

Universidad de Sevilla
Escuela Politécnica Superior de Ingeniería

Programación del Método de los Elementos Finitos para elasticidad plana y tratamiento del error

Proyecto Fin de Carrera presentado por Pablo Palomares Cuaresma,
en satisfacción de los requisitos necesarios para optar al grado de
Ingeniero Técnico Industrial, especialidad en Mecánica. Dirigido por el
profesor D. Sergio Miguel Marcos Ruíz

El alumno,

Fdo.: Pablo Palomares Cuaresma

El director,

Fdo.: Sergio Miguel Marcos Ruíz

A mis padres, Pablo y Gregoria.

A mis hermanos.

Dime y lo olvido, enséñame y lo recuerdo, involúcrame y lo aprendo.

Benjamins Franklin

AGRADECIMIENTOS

Quisiera agradecer a todos mis maestros/as y profesores/as el apoyo prestado durante todos mis años de estudio y mi más grato reconocimiento a sus palabras y sabios consejos. A mis compañeros, sin los cuales el largo camino recorrido no hubiese sido el mismo.

También quisiera agradecer a mi familia su inmenso apoyo, amor y confianza que en mí siempre han depositado ya que este trabajo es fruto de ello. Muy en especial a mis padres y mis abuelas, quienes me han dado todo desde pequeño.

Por último, dedicar todo mi trabajo a Estefanía como muestra de cariño y agradecimiento, por todos los grandes momentos compartidos con ella para poder llegar a este día como la persona que soy.

Índice

CAPÍTULO I. Introducción	2
1.1. Objetivos del proyecto	3
 CAPÍTULO II. Métodos	 6
2.1. Fundamentos de elasticidad.....	6
2.2. Elasticidad bidimensional	9
2.3. Principio de los trabajos virtuales (PTV)	12
2.4. Principios del MEF	12
2.4.1. Elemento triangular de tres nodos.....	13
2.4.2. Elemento cuadrangular de cuatro nodos	20
2.4.3. Campo de desplazamientos	24
2.4.4. Campo de tensiones	24
2.5. Estimación del error a posteriori	25
2.6. Tensión de von Mises	26
 CAPÍTULO III. Descripción del programa	 29
3.1. Función principal “MEFBID”	30
3.2. Función de entrada de datos.....	31
3.3. Función “KESTRUCT”	31
3.3.1. Subfunción “KELEMEN”	31
3.4. Función “FESTRUCT”	32
3.4.1. Subfunción “FEQUIV”	33
3.5. Función “SOLUC”	34
3.6. Función “TALIS”	34
3.7. Función “ERROR”	35
3.7.1. Subfunción “TDISC”	36
3.8. Función “TVM”	36
3.9. Subfunciones comunes	36
3.10. Variables del programa	38

CAPÍTULO IV. Resultados y análisis	43
4.1. Resultados del programa <i>MEFBID</i>	44
4.2. Resultados analíticos del problema	62
4.3. Resultados de otros programas	62
4.4. Análisis de los resultados	64
4.4.1. Análisis de los tiempos de cálculo	65
4.4.2. Análisis de los elementos utilizados	66
4.4.3. Análisis de la norma de energía	67
4.4.4. Análisis de la convergencia	68
4.5. Otro resultado	68
 CAPÍTULO V. Conclusiones	 72
5.1. Extensión del programa <i>MEFBID</i>	73
 CAPÍTULO VI. Referencias	 76
 ANEXO I. Código fuente.....	 78
I.1. Funciones principales.....	78
I.2. Subfunciones	90

CAPÍTULO I

Introducción

Contenido

1. Introducción.....	2
1.1. Objetivos del proyecto	3

1. Introducción

La Teoría de la Elasticidad determina el comportamiento de una estructura ante cargas exteriores por medio de la resolución de un sistema de ecuaciones de gobierno, teniendo en cuenta unas condiciones de contorno.

La Resistencia de materiales añade hipótesis que simplifican la Teoría de la Elasticidad en la resolución de las ecuaciones de gobierno, obteniéndose soluciones cercanas a las exactas y que se consideran validas.

Sin embargo, la Resistencia de materiales normalmente solo se puede aplicar a estructuras con nudos articulados y rígidos, siendo muy difícil su aplicación a estructuras más complejas como las estructuras continuas. Para este tipo de estructuras es necesario aplicar métodos aproximados como el Método de los elementos finitos (MEF). Además de análisis de estructuras, el MEF es una herramienta muy útil con aplicaciones en otras muchas áreas de la ingeniería, tales como movimientos de fluidos, transmisión de calor, redes eléctricas e incluso en áreas alejadas de ésta como son la medicina computacional o en procesos de farmacia y química.

Con la llegada de las computadoras digitales los problemas discretos pueden resolverse generalmente sin dificultad, no siendo así para los problemas continuos, que solo se pueden resolver de forma exacta mediante manipulaciones matemáticas, limitando el cálculo exacto a problemas extremadamente sencillos. Para vencer esta dificultad, el MEF utiliza un método de discretización, efectuando una aproximación, de tal manera que ésta se acerca a la solución continua a medida que crece el número de variables discretas. La mayoría de sistemas discretos, y así de sistemas continuos discretizados, pueden analizarse utilizando técnicas de cálculo matricial muy similares.

Es decir, una vez discretizado el problema continuo podemos trabajar matricialmente para su resolución, lo que hace que una de las etapas fundamentales en la aplicación con éxito del MEF al cálculo de estructuras es la implementación de la teoría en un programa de ordenador eficiente. Aunque en la actualidad ya existen en el mercado una gran cantidad de programas, un buen conocimiento de la programación de MEF puede ser de gran interés para el técnico, tanto o más que la propia teoría en la que se basa el método. Para el técnico que utiliza un programa para cálculo de estructuras es importante conocer la mejor forma de programar el MEF ya que esto representa la posibilidad de adaptarlo a su ordenador para incrementar el rendimiento de éste así como reducir el coste de esta fase del proyecto.

Muchos programas son los que nos permiten implementar el MEF, pero debido a su trabajo matricial, se hace necesario el uso de un programa que nos

permita fácilmente la declaración y el trabajo con matrices. Aunque FORTRAN continúa siendo un lenguaje de referencia en el uso del MEF, nuevos lenguajes y herramientas de programación permiten la simplificación en la escritura de los algoritmos al tiempo que optimizan los recursos disponibles. Este es el caso de Matlab, el cual nos permite, con un lenguaje de programación de alto nivel, trabajar de forma sencilla con vectores y matrices. Además nos ofrece la posibilidad, por medio de la herramienta pdetool, de dividir una figura bidimensional en elemento triangulares y volcar fácilmente la conectividad de cada elemento y las coordenadas de sus nodos.

Por tanto, resulta interesante, desde el punto de vista del ingeniero, poder implementar en Matlab un programa con el que poder calcular estructuras mediante el MEF, ya sea para uso comercial o para un uso meramente académico.

1.1. Objetivos del proyecto

El objetivo global del presente proyecto consiste en la programación en Matlab de un programa con el que calcular elementos bidimensionales de Tensión y Deformación plana, centrándose principalmente en el estudio mediante elementos triangulares de tres nodos y elementos cuadrangulares de cuatro nodos con cargas puntuales en nodos y distribuidas en el borde de un elemento. Para la consecución de este objetivo, se plantean los siguientes objetivos específicos:

- a) Implementación en ficheros de todas las matrices necesarias para la obtención de la matriz de rigidez del elemento para, a partir de esta, construir la matriz de rigidez completa de la estructura.
- b) Cálculo del vector de cargas equivalente de cada una de las fuerzas exteriores aplicadas a la estructura para construir el vector de cargas en nodos de la estructura.
- c) Resolución del sistema de ecuaciones matricial $[K] \cdot [\delta] = [F]$ donde se calculan los desplazamientos desconocidos de los nodos. Cálculo de la deformada de la estructura a partir del desplazamiento de los nodos.
- d) Cálculo de las tensiones en los nodos de los elementos como extrapolación de los valores en puntos óptimos del elemento a los nodos de éste y alisando los distintos valores para un mismo nodo.
- e) Estimación del error de discretización como norma energética de la diferencia entre la tensión alisada y la solución nodal discontinua.

- f) Cálculo de la tensión de Von Mises a partir de la solución alisada de las tensiones en los nodos de la estructura.

Para el análisis de los resultados se realizará el estudio de un problema tipo del que se conozca el resultado exacto. Además, se comparará los resultados obtenidos con el programa con los obtenidos en otro programa de cálculo de elementos finitos.

Como se observa, queda fuera del presente proyecto la creación de un motor de entrada de datos y la presentación detallada de los resultados obtenidos con el programa ya que la capacidad gráfica de Matlab es muy limitada. Esto se podría realizar de forma más eficiente con cualquier pre / pos procesador del mercado, por ejemplo la herramienta GID.

CAPÍTULO II

Métodos

Contenido

2. Métodos	6
2.1. Fundamentos de elasticidad	6
2.2. Elasticidad bidimensional	9
2.3. Principio de los trabajos virtuales (PTV)	12
2.4. Principios del MEF	12
2.4.1. Elemento triangular de tres nodos	13
2.4.2. Elemento cuadrangular de cuatro nodos	20
2.4.3. Campo de desplazamientos	24
2.4.4. Campo de tensiones	24
2.5. Estimación del error a posteriori	25
2.6. Tensión de von Mises	26

2. Métodos

Para poder llevar a cabo la implementación en Matlab del programa de cálculo necesario para el cálculo de estructuras de elasticidad bidimensional, previamente se realizará una exposición de los fundamentos teóricos necesarios para su realización.

En primer lugar se desarrollarán los fundamentos de elasticidad necesarios para resolver cualquier problema, aplicándolos más tarde a problemas de elasticidad bidimensional plana. Se realizará una breve explicación del Principio de los Trabajos Virtuales (PTV), necesarios para obtener la ecuación matricial de equilibrio de un elemento.

Por último se desarrollan los fundamentos necesarios para aplicar el MEF a estructuras bidimensionales.

