

Caracterización de las matrices de coeficientes para una superficie transversal regular isotrópica implementando el método de elementos finitos de frontera escalada en ambiente

Matlab

Jefersón David Manrique Tarazona, Camilo Andrés Osorio Pinzón

Trabajo de Grado para Optar al Título de Ingeniero mecánico

Director

Octavio Andrés González Estrada

Ingeniero Mecánico, PhD.

Codirectores

Jabid Eduardo Quiroga Méndez, PhD

Carlos Andrés Galán Pinilla, MSc

Universidad Industrial de Santander  
Facultad de ingenierías físico-mecánicas

Escuela de Ingeniería mecánica

Bucaramanga

2024

### **Dedicatoria**

Este proyecto está dedicado a mis padres, Nelson David Manrique y Nury Alexandra Tarazona, así como a mi abuela Diocelina, a mi hermano Yeison Manrique y demás familiares que me han brindado apoyo incondicionalmente a lo largo de esta etapa de mi vida, permitiéndome culminar satisfactoriamente mi desarrollo profesional.

A mis amigos y compañeros que siempre me animaron para seguir adelante en este camino a ser un profesional en esta rama de la ingeniería.

*Jeferson David Manrique Tarazona*

Dedico este proyecto de grado a mi abuela Beatriz Galvis, a mi madre Rosa Pinzón, a mi tío Benjamín Pinzón y hermana Vanesa Pinzón, quienes han sido mi fuente inagotable de inspiración y apoyo durante este viaje académico. Su aliento y sabiduría han iluminado mi camino, brindándome la fuerza y la confianza necesarias para alcanzar este logro. A mis seres queridos, amigos y mentores, gracias por ser mi constante motivación. Este trabajo es un tributo a su inquebrantable apoyo y creencia en mi capacidad. ¡Gracias por ser parte fundamental de mi éxito!

*Camilo Andrés Osorio Pinzón*

### **Agradecimiento**

Quisiéramos expresar nuestro profundo agradecimiento a todas las personas que contribuyeron al éxito de este proyecto de grado. En primer lugar, a los mentores Octavio Andrés González Estrada, Jabid Eduardo Quiroga Méndez, Carlos Andrés Galán Pinilla y Felipe Andrés Cediel Carreño, cuya orientación y conocimientos fueron fundamentales para el desarrollo de este trabajo. Agradecemos sinceramente su paciencia y dedicación.

Nuestro reconocimiento también se extiende a la Universidad Industrial de Santander, por brindar ese espacio de aprendizaje y crecimiento profesional. Agradezco a mis compañeros de clase por su colaboración y apoyo constante.

Finalmente, agradecemos a todos aquellos que, de una forma u otra, contribuyeron al desarrollo de este proyecto. Este logro no hubiera sido posible sin la valiosa ayuda de cada uno. ¡Gracias!

*Jeferson David Manrique Tarazona, Camilo Andrés Osorio Pinzón*

**Tabla de Contenido**

Introducción .....	10
Planteamiento y justificación .....	14
1. Objetivos.....	16
1.1 Objetivo General.....	16
1.2 Objetivos Específicos.....	16
2. Materiales y métodos .....	17
2.1 Formulación para una guía de ondas tridimensional mediante el método de elementos finitos de límite escalado (SBFEM). .....	17
2.2. Aplicación .....	21
2.3. Ecuación de movimiento.....	22
2.4. Algoritmo para calcular la matriz de rigidez por SBFEM.....	27
3. Resultados .....	29
3.1. Resultados numéricos .....	30
3. Conclusiones .....	34
Referencias Bibliográficas .....	34
Apéndices.....	38

**Lista de Tablas**

Tabla 1. Propiedades de las matrices de coeficientes. .... 30

**Lista de Figuras**

Figura 1. Transformación de coordenadas tridimensionales SBFEM con el centro de escala en el infinito.....	19
Figura 2. Diagrama esquemático del algoritmo propuesto para el cálculo de las matrices de coeficientes. ....	28
Figura 3. Geometría sección transversal.....	29
Figura 4. Mallado de la geometría.....	30
Figura 5. Matriz de coeficiente E1.....	31
Figura 6. Matriz de coeficiente E0.....	31
Figura 7. Matriz de masa M0.....	32
Figura 8. Matriz de coeficiente E2.....	32

**Lista de Apéndices**

Apéndice 1. Código solución matrices de coeficientes. .... 38

## Resumen

**Título:** Caracterización de las matrices de coeficientes para una sección transversal regular isotrópica implementando el método de elementos finitos de frontera escalada en ambiente Matlab.

**Autores:** Jefersón David Manrique Tarazona, Camilo Andrés Osorio Pinzón

**Palabras Clave:** Método de elementos finitos de limite escalado (SBFEM), Matriz de rigidez, Ondas guiadas.

### Descripción:

Los procesos de simulación permiten obtener soluciones aproximadas de diferentes fenómenos físicos, que enfrentan complejidad como: geometrías, condiciones de contorno o tipos de material. Implementando distintas técnicas a través de modelos matemáticos, se busca alcanzar un nivel de precisión cercano al comportamiento real que refleja el modelo físico en cuestión con el objetivo de mantener un costo computacional bajo. En este trabajo, se empleó el método de elementos finitos de limite escalado (SBFEM, por sus siglas en inglés), el cual se deriva del método de elementos finitos convencional (FEM), con el propósito de describir el procedimiento que se requiere para obtener las matrices de coeficientes con las que es posible generar la matriz de rigidez, la cual es utilizada en una gran variedad de aplicaciones mecánicas. Para empezar el desarrollo de nuevas técnicas, se creó un código que sirve como apoyo para estudiar las ondas guiadas, implementando este método generando las curvas de dispersión y discretizando la sección transversal de una guía de onda con un modelamiento tridimensional (3D) usando MATLAB como herramienta de programación.

---

\* Trabajo de Grado

\*\* Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería Mecánica.

Director: Octavio Andrés González Estrada PhD.

Codirector: Jabid Eduardo Quiroga Méndez PhD. Carlos Andrés Galán Pinilla MSc.

### Abstract

**Title:** Characterization of coefficient matrices for an isotropic regular transverse surface implementing the scaled boundary finite element method in the Matlab environment.

**Authors:** Jefersón David Manrique Tarazona, Camilo Andrés Osorio Pinzón

**Key Words:** Scaled Boundary Finite Element Method (SBFEM), Stiffness Matrix, Guided Waves.

#### Description:

The simulation processes allow obtaining approximate solutions for different physical phenomena that face complexities such as geometries, boundary conditions, or types of material. By implementing various techniques through mathematical models, the aim is to achieve a level of precision close to the real behavior reflected by the physical model, with the objective of maintaining low computational costs. In this work, the scaled boundary finite element method (SBFEM), derived from the conventional finite element method (FEM), was employed to describe the procedure required to obtain coefficient matrices that can be used to generate the stiffness matrix. This matrix is used in a wide range of mechanical applications. To initiate the development of new techniques, a code was created to support the study of guided waves. This involved implementing this new method to generate dispersion curves and discretizing the cross-section of a waveguide with three-dimensional (3D) modeling using MATLAB as a programming tool.

---

\* Degree Work

\*\* Faculty of Physical Mechanical Engineering. School of Mechanical Engineering.

Director: Octavio Andrés González Estrada PhD.

Codirector: Jabid Eduardo Quiroga Méndez PhD. Carlos Andrés Galán Pinilla MSc.

## Introducción

El análisis de estructuras y sistemas físicos ha experimentado una evolución gracias al método de elementos finitos (FEM), en donde se puede abarcar más de este método en las siguientes referencias (Celigüeta, 2000; Markopoulos, 2013; Oñate, 2009; Vázquez & Lopez, 2001), siendo una herramienta fundamental en ingeniería y ciencias aplicadas.

Dentro del amplio espectro del método de elementos finitos (FEM), los métodos semi-analíticos (SAFE, por sus siglas en inglés semi-analytical finite element), donde se puede explorar a fondo este tema consultando la referencia (Marzani et al., 2008) proporcionada, y el método de elementos finitos de límite escalado (SBFEM), han surgido como una técnica especializada que aborda con precisión la respuesta estructural a cargas y condiciones límite variadas (Bartoli, 2007). El SBFEM se enfoca en la solución de problemas con discontinuidades, como el contacto, la fractura y el daño, siendo más preciso en este tipo de problemas. El SBFEM es un método común para modelar la propagación de las ondas guiadas (Bulling et al., 2017), además de ser una técnica ampliamente utilizada en la ingeniería y la ciencia para modelar y resolver problemas complejos debido a su enfoque en problemas de mecánica de sólidos deformables y en la resolución de problemas con discontinuidades, ya que es una herramienta flexible donde pueden describirse materiales o geometrías como barras, láminas o cualquier otra estructura. Este método involucra la discretización de elementos finitos de la sección transversal, lo cual es una ventaja para la discretización eficiente de grietas (Gravenkamp et al., 2012).

