

Análisis estructural de marcos
rígidos por el método de rigidez
usando MATLAB

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería

Análisis estructural de marcos
rígidos por el método de rigidez
usando MATLAB

Trabajo de investigación presentado por
Carlos Alexander Becker Samayoa para optar
al grado académico de Licenciado en
Ingeniería Civil

BIBLIOTECA
DE LA
UNIVERSIDAD DEL VALLE DE GUATEMALA

Guatemala
2005

Guatemala, 3 de Mayo de 2005

Ing. Franklin Matzdorf
Director del Departamento de Ingeniería Civil
Facultad de Ingeniería
Universidad del Valle
Presente

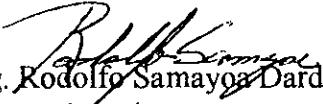
Señor Director:

Por este medio le comunico a usted que he asesorado el trabajo de tesis titulado: **“ANÁLISIS ESTRUCTURAL DE MARCOS RÍGIDOS POR EL MÉTODO DE RIGIDEZ USANDO MATLAB”**, que fue desarrollado por Carlos Alexander Becker Samayoa previo a obtener el título de Ingeniero Civil.

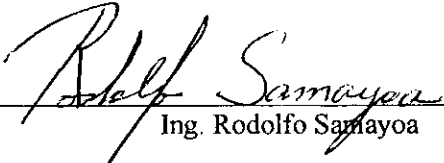
En mi calidad de asesor, he procedido a supervisar las etapas de desarrollo del marco teórico, recopilación de información, organización de datos, análisis de resultados, elaboración del programa en computadora, discusión de resultados y elaboración del informe final, por lo que le comunico a usted mi entera satisfacción del trabajo final para que el estudiante Carlos Alexander Becker pueda realizar su Examen de Graduación.

En espera de que usted quede igualmente satisfecho con el trabajo, me suscribo deseándole éxitos en todas sus actividades.


Atentamente,



Ing. Rodolfo Samayoa Dardón
Asesor de tesis


Vo.Bo.:

(f) 
Ing. Rodolfo Samayoa

Tribunal:

(f) 
Ing. Franklin Matzdorf

(f) 
Arq. María Elena Ortiz

(f) 
Ing. Rodolfo Samayoa

Fecha de aprobación: 14 de Junio, 2005

PREFACIO

Durante mis estudios de ingeniería en los cuales recibí cursos de matemáticas, computación, ciencias básicas y análisis estructural, por un lado aprendí a programar computadoras, por otro lado aprendí a plantear y resolver ecuaciones usando matrices y, por último, aprendí los métodos que se usan en el análisis estructural.

En un momento sentí que me hacía falta integrar todos los conocimientos que había adquirido durante mis estudios para facilitarme los cálculos estructurales. Es bien sabido que existen programas ya escritos por profesionales que resuelven muchos problemas, sin embargo, estos programas son muy caros donde se deben de pagar las licencias o son programas de difícil acceso y en muchos casos no hacen exactamente lo que uno quiere o necesita.

Por lo anterior, considero necesario que todo ingeniero debería ser capaz de integrar sus conocimientos para poder escribir programas en computadoras que se ajusten exactamente a sus necesidades y requerimientos a un costo muy bajo. Con esto podría hacer una combinación óptima de programas escritos por él mismo con programas escritos por otra persona.

Los ingenieros que escriben sus propios programas son generalmente mejores usuarios de programas que escriben otras personas. Todo esto sirvió de motivación para buscar los compiladores más versátiles junto con los métodos más completos para realizar cálculos estructurales.

Espero que este trabajo de tesis contribuya a fomentar la buena práctica de fomentar el uso de la teoría estructural en toda su potencia junto con el poder de cálculo de las computadoras modernas. Recordemos que hace unas décadas la capacidad de computación era muy limitada y por ende se tenía que recurrir a otras formas de cálculo. Pero ahora las computadoras personales ofrecen una capacidad de cálculo muy grande como se evidencia en muchos lados incluyendo esta tesis.

Quisiera aprovechar para agradecer a todas las personas que contribuyeron en mi formación y en especial a mis catedráticos de ingeniería estructural quienes me fueron guiando para que este trabajo resultara exitoso.

ÍNDICE

	Página
PREFACIO	v
ÍNDICE	vi
LISTA DE TABLAS	vii
LISTA DE FIGURAS	viii
RESUMEN	xi
 Capítulos	
I. INTRODUCCIÓN	1
II. OBJETIVOS	3
III. CONCEPTOS FUNDAMENTALES DEL ANÁLISIS	
ESTRUCTURAL	5
IV. MÉTODO DE RIGIDEZ	37
V. MATLAB	71
VI. PROGRAMA PARA CÁLCULO DE MARCOS	111
VII. CONCLUSIONES Y RECOMENDACIONES	149
VIII. BIBLIOGRAFÍA Y REFERENCIAS	151
IX. GLOSARIO DE TÉRMINOS	153

LISTA DE TABLAS

Tabla	Página
1. Reacciones de empotramiento	57
2. Comparación del método de rigidez con el de flexibilidad.....	69
3. Nombres y valores de variables en MATLAB	82
4. Funciones que crean matrices	89
5. Operaciones aritméticas entre escalares.....	91
6. Jerarquía de las operaciones aritméticas	91
7. Operaciones entre arreglos y matrices	92
8. Definición de las operaciones entre arreglos	93
9. Operadores relacionales	97
10. Operadores lógicos	99
11. Tabla de verdad para los operadores lógicos	100
12. Símbolos de los colores de línea en MATLAB	106
13. Símbolos de los estilos de marca en MATLAB	107
14. Símbolos de los estilos de línea en MATLAB	107
15. Valores de pos en el comando legend	107
16. Comandos de la función axis	109
17. Funciones adicionales para líneas y marcas	109
18. Comparación entre MATLAB y otros compiladores	110
19. Información en la matriz de nodos	116
20. Información en la matriz de miembros.....	117

LISTA DE FIGURAS

Figura	Página
1. Desplazamientos y giro en dos dimensiones.....	6
2. Viga en voladizo.....	6
3. Fuerzas y momento en dos dimensiones.....	7
4. Fuerzas y momentos en tres dimensiones.....	7
5. Cuerpo no rígido.....	9
6. Relación carga-desplazamiento.....	9
7. Ejemplos de trabajo externo.....	11
8. Unidad de volumen.....	12
9. Elemento sujeto al caso general de esfuerzos.....	13
10. Elemento diferencial sujeto a esfuerzos.....	18
11. Fuerzas sobre una barra.....	19
12. Dos sistemas de carga.....	23
13. Teorema de Maxwell.....	25
14. Igualdad de desplazamientos.....	26
15. Cuerpo sujeto a sistema de cargas.....	27
16. Coeficientes de flexibilidad.....	32
17. Simetría de coeficientes de flexibilidad.....	34
18. Coeficiente de influencia de rigidez.....	35
19. Estructuras libres de una viga.....	38
20. Ejemplo de redundantes.....	39
21. Viga de dos luces.....	42
22. Viga de dos luces mostrando rigideces.....	44
23. Grados de libertad de miembro prismático.....	46
24. Primera etapa del cálculo de rigideces.....	46
25. Segunda etapa del cálculo de rigideces.....	47
26. Tercera etapa del cálculo de rigideces.....	49
27. Miembro doblemente empotrado.....	51
28. Miembro doblemente empotrado con carga distribuida.....	53

29. Miembro doblemente empotrado con carga puntual.....	54
30. Vector en el plano.....	56
31. Rotación de ejes.....	58
32. Miembro prismático inclinado un ángulo θ	60
33. Nodos A y B referidos a dos sistemas.....	61
34. Matriz de rigidez total.....	63
35. Reacciones para formar vectores F_T y R	65
36. Miembro prismático sometido a un sistema de cargas.....	66
37. Miembro prismático con nodos restringidos.....	67
38. Tipos de apoyo.....	69
39. Grados de indeterminación.....	70
40. Ventana frontal de MATLAB.....	74
41. Menú para la ventana de comandos.....	76
42. Ventana de la historia de comandos.....	77
43. Ventana del espacio de trabajo.....	79
44. Ventana del editor de trabajo.....	79
45. Help Browser.....	80
46. Vector columna.....	85
47. Vector fila.....	85
48. Matriz.....	85
49. Arreglo tridimensional.....	86
50. Procedimiento para programar.....	96
51. Ploteo de la función $y = x^2 - 10x + 15$ de 0 a 10.....	104
52. Ploteo de la función $y(x)$ con título, etiquetas y rejilla.....	105
53. Ploteo de $f(x) = \sin(2x)$ y $f'(x) = 2 \cos(2x)$	106
54. Ploteo completo con título, etiquetas, leyenda, rejilla y varios estilos... ..	108
55. Ploteo de una línea recta usando LineWidth.....	110
56. Diagrama de flujo general del programa.....	113
57. Diagrama de flujo para el ingreso de datos.....	114
58. Matriz de nodos.....	116
59. Matriz de miembros.....	118

60. Figura del marco rígido 1.....	120
61. Marco rígido de dos niveles y tres luces.....	122
62. Ejemplo 1.....	131
63. Figura que muestra el programa usando “axis square”.....	133
64. Ejemplo 2.....	134
65. Figura que muestra el programa sin usar “axis square”.....	136
66. Ejemplo 3.....	136
67. Figura del ejemplo 3.....	138
68. Ejemplo 4.....	139
69. Figura del ejemplo 4.....	141
70. Ejemplo 5.....	142
71. Figura del ejemplo 5.....	143
72. Marco de varios niveles y luces.....	145
73. Ejemplo 7.....	147

RESUMEN

En este trabajo de tesis se prueba la utilidad de MATLAB para el cálculo de estructuras, así como también la supremacía y versatilidad del método de rigidez sobre otros métodos en el análisis estructural. Se hace especial énfasis en el cálculo de marcos rígidos. Lo más importante es hacer ver la utilidad de la combinación del método de rigidez con MATLAB. Ya que el método de rigidez sin una herramienta para realizar cálculos rápidos se puede volver inoperante.

Se desarrolla la teoría del método de rigidez basándose en los fundamentos del análisis estructural conformado por los métodos de energía. Es por eso que se enuncian y se demuestran los teoremas de Castigliano, Betti y Maxwell, los cuales permiten una mejor comprensión del método de rigidez. Luego de presentar los aspectos importantes del análisis estructural se explica el método de rigidez llegando a su ecuación general que en forma matricial se escribe así

$$\left(\begin{array}{c|c} K & K_{LR} \\ \hline K_{RL} & K_{RR} \end{array} \right) \left(\begin{array}{c} D \\ D_R \end{array} \right) + \left(\begin{array}{c} F \\ F_R \end{array} \right) = \left(\begin{array}{c} 0 \\ R \end{array} \right).$$

Se puede decir que en la ecuación anterior está toda la información necesaria del método de rigidez. Más adelante se explica el uso de MATLAB, el cual funciona como un excelente interprete y facilita de gran manera la programación de la computadora.

Los resultados que se obtuvieron son muy satisfactorios. Los cálculos se realizan rápidamente por lo que el factor tiempo ya no es un inconveniente aún para estructuras muy grandes. Editar y modificar el programa es también muy fácil por lo que el usuario es capaz de realizar cualquier cálculo que él requiera. El tamaño de las estructuras, en particular de los marcos rígidos, que permite MATLAB puede ser tan grande como la estructura más grande que se ve en la actualidad. En conclusión, el método de rigidez programado en MATLAB resulta fácil de usar y capaz de realizar cálculos muy complicados.

en este trabajo de tesis, se ha escogido para realizar el programa un nuevo software conocido como MATLAB.

El MATLAB facilita aún más que los compiladores la programación de computadoras. En esta tesis se exponen las características básicas del MATLAB y se explica cómo programar en este entorno.

Se puede decir en forma resumida que esta tesis consta del análisis de marcos rígidos por medio del método de rigidez usando MATLAB para efectuar los cálculos y obtener resultados.

Esta tesis está estructurada en cuatro capítulos para dar una exposición completa de la teoría de estructuras sobre la cual se basan todos los métodos del análisis estructural incluyendo el método de rigidez. Además se explican los elementos necesarios para la programación de computadoras.

En el primer capítulo se expone el fundamento del análisis estructural sobre el cual se desarrollan el método de flexibilidad y el método de rigidez. En este capítulo también se ordenan las ideas de desplazamientos y fuerzas que se usan directamente en los métodos de rigidez y flexibilidad.

En el segundo capítulo se desarrolla el método de rigidez que es la parte central de esta tesis. Junto con el método de rigidez se da una breve introducción al método de flexibilidad para hacer el contraste entre los dos métodos y dejar clara la razón del por qué se escoge el método de rigidez en lugar del método de flexibilidad. La exposición del método de rigidez es bastante completa y la notación que se usa en el segundo capítulo es la que se usa en el programa.

En el tercer capítulo se introduce a MATLAB y se explican todas las instrucciones y comandos que son utilizadas en el desarrollo del programa. Se hace notar la flexibilidad y sencillez de este software para escribir programas de computadoras.

Y en el último capítulo se explica con cierto detalle el programa que se ha escrito para realizar los cálculos. Al final de este capítulo se presentan varios ejemplos de marcos rígidos para ilustrar el uso del programa y su exactitud en los cálculos.

II. OBJETIVOS

Objetivo general:

Mostrar la utilidad del MATLAB para el cálculo estructural.

Objetivos específicos:

1. Explicar el método de rigidez.
2. Explicar el método de flexibilidad.
3. Entender la relación entre rigidez y flexibilidad.
4. Desarrollar un algoritmo para el cálculo de marcos rígidos usando el método de rigidez.
5. Conocer el software MATLAB.
6. Escribir un programa, dando su código en MATLAB, para efectuar los cálculos estructurales.
7. Ejemplificar con varios marcos rígidos el uso del programa en MATLAB.

III. CONCEPTOS FUNDAMENTALES DEL ANÁLISIS ESTRUCTURAL

3.1 Fuerzas y desplazamientos

Cuando una estructura está bajo la acción de fuerzas, esta estructura se deforma y como consecuencia los puntos dentro de la estructura se desplazarán hacia nuevas posiciones. En general, todos los puntos de la estructura sufrirán desplazamientos excepto, posiblemente, los puntos de apoyo.

El cálculo de los desplazamientos es una parte esencial del análisis estructural. En los métodos de rigidez y flexibilidad se calculan los desplazamientos de ciertos puntos importantes llamados nodos. Íntimamente relacionadas con los desplazamientos están las deformaciones.

La descripción del desplazamiento de un punto se logra por medio de un vector de seis componentes. Si se considera un cuerpo tridimensional, un punto en este cuerpo se puede desplazar linealmente y también puede girar. El desplazamiento lineal tiene sus tres componentes Δ_x , Δ_y y Δ_z que son llamadas a veces deflexiones y tiene sus otras tres componentes θ_x , θ_y y θ_z que son llamadas normalmente giros. Así que el desplazamiento se puede escribir, para el caso tridimensional, de la siguiente forma

$$D = \begin{pmatrix} \Delta_x \\ \Delta_y \\ \Delta_z \\ \theta_x \\ \theta_y \\ \theta_z \end{pmatrix}$$

Para el caso bidimensional el desplazamiento se describe por un vector de tres componentes. El desplazamiento lineal tiene dos componentes Δ_x y Δ_y y el giro sólo tiene una componente, θ_z , como se muestra en la figura 1.

$$D = \begin{pmatrix} \Delta_x \\ \Delta_y \\ \theta_z \end{pmatrix}$$

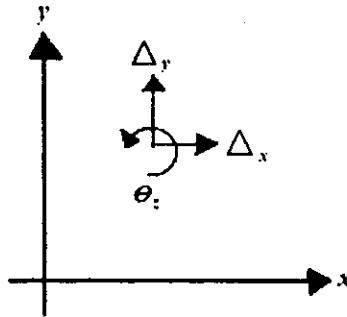


Figura 1. Desplazamientos y giro en dos dimensiones.

Para marcos rígidos planos, el desplazamiento de un punto quedan completamente descrito por el desplazamiento en x , Δ_x , el desplazamiento en y , Δ_y , y el giro θ_z .

La siguiente figura ilustra el desplazamiento del punto extremo de una viga en voladizo.

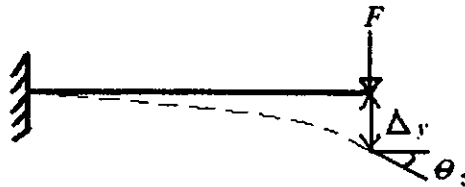


Figura 2. Viga en voladizo.

Como ya se dijo anteriormente, los desplazamientos se deben a fuerzas que actúan sobre la estructura. Si la relación entre las fuerzas externas y los desplazamientos es lineal, entonces se dice que el cuerpo es elástico o que la estructura se comporta elásticamente. Las fuerzas producen esfuerzos internos y si el cuerpo es elástico se pueden relacionar directamente los esfuerzos y las deformaciones.

La teoría de elasticidad es la teoría que relaciona linealmente fuerzas y desplazamientos o también esfuerzos y deformaciones.

La descripción de una fuerza puntual o una carga puntual, sobre un cuerpo en tres dimensiones, es por medio de un vector de seis componentes.

$$Q = \begin{pmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{pmatrix}$$

Este vector consta de dos partes, las fuerzas lineales, que son las fuerzas en x , y y z , y los momentos en x , y y z .

Para el caso de dos dimensiones el vector de carga queda descrito por tres componentes que son la fuerza en x , F_x , la fuerza en y , F_y , y el momento en z , M_z , las cuales se muestran en la figura 3.

$$Q = \begin{pmatrix} F_x \\ F_y \\ M_z \end{pmatrix}$$

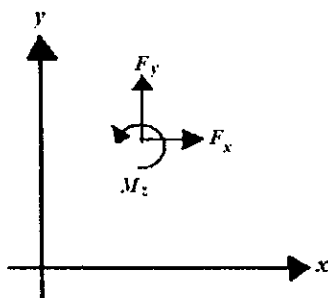


Figura 3. Fuerzas y momento en dos dimensiones.

En tres dimensiones se puede dibujar de la siguiente forma

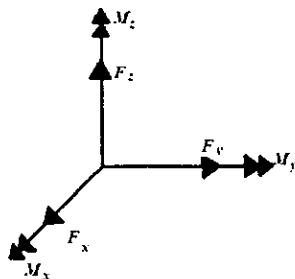


Figura 4. Fuerzas y momentos en tres dimensiones.

3.2 Métodos de energía

Los métodos de energía forman el marco conceptual teórico sobre el cual se basa el análisis estructural. Los conceptos de energía juegan un papel fundamental para un mejor entendimiento de todos los métodos usados en el cálculo estructural. Los métodos de energía ayudan también a comprender el comportamiento de las estructuras. Es más, para métodos más avanzados como el método de elementos finitos los métodos de energía se vuelven necesarios.

Entendiendo bien los métodos de energía se puede dar una formulación variacional la cual consiste en optimizar la energía potencial de los cuerpos elásticos. Las formulaciones variacionales permiten un ataque más general para el planteamiento de las ecuaciones. Sin embargo, en este trabajo de tesis no se necesita todavía hacer uso del cálculo variacional para plantear las ecuaciones básicas. Pero, sí queda formulado de una forma en la que se va a poder aplicar métodos más avanzados.

En este trabajo se incluyen y explican los conceptos de energía tales como trabajo real, trabajo virtual, energía de deformación interna, energía potencial, trabajo interno, etc. Las ecuaciones se plantean usando los conceptos anteriores y relaciones que intuitivamente son fáciles de aceptar, como por ejemplo, que el trabajo realizado por las fuerzas externas queda almacenado en forma de energía de deformación. El desarrollo de los conceptos anteriores permite tener un mejor entendimiento del método de rigidez.

El trabajo virtual se incluye para tener una exposición más completa, ya que para este trabajo de tesis no es absolutamente necesario. Al hacer una exposición de trabajo real y trabajo virtual juntos se tiene un refuerzo conceptual importante. También es bueno considerar que el trabajo virtual resulta muy útil cuando los cuerpos no están actuando elásticamente.

3.3 Trabajo real

La forma más directa de introducirse a los métodos de energía es por medio del trabajo real. El término de trabajo real implica que los desplazamientos y las fuerzas son reales, en contraposición al trabajo virtual en donde las fuerzas y desplazamientos son imaginarios.

Si un cuerpo que no es rígido se somete a una carga externa Q este cuerpo se deforma. El desplazamiento asociado con la carga Q se denota por la letra D , tal y como se muestra en la figura 5.

El desplazamiento D se mide en la dirección de la carga aplicada Q . La relación entre Q y D , para un material elástico, es lineal y se muestra en la figura 6

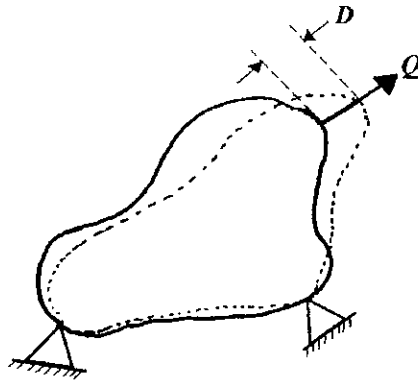


Figura 5. Cuerpo no rígido.

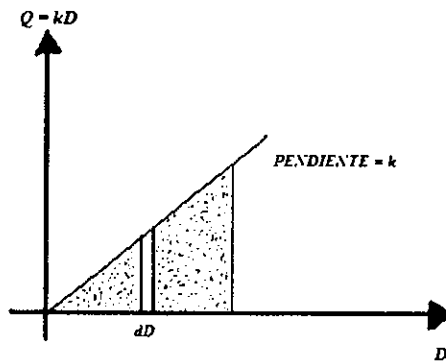


Figura 6. Relación carga-desplazamiento.

De la definición básica de trabajo, la cual podemos enunciar como trabajo es igual a fuerza por desplazamiento, se puede obtener una expresión simple para el trabajo externo de la fuerza Q sobre el cuerpo. Este trabajo externo se escribe por medio del símbolo W_E .

Si se considera un desplazamiento infinitesimal dD como se muestra en la figura 6 se tiene que el diferencial de trabajo externo realizado es

$$dW_E = QdD \quad (3-1)$$

Sin embargo, para un material elástico se tiene que

$$Q = kD \quad (3-2)$$

sustituyendo se obtiene

$$dW_E = kDdD \quad (3-3)$$

y luego se integra para obtener el trabajo total externo W_E

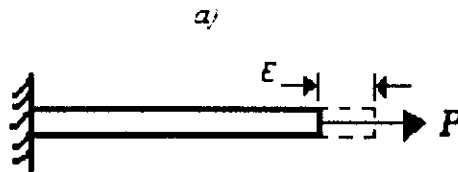
$$W_E = \frac{1}{2}kD^2 \quad (3-4)$$

y en su forma final queda la siguiente fórmula

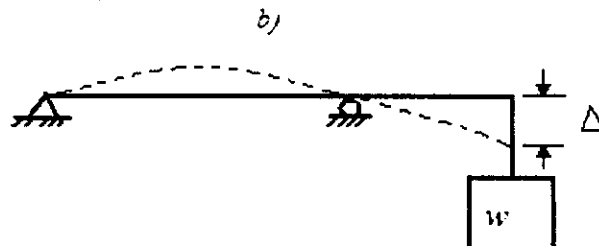
$$W_E = \frac{QD}{2} \quad (3-5)$$

Así se tiene que, para un cuerpo elástico el trabajo externo se puede escribir como el valor medio del producto de la fuerza externa Q y el desplazamiento externo resultante D .

A pesar de que la carga Q se ha mostrado como una fuerza lineal, la expresión $W_E = \frac{1}{2}QD$ es también válida para momentos externos. En el caso de un momento externo, Q tiene las dimensionales de momento que son fuerza por distancia y D se mide en radianes. La figura 7 muestra algunos casos de trabajo externo que se encuentran en el análisis estructural.



$$W_E = \frac{P\varepsilon}{2}$$



$$W_E = \frac{w\Delta}{2}$$

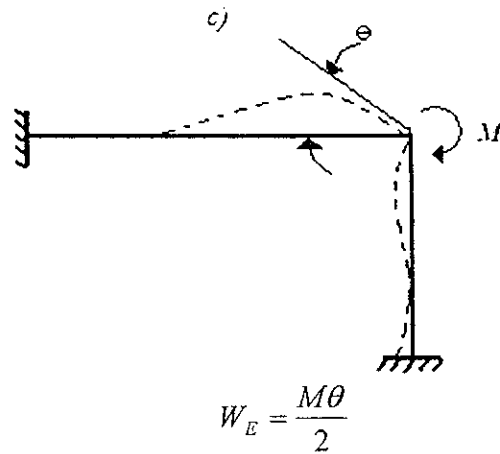


Figura 7. Ejemplos de trabajo externo.

En términos generales, para una estructura sujeta a n cargas externas el trabajo total externo se puede expresar de la siguiente forma

$$W_E = \frac{1}{2} \sum_{i=1}^n Q_i D_i \quad (3-6)$$

donde D_i es la deflexión medida en la dirección de la carga Q_i . La expresión anterior se puede escribir en forma matricial. Considerando las matrices Q y D de la siguiente forma

$$Q = \begin{pmatrix} Q_1 \\ Q_2 \\ \cdot \\ \cdot \\ Q_n \end{pmatrix} \quad D = \begin{pmatrix} D_1 \\ D_2 \\ \cdot \\ \cdot \\ D_n \end{pmatrix}$$

$$W_E = \frac{1}{2} Q^T D \quad (3-7)$$

$$W_E = \frac{1}{2} D^T Q \quad (3-8)$$

3.4 Trabajo real interno

Cuando un cuerpo elástico se somete a cargas externas, este cuerpo se deforma y se realiza trabajo externo. El trabajo externo realizado queda almacenado como energía interna dentro del cuerpo elástico. A esta energía interna almacenada se le llama de varias formas, siendo las más comunes energía potencial elástica y energía de deformación, por ahora se le llamará trabajo interno y se le denota por W_I .

Debido a la conservación de la energía, el trabajo externo se transforma en energía de deformación. Dicho de otra forma el trabajo externo se transforma en trabajo interno. Si los soportes de la estructura no se desplazan se puede establecer que el trabajo externo es igual al trabajo interno

$$W_E = W_I \quad (3-9)$$

En caso de que los soportes sí se desplacen entonces se considera el trabajo realizado por los soportes el cual se denota por W_R y la conservación de la energía queda expresada de la siguiente forma

$$W_E + W_R = W_I \quad (3-10)$$

A menos que se diga lo contrario en este trabajo de tesis se asume que los soportes no se mueven.

La ecuación 3-9 es la base para evaluar deflexiones por el método de trabajo real. Para esto es necesario tener expresiones para el trabajo interno o la energía potencial elástica. Se considerará un cuerpo elástico e isotrópico con módulo de elasticidad E y módulo de Poisson ν . Este caso es lo suficientemente general para los intereses en este trabajo de tesis y permitirá deducir expresiones y formas de trabajo interno fácilmente.

Considérese un elemento tridimensional cuyo volumen es $dV = dx dy dz$ tal y como se muestra en la figura 8

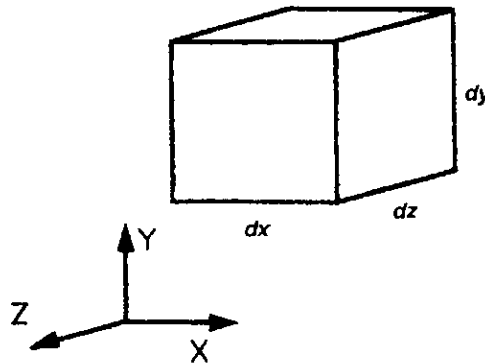


Figura 8. Unidad de volumen

Las direcciones de los esfuerzos se muestran en la figura 9. Asociado a cada esfuerzo axial σ se tiene una deformación ε y asociado a cada esfuerzo de corte τ se tiene una deformación γ .

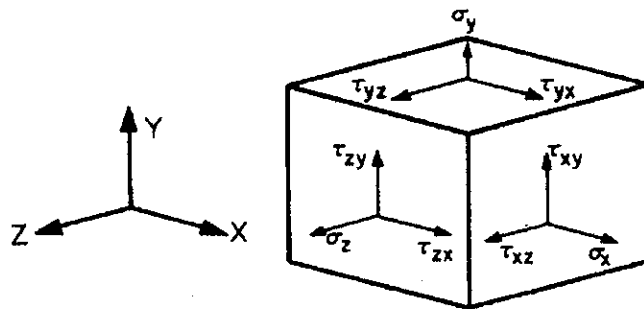


Figura 9. Elemento sujeto al caso general de esfuerzos

Para calcular el trabajo interno se recurre otra vez a la definición trabajo es igual a fuerza por desplazamiento. Recordando que esfuerzo por área es fuerza y deformación por longitud es desplazamiento se puede escribir

$$dW_I = \frac{1}{2}(\sigma_x dydz)(\varepsilon_x dx)$$

$$dW_I = \frac{1}{2} \sigma_x \varepsilon_x dx dy dz$$

$$dW_I = \frac{1}{2} \sigma_x \varepsilon_x dV$$

Si se consideran todas las componentes del tensor de esfuerzo y del tensor de deformación se obtiene la siguiente expresión

$$dW_I = \frac{1}{2}(\sigma_x \varepsilon_x + \sigma_y \varepsilon_y + \sigma_z \varepsilon_z + \sigma_{xy} \varepsilon_{xy} + \sigma_{yz} \varepsilon_{yz} + \sigma_{zx} \varepsilon_{zx}) dV \quad (3-11)$$

De la teoría de la elasticidad, el esfuerzo y la deformación se relacionan linealmente de la siguiente forma

$$\varepsilon_x = \frac{1}{E}(\sigma_x - \nu(\sigma_y + \sigma_z)) \quad (3-12)$$

$$\varepsilon_y = \frac{1}{E}(\sigma_y - \nu(\sigma_x + \sigma_z)) \quad (3-13)$$

$$\varepsilon_z = \frac{1}{E}(\sigma_z - \nu(\sigma_x + \sigma_y)) \quad (3-14)$$

$$\gamma_{xy} = \frac{2(1+\nu)}{E} \tau_{xy} \quad (3-15)$$

$$\gamma_{yz} = \frac{2(1+\nu)}{E} \tau_{yz} \quad (3-16)$$

$$\gamma_{zx} = \frac{2(1+\nu)}{E} \tau_{zx} \quad (3-17)$$

donde E es el módulo de elasticidad y ν es el coeficiente de Poisson. Si se escriben en forma matricial las ecuaciones anteriores quedan de la siguiente forma

$$\begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{pmatrix} = \frac{1}{E} \begin{pmatrix} 1 & -\nu & -\nu & 0 & 0 & 0 \\ -\nu & 1 & -\nu & 0 & 0 & 0 \\ -\nu & -\nu & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2(1+\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & 2(1+\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(1+\nu) \end{pmatrix} \begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{pmatrix}$$

$$\varepsilon = C \sigma \quad (3-18)$$

donde

ε = matriz columna de las deformaciones

C = matriz simétrica de las propiedades del material

σ = matriz columna de los esfuerzos

La expresión 3-18 se puede escribir en forma inversa de la manera siguiente

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{pmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2}-\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}-\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}-\nu \end{pmatrix} \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{pmatrix}$$

$$\sigma = B \varepsilon \quad (3-19)$$

donde B = matriz simétrica de las propiedades elásticas del material.

Nótese que B es la matriz inversa de C y viceversa o sea que

$$C = B^{-1}$$

$$B = C^{-1}$$

Usando la notación matricial se puede expresar el trabajo interno por unidad de volumen de una forma sucinta como se muestra a continuación.

$$\frac{dW_I}{dV} = \frac{1}{2} \sigma^T \varepsilon \quad (3-20)$$

ó

$$\frac{dW_I}{dV} = \frac{1}{2} \varepsilon^T \sigma \quad (3-21)$$

donde $\sigma^T = (\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx})$ es la transpuesta de σ y $\varepsilon^T = (\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx})$ es la transpuesta de ε .

Sustituyendo $\varepsilon = C\sigma$ en 3-20 se obtiene

$$\frac{dW_I}{dV} = \frac{1}{2} \sigma^T C \sigma \quad (3-22)$$

y sustituyendo $\sigma = B\varepsilon$ en 3-21 se obtiene

$$\frac{dW_I}{dV} = \frac{1}{2} \varepsilon^T B \varepsilon \quad (3-23)$$

La ecuación 3-23 expresa el trabajo interno por unidad de volumen en términos de los esfuerzos y de las propiedades elásticas del material. La ecuación 3-22 expresa el trabajo interno en términos de las deformaciones y de las propiedades elásticas del material. Estas ecuaciones están en términos generales y de ellas se deducirán fórmulas para situaciones más específicas como son los miembros prismáticos, por ejemplo vigas, columnas, etc. Los miembros prismáticos se consideran como elementos unidimensionales y por lo tanto tienen simplificaciones importantes.

3.5 Trabajo virtual

Trabajo virtual es el trabajo realizado por una fuerza real a lo largo de un desplazamiento imaginario o también se puede conceptualizar como el trabajo realizado por una fuerza imaginaria a lo largo de un desplazamiento real. El término virtual se usa en lugar del término imaginario queriendo decir que no es real. Para denotar cantidades virtuales como trabajo virtual, fuerza virtual o desplazamiento virtual se antepone el símbolo δ . De esta manera se tiene que δD es desplazamiento virtual, δQ es carga virtual y δW es trabajo virtual.

La utilidad del concepto de trabajo virtual la encontramos por primera vez en el curso de estática donde es fácil observar que para un cuerpo rígido, en equilibrio, sometido a un conjunto de fuerzas reales, el trabajo virtual resultante durante cualquier desplazamiento virtual es cero.

El concepto de trabajo virtual se puede extender para cuerpos elásticos o estructuras elásticas. Lo anterior se logra considerando que para un desplazamiento virtual las cargas externas realizan un trabajo virtual externo y este trabajo se traduce en un trabajo virtual interno debido a los esfuerzos dentro del cuerpo elástico. Formalmente se puede expresar el principio de trabajo virtual de la siguiente manera: Si un cuerpo o estructura está en equilibrio bajo un conjunto de fuerzas y si a este cuerpo se le da un desplazamiento virtual consistente con las restricciones de la estructura, entonces el trabajo virtual externo es igual al trabajo virtual interno. En forma de ecuación se escribe de la siguiente forma

$$\delta W_E = \delta W_I \quad (3-24)$$

De una forma similar al trabajo real se puede establecer el trabajo virtual externo e interno.

Si una fuerza real Q es sometida a un desplazamiento virtual δD , el trabajo virtual externo resultante es

$$\delta W_E = Q\delta D \quad (3-25)$$

El valor de δD es arbitrario, a menudo se escoge el valor de 1. Es importante hacer notar que para los desplazamientos virtuales la fuerza se considera constante. Además hay que hacer notar que la única diferencia con el trabajo real es el factor $\frac{1}{2}$.

Otro concepto de trabajo virtual resulta de considerar una carga virtual en lugar de un desplazamiento virtual. De esta forma, si una estructura está sujeta a una carga virtual δQ y a un desplazamiento real D , el trabajo virtual externo sobre esta estructura es

$$\delta W_E = \delta Q D \quad (3-26)$$

El valor de la fuerza virtual se mantiene constante y es usual tener el valor de 1 para δQ .

Una de las ventajas del concepto de trabajo virtual es que puede ser aplicado a cuerpos cuyo comportamiento no es linealmente elástico, mientras que el trabajo real como se ha expuesto anteriormente considera exclusivamente a cuerpos elásticos, que tienen una relación lineal entre esfuerzo y deformaciones.

En una estructura que se consideran varias cargas o varios desplazamientos las ecuaciones (3-25) y (3-26) se pueden generalizar a las siguientes ecuaciones.

$$\delta W_E = \sum_{i=1}^n Q_i \delta D_i \quad (3-27)$$

$$\delta W_E = \sum_{i=1}^n \delta Q_i D_i \quad (3-28)$$

Las expresiones anteriores se pueden escribir en forma matricial considerando los siguientes vectores columna.

$$Q = \begin{pmatrix} Q_1 \\ Q_2 \\ \cdot \\ \cdot \\ \cdot \\ Q_n \end{pmatrix} \quad \delta Q = \begin{pmatrix} \delta Q_1 \\ \delta Q_2 \\ \cdot \\ \cdot \\ \cdot \\ \delta Q_n \end{pmatrix}$$

$$D = \begin{pmatrix} D_1 \\ D_2 \\ \cdot \\ \cdot \\ \cdot \\ D_n \end{pmatrix} \quad \delta D = \begin{pmatrix} \delta D_1 \\ \delta D_2 \\ \cdot \\ \cdot \\ \cdot \\ \delta D_n \end{pmatrix}$$

Quedando el trabajo virtual externo expresado de la siguiente forma

$$\delta W_E = D^T \delta Q \quad (3-29)$$

ó

$$\delta W_E = \delta Q^T D \quad (3-30)$$

3.6 Trabajo virtual interno

Para poder calcular deflexiones o desplazamientos por el método de trabajo virtual, es necesario establecer algunas expresiones. Al igual que el trabajo real, primero se discute el trabajo interno virtual en términos generales y luego, en una sección posterior, se deducen para casos particulares, como lo son los miembros prismáticos.

Otra vez se considera un elemento diferencial cualquiera sometido ahora a esfuerzos virtuales, como se muestra en la figura 10

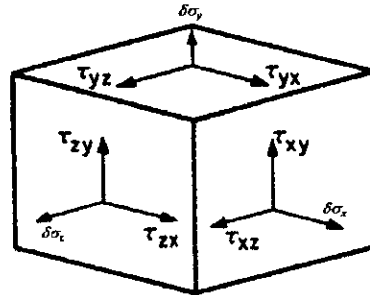


Figura 10. Elemento diferencial sujeto a esfuerzos.

Los esfuerzos virtuales son provocados por fuerzas virtuales. Si este elemento tiene desplazamientos reales que inducen deformaciones reales entonces, el trabajo virtual interno realizado por $\delta\sigma_x$ a lo largo de la deformación ϵ_x es

$$\delta W_I = (\delta\sigma_x dydz)(\epsilon_x dx) \quad (3-31)$$

$$\delta W_I = \delta\sigma_x \epsilon_x dV \quad (3-32)$$

Si se consideran todas las deformaciones posibles, el trabajo virtual interno total es

$$\delta W_I = (\delta\sigma_x \epsilon_x + \delta\sigma_y \epsilon_y + \delta\sigma_z \epsilon_z + \delta\tau_{xy} \gamma_{xy} + \delta\tau_{yz} \gamma_{yz} + \delta\tau_{zx} \gamma_{zx}) dV \quad (3-33)$$

La expresión anterior se puede escribir en forma matricial si se consideran

$$\delta\sigma = \begin{pmatrix} \delta\sigma_x \\ \delta\sigma_y \\ \delta\sigma_z \\ \delta\tau_{xy} \\ \delta\tau_{yz} \\ \delta\tau_{zx} \end{pmatrix} \quad \epsilon = \begin{pmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{pmatrix}$$

y, por lo tanto, el trabajo virtual interno se escribe de la siguiente forma

$$\delta W_I = \delta\sigma^T \epsilon dV \quad (3-34)$$

Además se tiene que existe una relación entre esfuerzos y deformaciones que es

$$\epsilon = C\sigma$$

por lo que el trabajo virtual interno se puede expresar en términos generales por la siguiente fórmula

$$\delta W_i = \delta \sigma^T C \sigma dV \quad (3-35)$$

Donde C es la matriz de propiedades de material descrita anteriormente.

