

DESARROLLO DE UNA HERRAMIENTA COMPUTACIONAL PARA EL ANÁLISIS  
DE FLEXIBILIDAD EN SISTEMAS DE TUBERÍAS.

GERSON FABIAN REY NIÑO  
RICARDO ANDRES GARCIA VILLAMIZAR

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍA MECÁNICA  
BUCARAMANGA

2020

DESARROLLO DE UNA HERRAMIENTA COMPUTACIONAL PARA EL ANÁLISIS  
DE FLEXIBILIDAD EN SISTEMAS DE TUBERÍAS.

GERSON FABIAN REY NIÑO  
RICARDO ANDRES GARCIA VILLAMIZAR

Trabajo de Grado para Optar el Título de Ingeniero Mecánico

DIRECTOR:  
MANUEL DE JESUS MARTINEZ  
PhD en ingeniería mecánica.

CODIRECTOR:  
WILLIAM PINTO HERNÁNDEZ.  
PhD en ingeniería mecánica.

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS  
ESCUELA DE INGENIERÍA MECÁNICA  
BUCARAMANGA

2020

## **DEDICATORIA**

A mis padres por el sacrificio y apoyo a lo largo de todos estos años de carrera.

A mi hermano por crecer al lado mío brindándome su compañía.

A mis amigos y familiares por sacarme una sonrisa en los momentos difíciles

A mi mejor amigo Sebastián Domínguez por esa excelente organización en el viaje a santa marta dándome un segundo aliento para culminar mi carrera universitaria.

**RICARDO ANDRES GARCIA VILLAMIZAR**

## **DEDICATORIA**

A mi padre por enseñarme el valor del trabajo humilde y apoyarme en la decisión de estudiar en esta prestigiosa universidad.

A mi madre por levantarme los ánimos cuando las cosas no salían como las esperaba.

A mi hermana Karen Rey por alegrarse de mis logros como si fueran suyos.

A mi hermano Alexander Rey por enseñarme el valor de la constancia.

A mis compañeros que me brindaban una comida o un lugar para dormir cuando las jornadas de estudio se alargaban.

A mi amada WAN por impulsarme en esta etapa final y resaltar mis capacidades para seguir adelante.

**GERSON FABIAN REY NIÑO**

## **AGRADECIMIENTOS**

Agradecemos al profesor William Pinto Hernandez y Manuel de Jesus Martínez por brindarnos todo el acompañamiento y orientación para el desarrollo de este proyecto, siempre de la mejor manera y disposición.

Agradecemos a todos los profesores de la escuela de ingeniería mecánica por todo el conocimiento brindado en el transcurso de la carrera.

De igual manera agradecemos al ingeniero Cesar Eduardo Gutierrez por asesorarnos en algunos temas de programación de la herramienta computacional. Y por último a nuestra alma mater la Universidad Industrial de Santander por brindarnos la oportunidad de formarnos como buenos profesionales.

## CONTENIDO

	Pág.
INTRODUCCIÓN .....	15
1. PLANTEAMIENTO DEL PROBLEMA.....	16
1.1 DESCRIPCIÓN DEL PROBLEMA .....	16
1.2 JUSTIFICACIÓN DEL PROBLEMA .....	17
2. OBJETIVOS.....	18
2.1 OBJETIVO GENERAL .....	18
2.2 OBJETIVOS ESPECÍFICOS.....	18
3. MARCO REFERENCIAL.....	19
3.1 GENERALIDADES .....	19
3.1.1 Consideraciones Geométricas. ....	19
3.2 SOLICITACIONES EN TUBOS POR PRESIÓN INTERNA .....	21
3.2.1 Tubos de Pared Delgada. ....	22
3.2.2 Tensiones Radiales y Tangenciales. ....	23
3.2.3 Esfuerzo Equivalente. ....	24
3.2.4 Tubos Rectos de Pared Gruesa.....	25
3.2.5 Estado General de los Esfuerzos.....	28
3.2.6 Solicitaciones por Flexión. ....	28
3.2.7 Solicitaciones por Torsión. ....	29
3.2.8 Solicitaciones por Temperatura .....	30
3.2.9 Dilatación Térmica de las Tuberías.....	32
3.3 CODOS.....	34
3.3.1 Deformación de los Codos Bajo Flexión. ....	35
3.4 SOPORTES .....	37

3.4.1 Soportes Flexibles. ....	37
3.4.2 Soportes Flexibles de Carga Constante. ....	38
3.4.3 Soportes Flexibles de Carga Variable.....	39
4. NORMA ASME B31.3 .....	41
4.1 METODOLOGÍA DE ANÁLISIS DE FLEXIBILIDAD SEGÚN EL CÓDIGO ASME B31.3. ....	41
4.1.1 Flexibilidad de las Tuberías. ....	41
4.1.2 Requisitos Especiales .....	42
4.1.3 Esfuerzos por Desplazamiento. ....	42
4.1.4 Deformaciones por Desplazamiento. ....	42
4.2 PROPIEDADES PARA EL ANALISIS DE FLEXIBILIDAD. ....	43
4.2.1 Esfuerzos de Flexibilidad. ....	44
4.2.2 Esfuerzo por Carga Sostenida. ....	45
4.3 LÍMITES DE ESFUERZOS OCASIONADOS POR CARGAS SOSTENIDAS Y DEFORMACIONES POR DESPLAZAMIENTO. ....	45
4.3.1 Esfuerzos por Cargas Sostenidas, SL. ....	45
4.3.2 Intervalo de Esfuerzo Permitido por Desplazamiento, SA. ....	46
5. CALFEM .....	47
6. DESARROLLO METODOLÓGICO.....	54
6.1 COMPORTAMIENTO BAJO CARGA SOSTENIDA.....	54
6.1.2 Comportamiento Bajo Carga Térmica.....	57
7. DESCRIPCION INTERFAZ.....	59
7.1 INTERFAZ GRAFICA PRINCIPAL.....	59
8. RESULTADOS.....	61
8.1 PRIMER CASO.....	61

8.2 SEGUNDO CASO.....	63
8.3 TERCER CASO .....	64
9. CONCLUSIONES .....	67
BIBLIOGRAFIA.....	68
ANEXOS.....	70

## LISTA DE ILUSTRACIONES

	Pág.
Ilustración 1. Sistema isométrico para esquema de tuberías en el espacio.....	20
Ilustración 2. Ubicación de la tubería en el sistema de coordenadas y la descripción de su trayectoria. ....	20
Ilustración 3. Cálculo de la tensión axial en tubo recto .....	22
Ilustración 4. Cálculo de la tensión tangencial en tubo recto de pared delgada. ...	23
Ilustración 5. Tensiones principales en la pared de tubos rectos.....	25
Ilustración 6. Tensiones en tubos de pared gruesa .....	25
Ilustración 7. Variación de las tensiones tangencial y radial en tubo de pared gruesa .....	26
Ilustración 8. Distribución de las tensiones en la pared de un tubo de régimen elástico.....	27
Ilustración 9. Distribución de las tensiones por flexión y la composición con las componentes axiales. ....	29
Ilustración 10. Tensiones cortantes en el tubo.....	29
Ilustración 11. El estado de las tensiones en un cubo elemental.....	30
Ilustración 12. Expansión de tubo recto. ....	31
Ilustración 13. Ubicación del sistema de coordenadas para analizar la expansión térmica. ....	32
Ilustración 14. Estudio de la dilatación térmica en el plano.....	32
Ilustración 15. Cálculo de la expansión térmica.....	33
Ilustración 16. Generación geométrica de un codo como superficie toroidal. ....	34
Ilustración 17. Deformación de los codos bajo flexión. ....	35
Ilustración 18. Soporte flexible de carga constante.....	39
Ilustración 19. Formas constructivas de instalación de soporte flexible de carga constante .....	39

Ilustración 20. Formas constructivas de instalación de soporte flexible de carga variable. ....	40
Ilustración 21. Grupo de funciones CALFEM. ....	47
Ilustración 22. Definición de la matriz de grados de libertad. ....	48
Ilustración 23. Matriz de grados de libertad. ....	48
Ilustración 24. Grados de libertad para un elemento tridimensional. ....	49
Ilustración 25. Matriz de rigidez para un elemento tridimensional. ....	50
Ilustración 26. Vector de carga para un elemento tridimensional. ....	51
Ilustración 27. Función de ensamblaje de las matrices elementales. ....	52
Ilustración 28. Grados de libertad en una viga simple. ....	54
Ilustración 29. Interfaz principal de la herramienta. ....	59

## LISTA DE TABLAS

	Pág.
Tabla 1. Reacciones de carga, caso 1.....	61
Tabla 2. Reacciones de expansión, caso 1. ....	62
Tabla 3. Desplazamiento por expansión, caso 1. ....	62
Tabla 4. Desplazamiento por carga, caso 1.....	62
Tabla 5. Esfuerzos por carga y expansión, caso 1. ....	62
Tabla 6. Reacciones de carga, casó 2.....	63
Tabla 7. Reacciones de expansión, casó 2. ....	63
Tabla 8. Desplazamiento por carga, casó 2.....	63
Tabla 9. Desplazamientos por expansión, casó 2.....	64
Tabla 10. Esfuerzos por carga y expansión, casó 2. ....	64
Tabla 11. Reacciones de carga, casó 3.....	65
Tabla 12. Reacciones de expansión, caso 3. ....	65
Tabla 13. Desplazamiento de expansión, caso 3.....	65
Tabla 14. Desplazamiento por carga, caso 3.....	66
Tabla 15. Esfuerzos de carga y expansión, caso 3.....	66

## LISTA DE ANEXOS

	Pág.
Anexos A. Resultados de esfuerzos interfaz .....	70
Anexo B. Resultados de Reacciones interfaz .....	71
Anexo C. Resultados de desplazamientos interfaz.....	72
Anexo D. Ingreso datos de material Interfaz.....	73
Anexo E. Ingreso de Geometría y condiciones de trabajo interfaz .....	74
Anexo F. Ingreso de coordenadas en el sistema interfaz .....	75
Anexo G. Código Principal.....	76
Anexo H. Función Graficar.....	99
Anexo I. Función Beam3ee.....	127
Anexo J. Función Solveqt .....	133
Anexo K. Función Extract .....	134
Anexo L. Función Beam3ss .....	135
Anexo M. Función Beam3et.....	139
Anexo N. Función Beam3sf .....	143

## RESUMEN

**TÍTULO:** DESARROLLO DE UNA HERRAMIENTA COMPUTACIONAL PARA EL ANÁLISIS DE FLEXIBILIDAD EN SISTEMAS DE TUBERÍAS\*.

**AUTORES:** RICARDO ANDRES GARCIA VILLAMIZAR, GERSON FABIAN REY NIÑO\*\*.

**PALABRAS CLAVE:** TUBERÍA, FLEXIBILIDAD, SISTEMAS, ANÁLISIS, ESFUERZOS, DEFORMACIONES, CALFEM, MATLAB.

### **DESCRIPCIÓN:**

El desarrollo de un análisis de flexibilidad es una de las etapas finales y más importantes en el diseño de un sistema de tuberías, su objetivo es obtener datos acerca de las deformaciones, desplazamientos y esfuerzos presentes en la tubería, para luego ser comparados con la norma aplicada y establecer soluciones adecuadas si se excede el límite admisible, actualmente los métodos manuales para el cálculo de flexibilidad resultan algo extenso y complejos, por lo que se han desarrollado software especializados para este tipo de análisis.

El uso de este tipo de programas se limita a compañías dedicadas a la explotación de hidrocarburos y otros sectores industriales, debido al elevado costo de estas herramientas su uso se dificulta en el ámbito académico, el proyecto presentado a continuación tiene como fin desarrollar una herramienta computacional para el análisis de flexibilidad en sistemas de tuberías, permitiendo así su uso en la universidad industrial de Santander y abriendo nuevas ramas de investigación en la escuela de ingeniería mecánica. El desarrollo del proyecto se hará mediante el software matemático MATLAB, del cual la universidad posee una licencia completa y su uso es frecuente en diferentes materias de análisis numérico, facilitando de este modo la implementación de esta herramienta.

---

\* Trabajo de grado

\*\* Facultad de ingenierías Físico-Mecánicas. Escuela de ingeniería mecánica. Director: Manuel de Jesús Martínez, PhD en ingeniería mecánica. Codirector: William Pinto Hernández, PhD en ingeniería mecánica.

## ABSTRACT

**TITLE:** DEVELOPMENT OF A COMPUTATIONAL TOOL FOR FLEXIBILITY ANALYSIS IN PIPE SYSTEMS\*.

**AUTHORS:** RICARDO ANDRES GARCIA VILLAMIZAR, GERSON FABIAN REY NIÑO\*\*.

**KEYWORDS:** PIPE, FLEXIBILITY, SYSTEMS, ANALYSIS, STRESSES, DISPLACEMENTS, CALFEM, MATLAB.

### **DESCRIPTION:**

The development of a flexibility analysis is one of the final and most important stages in the design piping system, its objective is to obtain data about the deformations, displacements and efforts present in the pipe, and then to be compared with the norm applied and establish appropriate solutions if the admissible limit is exceeded, currently manual methods for calculating flexibility are extensive and complex, so specialized software has been developed for this type of analysis.

The use of this type of program is limited to companies dedicated to the exploitation of hydrocarbons and other industrial sectors, due to the high cost of these tools, their use is difficult in the academic field, the project presented below is intended to develop a computational tool for the analysis of flexibility in piping systems, allowing its use in the industrial university of Santander and opening new branches of research in the school of mechanical engineering. The development of the project will be done using the mathematical software MATLAB, of which the university has a full license and its use is frequent in different subjects of numerical analysis, facilitating the implementation of this tool.

---

\* Degree work

\*\* Faculty of Physical-Mechanical Engineerings.school of mechanical engineering. Director: Manuel de Jesus Martinez, PhD in mechanical engineering.Co-director: William Pinto Hernandez, PhD PhD in mechanical engineering.

## INTRODUCCIÓN

La cadena industrial de hidrocarburos ha tenido un notable crecimiento en los últimos años, convirtiéndose en pieza clave para la economía mundial y el desarrollo económico de algunos países<sup>1</sup>. En Colombia este sector representa una fuente importante de los ingresos del país, por lo que es de suma importancia para esta industria contar con las herramientas de ingeniería necesarias para la implementación de nuevos proyectos que permitan evaluar el tratamiento industrial y transporte de hidrocarburos, siendo las tuberías uno de los elementos más importante del sistema.

La elaboración de un nuevo proyecto de tuberías requiere de una secuencia de pasos, donde es necesario tener en cuenta una serie de factores que son de gran importancia para obtener una solución tecnológica y económica, el análisis de flexibilidad es una de las etapas finales del proceso de diseño y es el factor más importante ya que este determina los esfuerzos, deformaciones y desplazamientos que se presenta en el sistema de acuerdo a las condiciones de trabajo planteadas, de este modo permitiendo obtener las zonas de criticidad del sistema y generar las soluciones correspondientes<sup>2</sup>.

Los análisis de flexibilidad en sistema de tuberías se pueden hacer de forma manual o por medio de herramientas computacionales, las cuales tienen una gran ventaja respecto a los métodos convencionales ya que son más precisas, brindando así un análisis más exacto. Este tipo de herramientas poseen licencias de gran costo, por lo que solo acceden a estas las grandes compañías de hidrocarburos, limitando así el uso académico de estas.

---

<sup>1</sup> FEDESARROLLO. (2014, diciembre 14). Evaluación de la contribución económica del sector de hidrocarburos colombiano frente a diversos escenarios de producción. Recuperado 6 de marzo de 2019, de <https://www.repository.fedesarrollo.org.co>

<sup>2</sup> RÁBEK, G. (1982). Instalaciones térmicas (1.ª ed.). Caracas, Venezuela: Universidad central de Venezuela.p59.

## **1. PLANTEAMIENTO DEL PROBLEMA**

### **1.1 DESCRIPCIÓN DEL PROBLEMA**

El análisis de flexibilidad en sistemas de tuberías es una de las etapas finales de diseño y la más importante, su objetivo es obtener datos acerca de las deformaciones, desplazamientos y esfuerzos presentes en la tubería para luego ser comparados con la norma aplicada y de este modo establecer si se encuentra dentro de los rangos admisibles de operación, este tipo de análisis son de vital importancia en los diferentes sectores industriales que requieren transporte de fluidos, la industria petrolera y del gas en especial debido a su gran crecimiento en la última década requieren de la ampliación de sus sistemas tuberías o del remplazo de sus líneas de tuberías ,por lo que exigen este tipo de análisis.

La manera más eficiente de realizar este análisis es mediante softwares especializados como CAEPIPE, CAESARII entre otros. Este tipo de herramientas facilitan la entrada y visualización de los datos necesarios para definir un sistema, junto con algunos métodos manuales permiten al usuario manejar fácilmente variables de entradas, además el trabajo de desarrollo de los modelos es rápido y genera gráficamente secciones y elementos críticos, brindando así una excelente idea de la flexibilidad del sistema mediante resultados confiables.

El acceso a este tipo de software es bastante limitado debido a los elevados costos de adquisición de sus licencias, normalmente solo tienen acceso las compañías privadas donde se manejan grandes montajes de tuberías, se puede afirmar que una licencia de este tipo de programa puede superar los \$30.000.000 millones de pesos colombianos, lo cual hace difícil su acceso para uso académico en países que su ubican por debajo de la producción mundial de hidrocarburos, como lo es el caso de Colombia.

## 1.2 JUSTIFICACIÓN DEL PROBLEMA

El costo elevado de las licencias de software para el análisis de flexibilidad en sistemas de tuberías limita su uso académico, en base a esta problemática se propone desarrollar una herramienta computacional que realice el cálculo de flexibilidad en sistemas de tuberías, es decir un programa que realice el cálculo de esfuerzos, deformaciones y desplazamientos y los compare con los esfuerzos admisible presentados por la norma ASME B31.3 y de este modo determinar las zonas críticas del sistema para generar las soluciones correspondientes.

La herramienta computacional se desarrollara mediante el software matemático Matlab, ya que este posee una interfaz de fácil manejo y comprensión para el usuario, por otro lado se cuenta con la licencia de este, por medio de los convenios que posee la Universidad Industrial de Santander con la entidad prestadora del servicio, facilitando el acceso a este software, reduciendo el costo económico del presente proyecto y facilitando el aprendizaje del programa a los demás estudiantes que ya poseen un conocimiento básico en Matlab proveniente del uso de este en otras áreas estudio de la facultad de ingeniería mecánica.

## **2. OBJETIVOS**

### **2.1 OBJETIVO GENERAL**

Desarrollar e implementar una herramienta computacional para el análisis de flexibilidad en sistemas de tuberías.

### **2.2 OBJETIVOS ESPECÍFICOS**

- Establecer los requerimientos para el análisis de flexibilidad en un sistema de tuberías y recopilar la bibliografía necesaria para desarrollar la herramienta computacional.
- Desarrollar la herramienta computacional mediante el lenguaje de programación del software matemático MATLAB.
- Desarrollar una interfaz de fácil manejo y comprensión que permita hacer el ingreso de datos al sistema.
- Implementar una serie de ejemplos que permitan verificar el correcto funcionamiento del programa.

## 3. MARCO REFERENCIAL

### 3.1 GENERALIDADES

Los sistemas de tuberías hoy en día son de gran relevancia en la industria, es por esto que, al momento de realizar la selección del tipo de tubería, sus dimensiones y geometría, es necesario garantizar que la característica de esta no contemple un gran cambio con los gradientes de temperatura y desplazamientos debidos a fuerzas externas e internas; asegurando la integridad y confiabilidad de dicho sistema y así evitar el mal funcionamiento.

El concepto de tubería se tomará como el de un cuerpo cilíndrico de sección circular que se limita de sus alrededores por una pared de espesor constante; creando así un cuerpo anular.

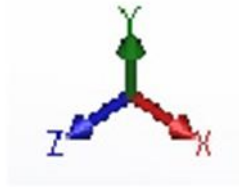
**3.1.1 Consideraciones Geométricas.** EL tratamiento geométrico de los sistemas de tuberías juega un papel muy importante dada su ubicación en el plano<sup>3</sup>, por lo que es importante establecer reglas geométricas, cabe resaltar que todos los programas tienen sus reglas geométricas por lo que se definirán algunas reglas generales:

- La tubería debe componerse únicamente de dos elementos, tramos rectos de longitud  $L$  y de codos que son cuerpos toroidales que poseen un radio de su línea de centro y un ángulo central ( $\theta$ )
- Para cambiar la dirección solo se podrán utilizar codos y las rectas adyacentes llegan y salen como tangentes inicial y final de este arco.
- El sistema de coordenadas se define de mano derecha, es decir cuando se giran los ejes en sentido positivo, el vector polar de giro se dirige hacia el tercer eje.

---

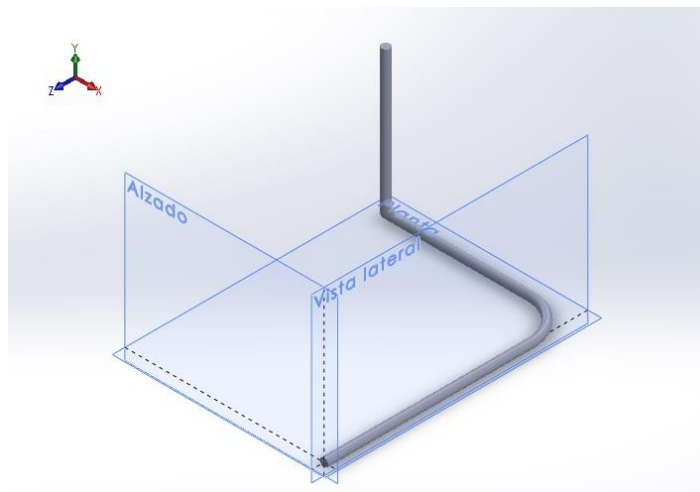
<sup>3</sup> Ibid., p. 64

Ilustración 1. Sistema isométrico para esquema de tuberías en el espacio.