La técnica de elementos finitos facilita el cálculo de las propiedades de propagación de las ondas guiadas. De igual forma, el modelado de ondas guiadas mediante el método de elementos finitos permite la simulación de diferentes escenarios, lo que facilita la identificación de los puntos críticos de la sección a estudiar y la elaboración de planes de mantenimiento y reparación (Li et al.,

2005). El método de elementos finitos de límite escalado (SBFEM) divide una estructura en pequeñas regiones llamadas subregiones o subdominios, donde se realizan los cálculos numéricos. Cada elemento se modela como una parte pequeña de la estructura y se consideran sus propiedades físicas y geométricas. Estos elementos se unen entre sí a través de sus nodos, lo que permite resolver el comportamiento global de la estructura (Bulling et al., 2017). El modelado de ondas guiadas mediante el método de elementos finitos implica el análisis de las ecuaciones de propagación de la onda en los elementos de la estructura, estas ecuaciones dependen de las propiedades elásticas y características geométricas de los elementos, así como de las condiciones de contorno y de las fuentes de excitación (Song, 2018). En este contexto, la matriz de rigidez juega un papel fundamental en la obtención de las curvas de dispersión para el análisis y comprensión de estas ondas, proporcionando la base matemática esencial para abordar la dinámica de los medios estructurales. Para las guías de onda prismáticas, es suficiente con discretizar solo la sección transversal con elementos finitos estándar y asumir ondas armónicas en la dirección de propagación. Esta técnica se denominada hoy en día como método de elementos finitos de guía de ondas o método semi-analítico de elementos finitos (SAFE). En este enfoque normalmente se utilizan elementos lineales o cuadráticos tradicionales para la discretización (Gravenkamp et al., 2013).

La matriz de rigidez es una representación matricial de la relación entre las fuerzas y los desplazamientos en los elementos finitos. Esta se utiliza para calcular las deformaciones, tensiones y el comportamiento estructural del sólido, considerando las propiedades dinámicas y de vibración del sistema. La matriz de rigidez es fundamental para garantizar la estabilidad y convergencia de las soluciones en el análisis de los elementos finitos, lo que la hace esencial en el estudio de problemas de ingeniería. Esta es necesaria para el cálculo de las curvas de dispersión en un

problema de ondas guiadas, ya que proporciona la información necesaria para resolver sistemas de ecuaciones lineales y analizar el comportamiento dinámico y vibratorio del sistema.

El cálculo de las curvas de dispersión describe analíticamente las características de varios modos de onda que permitan su uso para aplicaciones en ensayos no destructivos. Las curvas de dispersión en la propagación de ondas son gráficas que representan la relación entre la frecuencia y la velocidad de fase de una onda propagándose en un medio o estructura (Díaz et al., 2022), obteniéndose mediante la resolución de las ecuaciones de propagación de ondas para un medio o estructura específicos. Esto implica considerar las propiedades físicas y geométricas del medio, así como las condiciones de contorno (Blanco Polaina & Galán Fernandez, 2013). Obteniendo información sobre la propagación de las ondas en un medio, estas curvas indican la frecuencia de las ondas en función de la longitud de onda y la velocidad de propagación, prediciendo su comportamiento a partir de estas mediciones (Gómez & Ricón, 2007).

El cálculo numérico desarrollado en este proyecto se hace con ayuda del programa comercial MATLAB, permitiendo modelar el sólido con geometría de sección transversal regular con aplicación industrial, para luego calcular la matriz de rigidez. MATLAB es una herramienta poderosa y versátil ampliamente utilizada en la ingeniería para resolver problemas relacionados con el método de elementos finitos de límite escalado. La aplicación de MATLAB en el método de SBFEM se debe a su capacidad para realizar cálculos numéricos avanzados y su amplia gama de funciones dedicadas a la resolución de problemas de ingeniería (Pañeda, 2016; Rahman & Valdman, 2013). Este software proporciona un entorno de programación fácil de usar que permite al usuario implementar y resolver ecuaciones diferenciales, condiciones de contorno, propiedades y materiales específicos del problema. Además, ofrece un conjunto completo de herramientas para la visualización de datos, lo que facilita la interpretación y el análisis de los resultados generados

por el método de elementos finitos. Esto incluye la capacidad de generar gráficos en 2D y 3D, trazar curvas de deformación y esfuerzo, y animar el comportamiento de una estructura bajo diferentes cargas y condiciones (Song & Wolf, 1997). El paquete Partial Differential Equation (PDE) Toolbox de MATLAB, proporciona una interfaz intuitiva y fácil de usar para definir la geometría, generar la malla, establecer las condiciones de contorno y resolver las diferencias parciales asociadas al problema de elementos finitos (MathWorks, 2023).

Este proyecto se centra en explorar y caracterizar las matrices de coeficientes mediante el método de elementos finitos de límite escalado (SBFEM), dado que podría simplificar las operaciones y reducir el costo computacional requerido. A través de la comprensión detallada de este enfoque, se busca avanzar en la capacidad de modelar y predecir el comportamiento estructural en una amplia gama de aplicaciones. Al lograr una caracterización precisa de las matrices de coeficientes, el propósito último de este proyecto es contribuir al avance y las mejoras continuas de las técnicas de modelización estructural, proporcionando una base sólida para su aplicación en el cálculo de las curvas de dispersión para ondas guiadas.

### **Planteamiento y justificación**

Los materiales pueden sufrir daños ocasionados por servicios que afectan su estructura interna. Estos daños pueden ser causados por deformación, torsión, corrosión, fatiga, impactos, entre otros, además de posibles discontinuidades presentadas en el proceso de fabricación de la pieza a manufacturar. Esta es la razón por la cual se recurre a la aplicación de ensayos no destructivos.

El ultrasonido es uno de los ensayos más desarrollados hasta ahora en el conjunto de los no destructivos. El más reciente tipo, denominado Ondas guiadas o ultrasonido de largo alcance (GWT o LRUT por sus siglas en inglés), es de gran interés debido a la posibilidad que brinda de inspeccionar grandes distancias de material cubriendo toda su sección transversal. Sin embargo, a pesar de los grandes avances que ha tenido este ensayo, en la actualidad, la técnica no logra cubrir las distancias esperadas y su uso es complejo; esto se debe al fenómeno dispersivo y multinodal presente en el ultrasonido. Este fenómeno dispersivo se da, en parte, porque el transductor que genera las ondas produce un paquete de ellas dentro de un rango de frecuencia y velocidad. Por tanto, conocer la dispersión de la onda es fundamental para entender su comportamiento, aspecto abordado mediante enfoques de simulación empleando elementos finitos.

El método de elementos finitos (FEM, por sus siglas en inglés) es utilizado en gran medida para resolver problemas de vibración y propagación de ondas. Puede ofrecer soluciones eficientes para algunas integrales o diferenciales complejas sin soluciones analíticas, proporcionando datos computacionales precisos. Además, existe la posibilidad de mejorar dichas soluciones refinando los elementos utilizados en el estudio (Rose, 2014).

El método de elementos finitos de limite escalado (SBFEM por sus siglas en inglés) surgió como una extensión del método de elementos finitos tradicional (FEM). Se hace en algunos casos por sus ventajas en cuanto a eficiencia computacional y manejo de geometrías irregulares. En el contexto de ondas guiadas, el SBFEM puede ser preferido para modelar estas estructuras, ya que puede ofrecer una descripción más precisa y eficiente de los fenómenos de ondas guiadas. Varias investigadores como (Gravenkamp, 2014; Song, 2018; Wolf & Song, s/f), han explorado la aplicación del SBFEM para analizar estructuras con ondas guiadas, destacando la eficiencia computacional y la precisión en la descripción de fenómenos específicos.

la matriz de rigidez es una parte fundamental del proceso de formulación y resolución del sistema de ecuaciones que modela el comportamiento elástico y las interacciones de ondas en la estructura de un sistema. Asimismo, es uno de los problemas que invierte mayor complejidad al usar la técnica del método de elementos finitos de limite escalado, debido a las integrales a resolver. Por lo tanto, es necesario recurrir a métodos de integración precisos, como la cuadratura de Gauss. El análisis matricial de estructuras y, en consecuencia, el método de elementos finitos tiene como punto de partida la relación entre fuerzas y desplazamientos nodales para cada elemento individual.

Con base en lo anterior y debido a la necesidad de este método (SBFEM) para proporcionar una solución, en el presente trabajo de grado se pretende obtener y describir de forma eficaz las matrices de coeficientes para secciones transversales regulares isotrópicas, seleccionando el tipo de elemento finito (lineales o cuadráticos) más adecuado con el fin de lograr el menor costo computacional posible.

## 1. Objetivos

### 1.1 Objetivo General

Determinar las matrices de coeficientes para una sección transversal regular isotrópica implementando el método de elementos finitos de frontera escalada en ambiente Matlab.

### 1.2 Objetivos Específicos

- Describir el proceso para obtener las matrices de coeficientes de una superficie isotrópica discretizada con elementos finitos bidimensionales, como herramienta requerida para generar las curvas de dispersión de ondas guiadas mediante procesos de simulación con elementos finitos.
- Implementar los códigos para generar las matrices de coeficientes de una superficie arbitraria usando Matlab, con elementos finitos bidimensionales y aplicando el método SBFEM.

## 2. Materiales y métodos

### 2.1 Formulación para una guía de ondas tridimensional mediante el método de elementos finitos de límite escalado (SBFEM).