3.7 Formas de trabajo interno

El trabajo interno también conocido como energía de deformación o energía potencial, ya ha sido discutido en forma general para cualquier cuerpo elástico. Sin embargo se estará especialmente interesado en cuerpos relativamente simples como lo son las barras o cualquier miembro prismático. Se entiende por miembro prismático o barra, a cualquier cuerpo que tenga una de sus dimensiones sustancialmente mayor que las otras dos. Así, por ejemplo, una barra tiene una longitud L mucho mayor que su área transversal A .

En el caso de miembros prismáticos se harán algunas simplificaciones, con las cuales se obtendrán las fórmulas necesarias para este trabajo de tesis. Las fuerzas que se consideran sobre la barra serán la fuerza axial P_x , la fuerza de corte V_x y el momento flexionante M_x , tal y como se muestra en la figura 11.

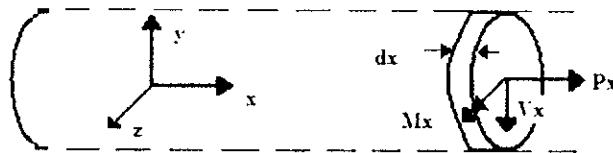


Figura 11. Fuerzas sobre una barra.

Las otras tres fuerzas, entre las cuales está el momento de torsión, pueden ser ignoradas para barras largas. Además la fuerza cortante tiene una contribución muy pequeña a la energía de deformación. Por lo anterior se tienen dos contribuciones al trabajo interno, una de ellas es la contribución de la fuerza axial P_x y la otra es la debida al momento flexionante M_x .

Tanto la fuerza axial P_x como el momento flexionante M_x producen esfuerzos axiales σ_x . Estos esfuerzos son fáciles de calcular usando los conocimientos adquiridos en los cursos de Resistencia de Materiales y quedan de la siguiente forma:

$$\sigma_x = \frac{P_x}{A} \quad (3-36)$$

$$\sigma_x = \frac{M_x y}{I} \quad (3-37)$$

donde A es el área transversal e I es el momento de inercia.

Recordando que el trabajo interno por unidad de volumen está dado por

$$\frac{dW_l}{dV} = \frac{1}{2} \sigma^T C \sigma$$

se puede escribir el trabajo interno de la siguiente forma

$$\begin{aligned} \frac{dW_l}{dV} &= \frac{1}{2} \sigma_x \frac{1}{E} \sigma_x \\ \frac{dW_l}{dV} &= \frac{1}{2} \left(\frac{P_x}{A} - \frac{M_x y}{I} \right) \frac{1}{E} \left(\frac{P_x}{A} - \frac{M_x y}{I} \right) \end{aligned}$$

finalmente se obtiene la siguiente fórmula para $\frac{dW_l}{dV}$

$$\frac{dW_l}{dV} = \frac{P_x^2}{2EA^2} - \frac{P_x M_x y}{AI} + \frac{M_x^2 y^2}{2EI^2} \quad (3-38)$$

Si se considera $dV = dx dy dz$ y $dA = dy dz$ la fórmula anterior se puede integrar sobre el área transversal obteniendo

$$\frac{dW_l}{dx} = \int_A \left(\frac{P_x^2}{2EA^2} - \frac{P_x M_x y}{AI} + \frac{M_x^2 y^2}{2EI^2} \right) dA$$

operando cada término queda lo siguiente

$$\int_A \frac{P_x^2}{2EA^2} dA = \frac{P_x^2}{2EA}$$

$$\int_A \frac{P_x M_x y}{AI} dA = \frac{P_x M_x}{AI} \int_A y dA = 0$$

$$\int_A \frac{M_x^2 y^2}{2EI^2} dA = \frac{M_x^2}{2EI^2} \int_A y^2 dA = \frac{M_x^2}{2EI}$$

y se puede escribir la fórmula final para $\frac{dW_I}{dx}$

$$\frac{dW_I}{dx} = \frac{P_x^2}{2EA} + \frac{M_x^2}{2EI} \quad (3-39)$$

A $\frac{dW_I}{dx}$ se le entiende como la cantidad de energía de deformación por unidad de longitud. Es decir que $\frac{dW_I}{dx}$ es una densidad lineal de energía. Vale la pena hacer notar que esta densidad lineal de energía queda en términos de la fuerza axial P_x y el momento flexionante M_x y como se puntualizó anteriormente, la energía debido al corte V_x es despreciable para el caso de miembros prismáticos con un área transversal relativamente pequeña.

Para una exposición más completa y como referencia se agrega la fórmula que incluye el corte

$$\frac{dW_I}{dx} = \frac{P_x^2}{2EA} + \frac{M_x^2}{2EI} + \frac{kV_x^2}{2GA} \quad (3-40)$$

En la fórmula anterior k es una constante que depende de la forma de la sección transversal y G es el módulo de corte, que en la teoría de la elasticidad queda en función del módulo de Young E y del módulo de Poisson ν , como lo describe la siguiente ecuación.

$$G = \frac{E}{2(1+\nu)} \quad (3-41)$$

Si se requiere la energía total en un miembro prismático, se tiene que integrar sobre toda la longitud obteniendo la fórmula final para el trabajo interno W_I .

$$W_I = \int_0^L \left(\frac{P_x^2}{2EA} + \frac{M_x^2}{2EI} \right) dx \quad (3-42)$$

Ahora es deseable expresar el trabajo interno para el caso de cargas virtuales. Una de las ventajas del trabajo virtual es que puede ser usado en cuerpos no necesariamente elásticos, mientras que las fórmulas que se han deducido para trabajo real, solamente se aplican a cuerpos elásticos.

La diferencia es que se consideran cargas virtuales o imaginarias que se denotan p_x para la fuerza axial virtual, m_x para el momento flexionante virtual y v_x para la fuerza de corte virtual.

El trabajo virtual interno ya se ha deducido en términos generales. Ahora se escribirá para el caso de un miembro prismático. Recordemos que el trabajo virtual interno por unidad de volumen es

$$\delta W_I = \delta \sigma^T C \alpha dV \quad (3-43)$$

para el caso de una barra

$$\delta W_I = \frac{1}{E} \delta \sigma_x \sigma_x dV \quad (3-44)$$

$$\delta \sigma_x = \frac{p_x}{A} - \frac{m_x y}{I} \quad (3-45)$$

Sustituyendo se obtiene la siguiente expresión

$$\delta W_I = \frac{1}{E} \left(\frac{p_x}{A^2} - \frac{m_x y}{I} \right) \left(\frac{P_x}{A} - \frac{M_x y}{I} \right) dV$$

$$\delta W_I = \left(\frac{p_x P_x}{EA^2} + \frac{m_x M_x y^2}{EI^2} - \frac{m_x P_x + M_x p_x}{EIA} y \right) dV$$

Si se escribe $dV = dx dy dz$ y $dA = dy dz$ y se integra sobre el área tenemos

$$\int_A \frac{p_x P_x}{EA^2} dA = \frac{p_x P_x}{EA}$$

$$\int_A \frac{m_x M_x y^2}{EI^2} dA = \frac{m_x M_x}{EI}$$

$$\int_A \frac{m_x P_x - M_x p_x}{EIA} y dA = 0$$

Finalmente se obtiene la fórmula de trabajo interno virtual por unidad de longitud

$$\frac{\delta W_I}{dx} = \frac{P_x P_x}{EA} + \frac{m_x M_x}{EI} \quad (3-46)$$

Para el trabajo total interno de un miembro prismático se tiene que integrar sobre toda la longitud L del miembro, con lo que se obtiene la siguiente fórmula

$$\delta W_I = \int_0^L \left(\frac{P_x P_x}{EA} + \frac{m_x M_x}{EI} \right) dx \quad (3-47)$$

3.8 Teoremas de Betti y Maxwell

El teorema de Betti y el teorema de Maxwell son muy importantes en el análisis estructural. Estos teoremas son la base para desarrollar el concepto de los coeficientes de influencia que a su vez forman el fundamento para desarrollar y comprender el método de flexibilidad y de rigidez. Además se tiene que el teorema de Betti se usa en la demostración del teorema de Castigliano.

Para enunciar claramente el teorema de Betti considérese un cuerpo elástico sometido a dos sistemas de fuerzas o cargas. El primer sistema de cargas es el sistema P que consta de cargas P_i y el segundo sistema o conjunto de cargas es el sistema Q que consta de las cargas Q_j . En la siguiente figura se ilustra los dos sistemas de cargas.

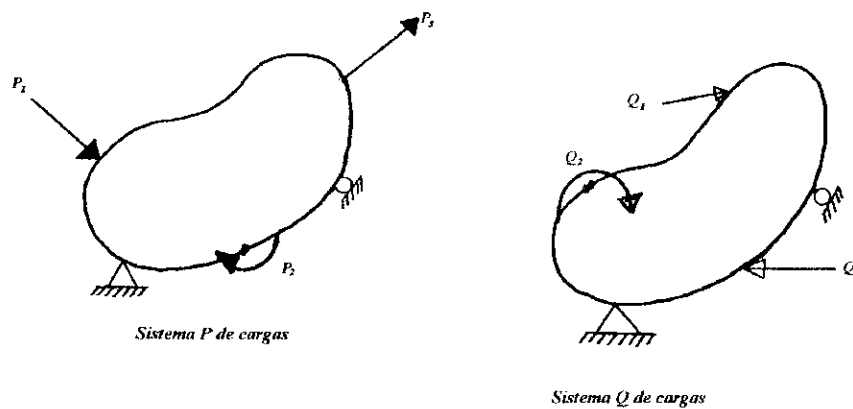


Figura 12. Dos sistemas de cargas.

El conjunto P de cargas deforman el cuerpo elástico produciendo desplazamientos Δ . De la misma forma el conjunto de cargas Q deforma el cuerpo ocasionando desplazamientos D . Si se consideran que hay m cargas en el sistema P y n cargas en el sistema Q , el teorema de Betti se puede enunciar y demostrar de la siguiente forma

Teorema de Betti

Para una estructura elástica, el trabajo realizado por un conjunto de cargas P a lo largo de los desplazamientos D debidos al conjunto de cargas Q , es igual al trabajo realizado por el conjunto de cargas Q a lo largo de los desplazamientos Δ ocasionados por el sistema de cargas P

$$\sum_{k=1}^m P_k D_k = \sum_{k=1}^n Q_k \Delta_k \quad (3-48)$$

Demostración

Si primero se aplica el sistema P al cuerpo elástico, entonces el trabajo externo realizado es

$$W_E = \frac{1}{2} \sum_{k=1}^m P_k \Delta_k \quad (3-49)$$

Si a continuación se aplica el sistema de cargas Q manteniendo constante Q el sistema P , se realiza un trabajo externo extra dado por

$$W_E = \frac{1}{2} \sum_{k=1}^n Q_k D_k + \sum_{k=1}^m P_k D_k \quad (3-50)$$

por lo que el trabajo externo total ocasionado por los dos sistemas de cargas es

$$W_E = \frac{1}{2} \sum_{k=1}^m P_k \Delta_k + \frac{1}{2} \sum_{k=1}^n Q_k D_k + \sum_{k=1}^m P_k D_k \quad (3-51)$$

si ahora se aplica primero el sistema Q de cargas el trabajo realizado es

$$W_E = \frac{1}{2} \sum_{k=1}^n Q_k D_k \quad (3-52)$$

y si a continuación actúa el sistema P realiza un trabajo extra dado por

$$W_E = \frac{1}{2} \sum_{k=1}^m P_k \Delta_k + \sum_{k=1}^n Q_k \Delta_k \quad (3-53)$$

por lo que el trabajo total es

$$W_E = \frac{1}{2} \sum_{k=1}^n Q_k D_k + \frac{1}{2} \sum_{k=1}^m P_k \Delta_k + \sum_{k=1}^n Q_k \Delta_k \quad (3-54)$$

si se igualan los dos trabajos

$$W_E = W_E$$

$$\frac{1}{2} \sum_{k=1}^m P_k \Delta_k + \frac{1}{2} \sum_{k=1}^n Q_k D_k + \sum_{k=1}^m P_k D_k = \frac{1}{2} \sum_{k=1}^n Q_k D_k + \frac{1}{2} \sum_{k=1}^m P_k \Delta_k + \sum_{k=1}^n Q_k \Delta_k$$

y se cancelan los términos iguales, quedando el teorema de Betti

$$\sum_{k=1}^m P_k D_k = \sum_{k=1}^n Q_k \Delta_k$$

Como una consecuencia del teorema de Betti se tiene el teorema de Maxwell, también conocido como el teorema de trabajos recíprocos de Maxwell. El teorema de Maxwell se puede considerar como un caso particular del teorema de Betti o como un corolario.

Para enunciar el teorema de Maxwell se considera una carga P que produce una deflexión o desplazamiento en el punto de aplicación a Δ_a y una deflexión Δ_b en otro punto b . Además se considera otra carga Q aplicada en el punto b y produce una deflexión D_b en el punto b y también provoca un desplazamiento en el punto a , como se muestra en la figura 13

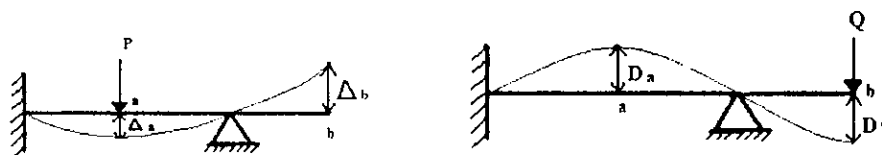


Figura 13. Teorema de Maxwell.

Con lo anterior se puede enunciar claramente el teorema de Maxwell y demostrarlo

Teorema de Maxwell

Sea P una fuerza aplicada en el punto a que produce un desplazamiento Δ_b en el punto b y sea Q una carga aplicada en el punto b que produce un desplazamiento D_a en el punto a .

Si $P=Q$ entonces $\Delta_b = D_a$.

Demostración

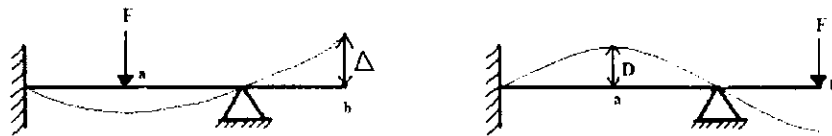
Por el teorema de Betti

$$PD_a = Q\Delta_b$$

por hipótesis $P=Q$ por lo tanto

$$D_a = \Delta_b$$

La figura anterior se dibuja ahora para el caso $F=P=Q$ como se muestra en la figura 14.



$$\Delta = D$$

Figura 14. Igualdad de desplazamientos

3.9 Teorema de Castigliano

Con los conceptos de trabajo expuestos anteriormente se tiene que la energía de deformación se puede expresar como una función de las cargas. El segundo teorema de Castigliano dice que los desplazamientos se obtienen sacando las derivadas parciales de la energía de deformación con respecto a las cargas. De la misma forma se puede decir que la energía de deformación es una función de los desplazamientos, y el primer teorema de Castigliano establece que las cargas resultan de calcular las derivadas parciales de la energía de deformación con respecto a los desplazamientos.

Se puede decir que el segundo teorema de Castigliano es el dual del primer teorema de Castigliano y viceversa. En esta sección se enuncian y demuestran formalmente los dos teoremas. Los teoremas de Castigliano juegan un papel fundamental en el análisis estructural, ya que su uso, aunque sea en forma implícita, está en muchas partes. El término de energía de deformación es sinónimo al de trabajo interno. También se usa energía potencial como sinónimo de trabajo interno, y si el material es elástico se le llama energía potencial elástica. Se demostrará antes el segundo teorema de Castigliano y luego el primer teorema de Castigliano.

Segundo Teorema de Castigliano

Si una estructura está sujeta a un conjunto de cargas, entonces el desplazamiento de cualquier carga es igual a la derivada parcial de la energía de deformación con respecto a la carga

$$D_i = \frac{\partial W_I}{\partial Q_i} \quad (3-55)$$

Demostración

Considérese un cuerpo cualquiera sujeto a un sistema de cargas, como se muestra en la figura 15

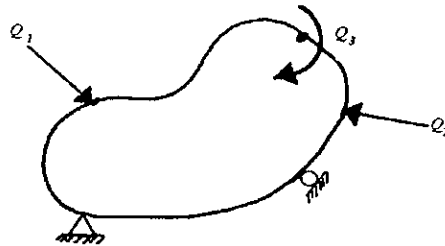


Figura 15. Cuerpo sujeto a sistema de cargas.

Asúmase que los soportes no se mueven entonces se tiene que

$$W_I = W_E$$

Donde W_I es el trabajo interno, dicho de otra forma W_I es la energía de deformación y W_E es el trabajo externo que depende de las cargas Q y de los desplazamientos D .

$$W_E = \frac{1}{2} \sum_{i=1}^n Q_i D_i$$

Se tiene que los desplazamientos dependen de las cargas y viceversa, las cargas dependen de los desplazamientos, por lo que la energía de deformación W_I se puede expresar como una función de las cargas

$$W_I = f(Q_1, Q_2, \dots, Q_n).$$

Si la i -ésima carga se incrementa un diferencial, el cambio de energía de deformación se puede expresar como

$$dW_I = \frac{\partial W_I}{\partial Q_i} dQ_i \quad (3-56)$$

el trabajo total interno quedaría ahora como

$$W_I + dW_I = \frac{1}{2} \sum_{j=1}^n Q_j D_j + \frac{\partial W_I}{\partial Q_i} dQ_i \quad (3-57)$$

ahora, si se aplica dQ_i a la estructura antes del conjunto de las n cargas, entonces la energía de deformación es

$$dW_I = \frac{1}{2} dQ_i dD_i \quad (3-58)$$

y si se permite que a continuación actúen el conjunto de n cargas entonces se obtiene

$$dW_I + W_I = \frac{1}{2} dQ_i dD_i + dQ_i D_i + \frac{1}{2} \sum_{j=1}^n Q_j D_j \quad (3-59)$$

la carga diferencial dQ_i permanecen constante durante la aplicación de las otras cargas.

$$W_I + dW_I = dW_I + W_I$$

$$\frac{1}{2} \sum_{j=1}^n Q_j D_j + \frac{\partial W_I}{\partial Q_i} dQ_i = \frac{1}{2} dQ_i dD_i + dQ_i D_i + \frac{1}{2} \sum_{j=1}^n Q_j D_j$$

se eliminan los términos iguales y se desprecia el término $dQ_i dD_i$ por ser un diferencial de mayor orden.

$$\frac{\partial W_I}{\partial Q_i} dQ_i = D_i dQ_i$$

de lo anterior fácilmente se obtiene el segundo teorema de Castigliano.

$$\frac{\partial W_I}{\partial Q_i} = D_i$$

El primer teorema de Castigliano se puede deducir en forma similar. A continuación se enuncia y se demuestra.

Primer Teorema de Castigliano

Si una estructura se deforma debido a un conjunto de cargas, entonces la carga es igual a la derivada parcial de la energía de deformación con respecto al desplazamiento de la carga.

$$Q_i = \frac{\partial W_i}{\partial D_i} \quad (3-60)$$

Demostración

Se considera un cuerpo cualquiera, sujeto a un sistema de cargas, tal y como se hizo en el primer teorema.

Se asume que los soportes no se mueven y, por lo tanto, no realizan trabajo, por lo que el trabajo externo es igual al trabajo interno.

Como ya se ha visto anteriormente, el trabajo externo depende de los desplazamientos y de las cargas y además las cargas dependen de los desplazamientos, por lo que la energía de deformación W_i se puede expresar como una función de los desplazamientos.

$$W_i = g(D_1, D_2, \dots, D_n)$$

Si el i -ésimo desplazamiento D_i se incrementa en un diferencial, el incremento de energía de deformación queda así

$$dW_i = \frac{\partial W_i}{\partial D_i} dD_i \quad (3-61)$$

el trabajo total interno quedaría ahora como

$$W_i + dW_i = \frac{1}{2} \sum_{j=1}^n Q_j D_j + \frac{\partial W_i}{\partial D_i} dD_i \quad (3-62)$$

si ahora se desplaza dD_i a la estructura antes de que se apliquen las n cargas entonces, la energía de deformación es

$$dW_i = \frac{1}{2} dQ_i dD_i \quad (3-63)$$

y si se permite que a continuación actúan las n cargas entonces se obtiene

$$dW_i + W_i = \frac{1}{2} dQ_i dD_i + Q_i dD_i + \frac{1}{2} \sum_{j=1}^n Q_j D_j \quad (3-64)$$

igualando se tiene que

$$W_i + dW_i = dW_i + W_i$$

$$\frac{1}{2} \sum_{j=1}^n Q_j D_j + \frac{\partial W_I}{\partial D_i} dD = \frac{1}{2} d(Q_i dD_i) + Q_i dD_i + \frac{1}{2} \sum_{j=1}^n Q_j D_j \quad (3-65)$$

se eliminan términos semejantes y se desprecia $d(Q_i dD_i)$ por ser diferencial de orden mayor

$$\frac{\partial W_I}{\partial D_i} dD_i = Q_i dD_i$$

de lo anterior fácilmente se obtiene el segundo teorema de Castigliano.

$$\frac{\partial W_I}{\partial D_i} = Q_i.$$

Debido a que se tienen fórmulas que expresan la energía de deformación para miembros prismáticos, ahora se puede aplicar el segundo teorema de Castigliano de la siguiente forma

Si se tiene que $W_I = \int_0^L \frac{M_x^2}{2EI} dx$ entonces se puede obtener el desplazamiento de la siguiente forma

$$D_i = \frac{1}{EI} \int_0^L M_x \frac{\partial M_x}{\partial Q_i} dx \quad (3-66)$$

y si se tiene que $W_I = \int_0^L \frac{P_x^2}{2EA} dx$ entonces

$$D_i = \int_0^L P_x \frac{\partial P_x}{\partial Q_i} \frac{dx}{EA} \quad (3-67)$$

3.10 Coeficientes de influencia

La relación entre cargas y desplazamientos es el punto clave para el entendimiento del comportamiento de las estructuras. Los coeficientes de influencia son las constantes que relacionan directamente las fuerzas con los desplazamientos. Una fuerza o carga Q aplicada en cualquier punto de un cuerpo, provoca que todos los puntos de este cuerpo se muevan de su posición original. Estos movimientos son llamados desplazamientos, y se escriben usando una D .

Es importante recordar que si se está en tres dimensiones la carga Q tiene seis componentes y el desplazamiento D de un punto tiene también seis componentes. En caso

de estar en dos dimensiones la carga Q y el desplazamiento D tienen tres componentes. Todo lo anterior está establecido en la primera sección de este capítulo.

Se introducirá el concepto de coeficiente de influencia en dos partes. En la primera parte se definen los coeficientes de influencia de flexibilidad y para la segunda parte se definen los coeficientes de influencia de rigidez. Tal y como se verá más adelante los coeficientes de rigidez son inversos de los de flexibilidad.

Para poder dar una definición precisa de coeficiente de influencia es necesario identificar claramente los grados de libertad. Debido a que el desplazamiento, de un punto en un cuerpo, queda descrito completamente por sus seis componentes; se dice que el punto en cuestión tiene seis grados de libertad. Hay un grado de libertad por cada componente del desplazamiento. De esta forma, si se consideran N puntos en un cuerpo tridimensional se tienen $6N$ grados de libertad. En caso de que se consideren N puntos en una estructura plana, o sea que se está en dos dimensiones, se tienen $3N$ grados de libertad. Para el caso bidimensional los tres grados de libertad de un punto son el desplazamiento horizontal, el desplazamiento vertical y el giro.

Para las cargas se tiene algo similar. Así, si se tiene una carga aplicada en un punto en un cuerpo tridimensional, ésta tiene seis componentes. Para el caso bidimensional la carga tiene tres componentes, las cuales son la fuerza horizontal, la fuerza vertical y el momento. Se usará un índice i o j para identificar cualquier grado de libertad. Dos grados de libertad distintos se tienen que identificar con dos índices o números distintos. Como se verá en el siguiente capítulo, es deseable tener una forma sistemática y ordenada de identificar los grados de libertad.

Ahora considérese una carga Q_j aplicada sobre el grado de libertad j , esta carga provoca desplazamientos por todo el cuerpo, en particular, el grado de libertad i se desplaza D_i . Con lo anterior se puede definir en forma clara y concisa los coeficientes de influencia de flexibilidad.

Definición: Coeficiente de influencia de flexibilidad

$$f_{ij} = \frac{D_i}{Q_j} \quad (3-68)$$

por lo que

$$D_i = f_{ij} Q_j \quad (3-69)$$

En caso de tener una estructura elástica f_{ij} es una constante. Así se tiene que si se duplica la carga, el desplazamiento se duplica y por ende la constante de proporcionalidad f_{ij} no cambia.

Para ilustrar el concepto de coeficiente de flexibilidad, considérese una viga en voladizo como se muestra en la figura 16. En esta viga se localizan dos puntos, el punto medio de la viga identificado con la letra a y el extremo derecho de la viga identificado con la letra b . Tanto el punto a como el punto b tienen tres grados de libertad, así que se tienen seis grados de libertad. El primer grado de libertad es el desplazamiento horizontal de a , el segundo grado de libertad es el desplazamiento vertical de a , el tercer grado de libertad es el giro de a ; el cuarto grado de libertad es el desplazamiento horizontal en b , el quinto grado de libertad es el desplazamiento vertical de b y el sexto grado de libertad es el giro en b . Luego se ilustran dos casos. En el primer caso se aplica una carga $Q_2=1$ en el segundo grado de libertad y se observan los desplazamientos en los grados de libertad 5 y 6. En el segundo caso se aplica una carga $Q_3=1$ en el tercer grado de libertad y se observan los desplazamientos en el quinto y sexto grados de libertad.

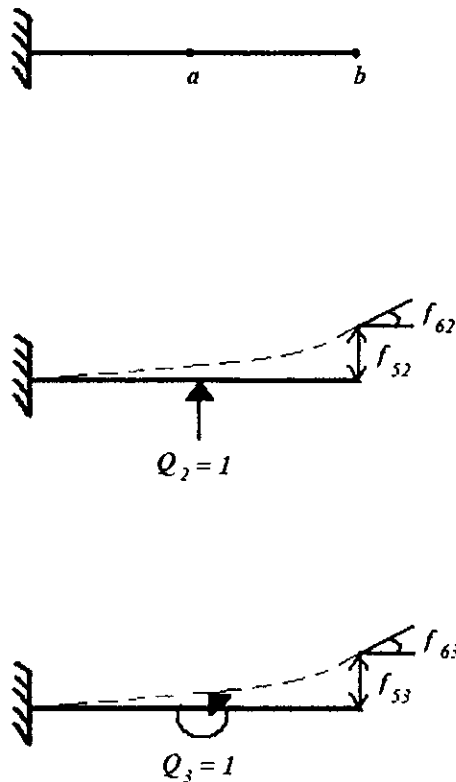


Figura 16. Coeficientes de flexibilidad.

Los coeficientes f_{ij} son desplazamientos causados por cargas unitarias. En caso de que la estructura sea elástica, se puede aplicar el principio de superposición, y se tiene la siguiente relación cuando se aplica simultáneamente las cargas Q_2 y Q_3 .

$$D_5 = f_{52}Q_2 + f_{53}Q_3$$

$$D_6 = f_{62}Q_2 + f_{63}Q_3$$

Para el caso considerado anteriormente e ilustrado en la figura 16. El coeficiente f_{52} es el desplazamiento vertical en b ocasionado por una carga vertical unitaria en a . El coeficiente f_{53} es el desplazamiento vertical en b ocasionado por un momento unitario en a . El coeficiente f_{62} es el giro en b ocasionado por una carga vertical unitaria en a y el coeficiente f_{63} es el giro en b provocado por un momento unitario en a .

La relación anterior se puede generalizar a cualquier número de cargas y desplazamientos, quedando escrita de la siguiente forma

$$\begin{aligned} D_1 &= f_{11}Q_1 + f_{12}Q_2 + \dots + f_{1n}Q_n \\ D_2 &= f_{21}Q_1 + f_{22}Q_2 + \dots + f_{2n}Q_n \\ &\vdots \\ &\vdots \\ &\vdots \\ D_n &= f_{n1}Q_1 + f_{n2}Q_2 + \dots + f_{nn}Q_n \end{aligned} \quad (3-70)$$

La ecuación anterior se puede escribir en forma matricial de la siguiente manera

$$\begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ \vdots \\ \vdots \\ D_n \end{pmatrix} = \begin{pmatrix} f_{11} & f_{12} & \dots & \dots & f_{1n} \\ f_{21} & f_{22} & \dots & \dots & f_{2n} \\ \vdots & \vdots & \dots & \dots & \vdots \\ \vdots & \vdots & \dots & \dots & \vdots \\ \vdots & \vdots & \dots & \dots & \vdots \\ f_{n1} & f_{n2} & \dots & \dots & f_{nn} \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \\ \vdots \\ \vdots \\ \vdots \\ Q_n \end{pmatrix} \quad (3-71)$$

$$D = FQ \quad (3-72)$$

Si se usan el teorema de Betti y los teoremas de Maxwell, es fácil demostrar que $f_{ij} = f_{ji}$, tal y como se establece formalmente en el siguiente teorema.

Teorema

$$f_{ij} = f_{ji}$$

Demostración

Considérese una carga Q_i aplicada en el grado de libertad i la cual produce una deflexión D_j en el grado de libertad j , por lo que se tiene

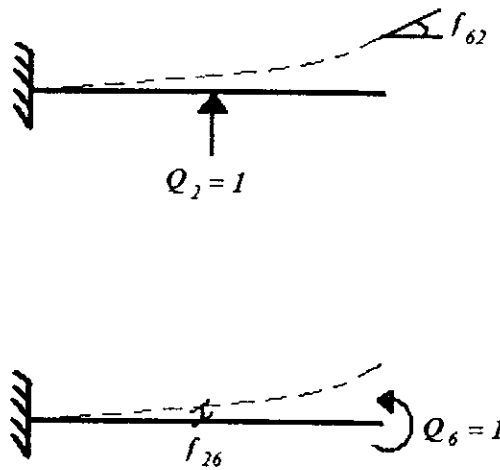
$$D_j = f_{ji} Q_i$$

Además, si se considera que se aplica una carga Q_j en el grado de libertad j ésta provoca un desplazamiento D_i en el grado de libertad i y se tiene que

$$D_i = f_{ij} Q_j$$

Aplicando el teorema de Maxwell, se tiene que si $Q_i = Q_j = 1$ entonces $D_j = D_i$
Por lo que $f_{ji} = f_{ij}$.

Para ilustrar el teorema anterior, considérese otra vez la viga en voladizo, solamente que con los grados de libertad 2 y 6 tal y como se muestra en la siguiente figura 17.



$$f_{26} = f_{62}$$

Figura 17. Simetría de los coeficientes de flexibilidad.

A continuación se da la definición de coeficiente de influencia de rigidez. En la siguiente definición se considera que se aplica sólo una carga Q_j que produce el desplazamiento D_i .

Definición: Coeficiente de influencia de rigidez

$$k_{ji} = \frac{Q_j}{D_i} \quad (3-73)$$

$$Q_j = k_{ji} D_i \quad (3-74)$$

Si se tiene un desplazamiento unitario k_{ji} es igual a Q_j . Dicho de otra forma k_{ji} es la fuerza necesaria aplicada en el grado de libertad j para producir un desplazamiento en una unidad en el grado de libertad i .

Para ilustrar los coeficientes de rigidez se usa otra vez la viga en voladizo donde se aplica una fuerza k_{25} en el grado de libertad 2 para tener un desplazamiento unitario en el grado de libertad 5. Ver figura 18.

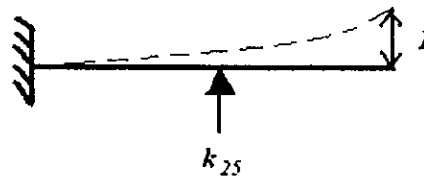


Figura 18. Coeficiente de influencia de rigidez.

Al igual que los coeficientes de flexibilidad, si la estructura es elástica se puede usar el principio de superposición y en términos generales queda de la siguiente forma

$$\begin{aligned} Q_1 &= k_{11}D_1 + k_{12}D_2 + \dots + k_{1n}D_n \\ Q_2 &= k_{21}D_1 + k_{22}D_2 + \dots + k_{2n}D_n \\ &\vdots \\ &\vdots \\ Q_n &= k_{n1}D_1 + k_{n2}D_2 + \dots + k_{nn}D_n \end{aligned} \quad (3-75)$$

En forma matricial se escribe

$$\begin{pmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_n \end{pmatrix} = \begin{pmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_n \end{pmatrix} \quad (3-76)$$

$$Q = KD \quad (3-77)$$

Es fácil en este momento probar que $K=F^{-1}$. Si se tiene que $D=FQ$ y si $Q=KD$ entonces $Q=KFQ$ por lo que $KF=I$ y por lo tanto $K=F^{-1}$.

A la matriz F se le suele llamar matriz de flexibilidad y a la matriz K se le suele llamar matriz de rigidez. También se puede ver fácilmente que K es simétrica debido a F es simétrica por lo que $k_{ij}=k_{ji}$ al igual que $f_{ij}=f_{ji}$. Para lo anterior, lo único que se usa es el que la inversa de una matriz simétrica también es simétrica.

IV. MÉTODO DE RIGIDEZ

4.1 Introducción

El método de rigidez y el método de flexibilidad son fundamentales en el análisis estructural. Estos métodos son aplicables generalmente a todo tipo de estructuras. Métodos más sofisticados, como el método de elementos finitos, son generalizaciones del método de rigidez. Al método de rigidez se le llama método de desplazamientos y al método de flexibilidad se le conoce como el método de fuerzas.

Tanto el método de rigidez como el método de flexibilidad se formulan usando algebra matricial. La formulación de ambos métodos es similar y las matrices proporcionan una notación adecuada. También se tiene que las matrices plantean los problemas en una forma ideal para la programación de computadoras. Este último hecho es una de las razones principales de la popularidad del método de rigidez en las últimas décadas.

El objetivo de esta tesis es desarrollar el método de rigidez, sin embargo se expone brevemente el método de flexibilidad para tener una exposición más completa y explicar el método de rigidez en un contexto más amplio. El método de fuerzas y el método de desplazamientos están íntimamente relacionados y se considera que para una mejor comprensión del método de rigidez es necesario un conocimiento básico del método de flexibilidad.

Al mismo tiempo se harán evidentes las ventajas que presenta el método de rigidez, el cual desde un inicio se le puede atacar en forma sistemática. En cambio, el método de flexibilidad no tiene una forma única para empezar a analizar la estructura y se necesita de bastante criterio para elegir las redundantes. Por esta razón y otras el énfasis es sobre el método de rigidez y al método de flexibilidad sólo se le da una breve introducción.

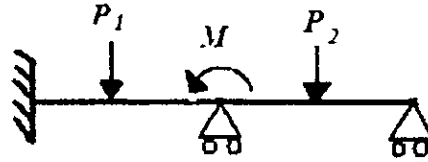
Finalmente, debe comprenderse que los métodos de flexibilidad y de rigidez pueden organizarse hasta formar un procedimiento altamente sistematizado para el análisis de una estructura. Una vez que se han comprendido los conceptos básicos del procedimiento, los métodos pueden aplicarse a estructuras de cualquier grado de dificultad.

4.2 Introducción al método de flexibilidad

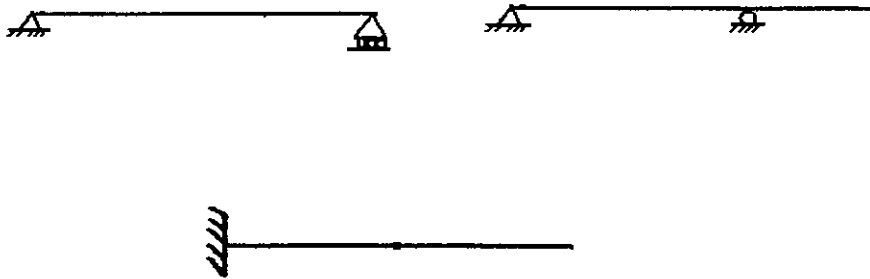
En el capítulo anterior se desarrolló el concepto de coeficiente de influencia y en particular, el coeficiente de influencia de flexibilidad f_{ij} . Estos conceptos expuestos anteriormente son el fundamento para el método de flexibilidad.

Para ilustrar el método de flexibilidad considérese el ejemplo de la figura 19. La viga mostrada en la figura es estáticamente indeterminada de segundo grado, por lo tanto, una estructura libre estáticamente determinada, puede obtenerse liberando dos fuerzas o

acciones redundantes. Se pueden escoger diferentes acciones para las redundantes quedando estructuras libres distintas. La figura 19 ilustra todo lo anterior.



a) Viga estáticamente indeterminada



b) Algunas estructuras libres estáticamente determinadas que pueden escogerse

Figura 19. Estructuras libres de una viga.

De todas las estructuras libres estáticamente determinadas, que se asocian a la estructura original, se debe elegir una de ellas. Esta elección queda a criterio de la persona y es la parte más difícil de sistematizar en el método de flexibilidad.

Una vez elegida la estructura libre asociada a la estructura original, se tiene que esta estructura es estáticamente determinada y que se han eliminado n reacciones Q_i , las cuales se llaman reacciones redundantes o fuerzas redundantes o simplemente las redundantes.

Las redundantes son las incógnitas, las cuales se pueden calcular usando los coeficientes de influencia de flexibilidad. Para ilustrar lo anterior se vuelve al ejemplo de la figura 19. Se escoge como estructura libre la viga en voladizo, por lo que las reacciones redundantes o las redundantes son Q_1 y Q_2 . Q_1 es la reacción del apoyo B y Q_2 es la reacción en el apoyo C. Nótese que para cada redundante se elimina un grado de libertad. Véase figura 20.

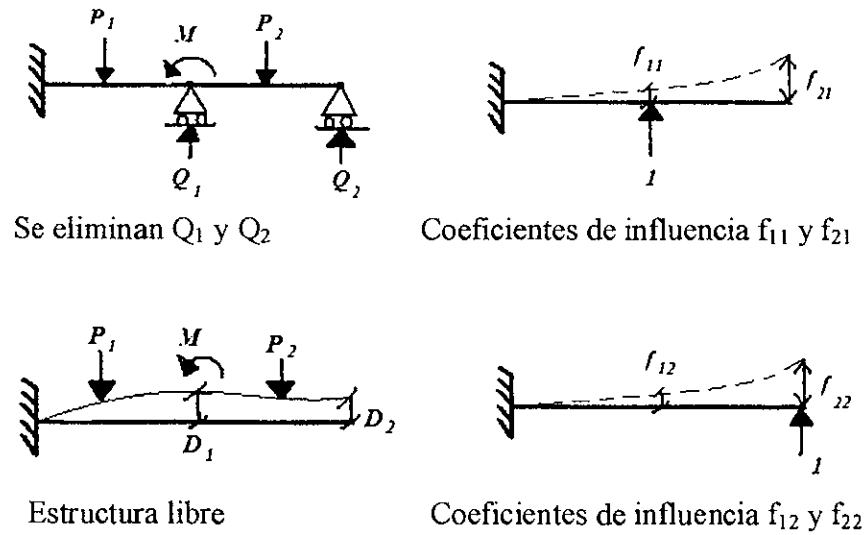


Figura 20. Ejemplo de redundantes.