Además de los fundamentos desarrollados aquí, se hace uso de conceptos matemáticos necesarios pero que no se incluyen su exposición. Para la adquisición de estos conceptos basta con acudir a las referencias bibliográficas expuestas. La información necesaria para comprender el funcionamiento de Matlab se realizara sobre el propio programa, indicando en cada caso el comportamiento de cada código desarrollado.

2.1. Fundamentos de elasticidad

Sea un cuerpo sometido a un sistema de fuerzas aplicadas cualesquiera, el estado de esfuerzos en un punto cualquiera del cuerpo está definido por el vector de esfuerzos:

$$\{\sigma\} = [\sigma_x \quad \sigma_y \quad \sigma_z \quad \tau_{xy} \quad \tau_{yz} \quad \tau_{zx}]^T \quad (2.1)$$

Este vector está formado por los esfuerzos normales $\sigma_x, \sigma_y, \sigma_z$ y los esfuerzos tangenciales $\tau_{xy}, \tau_{yz}, \tau_{zx}$. Estos esfuerzos constituyen las componentes de estado de esfuerzos de un punto considerado.

Las ecuaciones de equilibrio se obtienen aislando del cuerpo un elemento paralelepípedo recto de volumen diferencial con vértice en el punto anterior. Éste estará en equilibrio bajo la acción de las fuerzas que actúan en cada cara y de las fuerzas de volumen $q \cdot dx \cdot dy \cdot dz$. Las ecuaciones de equilibrio de la estática $\Sigma X = 0, \Sigma Y = 0, \Sigma Z = 0$, se obtienen las ecuaciones del paralelepípedo elemental:

$$\begin{aligned}
\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + q_x &= 0 \\
\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + q_y &= 0 \\
\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + q_z &= 0
\end{aligned} \tag{2.2}$$

Las ecuaciones Estática $\Sigma M_x = 0, \Sigma M_y = 0, \Sigma M_z = 0$, permiten determinar las relaciones entre esfuerzos cortantes:

$$\tau_{yz} = \tau_{zy}, \tau_{zx} = \tau_{xz}, \tau_{xy} = \tau_{yx} \tag{2.3}$$

Éstas son la expresión analítica del teorema de Cauchy. Este teorema reduce a 6 las componentes independientes de esfuerzos que son las componentes que intervienen en el vector de esfuerzos $\{\sigma\}$.

Del mismo modo, el estado de deformaciones en un punto cualquiera estará definido por el vector de deformaciones:

$$\{\varepsilon\} = [\varepsilon_x \quad \varepsilon_y \quad \varepsilon_z \quad \gamma_{xy} \quad \gamma_{yz} \quad \gamma_{zx}]^T \tag{2.4}$$

Este vector está formado por las deformaciones lineales $\varepsilon_x, \varepsilon_y, \varepsilon_z$ y las deformaciones angulares $\gamma_{xy}, \gamma_{yz}, \gamma_{zx}$ que constituyen las componentes del estado de deformaciones.

Cuando el cuerpo es de un material isótropo y linealmente elástico, como en nuestro caso, satisface la ley de Hooke generalizada que determina las componentes del estado de deformaciones en función de las componentes del estado de tensiones:

$$\begin{aligned}
\varepsilon_x &= \frac{\sigma_x}{E} - \frac{\nu}{E} \cdot (\sigma_y + \sigma_z) & \gamma_{xy} &= \frac{2 \cdot (1 + \nu) \cdot \tau_{xz}}{E} \\
\varepsilon_y &= \frac{\sigma_y}{E} - \frac{\nu}{E} \cdot (\sigma_z + \sigma_x) & \gamma_{yz} &= \frac{2 \cdot (1 + \nu) \cdot \tau_{yz}}{E} \\
\varepsilon_z &= \frac{\sigma_z}{E} - \frac{\nu}{E} \cdot (\sigma_x + \sigma_y) & \gamma_{zx} &= \frac{2 \cdot (1 + \nu) \cdot \tau_{zx}}{E}
\end{aligned} \tag{2.5}$$

De las ecuaciones anteriores se deducen las ecuaciones de Lamé que determinan las componentes del estado de esfuerzos en función del estado de deformaciones

$$\sigma_x = \lambda \cdot (\varepsilon_x + \varepsilon_y + \varepsilon_z) + 2 \cdot \mu \cdot \varepsilon_x$$

$$\begin{aligned}\sigma_y &= \lambda \cdot (\varepsilon_x + \varepsilon_y + \varepsilon_z) + 2 \cdot \mu \cdot \varepsilon_y \\ \sigma_z &= \lambda \cdot (\varepsilon_x + \varepsilon_y + \varepsilon_z) + 2 \cdot \mu \cdot \varepsilon_z\end{aligned}\quad (2.6)$$

$$\tau_{xy} = \frac{E \cdot \gamma_{xy}}{2 \cdot (1 + \nu)}$$

$$\tau_{yz} = \frac{E \cdot \gamma_{yz}}{2 \cdot (1 + \nu)}$$

$$\tau_{zx} = \frac{E \cdot \gamma_{zx}}{2 \cdot (1 + \nu)}$$

Siendo λ y μ los coeficientes de Lamé:

$$\lambda = \frac{\nu \cdot E}{(1 + \nu) \cdot (1 - 2 \cdot \nu)} \quad \mu = \frac{E}{2 \cdot (1 - 2 \cdot \nu)}$$

Las ecuaciones (2.6) se pueden expresar en la forma matricial

$$\{\sigma\} = [D] \cdot \{\varepsilon\} \quad (2.7)$$

Donde $[D]$ es la matriz esfuerzos-deformaciones o matriz constitutiva.

$$[D] = \frac{E}{(1 + \nu)(1 - 2\nu)} \begin{bmatrix} (1 - \nu) & \nu & \nu & 0 & 0 & 0 \\ \nu & (1 - \nu) & \nu & 0 & 0 & 0 \\ \nu & \nu & (1 - \nu) & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{(1 - 2 \cdot \nu)}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{(1 - 2 \cdot \nu)}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{(1 - 2 \cdot \nu)}{2} \end{bmatrix} \quad (2.8)$$

La matriz constitutiva es una matriz simétrica, función del módulo de elasticidad E y el coeficiente de Poisson μ .

La relación que existe entre las componentes del estado de deformaciones en un punto y las componentes del desplazamiento del punto ($u(x, y, z), v(x, y, z), w(x, y, z)$) son las relaciones de compatibilidad de las deformaciones, que matricialmente se pueden expresar de la siguiente forma:

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{bmatrix} = \begin{bmatrix} \partial/\partial x & 0 & 0 \\ 0 & \partial/\partial y & 0 \\ 0 & 0 & \partial/\partial z \\ \partial/\partial y & \partial/\partial x & 0 \\ 0 & \partial/\partial z & \partial/\partial y \\ \partial/\partial z & 0 & \partial/\partial x \end{bmatrix} \cdot \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.9)$$

De forma resumida queda

$$\{\varepsilon\} = [\partial] \cdot \{u\} \quad (2.10)$$

Donde $[\partial]$ es un operador matricial formado por operadores en derivadas parciales y $\{u\}$ el vector de desplazamiento del punto.

2.2. Elasticidad bidimensionales

El presente trabajo se reduce a la resolución de problemas de elasticidad bidimensional. Las estructuras bidimensionales son aquellas en las que todos los puntos del cuerpo se encuentran bajo un estado de tensión plana o un estado de deformación plana.

Se dice que una estructura bidimensional está en estado de tensión plano si una de sus dimensiones (espesor) es mucho menor que las otras dos y sobre ella actúan solamente cargas contenidas en su plano medio. Ejemplos de tensión plana son vigas de gran canto, placas con cargas en su plano...

Se dice que una estructura se encuentra en estado de deformación plano si una de sus dimensiones (longitud) es mucho mayor que las otras dos y actúan sobre ella cargas distribuidas en su longitud y contenidas en planos ortogonales al eje. De entre este tipo de problemas cabe destacar el cálculo de muros de contención, presas de gravedad, tuberías con presión interior...

La teoría de la elasticidad bidimensional permite reducir las ecuaciones de la Teoría de la Elasticidad enunciadas anteriormente a ecuaciones más sencillas con las que trabajar.

– Campo de desplazamientos

Las características de una estructura en estado de tensión o deformación plana permiten establecer la hipótesis de que todas las secciones perpendiculares al eje prismático z se deforman en su plano y de la misma forma. De esta forma, el

campo de desplazamiento de esta sección contenida en el plano $x - y$ queda completamente definido si se conocen los desplazamientos en las direcciones x e y de sus puntos. Es decir, el vector de desplazamientos de un punto se define como

$$\mathbf{u}(x, y) = \begin{Bmatrix} u(x, y) \\ v(x, y) \end{Bmatrix} = \begin{Bmatrix} u \\ v \end{Bmatrix} \quad (2.11)$$

– Campo de deformaciones y tensiones

Considerando el campo de desplazamientos anterior, en un estado de esfuerzos plano, los esfuerzos actúan en dicho plano y son nulos en la dirección z ($\sigma_z = \tau_{zx} = \tau_{yz} = 0$). En este caso, las deformaciones angulares en el plano z son nulas ($\gamma_{zx} = \gamma_{yz} = 0$), pero no así las deformaciones ε_z cuyo valor se obtiene a partir de (2.5).

De la misma forma, si consideramos que el plano que corresponde al estado plano es el xy , en un estado de deformación plano, las deformaciones en el eje z son nulas ($\varepsilon_z = \gamma_{zx} = \gamma_{yz} = 0$), no siendo nulos los esfuerzos en dicha dirección y cuyo valor se puede obtener a partir de (2.6).

