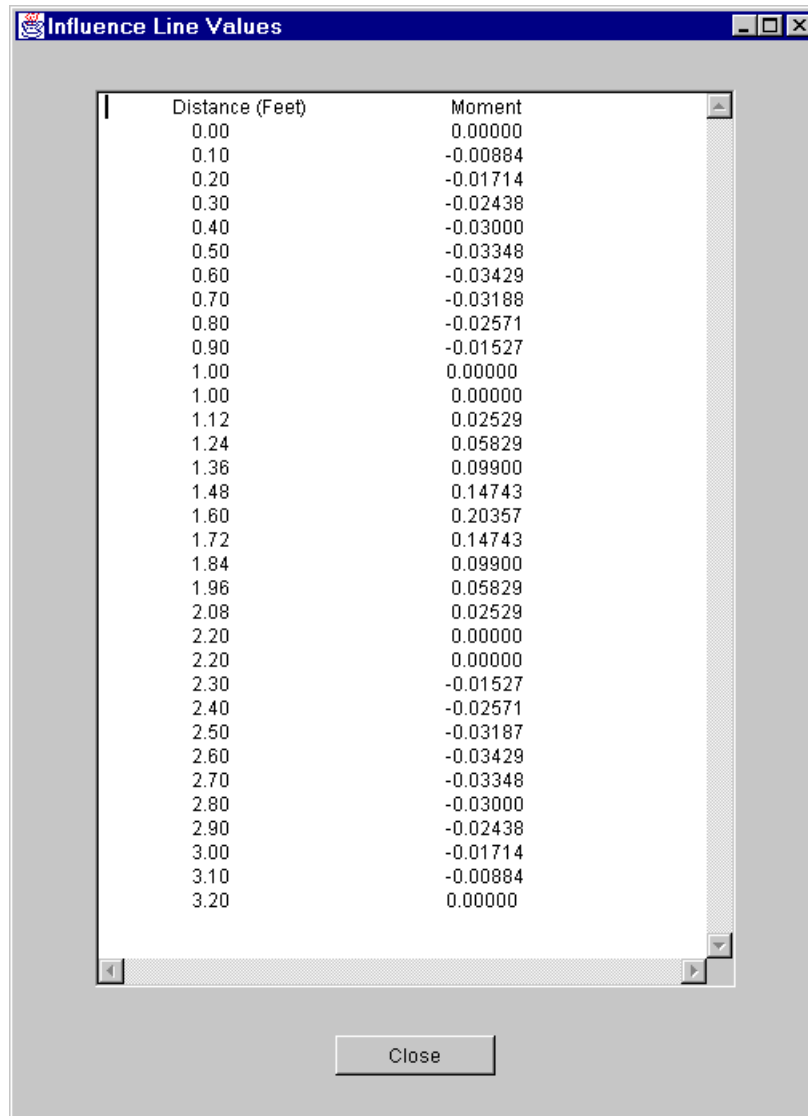


Figure 6.15 Output from Influence Line Applet for Moment at 0.5 Times the Length of the Second Span M(205).

6.5.4 Example 4

The fourth verification example is a four span continuous beam (see Figure 6.16). The two interior spans are 1.3 times the exterior spans. For this example influence lines for the support reactions were computed. These influence line ordinates were compared with those given in the Steel Construction Manual (AISC, 1986). The values in the Steel Construction Manual are listed in Table 6.5. The results obtained from the applet for the reaction at the first support are shown in Figure 6.17. The results for the reaction at the fourth support are given in Figure 6.18. As can be seen from Table 6.5 and Figures 6.17 and 6.18 the results from the applet and the values given by AISC (1986) are the same.