Esta formulación semi-analítica busca obtener la ecuación de onda gobernante utilizando el principio de trabajo virtual, generando y propagando microdeformaciones en el material (Galán-Pinilla et al., 2023; Song & Wolf, 1997). Las ecuaciones gobernantes de la elastodinámica lineal tridimensional en el dominio de la frecuencia expresadas en amplitudes de desplazamiento, se formulan como

$$\mathbf{L}^T \boldsymbol{\sigma} + \omega^2 \rho \mathbf{u} = 0. \quad (1)$$

Con la densidad de la masa  $\rho$ . Las amplitudes de tensión  $\{\boldsymbol{\sigma}\}$  y deformación  $\{\boldsymbol{\epsilon}\}$  se derivan de la ley de Hooke como

$$\boldsymbol{\sigma} = [\sigma_x \ \sigma_y \ \sigma_z \ \tau_{yz} \ \tau_{xz} \ \tau_{xy}]^T = \mathbf{D} \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} = [\epsilon_x, \ \epsilon_y, \ \epsilon_z, \ \gamma_{yz}, \ \gamma_{zx}, \ \gamma_{xy}]^T = \mathbf{L} \mathbf{u} \quad (2)$$

Con el operador diferencial  $\mathbf{L}$  y el vector de desplazamientos  $\mathbf{u}$

$$\mathbf{L} = \begin{bmatrix} \partial x & 0 & 0 & \partial y & \partial z & 0 \\ 0 & \partial y & 0 & \partial x & 0 & \partial z \\ 0 & 0 & \partial z & 0 & \partial x & \partial y \end{bmatrix}^T, \quad \mathbf{u} = [u_x \ u_y \ u_z]^T \quad (3)$$

Donde  $\mathbf{D}$  es la matriz de elasticidad que contiene las propiedades del material como el módulo de elasticidad  $[\mathbf{E}]$  y la relación de Poisson  $[\mathbf{\nu}]$  para materiales isotrópicos

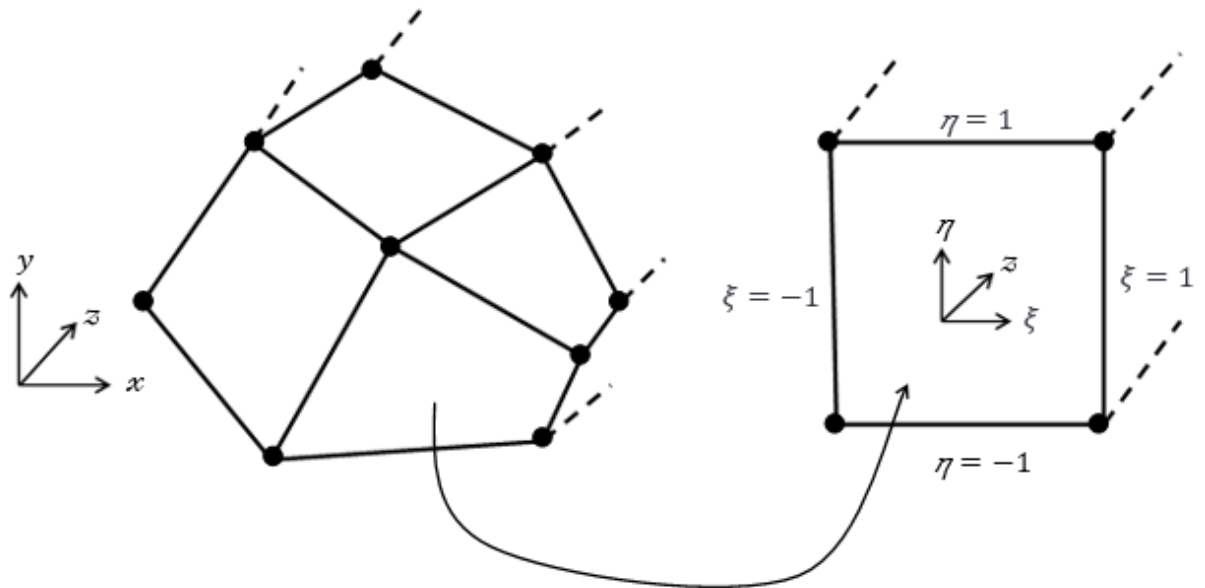
$$\mathbf{D} = \frac{E}{(1 + \nu)(1 - 2\nu)} \begin{bmatrix} 1 - \nu & 0 & 0 & 0 & 0 & 0 \\ & 1 - \nu & 0 & 0 & 0 & 0 \\ & & 1 - \nu & 0 & 0 & 0 \\ & & & \frac{1 - 2\nu}{2} & 0 & 0 \\ & & & & \frac{1 - 2\nu}{2} & 0 \\ & & & & & \frac{1 - 2\nu}{2} \end{bmatrix} \quad (4)$$

*Simétrico*

En SBFEM, para describir una guía de onda, se discretiza la sección transversal como se muestra en la figura 1, usando elementos isoparamétricos de superficie (Gravenkamp et al., 2015). En problemas 3D, se obtiene la geometría del elemento trasladando la malla 2D a lo largo de la dirección  $\{z\}$ , la cual se encuentra perpendicular a la superficie discretizada. Esto corresponde a tener el centro de escala del método de elementos finitos de límite escalado en el infinito (Man et al., 2012). De la misma forma, en el análisis de placas unidimensional (Gravenkamp, 2014), esto corresponde a tener la misma discretización y las mismas funciones de forma para cada valor de  $\{z\}$ .

**Figura 1.**

*Transformación de coordenadas tridimensionales SBFEM con el centro de escala en el infinito.*



La representación de la transformación de coordenadas globales  $(x,y)$  a coordenadas locales  $(\eta,\zeta)$  se muestra en la figura 1, similar al método de elementos finitos tradicional (FEM) cada elemento se representa interpolando sus coordenadas globales también conocidas como coordenadas nodales  $\{x\}$  y  $\{y\}$ , usando las funciones de forma  $\mathbf{N}(\eta,\zeta) = [N_1 N_2 \dots N_n]$  formuladas en las coordenadas locales (Lutz, 2007).

$$X(\eta,\xi) = \mathbf{N}(\eta,\zeta) \{x\}, \tag{5}$$

$$Y(\eta,\xi) = \mathbf{N}(\eta,\zeta) \{y\}, \tag{6}$$

La sección transversal de la guía de onda tridimensional es discretizada con elementos bidimensionales definidos en un sistema de coordenadas local  $(\eta, \xi)$ , la transformación de coordenadas que se muestra en la figura 1, se desarrolla de forma análoga al método de elementos finitos tradicional como

$$\begin{Bmatrix} \partial_\eta \\ \partial_\zeta \end{Bmatrix} = \mathbf{J} \begin{Bmatrix} \partial_x \\ \partial_y \end{Bmatrix} \quad (7)$$

Con la matriz jacobiana

$$\mathbf{J} = \begin{bmatrix} x_{,\eta} & y_{,\eta} \\ x_{,\zeta} & y_{,\zeta} \end{bmatrix} \quad (8)$$

Invirtiendo la ecuación (7) tenemos

$$\begin{Bmatrix} \partial_x \\ \partial_y \end{Bmatrix} = \frac{1}{|\mathbf{J}|} \begin{bmatrix} y_{,\zeta} & -y_{,\eta} \\ -x_{,\zeta} & x_{,\eta} \end{bmatrix} \begin{Bmatrix} \partial_\eta \\ \partial_\zeta \end{Bmatrix} \quad (9)$$

Donde  $|\mathbf{J}|$  es el valor absoluto del determinante de la matriz jacobiana. La formulación del operador diferencial  $\mathbf{L}$  en términos de las coordenadas de límite  $(\eta, \zeta)$  y la coordenada axial  $\{z\}$  dentro de la relación deformación-desplazamiento como

$$\boldsymbol{\varepsilon} = \mathbf{b}_1 \mathbf{u}_{,z} + \frac{1}{|\mathbf{J}|} (y_{,\zeta} \mathbf{b}_3 - x_{,\zeta} \mathbf{b}_2) \mathbf{u}_{,\eta} + \frac{1}{|\mathbf{J}|} (-y_{,\eta} \mathbf{b}_3 + x_{,\eta} \mathbf{b}_2) \mathbf{u}_{,\zeta} \quad (10)$$

Con

$$\mathbf{b}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T, \mathbf{b}_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \mathbf{b}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T \quad (11)$$

## 2.2. Aplicación

### 2.2.1. Discretización usando elementos isoparamétricos.

Los desplazamientos en la frontera son interpolados usando funciones de forma bidimensionales

$$\mathbf{u} = \begin{Bmatrix} u_x \\ u_y \\ u_z \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \dots & N_n & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \dots & 0 & N_n & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \dots & 0 & 0 & N_n \end{bmatrix} \mathbf{u}_n(Z) = \mathbf{N}(\eta, \zeta) \mathbf{u}_n(Z) \quad (12)$$

La interpolación de los desplazamientos en la frontera es similar a la interpolación bidimensional FEM, con la diferencia de que los desplazamientos ahora son función de la dirección  $\{Z\}$ . La discretización de la relación deformación-desplazamiento (Ecuación (13)) y la relación esfuerzo-desplazamiento (Ecuación (14)), se presentan de manera similar al caso de estructuras de placas (Galán-Pinilla et al., 2023; Gravenkamp et al., 2014).

$$\boldsymbol{\varepsilon}(z, \eta, \zeta) = \mathbf{B}_1 \mathbf{u}_{n,z}(z) + \mathbf{B}_2 \mathbf{u}_n(z). \quad (13)$$

$$\boldsymbol{\sigma}(z, \eta, \zeta) = \mathbf{D} \left( \mathbf{B}_1 \mathbf{u}_{n,z}(z) + \mathbf{B}_2 \mathbf{u}_n(z) \right). \quad (14)$$

Con

$$\mathbf{B}_1 = \mathbf{b}_1 \mathbf{N}, \quad \mathbf{B}_2 = \frac{1}{|J|} (y_{,\zeta} \mathbf{b}_3 - x_{,\zeta} \mathbf{b}_2) N_{,\eta} + \frac{1}{|J|} (-y_{,\eta} \mathbf{b}_3 + x_{,\eta} \mathbf{b}_2) N_{,\zeta} \quad (15)$$

Las funciones de forma presentes en la ecuación (12) pertenecientes a la matriz  $\mathbf{N}$  de  $3 \times 3n$ , están expresadas en términos de las coordenadas locales  $(\eta, \zeta)$ . Debido a la sencillez que presentan al momento de mallar, la discretización de la sección transversal de la guía de onda se realiza mediante elementos triangulares (lineales y cuadráticos) (Zienkiewicz et al., 2005) que permiten aproximar de forma sencilla diferentes contornos a diferencia de los elementos cuadriláteros.