- La longitud recorrida desde el comienzo desde el comienzo de la línea va a ser la variable principal, por esta razón se coloca el comienzo de la tubería en el origen del sistema de coordenadas, de tal manera que el primer elemento se ubique en uno de los planos principales que forman los tres ejes.
- No se deben ubicar componentes u elementos en planos oblicuos, es de resaltar que en las aplicaciones prácticas esto ocurre con menor frecuencia y se prefiere dirigir los tramos en direcciones paralelas y perpendiculares.

Ilustración 2. Ubicación de la tubería en el sistema de coordenadas y la descripción de su trayectoria.



- Se debe tener en cuenta la dirección deseada en los cambios de trayectoria, para esto se debe definir el signo del ángulo central de codo de este modo se tendrá una descripción geométrica precisa.

### 3.2 SOLICITACIONES EN TUBOS POR PRESIÓN INTERNA

La presión interna se considera la principal carga mecánica de las tuberías, los demás efectos dependen de algunas condiciones externas y varían de acuerdo con el lugar de operación, por lo que a continuación se analiza el comportamiento de la tubería bajo la presión interna<sup>4</sup>.

Se considerará un tubo recto de longitud  $L$  con diámetro exterior  $D_e$  e interior  $D_i$ , teniendo adentro la presión manométrica  $P$  contra la presión atmosférica.

El espesor de la pared es  $S$  y definen las siguientes ecuaciones básicas:

$$S = \frac{D_e - D_i}{2} \quad (1)$$

$$\text{ó} \quad D_e = D_i + 2S \quad (2)$$

El sistema de unidades utilizado debe ser coherente en todas sus dimensiones y unidades, de las dimensiones geométricas del tubo se definen las siguientes ecuaciones:

Sección transversal de flujo  $A_t = \frac{D_i^2 \pi}{4} \quad (3)$

Sección anular del materia  $A_a = \frac{D_e^2 - D_i^2}{4} \pi \quad (4)$

Momento de inercia de la sección  $I = \frac{D_e^4 - D_i^4}{64} \pi \quad (5)$

Momento de resistencia  $W = \frac{I}{D_e/2} \quad (6)$

Las solicitaciones para considerar son las siguientes:

- a) Tracción por la presión interna en dirección tangencial y axial.
- b) Compresión por la presión interna en dirección radial
- c) Flexión por el peso propio de la tubería y por la expansión térmica
- d) Torsión por las mismas cargas mecánicas.

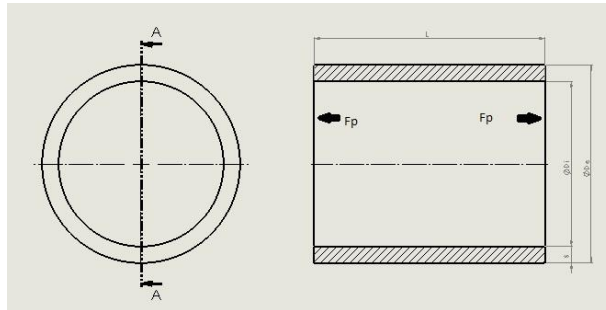
---

<sup>4</sup> Ibid., p. 69

Para el estudio de las solicitaciones mencionadas anteriormente se considerarán dos tipos de tuberías las de pared delgada y las de pared gruesa.

**3.2.1 Tubos de Pared Delgada.** La presión interna provoca en el tubo tres diferentes tensiones que se estudian a continuación<sup>5</sup>, para esto se somete un pedazo de tubo recto, tapado en ambos terminales para que de este modo actúe la presión, sobre las tapas se generan las siguientes fuerzas:

Ilustración 3. Cálculo de la tensión axial en tubo recto



$$F_p = PA_t \quad (7)$$

Estas fuerzas se transfieren al material del tubo y causan tensiones de tracción en dirección axial, paralelas con el eje que restablecen el equilibrio de las fuerzas externas e internas:

$$F_p = \sigma_a A_a \quad (8)$$

Igualando estas dos ecuaciones se despeja tensión  $\sigma_a$

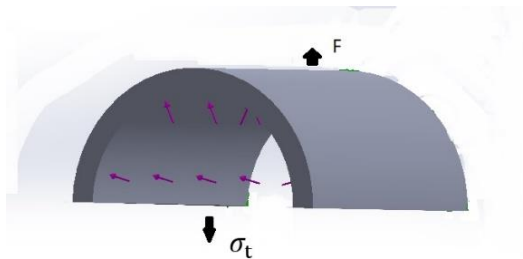
$$\sigma_a = P \frac{A_t}{A_a} = P \frac{D_i^2}{D_e^2 - D_i^2} = \frac{P}{4S} \frac{D_i^2}{D_m} = \frac{PD_i}{4S} \quad (9)$$

Esta tensión está siempre presente en la tubería excepto en el caso donde la tubería se encuentre abierta en ambos extremos, lo cual es imposible para el transporte de fluidos.

<sup>5</sup> Ibid., p. 71

**3.2.2 Tensiones Radiales y Tangenciales.** La presión interna es la causante de tensiones tangenciales y radiales, para analizar este tipo de tensiones se selecciona un elemento de longitud  $L$ , se efectúa un corte transversal en el cual las tensiones de la parte inferior se restablecen mediante las tensiones tangenciales  $\sigma_t$  presentes en el material<sup>6</sup>.

Ilustración 4. Cálculo de la tensión tangencial en tubo recto de pared delgada.



Sobre esta sección de tubería actúan las siguientes fuerzas:

- La fuerza externa que se deriva de la presión interna.
- La reacción interna del material de la tubería.

La fuerza interna y externa se encuentran en equilibrio y bajo esta condición se determina la tensión, para calcular esta fuerza externa se tomará un elemento con ángulo central  $d\varphi$  y se integrará la fuerza elemental  $dF$  que actúa sobre la superficie del mismo, cabe resaltar que la presión genera una fuerza perpendicular a la superficie en dirección  $dF$ , y esta se varía con el ángulo  $\varphi$ , siendo siempre radial. Para calcular la fuerza resultante se integran las componentes horizontal y vertical  $dF_h$  y  $dF_v$ , el cálculo de la fuerza elemental será:

$$|d_F| = PLd\varphi \frac{D_i}{2} \quad (10)$$

Con las componentes:

$$dF_h = \cos \varphi |d_F| \quad \text{y} \quad dF_v = \sen \varphi |d_F| \quad (11)$$

---

<sup>6</sup> Ibid., p. 72

Los límites de la integración son  $\varphi = 0^\circ \dots 180^\circ$

$$F_h = \int_0^\pi dF_h = 0 \text{ y } F_V = \int_0^\pi dF_V = PD_iL \text{ (12)}$$

De los resultados de la integración se obtiene que una fuerza vertical sobre la sección de corte de la tubería.

Las tensiones que actúan sobre las superficies de corte se equilibran:

$$2\sigma_t S l = PD_iL \text{ (13)}$$

De donde se despeja la tensión tangencial:

$$\sigma_t = \frac{PD_i}{2S} \text{ (14)}$$

Si se observa la tensión axial es aproximadamente la mitad de la tensión tangencial, la definición hallada anteriormente para para la tensión tangencial aplica para espesores pequeño relativo a el diámetro  $\frac{S}{D} \leq 0,05$ .

También es necesario determinar la componente radial de compresión, la cual varia bruscamente en la pared y debe compensar la presión interna, cabe resaltar que en la parte exterior es igual a 0 ya que no se generan tensiones de este tipo en esta parte, se define la tensión radial como:

$$\sigma_{rm} = -\frac{P}{2} \text{ (15)}$$

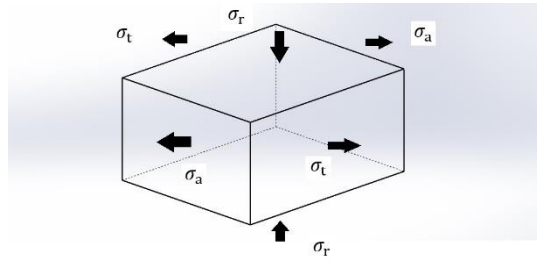
Donde el signo negativo indica que es una tensión de compresión

**3.2.3 Esfuerzo Equivalente.** Para tener una visión general de las tensiones del sistema es posible establecer una tensión equivalente, que representa el estado general del material y es un indicador de la resistencia y de la sollicitación máxima permisible. Según el teorema de Mohr se calcula el esfuerzo equivalente<sup>7</sup>:

---

<sup>7</sup> Ibid., p. 74

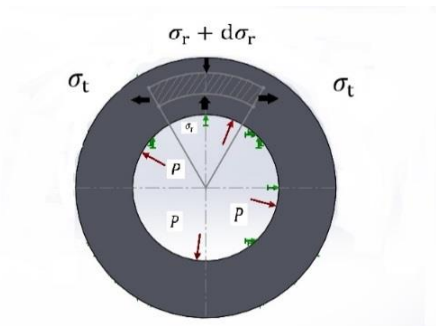
Ilustración 5. Tensiones principales en la pared de tubos rectos



$$\sigma_e = \sigma_t - \sigma_m = \frac{D_i P}{2S} + \frac{P}{2} = \frac{D_m P}{2S} \quad (16)$$

**3.2.4 Tubos Rectos de Pared Gruesa.** Las tensiones previamente halladas, solo presentan resultados satisfactorios para tubos de pared delgada, en caso en que el espesor sobrepase el 8% del diámetro, se presentara un error, por este motivo hay que llevar un estudio más profundo para determinar las tensiones axiales, radial y tangencial<sup>8</sup>.

Ilustración 6. Tensiones en tubos de pared gruesa



De acuerdo con las anotaciones de la figura se analiza el equilibrio de las fuerzas internas y el desplazamiento de un elemento diferencial que se ubican entre los radios  $r$  y  $r + d_r$ , teniendo el ángulo central  $d\varphi$ , el cálculo completo de estas tensiones tiene afín con el cálculo para tensiones de tubería de pared delgada por esta razón se resumirán los resultados más importantes:

<sup>8</sup> Ibid., p. 75

a) La tensión axial es constante e igual.

$$\sigma_a = \frac{PD_i}{4S} \quad (17)$$

b) La tensión tangencial se varía para el diámetro intermedio

$$D_i \leq D \leq D_e$$

$$\sigma_t = \frac{PD_i}{4S} \left( 1 + \frac{D_e^2}{D^2} \right) \quad (18)$$

c) La componente radial es:

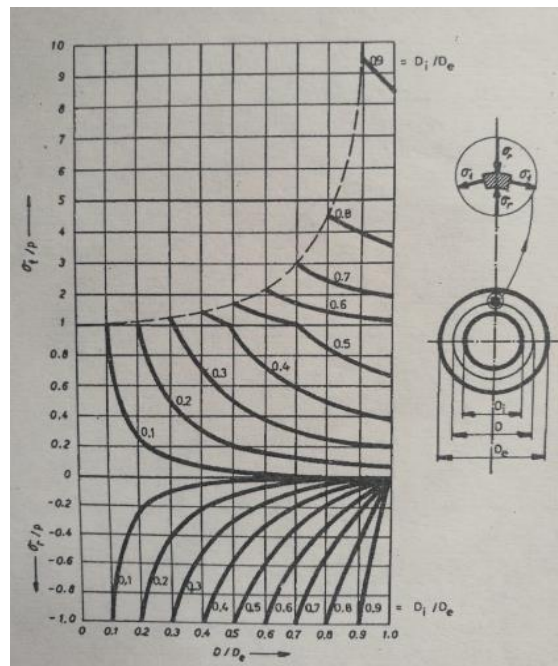
$$\sigma_r = \frac{PD_i}{4S} \left( 1 - \frac{D_e^2}{D^2} \right) \leq 0 \quad (19)$$

Y se varían entre los límites

$$\sigma_{ri} = -P \quad \text{y} \quad \sigma_{re} = 0$$

En la siguiente grafica se puede apreciar las variaciones de las tensiones radiales y tangenciales, el parámetro entre las curvas es la relación entre el diámetro interior e interior, se observa que los máximos de las dos tensiones están en la superficie interior del tubo y en el exterior tiene sus mínimos.

Ilustración 7. Variación de las tensiones tangencial y radial en tubo de pared gruesa



Fuente: G. RÁBEK. Instalaciones térmicas. Variación de las tensiones tangencial y radial en tubo de pared gruesa [imagen]. p. 76. [Consultado: 7 de octubre de 2019].

La tensión equivalente compuesta para tubos de pared gruesa se obtiene a partir de la aplicación del teorema de Mohr y del teorema de la equivalencia del trabajo de deformación<sup>9</sup>:

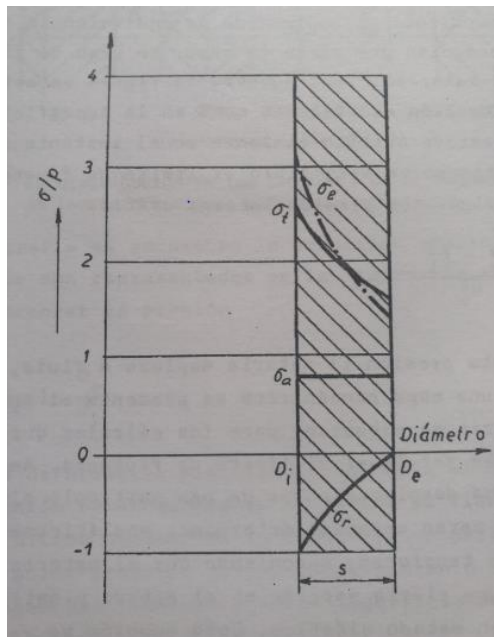
$$\sigma_e^M = \sigma_t - \sigma_r = 2 \left( \frac{PD_i}{4S} \right) \left( \frac{D_e^2}{D_i^2} \right) \quad (20)$$

$$\sigma_e^D = \frac{\sqrt{3}}{2} (\sigma_t - \sigma_r) = \sqrt{3} \left( \frac{PD_i}{4S} \right) \left( \frac{D_e^2}{D_i^2} \right) \quad (21)$$

A continuación, se puede observar la distribución de esfuerzos para las componentes axial, tangencial y equivalente previamente halladas, el parámetro de las curvas de relación será:

$$\frac{D_i}{D_e} = \frac{2}{3}$$

Ilustración 8. Distribución de las tensiones en la pared de un tubo de régimen elástico.



Fuente: G. RÁBEK. Instalaciones térmicas. Distribución de las tensiones en la pared de un tubo de régimen elástico [imagen]. p. 74. [Consultado: 7 de octubre de 2019].

<sup>9</sup> Ibid., p. 77

**3.2.5 Estado General de los Esfuerzos.** De acuerdo con la teoría de resistencia de materiales un tramo de tubería representa una viga, su carga se compone de la presión interna, del peso propio de la tubería más la de la masa de su aislante y de sus accesorios, estas componentes a su vez representan flexión en el plano vertical de la tubería que generaran posteriormente esfuerzos que actúan en dirección axial<sup>10</sup>.

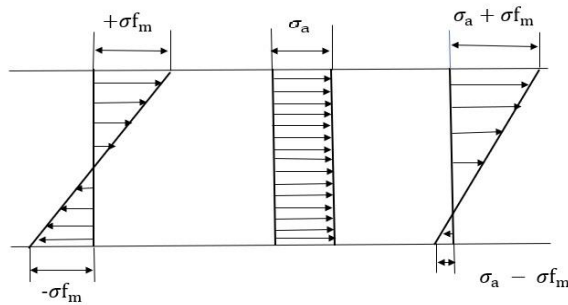
Para el caso en el que el plano de la tubería no coincida con la dirección de las cargas externas o de las trayectorias, se presentara un momento torsor que gira alrededor del eje de la tubería y que causa tensiones cortantes, de este modo el estado general de los esfuerzos será muy compuesto, a continuación, se presentaran las fórmulas y el estudio de estas de forma general sin entrar en detalles ya que corresponde a teoría de resistencia de materiales.

**3.2.6 Solicitaciones por Flexión.** Las sollicitaciones por flexión tienen una variación lineal a lo largo del tubo generando esfuerzos máximos en los hilos externos de la tubería, en su hilo exterior tensión máxima y en su hilo interior compresión máxima, la componente axial por la presión interna no tiene variación y permanece constante y el resultado de las dos tensiones tiene un comportamiento asimétrico, esto se debe la configuración de cada una de las tensiones mencionadas anteriormente, al momento de obtener una resultante en la parte superior se suman y en la parte inferior se restan.

---

<sup>10</sup> Ibid., p. 80

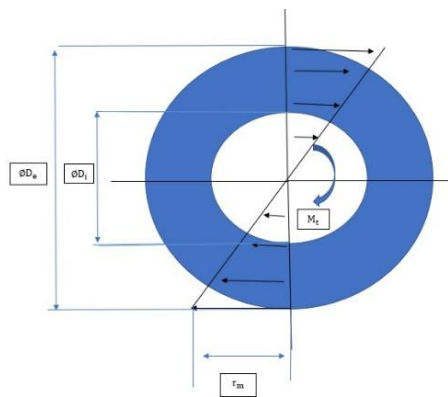
Ilustración 9. Distribución de las tensiones por flexión y la composición con las componentes axiales.



$$\sigma_{at} = \sigma_a + \sigma_{fl} = \frac{PD_i}{4S} + \frac{M_f}{I} \quad (22)$$

**3.2.7 Solicitaciones por Torsión.** El momento torsor causa tensiones cortantes en el plano perpendicular del tubo su distribución se observa a continuación<sup>11</sup>.

Ilustración 10. Tensiones cortantes en el tubo.

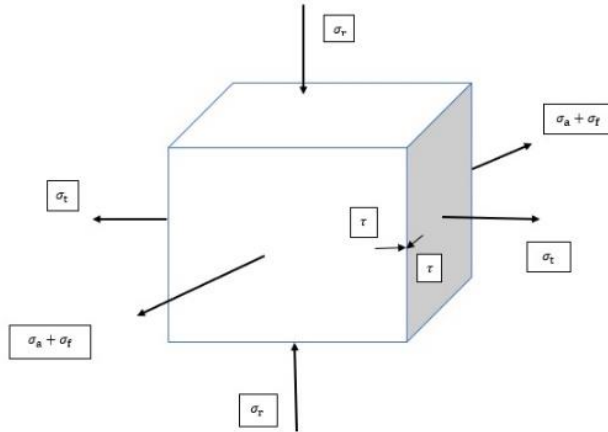


$$\tau = \frac{M_t}{I_p} = \frac{M_t}{2I} \quad (23)$$

Siendo  $I_p$  el momento polar de inercia.

<sup>11</sup> Ibid., p. 81

Ilustración 11. El estado de las tensiones en un cubo elemental.



Cortando un cubo elemental de la pared del tubo, se indican todas las componentes por presión interna, debido a los momentos torsor y flector, del teorema de la equivalencia de los trabajos de deformación se deduce<sup>12</sup>:

$$\sigma_e = \sqrt{3\sigma_a^2 \left(\frac{D_e^2}{D^2}\right) + (\sigma_{fm}^2 + 3\tau_m^2) \left(\frac{D_e}{D}\right)^2} \quad (24)$$

Del esfuerzo equivalente de la flexión y la torsión se puede demostrar que la tensión equivalente total  $\sigma_e$  tiene siempre su valor máximo en la superficie interna.

**3.2.8 Solicitaciones por Temperatura.** Los materiales tienden a alterar sus dimensiones con la temperatura y para los aceros al carbono a una temperatura de  $100[^\circ\text{C}]$  se pueden obtener dilataciones lineales aproximadas de  $1,1 \text{ mm}$  por metro, los valores exactos se pueden extraer de las tablas de la norma B31.3, que contiene estos valores para materiales comunes en la construcción de redes de tubería<sup>13</sup>.

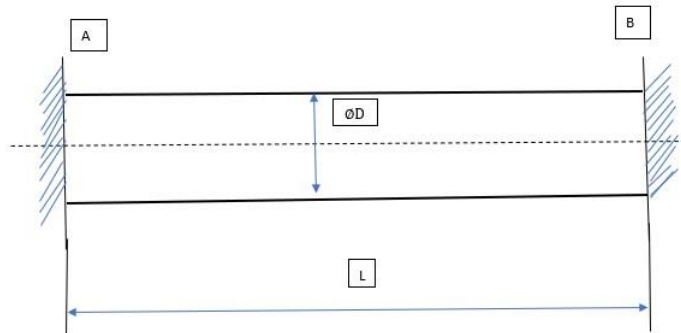
La expansión de un tubo se lleva a cabo en tres dimensiones, aunque para su estudio solo se tiene en cuenta el efecto lineal a lo largo de la tubería. En caso de que el cuerpo no tenga suficiente libertad para expandirse en dirección axial se

<sup>12</sup> Ibid., p. 81

<sup>13</sup>Ibid., p. 100

presentaran fuerzas elevadas permitiendo de este modo que el material alcance su límite de resistencia.

Ilustración 12. Expansión de tubo recto.



La elongación para un tramo de tuberías sin la restricción de los soportes sería:

$$\Delta L = \varepsilon_t L \quad (25)$$

Donde en  $\varepsilon_t$  es el coeficiente absoluto de expansión térmica y aplica para temperaturas entre  $21[^\circ C]$  hasta la temperatura  $t$ .

Los empotramientos impiden los desplazamientos lo cual genera una compensación mediante una fuerza de compresión  $F_t$ , aplicando la ley de Hooke se obtendrá la tensión resultante.