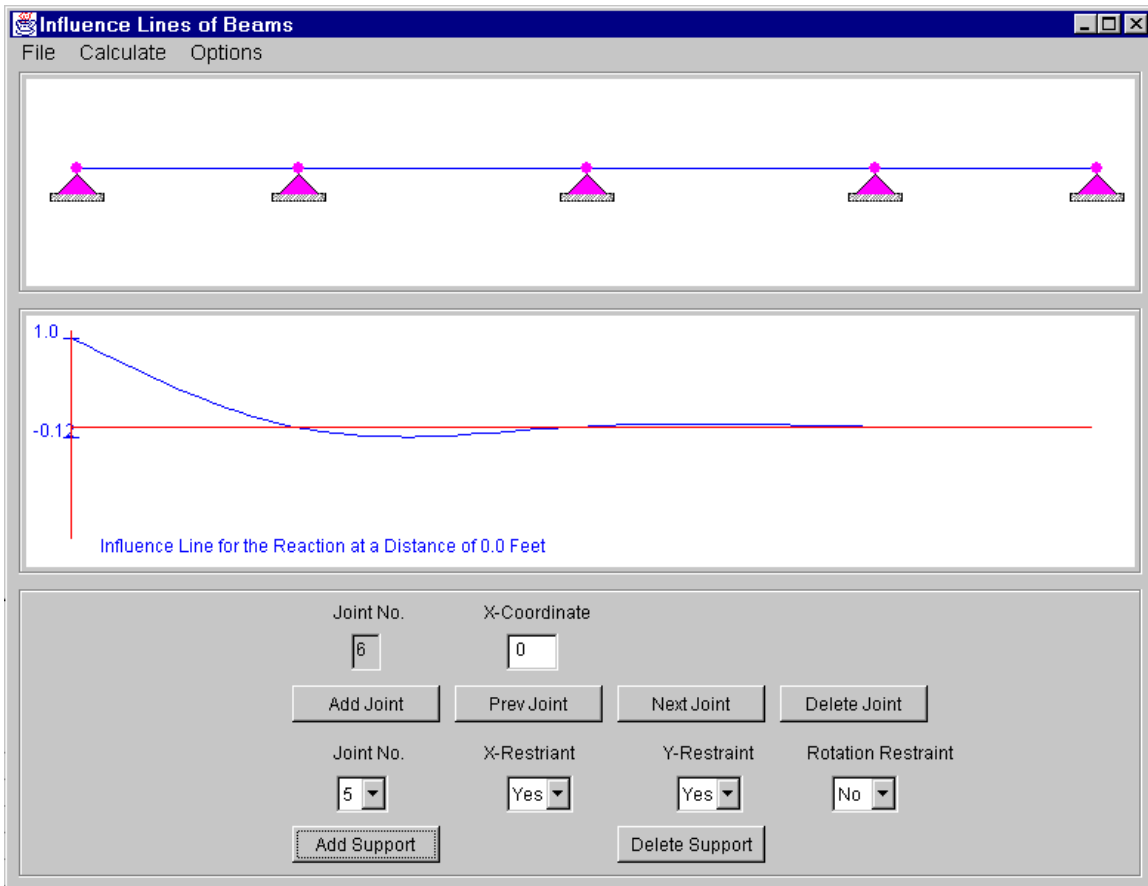


Figure 6.16 Four Span Beam.

Table 6.5 Influence Line Ordinates for Support Reactions for Four Span Beam.