### 2.3. Ecuación de movimiento

La ecuación semianalítica de elementos finitos de límite escalado se deriva del principio de trabajos virtuales (Deeks & Wolf, 2002), utilizando la formulación para un elemento de la discretización donde se formula la dinámica de la onda guiada energéticamente utilizando la ecuación del principio de trabajo virtual, considerando la ausencia de cargas en el cuerpo (Galán-Pinilla et al., 2023; Gravenkamp et al., 2014).

$$\delta U + \delta K = 0 \quad (16)$$

Donde  $U$  es la energía de deformación interna y  $K$  la energía cinética de la estructura. Al aplicar un campo de desplazamientos virtual  $\delta u$  a la estructura, la ley de Hooke presente en la ecuación (3) se escribe como

$$\delta \boldsymbol{\varepsilon} = \mathbf{L} \delta u \quad (17)$$

Aplicando la discretización de los desplazamientos tenemos

$$\delta \mathbf{u}(z, \eta, \zeta) = \delta u_n(Z) \quad (18)$$

Introduciendo la matriz de masa que utiliza la densidad del material  $\rho$  como

$$\mathbf{M}_0 = \int_{-1}^1 \int_{-1}^1 N^T \rho N |J| d\eta d\zeta \quad (19)$$

La energía de deformación y la energía cinética virtual se representa a continuación usando la matriz de masa como

$$\delta \mathbf{K} = \int_0^\infty \delta u_\eta^T(Z) \mathbf{M}_0 \ddot{u}_\eta(Z) dz \quad (20)$$

$$\delta \mathbf{U} = \int_v \delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dv \quad (21)$$

Usando la ecuación (13), la energía de deformación se reescribe como

$$\delta \mathbf{U} = \int_v [\mathbf{B}_1 \delta U_{n,z}(Z)]^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dv + \int_v [\mathbf{B}_2 \delta U_n(Z)]^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dv \quad (22)$$

Al aplicar la integración por partes al primer término de la ecuación anterior se tiene

$$\begin{aligned}
 \delta \mathbf{U} = & \int_A \delta u_n^T(\mathbf{Z}) \mathbf{B}_1^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dA - \int_v \delta u_n^T(\mathbf{Z}) \mathbf{B}_1^T \boldsymbol{\sigma}_{,z}(z, \eta, \zeta) |J| dv \\
 & + \int_v \delta u_n^T(\mathbf{Z}) \mathbf{B}_2^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dv
 \end{aligned} \tag{23}$$

Cuando la guía de onda se encuentra en el vacío, las tensiones superficiales no están presentes, lo cual conlleva a reducir el primer término a una integración en la interfaz de la guía de onda en ( $Z=0$ ).

$$\begin{aligned}
 \delta \mathbf{U} = & \int_A \delta u_n^T(\mathbf{0}) \mathbf{B}_1^T \boldsymbol{\sigma}(0, \eta, \zeta) |J| dA - \int_v \delta u_n^T(\mathbf{Z}) \mathbf{B}_1^T \boldsymbol{\sigma}_{,z}(z, \eta, \zeta) |J| dv \\
 & + \int_v \delta u_n^T(\mathbf{Z}) \mathbf{B}_2^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dv
 \end{aligned} \tag{24}$$

El primer término incluye las fuerzas nodales en la interfaz

$$\mathbf{q}_n(\mathbf{0}) = \int_A \mathbf{B}_1^T \boldsymbol{\sigma}(z, \eta, \zeta) |J| dA \tag{25}$$

Usando la expresión anterior y la ecuación (14), la energía de deformación se reescribe como

$$\begin{aligned}
 \delta \mathbf{U} = & - \int_v \delta u_n^T(z) (\mathbf{B}_1^T \mathbf{D} \mathbf{B}_1 U_{,zz}(z) + \mathbf{B}_1^T \mathbf{D} \mathbf{B}_2 U_{,z}(z) - \mathbf{B}_2^T \mathbf{D} \mathbf{B}_1 U_{,z}(z) \\
 & - \mathbf{B}_2^T \mathbf{D} \mathbf{B}_2 U(z)) |J| dz d\eta d\zeta + \delta u_n^T(\mathbf{0}) \mathbf{q}_n(0)
 \end{aligned} \tag{26}$$

Utilizando las matrices de coeficientes (Deeks & Wolf, 2002) del SBFEM

$$\mathbf{E}_0 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_1^T \mathbf{D} \mathbf{B}_1 |J| \, d\eta d\zeta \quad (27)$$

$$\mathbf{E}_1 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_2^T \mathbf{D} \mathbf{B}_1 |J| \, d\eta d\zeta \quad (28)$$

$$\mathbf{E}_2 = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_2^T \mathbf{D} \mathbf{B}_2 |J| \, d\eta d\zeta \quad (29)$$

$$\mathbf{M}_0 = \int_{-1}^1 \int_{-1}^1 \mathbf{N}^T \mathbf{D} \mathbf{N} |J| \, d\eta d\zeta \quad (30)$$

Las matrices de coeficientes están definidas mediante integrales, que pueden llegar a ser tediosas, por lo que hay que recurrir a métodos de integración numérica. Debido a la precisión y con un mínimo coste computacional, como método de integración numérica se hace uso de la cuadratura de Gauss (Oñate, 2009; Zienkiewicz et al., 2005).

Este método se basa en la construcción de una cuadratura que proporciona el resultado exacto para polinomios de grado  $2n-1$  o menor, donde  $n$  es el número de puntos de la cuadratura. El método de cuadratura de Gauss establece que cualquier función integrable puede ser aproximada con precisión por un polinomio de grado  $n$ . Este polinomio se ajusta a la función que se desea integrar, y se utilizan nodos y pesos específicos que se encuentran en libros como (Oñate, 2009; Zienkiewicz et al., 2005). Estos nodos y pesos se derivan de los polinomios de ortogonalidad

asociados al polinomio de grado  $\mathbf{n}$  y se utilizan como un conjunto de puntos de integración para aproximar la integral con precisión.

Reemplazando las matrices de coeficientes en la ecuación 26, la energía de deformación se reescribe como

$$\begin{aligned} \delta \mathbf{U} = & - \int_v \delta u_n^T(z) [E_0 U_{,zz}(z) + (\mathbf{E}_1^T - \mathbf{E}_1) U_{,z}(z) - \mathbf{E}_2 U(z) \\ & - \mathbf{M}_0 \ddot{u}(z)] |J| dz d\eta d\zeta + \delta u_n^T(\mathbf{0}) q_n(0) \end{aligned} \quad (31)$$

Para satisfacer la ecuación (16) en todos los desplazamientos  $\delta u_n(Z)$  lo que implica que el equilibrio se da en la dirección radial  $\{\xi\}$  siendo en este caso la misma dirección  $\{z\}$  y en el sentido de los elementos finitos en la dirección circunferencial  $(\eta, \zeta)$  es necesario que se cumplan las siguientes condiciones

$$\delta u_n^T(\mathbf{0}) q_n(0) = 0 \quad (32)$$

$$E_0 U_{,zz}(z) + (\mathbf{E}_1^T - \mathbf{E}_1) U_{,z}(z) - \mathbf{E}_2 U(z) - \mathbf{M}_0 \ddot{u}(z) = 0 \quad (33)$$

Con

$$\mathbf{E}_1^T = \int_{-1}^1 \int_{-1}^1 \mathbf{B}_1^T \mathbf{D} \mathbf{B}_2 |J| d\eta d\zeta \quad (34)$$

Teniendo en cuenta que las ecuaciones (32) y (33) son semianalíticas para los desplazamientos, mientras que la sección transversal se discretiza con elementos finitos en dos dimensiones, los desplazamientos  $u_n(z)$  pueden ser una función analítica arbitraria en función de

$\{z\}$ , dependiendo de la aplicación que se este trabajando diferentes procedimientos de solución han sido propuestos, con el objetivo de describir las ondas a lo largo de la estructura (guía de onda), la solución de los desplazamientos está dada por la expresión

$$\mathbf{u}_n(\mathbf{z}) = e^{\lambda z - i\omega t} \hat{\mathbf{u}}_n \quad (35)$$

Con  $\omega$  como la frecuencia angular,  $t$  es el tiempo,  $i$  la unidad imaginaria,  $\hat{\mathbf{u}}_n$  es el vector de las amplitudes de desplazamiento complejas y  $\lambda$  se define como

$$\lambda = iK \quad (36)$$

Donde  $K$  es el numero de onda en la dirección  $\{z\}$  de un modo particular de propagación, tenga en cuenta que la ecuación (35) es válida para estructuras que son homogéneas en la dirección de propagación  $\{z\}$  cuando se asume un comportamiento lineal del material (Gravenkamp, 2014). Después de sustituir la ecuación (35) en la (33), tenemos la ecuación del SBFEM generalizada para una sección transversal arbitraria.