Suponiendo que los apoyos B y C no tienen ningún desplazamiento real, se pueden plantear las siguientes ecuaciones.

$$D_1 + f_{11}Q_1 + f_{12}Q_2 = 0 \quad (4-1)$$

$$D_1 + f_{21}Q_1 + f_{22}Q_2 = 0$$

Las cuales se pueden generalizar a varios grados de libertad

$$\begin{aligned} D_1 + f_{11}Q_1 + f_{12}Q_2 + \dots + f_{1n}Q_n &= 0 \\ D_2 + f_{21}Q_1 + f_{22}Q_2 + \dots + f_{2n}Q_n &= 0 \\ \vdots & \\ \vdots & \\ D_n + f_{n1}Q_1 + f_{n2}Q_2 + \dots + f_{nn}Q_n &= 0 \end{aligned} \quad (4-2)$$

En forma matricial se puede escribir

$$\begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_n \end{pmatrix} + \begin{pmatrix} f_{11} & f_{12} & \dots & f_{1n} \\ f_{21} & f_{22} & \dots & f_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n1} & f_{n2} & \dots & f_{nn} \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (4-3)$$

o simplemente

$$D + FQ = 0 \quad (4-3)$$

La ecuación 4-3 tiene de incógnitas las reacciones que se han escogido como redundantes. Los coeficientes de influencia f_{ij} y los desplazamientos D_i se calculan de la estructura libre. Obsérvese que la ecuación 4-3 no considera movimientos de apoyos, los cuales coinciden directamente, con los mismos grados de libertad que las redundantes.

Si se consideran los movimientos de apoyos entonces la ecuación 4-3 no se iguala a cero si no que a los desplazamientos de los apoyos. Estos movimientos o desplazamientos se pueden deber a varias razones, por ejemplo, asentamientos. El movimiento de apoyos se escribe D_A el cual es un vector de n componentes donde n es igual al número de redundantes.

$$D_A = \begin{pmatrix} D_{A1} \\ D_{A2} \\ \cdot \\ \cdot \\ \cdot \\ D_{An} \end{pmatrix} \quad (4-4)$$

La ecuación 4-3 queda en forma completa de la siguiente forma

$$D + FQ = D_A \quad (4-5)$$

Si se despeja Q de la ecuación 4-5 se tiene que

$$Q = F^{-1} (D_A - D) \quad (4-6)$$

Como ya se vio en el capítulo anterior la inversa F^{-1} de la matriz de flexibilidad F , tiene un significado físico que es la matriz de rigidez K , o sea que $K = F^{-1}$.

Debido a que en el método de flexibilidad las incógnitas son las reacciones redundantes, este método es recomendable cuando el número de las redundantes sea reducido.

En el análisis estructural se requiere que además de las fuerzas Q , que da el método de flexibilidad, se calculen desplazamientos y fuerzas internas en diferentes puntos de la estructura. A estos puntos de interés para calcular desplazamientos y fuerzas internas se les llaman nodos. Para miembros prismáticos las fuerzas internas son, típicamente, el corte, la fuerza axial, el momento flexionante y el momento de torsión.

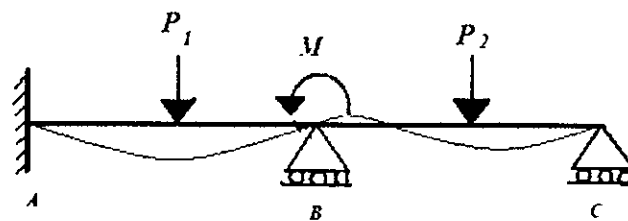
El método de rigidez resulta ser más práctico que el de flexibilidad para el cálculo de estas fuerzas internas, además de que desde un inicio se calculan los desplazamientos.

4.3 Método de rigidez

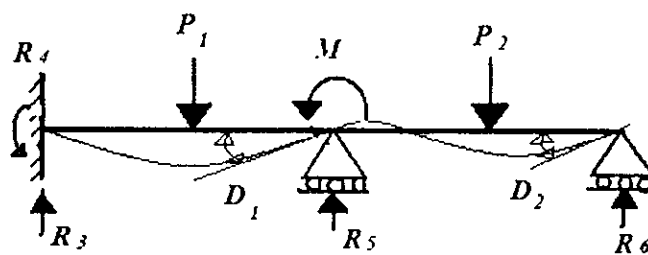
El método de rigidez también llamado método de desplazamientos se distingue del método de flexibilidad en los conceptos físicos, aunque los métodos son similares en su formulación matemática. En ambos métodos se usa el principio de superposición para el planteamiento de las ecuaciones fundamentales.

En el método de rigidez las incógnitas son los desplazamientos en los nodos de la estructura, mientras que en el método de flexibilidad las incógnitas son las fuerzas redundantes. El método de rigidez hace un uso extensivo de acciones y desplazamientos en miembros prismáticos doblemente empotrados.

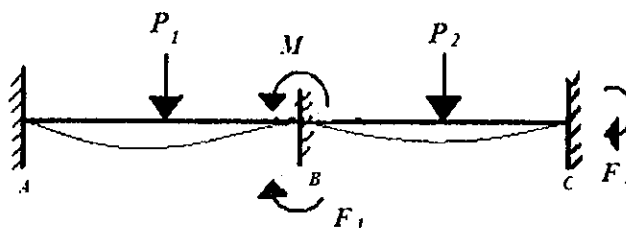
Para ilustrar el método de rigidez considérese la viga mostrada en la figura 21, la cual es una viga estáticamente indeterminada de segundo grado. Esta viga tiene un módulo de elasticidad E y un momento de inercia I y está sujeta a las cargas P_1 , M y P_2 . Los nodos B y C pueden rotar libremente, así que las incógnitas son los giros en los nodos B y C y estos giros se denotan por D_1 y D_2 respectivamente. O sea que D_1 es la rotación del nodo B y D_2 es la rotación del nodo C, tal y como se muestra en la parte a de la figura 21.



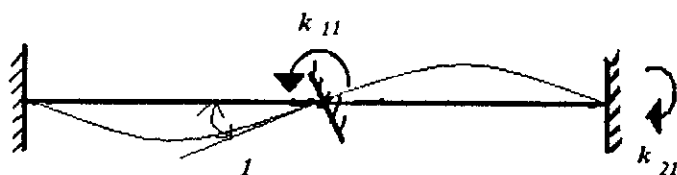
a) Viga original mostrando sólo acciones



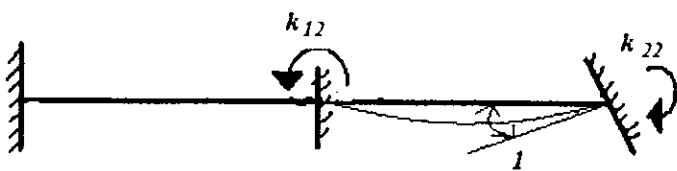
b) Viga original mostrando acciones, reacciones y giros



c) Viga con nodos empotrados, mostrando F_1 y F_2



d) .Nodo B con giro unitario, mostrando k_{11} y k_{21}



e) .Nodo C con giro unitario, mostrando k_{12} y k_{22}

Figura 21. Viga de dos luces.

El primer paso en el análisis por el método de rigidez consiste en aplicar fijaciones imaginarias en los nodos para prevenir todos los desplazamientos. Esta estructura fija se ilustra en la figura 21 c.

Para tener una notación consistente durante todo este trabajo, en el método de rigidez se usará R_i para las reacciones en todos los grados de libertad en la estructura original y F_i para las reacciones cuando se restringen imaginariamente todos los grados de libertad. Además para los coeficientes de rigidez se usará k_{ij} .

Las reacciones R_i junto con los desplazamientos D_i son las incógnitas. Las reacciones F_i debida a los grados de libertad restringidos imaginariamente se calculan tal y como se explicará posteriormente, o auxiliándose con la tabla 1 y por último, los coeficientes de rigidez se pueden calcular tal y como se explicará posteriormente.

Para generar los coeficientes de rigidez en los nodos B y C de la viga en cuestión, se inducen separadamente en la estructura fija valores de los desplazamientos desconocidos D_1 y D_2 . Un desplazamiento unitario correspondiente a D_1 consiste en una rotación unitaria en el nodo B tal y como se muestra en la figura 21 d. El desplazamiento D_2 permanece igual a cero por lo que las acciones correspondientes k_{11} y k_{21} son los coeficientes de rigidez. Estas rigideces están formadas por los pares producidos por las fijaciones en la viga en los nodos B y C. El cálculo de estos coeficientes de rigidez no es difícil cuando se cuenta con fórmulas para las reacciones en vigas empotradas.

Ahora se pueden plantear dos ecuaciones usando el principio de superposición. Las reacciones en los nodos B y C correspondientes a los giros D_1 y D_2 son cero en la estructura original por lo que

$$\begin{aligned} k_{11}D_1 + k_{12}D_2 + F_1 &= 0 \\ k_{21}D_1 + k_{22}D_2 + F_2 &= 0 \end{aligned} \quad (4-7)$$

si se escribe en forma matricial

$$\begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} + \begin{pmatrix} F_1 \\ F_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (4-8)$$

En términos generales se puede escribir

$$\begin{aligned} k_{11}D_1 + k_{12}D_2 + \dots + k_{1n}D_n + F_1 &= 0 \\ k_{21}D_1 + k_{22}D_2 + \dots + k_{2n}D_n + F_2 &= 0 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ k_{n1}D_1 + k_{n2}D_2 + \dots + k_{nn}D_n + F_n &= 0 \end{aligned}$$

en forma matricial queda

$$\begin{pmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \\ \cdot \\ \cdot \\ \cdot \\ D_n \end{pmatrix} + \begin{pmatrix} F_1 \\ F_2 \\ \cdot \\ \cdot \\ \cdot \\ F_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{pmatrix}$$

o simplemente

$$KD + F = 0 \quad (4-9)$$

Con la ecuación anterior se pueden hallar las incógnitas D , por medio del siguiente despeje.

$$D = -K^{-1}F \quad (4-10)$$

Vale la pena hacer notar que siempre se consideran todas las cargas que actúan inicialmente sobre la estructura, sin importar si la carga actúa directamente sobre un nodo en particular justamente donde se tiene un grado de libertad que se va a restringir. Se menciona lo anterior porque hay personas que prefieren hacer la diferencia entre una carga

aplicada directamente en un grado de libertad y otra carga que está aplicada en un punto que no es un nodo, esta distinción no es necesaria y en esta tesis no se considerará.

Una vez encontrados los desplazamientos D_i se pueden calcular las reacciones R_i , que son las otras incógnitas. Estos cálculos se pueden realizar de dos formas. La primera forma es aislar el miembro en cuestión de nodo a nodo y efectuar los cálculos usando conocimientos de resistencia de materiales. Debido a que ya se conocen los desplazamientos se pueden hallar las acciones R_i . La otra forma es la del método de rigidez, la cual ataca sistemáticamente el cálculo de las reacciones R_i y es la que se usa en este trabajo y se explica a continuación.

Para calcular las reacciones R_i se calculan antes la matriz K_{RL} y el vector F_R que se ilustra en la figura 22

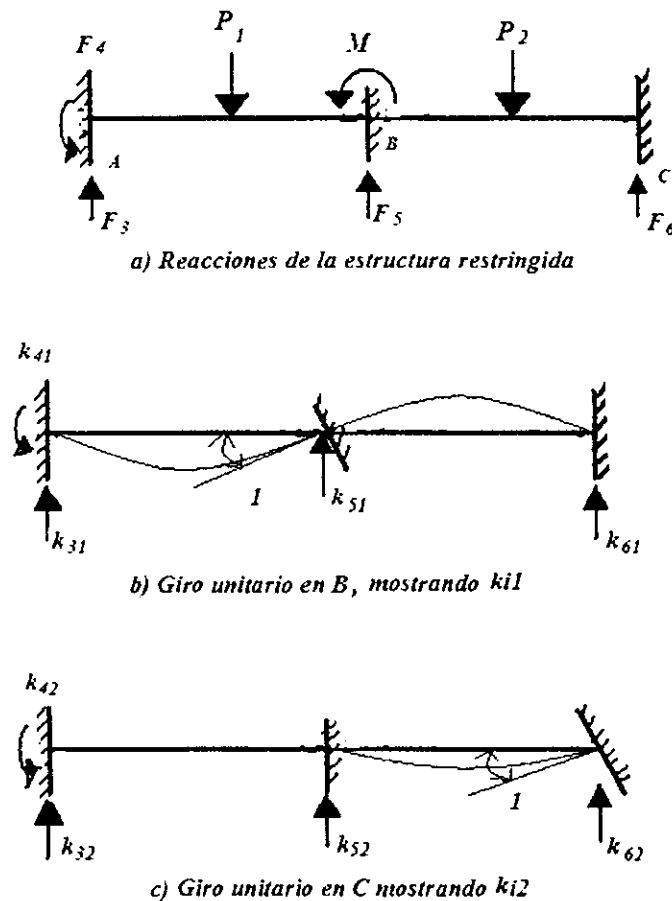


Figura 22. Viga de dos luces mostrando rigideces.

Las componentes F_i del vector F_R se calculan restringiendo los giros en el nodo B y en el nodo C y luego calculando las reacciones que corresponden a los grados de libertad de las R_i , esto se ilustra en la figura 22a. Para calcular las componentes k_{ij} de la matriz K_{RL} se calculan los coeficientes de rigidez de un giro unitario en B en los grados de libertad

correspondientes a las R_i y también con un giro unitario en C. Usando el principio de superposición se pueden plantear las siguientes ecuaciones

$$R_3 = F_3 + k_{31}D_1 + k_{32}D_2$$

$$R_4 = F_4 + k_{41}D_1 + k_{42}D_2$$

$$R_5 = F_5 + k_{51}D_1 + k_{52}D_2$$

$$R_6 = F_6 + k_{61}D_1 + k_{62}D_2$$

en forma matricial se puede escribir

$$\begin{pmatrix} R_3 \\ R_4 \\ R_5 \\ R_6 \end{pmatrix} = \begin{pmatrix} F_3 \\ F_4 \\ F_5 \\ F_6 \end{pmatrix} + \begin{pmatrix} k_{31} & k_{32} \\ k_{41} & k_{42} \\ k_{51} & k_{52} \\ k_{61} & k_{62} \end{pmatrix} \begin{pmatrix} D_1 \\ D_2 \end{pmatrix}$$

o simplemente

$$R = F_R + K_{RL}D$$

La matriz de rigidez total se puede escribir como se muestra a continuación

$$K_T = \begin{pmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \\ k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} \\ k_{41} & k_{42} & k_{43} & k_{44} & k_{45} & k_{46} \\ k_{51} & k_{52} & k_{53} & k_{54} & k_{55} & k_{56} \\ k_{61} & k_{62} & k_{63} & k_{64} & k_{65} & k_{66} \end{pmatrix}$$

4.4 Rigidez de un miembro prismático

En esta sección se considerará exclusivamente un miembro prismático de longitud L , área transversal A y momento de inercia I . Se consideran dos nodos, cada uno de ellos en uno de los extremos. Cada nodo tiene tres grados de libertad, los cuales son desplazamiento horizontal, desplazamiento vertical y giro. Debido a que se tienen dos nodos, hay un total de seis grados de libertad tal y como se muestra en la figura 23.

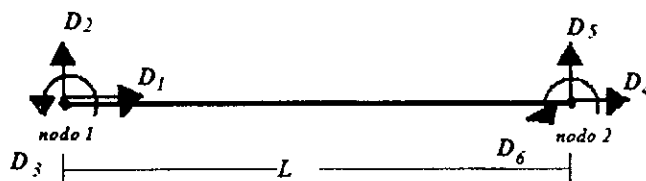


Figura 23. Grados de libertad de miembros prismáticos.

Para calcular los coeficientes de rigidez para este miembro prismático se usará el teorema de Castigliano, el cual nos dice que $D_i = \frac{\partial W_I}{\partial Q_i}$. La fórmula para el trabajo interno en un miembro prismático ya se ha deducido en el capítulo 3 y es la fórmula 3-42 que se vuelve a escribir para una referencia rápida

$$W_I = \int_0^L \left(\frac{P_x^2}{2EA} + \frac{M_x^2}{2EI} \right) dx$$

Recuérdese que los coeficientes de rigidez son simétricos, por lo que $k_{ij} = k_{ji}$. En la fórmula 3-42 P_x es la fuerza axial en función de x y M_x es el momento en función de x .

Se harán los cálculos en tres etapas. La primera etapa es un desplazamiento unitario para D_1 , luego un desplazamiento unitario para D_2 y por último un giro unitario para D_3 .

En la primera etapa se calcula k_{11} y por estática se puede determinar k_{41} . El desplazamiento unitario para D_1 junto con k_{11} y k_{41} se muestra en la figura 24.

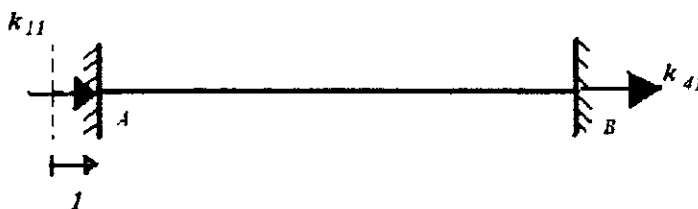


Figura 24. Primera etapa del cálculo de rigideces.

Para este caso $P_{(x)} = k_{11}$ y $M_{(x)} = 0$ por lo que

$$W_I = \int_0^L \frac{k_{11}^2}{2EA} dx$$

aplicando el teorema de Castigliano

$$\frac{\partial W_I}{\partial k_{11}} = \int_0^L \frac{2k_{11}}{2EA} dx$$

$$D_1 = \frac{k_{11}x}{EA} \Big|_0^L$$

$$1 = \frac{k_{11}L}{EA}$$

$$\boxed{k_{11} = \frac{EA}{L}} \quad (4-11)$$

Por estática $k_{41} = -k_{11}$, y se tiene que

$$\boxed{k_{41} = -\frac{EA}{L}} \quad (4-12)$$

En la segunda etapa se calculan k_{22} , k_{32} , k_{52} y k_{62} imponiendo exclusivamente un desplazamiento unitario para D_2 . Ver figura 25.

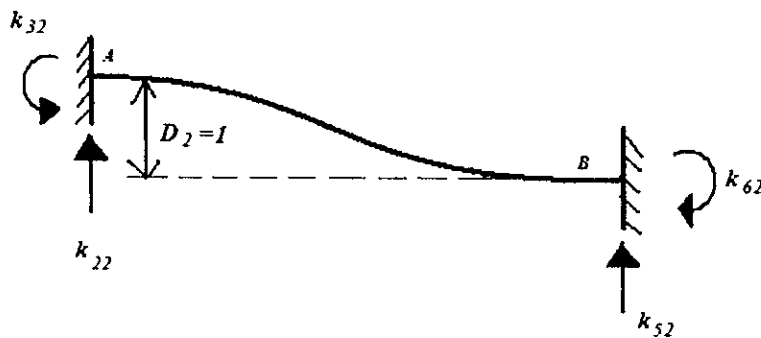


Figura 25. Segunda etapa del cálculo de rigideces.

En este caso se tiene que $P_{(x)} = 0$ y $M_{(x)} = k_{32} - k_{22}x$ por lo que

$$W_I = \int_0^L \frac{(k_{32} - k_{22}x)^2}{2EI} dx$$

aplicando el teorema de Castigliano

$$\frac{\partial W_I}{\partial k_{22}} = \int_0^L \frac{(k_{32} - k_{22}x)(-x)}{EI} dx$$

en este caso $\frac{\partial W_l}{\partial k_{22}} = D_2 = 1$ por lo que

$$1 = \frac{1}{EI} \left(\frac{-k_{32}x^2}{2} + \frac{1}{3}k_{22}x^3 \right) \Big|_0^L$$

$$1 = \frac{L^2}{EI} \left(\frac{L}{3}k_{22} - \frac{1}{2}k_{32} \right) \quad (4-13)$$

además se tiene que $\frac{\partial W_l}{\partial k_{32}} = D_3 = 0$

$$\frac{\partial W_l}{\partial k_{32}} = \int_0^L \frac{(k_{32} - k_{22}x)}{EI} dx$$

$$0 = \frac{1}{EI} \left(k_{32}x - \frac{k_{22}x^2}{2} \right) \Big|_0^L$$

$$0 = \frac{L}{EI} \left(k_{32} - \frac{k_{22}L}{2} \right) \quad (4-14)$$

trabajando simultáneamente las ecuaciones (4-13) y (4-14)

$$k_{32} = \frac{k_{22}L}{2}$$

sustituyendo

$$1 = \frac{L^3}{EI} \left(\frac{k_{22}}{3} - \frac{k_{22}}{4} \right)$$

se obtiene el valor k_{22}

$$1 = \frac{L^3}{EI} \left(\frac{k_{22}}{12} \right)$$

$$\boxed{k_{22} = \frac{12EI}{L^3}} \quad (4-15)$$

y el valor de k_{32}

$$k_{32} = \frac{6EI}{L^2} \quad (4-16)$$

Si se usa ahora equilibrio estático se obtiene que

$$k_{52} = -\frac{12EI}{L^3} \quad (4-17)$$

$$k_{62} = -\frac{6EI}{L^2} \quad (4-18)$$

Para la tercera y última etapa se calculan los coeficientes k_{23} , k_{33} , k_{53} y k_{63} imponiendo exclusivamente un giro unitario en el nodo A que corresponde a $D_3=1$. En la figura 26 se ilustra esta etapa.

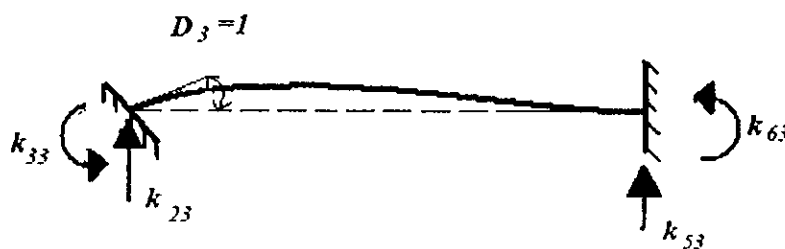


Figura 26. Tercera etapa del cálculo de rigideces.

En este caso se tiene que $P_{(x)}=0$ y $M_{(x)}=k_{33} - k_{23}x$ por lo que

$$W_1 = \int_0^L \frac{(k_{33} - k_{23}x)^2}{2EI} dx$$

aplicando el teorema de Castigliano

$$\frac{\partial W_1}{\partial k_{23}} = \int_0^L \frac{2(k_{33} - k_{23}x)(-x)}{EI} dx$$

en este caso $\frac{\partial W_1}{\partial k_{23}} = D_2 = 0$ por lo que

$$0 = \frac{1}{EI} \left(\frac{-k_{33}x^2}{2} + \frac{1}{3}k_{23}x^3 \right) \Big|_0^L$$

$$0 = \frac{k_{23}}{3}L - \frac{k_{33}}{2} \quad (4-19)$$

además se tiene que $\frac{\partial W_1}{\partial k_{33}} = D_3 = 1$

$$\frac{\partial W_1}{\partial k_{33}} = \int_0^L \frac{(k_{33} - k_{23}x)}{EI} dx$$

$$1 = \frac{1}{EI} \left(k_{33}x - \frac{k_{23}x^2}{2} \right) \Big|_0^L$$

$$1 = \frac{1}{EI} \left(k_{33}L - \frac{k_{23}L^2}{2} \right) \quad (4-20)$$

trabajando simultáneamente las ecuaciones (4-19) y (4-20)

$$k_{33} = \frac{2}{3}k_{23}L$$

sustituyendo

$$EI = k_{33}L \left(1 - \frac{3}{4} \right)$$

$$\boxed{k_{33} = \frac{4EI}{L}} \quad (4-21)$$

$$\boxed{k_{23} = \frac{6EI}{L^2}} \quad (4-22)$$

si se usa equilibrio estático se obtiene

$$\boxed{k_{53} = -\frac{6EI}{L^2}}$$

$$\boxed{k_{63} = \frac{2EI}{L}}$$

Cuando se aplicó el teorema de Betti y el de Maxwell se concluyó que los coeficientes de rigidez son simétricos es decir que $k_{ij}=k_{ji}$. Con todo lo anterior se puede escribir la matriz de rigidez de miembro, la cual se denotará por K_M .

$$K_M = \begin{pmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{pmatrix}$$

4.5 Fuerzas equivalentes en los nodos

En el método de rigidez todos los grados de libertad se restringen, esto hace que el método de rigidez se pueda sistematizar con mayor facilidad que el método de flexibilidad. Al restringir todos los grados de libertad, siempre quedan miembros prismáticos doblemente empotrados. En esta sección se explica cómo calcular las reacciones en los nodos en términos generales, usando el teorema de Castigliano, y luego se dan algunos ejemplos.

Considérese un miembro prismático doblemente empotrado cualquiera sometido a cierta distribución de cargas como se muestra en la figura 27.

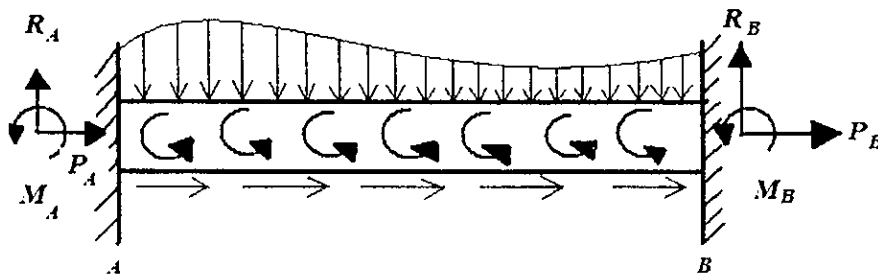


Figura 27. Miembro doblemente empotrado.

En la figura 27 se muestran también las reacciones en los nodos empotrados. Las cargas sobre el miembro prismático producen una fuerza axial $p_{(x)}$, una fuerza de corte $v_{(x)}$ y un momento $m_{(x)}$. Debido a que la contribución del corte a la energía interna es despreciable sólo se toman en cuenta $p_{(x)}$ y $m_{(x)}$. Si además se consideran las reacciones, se tienen las siguientes expresiones para $P_{(x)}$ y $M_{(x)}$.

$$P_{(x)} = P_A + p_{(x)} \quad (4-23)$$

$$M_{(x)} = M_A - R_A x + m_{(x)} \quad (4-24)$$

el trabajo interno en un miembro prismático es

$$W_I = \int_0^L \left(\frac{P_x^2}{2EA} + \frac{M_x^2}{2EI} \right) dx$$

aplicando el teorema de Castigliano y considerando que el nodo A no se desplaza

$$\frac{\partial W_I}{\partial P_A} = \frac{\partial W_I}{\partial M_A} = \frac{\partial W_I}{\partial R_A} = 0.$$

De lo anterior se obtienen tres ecuaciones

$$\begin{aligned} \frac{\partial W_I}{\partial P_A} &= \int_0^L \frac{P_{(x)}}{EA} \frac{\partial P_{(x)}}{\partial P_A} dx \\ \int_0^L \frac{P_{(x)}}{EA} dx &= 0. \end{aligned} \quad (4-25)$$

La ecuación anterior es la primera, para plantear la segunda ecuación se procede de la siguiente forma

$$\begin{aligned} \frac{\partial W_I}{\partial R_A} &= \int_0^L \frac{M_{(x)}}{EI} \frac{\partial M_{(x)}}{\partial R_A} dx \\ \int_0^L \frac{-xM_{(x)}}{EI} dx &= 0 \end{aligned} \quad (4-26)$$

la tercera y última ecuación se plantea así

$$\frac{\partial W_I}{\partial M_A} = \int_0^L \frac{M_{(x)}}{EI} \frac{\partial M_{(x)}}{\partial M_A} dx$$

$$\int_0^L \frac{M_{(x)}}{EI} dx = 0 \quad (4-27)$$

de las tres ecuaciones anteriores se obtienen las tres incógnitas R_A , P_A y M_A . Para obtener R_B , P_B y M_B se utiliza equilibrio estático.

A continuación se exponen dos ejemplos para ilustrar la teoría que se ha explicado. El primer ejemplo es una viga doblemente empotrada con una carga vertical uniformemente distribuida, como se muestra en la figura 28

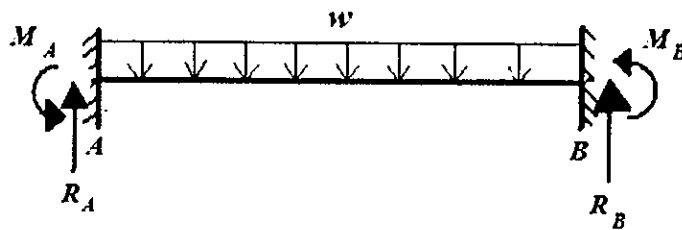


Figura 28. Miembro doblemente empotrado con carga distribuida.

Para este primer ejemplo

$$P_{(x)} = 0$$

$$M_{(x)} = M_A - R_A x + \frac{w}{2} x^2$$

por lo que

$$\int_0^L \frac{-xM_{(x)}}{EI} dx = \frac{1}{EI} \int_0^L \left(R_A x^2 - M_A x - \frac{w}{2} x^3 \right) dx$$

$$\int_0^L \frac{-xM_{(x)}}{EI} dx = \frac{1}{EI} \left(\frac{R_A L^3}{3} - \frac{M_A L^2}{2} - \frac{w L^4}{8} \right)$$

$$\frac{R_A L}{3} - \frac{M_A}{2} = \frac{w L^2}{8} \quad (4-28)$$

además

$$\int_0^L \frac{M(x)}{EI} dx = \frac{1}{EI} \int_0^L \left(M_A - R_A x + \frac{w}{2} x^2 \right) dx$$

$$\int_0^L \frac{M(x)}{EI} dx = \frac{1}{EI} \left(M_A L - \frac{R_A L^2}{2} + \frac{w L^3}{6} \right)$$

$$\frac{R_A L}{2} - M_A = \frac{w L^2}{6} \quad (4-29)$$

resolviendo simultáneamente (4-28) y (4-29) se tiene que

$$R_A = \frac{wL}{2} \quad \text{y} \quad M_A = \frac{wL^2}{12}$$

por equilibrio estático

$$R_B = \frac{wL}{2} \quad \text{y} \quad M_B = -\frac{wL^2}{12}$$

El segundo ejemplo es una viga doblemente empotrada con una carga vertical puntual como se muestra en la figura 29.

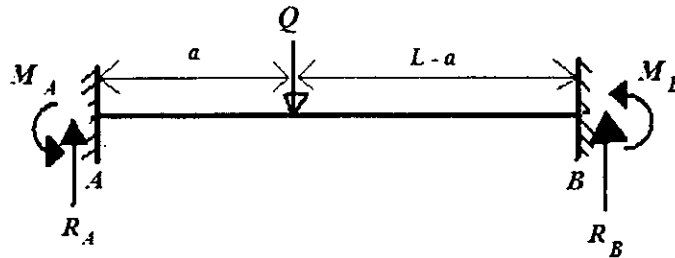


Figura 29. Miembro doblemente empotrado con carga puntual.

En este segundo ejemplo

$$P_{(x)} = 0$$

$$M_{(x)} = \begin{cases} M_A - R_A x & 0 \leq x \leq a \\ M_A - R_A x + Q(x-a) & a \leq x \leq L \end{cases}$$

por lo que

$$\int_0^L \frac{-xM(x)}{EI} dx = \frac{1}{EI} \int_0^a (M_A - R_A x)(-x) dx + \frac{1}{EI} \int_a^L (M_A - R_A x + Q(x-a))(-x) dx$$

$$\int_0^L \frac{-xM(x)}{EI} dx = \frac{1}{EI} \left(\frac{R_A L^3}{3} - \frac{M_A L^2}{2} + \frac{QaL^2}{2} - \frac{QL^3}{3} - \frac{Qa^3}{6} \right)$$

y además

$$\int_0^L \frac{M(x)}{EI} dx = \frac{1}{EI} \int_0^a (M_A - R_A x) dx + \frac{1}{EI} \int_a^L (M_A - R_A x + Q(x-a)) dx$$

$$\int_0^L \frac{M(x)}{EI} dx = \frac{1}{EI} \left(M_A L - \frac{R_A L^2}{2} + \frac{QL^2}{2} - QaL + \frac{Qa^2}{2} \right)$$

de lo anterior se plantean las siguientes ecuaciones simultaneas

$$\frac{R_A L^3}{3} - \frac{M_A L^2}{2} = \frac{QL^3}{3} + \frac{Qa^3}{6} - \frac{QaL^2}{2}$$

$$\frac{R_A L^2}{2} - M_A L = \frac{QL^2}{2} + \frac{Qa^2}{2} - QaL$$

y se obtienen los siguientes resultados

$$R_A = \frac{Q(L-a)^2}{L^3} (L+2a) \quad (4-30)$$

$$M_A = \frac{Qa(L-a)^2}{L^2} \quad (4-31)$$

y por equilibrio estático se obtiene

$$R_B = \frac{Qa^2(3L-2a)}{L^3} \quad (4-32)$$

$$M_B = -\frac{Qa^2(L-a)}{L^2} \quad (4-33)$$

Para el caso especial de $a = \frac{L}{2}$ se pueden simplificar

$$R_A = R_B = \frac{Q}{2}$$

$$M_A = -M_B = \frac{QL}{8}.$$

Para una exposición más completa, los dos ejemplos anteriores junto con otros casos se presentan convenientemente en la tabla 1.

4.6 Matriz de rotación

Los marcos rígidos y en general cualquier estructura pueden tener miembros prismáticos inclinados, haciendo cualquier ángulo respecto a otro miembro. Lo anterior hace necesario que se considere la rotación de sistemas de coordenadas en términos generales. Para el caso particular de marcos rígidos con vigas y columnas, a las columnas se les considerará miembros rotados 90° .

Para iniciar la descripción matemática de las rotaciones considérese un vector V en el plano como se muestra en la figura 30.

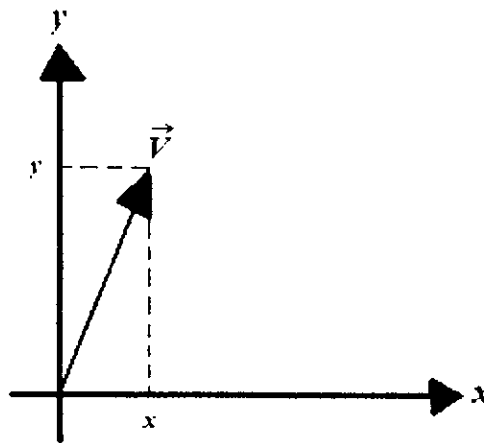
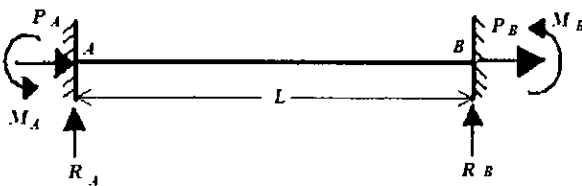
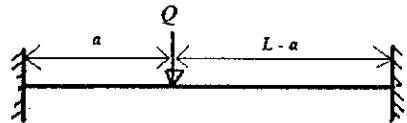
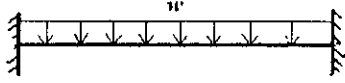
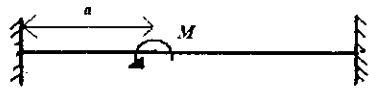
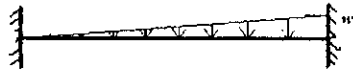
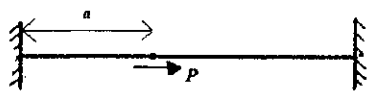
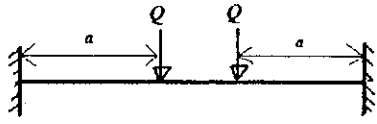
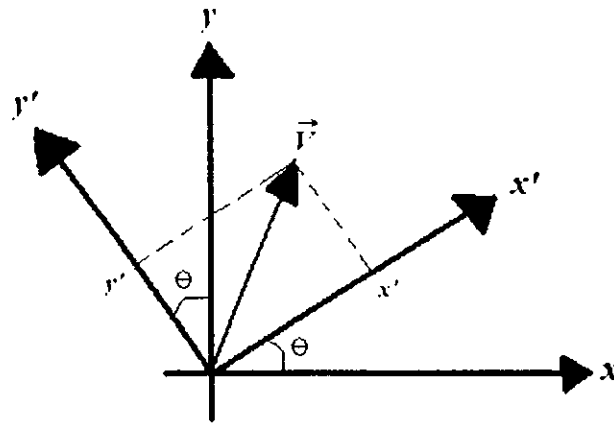


Figura 30. Vector en el plano.

El vector V tiene componentes (x,y) con respecto a este sistema de referencia y se puede escribir $\vec{V} = \begin{pmatrix} x \\ y \end{pmatrix}$. Si ahora se considera otro sistema de coordenadas rotado un ángulo θ tal y como se muestra en la figura 31a V se le podría expresar por medio de las componentes (x', y') y se podría escribir $\vec{V} = \begin{pmatrix} x' \\ y' \end{pmatrix}$.

Tabla 1
Reacciones de empotramiento producidas por algunas cargas

	
 $M_A = \frac{Qa(L-a)^2}{L^2} \quad M_B = -\frac{Qa^2(L-a)}{L^2}$ $R_A = \frac{Q(L-a)^2(L+2a)}{L^3} \quad R_B = \frac{Qa^2(3L-2a)}{L^3}$	 $M_A = -M_B = \frac{wL^2}{12}$ $R_A = R_B = \frac{wL}{2}$
 $M_A = \frac{M(L-a)(3a-L)}{L^2} \quad M_B = \frac{Ma(2L-3a)}{L^2}$ $R_A = -R_B = \frac{6Ma(L-a)}{L^3}$	 $M_A = \frac{wL^2}{30} \quad M_B = -\frac{wL^2}{20}$ $R_A = \frac{3wL}{20} \quad R_B = \frac{7wL}{20}$
 $P_A = -\frac{P(L-a)}{L} \quad P_B = -\frac{Pa}{L}$	 $M_A = -M_B = \frac{Qa(L-a)}{L}$ $R_A = R_B = Q$



Rotación de ejes

Figura 31. Rotación de ejes.

De lo anterior se puede relacionar fácilmente las componentes (x, y) con las componentes (x', y') de la siguiente forma

$$\begin{aligned} x &= x' \cos \theta - y' \operatorname{sen} \theta \\ y &= x' \operatorname{sen} \theta + y' \cos \theta \end{aligned} \quad (4-34)$$

La expresión anterior se puede escribir tomando la transformación inversa, es decir que las componentes (x', y') se expresan en términos de (x, y) , con lo que se obtiene.

$$\begin{aligned} x' &= x \cos \theta + y \operatorname{sen} \theta \\ y' &= -x \operatorname{sen} \theta + y \cos \theta \end{aligned} \quad (4-35)$$

Si las expresiones anteriores se escriben en forma matricial se obtiene lo siguiente

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \theta & -\operatorname{sen} \theta \\ \operatorname{sen} \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \quad (4-36)$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \theta & \operatorname{sen} \theta \\ -\operatorname{sen} \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (4-37)$$

Si además se considera el eje z , el cual se dejará fijo para el caso de marcos planos se tiene que

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos \theta & -\operatorname{sen} \theta & 0 \\ \operatorname{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} \quad (4-38)$$

Para este trabajo de tesis se definirá la matriz de rotación $r_{(\theta)}$ de la siguiente forma

$$r_{(\theta)} = \begin{pmatrix} \cos \theta & -\text{sen} \theta & 0 \\ \text{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (4-39)$$

Tal y como se ha expuesto es inmediato que la matriz $R_{(\theta)}$ es la matriz que transforma las componentes (x', y', z') a las componentes (x, y, z) y que la matriz que transforma las componentes (x, y, z) a (x', y', z') es $R_{(-\theta)}$. Algunas conclusiones fáciles que se pueden sacar sobre la matriz $R_{(\theta)}$ son las siguientes:

- 1) $r_{(\theta)}^{-1} = r_{(-\theta)}$.
- 2) $\text{Det} r_{(\theta)} = 1$ El determinante de $r_{(\theta)}$ es 1.
- 3) La matriz $r_{(\theta)}$ es ortogonal.
- 4) $r_{(\theta)}^T = r_{(\theta)}^{-1}$. La transpuesta de $r_{(\theta)}$ es igual a su inversa.