Location	Ra	Rb	Rc	Rd	Re
a	1.0000	0.0000	0.0000	0.0000	0.0000
0.1	0.8767	0.1460	-0.0289	0.0079	-0.0018
0.2	0.7548	0.2893	-0.0561	0.0154	-0.0034
0.3	0.6358	0.4269	-0.0797	0.0219	-0.0049
0.4	0.5209	0.5562	-0.0981	0.0270	-0.0060
0.5	0.4118	0.6744	-0.1095	0.0301	-0.0067
0.6	0.3097	0.7785	-0.1122	0.0308	-0.0069
0.7	0.2160	0.8660	-0.1043	0.0287	-0.0064
0.8	0.1322	0.9339	-0.0841	0.0231	-0.0052
0.9	0.0598	0.9795	-0.0500	0.0137	-0.0031
b	0.0000	1.0000	0.0000	0.0000	0.0000
0.1	-0.0574	0.9868	0.0895	-0.0243	0.0054
0.2	0.0940	0.9341	0.2012	-0.0531	0.0118
0.3	-0.1127	0.8494	0.3276	-0.0827	0.0184
0.4	-0.1167	0.7403	0.4614	-0.1093	0.0243
0.5	-0.1090	0.6145	0.5949	-0.1292	0.0288
0.6	-0.0925	0.4795	0.7209	-0.1389	0.0309
0.7	-0.0704	0.3430	0.8318	-0.1344	0.0299
0.8	-0.0455	0.2125	0.9203	-0.1123	0.0250
0.9	-0.0211	0.0956	0.9788	-0.0687	0.0153
c	0.0000	0.0000	1.0000	0.0000	0.0000
0.1	0.0153	-0.0687	0.9788	0.0956	-0.0211
0.2	0.0250	-0.1123	0.9203	0.2125	-0.0455
0.3	0.0299	-0.1344	0.8318	0.3430	-0.0704
0.4	0.0309	-0.1389	0.7209	0.4795	-0.0925
0.5	0.0288	-0.1292	0.5949	0.6145	-0.1090
0.6	0.0243	-0.1093	0.4614	0.7403	-0.1167
0.7	0.0184	-0.0827	0.3276	0.8494	-0.1127
0.8	0.0118	-0.0531	0.2012	0.9341	-0.0940
0.9	0.0054	-0.0243	0.0895	0.9868	-0.0574
d	0.0000	0.0000	0.0000	1.0000	0.0000
0.1	-0.0031	0.0137	-0.0500	0.9795	0.0598
0.2	-0.0052	0.0231	-0.0841	0.9339	0.1322
0.3	-0.0064	0.0287	-0.1043	0.8660	0.2160
0.4	-0.0069	0.0308	-0.1122	0.7785	0.3097
0.5	-0.0067	0.0301	-0.1095	0.6744	0.4118
0.6	-0.0060	0.0270	-0.0981	0.5562	0.5209
0.7	-0.0049	0.0219	-0.0797	0.4269	0.6358
0.8	-0.0034	0.1540	-0.0561	0.2893	0.7548
0.9	-0.0018	0.0079	-0.2890	0.1460	0.8767
e	0.0000	0.0000	0.0000	0.0000	1.0000