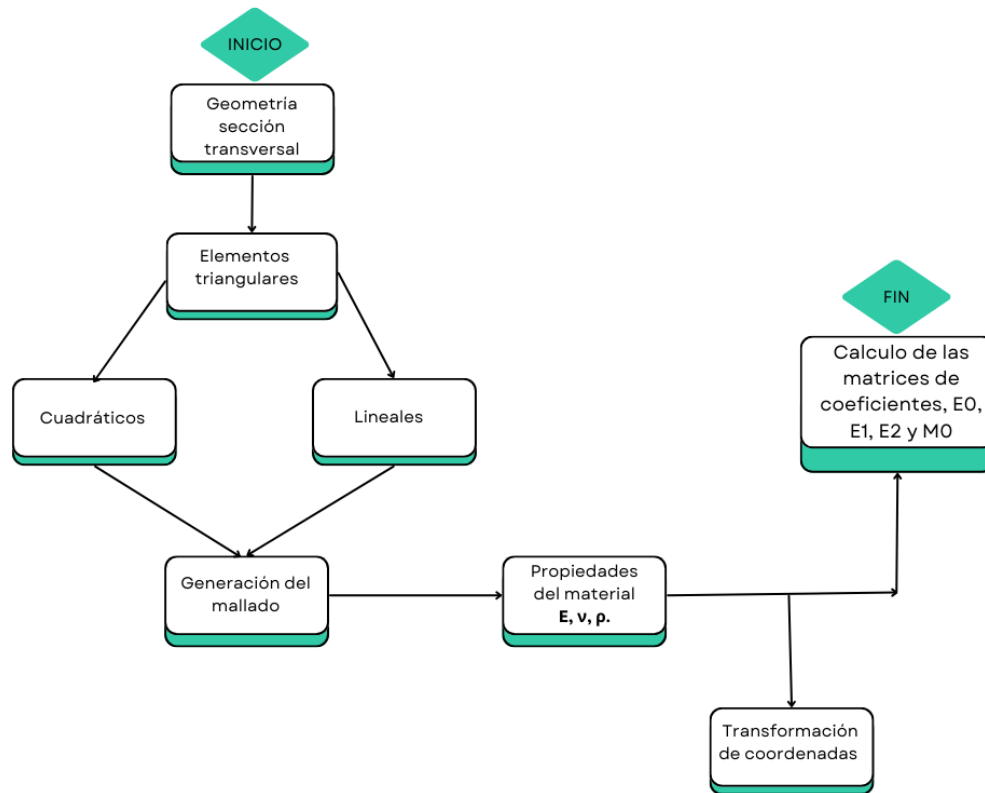
$$(\lambda^2 \mathbf{E}_0 + \lambda(\mathbf{E}_1^T - \mathbf{E}_1) - \mathbf{E}_2 u_n(z) - \omega^2 \mathbf{M}_0) \hat{\mathbf{u}}_n = 0 \quad (37)$$

#### 2.4. Algoritmo para calcular la matriz de rigidez por SBFEM

La figura 2, muestra el proceso desarrollado en la aplicación MATLAB para obtener las matrices de coeficientes por el método SBFEM.

**Figura 2.**

*Diagrama esquemático del algoritmo propuesto para el cálculo de las matrices de coeficientes.*



El primer paso es definir la geometría de la sección transversal de la estructura o componente que se desea analizar utilizando el PDE Toolbox de MATLAB.

El segundo paso implica la discretización de la sección transversal, generando el mallado para elementos triangulares (lineales o cuadráticos) de segundo orden. Esto se logra obteniendo las coordenadas globales de cada uno de los nodos que conforman el elemento.

El tercer paso es la definición de las propiedades del material, que incluyen la densidad ( $\rho$ ), el coeficiente de Poisson ( $\nu$ ) y el módulo de Young ( $E$ ) para una superficie isotrópica uniforme.

En cuarto paso, se realiza la transformación de coordenadas, pasar del sistema de coordenadas cartesianas  $x, y, z$  al sistema de coordenadas locales  $\eta$  y  $\zeta$ . Esto se logra utilizando las funciones de forma para el tipo de elemento seleccionado, las cuales representan las deformaciones y las condiciones de contorno en cada elemento.

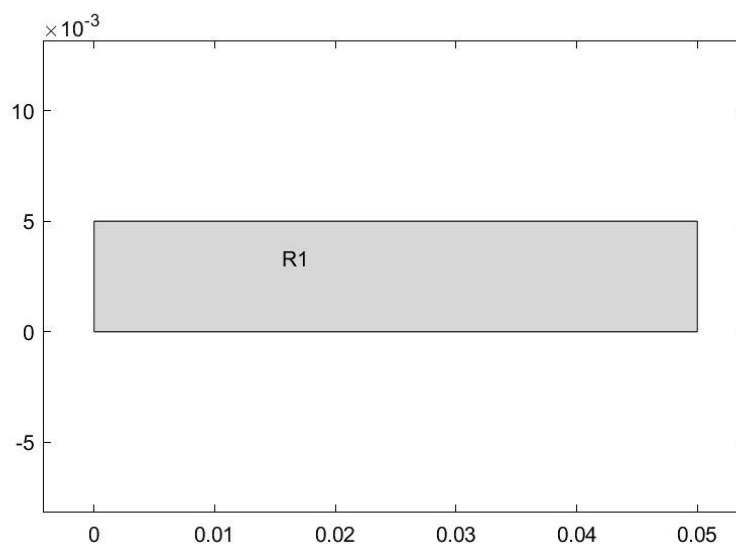
El quinto paso implica calcular las matrices de coeficientes  $E_0, E_1, E_2$ , y  $M_0$ , utilizando la integración a través del método de cuadratura de Gauss.

### 3. Resultados

En este capítulo, las matrices de coeficientes se determinan para una sección transversal rectangular de 5mm de espesor por 50mm de ancho, con propiedades de material para el aluminio:  $E = 70$  GPa, Poisson's 0.33 y densidad ( $\rho$ ) =  $2700 \text{ kg/m}^3$ , la geometría se puede visualizar en la figura 3.

#### Figura 3.

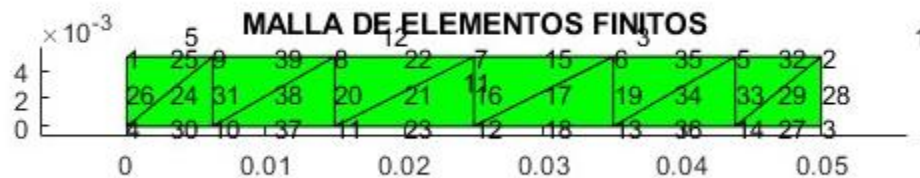
*Geometría sección transversal.*



En la figura 4. Se puede observar el mallado para la geometría de la sección transversal con elementos triangulares cuadraticos.

**Figura 4.**

*Mallado de la geometría.*



**3.1. Resultados numéricos**

En este capítulo, se optó por generar las gráficas de cada una de las matrices de coeficientes E0, E1, E2 y M0. Este enfoque se ha adoptado con el propósito de proporcionar una representación visual de la estructura de estas matrices y evaluar si cumplen con las propiedades delineadas por los autores Chongmin Song, Hauke Gravenkamp, en sus respectivos trabajos (Gravenkamp, 2014; Song, 2018). La elección de esta metodología se fundamenta en la complejidad numérica de las matrices, cuyo tamaño considerable dificulta su análisis directo.

La visualización gráfica se presenta como una alternativa eficaz para abordar esta complejidad y facilitar la identificación de patrones o características significativas que respalden las expectativas teóricas establecidas por los autores mencionados.

**Tabla 1.**

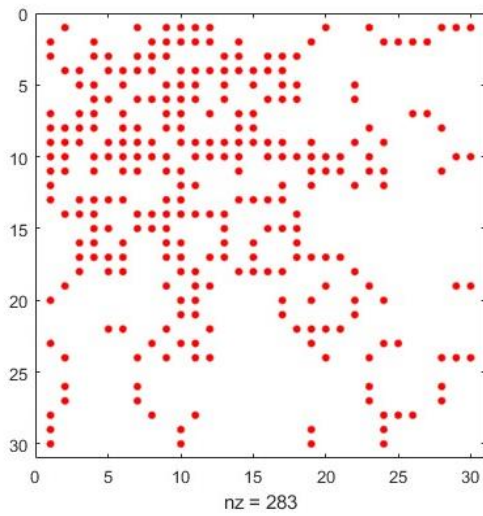
*Propiedades de las matrices de coeficientes.*

	<b>Simetria</b>	<b>Definición</b>
M0	Simétrico	Definida positiva
E0	Simétrico	Definida positiva