Como se puede observar, en ambos casos se verifica que $\sigma_z \cdot \varepsilon_z = 0$ y por ello el trabajo de deformación realizado por ellos siempre será nulo. Por esta razón los vectores de esfuerzo y deformación significativos de un punto se pueden reducir a los términos que producen trabajo de deformación en la estructura y se definen de la misma forma para tensión y deformación plana. Estos vectores se definen para ambos casos como:

$$\{\sigma\} = \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \quad \{\varepsilon\} = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} \quad (2.12)$$

– Relación tensión - deformación

La relación entre tensión y deformación, como se indicó en (2.7), se pueden indicar en la forma matricial

$$\{\sigma\} = [D] \cdot \{\varepsilon\}$$

Con las hipótesis indicadas anteriormente sobre la tensión y deformación en el eje z , la matriz constitutiva $[D]$ de un material isótropo se puede reducir dependiendo del estado plano en el que nos encontremos.

Para un estado de deformación plano, las deformaciones en el eje z son nulas ($\varepsilon_z = \gamma_{zx} = \gamma_{yz} = 0$), por ello la matriz constitutiva se puede obtener eliminando de (2.8) las filas y columnas 3, 5 y 6, con lo que se obtiene

$$[D] = \frac{E}{(1+\nu)(1+2\nu)} \begin{vmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{vmatrix} \quad (2.13)$$

Para un estado de tensión plano los esfuerzos en el eje z son nulos ($\sigma_z = \tau_{zx} = \tau_{yz} = 0$). En este caso, las deformaciones ε_z no son nulas, por lo que sustituyendo su valor en la relación matricial (2.7) la matriz constitutiva queda

$$[D] = \frac{E}{(1-\nu^2)} \begin{vmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{vmatrix} \quad (2.14)$$

Como se puede observar, la relación matricial entre tensión y deformación queda reducida en un estado plano a

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = [D] \cdot \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} \quad (2.15)$$

Donde la matriz constitutiva $[D]$ depende del estado plano en el que se encuentre la estructura, tomado el valor de (2.13) para deformación plana y (2.14) para tensión plana.

– Relación deformación - desplazamientos

La relación de compatibilidad de las deformaciones se puede obtener de (2.10). Teniendo en cuenta que el vector de desplazamientos ($\{u\}$) en elasticidad bidimensional queda reducido a (2.11) y el vector de deformaciones ($\{\varepsilon\}$) a (2.12), al operador matricial $[\partial]$ se le pueden eliminar los términos nulos y la relación entre deformación y desplazamientos (2.9) queda

$$\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \partial/\partial x & 0 \\ 0 & \partial/\partial y \\ \partial/\partial y & \partial/\partial x \end{bmatrix} \cdot \begin{Bmatrix} u \\ v \end{Bmatrix} \quad (2.16)$$

El planteamiento de los apartados 2.1 y 2.2 están basados en las descripciones de Vázquez et López (2001) y Oñate (1995).

2.3. Principio de los trabajos virtuales

La expresión integral de equilibrio se puede obtener haciendo uso del Principio de los Trabajos Virtuales (PTV). Este establece que la condición necesaria y suficiente para que una estructura esté en equilibrio es que la suma de los trabajos que realizan todas las fuerzas que actúan sobre la estructura sean nulos. Es decir, que la suma de los trabajos que realizan las fuerzas externas a la estructura y el trabajo que realizan las fuerzas internas de la estructura sean nulos (Vázquez et López, 2001).

$$W_e + W_i = 0 \quad (2.17)$$

Para una estructura con un estado de tensiones o deformaciones plano, teniendo en cuenta las tensiones y deformaciones que contribuyen al trabajo virtual de la estructura, la expresión del PTV puede escribirse como

$$\begin{aligned} \iint_A (\{\delta\epsilon_x\}\sigma_x + \{\delta\epsilon_y\}\sigma_y + \{\delta\gamma_{xy}\}\tau_{xy})t \cdot dA = \iint_A (\{\delta u\}q_x + \{\delta v\}q_y)t \cdot dA \\ + \oint_l (\{\delta u\}p_x + \{\delta v\}p_y)t \cdot ds + \sum_i (\{\delta u_i\}P_i^x + \{\delta v_i\}P_i^y) \end{aligned} \quad (2.18)$$

En esta ecuación, el primer miembro representa el trabajo que las tensiones interiores realizan sobre unas deformaciones virtuales. El segundo miembro de la ecuación representa el trabajo de las fuerzas exteriores sobre unos desplazamientos virtuales. El trabajo de las fuerzas exteriores queda como suma de los trabajos de las fuerzas repartidas por unidad de volumen (q_x y q_y), las fuerzas repartidas sobre el contorno (p_x y p_y) y las fuerzas puntuales (P_i^x y P_i^y).

2.4. Fundamentos del MEF

El MEF determina el comportamiento de una estructura ante las cargas sustituyendo la solución continua y exacta (imposible de conseguir en la mayoría de los casos) por una solución discreta y, por lo tanto, aproximada de la solución exacta. Para ello, divide la estructura en elementos finitos no diferenciales, interconectados entre sí a través de los nodos.

Después de realizar el estudio independiente de cada elemento se procede a recomponer la estructura restableciendo el equilibrio y compatibilidad de los nodos. Esto da lugar a un sistema de ecuaciones algebraicas resoluble que permite hallar

los desplazamientos en los nodos y calcular a partir de éstos las demás incógnitas del sistema, aplicando las ecuaciones de elasticidad descritas anteriormente.

A continuación se describirá para el problema bidimensional que nos ocupa las distintas fases del MEF para los elementos triangulares y cuadrangulares, es decir:

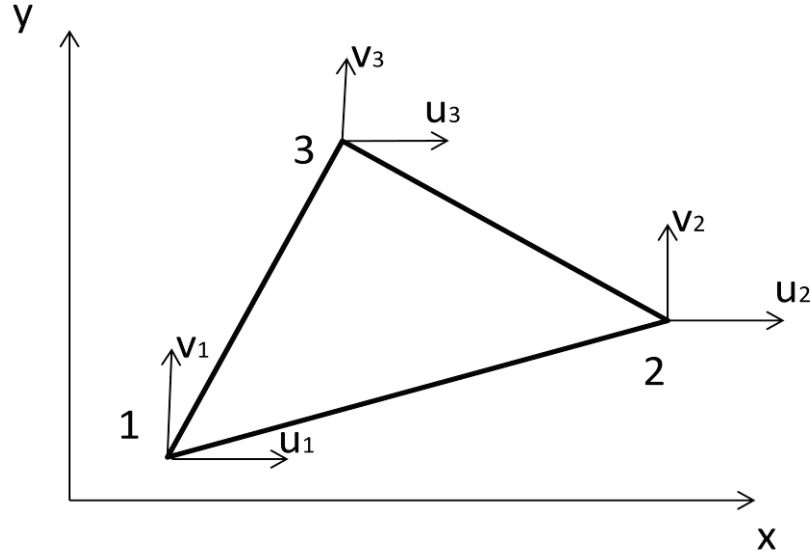
- Discretización del campo desplazamientos del elemento
- Discretización del campo de deformaciones
- Matriz de rigidez de la estructura
- Ecuaciones de equilibrio
- Respuesta de la estructura

Como se verá más adelante, para el cálculo de algunas de las matrices será necesario realizar el cálculo de algunas integrales. El cálculo analítico de estas integrales es, salvo en sencillas estructuras, inabordable. Por ello, para el cálculo de las integrales del elemento se hará uso de la integración numérica para evaluarlas de forma precisa y sencilla. El método de integración numérica más utilizado para el MEF es la cuadratura de Gauss-Legendre, aplicada en dos dimensiones en nuestro caso. Además, para poder hacer uso de esta integración numérica hay que realizar en el elemento una formulación isoparamétrica, ya que se necesita obtener unos límites de integración $[-1,1]$.

La formulación isoparamétrica transforma un elemento triangular lineal en un triángulo rectángulo con el origen de coordenadas en el vértice de mayor ángulo y transforma un elemento cuadrangular en un elemento cuadrado con el centro en el origen de coordenadas. Esta formulación exige la utilización de coordenadas naturales, las cuales se determinan mediante de relaciones de longitud y área.

2.4.1. Elemento triangular de tres nodos

Realizaremos en primer lugar el desarrollo del MEF considerando un elemento triangular de tres nodos, teniendo en cuenta que las matrices generales obtenidas se pueden adaptar a cualquier otro elemento bidimensional, como se verá más adelante para el elemento cuadrangular de cuatro nodos.

Figura 2.1. Elemento triangular de tres nodos. Grados de libertad nodales.

Considerando como elemento bidimensional un elemento triangular con nodos en cada uno de sus vértices, los desplazamientos nodales de un elemento aislado serán los que se indican en la figura 2.1. Como se dijo, el vector desplazamiento para un elemento con elasticidad bidimensional tomara la forma de (2.1). Los desplazamientos en ejes cartesianos de un punto cualquiera del elemento se puede definir en función de los desplazamientos de sus nodos como

$$u(x, y) = N_1 u_1 + N_2 u_2 + N_3 u_3 \quad (2.19)$$

$$v(x, y) = N_1 v_1 + N_2 v_2 + N_3 v_3$$

Donde N_i es la función de forma del nodo i del elemento. Esta expresión se puede poner en forma matricial

$$\{u_e\} = \begin{bmatrix} N_1 & N_2 & N_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & N_1 & N_2 & N_3 \end{bmatrix} \{\delta_e\} = [N_e] \{\delta_e\} \quad (2.20)$$

Las expresiones de las funciones de forma se obtienen aproximando el vector de desplazamientos del elemento finito por las funciones de desplazamiento

$$u(x, y) = \alpha_1 + \alpha_2 x + \alpha_3 y \quad (2.21)$$

$$v(x, y) = \alpha_4 + \alpha_5 x + \alpha_6 y$$

Que en su forma matricial queda

$$\{u_e\} = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix} \{\alpha\} = [P] \{\alpha\} \quad (2.22)$$

Donde $\{\alpha\}$ es un vector columna que contiene todos los coeficientes de (2.21). Si particularizamos esta ecuación a los desplazamientos en los nodos del elemento se obtiene la expresión, en forma matricial

$$\{\delta_e\} = \begin{bmatrix} 1 & x_1 & y_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_1 & y_1 \\ 1 & x_2 & y_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_2 & y_2 \\ 1 & x_3 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_3 & y_3 \end{bmatrix} \{\alpha\} = [C]\{\alpha\} \quad (2.23)$$

Si de esta ecuación matricial despejamos $\{\alpha\}$ y se sustituye en (2.22) nos queda

$$\{u_e\} = [P][C]^{-1}\{\delta_e\} \quad (2.24)$$

Comparando la ecuación matricial anterior con (2.20), se obtiene que la matriz de interpolación es

$$[N_e] = [P][C]^{-1} \quad (2.25)$$

Estas funciones de interpolación, también llamadas funciones de forma, tienen valor 1 en el nodo i y valor cero en los restantes nodos, es decir, son polinomios lagrangianos y, en consecuencia, el elemento triangular es un elemento lagrangiano. Las funciones de forma verifican que $N_1 + N_2 + N_3 = 1$ en cualquier punto del elemento.