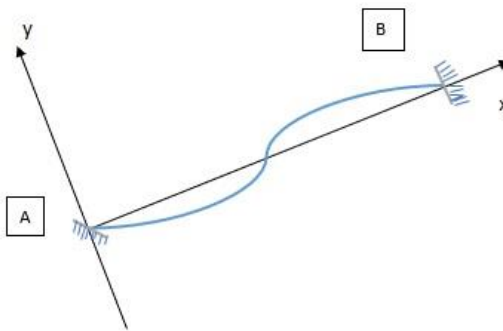
$$\sigma_T = \varepsilon_t E \quad (26)$$

Además de estas tensiones las reacciones de empuje hacia los empotramientos son grandes y se calculan de la siguiente manera:

$$F_t = A_t \sigma_t \quad (27)$$

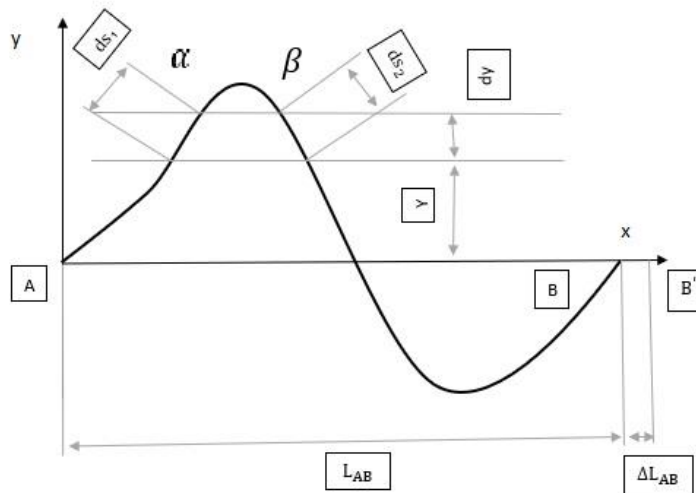
**3.2.9 Dilatación Térmica de las Tuberías.** Para el análisis de la dilatación térmica que sufren las tuberías se tomara como referencia un tramo de tubería empotrado a ambos lados y su debido sistema de coordenadas.

Ilustración 13. Ubicación del sistema de coordenadas para analizar la expansión térmica.



El origen de coordenadas se coloca en uno de los empotramientos de acuerdo con las reglas geométricas previamente establecidas para sistemas de tuberías, el otro empotramiento se encuentra sobre la abscisa<sup>14</sup>.

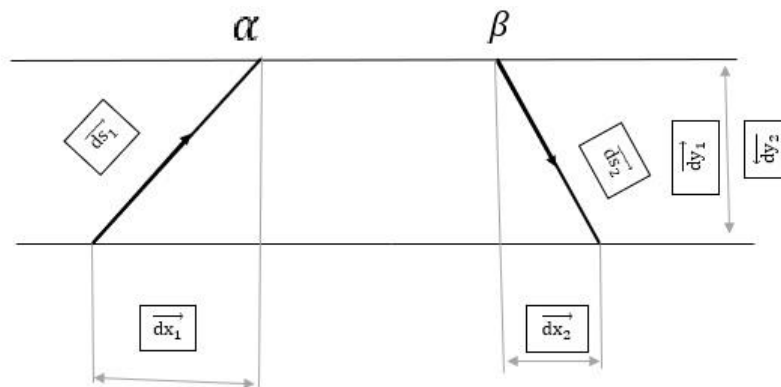
Ilustración 14. Estudio de la dilatación térmica en el plano.



<sup>14</sup> Ibid., p. 102

Se trazan líneas paralelas con altura  $y$  a la abscisa y a su vez separadas a una altura  $dy$ , las líneas paralelas cortan la tubería en los puntos,  $\alpha$ ,  $\beta$ ,  $\gamma$ ,  $\delta$ . La temperatura permanece constante a lo largo de la tubería y no hay cambio de material lo que significa que el coeficiente absoluto de expansión térmica no varía, la dilatación térmica aumenta proporcionalmente las dimensiones pero las posiciones mutuas conservan sus ángulos relativos, lo que significa que no distorsiona su trayectoria, tomando los diferencial  $ds_1$  y  $ds_2$  se puede observar que las longitudes horizontales de los elemento son diferentes debido a que poseen diferente inclinación sobre la abscisa.

Ilustración 15. Cálculo de la expansión térmica.



Se puede observar que las proyecciones en el eje paralelo ( $dx_1$  y  $dx_2$ ) van en la misma dirección y tienen diferente magnitud, mientras que las proyecciones verticales ( $dy_1$  y  $dy_2$ ) son iguales pero en diferente dirección, teniendo en cuenta las orientaciones y las magnitudes de las proyecciones, las componentes horizontales se suman y las componentes verticales se cancelan debido a sus orientaciones opuestas, de este modo se concluye que la elongación residual coincide entonces en la dirección de la abscisa y no tiene componente vertical.

De este modo la regla principal será:

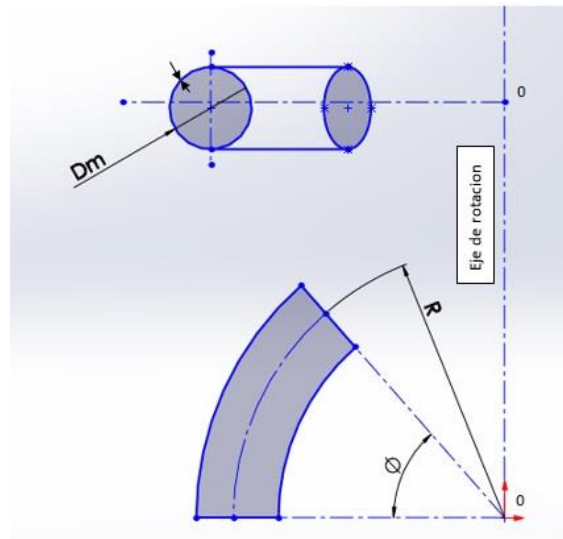
La expansión térmica tiene como efecto el desplazamiento del punto terminal a lo largo de la recta que une los dos puntos finales de la trayectoria y puede ser

calculada como la elongación de una barra recta cuya longitud es la distancia de los soportes.

### 3.3 CODOS

Los codos pertenecen a los elementos de máxima importancia en sistemas de tuberías y realizan los cambios de dirección de ramales rectos. Geométricamente pertenecen a los cuerpos toroidales y se generan girando un círculo alrededor de un eje. La distancia entre el eje del giro y el del círculo es el radio del codo que se marca a continuación por  $R^{15}$ .

Ilustración 16. Generación geométrica de un codo como superficie toroidal.



La presión interna provoca en la superficie interior del codo, es decir, hacía el centro de rotación una tensión mayor y en la exterior un esfuerzo menor que una tubería recta del mismo diámetro. Como consecuencia hay que aumentar el espesor hacía el centro y se puede reducir hacía afuera para compensar este fenómeno. Las

---

<sup>15</sup> Ibid., p. 83

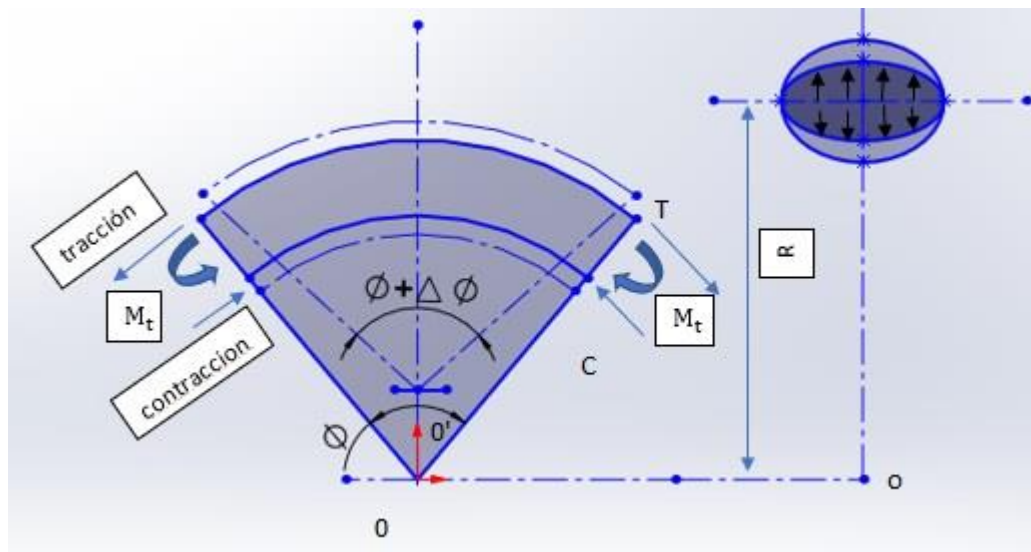
normas ANSI fijan en forma de una tabla el incremento del espesor de la pared en función de la relación R/D.

$$B_i = \frac{2R - \frac{D_e}{2}}{2R - D_e} \quad \text{factor de corrección en la superficie interior. (28)}$$

$$B_e = \frac{2R + \frac{D_e}{2}}{2R + D_e} \quad \text{factor de corrección en la superficie exterior. (29)}$$

**3.3.1 Deformación de los Codos Bajo Flexión.** Los codos muestran un comportamiento particular al cargarlos por un momento flector que actúa en el plano de su línea centro. Se constata la deformación de la sección circular que asume una forma similar elíptica y en consecuencia no se puede aplicar directamente los resultados comunes de la flexión de barras curvilíneas ya que este tratamiento excluye la deformación de la sección transversal<sup>16</sup>.

Ilustración 17. Deformación de los codos bajo flexión.



Se puede observar un codo con ángulo central  $\Phi$  y su sección circular en estado libre de cargas por flexión. Aplicando el momento flector  $M_f$  en el plano de la línea centro la sección se deforma y a la vez disminuye el radio y aumenta el ángulo

<sup>16</sup> Ibid., p. 84

central por  $\Delta\Phi$ . Para que se generen estas deformaciones deben estar presentes tales esfuerzos que no existen en caso de flexión sin alteración de la sección. Como se observa en la figura, esquemáticamente debe haber tensiones que se dirigen hacia la línea centro y además ejercen tracción en las líneas exteriores y compresión en las interiores<sup>17</sup>.

Estas tensiones adicionales se desplazan en su propia dirección y por eso absorben trabajo adicional de deformación, donde el momento flector  $M_f$  hace un giro mayor que el que se espera con un tratamiento convencional.

La variación del ángulo central es la siguiente:

$$\Delta\Phi = K \frac{R\Phi}{EI} M_f \quad (30)$$

Con la constante

$$K = 1 + \frac{9}{1+12\lambda^2} \quad (31)$$

Con el parámetro

$$\lambda = \frac{sR}{\left(\frac{Dm}{2}\right)^2} \quad (32)$$

Donde K se denomina el factor de Kármán o factor de flexibilidad, el parámetro  $\lambda$  es la característica de flexibilidad y se utilizan los símbolos K y h respectivamente.

Con el fin de darle al factor de flexibilidad una explicación simple, sin tomar en cuenta la alteración de la sección se deduce fácilmente que para este caso será  $K=1,0$ . El factor de flexibilidad corrige entonces la deformación considerando factores adicionales.

Los ensayos de Hovgaard concluyen que a partir de  $\lambda \geq 1,65$  ya no existe ninguna alteración de la sección y el codo se comporta como si tuviera la sección rígida.

---

<sup>17</sup> Ibid., p. 86

Hasta este valor el límite se podría considerarlo como un tramo recto equivalente a la longitud de

$$L_{ec} = \frac{0,4125Dm^2}{s} \quad (33)$$

Cuyo giro angular bajo la flexión  $M_f$  en su punto final es igual a  $\Delta\Phi$ . Es de interés que esta longitud equivalente es independiente del radio del codo R.

### 3.4 SOPORTES

Los soportes son accesorios usados para sostener una parte del peso del sistema de tuberías o cualquier carga impuesta verticalmente, además es uno de los factores más importante a la hora de realizar un diseño de este tipo, ya que los esfuerzos y desplazamientos pueden exceder las solicitaciones admisibles de material de la tubería y de la norma sobre la que se está diseñando, por lo que estos representan una solución a este tipo de problema.

**3.4.1 Soportes Flexibles.** Los soportes flexibles tienen como característica no modificar el plano del sistema de tuberías, pueden ser distribuidos en diferentes puntos del layout para aliviar la carga total compuesta por el peso de la tubería el aislante del material y el fluido a transportar, esto se debe a cargas de reacción ejercidas en sentido contrario por el accesorio<sup>18</sup>:

La efectividad de este tipo de soporte se debe al elemento amortiguador (resorte) el cual ofrece gran resistencia contra desplazamientos rápidos causados por carga dinámica, mientras que permite movimientos libres bajo desplazamientos aplicados proporcionalmente.

---

<sup>18</sup> Rentería, M. (2014). Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías. Lima, Perú: Pontificia universidad católica del Perú. p 46.

Este tipo de soportes se clasifica en soportes de carga variable y constante y su selección dependen de la variabilidad de carga que se obtenga mediante la siguiente formulación:

$$V = \frac{\Delta R}{HL} * 100\% \quad (34)$$

Dónde:

*V* Variabilidad (medida en %)

*R* Rigidez del resorte (N/m o lb/in)

*HL* Carga en operación en caliente (N o lb)

**3.4.2 Soportes Flexibles de Carga Constante.** El uso de un soporte de carga constante se hace cuando su porcentaje de variabilidad se encuentra por debajo del 6%, la principal ventaja de este tipo de soporte es su buen comportamiento para manejar las cargas de la tubería cuando se somete a expansión o contracción provocadas por la carga térmica, el uso de este tipo de soporte provee protección a equipos críticos que se encuentren en la línea tales como equipos rotatorios, válvula entre otros. Para la selección de este tipo de soportes se deben tener en cuenta las siguientes consideraciones<sup>19</sup>:

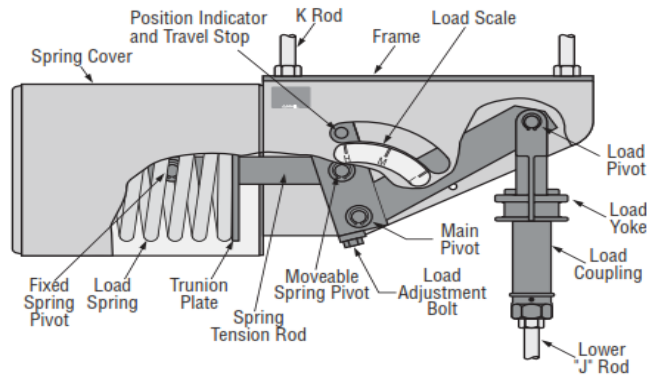
- Carga durante operación en caliente.
- Deformación máxima en operación en caliente.
- Ajuste de deformación.

La carga y la deformación se calculan en la etapa final del análisis de flexibilidad y el ajuste de la deformación es definido por el fabricante del accesorio.

---

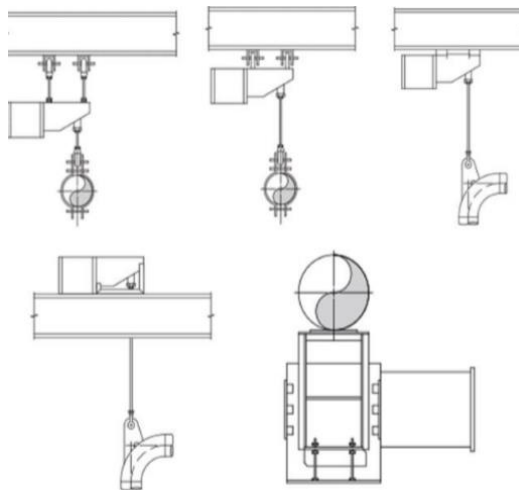
<sup>19</sup> Ibid., p.47

Ilustración 18. Soporte flexible de carga constante.



Fuente: Rentería, M. (2014). Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías. Soporte flexible de carga constante [imagen]. p.48. [Consultado: 17 de enero de 2020].

Ilustración 19. Formas constructivas de instalación de soporte flexible de carga constante



Fuente: Rentería, M. (2014). Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías. Soporte flexible de carga constante [imagen]. p.48. [Consultado: 18 de enero de 2020]

**3.4.3 Soportes Flexibles de Carga Variable.** Los soportes de carga variable son recomendados cuando su variabilidad no supera el 25 %, en la ilustración 20 se

pueden observar las disposiciones más comunes para este tipo de soporte. La carga en condición fría para este tipo de soporte se realiza bajo la siguiente formulación<sup>20</sup>:

$$CL = HL + \Delta \cdot R \quad (35)$$

Dónde:

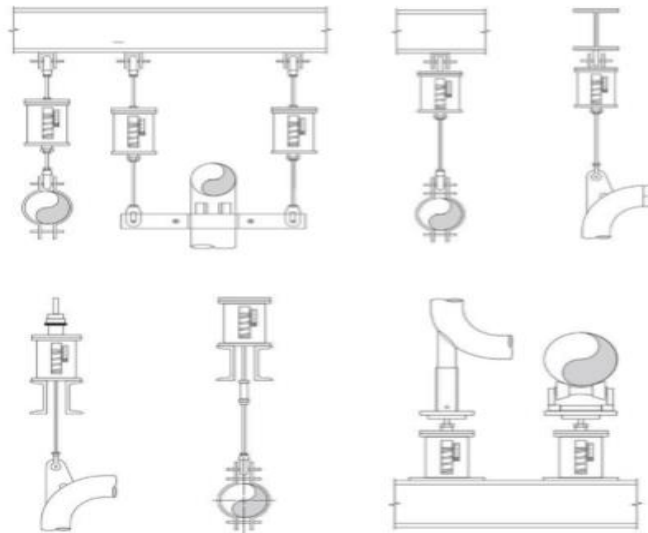
*CL* Carga en condiciones frías (lb)

*HL* Carga en operación en caliente (N o lb)

$\Delta$  Deformación debido a condiciones caliente (in)

*R* Rigidez del resorte (N/m o lb/in)

Ilustración 20. Formas constructivas de instalación de soporte flexible de carga variable.



Fuente: Rentería, M. (2014). Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías. Soporte flexible de carga constante [imagen]. p.50. [Consultado: 18 de enero de 2020].

<sup>20</sup> Ibid., p.49

## 4. NORMA ASME B31.3

El código B31 de ASME consiste en un número de secciones publicadas para el diseño de sistemas de tuberías sometidas a presión, clasificadas de acuerdo con el tipo de instalación contemplado para su desarrollo<sup>21</sup>. El estudio de este proyecto se basará en la norma B31.3 diseñada para el análisis de tuberías de proceso, se encuentran típicamente en las refinerías de petróleo, plantas químicas, farmacéuticas entre otras.

### 4.1 METODOLOGÍA DE ANÁLISIS DE FLEXIBILIDAD SEGÚN EL CÓDIGO ASME B31.3.

La metodología dispuesta por el código ASME B31.3, proporciona información útil respecto al límite de esfuerzos aceptados, además de la formulación necesaria para el cálculo de estos esfuerzos, Las soluciones de flexibilidad dependerán de la variación de las condiciones de diseño que se puedan establecer, entre las cuales se encuentra el cambio del tipo de soporte, la longitud de la tubería el tipo de material entre otras posibles soluciones que defina el diseñador<sup>22</sup>.

**4.1.1 Flexibilidad de las Tuberías.** Para evitar posibles fallas en el sistema de tuberías es de suma importancia que este cuente con la flexibilidad suficiente para contrarrestar los efectos de los esfuerzos y las deformaciones que puedan causar:

- a) Falla en la tubería y los accesorios por fatiga.
- b) Fugas en las conexiones, accesorios y equipos rotativos.
- c) distorsiones perjudiciales en el sistema o en los elementos que lo componen.

---

<sup>21</sup> The American Society of Mechanical Engineers. (2008a). *Código ASME para tuberías a presión, B31.3* NY, USA: ASME.  
<sup>22</sup> Rentería, M. (2014). *Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías*. Lima, Perú: Pontificia universidad católica del Perú.

**4.1.2 Requisitos Especiales.** Con la finalidad de garantizar que un sistema de tuberías cumpla con la norma aplicada es necesario tener en cuenta los siguientes requisitos<sup>23</sup>:

- a) Los esfuerzos totales no deberán superar los estipulados por la norma ASME B31.3, los cuales serán expuestos más adelante.
- b) Las fuerzas de reacción no deben ser perjudiciales para el sistema y sus elementos.
- c) Los desplazamientos calculados para el sistema deben estar en el límite recomendado.

#### **4.1.3 Esfuerzos por Desplazamiento.**

- a) Comportamiento elástico: Mediante aproximaciones los esfuerzos se pueden estimar de manera proporcional a las deformaciones totales por desplazamiento, siempre y cuando las deformaciones estén correctamente distribuidas y no se concentren en un solo punto.
- b) Comportamiento sobre deformado: un sistema de tubería que opere en la zona de deformación plástica, puede generar efectos perjudiciales debido a la acumulación de estas deformaciones en las zonas más críticas del sistema.  
Cuando se usen materiales poco dúctiles es recomendable compensar la falta de equilibrio del sistema, esto se consigue en la etapa de diseño mediante el uso de distribución correcta de los soportes.

**4.1.4 Deformaciones por Desplazamiento.** Este tipo de esfuerzos generan desplazamientos en la tubería ya sea por el efecto térmico, por las reacciones del sistema o por los desplazamientos previamente definidos, al momento de determinar las deformaciones totales, deberán considerarse simultáneamente.

---

<sup>23</sup> The American Society of Mechanical Engineers. (2008a). *Código ASME para tuberías a presión, B31.3* NY, USA: ASME.p38.

#### **a) DESPLAZAMIENTOS TÉRMICOS.**

Un sistema de tuberías presentara cambios si este experimenta cualquier cambio en su diferencia de temperatura. Si al sistema se le restringe la libertad de expansión o contracción debido a cualquier tipo de impedimento como, equipos conectados y elementos de sujeción como soportes, guías y anclajes, el sistema presentara un desplazamiento desde su lado libre.

#### **b) FLEXIBILIDAD DE LA SUJECIÓN.**

Si se considera una sujeción fija esta se debe tener en cuenta al momento de determinar el rango de esfuerzos por desplazamientos y sus reacciones.

#### **c) DESPLAZAMIENTOS IMPUESTOS EXTERNAMENTE.**

Los movimientos externos tendrán efecto en los desplazamientos totales del sistema, dentro de este tipo de desplazamientos podemos considerar las condiciones sísmicas y meteorológicas de una zona específica.