E1	No simétrico	Indefinida
E2	Simétrico	Semidefinida positiva

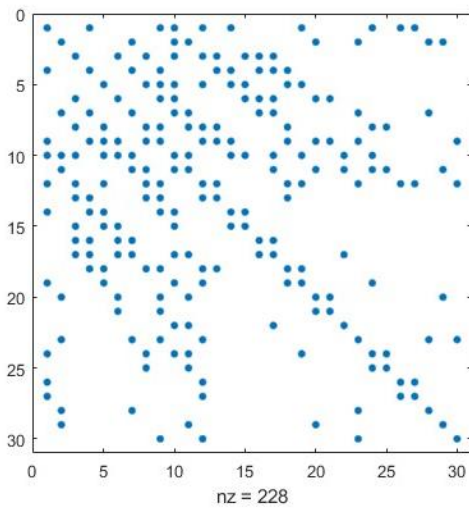
**Figura 5.**

*Matriz de coeficiente E1.*



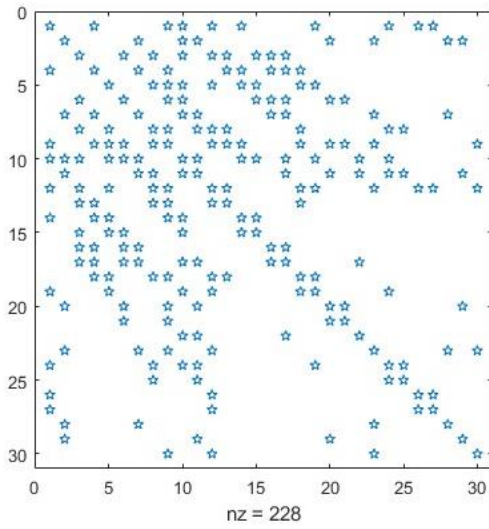
**Figura 6.**

*Matriz de coeficiente E0*



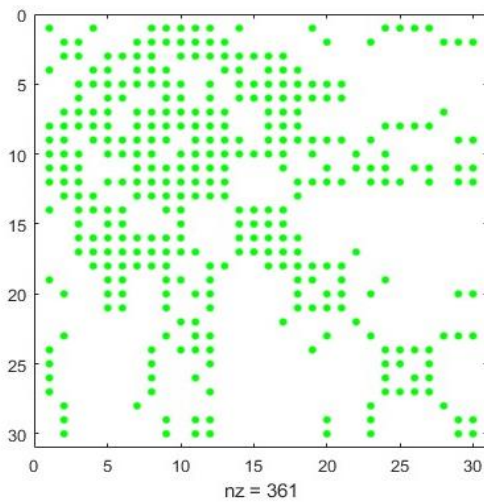
**Figura 7.**

*Matriz de masa  $M_0$ .*



**Figura 8.**

*Matriz de coeficiente  $E_2$ .*



Según (Gravenkamp, 2014; Song, 2018) las matrices de coeficientes para el SBFEM deben cumplir con los siguientes requisitos:

- 1. Las matrices de coeficientes  $E_0$ ,  $E_1$  y  $E_2$  del elemento finito, se obtienen ensamblando las matrices de coeficientes de cada uno de los elementos.**
- 2. La matriz  $M_0$  siempre es diagonal.**
- 3.  $M_0$  y  $E_0$  son simétricas y están definidas positivas, es decir, todos los eigenvalores son positivos.**
- 4.  $E_2$  es simétrica y semidefinida positiva ya que algunos de los eigenvalores de  $E_2$  son siempre cero. En consecuencia  $E_2$  siempre es singular. Físicamente, estas singularidades representan movimientos rígidos del cuerpo, correspondientes a traslaciones de la estructura completa a lo largo de uno de los ejes de coordenadas sin deformación.**
- 5. La matriz  $E_1$  es indefinida y no es simétrica. Esto sugiere que la matriz  $E_1$  puede producir productos internos positivos y negativos con ciertos vectores, lo que indica la presencia de autovalores positivos y negativos.**
- 6. La integración se puede realizar numéricamente de la misma manera que el cálculo de la matriz de rigidez de elementos finitos unidimensionales. Las reglas para la elección del orden de cuadratura de integración en el método convencional de elementos finitos son igualmente aplicables.**

Las matrices de coeficientes suelen ser simétricas en el SBFEM, ya que representan interacciones entre puntos simétricos a lo largo de la estructura. La simetría contribuye a la eficiencia computacional y a la estabilidad numérica.

### 3. Conclusiones

Este proyecto presentó de manera detallada el procedimiento para el cálculo de las matrices de coeficientes requeridas para implementar el método de elementos finitos de límite escalado (SBFEM) en problemas dinámicos tridimensionales.

Se implementó un código en Matlab que permite calcular las matrices de coeficientes por el método SBFEM, aplicando elementos triangulares y funciones de forma lineales o cuadráticas. La modularidad del código facilita su adaptación a problemas específicos con diferentes configuraciones geométricas y propiedades del material.

Se verificó la consistencia y validez de las matrices resultantes, confrontándola con la teoría propuesta por destacados autores en el campo, como Chongmin Song y Hauke Gravenkamp, demostrando su capacidad para generar las matrices de coeficientes necesarias para obtener la respuesta de la estructura ante ondas guiadas.

Este proyecto sirve como punto de partida para explorar nuevas fronteras en la modelización de ondas guiadas y sus diversas aplicaciones.

### Referencias Bibliográficas

Bartoli, I. (2007). *Structural health monitoring by ultrasonic guided waves*.

Blanco Polaina, M., & Galán Fernández, J. M. (2013). *Propagación de ondas mecánicas en barras, uniones y defectos*.

Bulling, J., Prager, J., & Korme, F. (2017). *Application of the Scaled Boundary Finite Element Method (SBFEM) for a numerical simulation of ultrasonic guided waves*.  
<https://doi.org/10.5162/sensor2017/c5.4>

- Celigüeta, J. T. (2000). *Método de los elementos finitos para análisis estructural*.
- Deeks, A. J., & Wolf, J. P. (2002). A virtual work derivation of the scaled boundary finite-element method for elastostatics. *Computational Mechanics*, 28(6), 489–504. <https://doi.org/10.1007/s00466-002-0314-2>
- Díaz, G., Castello, M. E., Amado, M., Ildz, J., Gutierrez, F., & B. Groth, E. (2022). *Curvas de dispersión aplicadas a la propagación de ondas guiadas en estructuras de sección circular hueca*. <https://revistas.uncu.edu.ar/ojs3/index.php/revicap>
- Galán-Pinilla, C. A., E-Quiroga, J., Peña-Ballesteros, D. Y., Acosta-Minoli, C. A., & González-Estrada, O. A. (2023). Comparative Study of Dispersion Curves for LAMB Waves Using Analytical Solutions and Semi-Analytical Methods. *Applied Sciences (Switzerland)*, 13(3). <https://doi.org/10.3390/app13031706>
- Gómez, Y., & Ricón, U. (2007). *Estudio de sistemas ultrasónicos basados en multitransductores para la detección de defectos en estructuras tipo placa*.
- Gravenkamp, H. (2014). *Numerical methods for the simulation of ultrasonic guided waves*.
- Gravenkamp, H., Birk, C., & Song, C. (2014). Computation of dispersion curves for embedded waveguides using a dashpot boundary condition. *The Journal of the Acoustical Society of America*, 135(3), 1127–1138. <https://doi.org/10.1121/1.4864303>
- Gravenkamp, H., Birk, C., & Van, J. (2015). Modeling ultrasonic waves in elastic waveguides of arbitrary cross-section embedded in infinite solid medium. *Computers and Structures*, 149, 61–71. <https://doi.org/10.1016/j.compstruc.2014.11.007>
- Gravenkamp, H., Man, H., Song, C., & Prager, J. (2013). The computation of dispersion relations for three-dimensional elastic waveguides using the Scaled Boundary Finite Element Method.

*Journal of Sound and Vibration*, 332(15), 3756–3771.  
<https://doi.org/10.1016/j.jsv.2013.02.007>

Gravenkamp, H., Song, C., & Prager, J. (2012). A numerical approach for the computation of dispersion relations for plate structures using the Scaled Boundary Finite Element Method. *Journal of Sound and Vibration*, 331(11), 2543–2557.  
<https://doi.org/10.1016/j.jsv.2012.01.029>

Li, B., Cheng, L., Deeks, A. J., & Teng, B. (2005). A modified scaled boundary finite-element method for problems with parallel side-faces. Part II. Application and evaluation. *Applied Ocean Research*, 27(4–5), 224–234. <https://doi.org/10.1016/j.apor.2005.11.007>

Lutz, L. (2007). *Wave Propagation in Infinite Domains. With Applications to Structure Interaction* (Vol. 31). Springer.

Man, H., Song, C., Gao, W., & Tin-Loi, F. (2012). A unified 3D-based technique for plate bending analysis using scaled boundary finite element method. *International Journal for Numerical Methods in Engineering*, 91(5), 491–515. <https://doi.org/10.1002/nme.4280>

Markopoulos, A. P. (2013). *Finite Element Method in Machining Processes*. Springer London.  
<https://doi.org/10.1007/978-1-4471-4330-7>

Marzani, A., Viola, E., Bartoli, I., Lanza di Scalea, F., & Rizzo, P. (2008). A semi-analytical finite element formulation for modeling stress wave propagation in axisymmetric damped waveguides. *Journal of Sound and Vibration*, 318(3), 488–505.  
<https://doi.org/10.1016/j.jsv.2008.04.028>

MathWorks. (2023). *Partial Differential Equation Toolbox. User's Guide*. [www.mathworks.com](http://www.mathworks.com)

Oñate, E. (2009). *Structural Analysis with the Finite Element Method. Linear Statics*.