Estas funciones se pueden expresar de la forma

$$N_i = \frac{(a_i + b_i x + c_i y)}{2A_e} \quad (2.26)$$

Donde A_e es el área del elemento y

$$\begin{aligned} a_i &= x_j y_k - x_k y_j \\ b_i &= y_j - y_k \\ c_i &= x_k - x_j \end{aligned} \quad (2.27)$$

Estas funciones de forma coinciden con las coordenadas de área del elemento, es decir, estas funciones de forma se pueden indicar en función de las coordenadas de área (ξ_1, ξ_2, ξ_3) como

$$N_1 = \xi_1, \quad N_2 = \xi_2, \quad N_3 = \xi_3. \quad (2.28)$$

Además, estas coordenadas de área no son independientes ya que se tiene que satisfacer la relación $\xi_1 + \xi_2 + \xi_3 = 1$ y por ello las funciones de forma pueden

indicarse en función únicamente de dos coordenadas naturales (ξ, η) , que transforman un triángulo cualquiera en un triángulo rectángulo, como se indica en la figura 2.2, con el vértice 1 en el origen del sistema de ejes ξ, η y sus lados definidos por las rectas $1 - \xi - \eta = 0$, $\xi = 0$, $\eta = 0$.

Con este cambio en el sistema de ejes, las funciones de forma quedan

$$N_1 = 1 - \xi - \eta, \quad N_2 = \xi, \quad N_3 = \eta. \quad (2.29)$$

Las coordenadas cartesianas de un punto del elemento pueden expresarse en función de las coordenadas naturales mediante estas funciones de forma como

$$\begin{aligned} x &= \sum_i^n N_i(\xi, \eta) x_i \\ y &= \sum_i^n N_i(\xi, \eta) y_i \end{aligned} \quad (2.30)$$

Siendo n el numero de nodos del elemento.

Una vez obtenidas las funciones de forma, discretización del campo de desplazamientos en el interior de un elemento, se procede a discretizar el campo de deformaciones. El vector de deformaciones en coordenadas cartesianas se obtiene sustituyendo (2.19) en (2.16) lo que nos da en forma matricial

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \tau_{xy} \end{bmatrix} = \begin{bmatrix} \partial N_1 / \partial x & 0 & \partial N_2 / \partial x & 0 & \partial N_3 / \partial x & 0 \\ 0 & \partial N_1 / \partial y & 0 & \partial N_2 / \partial y & 0 & \partial N_3 / \partial y \\ \partial N_1 / \partial y & \partial N_1 / \partial x & \partial N_2 / \partial y & \partial N_2 / \partial x & \partial N_3 / \partial y & \partial N_3 / \partial x \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{bmatrix} \quad (2.31)$$

Esta relación matricial se puede poner como

$$\{\varepsilon_e\} = [B_e] \{\delta_e\} \quad (2.32)$$

Donde $[B_e]$ es la matriz de deformación del elementos, definida por

$$[B_e] = [\partial][N_e] \quad (2.30)$$

Para poder obtener la discretización del campo de deformaciones en coordenadas naturales se hace uso de la regla de la derivación en cadena, la cual nos permite obtener, en forma matricial

$$\begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{Bmatrix} = [J_e] \begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{Bmatrix} \quad (2.33)$$

Donde $[J_e]$ es el jacobiano de la transformación de coordenadas naturales a cartesianas. De esta relación matricial se deduce

$$\begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{Bmatrix} = [J_e]^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix} = \frac{1}{|J_e|} \begin{bmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial y}{\partial \xi} \\ -\frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \xi} \end{bmatrix} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix} \quad (2.34)$$

Donde $|J_e|$ es el determinante del jacobiano. Este determinante también permite realizar el cambio del diferencial de área a coordenadas naturales de la forma

$$dx \cdot dy = |J_e| \cdot d\xi \cdot d\eta \quad (2.35)$$

Si introducimos en (2.29) los términos deducidos en (2.33) se obtiene la matriz de deformación en coordenadas naturales. La matriz de deformación se puede poner de la forma

$$\{B_e\} = [B_1 \quad B_2 \quad B_3] \quad (2.36)$$

Donde $[B_i]$ son las submatrices de la matriz de deformación que vienen definida por

$$[B_i] = \frac{1}{|J_e|} \begin{bmatrix} \frac{\partial y}{\partial \eta} \frac{\partial N_i}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial N_i}{\partial \eta} & 0 \\ 0 & \frac{\partial x}{\partial \xi} \frac{\partial N_i}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial x}{\partial \xi} \frac{\partial N_i}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial N_i}{\partial \xi} & \frac{\partial y}{\partial \eta} \frac{\partial N_i}{\partial \xi} - \frac{\partial y}{\partial \xi} \frac{\partial N_i}{\partial \eta} \end{bmatrix} \quad (2.37)$$

Estos términos se obtienen derivando las funciones de forma en coordenadas naturales (2.28) y las funciones (2.29). Para un elemento triangular de tres nodos, la matriz de deformación es

$$\{B_e\} = \frac{1}{|J_e|} \begin{bmatrix} y_2 - y_3 & 0 & y_3 - y_1 & 0 & y_1 - y_2 & 0 \\ 0 & x_3 - x_2 & 0 & x_1 - x_3 & 0 & x_2 - x_1 \\ x_3 - x_2 & y_2 - y_3 & x_1 - x_3 & y_3 - y_1 & x_2 - x_1 & y_1 - y_2 \end{bmatrix} \quad (2.38)$$

La expresión discretizada del vector de tensiones se obtiene sustituyendo la ecuación (2.31) en (2.15).

$$\{\sigma_e\} = [D][B_e]\{\delta_e\} \quad (2.39)$$

Para la obtención de las ecuaciones de equilibrio del elemento de la discretización se parte de la expresión (2.18) del PTV para un estado de elasticidad plana. El trabajo interior de la estructura se puede poner de la forma matricial

$$W_i = - \iint_A (\{\partial \varepsilon\}^T \{\sigma\}) t \cdot dA \quad (2.40)$$

Si sustituimos las expresiones (2.39) y (2.31) en la expresión anterior se obtiene

$$W_i = - \iint_A (\{\partial \delta_e\}^T [B_e]^T [D] [B_e] \{\delta_e\}) t \cdot dA \quad (2.41)$$

Si suponemos un desplazamiento virtual $\{\partial \delta_e\}$ unitario en una dirección cualquiera en uno de sus nodos y nulo en el resto, el trabajo de las fuerzas internas queda

$$W_i = - \left(\iint_A ([B_e]^T [D] [B_e]) t \cdot dA \right) \{\delta_e\} \quad (2.42)$$

El trabajo de las fuerzas exteriores es

$$\iint_A (\{\partial u_e\}^T [q_e]) t \cdot dA + \oint_l (\{\partial u_e\}^T [p_e]) t \cdot ds + \sum_i (\{\partial \delta_e\} [P_e]) \quad (2.43)$$

Sustituyendo (2.20) en la expresión anterior, y teniendo en cuenta la suposición de que el desplazamiento $\{\partial \delta_e\}$ es unitario en una dirección cualquiera en uno de los nodos y nulo en el resto, la expresión queda

$$\iint_A ([N_e]^T [q_e]) t \cdot dA + \oint_l ([N_e]^T [p_e]) t \cdot ds + \sum_i (\{\partial \delta_e\} [P_e]) \quad (2.44)$$

Si introducimos (2.42) y (2.44) en la ecuación de equilibrio del PTV, ecuación (2.17) se obtiene la ecuación matricial de equilibrio del elemento

$$\begin{aligned} & \left(\iint_A ([B_e]^T [D] [B_e]) t \cdot dA \right) \{\delta_e\} = \\ & \iint_A ([N_e]^T [q_e]) t \cdot dA + \oint_l ([N_e]^T [p_e]) t \cdot ds + [P_e] \end{aligned} \quad (2.45)$$

Esta ecuación de equilibrio equivale a

$$[K_e] \{\delta_e\} = [F_e] \quad (2.46)$$

Donde $[K_e]$ es la matriz de rigidez del elemento, $\{\delta_e\}$ es el vector de desplazamientos nodales y $[F_e]$ es el vector de fuerzas nodales del elemento. La matriz de rigidez del elemento y el vector de fuerzas nodales vienen dados por

$$[K_e] = \iint_A ([B_e]^T [D] [B_e]) t \cdot dA \quad (2.47)$$

$$[F_e] = [F_{Pe}] + [F_{qe}] + [F_{pe}] \quad (2.48)$$

Donde el vector de fuerzas nodales se descompone en fuerzas aplicadas directamente en nodos $[F_{Pe}]$, fuerzas nodales equivalentes a las fuerzas distribuidas en el volumen del elemento $[F_{qe}]$ y en fuerzas nodales equivalentes a las fuerzas distribuidas en la superficie del elemento. Estos tres vectores vienen dados, para elementos bidimensionales, por

$$[F_{Pe}] = [P_e]$$

$$[F_{qe}] = \iint_A ([N_e]^T [q_e]) t \cdot dA \quad (2.49)$$

$$[F_{pe}] = \oint_l ([N_e]^T [p_e]) t \cdot ds$$

Además, la obtención de la relación de equilibrio de una estructura se realiza de igual forma que la desarrollada para obtener la ecuación de equilibrio del elemento, sustituyendo los términos nodales del elemento por términos nodales de la estructura discretizado. Con ello, la ecuación matricial de equilibrio de la estructura es

$$[K_0]\{\delta_0\} = [F_0] \quad (2.50)$$

Donde $[K_0]$ es la matriz completa de rigidez de la estructura, $\{\delta_0\}$ es el vector de desplazamientos nodales de la estructura y $[F_0]$ es el vector de cargas nodales de la estructura. Las matrices y vectores anteriores se obtienen sumando las matrices elementales ampliadas al número de grados de libertad de la estructura.

$$[K_0] = \sum_i^n [K_0]_i$$

$$[F_0] = \sum_i^n [F_0]_i \quad (2.51)$$

Donde n es el número de elementos de la estructura y $[K_0]_i$ es la matriz de rigidez ampliada del elemento i y $[F_0]_i$ es el vector de fuerzas nodales ampliado del elemento i .