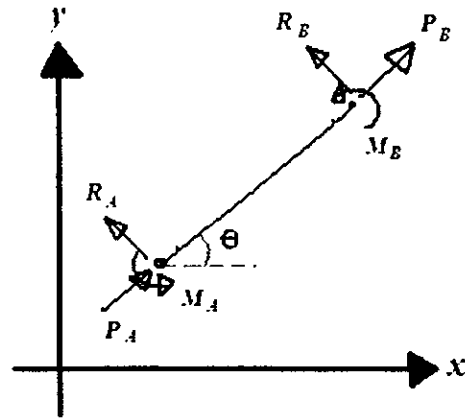
En el método de rigidez cada miembro considera dos nodos, el nodo inicial A y el nodo final B, en cada nodo se expresan las tres componentes de las fuerzas que son la fuerza axial, la fuerza de corte y el momento. Por lo que para cada miembro se tienen dos ternas de fuerza y si se quiere rotar todo el miembro hay que considerar la matriz de rotación total $R_{(\theta)}$ definida de la siguiente forma

$$R_{(\theta)} = \left(\begin{array}{ccc|ccc} \cos \theta & -\text{sen} \theta & 0 & 0 & 0 & 0 \\ \text{sen} \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & \cos \theta & -\text{sen} \theta & 0 \\ 0 & 0 & 0 & \text{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right)$$

en forma resumida

$$R_{(\theta)} = \left(\begin{array}{c|c} r_{(\theta)} & 0 \\ \hline 0 & r_{(\theta)} \end{array} \right) \quad (4-40)$$

Así, si se tiene un miembro prismático inclinado como se muestra en la figura 32 y la fuerza en los nodos se pueden transformar a sus componentes en el eje x horizontal y el eje y vertical.



*Miembro prismático inclinado
a un ángulo θ*

Figura 32. Miembro prismático inclinado a un ángulo θ .

En la figura anterior se exponen las seis fuerzas en los extremos del miembro, las cuales se pueden escribir como un vector columna

$$F' = \begin{pmatrix} P_A \\ R_A \\ M_A \\ P_B \\ R_B \\ M_B \end{pmatrix}$$

El vector de fuerza anterior tiene sus componentes referidas a un sistema girado un ángulo θ . Si se quiere referir este vector de fuerza al sistema de referencia x horizontal y y vertical se aplica la matriz de rotación total de la siguiente forma

$$\begin{pmatrix} F_{Ax} \\ F_{Ay} \\ F_{Az} \\ F_{Bx} \\ F_{By} \\ F_{Bz} \end{pmatrix} = \begin{pmatrix} \cos \theta & -\text{sen} \theta & 0 & 0 & 0 & 0 \\ \text{sen} \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & \cos \theta & -\text{sen} \theta & 0 \\ 0 & 0 & 0 & \text{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} P_A \\ R_A \\ M_A \\ P_B \\ R_B \\ M_B \end{pmatrix} \quad (4-41)$$

en forma resumida queda

$$F = R_{(\theta)} F' \quad (4-42)$$

Con los desplazamientos se puede hacer exactamente lo mismo que con las fuerzas por lo que se puede escribir la siguiente ecuación

$$\begin{pmatrix} D_{Ax} \\ D_{Ay} \\ D_{Az} \\ D_{Bx} \\ D_{By} \\ D_{Bz} \end{pmatrix} = \begin{pmatrix} \cos \theta & -\text{sen} \theta & 0 & 0 & 0 & 0 \\ \text{sen} \theta & \cos \theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & \cos \theta & -\text{sen} \theta & 0 \\ 0 & 0 & 0 & \text{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} D_{Ax'} \\ D_{Ay'} \\ D_{Az'} \\ D_{Bx'} \\ D_{By'} \\ D_{Bz'} \end{pmatrix}$$

o simplemente

$$D = R_{(\theta)} D' \quad (4-43)$$

Para la matriz de rigidez de un miembro prismático se considera primero en un sistema de coordenadas locales donde x' es el eje que se va del nodo inicial A al nodo final B, y' es un eje perpendicular al plano del marco, ver figura 33.

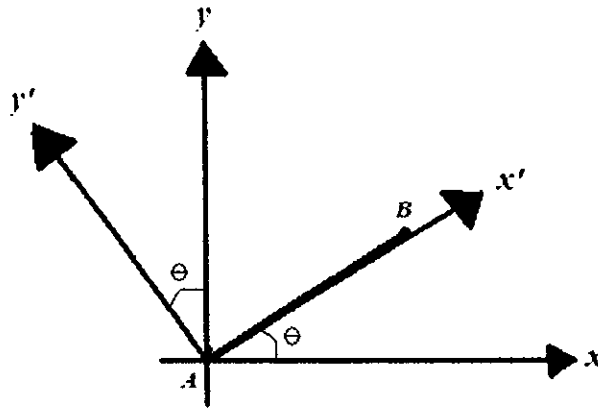


Figura 33. Nodos A y B referidos a dos sistemas.

Referido al sistema (x', y', z') se tiene la matriz de rigidez K' , por lo que se tiene relación entre los desplazamientos y fuerzas de la siguiente forma

$$F' = K' D' \quad (4-44)$$

La ecuación anterior se va a transformar para el sistema (x, y, z) , generalmente considerado como el sistema global, usándose la matriz de rotación $R_{(\theta)}$

$$R_{(\theta)} F' = R_{(\theta)} K' D'$$

ya se tiene que

$$F = R_{(\theta)} F'$$

por lo que

$$F = R_{(\theta)} K' D'$$

se inserta la identidad I entre K' y D'

$$F = R_{(\theta)} K' I D'$$

se expresa la identidad I como $R_{(\theta)}^T R_{(\theta)}$

$$F = R_{(\theta)} K' R_{(\theta)}^T R_{(\theta)} D'$$

ya se tiene que

$$D = R_{(\theta)} D'$$

por lo que

$$F = R_{(\theta)} K' R_{(\theta)}^T D \quad (4-45)$$

se obtiene la matriz de rigidez K respecto al sistema (x, y, z)

$$\boxed{K = R_{(\theta)} K' R_{(\theta)}^T} \quad (4-46)$$

y finalmente se tiene la ecuación original

$$F = KD \quad (4-47)$$

4.7 Matriz de rigidez total

En el método de rigidez, todos los coeficientes de rigidez deben ser incluidos en una sola matriz la cual es llamada matriz de rigidez total o global. Cuando varios miembros llegan a un mismo nodo y contribuyen a la rigidez de cierto grado de libertad, se usa el principio de superposición, es decir, simplemente se suman las rigideces. En la sección 4.4

4.8 Vector de carga total

Al igual que la matriz de rigidez total, las fuerzas que actúan en los nodos deben ensamblarse convenientemente. Cada miembro prismático tiene dos nodos y por lo tanto hace contribuciones a seis grados de libertad. Recuérdese que cada nodo tiene tres grados de libertad que pueden estar restringido o no estar restringidos. Generalmente se hace diferencia entre las fuerzas que actúan en algún punto del miembro y las fuerzas que actúan directamente en algún nodo, sin embargo en este trabajo se les consideran juntas en un solo vector de fuerza total F_T .

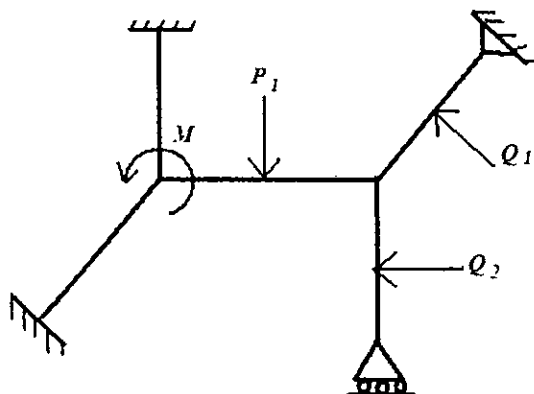
Debido a que varios miembros prismáticos pueden llegar a un solo nodo, el ensamble del vector de fuerza se hace siguiendo el principio de superposición, el cual implica una simple suma de fuerzas. En forma esquemática se ilustra a continuación

$$F_T = \begin{array}{c} \text{Miembro 1} \\ \left. \begin{array}{c} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_7 \\ F_8 \\ F_9 \end{array} \right\} \\ \text{Miembro 2} \end{array} \quad (4-48)$$

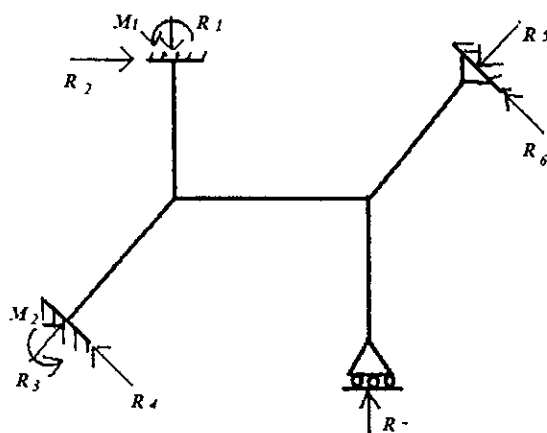
De la misma forma que la matriz de rigidez total los elementos del vector de carga total se deben reordenar para que los primeros grados de libertad sean los no restringidos. Este ordenamiento se puede y debe hacerse desde un inicio tal y como se procederá en esta tesis. El vector de carga total reordenado queda de la siguiente forma

$$F_T = \left(\begin{array}{c} F \\ F_R \end{array} \right) \begin{array}{l} \text{Grados de libertad no restringidos} \\ \text{Grados de libertad restringidos} \end{array} \quad (4-49)$$

Con la matriz de rigidez total K_T y con el vector de fuerza total F_T se pueden plantear fácilmente las ecuaciones básicas del método de rigidez. Además del vector F_T y la matriz K_T se tiene el vector de reacciones R , este vector de reacciones es incógnita y se puede calcular con facilidad, debido a que actúa sobre los grados de libertad que están restringidos y por lo tanto sus desplazamientos son cero. En caso de que los apoyos se desplacen se tiene que introducir este nuevo desplazamiento que no es cero. En la figura 35 se ilustra el origen del vector de carga F y el vector de reacciones R .



a) Fuerza que actúan sobre la estructura para formar el vector F_T



b) Reacciones que actúan sobre la estructura para formar el vector R

Figura 35. Reacciones para formar vectores F_T y R .

Con todo lo anterior se plantean las siguientes ecuaciones

$$K_T D_T + F_T = R_T \quad (4-50)$$

Esta ecuación involucra toda la estructura incluyendo grados de libertad restringidos y no restringidos. Desglosando la ecuación matricial (4-50), se tiene lo siguiente

$$\begin{pmatrix} K & K_{LR} \\ K_{RL} & K_{RR} \end{pmatrix} \begin{pmatrix} D \\ 0 \end{pmatrix} + \begin{pmatrix} F \\ F_R \end{pmatrix} = \begin{pmatrix} 0 \\ R \end{pmatrix} \quad (4-51)$$

En la expresión anterior se considera que los apoyos no se mueven y las fuerzas aplicadas en los nodos ya están incluidas en F_T . Las incógnitas son el vector D y el vector R , los cuales se calculan resolviendo las siguientes ecuaciones

$$\boxed{KD + F = 0} \quad (4-52)$$

$$\boxed{K_{RL}D + F_R = R} \quad (4-53)$$

En caso de que se considere movimientos de apoyos, se considera D_R que es el vector de desplazamientos de apoyos y la ecuación matricial se escribe de la siguiente forma

$$\left(\begin{array}{c|c} K & K_{LR} \\ \hline K_{RL} & K_{RR} \end{array} \right) \begin{pmatrix} D \\ D_R \end{pmatrix} + \begin{pmatrix} F \\ F_R \end{pmatrix} = \begin{pmatrix} 0 \\ R \end{pmatrix} \quad (4-54)$$

de donde salen dos ecuaciones matriciales

$$\boxed{KD + K_{LR}D_R + F = 0} \quad (4-55)$$

$$\boxed{K_{RL}D + K_{RR}D_R + F_R = R} \quad (4-56)$$

Estas son las ecuaciones básicas del método de rigidez con las que se encuentran las incógnitas D y R .

4.9 Análisis de un miembro individual

El método de rigidez permite calcular la fuerza axial, el corte y el momento en cada uno de los extremos de todos los miembros de la estructura. Debido a que ya se tiene la matriz de rigidez de cada elemento prismático, el vector de carga de cada elemento y ya se calcularon los desplazamientos de los nodos; el cálculo de las fuerzas internas es directo.

En la figura 36 se muestra un miembro prismático aislado sometido a un sistema de cargas cualquiera y con sus fuerzas axiales, fuerzas de corte y momento en ambos extremos.

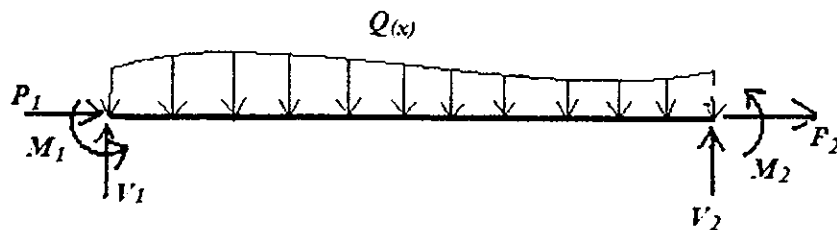


Figura 36. Miembro prismático sometido a un sistema de cargas.

En la figura 37 se muestra el miembro prismático sometido a un sistema de cargas cualquiera con los nodos restringidos y sus fuerzas de reacción de empotramiento, las cuales forman el vector F .

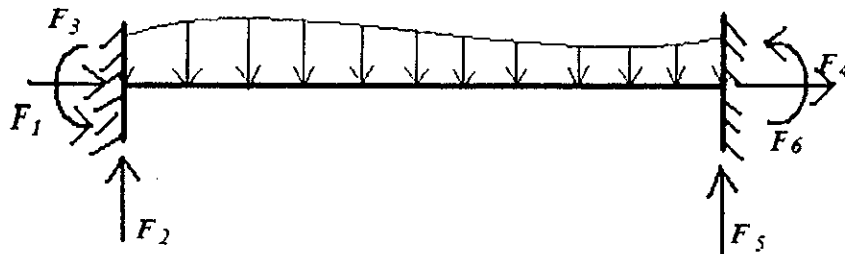


Figura 37. Miembro prismático con nodos restringidos.

El vector de desplazamiento D consta de sus seis componentes, donde D_1 y D_4 son los desplazamientos del nodo inicial y final respectivamente, D_2 y D_5 son los desplazamientos verticales del nodo inicial y final respectivamente, D_3 es el giro del nodo inicial y D_6 es el giro del nodo final. Este vector de desplazamiento es conocido de los cálculos efectuados anteriormente en el método de rigidez.

Hasta el momento se tienen tres vectores que son el vector de desplazamiento D que ya es conocido

$$D_M = \begin{pmatrix} D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \end{pmatrix}$$

el vector de fuerza F que también ya es conocido y se le escribe con sus seis componentes

$$F_M = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{pmatrix}$$

y el vector de reacción en los nodos R que por el momento es incógnita y se le escribe con sus seis componentes de la siguiente forma

$$R_M = \begin{pmatrix} P_1 \\ V_1 \\ M_1 \\ P_2 \\ V_2 \\ M_2 \end{pmatrix}$$

La matriz de rigidez de un elemento ya se ha calculado y se escribe a continuación otra vez.

$$K_M = \begin{pmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{pmatrix}$$

Aplicando exactamente las mismas ideas del método de rigidez a este elemento individual queda la siguiente ecuación

$$K_M D_M + F_M = R_M$$

y en forma directa se calcula R .

4.10 Marcos rígidos

En este trabajo de tesis se ilustra el método de rigidez analizando marcos rígidos. Un marco rígido es un arreglo de miembros prismáticos con uniones rígidas. Las uniones se consideran rígidas cuando son capaces de resistir momentos flexionantes. En comparación con los marcos rígidos están las armaduras en las cuales las uniones entre miembros prismáticos están articuladas y por lo tanto no resisten momentos flexionantes. Un

miembro prismático es un elemento estructural cuya longitud es mucho mayor al área transversal.

Debido a que las uniones en un marco rígido soportan momentos flexionantes se obtienen grados de indeterminación bastante altos. Es por eso que el cálculo de marcos rígidos es más complicado que el de las armaduras las cuales tienen un menor grado de indeterminación y se requiere para el análisis de marcos rígidos una mayor capacidad de computación.

Tabla 2

Comparación del método de rigidez con el método de flexibilidad.

Método de rigidez	Método de flexibilidad
Se formula en forma matricial	Se formula en forma matricial
No requiere criterio para aplicar el método.	Se requiere de criterio para aplicar el método.
Inicialmente calcula los desplazamientos	Inicialmente calcula las fuerzas.
Da un procedimiento sencillo y sistemático para calcular otras incógnitas como fuerzas y reacciones.	No hay un procedimiento sistemático para calcular desplazamientos y fuerzas internas en los miembros.
No se escogen fuerzas redundantes.	Es necesario elegir las fuerzas redundantes.
Se puede programar muy fácilmente en computadora.	Se puede programar en computadora.
Para marcos, el grado de indeterminación cinemática casi siempre es alto. Por lo tanto se tienen sistemas de ecuaciones con muchas incógnitas.	En marcos rígidos, las redundantes pueden ser relativamente pocas y se pueden plantear sistemas de ecuaciones relativamente pequeños.
Se deben restringir todos los nodos.	Se deben liberar los nodos con reacciones redundantes.

Los elementos estructurales en un marco rígido se consideran siempre elementos rectos y no arcos. Algunos ejemplos típicos de marcos rígidos se presentan en la figura 38.

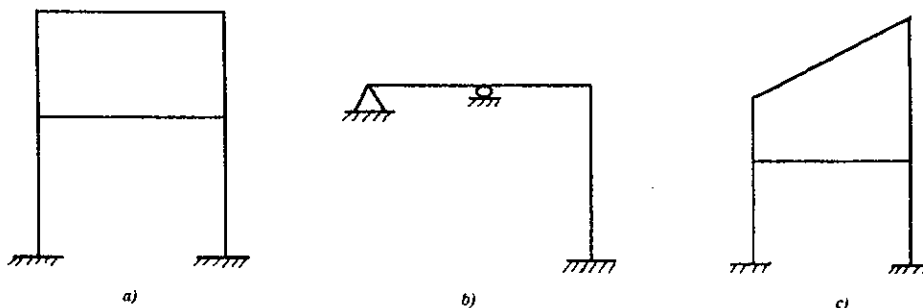


Figura 38. Tipos de apoyo.

El marco b de la figura 38 muestra los tres tipos de apoyos que puede tener un marco rígido. Así el apoyo del extremo izquierdo restringe los desplazamientos vertical y horizontal, y permite giros libremente. El apoyo de en medio sólo restringe el desplazamiento vertical permitiendo libremente el desplazamiento horizontal y el giro. Mientras que el apoyo inferior derecho restringe los dos desplazamientos y el giro.

Un marco rígido es estáticamente indeterminado cuando el número de reacciones desconocidas y fuerzas en los miembros excede a las ecuaciones de equilibrio disponibles. Para algunos marcos rígidos es relativamente fácil determinar su grado de indeterminación. Por ejemplo, hay seis incógnitas para las reacciones del marco rígido que se muestra en la figura 39a y hay tres ecuaciones de equilibrio estático, así que es fácil ver que este marco es estáticamente indeterminado en el tercer grado. En comparación considérese el marco de la figura 39b, este marco también tiene seis reacciones de incógnitas. Sin embargo, aún si las componentes de las reacciones son conocidas, la sección superior del marco sigue todavía estáticamente indeterminada, porque el número de las fuerzas de miembro excede a las ecuaciones disponibles. En este caso la sección superior es estáticamente indeterminada de tercer grado, así que podemos decir que el marco entero es estáticamente indeterminado con grado seis.

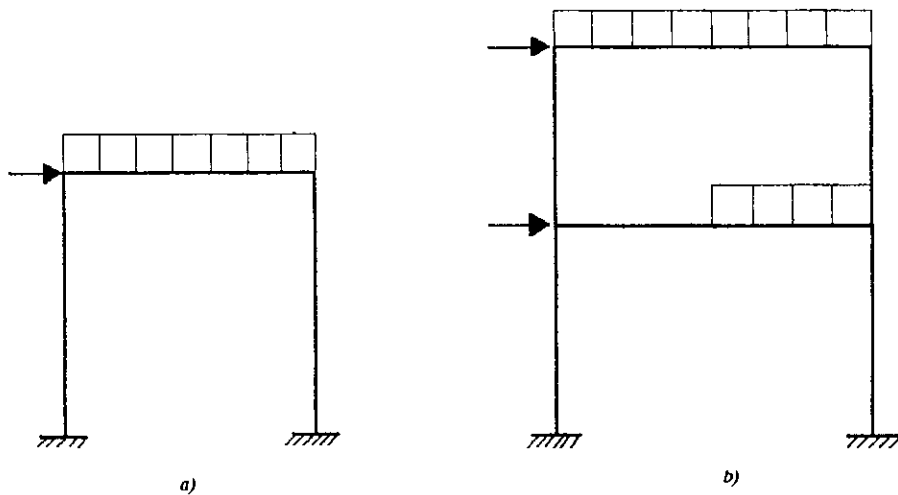


Figura 39. Grado de indeterminación.

De la discusión anterior se puede concluir que los marcos rígidos presentan fácilmente altos grados de indeterminación. Esto hace difícil el análisis estructural de los marcos rígidos. Sin embargo, con el uso de las computadoras y el método de rigidez se logra con mucha facilidad el cálculo de los marcos rígidos.

V. MATLAB

5.1 Introducción

MATLAB es un programa optimizado con el propósito de ejecutar cálculos científicos y de ingeniería. El nombre de MATLAB es la abreviatura de MATrix LABoratory. Este programa empezó como un programa diseñado para manipular matrices y ejecutar cálculos matriciales. De estos orígenes MATLAB se desarrolló, convirtiéndose en un sistema de computación flexible, capaz de resolver esencialmente cualquier problema técnico.

MATLAB ofrece el lenguaje de programación llamado también MATLAB y trae consigo una biblioteca extensa de funciones predefinidas, lo que hace que las tareas de programación técnica sean más fáciles y se lleven a cabo más eficientemente. En este trabajo de tesis se usa el lenguaje de programación MATLAB para analizar marcos rígidos. Particularmente en este capítulo se introduce este lenguaje de programación y se explican sus aspectos básicos.

MATLAB es un programa inmenso, con una variedad increíblemente rica de funciones. Aún la versión básica de MATLAB es mucho más rica que cualquier otro lenguaje de programación técnica como FORTRAN o C++. La versión más básica del MATLAB tiene más de mil funciones, esta versión no incluye los "toolkits" los cuales extienden la capacidad a mucho más funciones en varias especialidades. En este capítulo sólo se explican los aspectos esenciales para poder programar usando MATLAB y se introducen solamente las funciones que se necesitarán en este trabajo de tesis. No se pretende dar un tratamiento extensivo del MATLAB. En este trabajo de tesis será fácil convencerse del poder de MATLAB y la conveniencia de usarlo en cualquier trabajo de ingeniería como una herramienta necesaria.

Una de las características que identifica claramente a MATLAB es el uso de las matrices y los arreglos. Los arreglos son una generalización de las matrices a cualquier número de dimensiones. La flexibilidad que se permite en el uso de arreglos hace de la manipulación de datos y de la programación tareas más sencillas de efectuar. MATLAB puede convertir algoritmos desarrollados en el lenguaje de programación a código en C y C++. Si se usa MATLAB Excel Builder se pueden transformar algoritmos desarrollados en MATLAB a funciones de Excel y usar estas funciones desde Excel sin necesidad de tener MATLAB.

El MATLAB también hace posible graficar con bastante facilidad pues incluye varias funciones dedicadas a esta tarea. En este trabajo de tesis se hace uso del ploteo, facilitando la presentación de la información básica de los marcos rígidos. Este capítulo presenta las características esenciales de MATLAB que se usarán sin pretender una exposición exhaustiva de este programa.

5.2 Ventajas y desventajas de MATLAB

MATLAB tiene varias ventajas comparado con los lenguajes de programación convencionales. Algunas de sus ventajas y desventajas se expresan en esta sección.

Algunas de las ventajas son:

5.2.1 Fácil de usar. MATLAB es un lenguaje interpretado dicho de otra forma es un intérprete, como muchas versiones de Basic. Así como Basic, es muy fácil de usar. El programa se puede usar para calcular y evaluar expresiones escritas en la línea de comando, o se puede usar para ejecutar programas preescritos muy extensos. Los programas se pueden escribir y modificar fácilmente con el ambiente de desarrollo de programas directamente integrado. También se pueden corregir errores fácilmente usando el editor de MATLAB llamado "MATLAB editor/debugger".

Debido a que el lenguaje de MATLAB es tan fácil que resulta ser una herramienta ideal para la preescritura rápida de programas nuevos. Para hacer más fácil el uso de MATLAB se suministran muchas herramientas para el desarrollo de programas. Entre las principales están un editor integrado, documentación y manuales en línea, demostraciones extensas y un revisor llamado "workspace browser".

5.2.2 Funciones predefinidas. MATLAB viene completo con una extensa biblioteca de funciones predefinidas que dan soluciones a muchas tareas técnicas básicas. Por ejemplo, suponga que usted está escribiendo un programa que debe calcular la estadística asociada a un conjunto de datos. En la mayoría de lenguajes sería necesario escribir una subrutina para calcular la media, la desviación estándar, la mediana, etc; sin embargo éstas y otras funciones están incluidas en el lenguaje de MATLAB haciendo que el trabajo de programar sea mucho más fácil.

Particularmente, en este trabajo de tesis es necesario resolver sistemas de ecuaciones lineales y MATLAB provee varias funciones que resuelven sistemas de ecuaciones por ejemplo el comando *linsolve*. En los lenguajes de programación convencionales se tiene que programar haciendo más tardado y difícil la programación. Otras funciones usadas en este trabajo que facilitaron la tarea de programar fueron la transpuesta de una matriz, multiplicación de matrices, matriz inversa y formación de arreglos; todas ellas normalmente no incluidas en lenguajes de programación usuales.

5.2.3 Herramientas de propósito específico. Además de la biblioteca extensa de funciones, MATLAB ofrece muchas herramientas de propósito específico para resolver problemas complejos en áreas específicas. Por ejemplo, un usuario puede comprar herramientas estándar para resolver problemas en procesamiento de señal, sistemas de control, comunicaciones, procesamiento de imagen y redes neurales entre otras.

5.2.4 Independencia de ambiente. MATLAB es soportado por muchos sistemas de computación diferentes, dando así, una gran independencia de ambiente o de plataforma. Al momento de escribir esta tesis, el lenguaje MATLAB puede ser soportado en ambiente

Windows 95, Windows 98, Windows NT, Windows 2000 y Windows XP. Además muchas versiones diferentes de UNIX soportan MATLAB. Programas escritos en cualquier plataforma o ambiente corren en cualquier otro ambiente. Archivos de datos escritos en cualquier plataforma se pueden leer transparentemente en cualquier otro ambiente. Como resultado, programas escritos en MATLAB pueden emigrar a plataformas nuevas cuando el usuario necesite cambiar.

5.2.5 Dispositivos de ploteo e interfaces de graficación. A diferencia de otros lenguajes de programación, MATLAB tiene muchos comandos de ploteo y de imagen. Los ploteos y las imágenes se pueden ver en cualquier dispositivo de salida como impresoras o monitores. Esta capacidad hace que MATLAB sea una herramienta muy valiosa para la visualización de datos técnicos. Además MATLAB incluye herramientas que permiten al programador construir una interface gráfica del usuario (GUI) para su programa. Con esta capacidad puede diseñar programas sofisticados para el análisis de datos.

5.2.6 Compilador de MATLAB. La flexibilidad de MATLAB y la independencia de plataforma se logra por medio de la compilación de los programas de MATLAB en un código independiente llamado código p (p-code) y luego interpreta las instrucciones del código p. Este enfoque es similar al usado por el lenguaje Visual Basic de Microsoft. Desafortunadamente, los programas que resultan algunas veces son ejecutados lentamente debido a que el código de MATLAB es interpretado en lugar de ser compilado. Un compilador de MATLAB está disponible en forma separada. Éste puede compilar un programa de MATLAB en un programa ejecutable que corre más rápido que el código interpretado.

Algunas desventajas que presenta MATLAB se explican a continuación. MATLAB tiene dos desventajas. La primera es que es un intérprete, y debido a esto la ejecución de los programas es más lenta que los lenguajes compilados. Este problema puede ser mitigado escribiendo programas en MATLAB apropiadamente estructurados y usando el compilador de MATLAB.

La segunda desventaja es el costo. Una copia completa de MATLAB es de 5 a 10 veces más cara que un compilador convencional de Fortran o C. Esta desventaja puede ser disminuida si MATLAB se va a usar extensivamente dando mucho más beneficios que un compilador convencional. Existe una versión estudiantil de MATLAB que se puede conseguir a costo menor que la versión completa de MATLAB.

5.3 El ambiente de MATLAB

En cualquier programa de MATLAB la unidad fundamental de dato es el arreglo. Un arreglo es una colección de datos organizados en filas y columnas e identificados con un solo nombre. Datos individuales dentro de un arreglo pueden ser accesados por medio del nombre del arreglo seguido por subíndices que identifican la fila y la columna de un valor

particular. Aún los escalares se tratan como arreglos en MATLAB. Los escalares son arreglos de una fila y una columna.

Cuando se ejecuta MATLAB, se muestran en la pantalla varios tipos de ventanas que aceptan comandos o muestran información. Las tres ventanas más importantes en MATLAB son la de comando (Command Window), la de edición (Edit Window) y la de figura (Figure Window). En la ventana de comando, es donde se puede ingresar los comandos. La ventana de figura muestra ploteos y gráficos, y la ventana de edición permite crear y modificar programas.

Además de las tres ventanas mencionadas en el párrafo anterior MATLAB puede mostrar otras ventanas que dan información y ayuda muy útil al usuario. Estas otras ventanas le permiten al usuario examinar los valores de las variables definidas en la memoria.

Cuando se inicia MATLAB aparece una ventana frontal llamada MATLAB Desktop. La configuración típica de la ventana frontal del MATLAB Versión 7 se puede apreciar en la figura 40. Otra versiones de MATLAB tienen ventanas frontales con una configuración típica igual o parecida a la versión 7. La ventana frontal que se muestra en la figura 40 incluye la ventana de comandos (Command Window), la ventana de espacio de trabajo (Workspace Window o Workspace Browser) y la ventana de historia de comandos (Command History Window); además incluye la posibilidad de acceder directamente la ventana del directorio actual (Current Directory Window o Current Directory Browser).

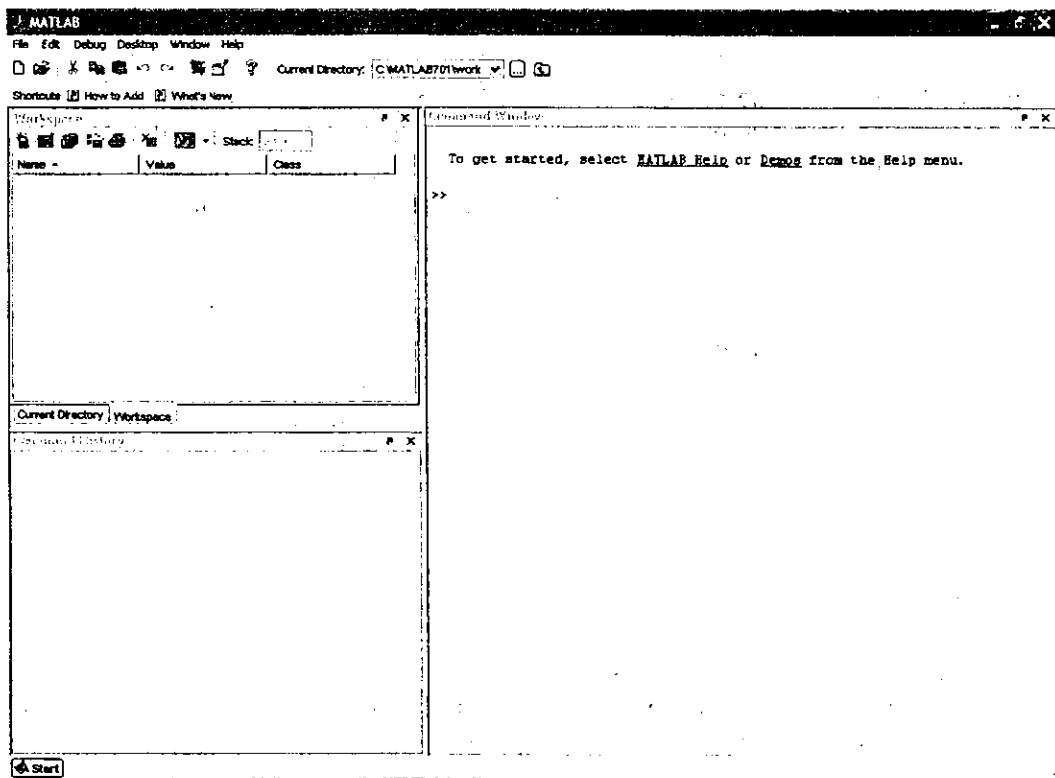


Figura 40. Ventana frontal de MATLAB.

Desde la ventana frontal fácilmente se puede acceder a la ventana de ayuda (Help Window), simplemente se acciona el botón con la marca ? que se encuentra en la barra de herramientas. Debido al tipo de trabajo que se desarrolla en la ventana de ayuda también se le llama "Help Browser". Ya se mencionaron las tres ventanas más importantes de MATLAB y además algunas ventanas que aparecen en la ventana frontal o que se pueden acceder directamente. Estas ventanas son consideradas como las herramientas básicas de MATLAB. A continuación se da un listado de las ventanas y luego una explicación concisa de cada una de ellas.

- Ventana de comandos (The Command Window)
- Ventana de edición (The Edit Window)
- Ventanas de figuras (The Figure Windows)
- Ventana de la historia de comandos (The Command History Window)
- Ventana de espacio de trabajo (The Workspace Browser)
- Ventana del directorio actual (The Current Directory Browser)
- Ventana de ayuda (The Help Browser)

5.3.1 La ventana de comandos. La ventana de comandos está en el lado derecho de la ventana frontal inicial (desktop). El usuario puede ingresar comandos interactivos a la par del símbolo >> el cual es conocido como "prompt command". Los comandos que ingrese el usuario serán ejecutados inmediatamente después de presionar la tecla enter. Como ejemplo se muestra a continuación un cálculo realizado en la ventana de comandos.

```
>> x1 = 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6
```

se presiona enter y aparece

```
>> x1 = 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6
```

```
x1 =  
2.4500
```

En caso de que la expresión a calcular sea demasiado larga para escribirla en una sola línea, se puede continuar en otra línea usando puntos suspensivos (...) así, la expresión anterior se podría escribir

```
>> x1 = 1 + 1/2 + 1/3 ...  
+ 1/4 + 1/5 + 1/6
```

Si se escribe solamente la expresión sin la variable *x1*, MATLAB guarda la respuesta en una variable llamada ans. Lo anterior se vería en la ventana de comandos de la siguiente forma.

$$? 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6}$$

$$\text{ans} = \\ 2.4500$$

La ventana de comandos sirve también para ejecutar programas. Solamente se escribe el nombre del programa a la par del símbolo `>>` en la ventana de comandos y se presiona enter para que corra el programa. Los programas normalmente están identificados como un archivo M.

Para limpiar la ventana de comandos se puede lograr por lo menos de dos formas. La primera es tecleando el comando `clc` a la par del símbolo `<<` y luego presionando enter. La segunda forma es presionando el botón derecho del ratón y aparece un menú y luego se activa con el ratón el comando “clear command window” como se muestra en la figura 41

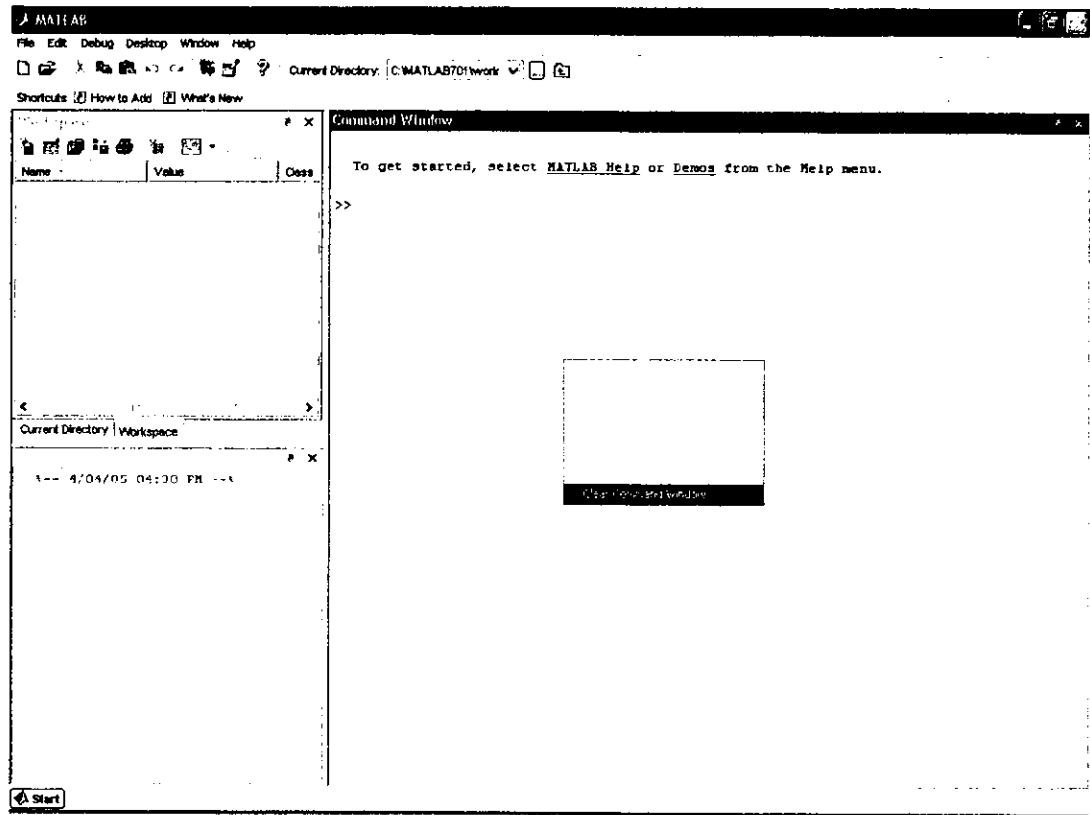


Figura 41. Menú para la ventana de comandos

5.3.2 La ventana de la historia de comandos. La ventana de historia de comandos (The Command History Window) aparece normalmente en la esquina inferior izquierda de la ventana frontal inicial. Esta ventana muestra una lista de los comandos que un usuario ha ingresado en la ventana de comandos. La lista de comandos previos puede contener ejecuciones previas del programa. Los comandos permanecen en la lista hasta que sean eliminados. Para reejecutar cualquier comando, use el ratón y simplemente presione dos

veces el botón izquierdo. Para eliminar uno, varios o todos los comandos de la ventana de la historia de comandos, presione el botón derecho del ratón y aparece un menú en la pantalla. En el menú se puede elegir la opción que se quiere como se muestra en la figura 42.