### **4.2 PROPIEDADES PARA EL ANALISIS DE FLEXIBILIDAD.**

**a)** Valores para el intervalo de esfuerzos: Los valores de desplazamientos térmicos a usarse para determinar deformaciones totales por desplazamiento o para calcular el intervalo de los esfuerzos, se determinarán usando la diferencia algebraica entre su valor a temperatura máxima del metal y este mismo a temperatura mínima del metal para el ciclo térmico bajo análisis<sup>24</sup>.

**b)** Valores para reacciones. Los valores para desplazamientos térmicos a usarse para determinar deformaciones totales por desplazamiento para el cálculo de reacciones en soportes o equipos conectados se determinarán como la diferencia algebraica entre el valor máximo (o mínimo) de temperatura del ciclo térmico bajo análisis y el valor de la temperatura esperada durante la instalación.

---

<sup>24</sup> Ibid., p.40

**c)** Módulo de elasticidad: el módulo de elasticidad de referencia  $E_a$  a 21°C y el módulo de elasticidad a temperatura máxima y mínima  $E_m$  debe considerarse de acuerdo al apéndice en el párrafo 319.3.1(a) o (b) de la norma ASME B31.3.

**d)** Relación de Poisson: Se considera 0.3 para todos los metales a cualquier temperatura.

**e)** Dimensiones: para cálculos de flexibilidad se debe considerar los espesores y diámetros externos de la tubería y sus accesorios.

**4.2.1 Esfuerzos de Flexibilidad.** Para determinar el intervalo de esfuerzos por desplazamiento  $SE$ , se debe calcular los esfuerzos por torsión y flexión para luego combinarlos, el resultado de estos no puede superar el intervalo de esfuerzos permitidos  $SA^{25}$ .

$$S_E = \sqrt{S_b^2 + 4S_t^2} \quad (36)$$

$$S_t = \frac{M_t}{2Z} \quad (37)$$

$$S_b = \frac{\sqrt{(I_i M_i)^2 + (I_o M_o)^2}}{Z} \quad (38)$$

$$Z = \frac{\pi(d^4 - d_i^4)}{32d} \quad (39)$$

Dónde:

$M_t$ : Momento torsional.

$S_b$ : Esfuerzo por flexión resultante.

$S_t$ : Esfuerzo por torsión.

$Z$ : Módulo de sección de tubería.

$I_i$ : Índice de momento planar sostenido. En la ausencia de datos más aplicables,  $I_i$  se considera como el valor más grande entre  $0,75I_i$  o 1.

$I_o$ : Índice de momento no planar sostenido. En la ausencia de datos más aplicables,  $I_o$  se considera como valor el más grande entre  $0,75I_o$  o 1.

---

<sup>25</sup> Ibid., p.41

$M_i$ : Momento planar debido a cargas sostenidas, ej. presión y peso.

$M_o$ : Momento no-planar debido a cargas sostenidas, ej. Presión y peso.

**4.2.2 Esfuerzo por Carga Sostenida.** La carga sostenida es aquella que el sistema debe soportar durante todo el tiempo, dentro de esta carga se encuentra el peso del propio sistema y su presión, a su vez estos generan esfuerzos longitudinales que se calculan de la siguiente manera<sup>26</sup>:

$$S_L = \sqrt{(|S_a| + S_b)^2 + (2S_t)^2} \quad (40)$$

$$S_a = \frac{I_a + F_a}{A_p} \quad (41)$$

$$S_t = \frac{M_t}{2Z} \quad (42)$$

Dónde:

$S_a$ :Esfuerzo por fuerza longitudinal sostenida.

$S_b$ : Esfuerzo por flexión resultante.

$S_t$ :Esfuerzo por torsión

$I_a$ : Índice de fuerza longitudinal sostenida, en ausencia de datos  $I_a$  se considera como 1.

$F_a$ : Fuerza longitudinal por cargas sostenidas, presión y peso.

$M_t$ : Momento torsional.

$Z$ : Módulo de sección sostenido.

### **4.3 LÍMITES DE ESFUERZOS OCASIONADOS POR CARGAS SOSTENIDAS Y DEFORMACIONES POR DESPLAZAMIENTO.**

**4.3.1 Esfuerzos por Cargas Sostenidas,  $S_L$ .** La suma de los esfuerzos por carga sostenida  $S_L$  generados por la presión y el peso propio no debe exceder  $S_h$ , donde

---

<sup>26</sup> Ibid., p.44

el valor de  $S_h$  se obtiene en la Tabla A-1 de la norma ASME B31.3, basado en la temperatura del metal bajo la condición de operación considerada<sup>27</sup>.

**4.3.2 Intervalo de Esfuerzo Permitido por Desplazamiento,  $S_A$ .** El intervalo de esfuerzo por desplazamiento  $S_E$ , determinado en un sistema de tuberías, no debe exceder el límite de esfuerzo permitido por desplazamiento, calculado por la ecuación:

$$S_A = f(1,25S_c + 0,25S_h) \quad (43)$$

Si  $S_h$  es mayor que  $S_L$ , la diferencia entre ellos puede sumarse al término  $0,25 S_h$  en la ecuación anterior. En ese caso, el intervalo de esfuerzo permitido es calculado por la ecuación:

$$S_A = f[1,25(S_c + S_h) - S_L] \quad (44)$$

Dónde:

$f$ : Factor del intervalo de esfuerzo, determinado por la ecuación

$$f = 6,0(N)^{-0,2} \leq f_m \quad (45)$$

Cuando se usa un valor de  $f > 1$  en las ecuaciones anteriores el valor máximo que pueden tener para  $S_c$  y  $S_h$  debe ser 138MPa 20ksi).

$f_m$  (Valor máximo del factor de intervalo de esfuerzos): 1,2 para materiales ferrosos con una resistencia específica mínima a la *tensión*  $\leq 517[MPa]$  y una temperatura del metal  $\leq 371^\circ C$  si no se cumplen estas condiciones  $f_m = 1$ .

$N$ : Número equivalente de ciclos de desplazamiento completo.

$S_c$ : Esfuerzo permitido *base*<sup>6</sup> a la temperatura esperada mínima del metal durante el ciclo de desplazamiento en consideración. (el valor de este se encuentra en la tabla A-1)

$S_h$ : Esfuerzo permitido *base*<sup>6</sup> a la temperatura esperada máxima del metal durante el ciclo de desplazamiento en consideración.

---

<sup>27</sup> Ibid., p.121

## 5. CALFEM

CALFEM es una herramienta computacional diseñada para el aprendizaje del método de elementos finitos, el nombre CALFEM es una abreviatura para "Computer Aided Learning of the Finite Element Method", sus principales aplicaciones radican en la solución problemas de mecánica estructural, él programa se basa fundamentalmente en una caja de herramientas "TOOLBOX" con una serie de funciones prediseñadas con el fin de facilitar el desarrollo de problemas mediante el método de elementos finitos, para el desarrollo del proyecto se hace uso de este toolbox y de las funciones desarrolladas en este.

Las funciones disponibles se organizan en grupos de la siguiente manera:

Ilustración 21. Grupo de funciones CALFEM.

<b>Groups of functions</b>	
<b>General purpose commands</b>	for managing variables, workspace, output etc
<b>Matrix functions</b>	for matrix handling
<b>Material functions</b>	for computing material matrices
<b>Element functions</b>	for computing element matrices and element forces
<b>System functions</b>	for setting up and solving systems of equations
<b>Statement functions</b>	for algorithm definitions
<b>Graphics functions</b>	for plotting

Fuente: P-E AUSTRELL, O DAHLBLOM, J LINDEMANN, A OLSSON, K-G OLSSON, K PERSSON, H PETTERSSON, M RISTINMAA, G SANDBERG, P-A WERNBERG. Calfem a finite element toolbox. Groups of functions [imagen]. P.9. [Consultado: 19 de mayo de 2020].

A continuación, se describirán las funciones, matrices y ecuaciones tomadas de CALFEM para el desarrollo de la herramienta computacional para el análisis de flexibilidad en sistemas de tuberías.

**a) Matriz Edof:** es la matriz general de grados de libertad, que contiene también el número de elementos en que se divide el sistema para su desarrollo, la topología de la matriz de grados de, está definida de la siguiente manera.

Ilustración 22. Definición de la matriz de grados de libertad.

$$edof = [el \quad \underbrace{dof_1 \quad dof_2 \quad \dots \quad dof_{ned}}_{\text{global dof.}}]$$

Fuente: P-E AUSTRELL et al. Calfem a finite element toolbox. Definición de la matriz de grados de libertad [imagen]. P.161. [Consultado: 19 de mayo de 2020].

Donde la primera columna corresponde a el número de elemento y las siguientes corresponde a los grados de libertad globales del elemento.

Ilustración 23. Matriz de grados de libertad.

$$Edof = \left[ \begin{array}{cccccc} el_1 & dof_1 & dof_2 & . & . & dof_{ned} \\ el_2 & dof_1 & dof_2 & . & . & dof_{ned} \\ \vdots & \vdots & \vdots & & & \vdots \\ el_{ned} & dof_1 & dof_2 & . & . & dof_{ned} \end{array} \right] \left. \vphantom{\begin{array}{cccccc} \end{array}} \right\} \text{one row for each element}$$

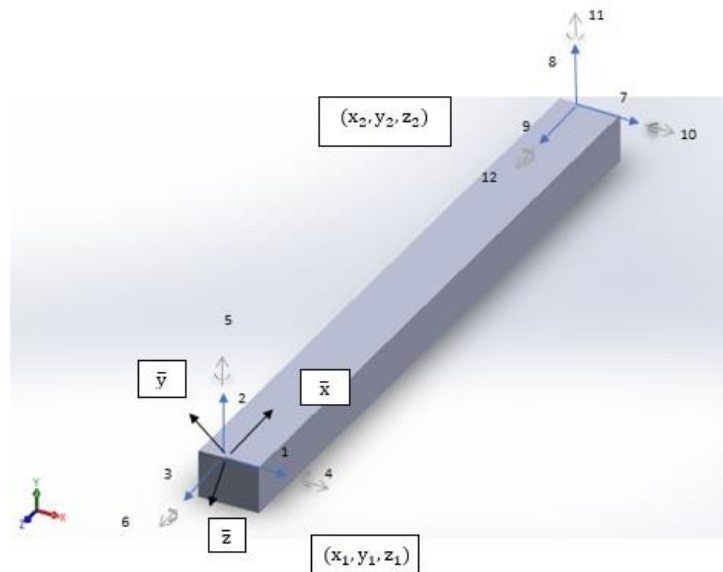
Fuente: P-E AUSTRELL et al. Calfem a finite element toolbox. Matriz de grados de libertad[imagen]. P.161. [Consultado: 19 de mayo de 2020].

**b) Matriz K:** Es la matriz de rigidez global la cual depende del número total de elementos en que se divide el sistema y por lo tanto del número total de grados de libertad, posteriormente esta matriz se ensamblara con cada una de las matrices de rigidez elementales, inicialmente es una matriz simétrica de “0”, la topología de la

matriz depende del número de grados de libertad, es decir si el número total de grados de libertad del sistema son 24, la matriz de rigidez será de 24x24.

**c) Matriz  $K_e$ :** Es la matriz de rigidez de un elemento, la cual relaciona los desplazamientos de los nodos con sus esfuerzos puntuales, en el desarrollo del proyecto aplica el caso de una matriz de rigidez para un elemento tridimensional.

Ilustración 24. Grados de libertad para un elemento tridimensional.



Las dimensiones de la matriz de rigidez dependen del número de grados de libertad que se tenga por elemento, para el caso de un elemento tridimensional se considerarán 12 grados de libertad por elemento, es decir 6 grados de libertad por nodo, al final se obtendrá una matriz de rigidez 12x12, cada uno de los elementos de la matriz dependerán de los desplazamientos y las rotaciones en cada uno de sus ejes.

Ilustración 25. Matriz de rigidez para un elemento tridimensional.

$$\bar{\mathbf{K}}^e = \begin{bmatrix} k_1 & 0 & 0 & 0 & 0 & 0 & -k_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_x}{L^3} & 0 & 0 & 0 & \frac{6EI_x}{L^2} & 0 & -\frac{12EI_x}{L^3} & 0 & 0 & 0 & \frac{6EI_x}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 & 0 & 0 & -\frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & k_2 & 0 & 0 & 0 & 0 & 0 & -k_2 & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_x}{L^2} & 0 & 0 & 0 & \frac{4EI_x}{L} & 0 & -\frac{6EI_x}{L^2} & 0 & 0 & 0 & \frac{2EI_x}{L} \\ -k_1 & 0 & 0 & 0 & 0 & 0 & k_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_x}{L^3} & 0 & 0 & 0 & -\frac{6EI_x}{L^2} & 0 & \frac{12EI_x}{L^3} & 0 & 0 & 0 & -\frac{6EI_x}{L^2} \\ 0 & 0 & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & -k_2 & 0 & 0 & 0 & 0 & 0 & k_2 & 0 & 0 \\ 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_x}{L^2} & 0 & 0 & 0 & \frac{2EI_x}{L} & 0 & -\frac{6EI_x}{L^2} & 0 & 0 & 0 & \frac{4EI_x}{L} \end{bmatrix}$$

in which  $k_1 = \frac{EA}{L}$  and  $k_2 = \frac{GK_x}{L}$ , and where

Fuente: P-E AUSTRELL et al. CalfeM a finite element toolbox. Matriz de rigidez para un elemento tridimensional. [Imagen]. P.146. [Consultado: 19 de mayo de 2020].

La longitud de cada uno de los elementos se calcula de acuerdo con la siguiente ecuación:

$$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (46)$$

Donde  $x_2$  y  $x_1$  representa el punto final e inicial de cada elemento en la dirección  $x$  y así para cada una de las coordenadas del elemento en las diferentes direcciones.

**d) Vector de carga por elemento ( $f_e$ ):** representa las cargas nodales equivalentes más las cargas aplicadas en los nodos en cada uno de los elementos que conforman el sistema.

Ilustración 26. Vector de carga para un elemento tridimensional.

$$\bar{f}_i^c = \begin{bmatrix} \frac{q_x L}{2} \\ \frac{q_y L}{2} \\ \frac{q_z L}{2} \\ \frac{q_w L}{2} \\ -\frac{q_x L^2}{12} \\ \frac{q_y L^2}{12} \\ \frac{q_x L}{2} \\ \frac{q_y L}{2} \\ \frac{q_z L}{2} \\ \frac{q_w L}{2} \\ \frac{q_x L^2}{12} \\ -\frac{q_y L^2}{12} \end{bmatrix}$$

Fuente: P-E AUSTRELL et al. Calfem a finite element toolbox. Vector de carga para un element tridimensional. [Imagen]. P.147. [Consultado: 19 de mayo de 2020].

**Vector Ep:** Se define como el vector de propiedades de cada uno de los elementos del sistema.

$$ep = [E \ G \ A \ I_y \ I_z \ K_v]$$

Dónde:

- $E$ : módulo de elasticidad o módulo de Young.
- $G$ : define el módulo de cizalladura
- $A$ : área transversal del elemento, en este caso área transversal de la tubería
- $I$ : momento de inercia.
- $K_v$ : módulo de torsión de la sección circular.

Cabe resaltar que el vector Ep está sujeto a cambios ya que puede tener menos o más propiedades de acuerdo con los cálculos a realizar.

**e) Vector eq:** vector de carga utilizado si se van a aplicar cargas distribuidas.

$$eq = [q_x \ q_y \ q_z \ q_w]$$

El vector contiene cargas distribuidas en cada una de las direcciones del sistema coordinado y  $q_w$  representa un par distribuido.

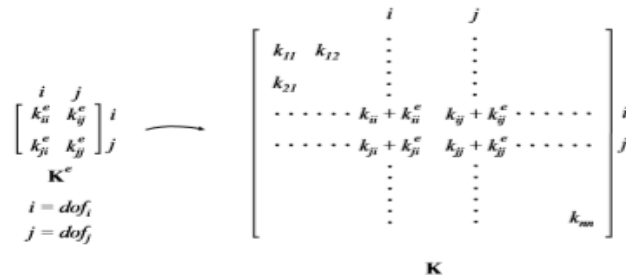
**f) Función beam3e:** se encarga de crear la matriz de rigidez y vector de carga de cada uno de los elementos del sistema, utilizando las coordenadas del elemento y cada uno de los vectores previamente definidos.

$$K_e = \text{beam3e}(ex, ey, ez, eo, ep)$$

$$[K_e, f_e] = \text{beam3e}(ex, ey, ez, eo, ep, eq)$$

**g) Función assem:** Usando la función assem, las matrices de rigidez y vectores de carga calculados para cada elemento se ensamblan en la matriz de rigidez global  $K$  y en el vector global de carga  $f$ .

Ilustración 27. Función de ensamblaje de las matrices elementales.



Syntax:

$$K = \text{assem}(\text{edof}, K, K_e)$$

$$[K, f] = \text{assem}(\text{edof}, K, K_e, f, f_e)$$

Fuente: P-E AUSTRELL et al. Calfem a finite element toolbox. Función de ensamblaje de las matrices elementales. [Imagen]. P.161. [Consultado: 19 de mayo de 2020].

Para el ensamble en la matriz de rigidez global, se requiere de la matriz global de grados de libertad, la matriz de rigidez por elemento, el vector de carga global y el vector de carga por elemento.

**h) Matriz boundary conditions (bc):** “bc” es la FUNCIÓN de CALFEM que permite definir las condiciones de frontera o restricciones de desplazamiento y rotación que tiene cada elemento.

$$bc = [1\ 0; 2\ 0; 3\ 0; 4\ 0; 5\ 0; 6\ 0; 7\ 0; 8\ 0; 9\ 0; 10\ 0; 11\ 0; 12\ 0]$$

Para un elemento tridimensional se cuentan con 12 grados de libertad, 6 por cada nodo, los cuales se enumeran de la siguiente manera:

- 1: desplazamiento en  $x$
- 2: desplazamiento en  $y$
- 3: desplazamiento en  $z$
- 4: rotación alrededor del eje  $x$  ( $\theta_x$ )
- 5: rotación alrededor del eje  $y$  ( $\theta_y$ )
- 6: rotación alrededor del eje  $z$  ( $\theta_z$ )

Si en ese grado de libertad se restringe su desplazamiento o rotación se le asigna el valor de “0”, automáticamente el sistema colocara la restricción en ese sentido.

**i) Función solveq:** calcula el vector de desplazamientos ( $a$ ) y reacciones ( $r$ ) considerando las condiciones de contorno previamente establecidas.

$$[a, r] = solveq(K, f, bc)$$

La función **solveq** resuelve el sistema de ecuaciones:

$$Ka = f \quad (47)$$

donde  $K$  es una matriz,  $a$  y  $f$  son vectores. La matriz  $K$  y el vector  $f$  deben estar predefinidos. La solución del sistema de ecuaciones se almacena en un vector creado por la función, si se dese prescribir algún valor para  $a$  el numero de fila y los valores correspondientes se dan en la matriz de condiciones de contorno.

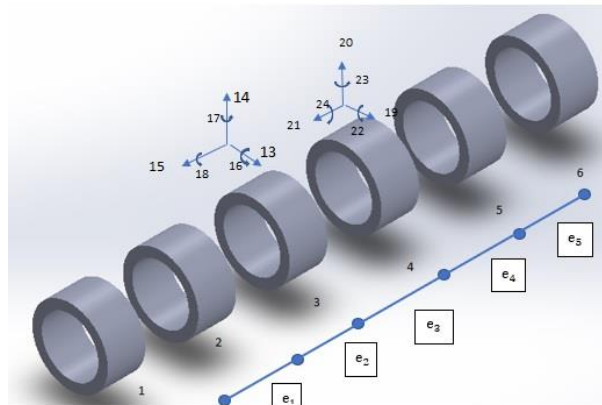
$$r = Ka - f \quad (48)$$

## 6. DESARROLLO METODOLÓGICO

### 6.1 COMPORTAMIENTO BAJO CARGA SOSTENIDA.

En el inicio del desarrollo del programa se discretiza todo el sistema de tubería en tramos, para posteriormente ser llevado a una matriz general de coordenadas; luego de obtener la matriz de coordenadas se debe generar la matriz de grados de libertad (Edof) donde se encuentra cada elemento con sus correspondientes grados de libertad, cabe resaltar que entre elementos se comparte la numeración de los grados de libertad como se observa en la siguiente imagen, donde el elemento 1 y 2 comparten los grados de libertad 4,5,6.

Ilustración 28. Grados de libertad en una viga simple.



Con las propiedades del material con el cual se desea desarrollar el sistema de tuberías (vector  $E_p$ ) y la geometría calculamos la matriz de rigidez de cada elemento, con su respectivo vector de carga, que se compone de las fuerzas externas y los propios pesos de la tubería ( $f_e$ ).

Posteriormente se calcula la matriz de rigidez con su vector de carga térmica (el cual se describirá más adelante), en caso de que el fluido a transportar se encuentre

a una temperatura diferente a la temperatura de referencia, en el caso donde sea igual a la temperatura ambiente no habría expansión térmica y por ende el desplazamiento sería 0.