Particularizando la matriz de rigidez del elemento para un elemento isoparamétrico tenemos, sustituyendo la ecuación del cambio de coordenadas (2.35) en la ecuación (2.47) se obtiene

$$[K_e] = \int_{-1}^1 \int_{-1}^1 ([B_e]^T [D] [B_e]) t \cdot |J_e| \cdot d\xi \cdot d\eta \quad (2.52)$$

Para el elemento triangular de tres nodos, la ecuación anterior queda

$$[K_e] = \int_0^1 \int_0^{1-\eta} ([B_e]^T [D] [B_e]) t \cdot |J_e| \cdot d\xi \cdot d\eta \quad (2.53)$$

Utilizando como integración numérica la cuadratura de Gauss-Legendre, la matriz de rigidez del elemento es

$$[K_e] = \sum_i (\phi(\xi_i, \eta_i)) W_i \quad (2.54)$$

Donde W_i son los pesos de los puntos de gauss y $(\phi(\xi_i, \eta_i))$ es la función

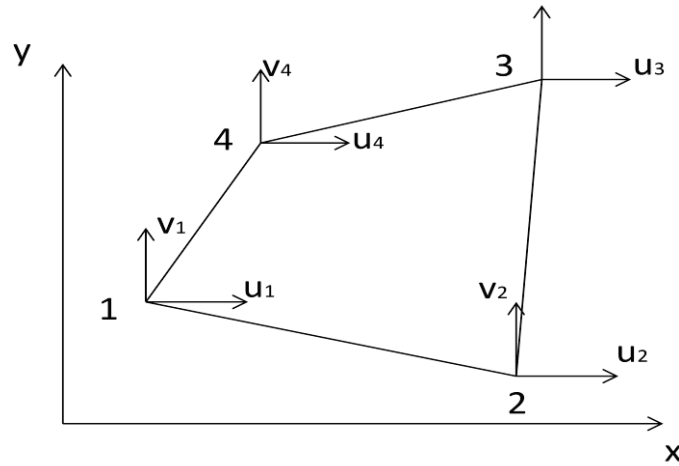
$$\phi(\xi_i, \eta_i) = [B_e]^T [D] [B_e] \cdot t \cdot |J_e| \quad (2.55)$$

Evaluada en los puntos de Gauss ξ_i, η_i .

2.4.2. Elemento cuadrangular de cuatro nodos

Como se especificó anteriormente, las ecuaciones generales obtenidas en el apartado anterior son directamente aplicables a otro tipo de elementos bidimensionales. Ahora se considerará como elemento bidimensional un elemento cuadrilateral con nodos en cada uno de sus vértices. Los desplazamientos nodales de un elemento aislado serán los que se indican en la figura 2.2.

Figura 2.2. Elemento cuadrangular lineal de cuatro nodos. Grados de libertad nodales.



El vector desplazamiento para un elemento con elasticidad bidimensional tomara la forma de (2.1). Los desplazamientos cartesianos de un punto cualquiera del elemento se puede definir en función de los desplazamientos de sus nodos como

$$\begin{aligned} u(x, y) &= N_1 u_1 + N_2 u_2 + N_3 u_3 + N_4 u_4 \\ v(x, y) &= N_1 v_1 + N_2 v_2 + N_3 v_3 + N_4 v_4 \end{aligned} \quad (2.56)$$

Donde N_i es la función de forma del nodo i del elemento. Esta expresión se puede poner en forma matricial

$$\{u_e\} = [N_e] \{\delta_e\} \quad (2.57)$$

En este caso, las funciones de desplazamiento son

$$\begin{aligned} u(x, y) &= \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy \\ v(x, y) &= \alpha_5 + \alpha_6 x + \alpha_7 y + \alpha_8 xy \end{aligned} \quad (2.58)$$

Que en su forma matricial queda

$$\{u_e\} = \begin{bmatrix} 1 & x & y & xy & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x & y & xy \end{bmatrix} \{\alpha\} = [P] \{\alpha\} \quad (2.59)$$

Donde $\{\alpha\}$ es un vector columna que contiene todos los coeficientes de (2.58). Si particularizamos esta ecuación a los desplazamientos en los nodos del elemento se obtiene una expresión idéntica a (2.25) pero añadiendo el termino xy en cada nodo

$$\{\delta_e\} = [C] \{\alpha\} \quad (2.60)$$

Con esto se obtiene una relación

$$[N_e] = [P][C]^{-1} \quad (2.61)$$

Las funciones de interpolación, al igual que las del elemento triangular, tienen valor 1 en el nodo i y valor cero en los restantes nodos, es decir, son polinomios lagrangianos y, en consecuencia, el elemento rectangular de cuatro nodos es un elemento lagrangiano. Las funciones de forma verifican que $N_1 + N_2 + N_3 + N_4 = 1$ en cualquier punto del elemento.

Para un elemento rectangular, las funciones de forma en coordenadas cartesianas son

$$\begin{aligned} N_1 &= \frac{(a-x)(b-y)}{4ab} & N_3 &= \frac{(a+x)(b+y)}{4ab} \\ N_2 &= \frac{(a+x)(b-y)}{4ab} & N_4 &= \frac{(a-x)(b-y)}{4ab} \end{aligned} \quad (2.62)$$

Si utilizamos en este caso las coordenadas de área lineales

$$\xi = \frac{x}{a} \quad \eta = \frac{y}{b} \quad (2.63)$$

Estas coordenadas naturales (ξ, η) , transforman un cuadrilátero cualquiera en un cuadrado, con el centro en el origen del sistema de ejes ξ, η . Con este cambio en el sistema de ejes, las funciones de forma quedan

$$\begin{aligned} N_1 &= \frac{(1-\xi)(1-\eta)}{4} & N_3 &= \frac{(1+\xi)(1+\eta)}{4} \\ N_2 &= \frac{(1+\xi)(1-\eta)}{4} & N_4 &= \frac{(1-\xi)(1-\eta)}{4} \end{aligned} \quad (2.64)$$

Las coordenadas cartesianas de un punto del elemento pueden expresarse en función de las coordenadas naturales, al igual que los elementos triangulares, mediante (2.29).

Una vez obtenidas las funciones de forma, discretización del campo de desplazamientos en el interior de un elemento, se procede a discretizar el campo de deformaciones. Al igual que los elementos triangulares, el campo de deformación de un elemento se puede expresar como (2.31), donde, en el caso de un elemento isoparamétrico cuadrangular se tiene:

$$\{B_e\} = [B_1 \quad B_2 \quad B_3 \quad B_4] \quad (2.65)$$

Donde $[B_1]$ son las submatrices de la matriz de deformación que vienen definida por (2.37). En este caso se obtienen cuatro submatrices ya que el elemento tiene cuatro nodos.

La expresión discretizada del vector de tensiones se obtiene sustituyendo la ecuación (2.31) en (2.15)

$$\{\sigma_e\} = [D][B_e]\{\delta_e\} \quad (2.66)$$

Para un elemento cuadrilateral de cuatro nodos la ecuación de equilibrio se obtiene por medio de las ecuaciones (2.40) - (2.45) y se obtiene la misma ecuación de equilibrio matricial

$$[K_e]\{\delta_e\} = [F_e]$$

Donde $[K_e]$ es la matriz de rigidez del elemento, $\{\delta_e\}$ es el vector de desplazamientos nodales y $[F_e]$ es el vector de fuerzas nodales del elemento. La matriz de rigidez del elemento y el vector de fuerzas nodales vienen dados por (2.47) y (2.48).

Donde el vector de fuerzas nodales se descompone en fuerzas aplicadas directamente en nodos $[F_{pe}]$, fuerzas nodales equivalentes a las fuerzas distribuidas en el volumen del elemento $[F_{qe}]$ y en fuerzas nodales equivalentes a las fuerzas distribuidas en la superficie del elemento. Estos tres vectores vienen dados por (2.49)

La relación de equilibrio de una estructura se realiza de igual forma que la desarrollada para obtener la ecuación de equilibrio del elemento, sustituyendo los términos nodales del elemento por términos nodales de la estructura discretizado. Con ello, la ecuación matricial de equilibrio de la estructura es (2.50)

Particularizando la matriz de rigidez del elemento (2.55) para un elemento isoparamétrico cuadrangular se tiene

$$[K_e] = \int_{-1}^1 \int_{-1}^1 ([B_e]^T [D] [B_e]) t \cdot |J_e| \cdot d\xi \cdot d\eta \quad (2.67)$$

Utilizando como integración numérica la cuadratura de Gauss-Legendre, la matriz de rigidez del elemento es

$$[K_e] = \sum_i \sum_j (\phi(\xi_i, \eta_j)) W_i W_j \quad (2.68)$$

Donde W_i y W_j son los pesos de los puntos de gauss y $(\phi(\xi_i, \eta_j))$ es la función

$$\phi(\xi_i, \eta_j) = [B_e]^T [D] [B_e] \cdot t \cdot |J_e| \quad (2.69)$$

Evaluada en los puntos de Gauss ξ_i, η_j .