5.3.3 La ventana de edición. La ventana de edición (The Editor Window) se usa para crear programas nuevos o dicho de otra forma para crear archivos M nuevos. Esta ventana también se usa para editar y depurar programas y archivos M ya creados. La ventana de edición no aparece directamente en la ventana frontal inicial, pero puede ser accesada inmediatamente usando el ratón sobre el icono “New M-File” que se encuentra en la barra de herramientas de la ventana frontal inicial. De esta forma se crea un archivo M nuevo. En caso de que se quiere un programa o un archivo M ya existente en la ventana de edición, se usa el ratón sobre el ícono “Open File” que se encuentra en la barra de herramientas.

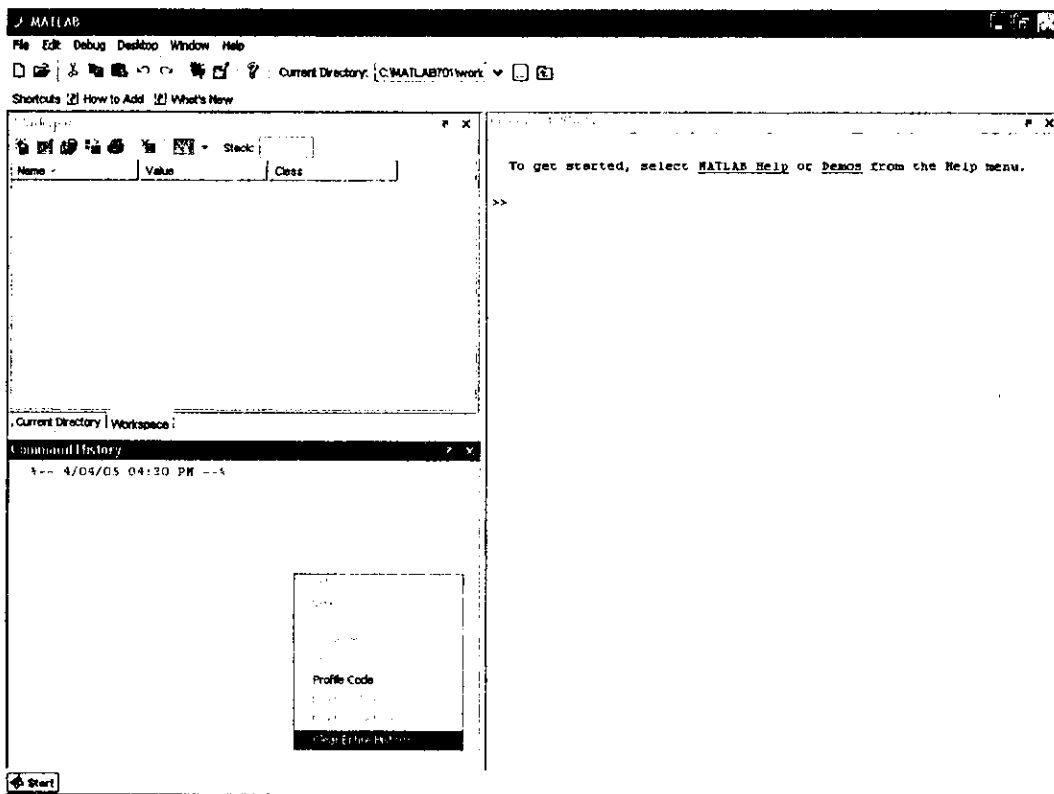


Figura 42. La ventana de la historia de comandos con su propio menú.

La ventana de edición es esencialmente un editor de texto para programar. Este editor expone las diferentes características del lenguaje en diferentes colores. La relación entre las diferentes partes de un programa y colores es la siguiente:

Comentarios	Verde
Variables y números	Negro
Caracteres y cadenas	Rojo
Instrucciones de programación	Azul

La ventana de edición también funciona como depurador de programas. Para guardar un programa o un archivo M se debe escoger apropiadamente su nombre. No se debe nombrar ningún programa de MATLAB con el mismo nombre que se le ha dado a las variables incluidas en el programa mismo. Tampoco se debe usar el nombre de funciones de MATLAB para nombrar programas o archivos M. Después que un programa ha sido apropiadamente nombrado y guardado, se puede ejecutar simplemente escribiendo su nombre en la ventana de comandos.

5.3.4 La ventana de espacio de trabajo. La ventana de espacio de trabajo (The Workspace Window) aparece en la parte superior izquierda de la ventana frontal inicial de MATLAB versión 7, con la opción de que aparezca en su lugar la ventana del directorio actual. Debido al tipo de trabajo que se puede realizar en la ventana de espacio de trabajo también se le llama “The Workspace Browser”.

El espacio de trabajo es la colección de todas las variables y arreglos que pueden ser usadas por MATLAB cuando un programa en particular o una función se está ejecutando. Todos los comandos, todos los programas y todos los archivos M ejecutados en la ventana de comandos comparten un espacio de trabajo común, así que todos comparten las mismas variables. Las funciones de MATLAB difieren de los programas o archivos M en el sentido de que tienen su propio espacio de trabajo.

Los contenidos del espacio de trabajo se pueden generar usando el comando *whos* en la ventana de comandos, pero más fácilmente, los contenidos del espacio de trabajo se pueden examinar en la ventana del espacio de trabajo (Workspace Browser). Esta ventana muestra gráficamente la misma información que el comando *whos* y la gráfica es dinámicamente actualizada cuando el contenido del espacio de trabajo cambia. La ventana del espacio de trabajo le permite al usuario cambiar el contenido de cualquier variable en el espacio de trabajo.

Una ventana de espacio de trabajo típica se muestra en la figura 43. Muestra la misma información que mostraría la ventana de comandos si se usara el comando *whos*. Si una de las variables que aparece en la ventana de espacio de trabajo se toca dos veces consecutivas con el ratón aparece el editor de arreglos (Array Editor). Este editor le permite al usuario modificar la información guardada en la variable, tal y como se puede apreciar en la figura 44.

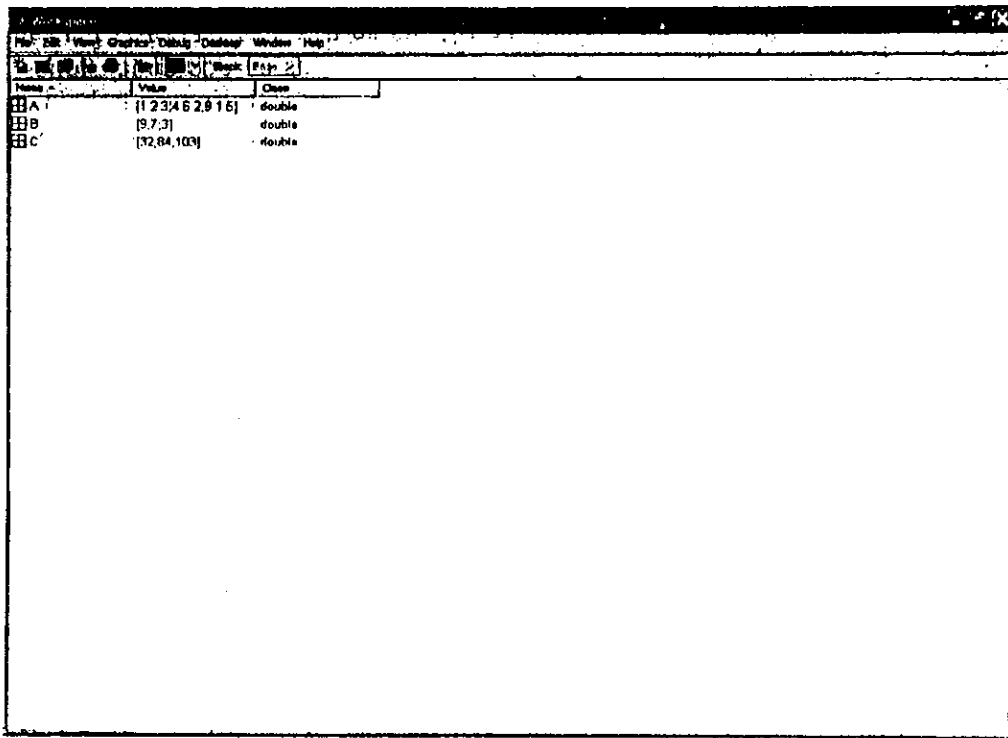


Figura 43. Ventana del espacio de trabajo

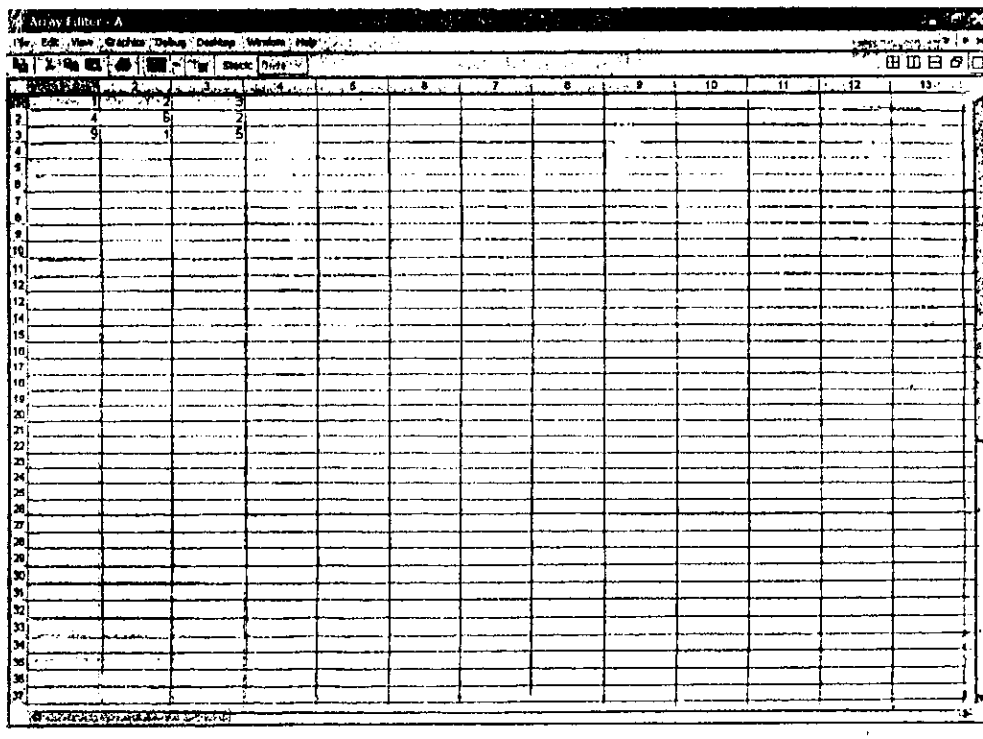


Figura 44. Ventana del editor de trabajo

5.3.5 La ventana de figuras. La ventana de figuras se usa en MATLAB para mostrar gráficas. Una figura puede ser una imagen, un ploteo de datos en dos dimensiones o tres dimensiones, o un elemento de la interface gráfica del usuario GUI. En este trabajo de tesis se usa la función de ploteo para graficar marcos los cuales aparecen en la ventana de figuras.

5.3.6 La ventana de ayuda. En MATLAB es muy importante saber cómo conseguir ayuda o información. MATLAB es tan extenso que es muy difícil conocer todos los comandos o todas las opciones que están disponibles por lo tanto el uso de ayuda se hace necesario.

Se puede acceder a la ventana de ayuda (Help Window) de varias formas. Una de ellas es accionar *Help* con el ratón. Este *Help* aparece en la ventana frontal inicial en la fila superior. Al accionar *Help* aparece un menú donde se puede elegir lo que se requiera. Otra forma de acceder a la ayuda es accionando con el ratón el botón con la marca ? que aparece en la barra de Herramientas. La tercera forma de acceder a la ayuda es escribiendo *helpdesk* o *helpwin* en la ventana de comandos. De las tres formas anteriores se accede al “Help Browser” el cual se muestra en la figura 45.

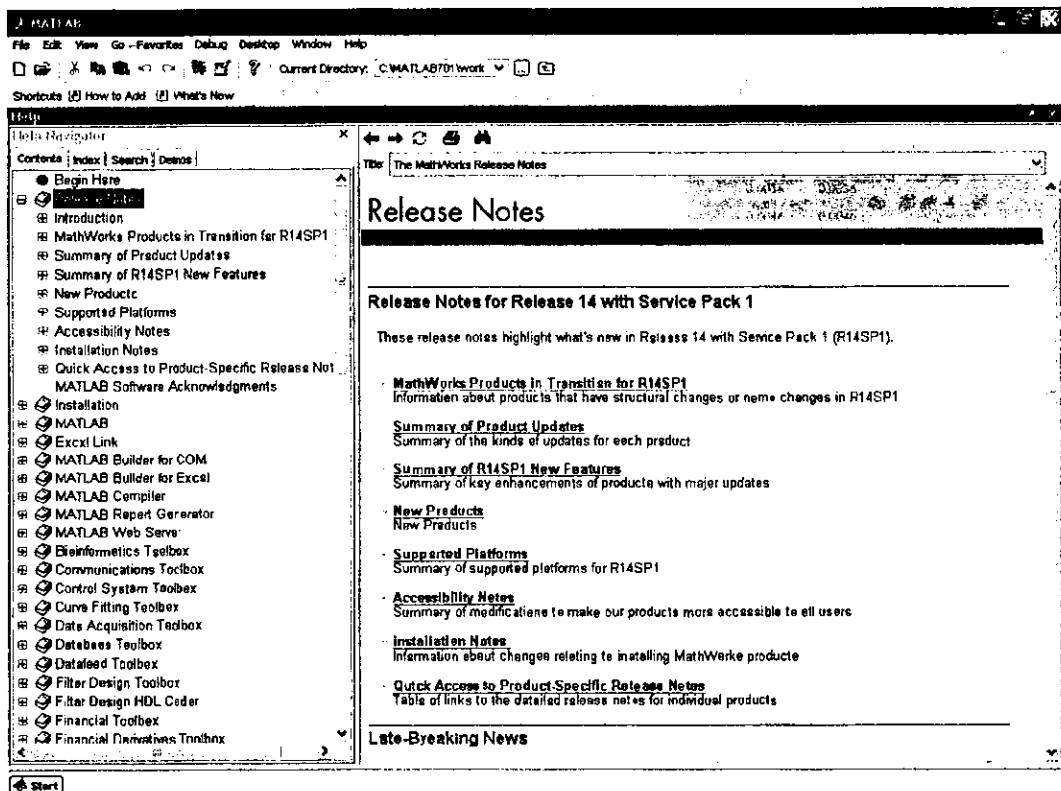


Figura 45. Help Browser

Además de las tres formas anteriores se puede teclear en la ventana de comandos *help* seguido del nombre de una función y MATLAB suministra la información pertinente

relacionada con la función. En caso de que se teclee solamente *help* entonces MATLAB desplegará una lista de todos los tópicos posibles de ayuda en la ventana de comandos.

También se puede acceder a la ayuda, escribiendo en la ventana de comandos el comando *lookfor*. Este comando tiene la ventaja que busca la información en forma apropiada y no exacta tal y como lo hace *help*. El comando *lookfor* debe teclearse seguido por el nombre apropiado de alguna función.

5.4 Variables en MATLAB

La unidad fundamental de datos en MATLAB es el arreglo. Un arreglo es una colección de datos organizados en filas y columnas. En la versión 7 de MATLAB se pueden considerar arreglos multidimensionales. En el programa de esta tesis se usan arreglos multidimensionales.

Los escalares son arreglos de una fila por una columna. Los vectores fila son arreglos de una fila por varias columnas. Los vectores columna son arreglos de varias filas por una columna y las matrices son arreglos de dos o más dimensiones.

Una variable en MATLAB es una región de memoria que contiene un arreglo. Esta variable es conocida o identificada por medio de un nombre especificado por el usuario. El contenido del arreglo se puede usar en cualquier momento o se puede modificar incluyendo su nombre.

Los nombres de las variables en MATLAB deben empezar con una letra, seguida por cualquier combinación de letras, números y el símbolo `_`.

Otros símbolos o caracteres disponible en el teclado como `+`, `*`, `^`, `?`, `%`, etc. incluyendo el espacio en blanco no pueden usarse para nombrar variables. En esta tesis el caracter y símbolo se consideran como sinónimos.

En MATLAB solamente los primeros 31 caracteres son significativos. En caso de que se usen más de 31 caracteres, del caracter en la posición 32 en adelante son ignorados. Si dos variables se declaran con nombres que solo difieren en el 32avo caracter, MATLAB las tratará como la misma variable. Es aconsejable nombrar variables con nombres cuya longitud sea menor de 31 caracteres.

MATLAB hace diferencia entre mayúsculas y minúsculas, lo que significa que mayúsculas y minúsculas son símbolos distintos. Por ejemplo la variable `Z` y la variable `z` son distintas.

Los dos tipos de variable más comunes en MATLAB son *double* y *char*. Las variables de tipo *double* consisten en escalares o arreglos de números de punto flotante de doble precisión con 64 bits. Este tipo de variable puede manejar, números reales,

5.5 Creación de variables en MATLAB

Las tres formas más comunes de crear variables en MATLAB son:

- Asignar datos a la variable por medio de una proposición de asignación.
- Ingreso de datos a la variable desde el teclado.
- Lectura de datos desde un archivo.

Estas formas de inicializar y crear variables en MATLAB serán discutidas en esta sección.

La forma más simple de crear una variable es asignarle un valor por medio de una proposición de asignación. En términos generales sería de la siguiente forma:

$$\text{Var} = \text{expresión}$$

Donde *Var* es el nombre de la variable y *expresión* es la proposición que asigna un valor o varios valores a la variable *Var*. La expresión puede ser una constante escalar o una matriz o alguna otra variable ya creada o cualquier combinación de lo anterior usando operaciones matemáticas como la suma, resta, multiplicación, división, potenciación, etc. Algunos ejemplos se muestran a continuación

```
> var2 = var/5;
>> Mat = [1,2,3];
>> x = i;
>> y = var + x;
```

En los ejemplos anteriores se crean cinco variables cuyos nombres son *var*, *var2*, *Mat*, *x* e *y*. Estas variables son del tipo *double*. Así se tiene que a *var* se le asigna el escalar 40 e instantáneamente queda creada la variable cuyo nombre es *var*. Luego a *var2* se le asigna el valor de *var/5* e inmediatamente queda creada la variable *var2*. A estas variables ya creadas se les puede cambiar su valor en cualquier momento. A la variable *Mat* se le asigna un vector fila (1,2,3). Como puede apreciarse crear variables en MATLAB es muy sencillo y directo.

Las variables pueden ser creadas asignándole matrices o arreglos multidimensionales. Las expresiones para asignar matrices requieren de un poco de cuidado por lo que se le dedicará una sección especial más adelante.

La segunda forma de crear variables es por medio de datos que el usuario ingresa directamente desde el teclado. Para esto se usa la función *input* la cual puede desplegar una cadena en la ventana de comandos y espera a que el usuario teclee los datos. Dos ejemplos se dan a continuación para ilustrar el uso del *input*.

```
y = input('Ingrese el valor de y: ')
MN = input('Ingrese la matriz de nodos: ')

```

Cuando el *input* es ejecutado MATLAB imprime en la ventana de comandos la cadena *Ingrese el valor de y:* y espera a que el usuario teclee el valor. Luego imprime la cadena *Ingrese la matriz de nodos:* y espera a que el usuario ingrese la matriz. Para ingresar una matriz se deben usar corchetes como se explicará más adelante. En caso de que no se teclee ningún número y se presiona ENTER entonces se crea una matriz vacía y guardada en la variable.

Si se quiere crear una variable del tipo *char* hay que agregar a la función *input* un segundo argumento denotado por 's' de la siguiente forma

$$x = \text{input}('Ingrese datos:', 's');$$

De esta forma *x* será una cadena de caracteres por lo tanto *x* será variable del tipo *char*.

La tercera forma de crear variables es leyéndolas de algún archivo. Si se tiene un archivo donde se guardan ciertas variables se pueden pasar y crear en MATLAB usando la función *load*.

El comando *load* lee las variables del archivo y las carga automáticamente en el espacio de trabajo de MATLAB. La forma básica del uso del comando *load* es la siguiente:

$$\text{load nombre del archivo}$$

donde nombre del archivo es el nombre del archivo que se va a cargar.

Cuando se crean variables en MATLAB para un uso temporal, con la posibilidad de ser usadas en un futuro es aconsejable guardarlas en un archivo. Para lograr lo anterior se usa la función *save* de la siguiente forma

$$\text{save nombre archivo var1 var2 var3}$$

de esta forma las variables *var1*, *var2* y *var3* se guardan en un archivo con el nombre, nombre archivo. Las variables anteriores pueden ser creadas otra vez en MATLAB usando el comando *load*.

5.6 Arreglos

En MATLAB la unidad básica de datos es el arreglo. Toda la información de cualquier variable está dispuesta en un arreglo. Se puede conceptualizar al arreglo como una generalización del concepto de matriz. Así como en una matriz en un arreglo la información se dispone en filas y columnas.

Algunos arreglos reciben nombre especial tal y como se establece a continuación:

- Escalar es un arreglo de una fila por una columna.
- Un vector columna es un arreglo de n filas por una columna y se puede visualizar como se muestra en la figura 46.

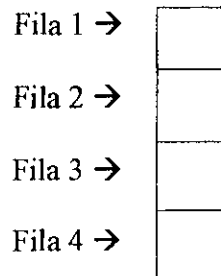


Figura 46. Vector columna como ejemplo de un arreglo unidimensional.

- Un vector fila es un arreglo de una fila por n columnas y se puede visualizar tal y como se muestra en la figura 47.

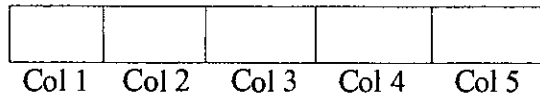


Figura 47. Vector fila como ejemplo de un arreglo unidimensional.

- Una matriz es un arreglo de n filas por n columnas y se puede visualizar tal y como se muestra en la figura 48.

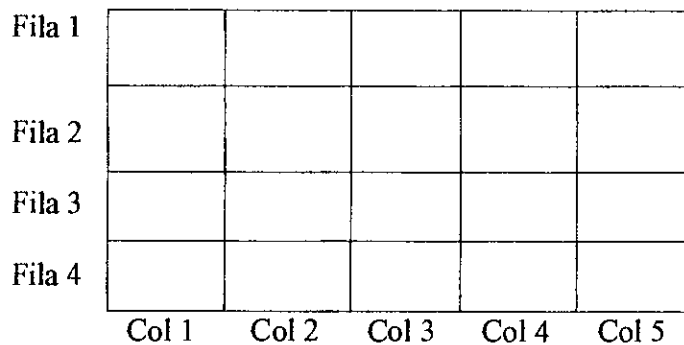


Figura 48. Matriz como ejemplo de un arreglo bidimensional.

- Un arreglo tridimensional es un arreglo de 1 matrices donde cada matriz es un arreglo de n filas por m columnas y se puede visualizar tal y como se muestra en la figura 49.

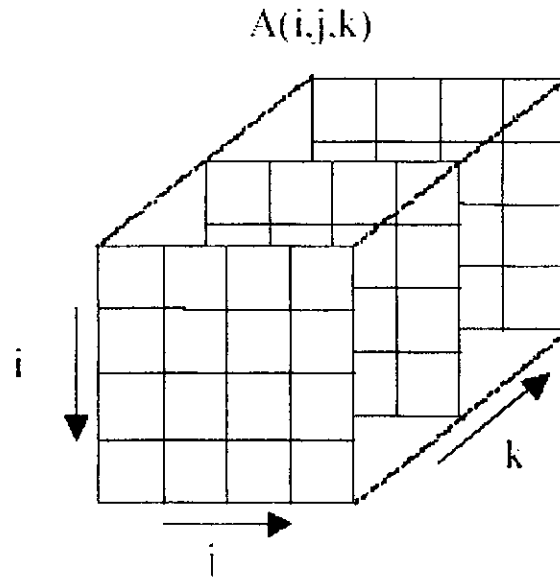


Figura 49. $A(i,j,k)$ es un arreglo tridimensional.

A los arreglos de tres o más dimensiones, frecuentemente, se les llaman hipermatrices.

El concepto de arreglo permite una definición precisa que se da a continuación.

Definición de arreglo

Un arreglo A es una función que a cada n -ada de índices (i_1, i_2, \dots, i_n) le asigna un valor numérico o una cadena de caracteres denotado por $A(i_1, i_2, \dots, i_n)$.

Los índices i son variables que toman valores enteros que van desde 1 hasta un valor máximo de n .

El objetivo de esta sección es explicar la forma de inicializar o crear arreglos. La forma básica de crear arreglos unidimensionales es usar los corchetes tal y como se explica a continuación.

Para crear vectores fila se abre corchetes y se teclean los valores de cada columna uno tras otro dejando un espacio entre ellos y se cierran corchetes. Para ilustrar esto, en la ventana de comando se teclea [1 2 3] se presiona enter y se obtiene lo siguiente.

```
>> [1 2 3]
```

```
ans =
```

```
1 2 3
```

También se pueden poner comas y se obtiene lo mismo.

```
>> [1,2,3]
```

```
ans =
```

```
1 2 3
```

Para crear vectores columna se deben colocar puntos y comas como se muestra en el siguiente ejemplo.

```
>> [1;2;3]
```

```
ans =
```

```
1
2
3
```

Para crear matrices se ponen espacios o comas y para cambiar de fila se teclea punto y coma, tal y como se muestra en el siguiente ejemplo.

```
>> [1,2,3;4,5,6;7,8,9]
```

```
ans =
```

```
1 2 3
4 5 6
7 8 9
```

Si se quiere nombrar la variable de una forma especial simplemente se escribe el nombre que se desee.

```
> A=[1 2 3;4 5 6;7 8 9]
```

```
A =
```

```
1 2 3
4 5 6
7 8 9
```

Para inicializar un arreglo de tres o más dimensiones se debe asignar una matriz por cada valor del tercer índice. Además se deben usar dos puntos para el lugar de los primeros índices, como se ilustra en el siguiente ejemplo.

```
>> A(:,:,1) = [1,2,3;4,5,6]
```

```
A =
```

```
 1  2  3
 4  5  6
```

```
>> A(:,:,2) = [7,8,9;10,11,12]
```

```
A(:,:,1) =
```

```
 1  2  3
 4  5  6
```

```
A(:,:,2) =
```

```
 7  8  9
10 11 12
```

```
>> A(:,:,3) = [13,14,15;16,17,18]
```

```
A(:,:,1) =
```

```
 1  2  3
 4  5  6
```

```
A(:,:,2) =
```

```
 7  8  9
10 11 12
```

```
A(:,:,3) =
```

```
13 14 15
16 17 18
```

Del ejemplo anterior se puede escribir que $A(2,1,3) = 16$, $A(2,2,2) = 11$ y así sucesivamente. Obsérvese que A es una función que a una terna ordenada de índices (i,j,k) le asigna el valor numérico $A(i,j,k)$.

Una forma bastante usada para crear vectores filas es por medio del operador dos puntos `:`. La manera de hacerlo en términos generales es

`a = valor inicial : incremento : valor final.`

Si en el lugar del incremento no se escribe nada MATLAB toma incrementos de una unidad entera. Los siguientes ejemplos ilustran el uso del operador dos puntos `:`.

```
>> a = 3:2:9
```

```
a =
```

```
3 5 7 9
```

```
>> b = 3:7
```

```
b =
```

```
3 4 5 6 7
```

En MATLAB existen varias funciones que generan matrices de forma inmediata. Por ejemplo `zeros(n,m)` genera una matriz de n filas por m columnas llena de ceros. Otras funciones para crear fácilmente matrices se resumen en la tabla 4.

Tabla 4
Funciones que crean matrices

Función	Propósito
zeros (n)	Genera una matriz nxn de ceros.
zeros (n,m)	Genera una matriz nxm de ceros.
ones (n)	Genera una matriz nxn de unos.
ones (n,m)	Genera una matriz nxm de unos.
eye (n)	Genera la matriz identidad nxn.
size (A)	Genera el vector fila siguiente: (número de filas, número de columnas)

Dos operadores importantes sobre matrices y que el MATLAB los ofrece, son la inversa de una matriz y la transpuesta de una matriz.

El operador que calcula la transpuesta se puede escribir de dos formas

Transpose (A)

o

A'

El operador que calcula la inversa de una matriz cuadrada A se escribe de la siguiente forma

inv (A)

Si se usa el operador transpuesta y el operador dos puntos, :, se pueden construir fácilmente vectores columna, por ejemplo.

```
>> b = [1:5]'
```

```
b =
```

```
1
2
3
4
5
```

5.7 Operaciones aritméticas

Las variables se pueden sumar, restar, multiplicar, dividir o exponenciar. Estas operaciones son las operaciones binarias básicas de la aritmética. Cuando se trata de escalares, estas operaciones se usan en MATLAB como se indica en la tabla 5 y siguen la jerarquía que se indica en la tabla 6.

Los arreglos también se pueden sumar, restar, multiplicar, dividir y exponenciar pero con ciertas restricciones. La multiplicación, la división y la exponenciación se pueden definir de dos formas distintas, una es siguiendo las reglas de álgebra matricial y la otra es elemento con elemento. Recuérdese que la multiplicación de matrices como se define en el álgebra matricial es de la siguiente forma

$$C_{(i,j)} = \sum_{k=1}^n A_{(i,k)} B_{(k,j)}$$

Tabla 5
Operaciones aritméticas entre escalares a y b.

OPERACIÓN	FORMA ALGEBRAICA	USO EN MATLAB
Suma	$a + b$	$A + b$
Resta	$a - b$	$a - b$
Multiplicación	$a \times b$	$A * b$
División	a / b	a / b
Exponencial	a^b	a^b

Tabla 6
Jerarquía de las operaciones aritméticas.

JERARQUÍA	OPERACIÓN
1	Los contenidos de todos los paréntesis son evaluados, empezando por el más interno.
2	Todas las exponenciaciones son calculadas, de izquierda a derecha.
3	Todas las multiplicaciones y divisiones son efectuadas, de izquierda a derecha.
4	Todas las sumas y las restas son efectuadas, de izquierda a derechas.

a la otra forma de multiplicar se le llamará multiplicación de arreglos. Se dice que dos arreglos tienen la misma estructura si tienen el mismo número de dimensión y cada dimensión tiene el mismo tamaño. Así, por ejemplo, dos matrices tienen la misma estructura si tienen el mismo número de filas y el mismo número de columnas.

Con las explicaciones anteriores se puede leer la tabla 7 donde se exponen las operaciones aritméticas entre matrices y arreglos.

Tabla 7
Operaciones entre arreglos y matrices.

OPERACIÓN	NOTACIÓN DE MATLAB	RESTRICCIÓN
Suma de arreglos	$A + B$	A y B tienen la misma estructura.
Resta de arreglos	$A - B$	A y B tienen la misma estructura.
Multiplicación de arreglos	$A * B$	A y B tienen la misma estructura o uno de los dos es un escalar.
División de arreglos por la derecha	$A ./ B$	A y B tienen la misma estructura o uno de los dos es un escalar.
División de arreglos por la izquierda	$A .\ B$	A y B tienen la misma estructura o uno de los dos es un escalar.
Exponenciación de arreglos	$A.^B$	A y B tienen la misma estructura o uno de los dos es un escalar.
Multiplicación de matrices	$A * B$	El número de columnas de A debe ser igual al número de filas de B.
División matricial derecha	A / B	B debe tener inversa y el número de filas de B igual al de columnas de A.

Las operaciones entre arreglos necesitan una explicación extra, la cual está resumida en la tabla 8. La forma de diferenciar operaciones entre arreglos y las operaciones normales entre matrices es por medio del punto previo al símbolo de la operación.

La suma matricial y la suma de arreglos se definen de la misma forma, la cual se hace elemento con elemento. La resta al igual que la suma se efectúa de la misma forma considerando matrices o arreglos en general. La multiplicación, la división y la exponenciación se definen de forma diferente para matrices que para arreglos en general.

Para dar las definiciones de operaciones entre arreglos, en la tabla 8 se considera a A como un arreglo tridimensional, a B como un arreglo tridimensional de la misma estructura que A y las variables a y b como escalares. Se consideran arreglos tridimensionales para

ilustrar mejor la definición. Para arreglos de otras dimensiones sería exactamente lo mismo.

Tabla 8
Definición de las operaciones entre arreglos.

OPERACIÓN	DEFINICIÓN
$C = A * B$	$C_{(i,j,k)} = A_{(i,j,k)} * B_{(i,j,k)}$
$C = a * B$	$C_{(i,j,k)} = a * B_{(i,j,k)}$
$C = A * b$	$C_{(i,j,k)} = A_{(i,j,k)} * b$
$C = A / B$	$C_{(i,j,k)} = A_{(i,j,k)} / B_{(i,j,k)}$
$C = a / B$	$C_{(i,j,k)} = a / B_{(i,j,k)}$
$C = A / b$	$C_{(i,j,k)} = A_{(i,j,k)} / b$
$C = A \setminus B$	$C_{(i,j,k)} = B_{(i,j,k)} / A_{(i,j,k)}$
$C = a \setminus B$	$C_{(i,j,k)} = B_{(i,j,k)} / a$
$C = A \setminus b$	$C_{(i,j,k)} = b / A_{(i,j,k)}$
$C = A \wedge B$	$C_{(i,j,k)} = A_{(i,j,k)} \wedge B_{(i,j,k)}$
$C = a \wedge B$	$C_{(i,j,k)} = a \wedge B_{(i,j,k)}$
$C = A \wedge b$	$C_{(i,j,k)} = A_{(i,j,k)} \wedge b$

La solución de sistemas de ecuaciones lineales en MATLAB se puede efectuar de varias formas. La forma matricial en la que se escribe un sistema de ecuaciones lineales es la siguiente

$$Ax = b$$

donde A es una matriz no singular de n filas por n columnas, b es un vector columna de n filas y x es el vector columna de n incógnitas. La solución de este sistema de ecuaciones se puede dar fácilmente en MATLAB, simplemente se usan las operaciones entre matrices de la siguiente manera

$$x = A \setminus b.$$

En este caso lo que se está haciendo es multiplicar la matriz inversa de A por el vector columna b. Es decir que x se podría escribir de la siguiente manera

$$x = \text{inv}(A) * b.$$

Sin embargo, MATLAB ofrece varias funciones para resolver sistemas de ecuaciones de una manera más conveniente y la función que se usará en esta tesis es la función *linsolve*, así que para resolver el sistema de ecuaciones $Ax = b$ se puede escribir la siguiente instrucción

$$x = \text{linsolve}(A,b).$$

5.8 Presentación de resultados

En MATLAB hay varias formas de presentar resultados. La forma más simple es omitir el punto y coma al final del cálculo y MATLAB presentará el resultado inmediatamente. Cuando se tiene un programa que realiza muchos cálculos y por ende se tienen muchos resultados, puede quedar muy confuso el despliegue de resultados si se omite el punto y coma. El programa de esta tesis está en el caso de que se tienen muchos cálculos por lo que es aconsejable usar el punto y coma al final de todas las líneas que contengan algún cálculo y el despliegue de resultados se realiza usando otras funciones.

Otra forma de presentar resultados es por medio de la función *disp*. La función *disp* acepta arreglos de argumentos y muestra los valores del arreglo en la ventana de comandos. Si el arreglo es del tipo *char*, entonces la cadena de caracteres contenida en el arreglo es mostrada.

A menudo la función *disp* se combina con las funciones *num2str* y *int2str*. La función *num2str* convierte un número a cadena de caracteres y la función *int2str* convierte un entero a cadena de caracteres. Ambas funciones son útiles para mostrar mensajes en la

ventana de comandos. Ejemplos del uso de la función *disp* con *num2str* se dan a continuación.

```
A=input('Ingrese la matriz A');
for ii=1:3
    titulo=['          Los Desplazamientos y el Giro del NODO
',num2str(ii),' son:'];
    a=['          Horizontal = ',num2str(A(ii,1)),';          Vertical =
',num2str(A(ii,2)),';          Giro = ',num2str(A(ii,3)),' rad'];
        disp(titulo)
        disp(' ')
        disp(a)
        disp(' ')
        disp(' ')
end
end
```

Ingresando la siguiente matriz A:

A =

```
7.8690  9.6740  0.0900
1.3450 34.6700  1.5670
-23.8970 987.0000  0.0001
```

la función *disp* da el siguiente resultado

Los Desplazamientos y el Giro del NODO 1 son:

Horizontal = 7.869; Vertical = 9.674; Giro = 0.08997 rad

Los Desplazamientos y el Giro del NODO 2 son:

Horizontal = 1.345; Vertical = 34.67; Giro = 1.567 rad

Los Desplazamientos y el Giro del NODO 3 son:

Horizontal = -23.897; Vertical = 987; Giro = 5e-005 rad

Tal y como se puede apreciar en el ejemplo la función *disp* se puede imprimir cualquier cadena de caracteres que uno desee. El espacio en blanco es considerado como un carácter o símbolo por lo tanto la presentación se puede hacer de muchas maneras. La función *num2str* permite pasar fácilmente variables de tipo *double* a variables tipo *char* e imprimirlas usando la función *disp*.

5.9 Programación en MATLAB

MATLAB es también un compilador y puede usarse para programar como se haría usando FORTRAN o C++. En la versión 7 de MATLAB el compilador funciona como un intérprete y no es necesario declarar variables. Sin embargo para un mejor control se aconseja definir las variables dentro del programa.