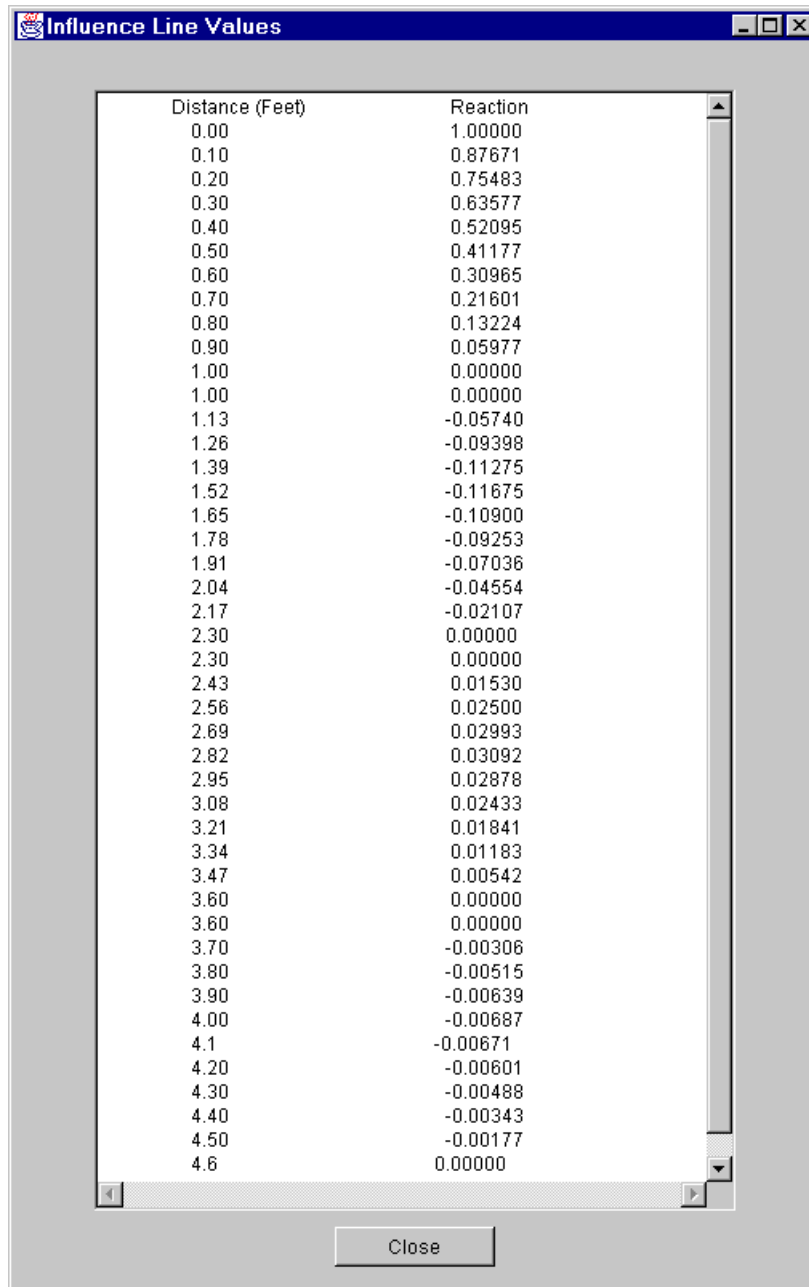


Figure 6.17 Results from Influence Line Applet for Reaction at the First Support for Four Span Beam Example.