- Pañeda, E. M. (2016). *MATLAB: Una herramienta para la didáctica del Método de los Elementos Finitos*. [www.fisem.org/web/unionhttp://asenmacformacion.com/ojs/index.php/union/index](http://www.fisem.org/web/unionhttp://asenmacformacion.com/ojs/index.php/union/index)
- Rahman, T., & Valdman, J. (2013). Fast MATLAB assembly of FEM matrices in 2D and 3D: Nodal elements. *Applied Mathematics and Computation*, 219(13), 7151–7158. <https://doi.org/10.1016/j.amc.2011.08.043>
- Rose, J. L. (2014). *Ultrasonic Guided Waves in Solid Media*.
- Song, C. (2018). *The scaled boundary finite element method: introduction to theory and implementation*.
- Song, C., & Wolf, J. P. (1997). The scaled boundary finite-element method-alias consistent infinitesimal finite-element cell method-for elastodynamics. En *Comput. Methods Appl. Mech. Engrg* (Vol. 147).
- Vázquez, M., & Lopez, E. (2001). *El método de los elementos finitos aplicado al análisis estructural*.
- Wolf, J. P., & Song, C. (s/f). *The scaled boundary finite-element method - a primer: derivations*. [www.elsevier.com/locate/compstruc](http://www.elsevier.com/locate/compstruc)
- Zienkiewicz, O. C., Taylor, R. L., & Zhu, J. Z. (2005). *The Finite Element Method: Its Basis and Fundamentals*. [www.cimne.upc.es](http://www.cimne.upc.es)

## Apéndices

### Apéndice A.

*Código solución matrices de coeficientes.*

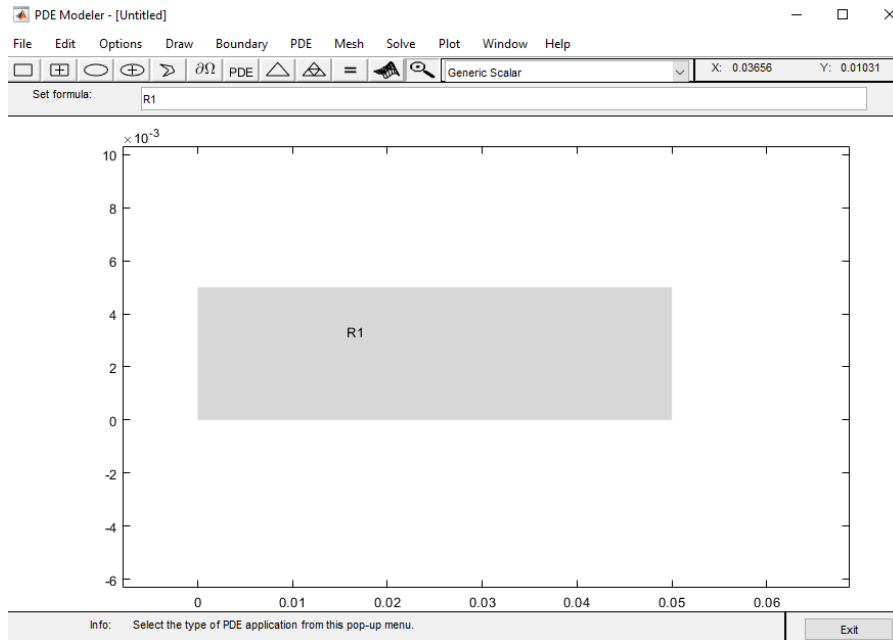
Luego, se realiza un recorrido a por cada códigos utilizado en este trabajo de investigación, donde se presentan 5 guiones programados para determinar las matrices de coeficientes por el SBFEM en la herramienta Matlab. El script ejecutable se define como '**stiffness\_Mat.m**'.

El procedimiento esta establecido para trabajar con el plano **X, Y** para discretizar la sección transversal y **Z** como el eje de propagación.

### Definición del modelo

```
model = createpde; %Crea un modelo para una PDE única (escalar) lineal o no lineal general.  
pdetool; %Inica PDE modeler para crear la geometria
```

Al ejecutar el código se abre una ventana que lo dirige hacia la PDE Tool de Matlab para generar la geometría de la sección transversal con ayuda de esta herramienta.



Para definir la geometría, se hace doble click en la misma y se introducen las dimensiones necesarias (teniendo en cuenta que estas dimensiones deben expresarse en metros). Posteriormente, se selecciona la opción **‘Draw’** y luego **‘Export Geometry Description’**.

Para generar el mallado se corre la siguiente sección del código

```
g = decsg(gd,sf,ns); %Descompone la matriz de descripcion geometrica gd en la matriz de geometria g
geometryFromEdges(model,g); %agrega la geometría 2-D descrita en g al contenedor del modelo.

mesh = generateMesh(model,'GeometricOrder','quadratic','Hmax',0.005); %Genere una malla para la geometria
% Hmax es un límite superior apropiado
```

En la sección del código **‘mesh’** se puede cambiar el tamaño y tipo de elemento ya sea lineal o cuadrático.

Enseguida se van a guardar una serie de ficheros y parámetros del problema.

```

% %Ficheros del mallado

XY = model.Mesh.Nodes';
Top = model.Mesh.Elements';

Top = Top(:, [1 4 2 5 3 6]);

% Parametros del problema

N    = size(XY,1);
Nel  = size(Top,1);
NNpe = size(Top,2);

```

Luego se pueden editar las propiedades de los materiales

```

% Propiedades del material

v=0.33;      %Poisson
E=70;       %[Gpa]
dens = 2800; %[kg/cm^3]

D = E/((1+v)*(1-2*v))*[1-v v v 0 0 0;v 1-v v 0 0 0;v v 1-v 0 0 0;0 0 0 (1-2*v)/2 0 0;0 0 0 0 (1-2*v)/2 0;0 0 0 0 0 (1-2*v)/2];
D = 1e9*(D);

```

Para generar la gráfica de la sección transversal con su respectivo mallado

```

% Dibujo de la malla
%=====

d_malla(N,Nel,NNpe,XY,Top);

```

Posteriormente se hace el cálculo de las matrices de coeficientes

```
% Calculo matriz de rigidez
global E0 E1 E2 M0;

E0(2*N,2*N)= 0;
E1(2*N,2*N)= 0;
E2(2*N,2*N)= 0;
M0(2*N,2*N)= 0;

DB_1(6,2*NNpe*Nel)= 0;
DB_2(6,2*NNpe*Nel)= 0;

matriz_E0(Nel,NNpe,D,XY,Top);
matriz_E1(Nel,NNpe,D,XY,Top);
matriz_E2(Nel,NNpe,D,XY,Top);
matriz_M0(Nel,NNpe,D,XY,Top, dens);
```

Con los scripts: **matriz\_E0.m**, **matriz\_E1.m**, **matriz\_E2.m**, **matriz\_M0.m** se hace la transformación de coordenadas para cada una de las matrices de coeficientes, utilizando las funciones de forma que son requeridas para cada tipo de elemento.

Script **matriz\_E0.m**

```

function matriz_E0(Nel,NNpe,D,XY,Top, E0, DB_1, DB_2)

% CALCULO DE MATRICES Y ENSAMBLADO

global E0 DB_1 DB_2;

%-----Inicio del calculo de la matriz de un solo elemento-----

for Ie= 1:Nel %inicia el ciclo en 1 hasta que llegue al número de elementos de la malla

    % --- coordenadas globales

    Xe = zeros(1,NNpe);
    Ye = zeros(1,NNpe);

    for j= 1:NNpe %inicia el ciclo en 1 hasta que llegue al número de nodos por elemento

        Xe(j)= XY(Top(Ie,j),1);
        Ye(j)= XY(Top(Ie,j),2);

    end

    %*****Cuadratico*****%
    CoordGL = [Xe' Ye'];
    E0_e = zeros(18);
    %-----Integracion númerica-----
    npg= 3; % número de puntos de gauss
    DB1=zeros(6,18);
    DB2=zeros(6,18);

    [CoordPG, H] = pgauss(npg);

    for i = 1:npg %inicia el ciclo en 1 hasta que llegue al número de puntos de gauss
        Xg = CoordPG(i,1); %reemplaza del valor de la coordenada  $\eta$ .
        Hi= H(i);

        for j = 1:npg
            Yg = CoordPG(j,2); %reemplaza del valor de la coordenada  $\zeta$ .

            %funciones de forma

            [Ne,dN, dNx, dNy] = fformaa_T6(Xg,Yg);

            Jac =[dN(2,2:3:18)*Xe' dN(2,2:3:18)*Ye'; dN(1,1:3:18)*Xe' dN(1,1:3:18)*Ye'];

            DetJac = det(Jac);
            InJac = 1/abs(DetJac);

```

```

%..... matriz DB1 y DB2.....

b3=[1 0 0 0 0 0; 0 0 0 1 0 0; 0 0 0 0 1 0]';
b2=[0 0 0 1 0 0; 0 1 0 0 0 0; 0 0 0 0 0 1]';
b1=[0 0 0 0 1 0; 0 0 0 0 0 1; 0 0 1 0 0 0]';

B1 = b1*Ne;
B2 = InJac*((Jac(2,2)*b3-Jac(2,1)*b2)*dNy+(-Jac(1,2)*b3+Jac(1,1)*b2)*dNx);

DB1 = D*B1 + DB1;
DB2 = D*B2 + DB2;

%.....calculo de la matriz E0.....

E0_e = E0_e + B1'*D*B1*DetJac*Hi;

end
end

if Ie == 1

    DB_1 = DB1;
    DB_2 = DB2;

else
    DB_1 = [DB_1, DB1];
    DB_2 = [DB_2, DB2];
end

%-----Fin del calculo de la matriz de un solo elemento-----