La bibliografía específica utilizada para el desarrollo de los apartados 2.4.1 y 2.4.2 son Oñate (1995), Vázquez et López (2001) y Zienkiewicz et Taylor (2004).

2.4.3. Campo de desplazamientos

Una vez obtenida la matriz completa de rigidez de la estructura $[K_0]$ y el vector de cargas equivalentes en nodos de la estructura $[F_0]$ se determina el comportamiento de la estructura, teniendo en cuenta las condiciones de contorno y las condiciones de carga anteriormente descritas (2.49).

Las condiciones de contorno determinan los grados de libertad en los que los desplazamientos son conocidos, ya sea por coacciones en los enlaces o por desplazamientos con valores dados. Esto permite reducir obtener un sistema de ecuaciones lineales con incógnitas en el vector de desplazamientos y en el vector de cargas (reacciones exteriores).

Para la resolución del sistema de ecuaciones se realiza una reducción de la matriz completa de rigidez de la estructura a una matriz del mismo orden que los grados de libertad no nulos de la estructura, llamada matriz de la estructura $[K]$. Para hallar esta matriz se desglosan los desplazamientos nodales en desplazamientos conocidos $\{\delta_c\}$ y desplazamientos desconocidos $\{\delta_d\}$. Las fuerzas nodales se desglosan en fuerzas desconocidas $\{F_d\}$ y fuerzas conocidas $\{F_c\}$. Teniendo esto en cuenta la ecuación de equilibrio (2.50) se puede poner de la forma

$$\begin{bmatrix} K & K_I \\ K_{II} & K_{III} \end{bmatrix} \begin{bmatrix} \delta_d \\ \delta_c \end{bmatrix} = \begin{bmatrix} F_c \\ F_d \end{bmatrix} \quad (2.70)$$

Despejando los términos desconocidos se obtienen dos sistemas de ecuaciones lineales resolubles.

$$[K]\{\delta_d\} = [K]^{-1}([F_c] - [K_I]\{\delta_c\}) \quad (2.72)$$

$$[F_d] = [K_{II}]\{\delta_d\} + [K_{III}]\{\delta_c\} \quad (2.73)$$

Resolviendo (2.66) se puede definir completamente el campo de desplazamientos de la estructura. Además, resolviendo (2.67) y restándole el vector de cargas exteriores en los nodos con coacciones exteriores se pueden obtener las reacciones en estas.

2.4.4. Campo de tensiones

Una vez obtenido los desplazamientos en los nodos, el campo de tensiones de cada elemento se puede obtener por medio de la ecuación (2.39).

$$\{\sigma_e\} = [D][B_e]\{\delta_e\}$$

Al obtenerse estas tensiones a partir de las derivadas de las funciones de forma, interpolación de los desplazamientos de los nodos, la aproximación es siempre de menor orden que la de éstas. Si las funciones de forma son de grado p , en general para los elementos usados, la aproximación de las tensiones será polinómica de grado $p - 1$ (Oñate, 1995).

Está demostrado que las tensiones obtenidas en el método de los elementos finitos pueden considerarse como un ajuste por mínimos cuadrados ponderados de la solución exacta. Debido a una propiedad de los puntos de la cuadratura de Gauss-Legendre (un polinomio de grado n y otro de grado $n - 1$, obtenido como ajuste por mínimos cuadrados del anterior, toman el mismo valor en los puntos de una cuadratura de orden n), se deduce del teorema de Herrmann (1972) que, si los gradientes de la aproximación se definen por un polinomio de grado $n - 1$, donde n es el grado del polinomio para la función de forma, las tensiones calculadas en estos puntos de la cuadratura de orden n es exacta (Zienkiewicz et Taylor, 2004).

Es decir, los puntos de la cuadratura de Gauss-Legendre tienen la propiedad de aproximar con mayor orden las tensiones y que, por consiguiente, las tensiones deben evaluarse en los puntos de la cuadratura y proceder a la extrapolación de los valores a los nodos.

Aunque la aproximación es mayor al extrapolar las tensiones obtenidas en los puntos de la cuadratura a los nodos, estas tensiones nodales obtenidas son discontinuas entre elementos por lo que habrá que realizar un alisado de las tensiones en cada uno de los nodos. Este alisado puede obtenerse como media de las tensiones nodales aportadas por cada elemento que concurre en el nodo.

La extrapolación de los valores de las tensiones de los puntos de la cuadratura a los nodos se puede realizar por varios métodos. Extrapolación y alisado global de tensiones, extrapolación y alisado local de tensiones o un método directo de la extrapolación local. Este último método, basado en la extrapolación de las tensiones por medio de funciones de forma, puede utilizarse para interpolar las tensiones de cualquier cuadratura $n \times n$ utilizando una función de interpolación adecuada. Además, se ha adaptado el método para poder obtener la extrapolación de las tensiones en elementos triangulares.

2.5. Estimación del error a posteriori

El error proporciona información sobre la calidad global de la solución y sobre la distribución del error en el dominio. Con la obtención de este error se pueden realizar un remallado concentrando los elementos allí donde la solución sea menos precisa con objeto de conseguir una solución con menor error.

Para un problema real el error exacto no se puede obtener salvo en casos especiales donde se conozca la solución exacta, pero existen procedimientos que nos permiten hacer una estimación del error para determinar la calidad de la solución obtenida.

Los errores en la solución de elementos finitos pueden ser de varios tipos: de discretización, error de redondeo, mala aproximación de la geometría... Estos últimos son más fáciles de controlar y de evitar a priori. Por consiguiente, para reducir el error global de la solución obtenida, habría que estudiar y reducir el error de discretización.

Se ha visto en el apartado 2.4.4 que la solución alisada σ_a del campo de tensiones es una mejor aproximación que la solución nodal discontinua dada por la ecuación 2.39 evaluada en éstos. Por consiguiente, puede considerarse un buen estimador del error en cada punto

$$e_\sigma = \sigma_a - \sigma \quad (2.73)$$

Lo más práctico es medir el error utilizando la norma energética, que proporciona una magnitud del error en términos escalares. Una medida directa es la norma de la energía, que puede asociarse a los errores en una misma cantidad.

$$\|e_\sigma\| = \left[\int_{\Omega} e_\sigma^T [D]^{-1} e_\sigma \cdot d\Omega \right]^{1/2} \quad (2.74)$$

Si se sustituye (2.73) en (2.75) se obtiene una expresión aproximada de la norma energética del error, siendo éste un buen estimador del error de cada elemento

$$\|e_\sigma\| = \left[\int_V [\sigma_a - \sigma]^T [D]^{-1} [\sigma_a - \sigma] \cdot dV \right]^{1/2} \quad (2.75)$$

Para el planteamiento del presente apartado se ha utilizado Oñate (1995), Restrepo (1998) y Zienkiewicz et Taylor (2004).

2.6. Tensión de Von Mises

Para poder obtener el coeficiente de seguridad de un material, dado por

$$n = \frac{\sigma_u}{\sigma} \quad (2.76)$$

Donde σ_u es la tensión última del material sometido a tracción, nos interesa reducir el campo de tensiones a uno simple que nos sirva de comparación.

Se define como tensión equivalente σ_{eq} la que existiría en una pieza de ese material sometido a tracción de tal forma que los dos estados tensionales tengan el mismo coeficiente de seguridad. Cuando el material alcanza su estado límite esta coincide con la tensión última del material.

Para la obtención de esta tensión equivalente se han propuesto varios criterios, entre los cuales la teoría de la energía de la distorsión de Von Mises es una de las más utilizadas para el caso de materiales dúctiles. Esta teoría expresa que el estado límite en un punto de un cuerpo en el que existe un estado tensional cualquiera comienza cuando la energía de distorsión por unidad de volumen en un entorno de dicho punto es igual a la energía de distorsión absorbida por unidad de volumen cuando el material alcanza la tensión límite en el ensayo a tracción.

Según esta teoría, para materiales dúctiles, las deformaciones plásticas no aparecerán hasta que se cumpla

$$\frac{1+\mu}{6E}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2] = \frac{1+\mu}{3E}\sigma_{eq}^2 \quad (2.77)$$

De donde se obtiene la tensión equivalente en función de las tensiones principales $\sigma_1, \sigma_2, \sigma_3$.

$$\sigma_{eq} = \sqrt{\frac{1}{2}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]} \quad (2.78)$$

Que se puede poner en función de las tensiones en cualquier punto como

$$\sigma_{eq} = \sqrt{[\sigma_x^2 + \sigma_y^2 + \sigma_z^2 - (\sigma_x\sigma_y + \sigma_y\sigma_z + \sigma_x\sigma_z) + 3(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2)]} \quad (2.79)$$

CAPÍTULO III

Descripción del programa

Contenido

3. Descripción del programa	29
3.1. Función principal “MEFBID”	30
3.2. Función de entrada de datos	31
3.3. Función “KESTRUCT”	31
3.3.1. Subfunción “KELEMEN”	31
3.4. Función “FESTRUCT”	32
3.4.1. Subfunción “FEQUIV”	33
3.5. Función “SOLUC”	34
3.6. Función “TALIS”	34
3.7. Función “ERROR”	35
3.7.1. Subfunción “TDISC”	36
3.8. Función “TVONM”	36
3.9. Subfunciones comunes	36
3.10. Variables del programa.....	38

3. Descripción del programa

En el presente capítulo se describirán las características principales del programa, incidiendo principalmente en la metodología desarrollada para la obtención de las matrices necesarias para el cálculo de la estructura.

El programa principal se divide en varias funciones, cada una de las cuales puede hacer uso de varias subfunciones. Cada una de las funciones que se han programado calcula una matriz necesaria para la resolución del problema.