El desarrollo de este software se basa en la superposición de las matrices de rigidez y los vectores de carga de cada elemento, para luego resolver la ecuación de equilibrio global, la cual relaciona las fuerzas externas con la matriz de rigidez y los desplazamientos de cada nodo, que denotaremos de forma compacta así <sup>28</sup>:

$$P = KD \quad (49)$$

P: vector de cargas externas

K: matriz de rigidez de la estructura

D: Vector de grados de libertad

En la mayoría de los casos las estructuras de este tipo se encuentran estáticamente indeterminadas por ende sus reacciones son desconocidas, por lo que es conveniente dividir las matrices implicadas en la ecuación anterior, separando los grados de libertad sujetos a las reacciones, donde los desplazamientos normalmente son nulos

$$\begin{pmatrix} P_a \\ P_b \end{pmatrix} = \begin{pmatrix} K_{aa} & K_{ab} \\ K_{ba} & K_{bb} \end{pmatrix} \begin{pmatrix} D_a \\ D_b \end{pmatrix} \quad (50)$$

$P_a$ : Subvector de fuerzas externas de los grados de libertad restringidos

$D_a$ : Subvector de desplazamientos de los grados de libertad restringidos

$P_b$ : Subvector de fuerzas externas de los grados de libertad libres

$D_b$ : Subvector de desplazamientos de los grados de libertad libres

$K_{aa}$ : Submatriz que relaciona los grados de libertad restringidos entre si

$K_{bb}$ : Submatriz que relaciona los grados de libertad libres

$K_{ab}, K_{ba}$ : Submatriz que corresponde a las relaciones cruzadas entre grados de libertad libres y restringidos, de lo anterior podemos notar que:

---

<sup>28</sup> Hernández, J. (2014c). Análisis matricial de estructuras Curso con MATLAB. Bogotá, Colombia: Universidad Nacional de Colombia.p10.

$$K_{ab} = K_{ba}^T \quad (51)$$

Resolviendo la segunda fila de la ecuación (51)

$$P_b = K_{ba}D_a + K_{bb}D_b \quad (52)$$

Podemos observar que el vector  $D_a$  de desplazamientos es conocido (nulos debido a que son las restricciones totales), y le corresponden un vector  $P_a$  de reacciones desconocidas, mientras sucede lo opuesto para los grados de libertad libres donde  $P_b$  es conocido, ya que está formado por las fuerzas externas, mientras que  $D_b$  es desconocido.

Despejando  $D_b$ :

$$D_b = K^{-1}_{bb}(P_b - K_{ba}D_a) \quad (53)$$

Ahora bien, a partir de la primera fila de la ecuación (50) se obtiene

$$P_a = K_{aa}D_a + K_{ab}D_b \quad (54)$$

Reemplazando la ecuación (53) en (54) obtenemos

$$P_a = K_{aa}D_a + K_{ab}K^{-1}_{bb}(P_b - K_{ba}D_a) \quad (55)$$

Para un  $D_a$  nulo (las restricciones sean 0, por ejemplo, empotramientos) las ecuaciones (53) y (54) se reducen en

$$D_b = K^{-1}_{bb}P_b \quad (56)$$

$$P_a = K_{ab}D_b \quad (57)$$

Al resolver la ecuación 56 obtenemos los desplazamientos en los grados de libertad libres, y en la ecuación 57 las reacciones en los apoyos.

Una vez calculados los desplazamientos de cada grado de libertad podemos obtener las fuerzas internas en cada sección de tubería con la ecuación

$$P_e = K_e D_e \quad (58)$$

En donde se refiere a la matriz de rigidez y desplazamientos cada uno de los nodos; con estos valores de las fuerzas internas y las ecuaciones de la Norma ASME B.31.3 se calculan los esfuerzos por carga sostenida como por expansión.

**6.1.2 Comportamiento Bajo Carga Térmica.** Otro aspecto para evaluar en la generación de esfuerzos y desplazamientos son los cambios de temperatura, suponiendo una estructura sin cargas externas a una temperatura donde los desplazamientos son nulos, es decir a la temperatura de referencia. La variación de la temperatura con respecto este estado referencia es:

$$\Delta T = T - T_{ref} \quad (59)$$

Esta puede ser positiva o negativa lo cual indicara si es tensión o compresión respectivamente; si el elemento es libre este se desplazará de  $L$  a  $L + dt$ . Si el comportamiento termo elástico es lineal, entonces  $dt$  es proporcional a  $L$  y  $\Delta T$

$$dt = \alpha L \Delta T \quad (60)$$

donde  $\alpha$  es el coeficiente de expansión térmica, asumiremos que este es uniforme a lo largo de todo el elemento.

El esfuerzo térmico se define así:

$$e_t = d_t/L = \alpha \Delta T \quad (61)$$

Suponiendo que al elemento también se encuentra asociada una fuerza mecánica axial, se realiza una superposición de esfuerzos, esta superposición es básica en análisis termo mecánicos y es:

$$e = e_M + e_T = \frac{\sigma}{E} + \alpha \Delta T \quad (62)$$

teniendo en cuenta que  $e = d/L$  y  $d = \vec{u}_{xj} - \vec{u}_{xi}$  definimos:

$$\frac{\vec{u}_{xj} - \vec{u}_{xi}}{L} = \frac{\sigma}{E} + \alpha \Delta T \quad (63)$$

Con el fin de convertirlo a fuerzas internas lo multiplicamos por A y E:

$$\frac{AE}{L} \left( \begin{array}{c} \rightarrow \\ U_{xj} \\ -\leftarrow \\ U_{xi} \end{array} \right) = A\sigma + EA\alpha\Delta T \quad (64)$$

Donde  $EA\alpha\Delta T$  es la fuerza térmica interna, definiéndolo de forma matricial

$$F_t = \frac{EA}{L} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

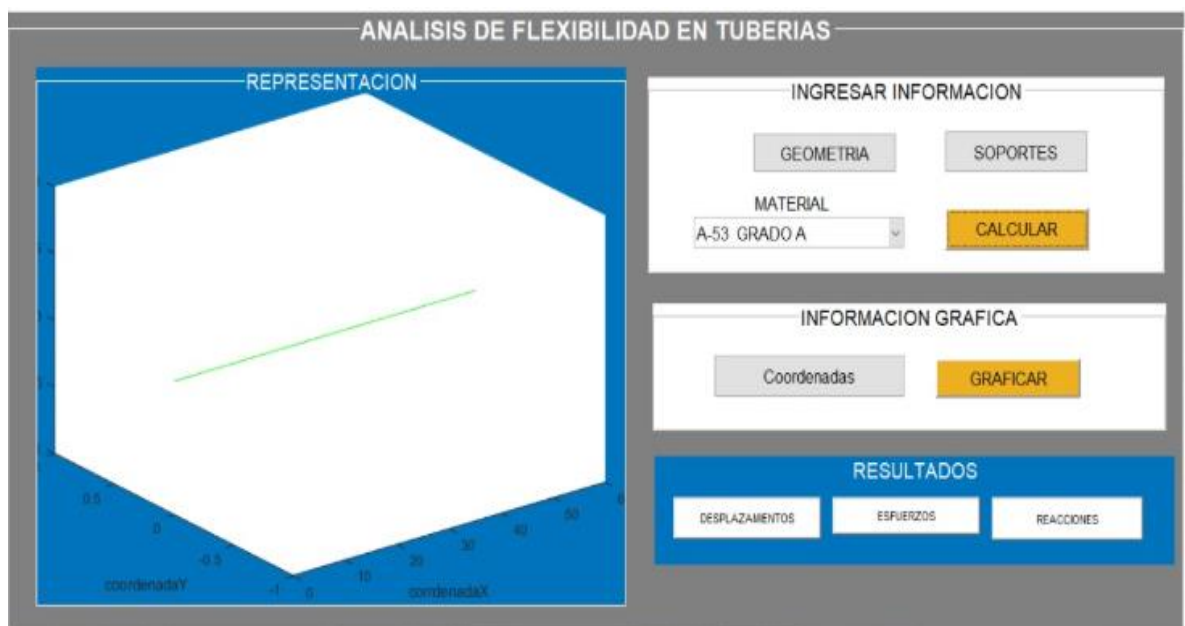
Nótese que las fuerzas térmicas solo actúan a lo largo de elemento como fue explicado anteriormente en la sección 3.2.9 (Dilatación térmica de las tuberías).

## 7. DESCRIPCION INTERFAZ

### 7.1 INTERFAZ GRAFICA PRINCIPAL

La función Guide de MATLAB proporciona la ventaja de poder elaborar una interfaz gráfica, para facilitar la interacción del usuario con el planteamiento numérico. En la Figura se puede observar la interfaz desarrollada, la cual consta de 4 paneles, 2 para ingresar la información del sistema, y los dos restantes para graficar y mostrar resultados numéricos.

Ilustración 29. Interfaz principal de la herramienta.



En el panel “ingresar información” se puede modificar las características geométricas y material de la tubería, al igual que las posiciones de los soportes a lo largo de la misma.

En la pestaña “Geometría”, el usuario puede introducir los parámetros geométricos de la tubería, los cuales son: Diámetro interior, diámetro exterior, presión de trabajo y temperatura de trabajo. Dichos valores son ingresados en unidades del sistema inglés. (Anexo E, Anexo D)

En la pestaña “Soportes”, el usuario especifica la ubicación de los soportes presentes en el sistema de tubería, realizando una restricción en dicha ubicación. Finalmente, en la pestaña “material” se presenta una lista de materiales comunes en sistemas de tubería, y adicionalmente se puede ingresar cualquier materia de su preferencia. (Anexo A)

En el panel “Información gráfica”, en la pestaña de “coordenadas”, se ingresan los valores de longitud y posición de sistema de tuberías, siendo 10 el primer tramo de tubería, 20 el segundo tramo y así de manera consecutiva. (Anexo F)

Con los botones “calcular” y “graficar” (de color amarillo) realizan los análisis correspondientes. Los resultados de esfuerzo, reacciones y desplazamientos se encuentran en la parte inferior de la interfaz y la representación gráfica de la tubería se puede observar al lado izquierdo de esta. (Anexo B, Anexo C)

El código de programación para el desarrollo de la Interfaz se encuentra en los anexos a partir del Anexo A, mostrando todas sus funciones y códigos auxiliares hasta el Anexo O.

## 8. RESULTADOS

Para la validación de los resultados se realiza una comparación del software para análisis de flexibilidad en tuberías CAE PIPE respecto a la herramienta desarrollada. Se evalúan tres casos, siendo el primero una tubería recta, el segundo caso, es un planteamiento con un codo (y dos planos), y el ultimo contiene dos codos y está presente en los tres planos.

Los datos para evaluar son reacciones, desplazamientos, esfuerzos de carga y expansión para cada caso.

### 8.1 PRIMER CASO

Se puede observar en las Tablas 1, 2, 3, 4 y 5 los resultados del análisis de flexibilidad en tuberías por medio del software, y los obtenidos por medio de la interfaz en MATLAB.

Condiciones:

- Tubería sin codos
- Longitud =60 [in]
- Material = A-53 GRADO – A
- Soportes al inicio y al final. (representación gráfica figura xx)

Tabla 1. Reacciones de carga, caso 1.

REACCIONES DE CARGA									
Node	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	103,7	108	4,1%	0	0	0,0%
20	0	0	0,0%	103,7	108	4,1%	0	0	0,0%
Node	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
10	0	0	0,0%	0	0	0,0%	1039	1083	4,2%
20	0	0	0,0%	0	0	0,0%	1039	1083	4,2%

Tabla 2. Reacciones de expansión, caso 1.

REACCIONES DE EXPANSIÓN									
Node	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	362300	379992	4,9%	0	0	0,0%	0	0	0,0%
20	362300	379992	4,9%	0	0	0,0%	0	0	0,0%
Node	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0	0	0,0%	0	0	0,0%

Tabla 3. Desplazamiento por expansión, caso 1.

DESPLAZAMIENTO DE EXPANSION									
Node	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0	0	0,0%	0	0	0,0%
Node	XX			YY			ZZ		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0	0	0,0%	0	0	0,0%

Tabla 4. Desplazamiento por carga, caso 1.

DESPLAZAMIENTO POR CAGA									
Node	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0	0	0,0%	0	0	0,0%
Node	XX			YY			ZZ		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0	0	0,0%	0	0	0,0%

Tabla 5. Esfuerzos por carga y expansión, caso 1.

NODO	SL			NODO	ST		
	E	T	ERROR		E	T	ERROR
10	42,68	44	-3,0%	10	28.378	29774	-4,7%
20	42,68	44	-3,0%	20	28.387	29774	-4,7%

## 8.2 SEGUNDO CASO

Se puede observar en las Tablas 6, 7, 8, 9 y 10 los resultados del análisis de flexibilidad en tuberías por medio del software, y los obtenidos por medio de la interfaz en MATLAB.

Condiciones:

- Tubería con un codo.
- Longitud = 60 [in] (Cada tramo de tubería)
- Material = A-53 GRADO – A
- Soportes al inicio y al final

Tabla 6. Reacciones de carga, caso 2.

REACCIONES DE CARGA									
NODE	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	384	-387	0,8%	0	0	0,0%
40	0	0	0,0%	391	-387	1,0%	0	0	0,0%
NODE	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	2447	-2472	-1,0%	0	0	0,0%	16588	-17032	-2,7%
40	16699	-17032	-2,0%	0	0	0,0%	2361	-2472	-4,7%

Tabla 7. Reacciones de expansión, caso 2.

REACCIONES DE EXPANSION									
NODE	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	-4011	3944	1,70%	0	0	0,00%	-3920	3660	7,10%
40	4011	3994	1,70%	0	0	0,00%	3920	3660	7,10%
NODE	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,00%	196240	199320	1,49%	0	0	0,00%
40	0	0	0,00%	196240	-196200	1,49%	0	0	0,00%

Tabla 8. Desplazamiento por carga, caso 2.

DESPLAZAMIENTO POR CARGA									
NODE	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	-0,008	-0,006	26,2%	0	0	0,0%

DESPLAZAMIENTO POR CARGA									
30	0	0	0,0%	-0,008	-0,006	22,5%	0	0	0,0%
40	0	0	0,0%	0	0	0,0%	0	0	0,0%
NODE	XX			YY			ZZ		
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	1E-04	0,00006	40,0%	0	0	0,0%	2E-04	0,0002	20,0%
30	2E-04	0,00016	20,0%	0	0	0,0%	1E-04	6E-05	40,0%
40	0	0	0,0%	0	0	0,0%	0	0	0,0%

Tabla 9. Desplazamientos por expansión, casó 2.

DESPLAZAMIENTO DE EXPANSION									
Node	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	0	0	0,0%		0	0,0%
20	0,082	0,059	28,0%	0	0	0,0%	0,11	-0,0849	22,8%
30	0,102	0,0849	17,1%	0	0	0,0%	0,08	-0,0599	24,7%
40	0	0	0,0%	0	0	0,0%		0	0,0%
NODE	XX			YY			ZZ		
10	0	0	0,0%	0	0	0,0%	0	0	0,0%
20	0	0	0,0%	0,0026	0,00181	30,4%	0	0	0,0%
30	0	0	0,0%	0,0026	0,00181	30,4%	0	0	0,0%
40	0	0	0,0%	0	0	0,0%	0	0	0,0%

Tabla 10. Esfuerzos por carga y expansión, casó 2.

Node	SL			Node	SE		
	E	T	ERROR		E	T	ERROR
10	700,65	702	0,2%	10	10171	9845	3,3%
20	111,3	102	9,1%	20	2372,5	2166	9,5%
30	95,8	102	6,1%	30	2042,5	2166	5,7%
40	705,322	702	0,5%	40	10330	9845	4,9%

### 8.3 TERCER CASO

Se puede observar en las Tablas 11, 12, 13, 14 y 15 los resultados del análisis de flexibilidad en tuberías por medio del software, y los obtenidos por medio de la interfaz en MATLAB.

Condiciones:

- Tubería con dos codos.
- Longitud = 60 [in] (Cada tramo de tubería)

- Material = A-53 GRADO – A
- Soportes al inicio y al final

Tabla 11. Reacciones de carga, caso 3.

REACCIONES DE CARGA									
NODE	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	675	665	1,5%	0	0	0,0%
60	0	0	0,0%	669	665	0,6%	0	0	0,0%
NODE	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	4659	5089	8,4%	0	0	0,0%	38010	39676	4,2%
60	4659	4994	6,7%	0	0	0,0%	36233	39676	8,7%

Tabla 12. Reacciones de expansión, caso 3.

REACCIONES DE EXPANSION									
NODE	Fx [lb]			Fy [lb]			Fz [lb]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	2940	2970	1,0%	0	0	0,0%	3820	3811	0,2%
60	2940	2970	1,0%	0	0	0,0%	3200	3811	16,0%
NODE	Mx [lb-in]			My [lb-in]			Mz [lb-in]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0	0	0,0%	192270	189961	1,2%	0	0	0,0%
60	0	0	0,0%	192270	189961	1,2%	0	0	0,0%

Tabla 13. Desplazamiento de expansión, caso 3.

DESPLAZAMIENTO DE EXPANSION									
NODE	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
20	0,0413	0,0602	31,4%	0,0	0,0	0%	0,0689	0,0402	71,4%
30	0,0529	0,0746	29,1%	0,0	0,0	0%	0,0300	0,0239	25,5%
40	0,0468	0,0746	37,3%	0,0	0,0	0%	0,0300	0,0159	88,7%
50	0,0406	0,0602	32,6%	0,0	0,0	0%	0,0689	0,0450	53,1%
60	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
NODE	XX [Rad]			YY [Rad]			ZZ [Rad]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
20	0,0	0,0	0%	0,0011	0,0835	98,7%	0,0	0,0	0%
30	0,0	0,0	0%	0,0016	0,0100	84,0%	0,0	0,0	0%
40	0,0	0,0	0%	0,0017	0,0100	83,0%	0,0	0,0	0%
50	0,0	0,0	0%	0,0010	0,0835	98,8%	0,0	0,0	0%
60	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%

Tabla 14. Desplazamiento por carga, caso 3.

DESPLAZAMIENTO POR CARGA									
NODE	X [inch]			Y [inch]			Z [inch]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
20	0,0	0,0	0%	0,016	0,017	5,7%	0,0	0,0	0%
30	0,0	0,0	0%	0,040	0,049	18,4%	0,0	0,0	0%
40	0,0	0,0	0%	0,039	0,049	20,2%	0,0	0,0	0%
50	0,0	0,0	0%	0,016	0,016	0,0%	0,0	0,0	0%
60	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
NODE	XX [Rad]			YY [Rad]			ZZ [Rad]		
	E	T	ERROR	E	T	ERROR	E	T	ERROR
10	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%
20	0,0	0,0	0%	0,0	0,0	0%	0,0004	0,0244	98,4%
30	0,0	0,0	0%	0,0	0,0	0%	0,0001	0,0094	98,9%
40	0,0	0,0	0%	0,0	0,0	0%	0,0001	0,0094	98,9%
50	0,0	0,0	0%	0,0	0,0	0%	0,0004	0,0244	98,4%
60	0,0	0,0	0%	0,0	0,0	0%	0,0	0,0	0%

Tabla 15. Esfuerzos de carga y expansión, caso 3.

NODE	SL			NODE	SE		
	E	T	ERROR		E	T	ERROR
10	1635	1538	6,3%	10	9133	8982	1,7%
20	356	413	13,8%	20	4216	3934	7,2%
30	311	325	4,3%	30	1913	1811	5,6%
40	308	325	5,2%	40	2001	1811	10,5%
50	348	413	15,7%	50	4216	3934	7,2%
60	1659	1638	1,3%	60	8919	8982	0,7%

## 9. CONCLUSIONES

El planteamiento numerico para el analisis de flexibilidad en sistemas de tuberia determina los valores correspondientes a las reacciones de carga y de expansion con un pocentaje de error pomedio del 4%,lo cual permite hacer un acercamiento a la posibilidad de implementar esta herramienta para evaluar el análisis de flexibilidad en sistemas de tuberías con fines académicos en la universidad industrial de Santander. Los errores son atribuidos a distintas causas, como los valores de las propiedades de los materiales tomados por el Software caepipe,ademas de las aproximaciones para descomponer la carga en los cambios de eje cordenado a lo largo del recorrido de la tuberia.

La precisión de los resultados del programa disminuye con la complejidad geométrica del sistema de tuberías a evaluar, debido a aproximaciones realizadas al representar numéricamente el análisis de la tubería en los codos de la misma. Lo cual permite enfocar futuros estudios en la optimización del análisis numérico para el cálculo de los esfuerzos y deformaciones en estos accesorios.

La interfaz del programa desarrollado en este trabajo de grado facilita al usuario el ingreso y la modificación de las condiciones geométricas y de trabajo en los sistemas de tuberías. La interfaz permite al operador ingresar un sistema particular y generar modificaciones en los valores específicos, sin alterar condiciones previamente establecidas; agilizando el proceso comparativo a diferentes parámetros o condiciones.

El programa realizado presenta valores de estimación aceptables, con buena precisión en concordancia con los valores de referencia, se presentan valores de error elevados en los desplazamientos; lo cual, no es causal de una mala predicción ya que los valores relativos numéricos están en el orden de  $10e-4$ , representando estadísticamente un sesgo aceptable dentro de los valores de estimación.

## BIBLIOGRAFIA

A.J.M. FERREIRA, Matlab codes for finite element analysis: solids and structures. Portugal: Springer, 2009.

AVALLONE EUGENE A., THEODORE BAUMEISTER III, SADEGH ALI M. Marks' standard handbook for mechanical engineers. Ed 11 – New York. Mc Graw Hill Inc. 2007.

BLANCO DIAZ ELENA, CERVERA RUIZ MIGUEL, SUAREZ ARROYO BENJAMIN, Analisis matricial de estructuras. Barcelona, España: Centro Internacional de Metodos Numericos en Ingenieria (CINME), 2015.

CAEPIPE User's Manual, Version 6.52a, ©2012, SST Systems, Inc. All Rights Reserved.

FEDESARROLLO. (2014, Diciembre 14). Evaluación de la contribución económica del sector de hidrocarburos colombiano frente a diversos escenarios de producción [en línea]. <https://www.repository.fedesarrollo.org.co> Recuperado 6 de marzo de 2019.

FISH JACOB, BELYTETSCHKO TED. A first course in finite elements. England. John Wiley & son's ltd, The Atrium, Southern Fate, Chichester, West Sussex, 2007,

GOMEZ-GORDO, ESTEBAN MELCHOR. Análisis de flexibilidad en sistemas de tuberías. Madrid, España, proyecto de fin de carrera para el título de ingeniero industrial. Universidad Carlos III de Madrid. Escuela Politécnica Superior, departamento de Ingeniería mecánica.