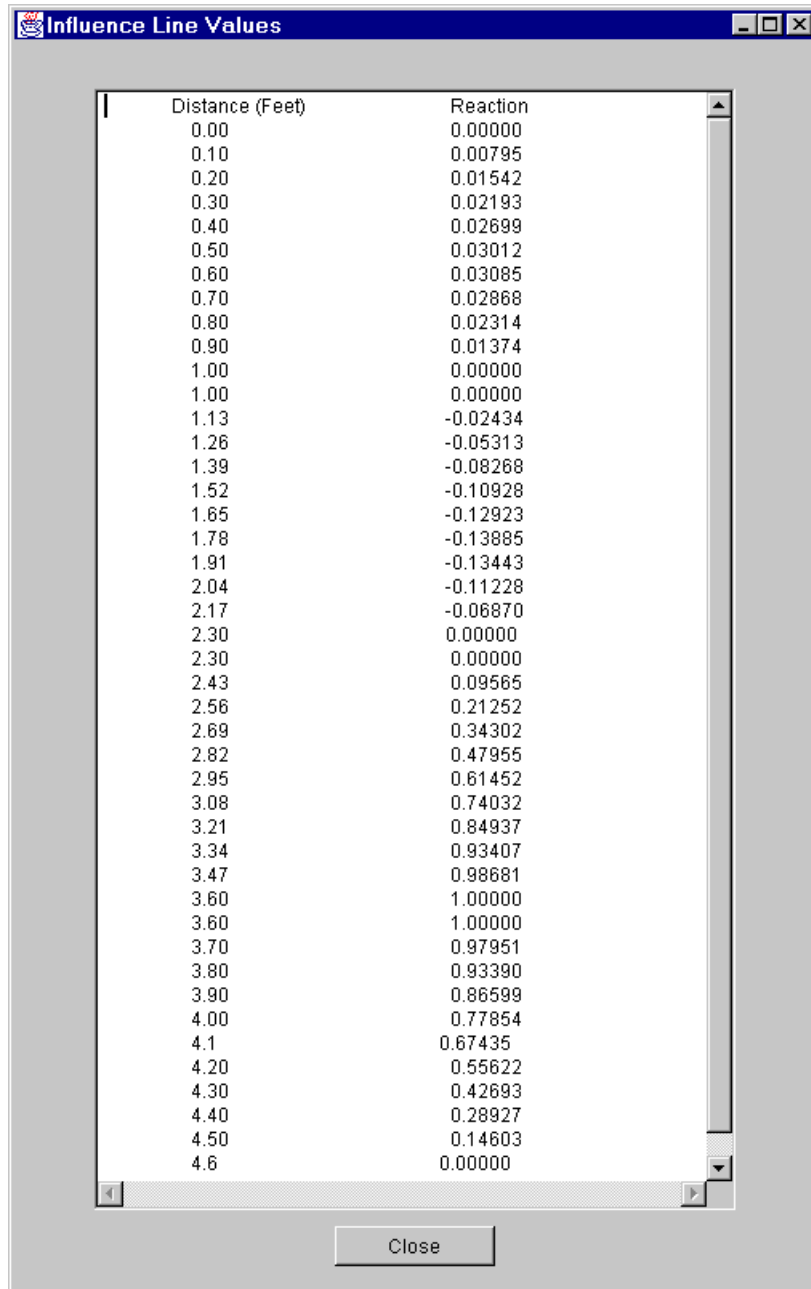


Figure 6.18 Results from Influence Line Applet for Reaction at the Fourth Support for Four Span Beam Example.

Chapter 7

Summary and Conclusions

The objective of this thesis was to develop a series of web based Java applets to assist engineering students in the understanding of fundamental structural engineering concepts. The topics covered include the determinacy of structures, analysis of determinate and indeterminate trusses, beams and frames and influence line analysis. This chapter presents a summary of these topics and conclusions from the development of the applets.

7.1 Summary

The first set of applets, the truss and frame determinacy applets test a student's understanding of the fundamental concept of stability and determinacy of a structure. These applets are interactive. A student can enter and check their answers for the truss or frame being displayed. The first set of determinacy applets also includes a three-hinged arch bridge applet. This applet enables a student to solve for support reactions in a determinate structure such as a determinate arch bridge with a uniform loading.

The determinate and indeterminate truss analysis applets demonstrate how to analyze different trusses. These applets show the results of the member forces and reactions of an entered truss. A user can easily gain an understanding of how the joint loads are distributed through a truss, without having to do hand computations. The determinate and indeterminate frame analysis applets demonstrate the analysis of different frame structures. The frame applets give reactions and member forces results for a frame with joint and member loads.

Finally, the last applet, the influence applet, draws the influence line responses of continuous beams. The beam geometry and the location and type of response required can be entered. The beam is analyzed and the influence line drawn for the selected

response. A table of the results can also be displayed. This applet demonstrates to a student the internal forces that need to be considered as a load moves across a beam.

7.2 Conclusions

Using the object oriented programming language Java to write these applets had several benefits. One benefit was the use of real world objects such as joint, support, and member. This simplified the development of the applets. Another was that a number of classes representing structural objects were reused in creating the different applets. Having these applets made up of real world objects made the programs easier to understand and debug. Also, the use of Java libraries reduced the amount of time and effort needed to create the graphic user interface. But perhaps the greatest benefit of using Java for the development of these structural analysis programs is that Java is both platform and operating system independent. This allows the applets to be accessed by any computer with Internet access without difficulty and hence, makes these tutorial applets widely available.