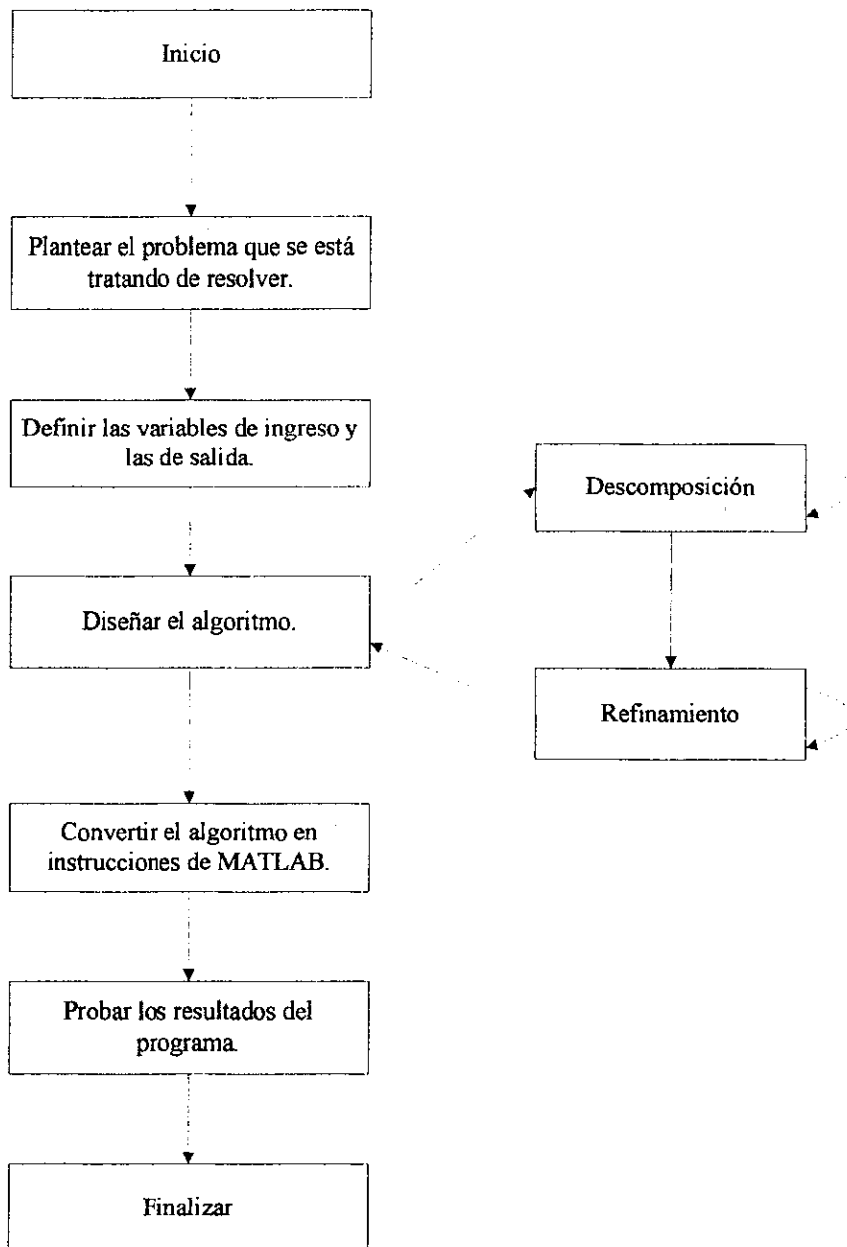


Figura 50. Procedimiento para programar.

Las instrucciones en cualquier compilador y específicamente en MATLAB se pueden dividir en dos grandes categorías, las *ramificaciones* y los *ciclos*. Con el uso de las instrucciones en MATLAB ya sean del tipo de *ramificaciones* o de *ciclos* los programas fácilmente se vuelven complejos, por lo tanto, es necesario seguir buenas costumbres en la programación y se sugiere se siga el siguiente procedimiento que en términos generales se muestra en la figura 50.

Previo a explicar las ramificaciones se definen los operadores relacionales y los operadores lógicos que son usados en las ramificaciones.

Los operadores relacionales son operadores con dos operandos numéricos o dos operandos de cadenas que dan como resultado verdadero o falso, dependiendo de la relación entre los dos operandos. En MATLAB verdadero es 1 y falso es 0. La forma general de un operador relacional es la siguiente:

$$a_1 \text{ op } a_2$$

donde a_1 y a_2 son expresiones aritméticas, variables o cadenas de caracteres y op es cualquiera de los operadores relacionados que ofrece MATLAB los cuales están descritos en la tabla 9.

Tabla 9
Operadores relacionales

OPERADOR	DESCRIPCIÓN
==	Igual a
~=	No igual a
>	Mayor a
>=	Mayor o igual a
<	Menor a
<=	Menor o igual a

Si la relación entre a_1 y a_2 es verdadera, entonces la operación del operador relacional da igual a 1, en caso de que no sea verdadera da igual a 0, así por ejemplo

```
>> 3 < 4
ans =
    1
>> 3 <= 4
ans =
    1
>> 3 == 4
ans =
    0
```

También se puede usar algunos de los operadores relacionales con variables tipo *char* como se ejemplifica a continuación.

```
>> 'cadena' == 'cadena'
ans =
    1    1    1    1    1    1
>> 'x' == 'y'
ans =
    0
>> 'x' ~= 'y'
ans =
    1
```

Se debe tener mucho cuidado en no confundir el operador relacional `==` con el operador de asignación de valores `=`.

El operador `==` da 1 cuando los dos valores que se están comparando son iguales, y da 0 cuando los dos valores que están siendo comparados son diferentes. Similarmente, el operador `~=` da 0 cuando los dos valores que están siendo comparados son iguales y da 1 cuando son diferentes. Estos operadores generalmente son seguros de usar cuando se comparan cadenas, pero pueden dar sorpresas desagradables cuando dos valores numéricos son comparados. Debido al redondeo de las cifras, dos números teóricamente iguales podrían diferir levemente.

Para ilustrar lo anterior considérese el siguiente ejemplo

```
>> a = 0;
>> b = sin(pi);
```

Teóricamente a y b son iguales y `a == b` debería de dar 1. Sin embargo los resultados de los cálculos en MATLAB dan lo siguiente

```
>> a == b
ans =
```

0

Lo anterior es debido a que b no es exactamente igual a cero sino que es casi cero por los cálculos que se efectuaron en el MATLAB

```
>> b
b =
1.2246e-016.
```

Estos problemas de redondeo deben evitarse en cualquier programa de MATLAB que use operadores relacionales.

Junto con los operadores relacionales se pueden usar los operadores lógicos los cuales son operadores que dan resultados lógicos entre los operandos. En MATLAB hay un operador lógico unario llamado operador NOT. También hay tres operadores lógicos binarios llamados AND, OR y OR exclusivo.

La forma general para el operador NOT sobre un operando L es

$$\sim L$$

y la forma general para un operador binario entre los operandos L_1 y L_2 es

$$L_1 \text{ op } L_2$$

La tabla 10 muestra los símbolos que se usan en MATLAB para los operadores lógicos.

Tabla 10
Operadores lógicos.

OPERADOR LÓGICO EN MATLAB	OPERACIÓN
&	AND
	OR
xor	OR exclusivo
~	NOT

La tabla 11 muestra los valores que se toman utilizando los operadores lógicos al ingresar los operandos L_1 y L_2 .

Tabla 11
Tabla de verdad para los operadores lógicos

L_1	L_2	$L_1 \& L_2$	$L_1 L_2$	$\text{xor}(L_1, L_2)$	$\sim L_1$
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

En la jerarquía de las operaciones, los operadores lógicos se valúan después de todas las operaciones aritméticas y todas las operaciones relacionales. El orden en el cual se evalúan los operadores es la siguiente:

- Primero, todos los operadores aritméticos se evalúan de primero siguiendo la jerarquía descrita anteriormente.
- Segundo, todos los operadores relacionales se evalúan trabajando de izquierda a derecha.
- Tercero, todos los operadores \sim son evaluados de izquierda a derecha.
- Cuarto, todos los operadores $\&$ son evaluados de izquierda a derecha.
- Quinto, todos los operadores $|$ son evaluados de izquierda a derecha.

Al igual que con las operaciones aritméticas, los paréntesis se pueden usar para cambiar el orden de la evaluación en operaciones lógicas.

Las ramificaciones en MATLAB son proposiciones que permiten seleccionar y ejecutar partes específicas de un código de programación mientras que se ignoran otras partes del código del programa. Las principales ramificaciones que ofrece MATLAB son el *if* y el *switch*. El *if* tiene la siguiente forma general:

```

if expresion_de_control_1
    Instruccion 1
    Instruccion 2
    ...
elseif expresion_de_control_2
    Instruccion 1
    Instruccion 2
    ...
else
    Instruccion 1
    Instruccion 2

```

```

...
end

```

donde la expresión de control controla el *if*. Si la expresión_de_control_1 no es cero entonces el programa ejecuta las instrucciones que están a continuación de esta expresión y luego pasa a end. Si en caso la expresión_de_control_1 fuera cero el programa chequea expresión_de_control_2 y si es distinta de cero entonces el programa ejecuta las instrucciones que están justo a continuación y luego pasa a end. Si todas las expresiones de control son cero entonces se ejecutan las instrucciones que están justo a continuación de else y pasa a end.

Puede haber cualquier número de *elseif* en un *if* pero debe de haber a lo más un *else*. Si no hay *else* el programa continua después del *end* sin ejecutar ninguna parte del *if*. En la mayoría de casos las expresiones de control son alguna combinación de operadores lógicos y relacionales. A continuación se da un ejemplo del uso del *if* donde la nota de una materia se pasa de números a letras, siguiendo las siguientes condiciones, si la nota es mayor de 95 se le asigna A, si la nota está entre 86 y 95 se le asigna B, si está entre 76 y 85 se le asigna C, si está entre 66 y 75 se le asigna D y si es menor de 66 se le asigna F.

```

if nota > 95
    disp ('La nota es A.');
```

```

else
    if nota > 86
        disp ('La nota es B.');
```

```

    else
        if nota > 76
            disp ('La nota es C.');
```

```

        else
            if nota > 66
                disp ('La nota es D.');
```

```

            else
                disp ('La nota es F.');
```

```

            end
        end
    end
end
end
end

```

La otra instrucción que ofrece el MATLAB es el *switch*. Esta ramificación se puede escribir en términos generales de la siguiente forma

```

switch (expresion_de_switch)
    case expresion_de_caso_1,
        Instruccion 1
        Instruccion 2
        ...
    case expresion_de_caso_2,
        Instruccion 1
        Instruccion 2
        ...
    ...
otherwise,
    Instruccion 1

```

```

Instruccion 2
...
end

```

En MATLAB los ciclos son comandos que nos permiten ejecutar secuencias de instrucciones dos o más veces. Hay dos formas básicas de ciclos que son los ciclos *while* y los ciclos *for*. La principal diferencia entre estos dos tipos de ciclos es la forma en que se controla la repetición. En un ciclo *while* las instrucciones son repetidas un número indefinido de veces hasta que una condición especificada por el usuario es satisfecha. En contraste en un ciclo *for* las instrucciones son repetidas en un número específico de veces, y el número de repeticiones es conocido antes que el ciclo empiece.

El ciclo *while* consiste en un bloque de instrucciones que es repetido indefinidamente hasta que se satisface la condición especificada en una expresión a la par de *while*. La forma general del ciclo *while* es la siguiente

```

while expresion
    Instruccion 1
    Instruccion 2
    Instruccion 3
    ...
end

```

Si la expresión es verdadera el bloque de instrucciones se ejecuta y regresa a la expresión de *while*, si la expresión es todavía verdadera las instrucciones se ejecutan otra vez. Este proceso se repetirá hasta que la expresión sea falsa. Cuando la expresión de *while* sea falsa el programa ejecutará la primera instrucción después de *end*. A continuación se ilustra el uso de el ciclo *while* para calcular el factorial de un número entero.

```

N = input('Ingrese el número: ');
n_factorial = 1;
n = 1;
while n <= N
    n_factorial = n_factorial * n;
    n = n+1;
end
disp (n_factorial);

```

El ciclo *for* es un ciclo que ejecuta las instrucciones un número de veces especificado. La forma general del ciclo *for* es

```

for indice = expresion
    Instruccion 1
    Instruccion 2
    ...
    Instruccion n
end

```

donde el índice es la variable del ciclo también conocida como índice del ciclo y expresión es la expresión que nos sirve para controlar el ciclo. La expresión normalmente es una

matriz donde cada columna es almacenada en la variable índice por cada ciclo. Así el ciclo *for* ejecuta las instrucciones una vez por cada columna. La expresión típica en el ciclo *for* es un vector fila que toma la siguiente forma

primero:incremento:último

La expresión anterior es una manera breve de escribir vectores fila tal y como se explicó anteriormente. Como ejemplo vamos a calcular el factorial de un número entero usando el ciclo *for*

```
n = input('Ingrese el número: ');
n_factorial = 1
for ii = 1:n
    n_factorial = n_factorial * ii;
end
```

Es posible que un ciclo esté completamente dentro de otro ciclo. Si un ciclo está completamente dentro de otro, se dice que estos ciclos están anidados. En el programa de esta tesis los ciclos anidados se usan ampliamente. El siguiente ejemplo muestra dos ciclos anidados para calcular y escribir el producto de dos enteros.

```
for ii = 1:3
    for jj = 1:3
        producto = ii * jj;
        disp ([num2str(ii), '+', num2str(jj), '=', num2str(producto)]);
    end
end
```

Si el programa anterior se corre da como resultado lo siguiente

```
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
```

5.10 Ploteo

Una de las características más poderosas de MATLAB es su capacidad de ploteo. El ploteo de datos en MATLAB es relativamente fácil. Por ejemplo, para plotear un conjunto de datos se puede crear un par de vectores y se usa la función *plot*. Los resultados de la función *plot* siempre aparecen en la ventana de figuras y no en la ventana de comandos. La ventana de figuras normalmente no aparece en la ventana frontal inicial. Sin embargo, al usar la función *plot* la ventana de figuras aparece automáticamente.

Para mostrar el uso de la función *plot* se puede crear un vector x de 0 hasta 10 con incrementos de una unidad y el vector y que se define de la siguiente manera $y = x^2 - 10x + 15$. Luego simplemente se le pide que plotee x y y . Todo lo anterior se puede escribir en la ventana de comandos de la siguiente forma

```
x = 0:1:10;
y = x.^2-10.*x+15;
plot(x,y);
```

Cuando la función *plot* es ejecutada en MATLAB abre de inmediatamente la ventana de figuras y despliega la gráfica que se muestra en la siguiente figura.

La presentación de la gráfica mostrada en la figura 51, puede ser mejorada añadiendo títulos y etiquetas a los ejes. La función que se usa para escribir títulos es *title*, la función para ponerle etiqueta al eje x es *xlabel* y para ponerle etiqueta al eje y es la función *ylabel*. También se puede usar la función *grid on* que le agrega una rejilla a la gráfica. En caso de que se quiera eliminar la rejilla se usa la función *grid off*. A continuación se ejemplifica el uso de estas funciones.

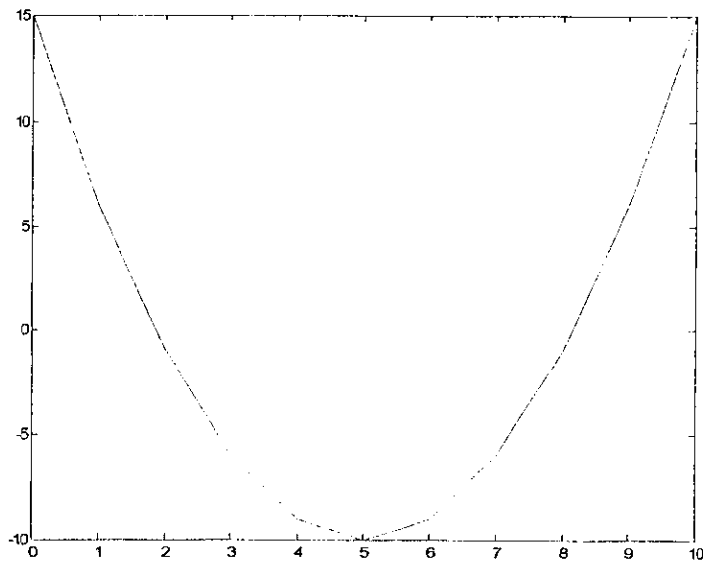


Figura 51. Ploteo de la función $y = x^2 - 10x + 15$ de 0 a 10.

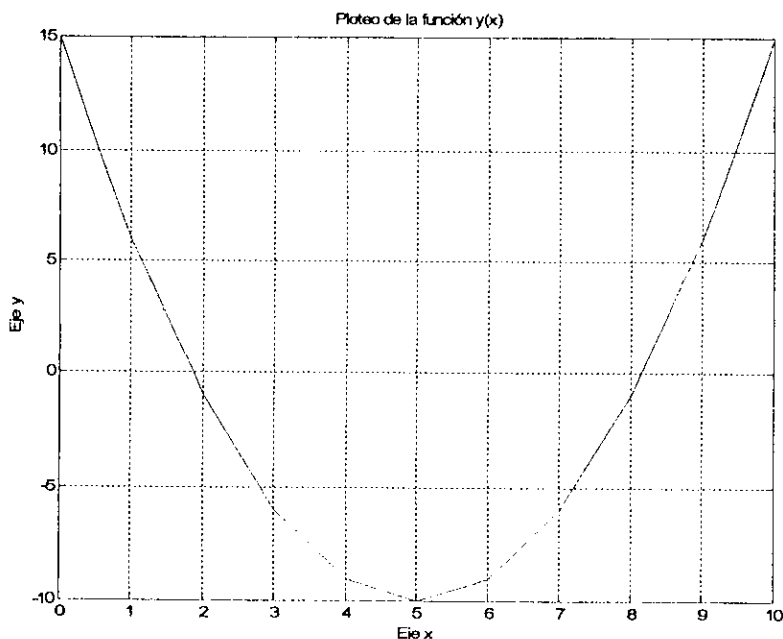


Figura 52. Ploteo de la función $y(x)$ con título, etiquetas y rejilla.

```
x = 0:1:10;
y = x.^2-10.*x+15;
plot(x,y);
title('Ploteo de la función y(x)');
xlabel('Eje x');
ylabel('Eje y');
grid on;
```

Ya una vez creado un ploteo este se puede mandar a imprimir en una impresora usando el comando *print* o presionando con el ratón el icono de “Print Figure”.

En MATLAB es posible efectuar ploteos múltiples en una misma gráfica. Para un ploteo múltiple simplemente se pone más de un conjunto de valores (x,y) en la función de ploteo *plot*. Por ejemplo, supongamos que se quiere plotear la función $f(x) = \sin(2x)$ y su derivada $f'(x) = 2 \cos(2x)$, entonces se generan un conjunto de valores (x, y) para cada una de las funciones y se usa la función *plot* como se muestra a continuación.

```
x = 0:pi/100:2*pi;
y1 = sin(2*x);
y2 = 2*cos(2*x);
plot(x,y1,x,y2);
```

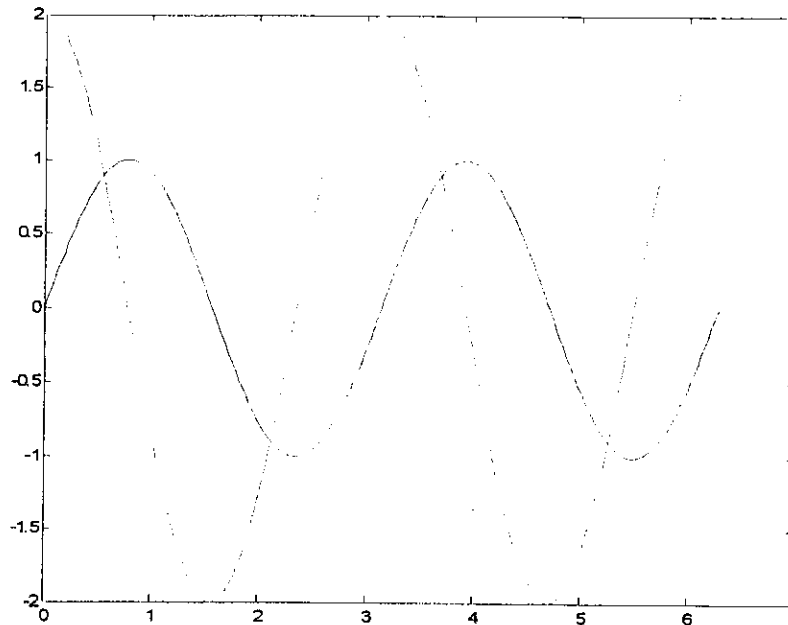


Figura 53. Ploteo de $f(x) = \sin(2x)$ y $f'(x) = 2 \cos(2x)$

MATLAB permite al programador seleccionar el color y estilo de la línea que va a ser ploteada. También permite usar diferentes tipos de marcas para datos puntuales. Estas características se pueden seleccionar usando una cadena de caracteres de atributos después de los vectores x & y en la función de ploteo. La cadena de atributos puede tener hasta tres componentes, con la primera componente especificando el color de la línea, la segunda componente especificando el estilo de la marca y la tercera componente especificando el estilo de la línea. Los símbolos para diferentes colores, marcas y estilos de línea se muestran en la siguientes tablas.

Tabla 12
Símbolos de los colores de línea en MATLAB

SÍMBOLO	COLOR
y	Amarillo
m	Magenta
c	Cian
r	Rojo
g	Verde
b	Azul
w	Blanco
k	Negro

Tabla 13
Símbolos de los estilos de marca en MATLAB

SÍMBOLO	ESTILO DE MARCA
.	Punto
o	Círculo
x	Marca x
+	Más
*	Estrella
s	Cuadrado
d	Diamante
v	Triángulo (abajo)
^	Triángulo (arriba)
<	Triángulo (izquierda)
>	Triángulo (derecha)
p	Pentagrama
h	Hexagrama
<none>	No marca

Tabla 14
Símbolos de los estilos de línea en MATLAB

SÍMBOLO	ESTILO DE LÍNEA
-	Sólida
:	Punteada
-.	Guión-punto
--	Guión-guión
<none>	No línea

Varios estilos pueden mezclarse en cualquier combinación y más de un estilo se puede usar en caso de tener más de un par de vectores. Se pueden incluir leyendas en las gráficas usando la función *legend*. La forma básica de esta función es

Legend ('cadena1', 'cadena2', ..., pos)

donde *cadena1*, *cadena2*, etc. son etiquetas asociadas con las líneas ploteadas y *pos* es un entero especificando el lugar de la leyenda. Los valores posibles para *pos* están dados en la siguiente tabla.

Tabla 15
Valores de pos en el comando *legend*

VALOR DE POS	SIGNIFICADO
0	La mejor colocación
1	Esquina superior derecha
2	Esquina superior izquierda
3	Esquina inferior izquierda
4	Esquina inferior derecha
-1	A la derecha de la gráfica

A continuación se incluye un ejemplo para ilustrar el uso de las funciones anteriores.

```
x = 0:pi/100:2*pi;
y1 = sin (2*x);
y2 = 2*cos(2*x);
plot (x,y1,'k-',x,y2,'b--');
title ('Ploteo de la función f(x)= sen 2x y su derivada');
xlabel ('Eje x');
ylabel ('Eje y');
legend ('f(x)', 'd/dx f(x)',4);
grid on;
```

MATLAB muestra los rangos del eje x y el eje y lo suficientemente amplios para mostrar todo punto en los datos que se ingresan. Sin embargo, a veces es necesario cambiar el rango de los ejes. Esto se puede hacer usando la función *axis*. En su forma general la función *axis* se escribe

```
axis([xmin xmax ymin ymax]).
```

Escrita de esta forma MATLAB coloca el rango de la x desde xmin a xmax y el rango de la y desde ymin hasta ymax. Otras funciones útiles para la presentación de los ejes se dan en la tabla 16.

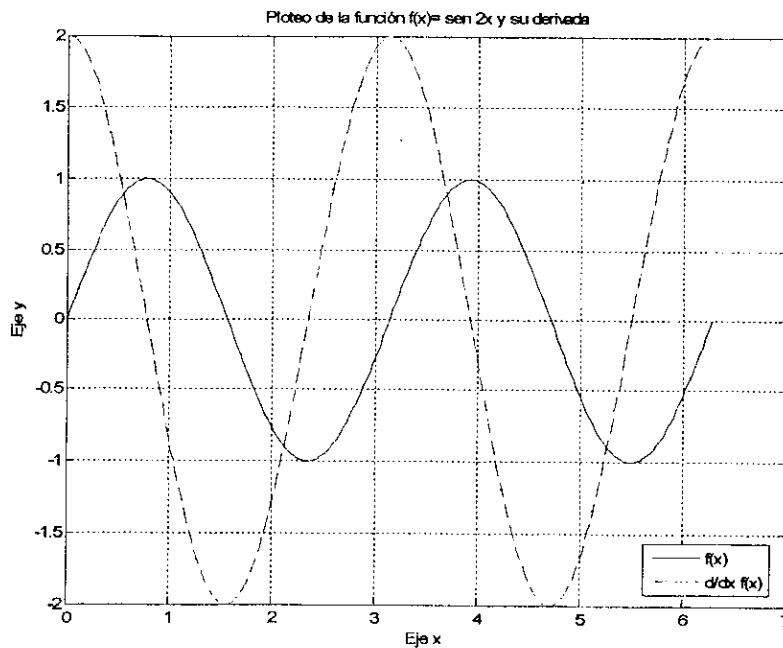


Figura 54. Un ploteo completo con título, etiquetas, leyenda, rejilla y varios estilos de línea.

Tabla 16
Comandos de la función *axis*.

COMANDO	DESCRIPCIÓN
<i>axis equal</i>	Pone los incrementos en los dos ejes iguales.
<i>axis square</i>	Hace la caja de los ejes cuadrada.
<i>axis normal</i>	Cancela los efectos de <i>axis equal</i> y <i>axis square</i> .
<i>axis off</i>	Elimina todas las etiquetas, marcas y fondos.
<i>axis on</i>	Regresa todas las etiquetas, marcas y fondos.

Normalmente, un nuevo ploteo se crea cada vez que se use la función *plot*, y la información previa se elimina. Este comportamiento se puede evitar con el comando *hold on*. Con este comando todos los ploteos adicionales aparecerán encima de los ploteos previos. Para cancelar el efecto de *hold on* se puede usar el comando *hold off*. Debido a que la función *plot* presenta sus resultados en la ventana de figuras se recomienda limpiar la ventana de figuras siempre y esto se logra usando la función *clf*.

Además de poder poner color, estilo de línea y tipo de marca, también es posible agregar cuatro propiedades adicionales. Estas cuatro propiedades se pueden lograr usando las funciones que se explican en la siguiente tabla.

Tabla 17
Funciones adicionales para las líneas y marcas.

FUNCIÓN	DESCRIPCIÓN
<i>LineWidth</i>	Especifica el ancho de cada línea.
<i>MarkerEdgeColor</i>	Especifica el color de la marca o el borde.
<i>MarkerFaceColor</i>	Especifica el color del cuerpo para marcas llenas.
<i>MarkerSize</i>	Especifica el tamaño de la marca.

Estas propiedades adicionales se especifican en la función *plot* después de los datos que van a ser ploteados.

Para plotear líneas rectas de un punto a otro punto se puede hacer fácilmente usando la función *plot*. Lo único que se requiere son las coordenadas del punto inicial y las coordenadas del punto final. Por ejemplo, si se quiere una línea recta de (2,3) a (4,7) se puede hacer de la siguiente forma

```
axis ([0 5 0 8]);
hold on;
```

```
plot(x,y,'m','LineWidth',5);
```

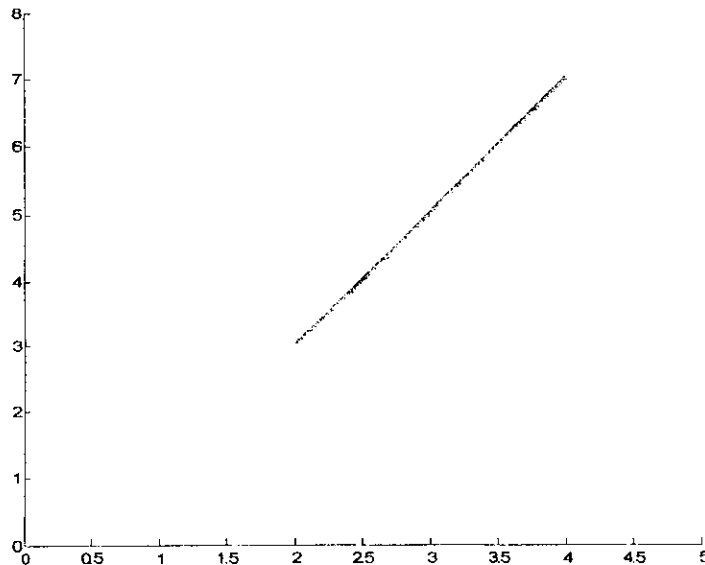


Figura 55. Ploteo de una línea recta usando LineWidth.

Tabla 18

Comparación entre MATLAB y otros compiladores, como C++, Visual Basic.

MATLAB	Otros compiladores
Todas las variables son matriciales.	Las variables no son matriciales.
Usa precisión doble sin especificarlo.	Se debe de especificar la precisión doble.
No se declaran las variables.	Hay que declarar las variables.
Las operaciones matriciales como la multiplicación, suma, transpuesta, inversa, etc. MATLAB las efectúa directamente.	Las operaciones matriciales hay que programarlas.
Tiene una amplio suministro de funciones predeterminadas que pueden ser usadas.	El uso de funciones predeterminadas es más limitado que en el MATLAB.
Facilita el ploteo y la graficación.	No facilita el ploteo y la graficación.
Ocupa bastante espacio de memoria.	Ocupa relativamente poco espacio de memoria.
Funciona como intérprete.	Funciona realmente como compilador.

VI. PROGRAMA PARA CÁLCULO DE MARCOS

6.1 Introducción

Este capítulo está dedicado a la presentación y explicación del programa de computadora que se ha elaborado para el análisis estructural de marcos rígidos. La elaboración de este programa, es el objetivo principal de este trabajo de tesis. Los dos elementos básicos que se usaron para escribir el programa son el método de rigidez y MATLAB. El método de rigidez está expuesto en el capítulo 4, y los aspectos esenciales de MATLAB que se usan en el programa están descritos en el capítulo 5. Como ya se ha mencionado anteriormente, las computadoras modernas han hecho posible que se puedan efectuar un gran número de operaciones matemáticas y lógicas rápidamente y a un bajo costo. Debido a esto, el método de rigidez se ha convertido en el mejor método para el análisis de estructuras, dejando a los métodos iterativos como el método de Cross y el de Kani en segundo plano. Además desde el punto de vista teórico el método de rigidez es el más fundamental y completo.

La formulación del método de desplazamientos es matricial. MATLAB manipula las variables en forma matricial y permite programar fácilmente. Al usar MATLAB para programar el método de rigidez todo se facilita y se obtienen excelentes resultados. Es por estas razones que en esta tesis se propone el uso del método de rigidez con MATLAB para el análisis estructural.

Además de la presentación y explicación del programa se incluyen algunos ejemplos para ilustrar su utilidad. El lector podrá involucrarse fácilmente en los detalles del programa y si así lo prefiere podrá hacer modificaciones, mejoras o ampliaciones que se adapten mejor a sus necesidades. En este momento es oportuno puntualizar que el método de rigidez puede generalizarse fácilmente al método de elementos finitos, el cual es muy útil para el análisis de estructuras más complejas.

6.2 Estructura general del programa

La estructura general del programa es bastante simple y se puede resumir en los siguientes pasos.

- Ingreso de datos.
- Numeración de los grados de libertad.
- Cálculo de la matriz de rigidez elemental en coordenadas locales.
- Cálculo del vector de fuerza elemental en coordenadas locales.

- Cálculo de la matriz de rotación de coordenadas locales a coordenadas globales.
- Transformar cada matriz de rigidez elemental de coordenadas locales a coordenadas globales.
- Ensamblar cada matriz de rigidez elemental en coordenadas globales a la matriz de rigidez total en los lugares apropiados.
- Transformar cada vector de fuerza elemental de coordenadas locales a coordenadas globales.
- Ensamblar cada vector de fuerza elemental en coordenadas globales al vector de fuerza total.
- Calcular los desplazamientos.
- Calcular las reacciones en los apoyos.
- Calcular las fuerzas en los extremos de cada miembro.
- Presentación de resultados.

Los pasos que se acaban de describir se presentan en forma esquemática en un diagrama de flujo que se muestra en la figura 56.

6.3 Ingreso de datos

El primer dato que pide el programa es la regularidad del marco y está bajo una variable que usa el nombre de regularidad. Se consideran tres niveles de regularidad, en el nivel 0 no tiene ningún tipo de regularidad y por lo tanto en el siguiente paso se deben de ingresar la matriz de nodos y la matriz de miembros. Es decir, si se ingresa el valor de 0 en la variable de regularidad entonces el programa pide que se ingrese la matriz de nodos y la matriz de miembros. En este programa el ingreso de datos se puede ilustrar por medio del diagrama de flujo que se ilustra en la figura 57.

El ingreso de la regularidad en este programa es bastante sencillo. Al correr el programa aparece lo siguiente:

```
>> Rigidez  
Ingrese nivel de regularidad :
```

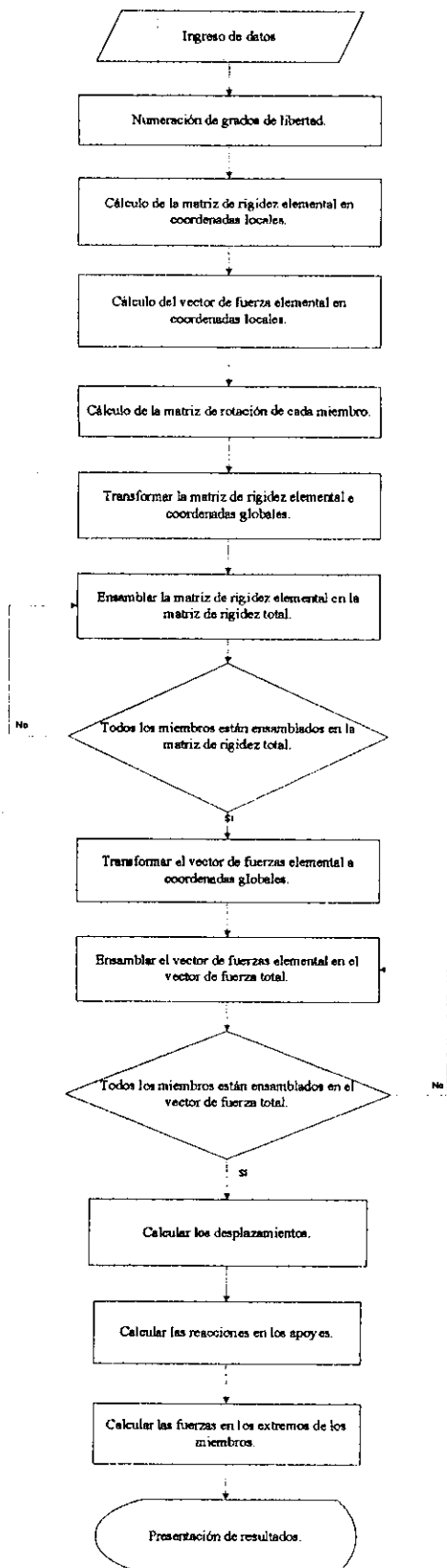


Figura 56. Diagrama de flujo general del programa

Inmediatamente después de la frase “Ingrese nivel de regularidad:” se escribe 0 ó 1 ó 2, y el programa solicitará más datos según sea el caso. Se sugiere que antes de correr el programa se tengan ya preparadas las matrices o datos de ingreso en el espacio de trabajo y no esperar hasta después de correr el programa.

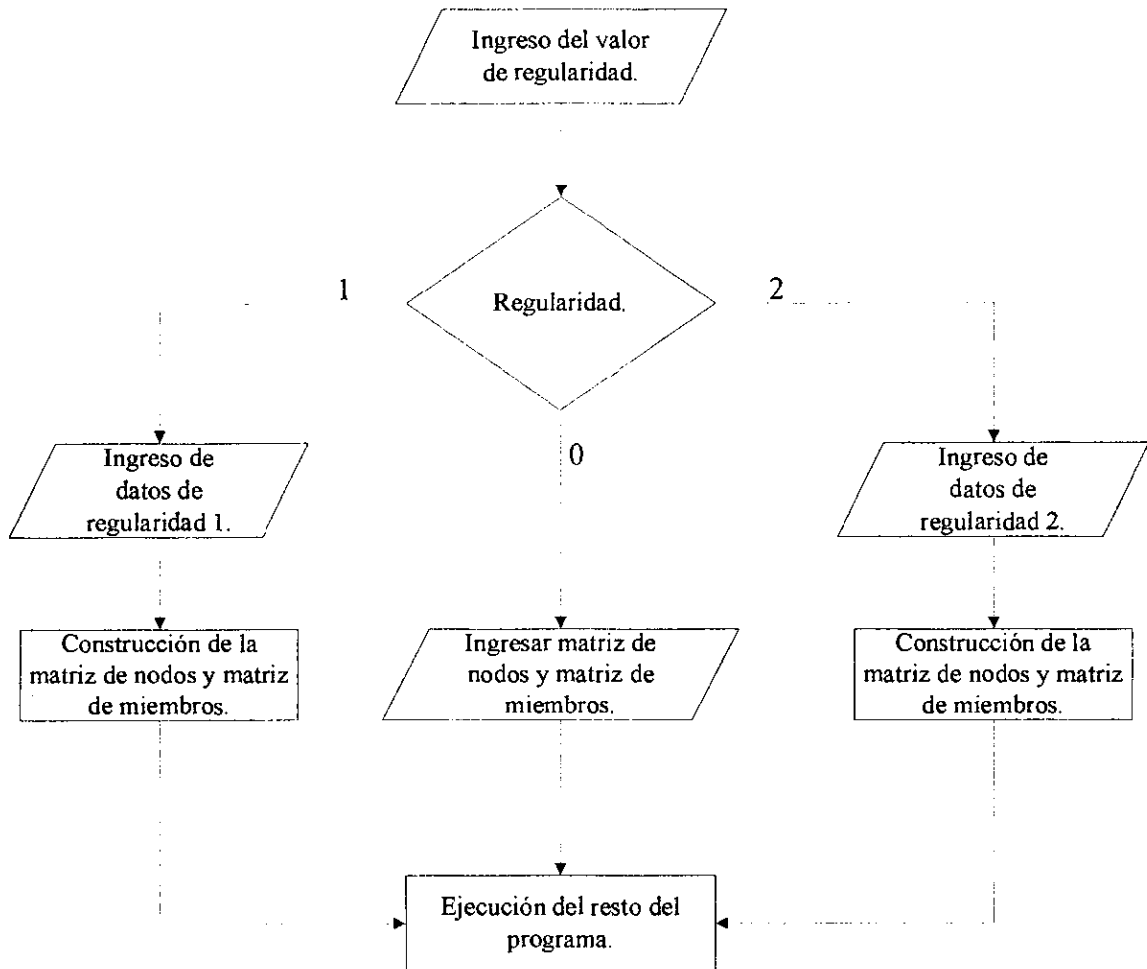


Figura 57. Diagrama de flujo para el ingreso de datos.

Si la estructura es un marco donde la altura de las columnas es la misma para cada piso, la carga distribuida sobre las vigas es la misma sobre cada luz, la longitud de las luces es la misma para cada piso, las cargas puntuales son las mismas en cada luz y las cargas horizontales solo dependen de los pisos, entonces a esta estructura la podemos considerar que tiene regularidad 1. Por lo tanto, si se ingresa el valor de 1 en la variable de regularidad, el programa solicita que se ingresen los siguientes datos:

- Vector de altura de pisos.
- Vector de longitud de luces.
- Módulo de elasticidad.
- Vector de área de vigas.

- Vector de área de columnas.
- Vector de inercia de vigas.
- Vector de inercia de columnas.
- Vector de fuerzas horizontales.
- Vector de cargas distribuidas sobre las vigas.
- Vector de cargas puntuales al centro de las vigas.

Luego de ingresar estos datos el programa mismo genera la matriz de nodos y la matriz de miembros sin necesidad de que se tenga que preocupar por la formación de dichas matrices.

Si la estructura es un marco donde todas las luces son de igual longitud, todas las alturas de los pisos son iguales, las cargas distribuidas y puntuales sobre las vigas son iguales y todas las características de las vigas y columnas son las mismas, entonces a esta estructura la podemos considerar que tiene regularidad 2. Por lo tanto, si se ingresa el valor de 2 en la variable de regularidad el programa solicita que se ingresen los siguientes datos:

- Número de pisos.
- Número de luces.
- Altura de piso.
- Longitud de luz.
- Módulo de elasticidad.
- Área de las vigas.
- Área de las columnas.
- Inercia de las vigas.
- Inercia de las columnas.
- Fuerza horizontal en los pisos.
- Carga distribuida sobre las vigas.
- Carga puntual al centro de las vigas.