HURTADO, Jorge Eduardo. Análisis matricial de estructuras: Curso con MATLAB. Bogotá, Colombia: Universidad Nacional de Colombia.p10.

HUSTON RONALD, HAROLD JOSEPHS. Practical stress analysis in engineering design. Ed 3. New York.: Crc press Taylor & Francis group

LÓPEZ LIÉVANO, Daniel Ramon. Análisis de esfuerzos de origen térmico y mecánico, a una tubería de transporte de vapor de una planta de procesos, con modelado por elementos finitos, México, Julio 2012. Trabajo de grado para obtener el título de MAESTRO EN CIENCIAS EN INGENIERÍA MECÁNICA. Instituto Politécnico Nacional, Escuela superior de Ingeniería mecánica y eléctrica, sección de estudios de posgrados e investigación, México D.F.

RÁBEK, G. (1982). Instalaciones térmicas (1.a ed.). Caracas, Venezuela: Universidad central de Venezuela.p59.

RENTERÍA VIDAURRE, Marcelo. Diseño de metodología para el análisis de flexibilidad en sistemas de tuberías. Lima, Perú, 2014, p 46. Proyecto fin de carrera. Pontificia universidad católica del Perú. Facultad de ciencias e ingeniería.

SAM KANNAPPAN, P.E. Introduction to pipe stress analysis, Knoxville, Tennessee. John Wiley & son's ltd, 1986.

THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS. Tuberías de proceso: Código ASME para tuberías a presión, B31.3. (revisión de ASME N31.3-2008). 31 de marzo de 2011. Three Park Avenue New York, USA

# ANEXOS

## Anexos A. Resultados de esfuerzos interfaz

**ESFUERZOS**

**CARGA EN OPERACION**

	NODO	SL	SH
1			
2			
3			
4			

**EXPANSION**

	NODO	SE	SA
1			
2			
3			
4			

**ACEPTAR**

## Anexo B. Resultados de Reacciones interfaz

REACCIONES							
CARGA SOSTENIDA							
NODO	X	Y	Z	Mx	My	Mz	

EXPANSION							
NODO	X	Y	Z	Mx	My	Mz	

TOTALES							
NODO	X	Y	Z	Mx	My	Mz	

**ACEPTAR**

## Anexo C. Resultados de desplazamientos interfaz

DESPLAZAMIENTO							
CARGA SOSTENIDA							
NODO	X	Y	Z	Xx	Yy	Zz	

EXPANSION							
NODO	X	Y	Z	Xx	Yy	Zz	

TOTALES							
NODO	X	Y	Z	Xx	Yy	Zz	

ACEPTAR

## Anexo D. Ingreso datos de material Interfaz

**INGRESAR DATOS DEL MATERIAL**

POISON	<input type="text"/>
E	<input type="text"/>
DENSIDAD	<input type="text"/>
COEFICIENTE EXPANSION TERMICA	<input type="text"/>
PESO TUBERIA	<input type="text"/>
ESFUERZO Max PERMISIBLE (Tref)	<input type="text"/>

**Anexo E. Ingreso de Geometría y condiciones de trabajo interfaz**

GEOMETRIA	
DIAMETRO EXTERIOR [ Inch ]	DIAMETRO INTERIRO [ Inch ]
<input type="text"/>	<input type="text"/>

CONDICIONES DE TRABAJO	
TEMPERATURA DE TRABAJO [ °F ]	PRESION DE TRABAJO [ PSI ]
<input type="text"/>	<input type="text"/>

## Anexo F. Ingreso de coordenadas en el sistema interfaz

**COORDENADAS**

**VALORES**

NODE

X

Y

Z

TYPE

	#NODO	X	Y	Z	TYPE
1					
2					
3					
4					

## Anexo G. Código Principal

### CODIGO PRINCIPAL

```
function varargout = GUIprin(varargin)
clc
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @GUIprin_OpeningFcn, ...
                  'gui_OutputFcn', @GUIprin_OutputFcn, ...
                  'gui_LayoutFcn', [], ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
global M ;

[coor dx,coor dy,coor dz,Pc]=GRAFICAR(M);

plot3(coor dx,coor dy,coor dz,'g')
```

```

xlabel('corrdenadaX');
ylabel('coordenadaY');
zlabel('coordenadaZ');

% hObject   handle to pushbutton2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles   structure with handles and user data (see GUIDATA)

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
material=get(handles.popupmenu1,'value');
global pois E rho_st alpha dtemp Sc SH
switch material
    case 1
        if dtemp<500
            SH=16000;
        elseif dtemp>=500 & dtemp<600
            SH=15300;
        elseif dtemp>=600 & dtemp<650
            SH=14600;
        elseif dtemp>=650 & dtemp<700
            SH=12700;
        elseif dtemp>=700 & dtemp<750
            SH=10700;
        elseif dtemp>=750 & dtemp<800
            SH=9200;
        elseif dtemp>=800 & dtemp<850
            SH=7900;
        elseif dtemp>=850 & dtemp<900

```

```

    SH=5900;
elseif dtemp>=900 & dtemp<950
    SH=4000;
else
    SH=2500;
end
pois=0.3;    % LB/IN
E=20.46e6;   %modulo lb/inch
rho_st=0.243; %densidad
alpha=6.512e-6;%coeficiente de expansion termica

Sc=16000;
case 2
    if dtemp<400
        SH=20000;
    elseif dtemp>=400 & dtemp<500
        SH=19900;
    elseif dtemp>=500 & dtemp<600
        SH=19000;
    elseif dtemp>=600 & dtemp<650
        SH=17900;
    elseif dtemp>=650 & dtemp<700
        SH=17300;
    elseif dtemp>=700 & dtemp<750
        SH=16700;
    elseif dtemp>=750 & dtemp<800
        SH=13900;
    elseif dtemp>=800 & dtemp<850
        SH=11400;
    elseif dtemp>=850 & dtemp<900

```

```

    SH=8700;
elseif dtemp>=900 & dtemp<950
    SH=5900;
elseif dtemp>=950 & dtemp<1000
    SH=4000;
else
    SH=2500;
end
pois=0.3;
E=29007547.6;
rho_st=0.2817928776;
alpha=7.3e-6; %COLOCAR EL VALOR
Sc=20000;
case 3
if dtemp<400
    SH=20000;
elseif dtemp>=400 & dtemp<500
    SH=18600;
elseif dtemp>=500 & dtemp<600
    SH=17500;
elseif dtemp>=600 & dtemp<650
    SH=16600;
elseif dtemp>=650 & dtemp<700
    SH=16200;
elseif dtemp>=700 & dtemp<750
    SH=15800;
elseif dtemp>=750 & dtemp<800
    SH=15500;
elseif dtemp>=800 & dtemp<850
    SH=15200;

```

```
elseif dtemp>=850 & dtemp<900
    SH=14900;
elseif dtemp>=900 & dtemp<950
    SH=14600;
elseif dtemp>=950 & dtemp<1000
    SH=14300;
else
    SH=14000;
end
pois=0.3;
E=28282358.91;
rho_st=0.2854056068;
alpha=9.4e-6;
Sc=20000;
case 4
if dtemp<400
    SH=20000;
elseif dtemp>=400 & dtemp<500
    SH=19900;
elseif dtemp>=500 & dtemp<600
    SH=19000;
elseif dtemp>=600 & dtemp<650
    SH=17900;
elseif dtemp>=650 & dtemp<700
    SH=17300;
elseif dtemp>=700 & dtemp<750
    SH=16700;
elseif dtemp>=750 & dtemp<800
    SH=13900;
elseif dtemp>=800 & dtemp<850
```

```

    SH=11400;
elseif dtemp>=850 & dtemp<900
    SH=8700;
elseif dtemp>=900 & dtemp<950
    SH=5900;
elseif dtemp>=950 & dtemp<1000
    SH=4000;
else
    SH=2500;
end
pois=0.3;
E=30457924,98;
rho_st=0.2835992422; %posible fuente de error
alpha=7.3e-6;
Sc=20000;
case 5

if dtemp<400
    SH=20000;
elseif dtemp>=400 & dtemp<500
    SH=19900;
elseif dtemp>=500 & dtemp<600
    SH=19000;
elseif dtemp>=600 & dtemp<650
    SH=17900;
elseif dtemp>=650 & dtemp<700
    SH=17300;
elseif dtemp>=700 & dtemp<750
    SH=16700;
elseif dtemp>=750 & dtemp<800

```

```

    SH=13900;
elseif dtemp>=800 & dtemp<850
    SH=11400;
elseif dtemp>=850 & dtemp<900
    SH=8700;
elseif dtemp>=900 & dtemp<950
    SH=5900;
elseif dtemp>=950 & dtemp<1000
    SH=4000;
else
    SH=2500;
end
pois=0.3;
E=31908302,36;
rho_st=0.2745674192;
alpha=5.5e-6;
Sc=20000;
case 6
if dtemp<100
    SH=23300;
elseif dtemp>=100 & dtemp<200
    SH=23000;
elseif dtemp>=200 & dtemp<300
    SH=22000;
elseif dtemp>=300 & dtemp<400
    SH=21200;
elseif dtemp>=400 & dtemp<500
    SH=20500;
elseif dtemp>=500 & dtemp<600
    SH=19600;

```

```
elseif dtemp>=600 & dtemp<650
    SH=18400;
elseif dtemp>=650 & dtemp<700
    SH=17800;
elseif dtemp>=700 & dtemp<750
    SH=17200;
elseif dtemp>=750 & dtemp<800
    SH=14800;
elseif dtemp>=800 & dtemp<850
    SH=12000;
elseif dtemp>=850 & dtemp<900
    SH=9300;
elseif dtemp>=900 & dtemp<950
    SH=6700;
elseif dtemp>=950 & dtemp<1000
    SH=4000;
else
    SH=2500;
end
pois=0.3;
E=29732736.29;
rho_st=0.2817928776;
alpha=7.3e-6;
Sc=23300;
case 7
    if dtemp<200
        SH=38700;
    elseif dtemp>=200 & dtemp<300
        SH=38500;
    elseif dtemp>=300 & dtemp<400
```

```

    SH=36400;
elseif dtemp>=400 & dtemp<500
    SH=35100;
elseif dtemp>=500 & dtemp<600
    SH=34500;
elseif dtemp>=600 & dtemp<650
    SH=34500;

else
    SH=NA;
end
pois=0.3;
E=29007547.6;
rho_st=0.2817928776;
alpha=5.5e-6;
Sc=38700;
case 8
if dtemp<400
    SH=20000;
elseif dtemp>=400 & dtemp<500
    SH=20000;
elseif dtemp>=500 & dtemp<600
    SH=19300;
elseif dtemp>=600 & dtemp<650
    SH=18500;
elseif dtemp>=650 & dtemp<700
    SH=18200;
elseif dtemp>=700 & dtemp<750
    SH=17900;
elseif dtemp>=750 & dtemp<800

```

```

    SH=17700;
elseif dtemp>=800 & dtemp<850
    SH=17400;
elseif dtemp>=850 & dtemp<900
    SH=17200;
elseif dtemp>=900 & dtemp<950
    SH=16900;
elseif dtemp>=950 & dtemp<1000
    SH=15900;
else
    SH=9900;
end
pois=0.3;
E=28282358.91;
rho_st=0.289018336;
alpha=9.4e-6;
Sc=20000;
case 9
if dtemp<400
    SH=20000;
elseif dtemp>=400 & dtemp<500
    SH=19900;
elseif dtemp>=500 & dtemp<600
    SH=19000;
elseif dtemp>=600 & dtemp<650
    SH=17900;
elseif dtemp>=650 & dtemp<700
    SH=17300;
elseif dtemp>=700 & dtemp<750
    SH=16700;

```

```

elseif dtemp>=750 & dtemp<800
    SH=13900;
elseif dtemp>=800 & dtemp<850
    SH=11400;
elseif dtemp>=850 & dtemp<900
    SH=8700;
elseif dtemp>=900 & dtemp<950
    SH=5900;
elseif dtemp>=950 & dtemp<1000
    SH=4000;
else
    SH=2500;
end
pois=0.3;
E=31183113.67;
rho_st=0.2854056068;
alpha=7.3e-6;
Sc=20000;
case 10
    Otrosmat;
end

```

*% --- Executes on button press in pushbutton5.*

```
function pushbutton5_Callback(hObject, eventdata, handles)
```

```
global Dext Dint P dtemp bound MSLw MSLt MSE MDw MAtem MAtot MRc MRtem
MRt pois E rho_st alpha W M Sc SH
```

```
[coordx,coordy,coordz,Pc]=GRAFICAR(M);
```

```

Gs= E/(2*(1+pois));
gr=32.2;
A_tr=(pi/4)*((Dext^2)-(Dint^2));
A_t=(pi/4)*(Dint^2);
W=A_t*gr*rho_st/100;
I_y=(((Dext^4)-(Dint^4))/64)*pi;
I_z=(((Dext^4)-(Dint^4))/64)*pi;
K_v= (((Dext^4)-(Dint^4))/32)*pi;    %A_tr*rho_st

```

```

ep=[E Gs A_tr I_y I_z K_v alpha];

```

%%MATRIZ DE NODOS

```

Nno1=size(coordx);    %Numero de nodos

```

```

Nno=Nno1(1,2);

```

```

for i = 1:Nno

```

```

    Nn1(i) = i;        %Numero de nodo

```

```

    Xn1(i) = (Nn1(i)*6)-5;

```

```

    Yn1(i) = (6*Nn1(i))-4;

```

```

    Zn1(i) = (6*Nn1(i))-3;

```

```

    Mxn1(i) = (6*Nn1(i))-2;

```

```

    Myn1(i) = (6*Nn1(i))-1;

```

```

    Mzn1(i) = 6*Nn1(i);

```

```

end

```

```

Nn = transpose(Nn1) ;    %%numero del nodo

```

```

Xn = transpose(Xn1);    %%grado de libertad en X

```

```

Yn = transpose(Yn1);    %%grado de libertad en Y

```

```

Zn = transpose(Zn1);    %%grado de libertad en Z

```

```

Mxn = transpose(Mxn1); %%grado de libertad en Mx
Myn = transpose(Myn1); %%grado de libertad en My
Mzn = transpose(Mzn1); %%grado de libertad en Mz

Nodos = [Nn Xn Yn Zn Mxn Myn Mzn]; %% matriz de nodos

```

```

%%MATRIZ EDOF

```

```

for j = 1:Nno-1
    N1(j) = j;
    A1(j) = Nodos(j,2);
    B1(j) = Nodos(j,3);
    C1(j) = Nodos(j,4);
    D1(j) = Nodos(j,5);
    E1(j) = Nodos(j,6);
    F1(j) = Nodos(j,7);
    G1(j) = Nodos(j+1,2);
    H1(j) = Nodos(j+1,3);
    I1(j) = Nodos(j+1,4);
    J1(j) = Nodos(j+1,5);
    K1(j) = Nodos(j+1,6);
    L1(j) = Nodos(j+1,7);

```

```

end

```

```

NN = transpose(N1);
AA = transpose(A1);
BB = transpose(B1);
CC = transpose(C1);
DD = transpose(D1);
EE = transpose(E1);

```

```

FF = transpose(F1);
GG = transpose(G1);
HH = transpose(H1);
II = transpose(I1);
JJ = transpose(J1);
KK = transpose(K1);
LL = transpose(L1);

```

```

Edof = [NN AA BB CC DD EE FF GG HH II JJ KK LL];

```

```

%UBICACION DE LOS NUEVOS NODOS

```

```

C7=size(M);
Cpv=size(Pc);
for i=2:Cpv(1,2)
    Pp(i)=M(i,4);
    CN(1)=1;
    if C7(1,1)==2
        C(i)=CN(i-1)+Pc(1,i)-1;
    else
        if Pp(i)==0
            C(i)=CN(i-1)+Pc(1,i);
        else
            C(i)=CN(i-1)+Pc(1,i)-1;
        end
    end
    CN=[CN,C(i)];
end

```

```

C1=size(CN);
V = zeros(Nodos(end,1),1);

for i=1:C1(1,2)
    V(CN(1,i),1)=M(i,1);

end
Nodos = [Nodos V];

bc = [];
C2 = size(bound);
C3 = size(Nodos);
b1=[];b2=[];b3=[];b4=[];b5=[];b6=[];
b11=[];b12=[];b13=[];b14=[];b15=[];b16=[];

for i = 1:C2(1,1)
    for j = 1:C3(1,1)
        if bound(i,1)==Nodos(j,8)
            b1 = Nodos(j,2);
            b2 = Nodos(j,3);
            b3 = Nodos(j,4);
            b4 = Nodos(j,5);
            b5 = Nodos(j,6);
            b6 = Nodos(j,7);

            b11 = bound(i,2);
            b12 = bound(i,3);
            b13 = bound(i,4);
            b14 = bound(i,5);

```

```

        b15 = bound(i,6);
        b16 = bound(i,7);

    end

end

bc = [bc; b1 b11; b2 b12; b3 b13; b4 b14; b5 b15; b6 b16];
end

K=zeros(Edof(end,end)); f=zeros(Edof(end,end),1); ftmp=zeros(Edof(end,end),1);
Kt=zeros(Edof(end,end)); ft=zeros(Edof(end,end),1);

ex=[coordx(1:end-1)',coordx(2:end)'];
ey=[coordy(1:end-1)',coordy(2:end)'];
ez=[coordz(1:end-1)',coordz(2:end)'];

%CARGA ESTATICA

C4=size(Edof);

for ii=1:C4(1,1)
    [Ke1,fe1]=beam3ee(ex(ii,:),ey(ii,:),ez(ii,:),ep,W);
    [K, f]=assem2e(Edof(ii,:),K,Ke1,f,fe1);
end

[a,r]=solveqt(K,f,bc);
DsW=a;
RcW=r;

```

```

Ed=extract(Edof,a);
Fie=[];
for iii=1:C4(1,1)
    es=beam3ss(ex(iii,:),ey(iii,:),ez(iii,:),W,ep,Ed(iii,:));
    Fie=[Fie;es];
end

%CARGA TERMICA
for jj=1:C4(1,1)
    [Ke,fe]=beam3et(ex(jj,:),ey(jj,:),ez(jj,:),ep,W,dtemp);
    [K,f]=assem2e(Edof(jj,:),K,Ke,f,fe);
end
[a,r]=solveqt(K,f,bc);

aaa=a-DsW;

Edt=extract(Edof,aaa);
Fit=[];
for iii=1:C4(1,1)
    es=beam3sf(ex(iii,:),ey(iii,:),ez(iii,:),dtemp,ep,Edt(iii,:));
    Fit=[Fit;es];
end

%MATRIZ DE FUERZAS INTERNAS POR CARGA SOSTENIDA
Fiee=[];
C5=size(Fie);
for i = 1:C5(1,1)

```

```

if rem(i,2)~=0
    Fiee=[Fiee;Fie(i,:)];
elseif i==C5(1,1)
    Fiee=[Fiee;Fie(i,:)];
end
end

```

**%MATRIZ DE FUERZAS INTERNAS POR EXPANSION**

```

Fite=[];
C6=size(Fit);
for i = 1:C6(1,1)

    if rem(i,2)~=0
        Fite=[Fite;Fit(i,:)];
    elseif i==C6(1,1)
        Fite=[Fite;Fit(i,:)];
    end
end

```

```

Fitt=Fite-Fiee;

```

**%ESFUERZOS**

**%MODULO DE LA SECCION**

```

Zs=(pi*(Dext^2-Dint^2))/Dext; %Zs:modulo de la seccion

```

```

SA=1.25*Sc+0.25*SH;

```

```

SLw=[];

```

```

SE=[];

```

```

    %MATRICES DE RESULTADOS
    %CARGA
    %DESPLAZAMIENTOS POR CARGA
    Dw=[];
    for i=1:C1(1,2)
        beta=(6*CN(1,i));
        Dw1=DsW(beta-5,1);
        Dw2=DsW(beta-4+1,1);
        Dw3=DsW(beta-3,1);
        Dw4=DsW(beta-2,1);
        Dw5=DsW(beta-1,1);
        Dw6=DsW(beta,1);
        Dw=[Dw;Dw1 Dw2 Dw3 Dw4 Dw5 Dw6];
    end
    MDw=[M(:,1) Dw];

    %REACCIONES POR CARGA
    Rc=[];
    for i=1:C2(1,1)
        omega=(i*6);
        Rcw6=RcW(bc(omega,1),1);
        Rcw5=RcW(bc(omega-1,1),1);
        Rcw4=RcW(bc(omega-2,1),1);
        Rcw3=RcW(bc(omega-3,1),1);
        Rcw2=RcW(bc(omega-4,1),1);
        Rcw1=RcW(bc(omega-5,1),1);
        Rc=[Rc;Rcw1 Rcw2 Rcw3 Rcw4 Rcw5 Rcw6];
    end
    MRc=[bound(:,1) Rc];

```

```
%TOTALES
```

```
%DESPLAZAMIENTOS TOTALES
```

```
Atotal= []; %Desplazamientos totales
```

```
for i=1:C1(1,2)
```

```
    Oo=(6*CN(1,i));
```

```
    Atot1=a(Oo-5,1);
```

```
    Atot2=a(Oo-4,1);
```

```
    Atot3=a(Oo-3,1);
```

```
    Atot4=a(Oo-2,1);
```

```
    Atot5=a(Oo-1,1);
```

```
    Atot6=a(Oo,1);
```

```
    Atotal=[Atotal;Atot1 Atot2 Atot3 Atot4 Atot5 Atot6];
```

```
end
```

```
MAtot=[M(:,1),Atotal];
```

```
%REACCIONES TOTALES
```

```
Rt=[];
```

```
for i=1:C2(1,1)
```

```
    omega=i*6;
```

```
    Rt6=r(bc(omega),1);
```

```
    Rt5=r(bc(omega)-1,1);
```

```
    Rt4=r(bc(omega)-2,1);
```

```
    Rt3=r(bc(omega)-3,1);
```

```
    Rt2=r(bc(omega)-4,1);
```

```
    Rt1=r(bc(omega)-5,1);
```

```
    Rt=[Rt;Rt1 Rt2 Rt3 Rt4 Rt5 Rt6];
```

```

end
MRt=[bound(:,1),Rt];
%EXPANSION
%DESPLAZAMIENTO

Atem=Atotal-Dw;
MAtem=[M(:,1) Atem];

%REACCIONES
Rtem=Rt-Rc;
MRtem=[bound(:,1),Rtem];

%ESFUERZOS
for i=1:C1(1,2)

    Mw=sqrt(((Fiee(CN(1,i),5))^2)+((Fiee(CN(1,i),6))^2));
    SLwe=Mw/Zs;
    SLw=[SLwe;SLw];

    if C7(1,1)==2
        if C2(1,1)==1
            SE=[0 SA;0 SA];
        else
            if coordx(1,1)~=coordx(1,end)
                SEe=Rtem(1,1)/A_tr;
                SE=[SEe SA;SEe SA];
            elseif coordy(1,1)~=coordy(1,end)
                SEe=Rtem(1,2)/A_tr;
                SE=[SEe SA;SEe SA];
            else