There are some limitations to the applets that were developed. The truss and frame applets are limited in the complexity of the structures that can be solved, even though they were developed using the same methods of analysis of large, more complex programs. These applets could be further developed to be more comparable to commercial software programs; but it was not necessary for the scope of this project. For instance, the scope of this project includes the solving of member forces but not the checking of members for overstress, buckling or excessive deflection. The inclusion of these checks would not require much additional work in that most of the required user interface and analysis have been programmed. These enhancements would make these applets more useful especially in structural design courses.

Future development of these applets could enhance the quality of interaction by giving the user a better understanding of basic structural engineering concepts. Some possible future modifications of the applets developed would be the capability to draw shear and

moment diagrams for the frame applets. Another possible modification of the frame applets would be to show the deflected shape of the frame.

The applets developed serve the purpose of being interactive, educational tools in the understanding of fundamental structural engineering concepts. These applets are accessible to students through the Internet allowing them easy access and the opportunity to learn at their own pace.

References

- American Association of Steel Construction (AISC) (1986), Manual of Steel Construction, Load and Resistance Factor Design (LFRD), Chicago, Illinois.
- Barker, Richard M. (1997), Design of Highway Bridges, John Wiley & Sons, Inc., New York, New York.
- Fleming, John F. (1997), Analysis of Structural Systems, Prentice-Hall, Inc., Upper Saddle River, New Jersey.
- Foley, Christopher M. and Schinler, D. (2003), “Automated Design of Steel Frames Using Advanced Analysis and Object-Oriented Evolutionary Computation”, Journal of Structural Engineering, May 2003.
- Gao, Yong (2003), “Java Powered Virtual Laboratory for Nonlinear Structural Dynamic Analysis”, University of Illinois at Urbana Champaign.
- Haque, M.E. (2001), “Web-based Visualizaton Techniques for Structural Design Education”, Proceedings of the 2001 American Society for Engineering Education Annual Conference Exposition.
- Hibbeler, Russell C. (1997), Structural Analysis, Prentice-Hall, Inc., Upper Saddle River, New Jersey.
- Holzer, S.M. (1998), “Computer Analysis of Structures Course Notes,” Virginia Polytechnic Institute and State University, Department of Civil and Environmental Engineering.
- Jiang, H., Kurama, Y. C., Fanella, D.A. (2002), “WWW-Based Virtual Laboratories for Reinforced Concrete Education”, University of Notre Dame.
- Kassamali, A. (1990), Structural Analysis, Second Edition. PWS Publishing.

Kennedy, J.B., and Madugula, M.K.S. (1990), Elastic Analysis of Structures, Classical and Matrix Methods, Harper and Row Publishers, New York.

Kinney, J.S. (1957), Intermediate Structural Analysis, Addison-Wesley, Reading, Massachusetts.

Mishra, Trinayana (2001), "Real -Time Communication and Control of a Sensored Physical Environment", Massachusetts Institute of Technology.

RISA Technologies (2001), "RISA-2D Users Manual".

Roark, Raymond J. (1965), Formulas for Stress and Strain, McGraw-Hill, Inc., New York, New York.

Rojiani, K. B., Kim, Y.Y., and Kapania, R.K. (2000), "Web-Based Java Applets for Teaching Engineering Mechanics." Proceedings, ASEE Annual Conference, June, St. Louis, MO, Session 2620.

Sun Microsystems, Inc. (2003), <http://www.java.sun.com>.

Tonias, Demetrios E. (1995), Bridge Engineering: Design, Rehabilitation, and Maintenance of Modern Highway Bridges, McGraw-Hill, Inc., New York, New York.

Vita

Robert Schottler was born on July 11, 1974 in Washington, D.C. He attended Virginia Polytechnic Institute and State University and obtained his Bachelor of Science degree in Civil Engineering in 1997. He later attended Virginia Polytechnic Institute and State University and earned his Master of Science degree in Civil Engineering in 2004.

Robert Schottler