```

```
%..... Ensamblado de la Matriz Global con TRIANGULO CUADRATICO .....  
  
LM=[2*Top(Ie,1)-1;  
    2*Top(Ie,1);  
    2*Top(Ie,2)-1;  
    2*Top(Ie,2);  
    2*Top(Ie,3)-1;  
    2*Top(Ie,3);  
    2*Top(Ie,4)-1;  
    2*Top(Ie,4);  
    2*Top(Ie,5)-1;  
    2*Top(Ie,5);  
    2*Top(Ie,6)-1;  
    2*Top(Ie,6)];  
  
% Ensamble final  
  
for j= 1:2*NNpe  
    for k= 1:2*NNpe  
        Ien= LM(j);  
        Jen= LM(k);  
  
        E0(Ien,Jen)= E0(Ien,Jen) + E0_e(j,k);  
    end  
end  
---
```

Script **matriz\_E1.m**

```

function matriz_E1(Nel,NNpe,D,XY,Top, DB_1, DB_2)
% CALCULO DE MATRICES Y ENSAMBLADO

global E1 DB_1 DB_2;

%-----Inicio del calculo de la matriz de un solo elemento-----

for Ie= 1:Nel %inicia el ciclo en 1 hasta que llegue al número de elementos de la malla

    % --- coordenadas globales

    Xe = zeros(1,NNpe);
    Ye = zeros(1,NNpe);

    for j= 1:NNpe %inicia el ciclo en 1 hasta que llegue al número de nodos por elemento

        Xe(j)= XY(Top(Ie,j),1);
        Ye(j)= XY(Top(Ie,j),2);

    end

    %*****Cuadratico*****%
    CoordGL = [Xe' Ye'];
    E1_e = zeros(18);
    %-----Integracion numérica-----
    npg= 3; % número de puntos de gauss
    DB1=zeros(6,18);
    DB2=zeros(6,18);

    [CoordPG, H] = pgauss(npg);

    for i = 1:npg %inicia el ciclo en 1 hasta que llegue al número de puntos de gauss
        Xg = CoordPG(i,1); %reemplaza del valor de la coordenada  $\eta$ .
        Hi= H(i);

        for j = 1:npg
            Yg = CoordPG(j,2); %reemplaza del valor de la coordenada  $\zeta$ .

            %funciones de forma

            [Ne,dN, dNx, dNy] = fformaa_T6(Xg,Yg);

            Jac =[dN(2,2:3:18)*Xe' dN(2,2:3:18)*Ye'; dN(1,1:3:18)*Xe' dN(1,1:3:18)*Ye'];

            DetJac = det(Jac);
            InJac = 1/abs(DetJac);

```

```

%..... matriz DB1 y DB2.....

b3=[1 0 0 0 0 0; 0 0 0 1 0 0; 0 0 0 0 1 0]';
b2=[0 0 0 1 0 0; 0 1 0 0 0 0; 0 0 0 0 0 1]';
b1=[0 0 0 0 1 0; 0 0 0 0 0 1; 0 0 1 0 0 0]';

B1 = b1*Ne;
B2 = InJac*((Jac(2,2)*b3-Jac(2,1)*b2)*dNy+(-Jac(1,2)*b3+Jac(1,1)*b2)*dNx);

DB1 = D*B1 + DB1;
DB2 = D*B2 + DB2;

%.....calculo de la matriz de rigidez.....

E1_e = E1_e + B2'*D*B1*DetJac*Hi;

    end
end

if Ie == 1

    DB_1 = DB1;
    DB_2 = DB2;

else

    DB_1= [DB_1, DB1];
    DB_2= [DB_2, DB2];

end

%-----Fin del calculo de la matriz de un solo elemento-----

```

```
%..... Ensamblado de la Matriz Global con TRIANGULO CUADRATICO .....  
  
LM=[2*Top(Ie,1)-1;  
    2*Top(Ie,1);  
    2*Top(Ie,2)-1;  
    2*Top(Ie,2);  
    2*Top(Ie,3)-1;  
    2*Top(Ie,3);  
    2*Top(Ie,4)-1;  
    2*Top(Ie,4);  
    2*Top(Ie,5)-1;  
    2*Top(Ie,5);  
    2*Top(Ie,6)-1;  
    2*Top(Ie,6)];  
  
% Ensamble final  
  
for j= 1:2*NNpe  
    for k= 1:2*NNpe  
        Ien= LM(j);  
        Jen= LM(k);  
  
        E1(Ien,Jen)= E1(Ien,Jen) + E1_e(j,k);  
    end  
end  
end
```

Script **matriz\_E2.m**

```

function matriz_E2(Nel,NNpe,D,XY,Top,DB_1, DB_2)
% CALCULO DE MATRICES Y ENSAMBLADO

global E2 DB_1 DB_2;

%-----Inicio del calculo de la matriz de un solo elemento-----

for Ie= 1:Nel %inicia el ciclo en 1 hasta que llegue al número de elementos de la malla

    % --- coordenadas globales

    Xe = zeros(1,NNpe);
    Ye = zeros(1,NNpe);

    for j= 1:NNpe %inicia el ciclo en 1 hasta que llegue al número de nodos por elemento

        Xe(j)= XY(Top(Ie,j),1);
        Ye(j)= XY(Top(Ie,j),2);

    end

    %*****Cuadratico*****%
    CoordGL = [Xe' Ye'];
    E2_e = zeros(18);
    %-----Integracion númerica-----
    npg= 3; % número de puntos de gauss
    DB1=zeros(6,18);
    DB2=zeros(6,18);

    [CoordPG, H] = pgauss(npg);

    for i = 1:npg %inicia el ciclo en 1 hasta que llegue a npg
        Xg = CoordPG(i,1); %reemplaza del valor de la coordenada η.
        Hi= H(i);

        for j = 1:npg
            Yg = CoordPG(j,2);

            %funciones de forma

            [Ne,dN, dNx, dNy] = fformaa_T6(Xg,Yg);

            Jac =[dN(2,2:3:18)*Xe' dN(2,2:3:18)*Ye'; dN(1,1:3:18)*Xe' dN(1,1:3:18)*Ye'];

            DetJac = det(Jac);
            InJac = 1/abs(DetJac);

```

```

%..... Ensamblado de la Matriz Global con TRIANGULO CUADRATICO .....

LM=[2*Top(Ie,1)-1;
    2*Top(Ie,1);
    2*Top(Ie,2)-1;
    2*Top(Ie,2);
    2*Top(Ie,3)-1;
    2*Top(Ie,3);
    2*Top(Ie,4)-1;
    2*Top(Ie,4);
    2*Top(Ie,5)-1;
    2*Top(Ie,5);
    2*Top(Ie,6)-1;
    2*Top(Ie,6)];

% Ensamble final

for j= 1:2*NNpe
    for k= 1:2*NNpe
        Ien= LM(j);
        Jen= LM(k);

        E2(Ien,Jen)= E2(Ien,Jen) + E2_e(j,k);
    end
end
end

if Ie == 1

    DB_1 = DB1;
    DB_2 = DB2;

else

    DB_1= [DB_1, DB1];
    DB_2= [DB_2, DB2];

end

%-----Fin del calculo de la matriz de un solo elemento-----

```

Script **matriz\_M0.m**


---

```

function matriz_M0(Nel,NNpe,D,XY,Top, dens);

% CALCULO DE MATRICES Y ENSAMBLADO

global M0;

%-----Inicio del calculo de la matriz de un solo elemento-----

for Ie= 1:Nel %inicia el ciclo en 1 hasta que llegue al número de elementos de la malla

    % --- coordenadas globales|

    Xe = zeros(1,NNpe);
    Ye = zeros(1,NNpe);

    for j= 1:NNpe %inicia el ciclo en 1 hasta que llegue al número de nodos por elemento

        Xe(j)= XY(Top(Ie,j),1);
        Ye(j)= XY(Top(Ie,j),2);

    end

%*****Cuadratico*****%
CoordGL = [Xe' Ye'];
M0_e = zeros(18);
%-----Integracion numérica-----

npg= 3; % número de puntos de gauss
[CoordPG, H] = pgauss(npg);

for i = 1:npg %inicia el ciclo en 1 hasta que llegue a npg
    Xg = CoordPG(i,1); %reemplaza del valor de la coordenada η.
    Hi= H(i);
    for j = 1:npg
        Yg = CoordPG(j,2); %reemplaza del valor de la coordenada ζ.

        %funciones de forma
        [Ne,dN,] = fformaa_T6(Xg,Yg);
        Jac =[dN(2,2:3:18)*Xe' dN(2,2:3:18)*Ye'; dN(1,1:3:18)*Xe' dN(1,1:3:18)*Ye'];
        DetJac = det(Jac);

        %.....calculo de la matriz de rigidez.....
        M0_e = M0_e + Ne'*dens*Ne*DetJac*Hi;
    end
end
%-----Fin del calculo de la matriz de un solo elemento-----

```

```
%..... Ensamblado de la Matriz Global con TRIANGULO CUADRATICO .....

LM=[2*Top(Ie,1)-1;
    2*Top(Ie,1);
    2*Top(Ie,2)-1;
    2*Top(Ie,2);
    2*Top(Ie,3)-1;
    2*Top(Ie,3);
    2*Top(Ie,4)-1;
    2*Top(Ie,4);
    2*Top(Ie,5)-1;
    2*Top(Ie,5);
    2*Top(Ie,6)-1;
    2*Top(Ie,6)];

% Ensamble final

for j= 1:2*NNpe
    for k= 1:2*NNpe
        Ien= LM(j);
        Jen= LM(k);

        M0(Ien,Jen)= M0(Ien,Jen) + M0_e(j,k);
    end
end
end
```