```

```

        SEe=Rtem(1,3)/A_tr;
        SE=[SEe SA;SEe SA];
    end
end
else
    St=Fitt(CN(1,i),4)/(2*Zs);
    Sb=sqrt((((Fitt(CN(1,i),5))^2)+((Fitt(CN(1,i),6))^2))/Zs);
    SEe=sqrt(Sb^2+(4*(St^2)));
    SE=[SE;SEe SA];

end
end

MSLw=[M(:,1) SLw];

for iv=1:C7(1,1)
    if C7(1,1)>3

        if iv==1 | iv==C7(1,1)

            SEE(iv)=SE(iv,1)/4.6;
        else

            SEE(iv)=SE(iv,1)/4;
        end
    else

        SEE(iv)=SE(iv,1);
    end
end

```

```

end

MSE=[M(:,1) SEE' SE(:,2)];
Sp=(P*Dext)/(4*((Dext-Dint)/2));

SLt=[];
for i=1:C1(1,2)
    if i==1 | i==C7(1,1)
        SLtw=Sp+SLw(i,1)/4;
        SLt=[SLt;SLtw SH];
    else
        SLtw=Sp+SLw(i,1);
        SLt=[SLt;SLtw SH];
    end
end
MSLt=[M(:,1) SLt];

Resrea(MRc,MRtem,MRt);
Resdes(MDw,MAtem,MAtot);
Resesf(MSLw,MSLt,MSE);

```

## Anexo H. Función Graficar

### FUNCIÓN GRAFICAR

```
function [coordx,coordy,coordz,Pc]=GRAFICAR(M)
```

```
N = size(M);
```

```
n = N(1,1);
```

```
X=[];Y=[];Z=[];Pp=[];
```

```
for i=2:n
```

```
    Pp(i)=M(i,5);
```

```
    X(1)=0;
```

```
    Y(1)=0;
```

```
    Z(1)=0;
```

```
    if Pp(i)==0
```

```
        X(i) = M(i,2)+X(i-1);
```

```
        Y(i)= M(i,3)+Y(i-1);
```

```
        Z(i)= M(i,4)+Z(i-1);
```

```
    elseif Pp(i)==1
```

```
        if M(i,2)~=0
```

```
            R(i)= sqrt((M(i,2))^2);
```

```
            if Y(i-1)-Y(i-2)>0 & M(i,2)<0
```

```
Y(i)=Y(i-1)+R(i);
X(i)=X(i-1);
Z(i)=Z(i-1)-R(i);
elseif Y(i-1)-Y(i-2)>0 & M(i,2)>0
```

```
Y(i)=Y(i-1)+R(i);
X(i)=X(i-1);
Z(i)=Z(i-1)+R(i);
```

```
elseif Y(i-1)-Y(i-2)<0 & M(i,2)<0
```

```
X(i)=X(i-1);
Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1)+R(i);
```

```
elseif Y(i-1)-Y(i-2)<0 & M(i,2)>0
```

```
X(i)=X(i-1);
Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1)-R(i);
```

```
elseif Z(i-1)-Z(i-2)>0 & M(i,2)<0
```

```
X(i)=X(i-1);
Y(i)=Y(i-1)+R(i);
Z(i)=Z(i-1)+R(i);
```

```
elseif Z(i-1)-Z(i-2)>0 & M(i,2)>0
```

```
X(i)=X(i-1);
Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1)+R(i);
```

```
elseif Z(i-1)-Z(i-2)<0 & M(i,2)<0
```

```
X(i)=X(i-1);  
Y(i)=Y(i-1)-R(i);  
Z(i)=Z(i-1)-R(i);
```

```
elseif Z(i-1)-Z(i-2)<0 & M(i,2)>0
```

```
X(i)=X(i-1);  
Y(i)=Y(i-1)+R(i);  
Z(i)=Z(i-1)-R(i);
```

```
end
```

```
elseif M(i,3)~=0
```

```
R(i)=sqrt((M(i,3))^2);
```

```
if X(i-1)-X(i-2)>0 & M(i,3)<0
```

```
X(i)=X(i-1)+R(i);  
Y(i)=Y(i-1);  
Z(i)=Z(i-1)+R(i);
```

```
elseif X(i-1)-X(i-2)>0 & M(i,3)>0
```

```
X(i)=X(i-1)+R(i);  
Y(i)=Y(i-1);  
Z(i)=Z(i-1)-R(i);
```

```
elseif X(i-1)-X(i-2)<0 & M(i,3)<0
```

```
X(i)=X(i-1)-R(i);  
Y(i)=Y(i-1);  
Z(i)=Z(i-1)-R(i);
```

elseif  $X(i-1)-X(i-2)<0$  &  $M(i,3)>0$

$X(i)=X(i-1)-R(i);$

$Y(i)=Y(i-1);$

$Z(i)=Z(i-1)+R(i);$

elseif  $X(i-1)-X(i-2)>0$  &  $M(i,3)<0$

$X(i)=X(i-1)-R(i);$

$Y(i)=Y(i-1);$

$Z(i)=Z(i-1)+R(i);$

elseif  $Z(i-1)-Z(i-2)>0$  &  $M(i,3)>0$

$X(i)=X(i-1)+R(i);$

$Y(i)=Y(i-1);$

$Z(i)=Z(i-1)+R(i);$

elseif  $Z(i-1)-Z(i-2)<0$  &  $M(i,3)<0$

$X(i)=X(i-1)+R(i);$

$Y(i)=Y(i-1);$

$Z(i)=Z(i-1)-R(i);$

elseif  $Z(i-1)-Z(i-2)<0$  &  $M(i,3)>0$

$X(i)=X(i-1)-R(i);$

$Y(i)=Y(i-1);$

$Z(i)=Z(i-1)-R(i);$

end

elseif  $M(i,4)\neq 0$

$R(i)=\sqrt{(M(i,4))^2};$

if  $X(i-1)-X(i-2)>0$  &  $M(i,4)<0$

$X(i)=X(i-1)+R(i);$

```

Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1);
elseif X(i-1)-X(i-2)>0 & M(i,4)>0
X(i)=X(i-1)+R(i);
Y(i)=Y(i-1)+R(i);
Z(i)=Z(i-1);

elseif X(i-1)-X(i-2)<0 & M(i,4)<0

X(i)=X(i-1)-R(i);
Y(i)=Y(i-1)+R(i);
Z(i)=Z(i-1);
elseif X(i-1)-X(i-2)<0 & M(i,4)>0
X(i)=X(i-1)-R(i);
Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1);
elseif Y(i-1)-Y(i-2)<0 & M(i,4)<0
X(i)=X(i-1)-R(i);
Y(i)=Y(i-1)-R(i);
Z(i)=Z(i-1);
elseif Y(i-1)-Y(i-2)>0 & M(i,4)>0
X(i)=X(i-1)-R(i);
Y(i)=Y(i-1)+R(i);
Z(i)=Z(i-1);
elseif Y(i-1)-Y(i-2)>0 & M(i,4)<0
X(i)=X(i-1)+R(i);
Y(i)=Y(i-1)+R(i);
Z(i)=Z(i-1);

elseif Y(i-1)-Y(i-2)<0 & M(i,4)>0

```

```
X(i)=X(i-1)+R(i);
```

```
Y(i)=Y(i-1)-R(i);
```

```
Z(i)=Z(i-1);
```

```
end
```

```
end
```

```
end
```

```
end
```

```
Pipe = M(:,5)';
```

```
Mc = [X; Y; Z;Pipe];
```

```
% DISCRETIZACION DE LA TUBERIA
```

```
coordx=[];
```

```
coordy=[];
```

```
coordz=[];
```

```
for i=2:n
```

```
    Xx(1) = Mc(1,i-1);
```

```
    Yy(1) = Mc(2,i-1);
```

```
    Zz(1) = Mc(3,i-1);
```

```
    Pp(i) = Mc(4,i);
```

```
    if Pp(i) == 0
```

```
if Mc(1,i) - Mc(1,i-1) ~= 0
```

```
    Xxl(i) = Mc(1,i) - Mc(1,i-1);
```

```
    Pxl(i) = sqrt(Xxl(i)^2)/2;
```

```
    Pc(i) = round(Pxl(i))+1;
```

```
    X1(i) = Xxl(i)/Pc(i);
```

```
    for j = 2:Pc(i)
```

```
        Xx(j) = Xx(j-1)+X1(i);
```

```
        Yy(j) = Mc(2,i-1);
```

```
        Zz(j) = Mc(3,i-1);
```

```
    end
```

```
    coordx=[coordx,Xx];
```

```
    coordy=[coordy,Yy];
```

```
    coordz=[coordz,Zz];
```

```
    Xx=0;
```

```
    Yy=0;
```

```
    Zz=0;
```

```
elseif Mc(2,i) - Mc(2,i-1) ~= 0
```

```
    Yy1(i) = Mc(2,i) - Mc(2,i-1);
```

```
    Pyl(i) = sqrt(Yy1(i)^2)/2;
```

```
    Pc(i) = round(Pyl(i))+1;
```

```
    Y1(i) = Yy1(i)/Pc(i);
```

```
    for j = 2:Pc(i)
```

```
        Xx(j) = Mc(1,i-1);
```

```
Yy(j) = Yy(j-1)+Y1(i);  
Zz(j) = Mc(3,i-1);
```

```
end
```

```
coordx=[coordx,Xx];  
coordy=[coordy,Yy];  
coordz=[coordz,Zz];  
Xx=0;  
Yy=0;  
Zz=0;
```

```
else
```

```
Zz1(i) = Mc(3,i) - Mc(3,i-1);  
Pzl(i)= sqrt(Zz1(i)^2)/2;  
Pc(i) = round(Pzl(i))+1;  
Z1(i) = Zz1(i)/Pc(i);
```

```
for j = 2:Pc(i)
```

```
Xx(j) = Mc(1,i-1);  
Yy(j) = Mc(2,i-1);  
Zz(j) = Zz(j-1)+Z1(i);
```

```
end
```

```
coordx=[coordx,Xx];  
coordy=[coordy,Yy];  
coordz=[coordz,Zz];  
Xx=0;  
Yy=0;
```

```
Zz=0;  
end
```

```
% DISCRETIZACION DEL CODO
```

```
elseif Pp(i)==1  
    if M(i,2) ~= 0  
  
        R(i)= sqrt((M(i,2))^2);  
  
        Pcl(i) = pi*R(i)/2;  
        Pc(i) = round(Pcl(i))+1;  
  
        if Mc(2,i-1)-Mc(2,i-2)>0 & M(i,2)>0  
  
            for j = 1:Pc(i)  
                theta(j) = (j-1)*pi/(2*Pc(i));  
                A(j)=0;  
                B(j)=R(i)*cos(theta(j));  
                C(j)=R(i)*sin(theta(j));  
            end  
  
            Re=[1 0 0;0 0 1;0 -1 0];  
            D=[A; B; C];  
            T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)+R(i)];  
  
            codo=Re*D+T;  
            Xx=codo(1,:);  
            Yy=codo(2,:);  
            Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
coordy=[coordy, Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)>0 & M(i,2)<0
```

```
for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=0;
B(j)=R(i)*sin(theta(j));
C(j)=R(i)*cos(theta(j));
end
```

```
D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)-R(i)];
```

```
codo=D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy, Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)<0 & M(i,2)<0
```

```
for j = 1:Pc(i)  
theta(j) = (j-1)*pi/(2*Pc(i));  
A(j)=0;  
B(j)=-R(i)*sin(theta(j));  
C(j)=-R(i)*cos(theta(j));  
end
```

```
D=[A; B; C];  
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)+R(i)];
```

```
codo=D+T;  
Xx=codo(1,:);  
Yy=codo(2,:);  
Zz=codo(3,:);  
coordx=[coordx,Xx];  
coordy=[coordy,Yy];  
coordz=[coordz,Zz];  
Xx=0;  
Yy=0;  
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)<0 & M(i,2)>0
```

```
for j = 1:Pc(i)  
theta(j) = (j-1)*pi/(2*Pc(i));  
A(j)=0;  
B(j)=R(i)*cos(theta(j));  
C(j)=R(i)*sin(theta(j));
```

end

```
Re=[1 0 0;0 0 -1;0 1 0];
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)-R(i)];
```

```
codo=Re*D+T;
```

```
Xx=codo(1,:);
```

```
Yy=codo(2,:);
```

```
Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
```

```
coordy=[coordy,Yy];
```

```
coordz=[coordz,Zz];
```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)>0 & M(i,2)>0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=0;
```

```
B(j)=R(i)*cos(theta(j));
```

```
C(j)=R(i)*sin(theta(j));
```

```
end
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1); Mc(2,i-1)-R(i); Mc(3,i-1)];
```

```
codo=D+T;
```

```

Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coor dx=[coor dx,Xx];
coor dy=[coor dy,Yy];
coor dz=[coor dz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```

elseif Mc(3,i-1)-Mc(3,i-2)>0 & M(i,2)<0

```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=0;
B(j)=R(i)*sin(theta(j));
C(j)=R(i)*cos(theta(j));
end

```

```

Re=[1 0 0;0 0 -1;0 1 0];
D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1)+R(i); Mc(3,i-1)];

```

```

codo=Re*D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coor dx=[coor dx,Xx];
coor dy=[coor dy,Yy];
coor dz=[coor dz,Zz];

```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)<0 & M(i,2)>0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=0;
```

```
B(j)=-R(i)*cos(theta(j));
```

```
C(j)=-R(i)*sin(theta(j));
```

```
end
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1); Mc(2,i-1)+R(i); Mc(3,i-1)];
```

```
codo=D+T;
```

```
Xx=codo(1,:);
```

```
Yy=codo(2,:);
```

```
Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
```

```
coordy=[coordy,Yy];
```

```
coordz=[coordz,Zz];
```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)<0 & M(i,2)<0
```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=0;
B(j)=R(i)*sin(theta(j));
C(j)=R(i)*cos(theta(j));
end

Re=[1 0 0;0 0 1;0 -1 0];
D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1)-R(i); Mc(3,i-1)];

codo=Re*D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;
end
elseif M(i,3) ~= 0

R(i)= sqrt((M(i,3))^2);

Pcl(i) = pi*R(i)/2;
Pc(i) = round(Pcl(i))+1;
if Mc(1,i-1)-Mc(1,i-2)>0 & M(i,3)>0

```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=R(i)*sin(theta(j));
B(j)=0;
C(j)=R(i)*cos(theta(j));
end

```

```

D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)-R(i)];

```

```

codo=D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```

elseif Mc(1,i-1)-Mc(1,i-2)>0 & M(i,3)<0

```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=R(i)*cos(theta(j));
B(j)=0;
C(j)=R(i)*sin(theta(j));
end

```

```

Re=[0 0 1;0 1 0;-1 0 0];
D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)+R(i)];

```

```

codo=Re*D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```

elseif Mc(1,i-1)-Mc(1,i-2)<0 & M(i,3)<0

```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=R(i)*cos(theta(j));
B(j)=0;
C(j)=R(i)*sin(theta(j));
end

```

```

Re=[0 0 -1;0 1 0;1 0 0];
D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)-R(i)];

```

```

codo=Re*D+T;
Xx=codo(1,:);

```

```

Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```
elseif Mc(1,i-1)-Mc(1,i-2)<0 & M(i,3)>0
```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=-R(i)*sin(theta(j));
B(j)=0;
C(j)=-R(i)*cos(theta(j));
end

```

```

D=[A; B; C];
T=[Mc(1,i-1); Mc(2,i-1); Mc(3,i-1)+R(i)];

```

```

codo=D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;

```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)>0 & M(i,3)>0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=R(i)*sin(theta(j));
```

```
B(j)=0;
```

```
C(j)=R(i)*cos(theta(j));
```

```
end
```

```
Re=[0 0 -1;0 1 0;1 0 0];
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1)+R(i); Mc(2,i-1); Mc(3,i-1)];
```

```
codo=Re*D+T;
```

```
Xx=codo(1,:);
```

```
Yy=codo(2,:);
```

```
Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
```

```
coordy=[coordy,Yy];
```

```
coordz=[coordz,Zz];
```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)>0 & M(i,3)<0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));  
A(j)=R(i)*cos(theta(j));  
B(j)=0;  
C(j)=R(i)*sin(theta(j));  
end
```

```
D=[A; B; C];  
T=[Mc(1,i-1)-R(i); Mc(2,i-1); Mc(3,i-1)];
```

```
codo=D+T;  
Xx=codo(1,:);  
Yy=codo(2,:);  
Zz=codo(3,:);  
coordx=[coordx,Xx];  
coordy=[coordy,Yy];  
coordz=[coordz,Zz];  
Xx=0;  
Yy=0;  
Zz=0;
```

```
elseif Mc(3,i-1)-Mc(3,i-2)<0 & M(i,3)>0
```

```
for j = 1:Pc(i)  
theta(j) = (j-1)*pi/(2*Pc(i));  
A(j)=R(i)*sin(theta(j));  
B(j)=0;  
C(j)=R(i)*cos(theta(j));  
end
```

```

Re=[0 0 1;0 1 0;-1 0 0];
D=[A; B; C];
T=[Mc(1,i-1)-R(i); Mc(2,i-1); Mc(3,i-1)];

```

```

codo=Re*D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```

elseif Mc(3,i-1)-Mc(3,i-2)<0 & M(i,3)<0

```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=-R(i)*sin(theta(j));
B(j)=0;
C(j)=-R(i)*cos(theta(j));
end

```

```

D=[A; B; C];
T=[Mc(1,i-1)+R(i); Mc(2,i-1); Mc(3,i-1)];

```

```

codo=D+T;
Xx=codo(1,:);

```

```

    Yy=codo(2,:);
    Zz=codo(3,:);
    coordx=[coordx,Xx];
    coordy=[coordy,Yy];
    coordz=[coordz,Zz];
    Xx=0;
    Yy=0;
    Zz=0;
end
elseif M(i,4) ~= 0

    R(i)= sqrt((M(i,4))^2);

    Pcl(i) = pi*R(i)/2;
    Pc(i) = round(Pcl(i))+1;
    if Mc(1,i-1)-Mc(1,i-2)>0 & M(i,4)>0

        for j = 1:Pc(i)
            theta(j) = (j-1)*pi/(2*Pc(i));
            A(j)=R(i)*cos(theta(j));
            B(j)=R(i)*sin(theta(j));
            C(j)=0;
        end

        Re=[0 1 0;-1 0 0;0 0 1];
        D=[A; B; C];
        T=[Mc(1,i-1); Mc(2,i-1)+R(i); Mc(3,i-1)];

        codo=Re*D+T;
        Xx=codo(1,:);

```

```

    Yy=codo(2,:);
    Zz=codo(3,:);
    coordx=[coordx,Xx];
    coordy=[coordy,Yy];
    coordz=[coordz,Zz];
    Xx=0;
    Yy=0;
    Zz=0;

```

```

elseif Mc(1,i-1)-Mc(1,i-2)>0 & M(i,4)<0

```

```

    for j = 1:Pc(i)
        theta(j) = (j-1)*pi/(2*Pc(i));
        A(j)=R(i)*sin(theta(j));
        B(j)=R(i)*cos(theta(j));
        C(j)=0;
    end

```

```

    D=[A; B; C];
    T=[Mc(1,i-1); Mc(2,i-1)-R(i); Mc(3,i-1)];

```

```

    codo=D+T;
    Xx=codo(1,:);
    Yy=codo(2,:);
    Zz=codo(3,:);
    coordx=[coordx,Xx];
    coordy=[coordy,Yy];
    coordz=[coordz,Zz];
    Xx=0;
    Yy=0;

```

```
Zz=0;
```

```
elseif Mc(1,i-1)-Mc(1,i-2)<0 & M(i,4)<0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=-R(i)*sin(theta(j));
```

```
B(j)=-R(i)*cos(theta(j));
```

```
C(j)=0;
```

```
end
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1); Mc(2,i-1)+R(i); Mc(3,i-1)];
```

```
codo=D+T;
```

```
Xx=codo(1,:);
```

```
Yy=codo(2,:);
```

```
Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
```

```
coordy=[coordy,Yy];
```

```
coordz=[coordz,Zz];
```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(1,i-1)-Mc(1,i-2)<0 & M(i,4)>0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=R(i)*cos(theta(j));
```

```
B(j)=R(i)*sin(theta(j));
```

```
C(j)=0;
```

```
end
```

```
Re=[0 -1 0;1 0 0;0 0 1];
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1); Mc(2,i-1)-R(i); Mc(3,i-1)];
```

```
codo=Re*D+T;
```

```
Xx=codo(1,:);
```

```
Yy=codo(2,:);
```

```
Zz=codo(3,:);
```

```
coordx=[coordx,Xx];
```

```
coordy=[coordy,Yy];
```

```
coordz=[coordz,Zz];
```

```
Xx=0;
```

```
Yy=0;
```

```
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)>0 & M(i,4)>0
```

```
for j = 1:Pc(i)
```

```
theta(j) = (j-1)*pi/(2*Pc(i));
```

```
A(j)=R(i)*cos(theta(j));
```

```
B(j)=R(i)*sin(theta(j));
```

```
C(j)=0;
```

```
end
```

```
D=[A; B; C];
```

```
T=[Mc(1,i-1)-R(i); Mc(2,i-1); Mc(3,i-1)];
```

```

codo=D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];
coordy=[coordy,Yy];
coordz=[coordz,Zz];
Xx=0;
Yy=0;
Zz=0;

```