Las doce variables que se solicitan en el nivel de regularidad 2 son escalares, por lo tanto en este caso el ingreso de datos es bastante sencillo y rápido, solamente se deben de ingresar doce números escalares en forma secuencial. Luego el programa construye la matriz de nodos y la matriz de miembros que son los datos realmente necesarios para que corra el programa.

En caso de que se tenga que ingresar vectores o matrices y estemos usando MATLAB, se sugiere que antes de correr el programa o de llamarlo se construyan las matrices en la ventana de comandos y se guarden en el espacio de trabajo, dándoles siempre un nombre adecuado. El ingreso se facilita aún más si se definen las matrices en la ventana de comandos usando la función *zeros (n,m)* y luego se editan en el editor de arreglos. Cuando ya se tienen las matrices en el espacio de trabajo se corre el programa y se llaman a las variables directamente desde el espacio de trabajo.

En el caso de tener regularidad 0 se deben de ingresar la matriz de nodos y la matriz de miembros. A continuación se explica la información que lleva cada una de estas matrices.

La matriz de nodos es una matriz que tiene n filas por 8 columnas, donde n es el número de nodos que tiene el marco. La primera fila de esta matriz contiene toda la información pertinente del primer nodo y así sucesivamente hasta llegar al nodo n-ésimo. En términos generales se puede decir que la i-ésima fila de la matriz de nodos contiene toda la información del nodo i-ésimo. Luego la información que viene en cada columna se explica en la siguiente tabla:

Tabla 19
Información en la matriz de nodos.

COLUMNA	INFORMACIÓN
1	Restricción en x.
2	Restricción en y.
3	Restricción en z.
4	Coordenada en x.
5	Coordenada en y.
6	Fuerza externa sobre el nodo en x.
7	Fuerza externa sobre el nodo en y.
8	Fuerza externa sobre el nodo en z.

Los valores que se ingresan en las restricciones de nodo que ocupan las tres primeras columnas de la matriz de nodos son de la siguiente forma. En la restricción en x se ingresa 1 si el desplazamiento horizontal en el nodo está restringido, en caso contrario se ingresa 0. La restricción en y es igual que la restricción en x solamente que se consideran los desplazamientos verticales. Y la restricción en z considera los giros y sigue la misma forma de ingreso que la restricción en x y en y. En las otras columnas se deben de ingresar los datos tal y como se explica en la tabla 19.

La matriz de nodos MN en forma esquemática se puede visualizar tal y como se muestra en la figura 58.

$$MN = \begin{matrix} & r_x & r_y & r_z & x & y & f_x & f_y & m_z \\ \text{nodo 1} & & & & & & & & \\ \text{nodo 2} & & & & & & & & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \text{nodo n} & & & & & & & & \end{matrix}$$

Figura 58. Matriz de nodos.

En esta figura r_x , r_y y r_z son las restricciones en los grados de libertad del nodo en cuestión. Las restricciones toman valores de 0 ó 1 tal y como se explicó anteriormente. Las coordenadas del nodo referido al sistema global se ingresan en las columnas x e y . Por último, si hay fuerzas que actúan directamente sobre el nodo se ingresan en las últimas tres columnas, siendo f_x la fuerza horizontal, f_y la fuerza vertical y m_z el momento.

La matriz de miembros es una matriz de m filas por 14 columnas, donde m es el número de miembros que tiene el marco. La primera fila de esta matriz contiene toda la información pertinente del primer miembro y así sucesivamente hasta llegar al miembro m -ésimo. En términos generales se puede decir que la i -ésima fila de la matriz de miembros contiene toda la información del miembro i -ésimo. Luego la información que viene en cada columna se explica en la siguiente tabla:

Tabla 20
Información de la matriz de miembros.

COLUMNA	INFORMACIÓN
1	Nodo inicial
2	Nodo final.
3	Área transversal del miembro.
4	Módulo de elasticidad del miembro.
5	Momento de inercia del miembro.
6	Carga puntual en x referido a coordenadas locales.
7	Posición de la carga anterior.
8	Carga puntual en y referido a coordenadas locales.
9	Posición de la carga anterior.
10	Carga puntual en z referido a coordenadas locales.
11	Posición de la carga anterior.
12	Carga uniformemente distribuida en x referido a coordenadas locales.
13	Carga uniformemente distribuida en y referida a coordenadas locales.
14	Carga linealmente distribuida en y referida a coordenadas locales.

Los valores que se ingresan en las dos primeras columnas son el número de nodo inicial y el número de nodo final del miembro en cuestión. El número de nodo está dado por la matriz de nodos que corresponde al número de fila. Luego vienen las columnas de área transversal, módulo de elasticidad y momento de inercia del miembro. Las siguientes seis columnas son para ingresar información acerca de las cargas puntuales y su posición dentro del miembro, las fuerzas puntuales se ingresan referidas al sistema de coordenadas local del miembro y su posición es la posición relativa dentro del miembro, por lo tanto se

debe de ingresar un número entre 0 y 1 para la posición relativa, donde 0 significa que la carga puntual está en el nodo inicial del miembro y 1 significa que la carga puntual está localizada en el nodo final del miembro. En caso que la carga puntual se encuentre en un punto intermedio del miembro la posición relativa es un número de la forma 0.x. Se sugiere que si una carga puntual está aplicada directamente a un nodo nunca se ingrese en la matriz de miembros sino que se ingrese en la matriz de nodos, lo anterior se hace para evitar repeticiones indebidas, por lo tanto, en la posición relativa de las cargas puntuales nunca debería de haber un valor de 0 o de 1. En la columna 12 va la carga uniformemente distribuida en x referida al sistema de coordenadas locales. En la columna 13 se escribe la carga uniformemente distribuida en la dirección del eje y, siempre referida al sistema de coordenadas locales. En la última columna se refiere a una carga linealmente distribuida desde 0 hasta un máximo y se ingresa su valor máximo, para este caso de carga solo se incluyó en la dirección en y.

La matriz de miembro MM en forma esquemática se puede visualizar tal y como se muestra en la figura 59.

	<i>nodo</i>	<i>nodo</i>	<i>A</i>	<i>E</i>	<i>I</i>	P_x	<i>x</i>	P_y	<i>y</i>	m_z	<i>z</i>	w_x	w_y	<i>wl</i>																																																												
	<i>inicial</i>	<i>final</i>																																																																								
miembro 1	<table style="width: 100%; height: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> <td style="border: 1px solid black; width: 10%;"></td> </tr> <tr> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> </tr> <tr> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> </tr> <tr> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> <td style="border: 1px solid black;"></td> </tr> </table>																																																																									
miembro 2																																																																										
.																																																																										
.																																																																										
.																																																																										
miembro m																																																																										

Figura 59. Matriz de miembros.

En la figura anterior nodo inicial es el número de nodo del extremo inicial del miembro, nodo final es el número de nodo del extremo final del miembro. *A* es el área transversal del miembro, *E* es el módulo de elasticidad del material e *I* es el momento de inercia del miembro. P_x y *x* son la carga puntual en x y su posición relativa en el miembro. P_y y *y* es la carga puntual en y y su posición relativa en el miembro. m_z y *z* es el momento puntual y su posición relativa en el miembro. w_x es la carga uniformemente distribuida en la dirección axial del miembro, w_y es la carga uniformemente distribuida en la dirección transversal del miembro. Y por último, *wl* es la carga linealmente distribuida en la dirección transversal del miembro.

6.4 Presentación de resultados

Por medio del método de rigidez se calculan tres resultados. El primero de estos resultados son los desplazamientos en los nodos, el cual queda en forma matricial como un vector columna denotada por D . El segundo de estos resultados son las fuerzas en los extremos de cada miembro, quedando en un arreglo denotado por RM . Y por último, se calculan las reacciones en los apoyos las cuales quedan guardadas en el vector denotado por R .

Para que la lectura de los datos sea fácil se usó la función *disp* que sirve para desplegar los resultados que da el programa. Se pensó que aparecieran los resultados referidos al miembro o al nodo al que pertenecían.

De esta forma, para dar los resultados de las fuerzas en los extremos de cada miembro del marco rígido, se escribe un título que dice "Las fuerzas en el extremo inicial del miembro m son:" y en la siguiente línea aparecen los resultados ordenadamente como fuerza axial, corte y momento. Inmediatamente después aparece otro título que dice "Las fuerzas en el extremo final del miembro m son:" y en la siguiente línea aparecen los resultados en forma ordenada como fuerza axial, corte y momento. Además aparece una figura que en forma esquemática muestra la localización del miembro en cuestión. Un ejemplo de despliegue de los resultados de las fuerzas en los miembros del marco rígido 1 se muestra a continuación.

FUERZAS EN LOS EXTREMOS DEL MIEMBRO

Las fuerzas en el extremo inicial del miembro 1 son:
 Axial = 400; Corte = -80.6723; Momento = -62.7451

Las fuerzas en el extremo final del miembro 1 son:
 Axial = -400; Corte = 80.6723; Momento = -179.2717

Las fuerzas en el extremo inicial del miembro 2 son:
 Axial = 400; Corte = 80.6723; Momento = 62.7451

Las fuerzas en el extremo final del miembro 2 son:
 Axial = -400; Corte = -80.6723; Momento = 179.2717

Las fuerzas en el extremo inicial del miembro 3 son:
 Axial = 80.6723; Corte = 400; Momento = 179.2717

Las fuerzas en el extremo final del miembro 3 son:
 Axial = -80.6723; Corte = 400; Momento = -179.2717

Y junto con estos resultados aparece la figura que se muestra a continuación.

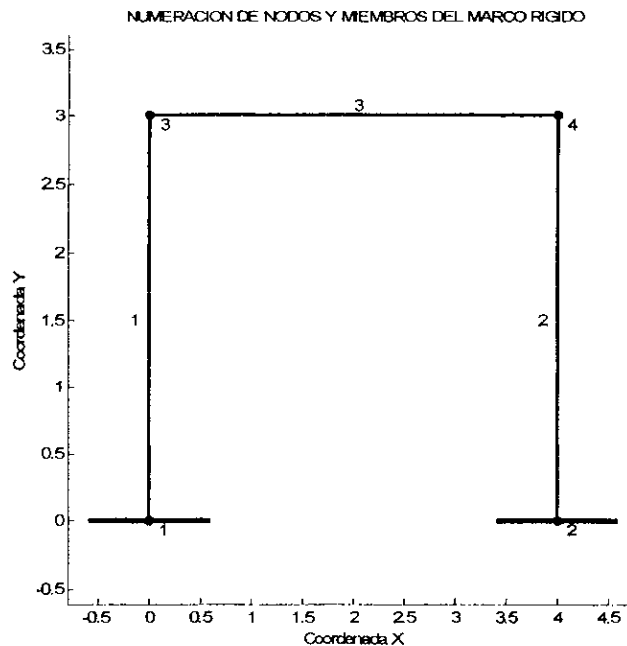


Figura 60. Figura del marco rígido 1.

La presentación de los desplazamientos de los nodos se hace de la siguiente forma: Primero aparece un título que dice "Los desplazamientos y el giro del NODO n son:", luego en la segunda fila aparecen ordenadamente el desplazamiento en x, el desplazamiento en y y el giro. En el siguiente ejemplo referido a la figura 60 se ilustra este despliegue de resultados.

DESPLAZAMIENTOS EN LOS NODOS

Los Desplazamientos y el Giro del NODO 1 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 2 son:
Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 3 son:
Horizontal = 4.4818e-005; Vertical = -0.00066667; Giro = -9.7106e-005 rad

Los Desplazamientos y el Giro del NODO 4 son:
Horizontal = -4.4818e-005; Vertical = -0.00066667; Giro = 9.7106e-005 rad

Y por último, para presentar los resultados para las reacciones en los apoyos se sigue un procedimiento parecido a los desplazamientos. De esta forma, primero aparece un título que dice “Las reacciones en el Apoyo n son:” y luego en la siguiente fila aparecen ordenadamente la reacción horizontal, la reacción vertical y el momento. El número de apoyo es el mismo que el número de nodo. Solamente se consideran apoyos donde por lo menos un grado de libertad está restringido a diferencia de los desplazamientos donde se dan todos los nodos. Para ilustrar la manera en que se presentan los resultados en la computadora se da el siguiente ejemplo referido a la figura 60.

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 1 son:
Horizontal = 80.6723; Vertical = 400; Momento = -62.7451

Las Reacciones en el Apoyo 2 son:
Horizontal = -80.6723; Vertical = 400; Momento = 62.7451

El programa permite trabajar en cualquier sistema de unidades, por lo tanto es el usuario el responsable de saber en que unidades están los resultados. Es por eso que solo aparecen los números sin mencionar las dimensionales. La única excepción es el giro, porque sin importar el sistema de unidades que se esté usando el ángulo girado siempre se dará en radianes.

Para una rápida y fácil referencia se incluye una figura que muestra directamente los nodos y los miembros con su numeración respectiva. Esto facilita al usuario localizar cada uno de los miembros en el marco rígido. Los miembros, como las vigas y columnas, salen en color azul y su número respectivo también en color azul. En el caso de los nodos

aparecen puntos de color negro con su número respectivo en color negro. Esto facilita la lectura ya que los números negros se refieren a los nodos y los números azules al miembro. Los números siempre aparecen relativamente cerca al miembro o nodo correspondiente.

La figura también muestra claramente el sistema de coordenadas global. Generalmente el eje x aparece horizontal y el eje y aparece vertical. Para una mejor presentación de la figura se distinguen los apoyos que están completamente empotrados por medio de un segmento de línea de color negro, que justamente aparece encima del nodo correspondiente.

Para ilustrar la manera en la que se presenta la figura se incluye un ejemplo de un marco regular de dos niveles y tres luces en la figura 61. Nótese la facilidad de la lectura para identificar miembros, nodos y apoyos que están totalmente empotrados.

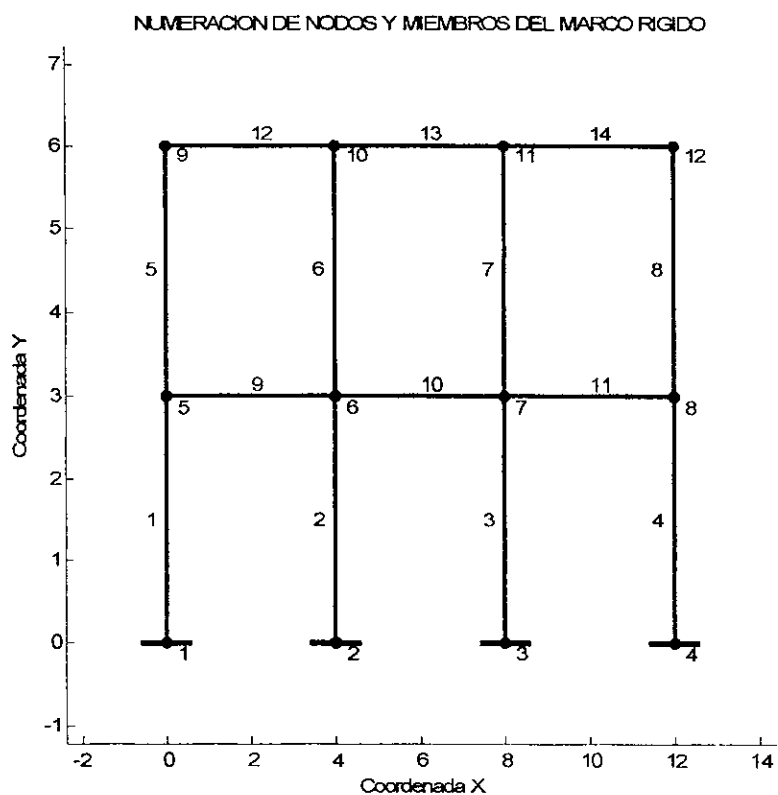


Figura 61. Marco rígido regular de dos niveles y tres luces.

En este programa se obliga a que los ejes se presenten en forma cuadrada, ya que es la forma más fácil de incluir en una figura cuadrada cualquier marco rígido, por lo tanto la escala del eje x y del eje y puede variar, tal y como se muestra en la figura 61.

6.5 Código del programa en MATLAB

El programa ha sido escrito usando MATLAB. La programación en MATLAB está explicada en el capítulo 5 de este trabajo. Para una exposición completa se incluye el código de programa a continuación. El lector podrá seguirlo sin mucha dificultad.

```

% *****
% *          PROGRAMA PARA          *
% *   ANALISIS DE MARCOS RIGIDOS   *
% *   USANDO EL METODO DE RIGIDEZ  *
% *           POR                   *
% *   ALEXANDER BECKER             *
% * *****                         *
%
% Este es un programa para analizar marcos rígidos usando el método de
% rigidez. Para iniciar el programa se le pide al usuario que dé el nivel de
% regularidad de la estructura. Si el marco no muestra ninguna regularidad
% se ingresa cero en la variable de regularidad. Si muestra algo de
% regularidad se ingresa 1 y si muestra mucha regularidad se ingresa 2
regularidad = input('Ingrese nivel de regularidad :');

%MN es la matriz de nodos y MM es la matriz de miembros

if regularidad == 0
    MN = input('Ingrese la matriz de nodos MN :');
    MM = input('Ingrese la matriz de miembro MM :');
    [nn, nnn]=size(MN);
    [nm, nmm]=size(MM);
elseif regularidad == 1
    Hp = input('Ingrese vector de altura de pisos : ');
    Ll = input('Ingrese vector de longitud de luces : ');
    me = input('Ingrese módulo de elasticidad : ');
    Av = input('Ingrese vector de area de vigas : ');
    Ac = input('Ingrese vector de area de columnas : ');
    Iv = input('Ingrese vector de inercia de vigas : ');
    Ic = input('Ingrese vector de inercia de columnas : ');
    Fh = input('Ingrese vector de fuerzas horizontales : ');
    Cd = input('Ingrese vector de cargas distribuidas sobre las vigas : ');
    Cp = input('Ingrese vector de cargas puntuales sobre las vigas : ');

%A partir de los datos ingresados en regularidad 1 se construyen MM yMN.

    [np1, np]=size(Hp);
    [nl1, nl]=size(Ll);
    nn=(nl+1)*(np+1);
    nm=(nl+1)*np+(nl*np);
    MN(1, :)= [1, 1, 1, 0, 0, 0, 0, 0];
    MN(nl+2, :)= [0, 0, 0, 0, Hp(1), Fh(1), 0, 0];
    for ii = 2:nl+1
        x(1)=0;
        x(ii)=x(ii-1)+Ll(ii-1);
        MN(ii, :)= [1, 1, 1, x(ii), 0, 0, 0, 0];
        for jj = 2:np+1
            y(1)=0;
            y(jj)=y(jj-1)+Hp(jj-1);
            MN((jj-1)*(nl+1)+1, :)= [0, 0, 0, 0, y(jj), Fh(jj-1), 0, 0];
        end
    end
end

```

```

        MN((jj-1)*(nl+1)+ii,:)=[0,0,0,x(ii),y(jj),0,0,0];
    end
end

for ii = 1:np
    for jj = 1:nl+1
        MM((ii-1)*(nl+1)+jj,:) = [(ii-1)*(nl+1)+jj,ii*(nl+1)+jj,...
            Ac(ii),me,Ic(ii),0,0,0,0,0,0,0,0];
    end
    for jj = 1:nl
        MM((nl+1)*np+jj+(ii-1)*nl,:) = [ii*(nl+1)+jj,...
            ii*(nl+1)+jj+1,Av(jj),me,Iv(jj),0,0,Cp(jj),...
            0.5,0,0,0,Cd(jj),0];
    end
end

else
    np = input('Ingrese número de pisos : ');
    nl = input('Ingrese número de luces : ');
    hp = input('Ingrese altura de piso : ');
    ll = input('Ingrese longitud de luz : ');
    me = input('Ingrese el módulo de elasticidad : ');
    av = input('Ingrese el área de las vigas : ');
    ac = input('Ingrese el área de las columnas : ');
    iv = input('Ingrese la inercia de las vigas : ');
    ic = input('Ingrese la inercia de las columnas : ');
    fh = input('Ingrese fuerza horizontal en los pisos : ');
    cd = input('Ingrese carga distribuida sobre las vigas : ');
    cp = input('Ingrese carga puntual al centro de las vigas : ');

    *A partir de los datos ingresados en regularidad 2 se construyen MM y MN.

    nn=(nl+1)*(np+1);
    nm=(nl+1)*np+(nl*np);
    for ii = 1:nl+1
        MN(ii,:)=[1,1,1,ll*(ii-1),0,0,0,0];
    end
    for jj = 1:np
        MN(jj*(nl+1)+1,:) = [0,0,0,0,jj*hp,fh,0,0];
    end
    for ii = 1:nl
        for jj = 1:np
            MN(jj*(nl+1)+1+ii,:) = [0,0,0,ii*ll,jj*hp,0,0,0];
        end
    end
    for ii = 1:np
        for jj = 1:nl+1
            MM((ii-1)*(nl+1)+jj,:) = [(ii-1)*(nl+1)+jj,...
                (ii*(nl+1)+jj),ac, me,ic,0,0,0,0,0,0,0,0];
        end
        for jj=1:nl
            MM((nl+1)*np+jj+(ii-1)*nl,:)=[(ii*(nl+1)+jj),...
                (ii*(nl+1)+jj+1),av,me,iv,0,0,cp,0.5,0,0,0,cd,0];
        end
    end
end

end

% Con las matrices de nodo y de miembro se define la matriz RESTRICION la
% cual suma en forma acumulada los grados de libertad restringidos.
% El primer índice indica el nodo,el segundo índice se refiere al grado de
% libertad. Cada nodo tiene tres grados de libertad.

```

```

RESTRICCION(1,1)=MN(1,1);
RESTRICCION(1,2)=RESTRICCION(1,1)+MN(1,2);
RESTRICCION(1,3)=RESTRICCION(1,2)+MN(1,3);
for ii = 2:nn
    for jj = 1:3
        if jj == 1
            RESTRICCION(ii,jj)=RESTRICCION(ii-1,3)+MN(ii,jj);
        else
            RESTRICCION(ii,jj)=RESTRICCION(ii,jj-1)+MN(ii,jj);
        end
    end
end

%glnr es el número total de grados de libertad no restringidos mientras que
%restriccion(nn,3) es número de grados de libertad restringidos

glnr=RESTRICCION(nn,3);
glnr=(3*nn)-glnr;

%La matriz GL reordena los grados de libertad poniendo primero los no
%restringidos y de último los restringidos

for ii = 1:nn
    for jj = 1:3
        if MN(ii,jj) == 0
            GL(ii,jj)=3*(ii-1)+(jj-RESTRICCION(ii,jj));
        else
            GL(ii,jj)=glnr+RESTRICCION(ii,jj);
        end
    end
end

% A continuación se calcula la matriz de rigidez de un miembro prismático
% referido a sus coordenadas locales. KM es la matriz de rigidez de un
% miembro

for ii = 1:nm
    deltax=MN(MM(ii,2),4)-MN(MM(ii,1),4);
    deltay=MN(MM(ii,2),5)-MN(MM(ii,1),5);
    Lon(ii)=sqrt(deltax^2+deltay^2);
    Cos(ii)=deltax/Lon(ii);
    Sen(ii)=deltay/Lon(ii);
    A=MM(ii,3);
    E =MM(ii,4);
    I =MM(ii,5);
    L =Lon(ii);
    KM(:, :, ii)=[E*A/L, 0, 0, -E*A/L, 0, 0 ;
                  0, 12*E*I/(L^3), 6*E*I/(L^2), 0, -12*E*I/(L^3), 6*E*I/(L^2);
                  0, 6*E*I/(L^2), 4*E*I/L, 0, -6*E*I/(L^2), 2*E*I/L ;
                  -E*A/L, 0, 0, E*A/L, 0, 0 ;
                  0, -12*E*I/(L^3), -6*E*I/(L^2), 0, 12*E*I/(L^3), -6*E*I/(L^2);
                  0, 6*E*I/(L^2), 2*E*I/L, 0, -6*E*I/(L^2), 4*E*I/L];
end

%Para pasar cada miembro del marco plano al sistema de coordenadas global
%se construye primero la matriz de rotación y luego se efectúa la
%transformación.

for ii = 1:nm
    ROT(:, :, ii)=[Cos(ii), -Sen(ii), 0, 0, 0, 0;
                  Sen(ii), Cos(ii), 0, 0, 0, 0;
                  0, 0, 1, 0, 0, 0;

```

```

    0,0,0,Cos(ii),-Sen(ii),0;
    0,0,0,Sen(ii),Cos(ii),0;
    0,0,0,0,0,1];
KMG(:, :, ii)=ROT(:, :, ii)*KM(:, :, ii)*ROT(:, :, ii)';
end

```

*Ahora se construye el vector de fuerzas equivalentes en los nodos para
 *cada miembro denotado por FM, referido al sistema de coordenadas locales.
 *Luego se pasa FM a coordenadas globales quedando FMG.

```

for ii = 1:nn
  FM(:, ii)=[(-MM(ii,6)*(1-MM(ii,7))-MM(ii,12)*Lon(ii)/2);
    (-MM(ii,8)*((1-MM(ii,9))^2)*(1+2*MM(ii,9))+...
    6*MM(ii,10)*MM(ii,11)*(1-MM(ii,11))/Lon(ii)-...
    MM(ii,13)*Lon(ii)/2 -3*MM(ii,14)*Lon(ii)/20);
    (-MM(ii,8)*MM(ii,9)*((1-MM(ii,9))^2)*Lon(ii)+...
    MM(ii,10)*(1-MM(ii,11))*(3*MM(ii,11))-1)-...
    MM(ii,13)*(Lon(ii)^2)/12-MM(ii,14)*(Lon(ii)^2)/30);
    (-MM(ii,6)*MM(ii,7)-MM(ii,12)*Lon(ii)/2);
    (-MM(ii,8)*((MM(ii,9)^2)*(3-2*MM(ii,9)))-...
    6*MM(ii,10)*MM(ii,11)*(1-MM(ii,11))/Lon(ii)-...
    MM(ii,13)*Lon(ii)/2-7*MM(ii,14)*Lon(ii)/20);
    (MM(ii,8)*(MM(ii,9)^2)*(1-MM(ii,9))*Lon(ii)+...
    MM(ii,10)*MM(ii,11)*(2-3*MM(ii,11))+MM(ii,13)*(Lon(ii)^2)/12+...
    MM(ii,14)*(Lon(ii)^2)/20)];
  FMG(:, ii)=ROT(:, :, ii)*FM(:, ii);
end

```

*En el siguiente paso se ensambla la matriz de rigidez total KT.

```

KT = zeros(3*nn,3*nn);
for ii = 1:nn
  for ji = 1:2
    for jj = 1:2
      for ki = 1:3
        for kj = 1:3
          KT(GL(MM(ii,ji),ki),GL(MM(ii,jj),kj))=...
            KT(GL(MM(ii,ji),ki),GL(MM(ii,jj),kj))+...
            KMG(ki+3*(ji-1),kj+3*(jj-1),ii);
        end
      end
    end
  end
end
end

```

*Ahora se ensambla el vector de fuerza total denotado por FT

```

for ii = 1:3*nn
  FT(ii)=0;
end

for ii = 1:nn
  for ji = 1:2
    for ki = 1:3
      FT(GL(MM(ii,ji),ki))=FT(GL(MM(ii,ji),ki))+FMG(ki+3*(ji-1),ii);
    end
  end
end

for ii = 1:nn
  for jj=1:3

```

```

        TEMP(GL(ii,jj))=MN(ii,5+jj);
    end
end

for ii=1:3*nn
    FT(ii) = FT(ii)-TEMP(ii);
end

%Para completitud se definen y calculan las matrices K,KRL y
%los vectores F y FR.
%K es la submatría de KT donde solamente incluye grados de libertad no
%restringidos.
%KRL es la submatriz de KT que incluye gados de libertad restringidos con
%grados de libertad no restringidos.
%F es el subvector de FT que solamente incluye grados de libertad no
%restringidos.
%FR es el subvector de F que incluye los grados de libertad restrigidos.

for ii = 1:glnr
    for jj = 1:glnr
        K(ii,jj)=KT(ii,jj);
    end
end

for ii = 1:glnr
    Ft(ii)=FT(ii);
end

F =Ft';

for ii = 1:glr
    for jj = 1:glnr
        KRL(ii,jj)=KT(ii+glnr,jj);
    end
end

for ii = 1:glr
    FRT(ii) = FT(ii+glnr);
end

FR=FRT';

%A continuación se encuentran los desplazamientos que forman el vector D,
%resolviendo el sistema de ecuaciones KD=F.

D=linsolve(K,-F);

%Por medio de una simple multiplicación matricial se hallan las reacciones
%en los apoyos, dado por el vector R=KRL*D+FR.

R=(KRL*D)+FR;

%Queda todavía que calcular las fuerzas en los extremos de cada miembro,por
%lo tanto se tiene que pasar los desplazamientos a coordenadas locales.

for ii=1:3*nn
    if ii<=glnr
        DT(ii)=D(ii);
    else
        DT(ii)=0;
    end
end
end

```

```

*En la siguiente parte del programa se calculan las fuerzas en los extremos
%de los miembros y se despliegan los resultados de una forma fácil de
%reconocer
disp(' ');
disp(' ');
disp('-----');
disp('          FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS ');
disp('-----');
disp(' ');
for ii=1:nm
    for ji=1:2
        for ki=1:3
            jj=ki+(3*(ji-1));
            DMG(jj)=DT(GL(MM(ii,ji),ki));
        end
    end
    DM(:,ii)=ROT(:,,ii)*DMG';
    RM(:,ii)=(KM(:,,ii)*DM(:,ii))+FM(:,ii);
    tituloi=['          Las fuerzas en el extremo inicial del miembro ',...
            num2str(ii),' son:'];
    aa=['          Axial = ',num2str(RM(1,ii)),';          Corte = ',...
        num2str(RM(2,ii)),';          Momento = ',num2str(RM(3,ii))];
    titulof=['          Las fuerzas en el extremo final del miembro ',...
            num2str(ii),' son:'];
    bb=['          Axial = ',num2str(RM(4,ii)),';          Corte = ',...
        num2str(RM(5,ii)),';          Momento = ',num2str(RM(6,ii))];
    disp(tituloi);
    disp(aa);
    disp(' ');
    disp(titulof);
    disp(bb);
    disp(' ');
    disp('-----');
    disp(' ');
end

*Ahora se despliegan los resultados de los desplazamientos en los nodos
%con respecto al sistema de coordenadas global. Se especifican el
%desplazamiento horizontal, vertical y el giro.
disp(' ');
disp(' ');
disp('-----');
disp('          DESPLAZAMIENTOS EN LOS NODOS ');
disp('-----');
disp(' ');
for ii=1:nn
    for jj=1:3
        kk = 3*(ii-1) + jj;
        if MN(ii,jj)==1
            DC(kk)=0;
        else
            DC(kk)=D(kk-RESTRICION(ii,jj));
        end
    end
    titulo=['          Los Desplazamientos y el Giro del NODO ',...
            num2str(ii),' son:'];
    a=['          Horizontal = ',num2str(DC(3*(ii-1)+1)),...
        '          Vertical = ',num2str(DC(3*(ii-1)+2)),...
        '          Giro = ',num2str(DC(3*(ii-1)+3)),' rad'];
    disp(titulo);
    disp(a);
    disp(' ');
    disp(' ');
end

```

```

end

%En esta parte se programa para presentar las reacciones en los apoyos. Los
%apoyos están identificados por el número de nodo donde se encuentran.
disp(' ');
disp(' ');
disp('-----');
disp('                      REACCIONES EN LOS APOYOS ');
disp('-----');
disp(' ');for ii = 1:nn
    for jj = 1:3
        if MN(ii,jj) == 1
            RA(ii,jj)=R(RESTRICCION(ii,jj));
        else
            RA(ii,jj)=0;
        end
    end
end
if MN(ii,1)+MN(ii,2)+MN(ii,3) >= 1
    tit=['                      Las Reacciones en el Apoyo ',...
        num2str(ii),' son:'];
    a=['          Horizontal = ',num2str(RA(ii,1)),...
        ' ;                      Vertical = ',num2str(RA(ii,2)),...
        ' ;                      Momento = ',num2str(RA(ii,3))];
    disp(tit);
    disp(a);
    disp(' ');
    disp(' ');
else
    end
end
end

%Este programa incluye una gráfica esquemática del marco
%rígido, mostrando el número de nodo y el número de miembro.
%A continuación se programa el ploteo del marco.

clf;
deltax=max(MN(:,4))-min(MN(:,4));
deltay=max(MN(:,5))-min(MN(:,5));
xmin=min(MN(:,4))-deltax/5;
ymin=min(MN(:,5))-deltay/5;
xmax=max(MN(:,4))+deltax/5;
ymax=max(MN(:,5))+deltay/5;
hold on

if regularidad == 1 | regularidad == 2
    factorx=1+(nl/20);
    factory=1+(np/20);
else
    factorx=1;
    factory=1;
end

for ii = 1:nn
    x=[MN(MM(ii,1),4),MN(MM(ii,2),4)];
    y=[MN(MM(ii,1),5),MN(MM(ii,2),5)];
    plot(x,y,'b-','linewidth',2.5)
    a=(MN(MM(ii,1),4)+MN(MM(ii,2),4))/2;
    b=(MN(MM(ii,1),5)+MN(MM(ii,2),5))/2;
    if MN(MM(ii,1),4) == MN(MM(ii,2),4)
        a=a-deltax/(20*factorx);
    elseif MN(MM(ii,1),5) == MN(MM(ii,2),5)
        b=b+deltay/(40*factory);
    else

```

```

    a=a-deltax/(30*factorx);
    b=b+deltay/(30*factory);
end

text(a,b,num2str(ii),'Color','blue');
if MN(MM(ii,1),1) == 1 & MN(MM(ii,1),2) == 1 &...
    MN(MM(ii,1),3) == 1
    dx=MN(MM(ii,2),4)-MN(MM(ii,1),4);
    dy=MN(MM(ii,2),5)-MN(MM(ii,1),5);
    aa=MN(MM(ii,1),4)-(deltax/(10*factorx))*(dy/Lon(ii));
    bb=aa+(deltax/(5*factorx))*(dy/Lon(ii));
    cc=MN(MM(ii,1),5)-(deltay/(10*factory))*(dx/Lon(ii));
    dd=cc+(deltay/(5*factory))*(dx/Lon(ii));
    x=[aa,bb];
    y=[dd,cc];
    plot(x,y,'k','linewidth',3);
elseif MN(MM(ii,2),1) == 1 & MN(MM(ii,2),2) == 1 &...
    MN(MM(ii,2),3) == 1
    dx=MN(MM(ii,2),4)-MN(MM(ii,1),4);
    dy=MN(MM(ii,2),5)-MN(MM(ii,1),5);
    aa=MN(MM(ii,2),4)-(deltax/(10*factorx))*(dy/Lon(ii));
    bb=aa+(deltax/(5*factorx))*(dy/Lon(ii));
    cc=MN(MM(ii,2),5)-(deltay/(10*factory))*(dx/Lon(ii));
    dd=cc+(deltay/(5*factory))*(dx/Lon(ii));
    x=[aa,bb];
    y=[dd,cc];
    plot(x,y,'k','linewidth',3);
end

end

axis([xmin xmax ymin ymax])
axis square
plot(MN(:,4),MN(:,5),'k.','MarkerSize',20.0);
title (' NUMERACION DE NODOS Y MIEMBROS DEL MARCO RIGIDO');
xlabel ('Coordenada X');
ylabel ('Coordenada Y');

for ii=1:nn
    text(MN(ii,4)+(deltax/(40*factorx)),MN(ii,5)-(deltay/(40*factory)),...
        num2str(ii));
end

hold off

```

Nótese que los comentarios están precedidos por el símbolo % y es en estas líneas que se explica y documenta el programa. En caso de que se pueda apreciar en colores los comentarios aparecen de color verde.

Vale la pena mencionar que el código que aparece escrito anteriormente hace el uso de los puntos suspensivos para cambiar de línea, lo cual fue necesario para que algunas líneas de programación cupieran en el formato de la tesis.

6.6 Ejemplos

En este trabajo de tesis se incluyen varios ejemplos para ilustrar el uso y las bondades del programa que se ha escrito. Debido a que el primer dato que pide el programa es la regularidad y que se tienen tres casos de regularidad se darán ejemplos para los tres casos. Los ejemplos se dividen de la siguiente manera: Cuatro para la regularidad 0, uno para la regularidad 1 y dos para regularidad 2. Los ejemplos de regularidad 0 y uno de regularidad 2 son extraídos de las referencias 4 y 5, respectivamente. De estos ejemplos se pudieron comparar respuestas dando excelentes resultados.

Ejemplo 1 (Regularidad 0):

Analice el marco plano mostrado en la figura 62, suponga que $P = 5$ kips, $L = 5$ pies y $E = 10,000$ kips/plg². El momento de inercia y el área de la sección transversal de los miembros 1 y 3 son 80 plg⁴ y 4 plg², respectivamente, y para el miembro 2 son 150 plg⁴ y 5 plg², respectivamente.

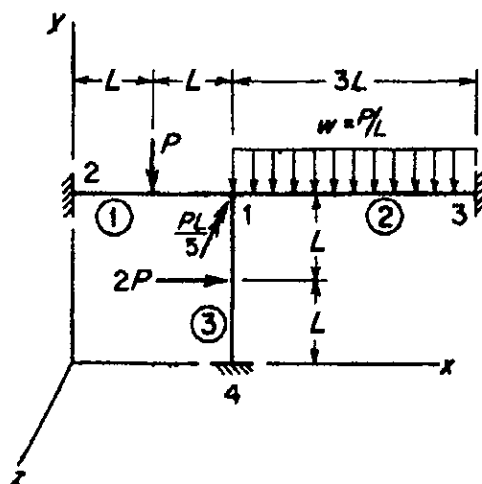


Figura 62. Ejemplo 1.

Solución:

Lo primero que se hace es construir la matriz de nodos y la matriz de miembros, usando la ventana de comandos y la ventana de espacio de trabajo. Por ejemplo, la matriz de nodos en la ventana de comandos se vería de la siguiente manera:

$$\begin{array}{l}
 \text{MN} = \\
 \begin{array}{cccccccc}
 0 & 0 & 0 & 120 & 120 & 0 & 0 & -60 \\
 1 & 1 & 1 & 0 & 120 & 0 & 0 & 0 \\
 1 & 1 & 1 & 300 & 120 & 0 & 0 & 0
 \end{array}
 \end{array}$$

1 1 1 120 0 0 0 0

Ingrese nivel de regularidad : 0

Ingrese la matriz de nodos MN : MN

Ingrese la matriz de miembro MM : MM

Dando los siguientes resultados:

FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS

Las fuerzas en el extremo inicial del miembro 1 son:

Axial = -2.8367; Corte = 2.4152; Momento = 74.8694

Las fuerzas en el extremo final del miembro 1 son:

Axial = 2.8367; Corte = 2.5848; Momento = -85.0487

Las fuerzas en el extremo inicial del miembro 2 son:

Axial = 2.3639; Corte = 7.2027; Momento = 192.0483

Las fuerzas en el extremo final del miembro 2 son:

Axial = -2.3639; Corte = 7.7973; Momento = -245.554

Las fuerzas en el extremo inicial del miembro 3 son:

Axial = 9.7876; Corte = 4.7993; Momento = 142.9186

Las fuerzas en el extremo final del miembro 3 son:

Axial = -9.7876; Corte = 5.2007; Momento = -166.9996

DESPLAZAMIENTOS EN LOS NODOS

Los Desplazamientos y el Giro del NODO 1 son:

Horizontal = 0.0085102; Vertical = -0.029363; Giro = -0.00074386 rad

Los Desplazamientos y el Giro del NODO 2 son:

Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 3 son:
Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 4 son:
Horizontal = 0; Vertical = 0; Giro = 0 rad

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 2 son:
Horizontal = -2.8367; Vertical = 2.4152; Momento = 74.8694

Las Reacciones en el Apoyo 3 son:
Horizontal = -2.3639; Vertical = 7.7973; Momento = -245.554

Las Reacciones en el Apoyo 4 son:
Horizontal = -4.7993; Vertical = 9.7876; Momento = 142.9186

Además de los resultados anteriores el programa da la siguiente figura para identificar miembros y nodos.