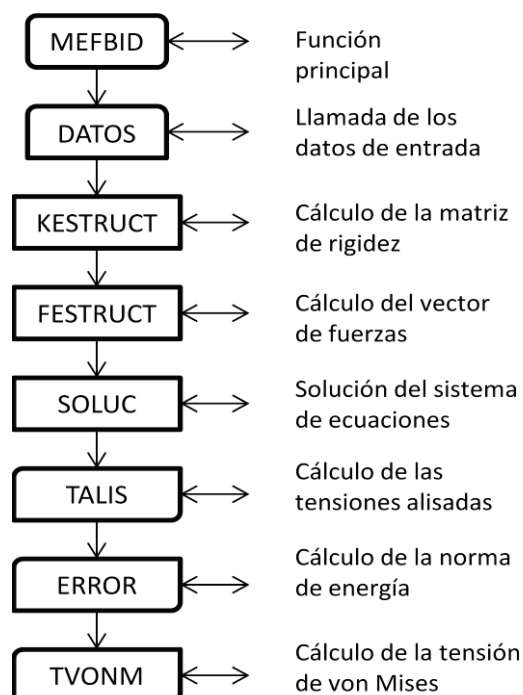
El programa se ha realizado con el fin de poder calcular estructuras de elasticidad plana con material elástico isótropo constante en toda la estructura, es decir, problemas en los que la estructura se encuentra en un estado tensional plano o en un estado de deformación plana. Para su resolución pueden utilizarse elementos isoparamétricos triangulares de tres nodos y cuadriláteros de cuatro nodos.

Las cargas utilizadas son cargas puntuales nodales y cargas repartidas en uno de los lados del elemento.

Por tratarse de un programa de cálculo de estructuras con elasticidad bidimensional mediante la metodología del MEF, se ha decidido denominar al programa “MEFBID”.

En la figura 3.1 se muestra el diagrama de flujo principal del programa “MEFBID”.

Figura3.1. Diagrama de flujo del programa “MEFBID”.



Como se puede observar el programa sigue el flujo normal de los programas de cálculo de estructuras mediante el MEF. Se podría dividir el programa en cuatro módulos diferentes. Un primer módulo de entrada de datos, un segundo módulo de cálculo de la solución del sistema de ecuaciones y un tercer módulo donde se analizan los resultados, calculando las tensiones en los nodos y la norma de energía de los elementos. Además también se podría incluir un cuarto módulo para mostrar por pantalla los datos calculados.

Como se dijo, el primer módulo de discretización y entrada de datos y el último módulo de representación gráfica se pueden ejecutar de una forma más sencilla y eficiente con un pre/pos procesador.

Aunque el contenido de las funciones programadas se realiza en los Anexos adjuntos, se realizara una breve exposición de las funciones con el fin de explicar la metodología seguida para su obtención, siguiendo orden de aparición en el programa. Para ello se expresaran como funciones las que aparecen en el código de la función principal y subfunciones a las que son llamadas en el interior de las funciones. Las subfunciones que aparecen en más de una función se describen en el apartado 3.9.

3.1. Función principal “MEFBID”

Es la función principal del programa y contiene todas las subfunciones necesarias para la resolución del problema. Sigue el diagrama de flujo de la figura 3.1 y es la función que se debe llamar en Matlab para la ejecución del programa.

Además de todas las subrutinas, la función “MEFBID” contiene un reloj con el que calcula los tiempos de ejecución del programa. Con ello se puede obtener los puntos donde el programa debe ser mejorado si su tiempo de ejecución es largo, teniendo en cuenta que la CPU del computador puede estar ejecutando otros programas. Por último, la función vuelca en un Excel los resultados obtenidos con el fin de poder estudiarlos.

Para el intercambio de datos entre las funciones se usaran variables locales, introduciéndolas en la llamada a las funciones a las que les sean necesarios. Para los datos de entrada al programa se usan variables globales, ya que estas son usadas en la mayoría de las subfunciones.

Para el cambio de datos de entrada basta con introducir el nombre de la función de datos en la línea 10 de la función “MEFBID”.

3.2. Función de entrada de datos

Es la función que contiene los datos necesarios para poder resolver el problema, además de calcular algunas constantes que facilitan la ejecución del programa.

Los datos de entrada que hay que indicarle al programa son: coordenadas de los nodos de la estructura, conectividad de los elementos, grados de libertad de cada nodo, tipo de elemento usado, número de puntos para la cuadratura de Gauss – Legendre, espesor de la estructura, grados de libertad con desplazamiento conocido, valor de los desplazamientos conocidos, tipo de problema, coeficiente de Poisson y modulo de Young del material. Además, para conocer el sistema de fuerzas al que queda sometida la estructura se necesita el tipo de fuerza, nodos entre los que se encuentra aplicada y el valor de la fuerza aplicada.

La función de datos además calcula del número de elementos y de nodos que se han introducido en la estructura.

3.3. Función “KESTRUCT”

Esta función calcula la matriz completa de rigidez de la estructura por medio de la ecuación 2.51. Para ello, para cada elemento calcula la matriz de rigidez del elemento y la ensambla en los grados de libertad correspondiente a los nodos del elemento en la matriz de rigidez de la estructura. El ensamblaje de la matriz de rigidez se realiza dependiendo del elemento usado.

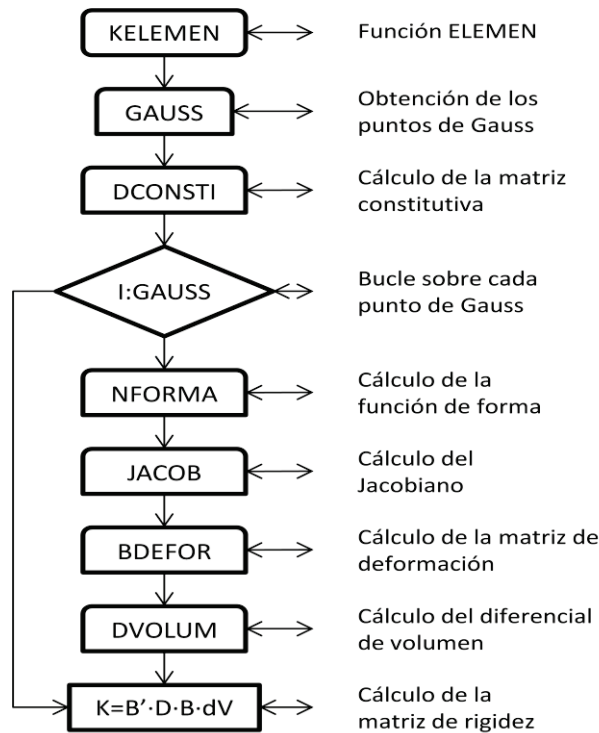
Para el cálculo de la matriz de rigidez de cada elemento se realiza la llamada a la función “KELEMEN”, dándole a esta las coordenadas de los nodos del elemento.

3.3.1. Subfunción “KELEMEN”

Es una Subfunción que calcula la matriz de rigidez de un elemento dado por las coordenadas de sus nodos, dato de entrada de la función. La matriz de rigidez del elemento se calcula por medio de las ecuaciones 2.54 y 2.55 para elementos triangulares y las ecuaciones 2.68 y 2.69 para elementos cuadrilaterales, es decir, el cálculo de la matriz de rigidez del elemento depende del elemento usado para la discretización.

En la figura 3.2 se puede observar el flujo de la Subfunción para obtener la matriz de rigidez de un elemento.

Figura 3.2. Diagrama de flujo de la subfunción “KELEMEN”



3.4. Función “FESTRUCT”

Función que ensambla el vector de fuerzas equivalentes aplicada en los nodos de la estructura. Para cada una de las fuerzas aplicadas en la estructura, obtiene de los datos de entrada los nodos donde se aplica y el valor de cada fuerza y calcula el vector de carga equivalente en nodos como suma de los vectores de carga de cada fuerza. Para el cálculo de este vector se realiza la llamada a la subfunción “FEQUIV”.

Los datos necesarios para el cálculo de la fuerza equivalente varían dependiendo si la carga aplicada es puntual aplicada en los nodos o si la carga es repartida en uno de los lados del elemento. Para la carga puntual se obtiene el nodo donde la carga se aplica y la componente x e y de la carga, por lo que la variable “*nodofi*” será un vector de longitud 1 y la variable “*fuerzai*” será un vector de longitud 2. Para la carga distribuida se obtienen los nodos entre los que se aplica y las componentes normales y tangenciales en cada nodo, por ello la variable “*nodofi*” será un vector de longitud 2 y la variable “*fuerzai*” será una matriz de tamaño 2 x 2.

3.4.1. Subfunción “FEQUIV”

Esta subfunción calcula el vector fuerza equivalente en nodos de cada una de las fuerzas aplicadas a la estructura. Para ello en primer lugar calcula la carga equivalente en nodos y después la ensambla en los grados de libertad del nodo en un vector ampliado a los grados de libertad de la estructura relleno de seros.

Como se puede observar en la ecuación 2.49, para una carga puntual, la fuerza equivalente es igual a la fuerza aplicada, por lo que basta con poner estas fuerzas en los grados de libertad correspondientes al nodo donde se aplica.

$$[F_{pe}] = [P_e] = \begin{bmatrix} P_x \\ P_y \end{bmatrix}$$

Para cargas distribuidas sobre un lado, estas lo hacen en direcciones normales y tangenciales al lado donde se aplican,

$$[F_{pe}] = \oint_l \left([N_e]^T \begin{bmatrix} p_x \\ p_y \end{bmatrix} \right) \cdot ds = [F_{pe}] = \oint_l \left([N_e]^T \begin{bmatrix} \tau \cdot dx - \sigma \cdot dy \\ \sigma \cdot dx + \tau \cdot dy \end{bmatrix} \right)$$

Si, al estar la carga distribuida en un lado del elemento, se realiza una integración unidimensional (se supone en el eje ξ en coordenadas naturales) de orden igual al número de nodos que existen en el lado cargado,

$$\int_{-1}^{+1} \left([N_e] \begin{bmatrix} \tau \cdot J_{11} - \sigma \cdot J_{12} \\ \sigma \cdot J_{11} + \tau \cdot J_{12} \end{bmatrix} \cdot d\xi \right) = \sum_{p=1}^n \left[[N_e] \begin{bmatrix} \tau \cdot J_{11} - \sigma \cdot J_{12} \\ \sigma \cdot J_{11} + \tau \cdot J_{12} \end{bmatrix} \right]_p W_p$$

Donde $[N_e]$ es la función de forma del elemento unidimensional, J es la matriz del jacobiano del elemento, n el número de puntos de la cuadratura unidimensional y σ y τ las tensiones normales y tangenciales en los puntos de la cuadratura. Estos valores se obtienen en función de los valores nodales dados,

$$\begin{Bmatrix} \sigma \\ \tau \end{Bmatrix} = \sum_{i=1}^n [N_e] \begin{Bmatrix} \sigma_i \\ \tau_i \end{Bmatrix}$$

Los datos de entrada a esta subfunción son los nodos donde se aplica y el valor de una fuerza.