```
elseif Mc(2,i-1)-Mc(2,i-2)>0 & M(i,4)<0
```

```

for j = 1:Pc(i)
theta(j) = (j-1)*pi/(2*Pc(i));
A(j)=R(i)*sin(theta(j));
B(j)=R(i)*cos(theta(j));
C(j)=0;
end

```

```

Re=[0 -1 0;1 0 0;0 0 1];
D=[A; B; C];
T=[Mc(1,i-1)+R(i); Mc(2,i-1); Mc(3,i-1)];

```

```

codo=Re*D+T;
Xx=codo(1,:);
Yy=codo(2,:);
Zz=codo(3,:);
coordx=[coordx,Xx];

```

```
coordy=[coordy, Yy];  
coordz=[coordz, Zz];  
Xx=0;  
Yy=0;  
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)<0 & M(i,4)>0
```

```
for j = 1:Pc(i)  
theta(j) = (j-1)*pi/(2*Pc(i));  
A(j)=-R(i)*cos(theta(j));  
B(j)=-R(i)*sin(theta(j));  
C(j)=0;  
end
```

```
D=[A; B; C];  
T=[Mc(1,i-1)+R(i); Mc(2,i-1); Mc(3,i-1)];
```

```
codo=D+T;  
Xx=codo(1,:);  
Yy=codo(2,:);  
Zz=codo(3,:);  
coordx=[coordx, Xx];  
coordy=[coordy, Yy];  
coordz=[coordz, Zz];  
Xx=0;  
Yy=0;  
Zz=0;
```

```
elseif Mc(2,i-1)-Mc(2,i-2)<0 & M(i,4)<0
```

```
    for j = 1:Pc(i)
```

```
        theta(j) = (j-1)*pi/(2*Pc(i));
```

```
        A(j)=R(i)*sin(theta(j));
```

```
        B(j)=R(i)*cos(theta(j));
```

```
        C(j)=0;
```

```
    end
```

```
    Re=[0 1 0;-1 0 0;0 0 1];
```

```
    D=[A; B; C];
```

```
    T=[Mc(1,i-1)-R(i); Mc(2,i-1); Mc(3,i-1)];
```

```
    codo=Re*D+T;
```

```
    Xx=codo(1,:);
```

```
    Yy=codo(2,:);
```

```
    Zz=codo(3,:);
```

```
    coordx=[coordx,Xx];
```

```
    coordy=[coordy,Yy];
```

```
    coordz=[coordz,Zz];
```

```
    Xx=0;
```

```
    Yy=0;
```

```
    Zz=0;
```

```
end
```

```
end
```

```
end
```

```
end
```

## Anexo I. Función Beam3ee

### FUNCIÓN BEAM3EE

```
function [Ke1, fe1]=beam3ee(ex,ey,ez,ep,W)
```

```
b=[ ex(2)-ex(1); ey(2)-ey(1); ez(2)-ez(1) ];  
L=sqrt(b'*b); %L=longitud del tramo  
Lx=sqrt(b(1,1)*b(1,1));  
Ly=sqrt(b(2,1)*b(2,1));  
Lz=sqrt(b(3,1)*b(3,1));  
Wt=W*L;
```

```
if ex(1)==ex(2) & ez(1)==ez(2)
```

```
    if ey(2) > ey(1)  
        eq=[-W 0 0 0];
```

```
    elseif ey(2) < ey(1)  
        eq=[W 0 0 0];
```

```
    end
```

```
elseif ey(2)==ey(1)
```

```
    eq=[0 -W 0 0];
```

```
else
```

```
    if ex(1)==ex(2)  
        omega=atan(Lz/Ly);
```

```
        if b(2,1)<0 & b(3,1)>0
```

```

Wz=-Wt*cos(omega)/Lz;
Wy=-Wt*sin(omega)/Ly;

elseif b(2,1)>0 & b(3,1)>0
Wz=-Wt*cos(omega)/Lz;
Wy=-Wt*sin(omega)/Ly;
else
Wz=-Wt*cos(omega)/Lz;
Wy=Wt*sin(omega)/Ly;

end
eq=[Wy 0 Wz 0];
else
omega=atan(Ly/Lx);
if b(1,1)<0 & b(2,1)>0
Wx=-Wt*sin(omega)/Lx;
Wy=-Wt*cos(omega)/Ly;
elseif b(1,1)>0 & b(2,1)>0
Wx=-Wt*sin(omega)/Lx;
Wy=-Wt*cos(omega)/Ly;
else
Wx=Wt*sin(omega)/Lx;
Wy=-Wt*cos(omega)/Ly;
end
eq=[Wx Wy 0 0];
end
end

if nargin==4
eq=[0 0 0 0];

```

end

```
qx=eq(1); qy=eq(2); qz=eq(3); qw=eq(4);
```

```
%
```

```
E=ep(1); Gs=ep(2);
```

```
A=ep(3);
```

```
ly=ep(4); lz=ep(5);
```

```
Kv=ep(6);
```

```
a=E*A/L ; b=12*E*lz/L^3 ; c=6*E*lz/L^2;
```

```
d=12*E*ly/L^3 ; e=6*E*ly/L^2 ; f=G*s*Kv/L;
```

```
g=2*E*ly/L ; h=2*E*lz/L ;
```

```
Kle=[ a 0 0 0 0 0 -a 0 0 0 0 0 ;
```

```
0 b 0 0 0 c 0 -b 0 0 0 c ;
```

```
0 0 d 0 -e 0 0 0 -d 0 -e 0 ;
```

```
0 0 0 f 0 0 0 0 0 -f 0 0 ;
```

```
0 0 -e 0 2*g 0 0 0 e 0 g 0 ;
```

```
0 c 0 0 0 2*h 0 -c 0 0 0 h ;
```

```
-a 0 0 0 0 0 a 0 0 0 0 0 ;
```

```
0 -b 0 0 0 -c 0 b 0 0 0 -c ;
```

```
0 0 -d 0 e 0 0 0 d 0 e 0 ;
```

```
0 0 0 -f 0 0 0 0 0 f 0 0 ;
```

```
0 0 -e 0 g 0 0 0 e 0 2*g 0 ;
```

```
0 c 0 0 0 h 0 -c 0 0 0 2*h];
```

```
%fuerza por carga en cada elemento
```

```
fce=L/2*[qx qy qz qw -1/6*qz*L 1/6*qy*L qx qy qz qw 1/6*qz*L -1/6*qy*L]';
```

```
%fuerza total en el elemento
```

```
fle=fce;
```

```

if ex(1) == ex(2) & ey(1) == ey(2)
    if ez(2) == ez(1)
        disp('cesar')
    else
        if ez(2) > ez(1) %puntos viga en Z
            Lambda = [0 0 1 ; 0 1 0 ; -1 0 0];
        else
            Lambda = [0 0 -1 ; 0 1 0 ; 1 0 0];
        end
    end
end
%elseif ex(1)==ex(2) & ey(1)==ey(2) & ez(1)==ez(2)
% Lambda=[1 0 0;0 1 0;0 0 1];
else

    CXx = (ex(2)-ex(1))/L;
    CYx = (ey(2)-ey(1))/L;
    CZx = (ez(2)-ez(1))/L;
    D = sqrt(CXx*CXx + CYx*CYx);
    CXy = -CYx/D;
    CYy = CXx/D;
    CZy = 0;
    CXz = -CXx*CZx/D;
    CYz = -CYx*CZx/D;
    CZz = D;
    Lambda = [CXx CYx CZx ;CXy CYy CZy ;CXz CYz CZz] ;

end
G = [Lambda zeros(3,9);
     zeros(3) Lambda zeros(3,6);

```

```

zeros(3,6) Lambda zeros(3);
zeros(3,9) Lambda];

Ke1=G'*Kle*G; fe1=G'*fle;
%-----
if nargout==0
    disp('Element stiffness matrix: ');
    disp(Ke1);

    if nargin==5
        disp('Element load vector: ');
        disp(fe1)
    end
    return
end
Ke=Ke1;
if ex(1)==ex(2) & ey(1)==ey(2) & ez(1)==ez(2)
    Lambda=[1 0 0;0 1 0;0 0 1];
end
if nargin==5 fe=fe1; end

```

## función ASSEM2

```

function [K,f]=assem2e(edof,K,Ke,f,fe)
[nie,n]=size(edof);
t=edof(:,2:n);
for i = 1:nie
    K(t(i,:),t(i,:)) = K(t(i,:),t(i,:))+Ke;
    if nargin==5

```

```
f(t(i,:))=f(t(i,:))+fe;  
end  
end
```

## Anexo J. Función Solveqt

### FUNCIÓN SOLVEQT

```
function [d,Q]=solveqt(K,f,bc)
if nargin==2 ;
    d=K\f ;
elseif nargin==3;
    [nd,nd]=size(K);
    fdof=[1:nd]';
%
    d=zeros(size(fdof));
    Q=zeros(size(fdof));
%
    pdof=bc(:,1);
    dp=bc(:,2);
    fdof(pdof)=[];
%
    s=K(fdof,fdof)\(f(fdof)-K(fdof,pdof)*dp);
%
    d(pdof)=dp;
    d(fdof)=s;
end
Q=K*d-f;
```

## Anexo K. Función Extract

### FUNCIÓN EXTRACT

```
function [ed]=extract(edof,a)
[nie,n]=size(edof);
%
t=edof(:,2:n);
%
for i = 1:nie
    ed(i,1:(n-1))=a(t(i,:));
end
%
```

## Anexo L. Función Beam3ss

### FUNCIÓN BEAM3SS

```
function [es,edi,eci]=beam3ss(ex,ey,ez,W,ep,ed,n)
    if nargin<=6 n=2; end;
    if nargin>5
        if ex(1)==ex(2) & ez(1)==ez(2)

            if ey(2) > ey(1)
                eq=[-W 0 0 0];
            elseif ey(2) < ey(1)
                eq=[W 0 0 0];
            end

        else
            eq=[0 -W 0 0];
        end

        qx=eq(1); qy=eq(2); qz=eq(3); qw=eq(4);
    else
        qx=0;qy=0;qz=0;qw=0;
    end
    %
    b=[ ex(2)-ex(1); ey(2)-ey(1); ez(2)-ez(1) ];
    L=sqrt(b'*b);
    %
    %
```

```

Ely=ep(1)*ep(4); Elz=ep(1)*ep(5);
EA=ep(1)*ep(3); GKv=ep(2)*ep(6);
%
if ex(1) == ex(2) & ey(2)==ey(1)
    if ez(2)>ez(1)
        An = [0 0 1; 0 1 0;-1 0 0];
    else
        An = [0 0 -1; 0 1 0;1 0 0];
    end
else
    CXx = (ex(2)-ex(1))/L;
    CYx = (ey(2)-ey(1))/L;
    CZx = (ez(2)-ez(1))/L;

    D = sqrt(CXx*CXx+CYx*CYx);

    CXy = -CYx/D;
    CYy = CXx/D;
    CZy = 0;

    CXz = (-CXx*CZx)/D;
    CYz = (-CYx*CZx)/D;
    CZz = D;

    An = [CXx CYx CZx;CXy CYy CZy;CXz CYz CZz];
end

%
G=[ An    zeros(3) zeros(3) zeros(3);
    zeros(3) An    zeros(3) zeros(3);

```

```

zeros(3) zeros(3) An zeros(3);
zeros(3) zeros(3) zeros(3) An ];
%
u=G*ed' - [0;           % u is the local element dis-
            0;           % placement vector minus the
            0;           % particular solution to the
            0;           % beam's diff.eq:s
            0;
            0;
            -qx*L^2/2/EA;
            qy*L^4/24/EIz;
            qz*L^4/24/EIy;
            -qw*L^2/2/GKv;
            -qz*L^3/6/EIy;
            qy*L^3/6/EIz];
%
C=[0 1 0 0 0 0 0 0 0 0 0;
   0 0 0 0 0 1 0 0 0 0 0;
   0 0 0 0 0 0 0 0 0 1 0 0;
   0 0 0 0 0 0 0 0 0 0 0 1;
   0 0 0 0 0 0 0 0 -1 0 0 0;
   0 0 0 0 1 0 0 0 0 0 0 0;
   L 1 0 0 0 0 0 0 0 0 0 0;
   0 0 L^3 L^2 L 1 0 0 0 0 0 0;
   0 0 0 0 0 0 L^3 L^2 L 1 0 0;
   0 0 0 0 0 0 0 0 0 0 L 1;
   0 0 0 0 0 0 -3*L^2 -2*L -1 0 0 0;
   0 0 3*L^2 2*L 1 0 0 0 0 0 0];

m=inv(C)*u;

```

```

for i=1:n
eci(i,1)=((i-1)*L/(n-1))';
x=eci(i,1);
es(i,:)=([EA 0 0 0 0 0 0 0 0 0 0;
0 0 -6*Elz 0 0 0 0 0 0 0 0;
0 0 0 0 0 0 -6*Ely 0 0 0 0 0;
0 0 0 0 0 0 0 0 0 GKv 0;
0 0 0 0 0 0 -6*Ely*x -2*Ely 0 0 0 0;
0 0 6*Elz*x 2*Elz 0 0 0 0 0 0 0];*m + [-qx*x;
-qy*x;
-qz*x;
-qw*x;
-qz*x^2/2;
qy*x^2/2]);

%
edi(i,:)=([x 1 0 0 0 0 0 0 0 0 0;
0 0 x^3 x^2 x 1 0 0 0 0 0;
0 0 0 0 0 0 x^3 x^2 x 1 0 0;
0 0 0 0 0 0 0 0 0 x 1]*m + [-qx*x^2/2/EA;
qy*x^4/24/Elz;
qz*x^4/24/Ely;
-qw*x^2/2/GKv]);

end;

```

## Anexo M. Función Beam3et

### FUNCIÓN BEAM3ET

```
function [Ke1, fe1]=beam3et(ex,ey,ez,ep,W,dtemp)
```

```
    b=[ ex(2)-ex(1); ey(2)-ey(1); ez(2)-ez(1) ];
```

```
    L=sqrt(b'*b); n1=b/L;
```

```
    if ex(1)==ex(2) & ez(1)==ez(2)
```

```
        if ey(2) > ey(1)
```

```
            eq=[-W 0 0 0];
```

```
        elseif ey(2) < ey(1)
```

```
            eq=[W 0 0 0];
```

```
        end
```

```
    else
```

```
        eq=[0 -W 0 0];
```

```
    end
```

```
    %
```

```
    if nargin==4; eq=[0 0 0 0]; end
```

```

qx=eq(1); qy=eq(2); qz=eq(3); qw=eq(4);
%
E=ep(1); Gs=ep(2);
A=ep(3);
ly=ep(4); lz=ep(5);
Kv=ep(6);

a=E*A/L ; b=12*E*lz/L^3 ; c=6*E*lz/L^2;
d=12*E*ly/L^3 ; e=6*E*ly/L^2 ; f=G*s*Kv/L;
g=2*E*ly/L ; h=2*E*lz/L ;

Kle=[ a 0 0 0 0 0 -a 0 0 0 0 0 ;
      0 b 0 0 0 c 0 -b 0 0 0 c ;
      0 0 d 0 -e 0 0 0 -d 0 -e 0 ;
      0 0 0 f 0 0 0 0 0 -f 0 0 ;
      0 0 -e 0 2*g 0 0 0 e 0 g 0 ;
      0 c 0 0 0 2*h 0 -c 0 0 0 h ;
      -a 0 0 0 0 0 a 0 0 0 0 0 ;
      0 -b 0 0 0 -c 0 b 0 0 0 -c ;
      0 0 -d 0 e 0 0 0 d 0 e 0 ;
      0 0 0 -f 0 0 0 0 0 f 0 0 ;
      0 0 -e 0 g 0 0 0 e 0 2*g 0 ;
      0 c 0 0 0 h 0 -c 0 0 0 2*h];

%fuerza por carga en cada elemento
fce=L/2*[qx qy qz qw -1/6*qz*L 1/6*qy*L qx qy qz qw 1/6*qz*L -1/6*qy*L]*0;
%fuerza por temperatura en el elemento
fte=ep(1)*ep(3)*ep(7)*dtemp.*[1;0;0;0;0;0;-1;0;0;0;0;0];
%fuerza total en el elemento
fle=fce+fte;

```

```

if ex(1) == ex(2) & ey(1) == ey(2)
    if ez(2) > ez(1)
        Lambda = [0 0 1 ; 0 1 0 ; -1 0 0];
    else
        Lambda = [0 0 -1 ; 0 1 0 ; 1 0 0];
    end
else
    CXx = (ex(2)-ex(1))/L;
    CYx = (ey(2)-ey(1))/L;
    CZx = (ez(2)-ez(1))/L;
    D = sqrt(CXx*CXx + CYx*CYx);
    CXy = -CYx/D;
    CYy = CXx/D;
    CZy = 0;
    CXz = -CXx*CZx/D;
    CYz = -CYx*CZx/D;
    CZz = D;
    Lambda = [CXx CYx CZx ;CXy CYy CZy ;CXz CYz CZz];
end
G = [Lambda zeros(3,9);
     zeros(3) Lambda zeros(3,6);
     zeros(3,6) Lambda zeros(3);
     zeros(3,9) Lambda];
Ke1=G'*Kle*G; fe1=G'*fle;
%-----
if nargout==0
    disp('Element stiffness matrix: ');
    disp(Ke1);

if nargin==6

```

```
disp('Element load vector: ');  
disp(fe1)  
end  
return  
end  
Ke=Ke1;  
  
if nargin==6 fe=fe1; end
```

## Anexo N. Función Beam3sf

### FUNCIÓN BEAM3SF

```
function [es,edi,eci]=beam3sf(ex,ey,ez,dtemp,ep,edt)
    if nargin<=6
        n=2;
    end
    if nargin>6
        qx=ep(1)*ep(3)*ep(7)*dtemp; qy=0; qz=0; qw=0;
    else
        qx=0;qy=0;qz=0;qw=0;
    end
    %
    b=[ ex(2)-ex(1); ey(2)-ey(1); ez(2)-ez(1) ];
    L=sqrt(b'*b);
    %
    %
    Ely=ep(1)*ep(4); Elz=ep(1)*ep(5);
    EA=ep(1)*ep(3); GKv=ep(2)*ep(6);
    %
    if ex(1) == ex(2) & ey(2)==ey(1)
        if ez(2)>ez(1)
            An = [0 0 1; 0 1 0;-1 0 0];
        else
            An = [0 0 -1; 0 1 0;1 0 0];
        end
    else
        CXx = (ex(2)-ex(1))/L;
```

```

CYx = (ey(2)-ey(1))/L;
CZx = (ez(2)-ez(1))/L;

D = sqrt(CXx*CXx+CYx*CYx);

CXy = -CYx/D;
CYy = CXx/D;
CZy = 0;

CXz = (-CXx*CZx)/D;
CYz = (-CYx*CZx)/D;
CZz = D;

An = [CXx CYx CZx;CXy CYy CZy;CXz CYz CZz];
end

%
G=[ An  zeros(3) zeros(3) zeros(3);
    zeros(3) An  zeros(3) zeros(3);
    zeros(3) zeros(3) An  zeros(3);
    zeros(3) zeros(3) zeros(3) An  ];

%
u=G*edt' - [0;          % u is the local element dis-
             0;          % placement vector minus the
             0;          % particular solution to the
             0;          % beam's diff.eq:s
             0;
             -qx*L^2/2/EA;
             qy*L^4/24/EIz;

```

```

    qz*L^4/24/Ely;
    -qw*L^2/2/GKv;
    -qz*L^3/6/Ely;
    qy*L^3/6/Elz];
%
C=[0 1 0 0 0 0 0 0 0 0 0;
   0 0 0 0 0 1 0 0 0 0 0;
   0 0 0 0 0 0 0 0 0 1 0;
   0 0 0 0 0 0 0 0 0 0 1;
   0 0 0 0 0 0 0 0 -1 0 0;
   0 0 0 0 1 0 0 0 0 0 0;
   L 1 0 0 0 0 0 0 0 0 0;
   0 0 L^3 L^2 L 1 0 0 0 0 0;
   0 0 0 0 0 0 L^3 L^2 L 1 0;
   0 0 0 0 0 0 0 0 0 0 L 1;
   0 0 0 0 0 0 -3*L^2 -2*L -1 0 0;
   0 0 3*L^2 2*L 1 0 0 0 0 0 0];

m=inv(C)*u;

for i=1:n
    eci(i,1)=((i-1)*L/(n-1))';
    x=eci(i,1);
    es(i,:)=([EA 0 0 0 0 0 0 0 0 0 0;
              0 0 -6*Elz 0 0 0 0 0 0 0 0;
              0 0 0 0 0 0 -6*Ely 0 0 0 0;
              0 0 0 0 0 0 0 0 0 GKv 0;
              0 0 0 0 0 0 -6*Ely*x -2*Ely 0 0 0;
              0 0 6*Elz*x 2*Elz 0 0 0 0 0 0 0])*m + [-qx*x;
                                                       -qy*x;

```

```

- qz*x;
- qw*x;
- qz*x^2/2;
  qy*x^2/2)];

%
edi(i,:) = ([x 1 0 0 0 0 0 0 0 0 0;
            0 0 x^3 x^2 x 1 0 0 0 0 0;
            0 0 0 0 0 0 x^3 x^2 x 1 0 0;
            0 0 0 0 0 0 0 0 0 x 1]*m + [-qx*x^2/2/EA;
            qy*x^4/24/EIz;
            qz*x^4/24/EIy;
            -qw*x^2/2/GKv]);

end

```