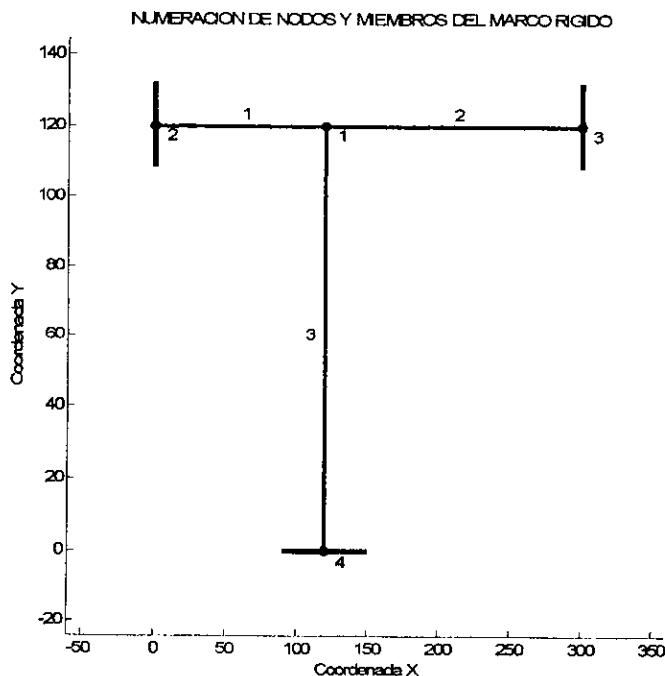


Figura 63. Figura que muestra el programa usando "axis square".

Ejemplo 2 (Regularidad 0):

El marco plano mostrado en la figura 64 debe ser analizado sobre la base de los siguientes datos numéricos: $E = 30,000$ kips/plg², $P = 400$ lb, $L = 30$ plg. Para los miembros 1 y 2, $I_z = 12$ plg⁴ y $A_x = 4$ plg². Para el miembro 3, $I_z = 14$ plg⁴ y $A_x = 6$ plg².

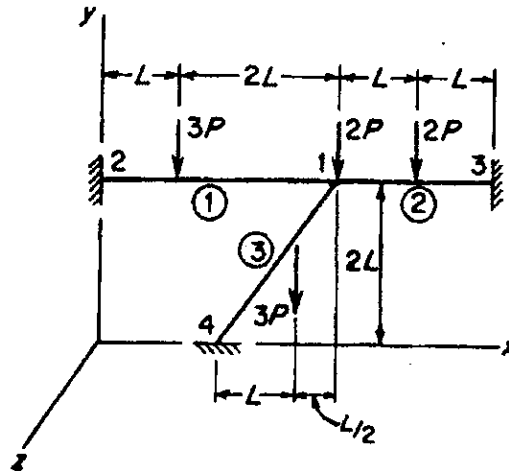


Figura 64. Ejemplo 2.

Solución:

Para este ejemplo la matriz de nodos es la siguiente:

$$\begin{matrix}
 MN = \\
 \begin{matrix}
 0 & 0 & 0 & 90.0000 & 60.0000 & 0 & -0.8000 & 0 \\
 1.0000 & 1.0000 & 1.0000 & 0 & 60.0000 & 0 & 0 & 0 \\
 1.0000 & 1.0000 & 1.0000 & 150.0000 & 60.0000 & 0 & 0 & 0 \\
 1.0000 & 1.0000 & 1.0000 & 45.0000 & 0 & 0 & 0 & 0
 \end{matrix}
 \end{matrix}$$

Ingrese nivel de regularidad :0

Ingrese la matriz de nodos MN :MN

Ingrese la matriz de miembro MM :MM

FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS

Las fuerzas en el extremo inicial del miembro 1 son:
 Axial = -0.67653; Corte = 0.94247; Momento = 17.774

Las fuerzas en el extremo final del miembro 1 son:
 Axial = 0.67653; Corte = 0.25753; Momento = -4.9518

Las fuerzas en el extremo inicial del miembro 2 son:
 Axial = 1.0148; Corte = 0.45809; Momento = 8.6984

Las fuerzas en el extremo final del miembro 2 son:
 Axial = -1.0148; Corte = 0.34191; Momento = -5.213

Las fuerzas en el extremo inicial del miembro 3 son:
 Axial = 3.1873; Corte = 0.27631; Momento = 6.4695

Las fuerzas en el extremo final del miembro 3 son:
 Axial = -2.2273; Corte = 0.44369; Momento = -3.7466

DESPLAZAMIENTOS EN LOS NODOS

Los Desplazamientos y el Giro del NODO 1 son:
 Horizontal = 0.0005074; Vertical = -0.0018739; Giro = 0.00015928 rad

Los Desplazamientos y el Giro del NODO 2 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 3 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 4 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 2 son:
 Horizontal = -0.67653; Vertical = 0.94247; Momento = 17.774

Las Reacciones en el Apoyo 3 son:
 Horizontal = -1.0148; Vertical = 0.34191; Momento = -5.213

Las Reacciones en el Apoyo 4 son:
 Horizontal = 1.6913; Vertical = 2.7156; Momento = 6.4695

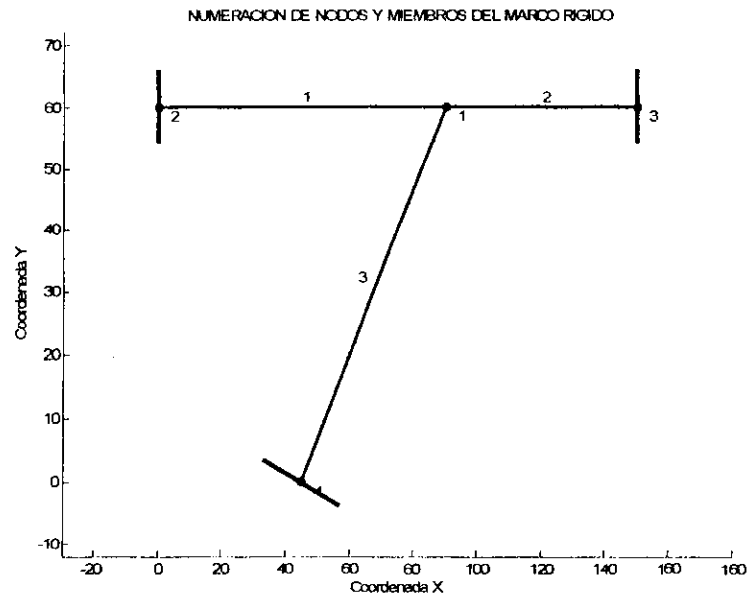


Figura 65. Figura que muestra el programa para identificar miembros y nodos sin usar "axis square".

Ejemplo 3 (Regularidad 0):

Analice el marco plano mostrado en la figura 66 para los efectos de las cargas mostradas. Suponga que todos los miembros tienen $E = 10,500 \text{ kips/plg}^2$, $I_z = 26 \text{ plg}^4$, $A_x = 8 \text{ plg}^2$ y también suponga que $L = 60 \text{ plg}$ y $P = 2 \text{ kips}$.

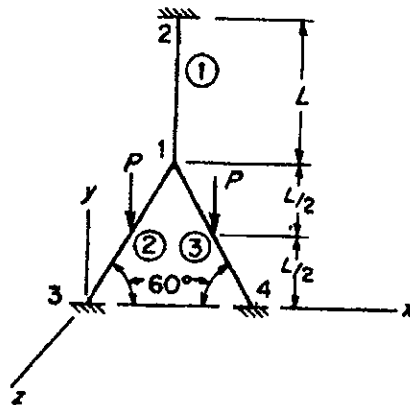


Figura 66. Ejemplo 3.

Solución:

Ingrese nivel de regularidad :0

Ingrese la matriz de nodos MN :MN

Ingrese la matriz de miembro MM :MM

FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS

Las fuerzas en el extremo inicial del miembro 1 son:

Axial = -0.86861; Corte = -3.0965e-007; Momento = -9.5328e-006

Las fuerzas en el extremo final del miembro 1 son:

Axial = 0.86861; Corte = 3.0965e-007; Momento = -9.0463e-006

Las fuerzas en el extremo inicial del miembro 2 son:

Axial = 1.5175; Corte = 0.50306; Momento = 8.7661

Las fuerzas en el extremo final del miembro 2 son:

Axial = 0.21458; Corte = 0.49694; Momento = -8.5544

Las fuerzas en el extremo inicial del miembro 3 son:

Axial = -0.21458; Corte = 0.49694; Momento = 8.5544

Las fuerzas en el extremo final del miembro 3 son:

Axial = -1.5175; Corte = 0.50306; Momento = -8.7661

DESPLAZAMIENTOS EN LOS NODOS

Los Desplazamientos y el Giro del NODO 1 son:

Horizontal = 1.898e-008; Vertical = -0.00062044; Giro = -5.3464e-011 rad

Los Desplazamientos y el Giro del NODO 2 son:

Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 3 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 4 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 2 son:
 Horizontal = $-5.4347e-007$; Vertical = 0.86861; Momento = $-9.0463e-006$

Las Reacciones en el Apoyo 3 son:
 Horizontal = 0.32308; Vertical = 1.5657; Momento = 8.7661

Las Reacciones en el Apoyo 4 son:
 Horizontal = -0.3231 ; Vertical = 1.5657; Momento = -8.7661

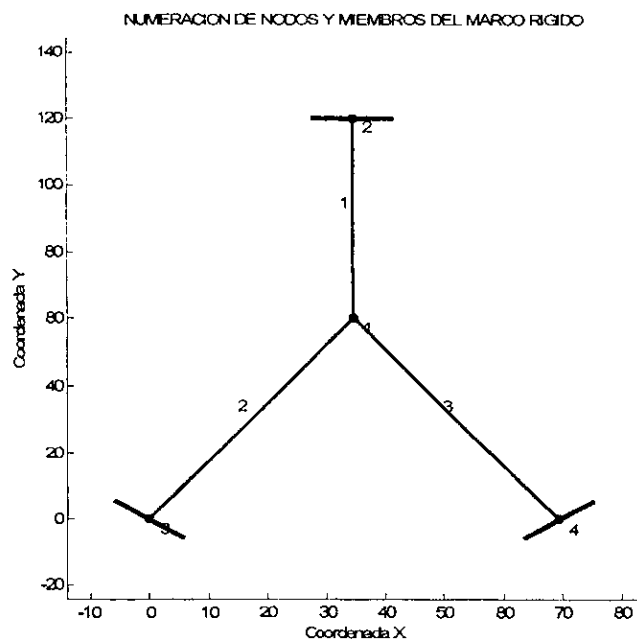


Figura 67. Figura del ejemplo 3.

Ejemplo 4 (Regularidad 2):

Determinése los desplazamientos y rotaciones de los nodos en el portal mostrado en la figura 68. Así como también las reacciones en los apoyos y las fuerzas en los extremos de los miembros. Utilizar los siguientes datos: $E = 30 \times 10^6$ psi, $A = 6.8$ in² e $I = 65$ in⁴.

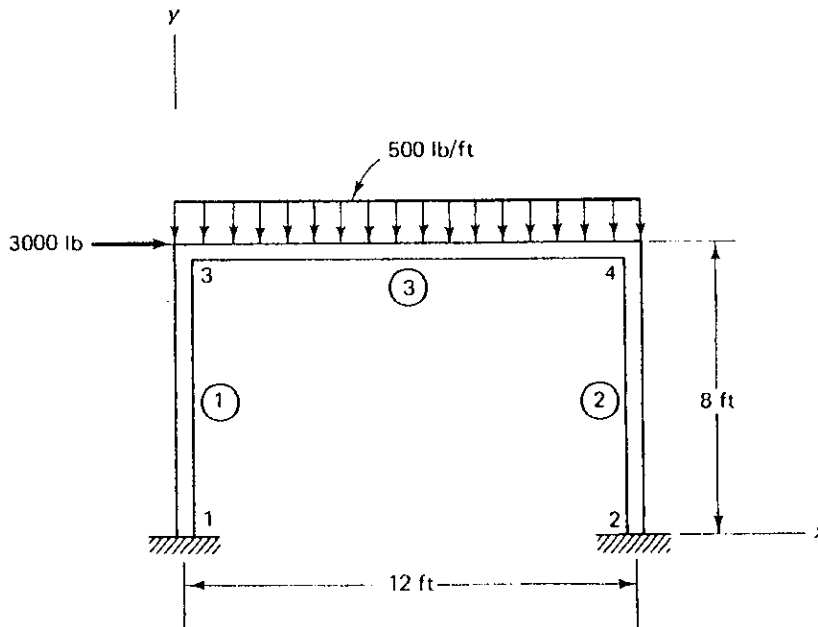


Figura 68. Ejemplo 4.

Solución:

Este portal es un marco rígido de un nivel con una luz, se puede usar regularidad 1 o regularidad 2. En este caso utilizaremos regularidad 2.

El ingreso de datos queda de la siguiente manera:

Ingrese nivel de regularidad :2
 Ingrese número de pisos : 1
 Ingrese número de luces : 1
 Ingrese altura de piso : 96
 Ingrese longitud de luz : 144
 Ingrese el módulo de elasticidad : 30000000
 Ingrese el área de las vigas : 6.8
 Ingrese el área de las columnas : 6.8
 Ingrese la inercia de las vigas : 65
 Ingrese la inercia de las columnas : 65
 Ingrese fuerza horizontal en los pisos : 3000
 Ingrese carga distribuida sobre las vigas : -41.66667
 Ingrese carga puntual al centro de las vigas : 0

Los desplazamientos de los nodos, dado en pulgadas y radianes, quedan de la siguiente forma:

DESPLAZAMIENTOS EN LOS NODOS

Los Desplazamientos y el Giro del NODO 1 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 2 son:
 Horizontal = 0; Vertical = 0; Giro = 0 rad

Los Desplazamientos y el Giro del NODO 3 son:
 Horizontal = 0.091766; Vertical = -0.0010358; Giro = -0.0013874 rad

Los Desplazamientos y el Giro del NODO 4 son:
 Horizontal = 0.090119; Vertical = -0.0017877; Giro = -3.883e-005 rad

Las reacciones en los apoyos 1 y 2, dada en libras y lb*in, quedan de la siguiente forma:

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 1 son:
 Horizontal = -665.7828; Vertical = 2201.1786; Momento = 60138.5227

Las Reacciones en el Apoyo 2 son:
 Horizontal = -2334.2172; Vertical = 3798.8219; Momento = 112831.1616

Al correr el programa nos da las fuerzas en los extremos de los miembros de la siguiente forma:

FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS

Las fuerzas en el extremo inicial del miembro 1 son:

Axial = 2201.1786; Corte = 665.7828; Momento = 60138.5227

Las fuerzas en el extremo final del miembro 1 son:
Axial = -2201.1786; Corte = -665.7828; Momento = 3776.6266

Las fuerzas en el extremo inicial del miembro 2 son:
Axial = 3798.8219; Corte = 2334.2172; Momento = 112831.1616

Las fuerzas en el extremo final del miembro 2 son:
Axial = -3798.8219; Corte = -2334.2172; Momento = 111253.6891

Las fuerzas en el extremo inicial del miembro 3 son:
Axial = 2334.2172; Corte = 2201.1786; Momento = -3776.6266

Las fuerzas en el extremo final del miembro 3 son:
Axial = -2334.2172; Corte = 3798.8219; Momento = -111253.6891

Y el programa muestra la siguiente figura para identificar nodos y miembros:

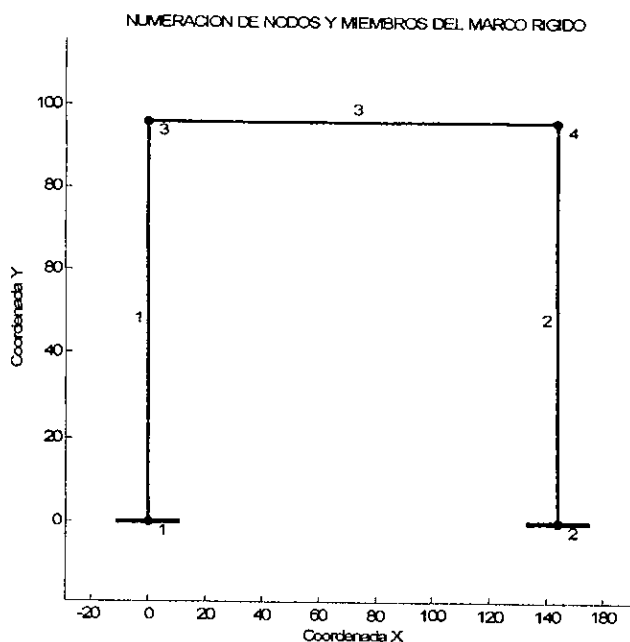


Figura 69. Figura del ejemplo 4.

Para la deflexión en el centro de la viga BC, simplemente se busca en el nodo 3.

Los Desplazamientos y el Giro del NODO 3 son:

Horizontal = $3.7682e-016$; Vertical = -0.41665 ; Giro = $2.6792e-018$ rad

Nótese que el desplazamiento horizontal y el giro son virtualmente cero. La deflexión es el desplazamiento vertical cuyo resultado es -0.41665 pulgadas.

Para las reacciones en los apoyos se obtiene directamente de las respuestas que da el programa.

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 1 son:

Horizontal = 8417.6618 ; Vertical = 12000 ; Momento = -191038.6002

Las Reacciones en el Apoyo 5 son:

Horizontal = -8417.6618 ; Vertical = 12000 ; Momento = 191038.6002

Para identificar los apoyos se ve la figura 71, que proporciona el programa.

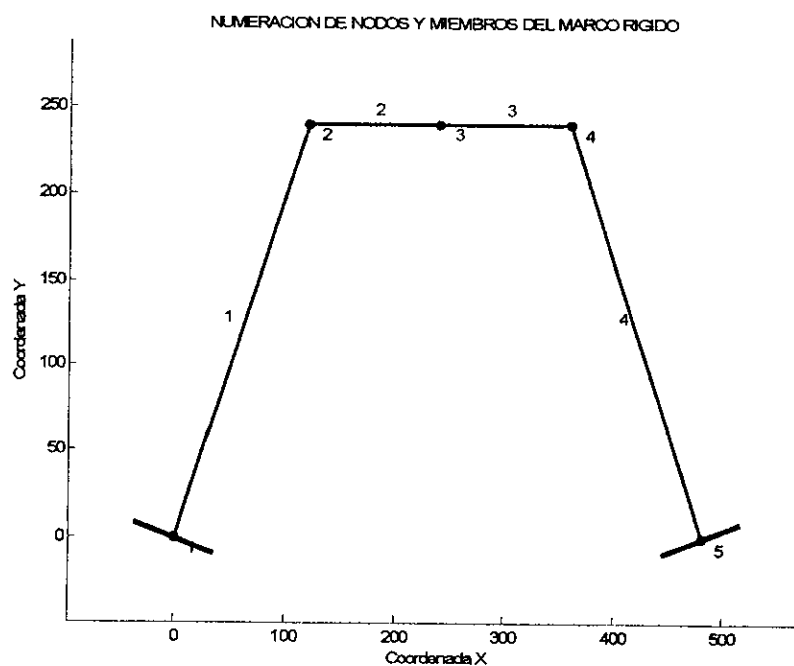


Figura 71. Figura del ejemplo 5.

De la figura 71 los nodos B y C corresponden a los números 2 y 4 respectivamente, cuyos desplazamientos son:

Los Desplazamientos y el Giro del NODO 2 son:
Horizontal = 0.0044894; Vertical = -0.01191; Giro = -0.0029056 rad

Los Desplazamientos y el Giro del NODO 4 son:
Horizontal = -0.0044894; Vertical = -0.01191; Giro = 0.0029056 rad

Para ilustrar el uso del nivel de regularidad 1 se presenta el siguiente ejemplo.

Ejemplo 6 (Regularidad 1):

Considere un marco de seis niveles y cinco luces. El primer nivel tiene una altura de 5 metros, el segundo nivel tiene una altura de 4 metros y los demás niveles tienen 3 metros de altura. La luz central tiene 6 metros de largo, mientras que las otras luces tienen 4.5 metros de largo. La carga sobre la estructura está uniformemente distribuida sobre las vigas y tiene un valor de 8,000 N/m, en las vigas cortas y 12,000 en las vigas largas. Para las columnas considérese un área transversal 0.6 m^2 y una inercia de 0.3 m^4 para los dos primeros pisos. Para los otros pisos tome un área de 0.45 m^2 y 0.2 m^4 . Para las vigas de luces cortas se tiene un área transversal de 0.4 m^2 y un inercia de 0.25 m^4 ; y para las vigas de luz larga se tiene un área transversal de 0.5 m^2 y una inercia de 0.3 m^4 . Para módulo de elasticidad utilizar $2 \times 10^6 \text{ Pa}$.

Solución:

Este es un ejemplo para usar nivel de regularidad 1 y los datos deben ingresarse de la siguiente manera:

Ingrese nivel de regularidad :1
 Ingrese vector de altura de pisos : [5,4,3,3,3]
 Ingrese vector de longitud de luces : [4.5,4.5,6,4.5,4.5]
 Ingrese módulo de elasticidad : 2000000
 Ingrese vector de area de vigas : [0.4,0.4,0.5,0.4,0.4]
 Ingrese vector de area de columnas : [0.6,0.6,0.45,0.45,0.45,0.45]
 Ingrese vector de inercia de vigas : [0.25,0.25,0.3,0.25,0.25]
 Ingrese vector de inercia de columnas : [0.3,0.3,0.2,0.2,0.2,0.2]
 Ingrese vector de fuerzas horizontales : [0,0,0,0,0]
 Ingrese vector de cargas distribuidas sobre las vigas : [-8000,-8000,-12000,-8000,-8000]
 Ingrese vector de cargas puntuales al centro de las vigas : [0,0,0,0,0]

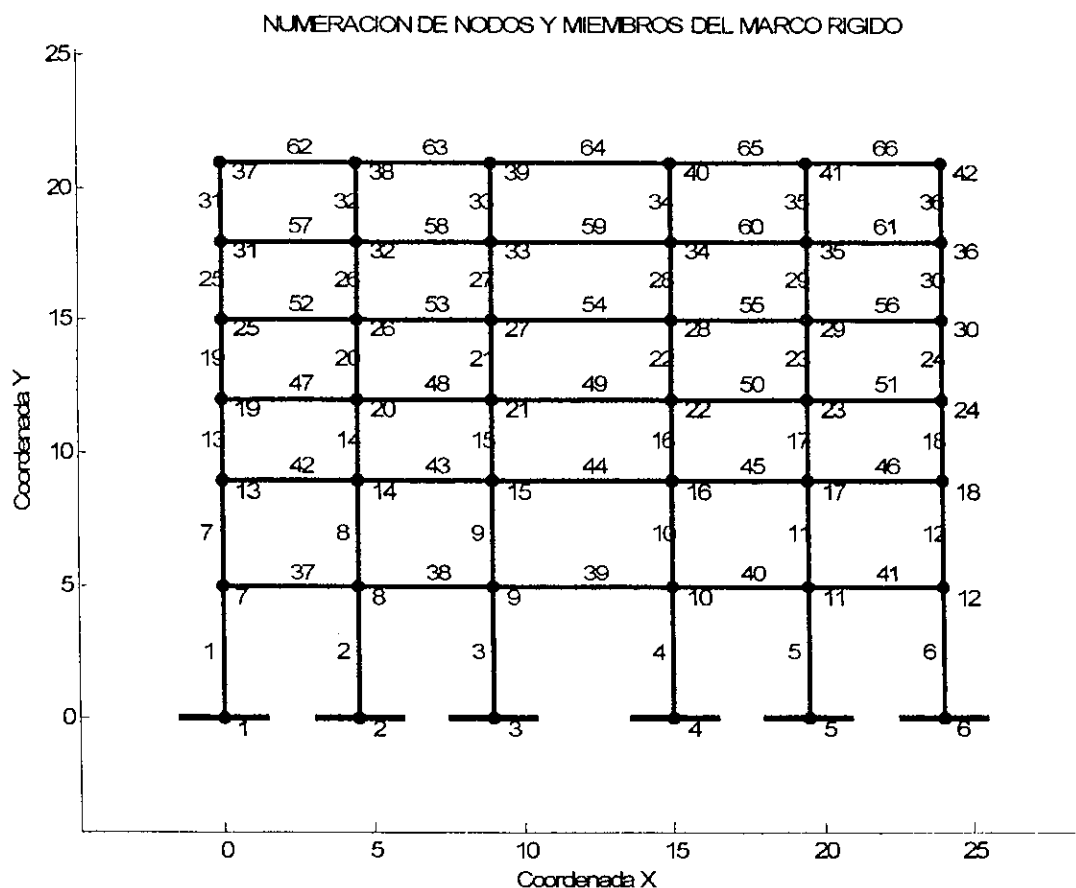


Figura 72. Marco de varios niveles y luces.

Como puede observarse el ingreso de datos es relativamente simple y no requiere mucho tiempo. Si se trabaja en MATLAB es fácil editar la matriz de nodos y la matriz de miembros para agregar y/o quitar todo tipo de carga sobre los miembros, así como también cambiar áreas transversales, inercias y/o módulos de elasticidad. Por lo anterior, el marco ingresado se puede analizar en un sin número de situaciones de cargas. En la figura 72 se muestra el marco identificando miembros y nodos, esta figura la proporciona directamente el mismo programa. Debido a la gran cantidad de nodos y miembros no se presentan todos los resultados, pues ocuparían mucho espacio, sin embargo mostramos las reacciones en los apoyos 1, 2, 3, 4 y 5. Los resultados están en Newtons y N*m.

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 1 son:
 Horizontal = 6677.7372; Vertical = 172201.8132; Momento = -15673.1542

Las Reacciones en el Apoyo 2 son:

Horizontal = 6402.3845; Vertical = 215746.8933; Momento = -14475.2616

Las Reacciones en el Apoyo 3 son:
Horizontal = 5000.0645; Vertical = 260051.2935; Momento = -10155.7675

Las Reacciones en el Apoyo 4 son:
Horizontal = -5000.0645; Vertical = 260051.2935; Momento = 10155.7675

Las Reacciones en el Apoyo 5 son:
Horizontal = -6402.3845; Vertical = 215746.8933; Momento = 14475.2616

Las Reacciones en el Apoyo 6 son:
Horizontal = -6677.7372; Vertical = 172201.8132; Momento = 15673.1542

Es fácil buscar cuales son las fuerzas en los extremos de los miembros, por ejemplo, si requerimos la viga central del primer piso y la columna inferior izquierda de la estructura se localizan en la figura 72 como los miembros número 39 y número 1 respectivamente, y se buscan en los resultados que proporciona el programa mostrados a continuación.

FUERZAS EN LOS EXTREMOS DE LOS MIEMBROS

Las fuerzas en el extremo inicial del miembro 1 son:
Axial = 172201.8132; Corte = -6677.7372; Momento = -15673.1542

Las fuerzas en el extremo final del miembro 1 son:
Axial = -172201.8132; Corte = 6677.7372; Momento = -17715.5317

Las fuerzas en el extremo inicial del miembro 39 son:
Axial = -12655.0469; Corte = 36000; Momento = 32092.6773

Las fuerzas en el extremo final del miembro 39 son:
Axial = 12655.0469; Corte = 36000; Momento = -32092.6773

Ejemplo 7 (Regularidad 2):

Para mostrar la utilidad de este programa en marcos regulares, se considera un marco de 4 niveles y 3 luces. Donde la altura de los pisos es 3.5 m y las luces de 5 m. Para área de los miembros se usa 0.8 m², inercia de los miembros se usa 0.4 m⁴ y de módulo de

elasticidad 2×10^6 Pa. Para cargas se considera únicamente las cargas distribuidas sobre las vigas con un valor de 5,000 N/m.

Solución:

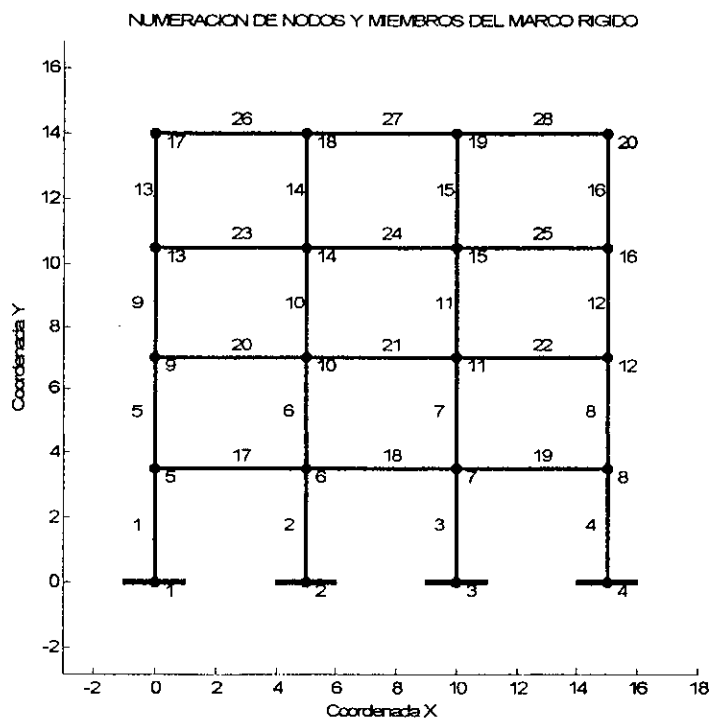


Figura 73. Ejemplo 7.

La figura 73 muestra el marco con la numeración de nodos y miembros. El programa nos da los desplazamientos de todos los nodos, las fuerzas en los extremos de todos los miembros y las reacciones en los apoyos.

Para este marco los datos se deben de ingresar de la siguiente forma:

Ingrese nivel de regularidad : 2
 Ingrese número de pisos : 4
 Ingrese número de luces : 3
 Ingrese altura de piso : 3.5
 Ingrese longitud de luz : 5
 Ingrese el módulo de elasticidad : 2000000
 Ingrese el área de las vigas : 0.8
 Ingrese el área de las columnas : 0.8
 Ingrese la inercia de las vigas : 0.4
 Ingrese la inercia de las columnas : 0.4
 Ingrese fuerza horizontal en los pisos : 0
 Ingrese carga distribuida sobre las vigas : -5000

Ingrese carga puntual al centro de las vigas : 0

Obsérvese la sencillez y rapidez en el ingreso de los datos para este tipo de marco, el programa con estos datos se hace cargo de la construcción de la matriz de miembros y la matriz de nodos. Se presentan a continuación las reacciones en los apoyos que da el programa.

REACCIONES EN LOS APOYOS

Las Reacciones en el Apoyo 1 son:
Horizontal = 3750.6111; Vertical = 63699.1506; Momento = -5381.3015

Las Reacciones en el Apoyo 2 son:
Horizontal = 1417.523; Vertical = 86300.8494; Momento = -2098.7472

Las Reacciones en el Apoyo 3 son:
Horizontal = -1417.523; Vertical = 86300.8494; Momento = 2098.7472

Las Reacciones en el Apoyo 4 son:
Horizontal = -3750.6111; Vertical = 63699.1506; Momento = 5381.3015

VII. CONCLUSIONES Y RECOMENDACIONES

El método de rigidez es un método completo para analizar marcos rígidos. El planteamiento es sencillo y fácilmente se ensambla el sistema de ecuaciones. Si se tiene la disponibilidad de una computadora se recomienda usar el método de rigidez.

El método de flexibilidad tiene el inconveniente de tener que elegir las redundantes. La elección de las redundantes requiere de mucho criterio, lo que hace extremadamente difícil tener un algoritmo sencillo para iniciar el método de flexibilidad.

El método de rigidez se puede atacar sistemáticamente sin necesidad de tener que aplicar criterios desde el inicio. Esto hace que sea fácil desarrollar un algoritmo para este método. Se recomienda usar el método de rigidez sobre el método de flexibilidad.

En el método de rigidez es muy sencillo especificar las condiciones de frontera. Ya que basta con un cero o con un uno para determinar si el apoyo está restringido o si está libre.

Otro aspecto positivo del método de rigidez es que los datos necesarios para que se realicen los cálculos se pueden ingresar de una manera simple, resumidos en dos matrices. En este trabajo de tesis se les llamaron matriz de nodos y matriz de miembros.

Tanto el método de flexibilidad como el método de rigidez se formulan en forma matricial. Por ejemplo, se tiene la matriz de flexibilidad, la matriz de rigidez, la matriz de rotación, el vector de desplazamientos, etc. MATLAB hace uso de las matrices de una forma extensiva y sencilla. Por lo tanto, MATLAB se ajusta muy bien para programar el método de rigidez.

MATLAB permite la edición de las matrices que se ingresan en el programa de una manera sencilla. Así que el usuario sobre una misma estructura puede probar con facilidad diferentes sistemas de cargas.

El hecho de que MATLAB utilice a las variables como matrices facilita la programación, por que no hay que programar la multiplicación de matrices, la inversa de matrices, la transpuesta de matrices y no hay necesidad de programar la solución de sistema de ecuaciones lineales. Existen comandos simples para realizar todas éstas y otras operaciones matriciales.

La velocidad de cálculo de las computadoras personales es lo suficientemente alta como para que se puedan resolver sistemas gigantescos de ecuaciones lineales en pocos segundos. Se consideran sistemas gigantescos de ecuaciones a los que tienen varias miles de incógnitas. Con esta capacidad se pueden analizar marcos rígidos de muchos niveles y muchas luces.

MATLAB presenta varias ventajas sobre otros compiladores conocidos. Entre ellas están el uso sistemático de las matrices. Toda variable es una matriz. En MATLAB no es necesario declarar variables, lo cual facilita bastante la programación ya que se puede ingresar una variable de uso temporal en cualquier momento. No es necesario dimensionar las matrices. Usa doble precisión sin tener que especificarla. El entorno de MATLAB es muy amigable.

Se recomienda a las personas que quieran analizar estructuras que utilicen MATLAB como una herramienta para realizar sus cálculos de una forma sencilla y rápida, obteniendo resultados satisfactorios.

Los resultados que se obtuvieron en este trabajo de tesis usando el método de rigidez y MATLAB son totalmente satisfactorios. Dando resultados muy precisos y presentándolos de una manera muy clara, a pesar de que no se está usando MATLAB en toda su potencia y tampoco se están aprovechando todas las bondades del método de rigidez.

Finalmente, se puede concluir que el uso combinado del método de rigidez con MATLAB da resultados muy satisfactorios en el análisis de marcos rígidos. Y la principal recomendación de esta tesis es precisamente que se use para el análisis estructural de marcos rígidos y otros tipos de estructuras el método de rigidez programado en MATLAB.

VIII. BIBLIOGRAFÍA Y REFERENCIAS

- 1) Chandrupatla, Tirupathi y Ashok D. Belegundu. 1999. *Introduction to Finite Elements in Engineering*. 2ª ed. New Jersey, Prentice-Hall, Inc. 462 págs.
- 2) Chapman, Stephen. 2002. *MATLAB Programming for Engineers*. 2ª ed. New York, BROOKS/COLE, Thomson Learning, Inc. 540 págs.
- 3) Felton, Lewis y Richard B. Nelson. 1997. *Matrix Structural Analysis*. New York, John Wiley & Sons, Inc. 700 págs.
- 4) Gere, James y William Weaver, Jr. 1965. *Analysis of Framed Structures*. Princeton, N.J., D. Van Nostrand Company, Inc. 535 págs.
- 5) Laursen, Harold I. 1969. *Structural Analysis*. International Student Edition, Tokyo, McGraw-Hill Hogakusha, Ltd. 486 págs.
- 6) Luthe, Rodolfo. 1971. *Análisis Estructural*. 1ª ed. México, D.F., Representaciones y servicios de ingeniería, S.A. 681 págs.
- 7) Pérez, César. 2002. *MATLAB y sus Aplicaciones en las Ciencias y la Ingeniería*. España, Prentice-Hall, Pearson Educación, S.A. 520 págs.
- 8) Tuma, Jan y R. K. Munshi. 1971. *Advanced Structural Analysis*. U.S.A., McGraw-Hill Book Company, Inc. 277 págs.

IX. GLOSARIO DE TÉRMINOS

- A = Área transversal de un miembro prismático.
 D = Vector de desplazamiento.
 E = Módulo de elasticidad.
 F = Matriz de flexibilidad o Vector de fuerza.
 F_M = Vector de fuerza de miembro.
 F_T = Vector de fuerza total.
 F_R = Vector de fuerza para los grados de libertad restringidos.
 f_{ij} = Coeficientes de influencia de flexibilidad.
 f_x = Carga horizontal puntual sobre un nodo.
 f_y = Carga vertical puntual sobre un nodo.
 I = Momento de inercia de un miembro prismático.
 K = Matriz de rigidez.
 K_M = Matriz de rigidez de miembro o elemental.
 K_{LR} = Matriz de rigidez de grados de libertad no restringidos a restringidos.
 K_{RL} = Matriz de rigidez de grados de libertad restringidos a no restringidos.
 K_{RR} = Matriz de rigidez de grados de libertad restringidos a restringidos.
 K_T = Matriz de rigidez total.
 k_{ij} = Coeficientes de influencia de rigidez.
 L = Longitud de un miembro prismático.
 M_x = Momento flexionante sobre un miembro prismático.
 m_z = Momento flexionante puntual sobre un miembro prismático o sobre un nodo.
 MM = Matriz de miembros.
 MN = Matriz de nodos.
 P = Vector de carga o fuerzas.
 P_x = Fuerza axial en un miembro prismático.
 p_x = Carga axial puntual sobre un miembro prismático.
 p_y = Carga perpendicular puntual sobre un miembro prismático.
 Q = Vector de cargas o de fuerzas.
 R = Reacciones en los apoyos.

$r_{(\theta)}$ = Matriz de rotación.

$R_{(\theta)}$ = Matriz de rotación total.

W_E = Trabajo real externo.

W_I = Trabajo interno o energía de deformación.

W_R = Trabajo realizado por las reacciones de los apoyos.

w_x = Carga axial uniformemente distribuida sobre un miembro prismático.

w_y = Carga perpendicular uniformemente distribuida sobre un miembro prismático.

wl = Carga perpendicular linealmente distribuida sobre un miembro prismático.

δW_E = Trabajo virtual externo.

δW_I = Trabajo virtual interno.

γ = Deformación transversal.

ε = Deformación axial.

σ = Esfuerzo axial.

τ = Esfuerzo de corte.

ν = Módulo de Poisson.



Imprenta "GORA"

25 Av. 25-71, Zona 5

Telefax: 2335-5733 - 5218-7292