3.5. Función “*SOLUC*”

Esta función resuelve el sistema de ecuaciones 2.70, obteniendo el vector de desplazamientos nodales desconocidos y el vector de fuerzas nodales desconocidas. Para ello resuelve dos sistemas de ecuaciones, las ecuaciones 2.71 y 2.72.

Para resolver este sistema primero es necesario un reordenamiento de la matriz de rigidez y de fuerza calculada así como introducir en la matriz de desplazamiento los valores dados. Para ello, se obtienen de los datos de entrada los grados de libertad con desplazamiento nulo, los grados de libertad donde los desplazamientos son desconocidos y los grados de libertad donde el desplazamiento es no nulo.

Con estos vectores que indican los grados de libertad, se pueden resolver la ecuación 2.71 para obtener los desplazamientos nodales desconocidos y la ecuación 2.72 para obtener las reacciones en los apoyos.

3.6. Función “*TALIS*”

Esta función calcula las tensiones nodales alisadas de la estructura. Para ello se procederá como se apuntó en el apartado 2.4.

En primer lugar se calculan las tensiones en los puntos de la cuadratura, siendo esta calculada en cada punto como

$$\{\sigma_i\} = [D][B_p]\{\delta_e\}$$

Siendo $\{\sigma_i\}$ la tensión en un punto de la cuadratura, $[B_p]$ la matriz de deformación particularizada en el punto de la cuadratura donde se quiere calcular la tensión y $\{\delta_e\}$ el vector de desplazamientos nodales del elemento. Una vez calculado el vector tensión en cada uno de los puntos de la cuadratura se realiza la extrapolación de las tensiones por el método directo. Las tensiones en los nodos del elemento se pueden poner como

$$\{\sigma\} = \sum_{i=1}^p [N_i]\{\sigma_i\}$$

Donde p es el número de puntos de la cuadratura y $[N_i]$ es una función de forma que extrapola el valor de las tensiones de los puntos de la cuadratura a los nodos del elemento. Esta función de forma depende del número de puntos de la

cuadratura, siendo para cuadraturas $n \times n$ igual a la función de forma del elemento lagrangiano con igual número de nodos que puntos de la cuadratura y realizando el cambio de variable

$$s = \frac{\xi}{p(\xi)}$$

$$t = \frac{\eta}{p(\eta)}$$

Donde p indica las coordenadas de los puntos de la cuadratura, por lo que basta con usar las funciones de forma con coordenadas $1/p$, teniendo en cuenta que la numeración de los puntos de la cuadratura debe ser la misma que la de los nodos del elemento. Para cuadraturas de un solo punto, la función de forma será la matriz identidad de orden igual al número de nodos del elemento.

Para elementos triangulares de tres nodos, las funciones de forma son la matriz identidad de orden tres para un solo punto de cuadratura y la función de forma del propio elemento para cuadraturas de tres puntos. En éste último caso las coordenadas de los nodos son

$$\xi = [-1/3 \quad 5/3 \quad -1/6]$$

$$\eta = [-1/3 \quad -1/6 \quad 5/3]$$

Como se puede observar, en todos los casos las funciones de forma toman valor 1 en el punto de la cuadratura y 0 en los restantes.

Por último, debido a la no continuidad de las tensiones nodales entre elementos adyacentes, se realiza un alisado de las tensiones en los nodos, siendo el valor de las tensiones la media aritmética de las tensiones de un nodo como perteneciente a todos los elementos que en él concurren.

3.7. Función “ERROR”.

Esta función calcula la expresión aproximada de la norma energética del error, estimación del error de discretización, dada por la ecuación 2.75 para cada uno de los elementos de la estructura.

Para ello se obtiene las tensiones alisadas en los nodos de cada elemento y se calcula la tensión discreta en cada nodo. El valor de estas dos matrices es interpolado a los puntos de la cuadratura con el fin de calcular la integral mediante cuadratura de Gauss-Legendre.

3.7.1. Subfunción “*TDISC*”.

Esta subfunción calcula las tensiones discretas en los nodos, aplicando la ecuación 2.39 directamente en los nodos del elemento. Para ello calcula la matriz de deformación en los nodos del elemento y aplica la ecuación anterior con los datos de un elemento dado.

3.8. Función “*TVONM*”.

Esta función calcula la tensión de von Mises en cada uno de los nodos de la estructura a partir de las tensiones alisadas calculadas en estos. El cálculo de la tensión de von Mises se realiza aplicando la ecuación 2.79. Para ello basta aplicar la ecuación anterior a cada uno de los nodos de la estructura.

3.9. Subfunciones comunes

En este apartado se describen las subfunciones que se utilizan en varias funciones.

- Subfunción “*DCONSTI*”

Esta subfunción calcula la matriz constitutiva de la estructura. Esta matriz constitutiva se obtiene dependiendo del estado tensional en el que se encuentre la estructura. Si se encuentra en estado tensional plano se calcula la matriz por medio de la ecuación 2.14 y si se encuentra en un estado de deformación plano se obtiene con la ecuación 2.13.

Esta matriz será la misma para todos los elementos de la estructura ya que sólo puede estar formado por un material con características constantes en todo su volumen.

- Subfunción “*GAUSS*”.

Esta subfunción devuelve las coordenadas de los puntos para la cuadratura de Gauss – Legendre así como los pesos de los mismos. Para elementos triangulares se permiten cuadraturas de 1 y 3 puntos y para elementos

cuadrangulares cuadraturas de 1×1 y 2×2 puntos. Para cargas repartidas sobre un lado del elemento se usan cuadraturas de dos puntos.

- Subfunción “*NFORMA*”.

Esta subfunción calcula la matriz de forma de un elemento y la derivada de esta en coordenadas isoparamétricas para un punto dado. La matriz de forma es

$$[N_1 \quad N_2 \quad N_3 \dots]$$

Donde N_i viene dado por la ecuación 2.28 para elementos triangulares y por la ecuación 2.64 para elementos cuadrangulares. La derivada de la función de forma es

$$\begin{bmatrix} \frac{N_1}{d\xi} & \frac{N_2}{d\xi} & \frac{N_3}{d\xi} \dots \\ \frac{N_1}{d\eta} & \frac{N_2}{d\eta} & \frac{N_3}{d\eta} \dots \end{bmatrix}$$

Esta derivada se usa, como se indica más adelante, para el cálculo de la matriz de deformación del elemento.

- Subfunción “*JACOB*”.

Esta subfunción calcula la matriz jacobiano del elemento, dada esta por la ecuación 2.33. Además calcula el determinante y la matriz inversa del jacobiano.

- Subfunción “*DVOLUM*”

Calcula el diferencial de volumen para cada punto de la cuadratura de un elemento. Este diferencial se obtiene de las ecuaciones 2.54 y 2.55, siendo para elementos triangulares

$$dV = t \cdot |J_e| \cdot W_i$$

Y para elementos cuadrangulares, teniendo en cuenta el doble sumatorio necesario para el cálculo de la cuadratura, es

$$dV = t \cdot |J_e| \cdot W_i \cdot W_j$$

- Subfunción “*BDEFOR*”.

Calcula la función de deformación en un punto, dada la derivada de la matriz de forma y la matriz inversa del jacobiano. Para la obtención de esta matriz se realiza la multiplicación matricial

$$B_e = I \cdot [J_e]_{\text{amp}}^{-1} \cdot [\partial][N_e]_{\text{amp}}$$

Siendo las matrices

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$[J_e]_{\text{amp}}^{-1} = \begin{bmatrix} [J_e]^{-1} & [0] \\ [0] & [J_e]^{-1} \end{bmatrix}$$

$$[\partial][N_e]_{\text{amp}} = \begin{bmatrix} [\partial][N_e]_i & [0] \\ [0] & [\partial][N_e]_i \dots \end{bmatrix}$$

Donde $[0]$ es una matriz de ceros de 2×1 y $[\partial][N_e]_i$ la derivada de la matriz de forma i del punto.

3.10. Variables del programa

En el presente apartado se muestra un listado con las variables utilizadas en el programa “*MEFBID*”. Las variables de entrada al programa son los datos necesarios que se tienen que introducir para que el programa calcule la estructura.

- Variables de entrada al programa.
 - *TITULO* → variable que contiene la cadena de caracteres que identifican la función que contiene los datos del problema.
 - *coor* → matriz que contiene las coordenadas, x en la primera columna y coordenadas y en la segunda, de los nodos de la estructura.
 - *conectividad* → matriz que contiene los nodos que componen cada elemento de la estructura.
 - *dim* → constante que indica los grados de libertad de cada nodo de la estructura.
 - *elemento* → indica el tipo de elemento en el que se discretiza la estructura. Las posibilidades son:
 - 1. Elemento triangular de tres nodos
 - 2. Elemento cuadrilateral de cuatro nodos

- *nodos* → indica el número de nodos de la estructura.
 - *elementos* → indica el número de elementos en los que se ha discretizado la estructura.
 - *gauss* → número de puntos de gauss para la cuadratura de Gauss – Legendre. Las posibilidades son:
 - 1. Para elementos triangulares y cuadrilaterales
 - 2. Para elementos cuadrilaterales
 - 3. Para elementos triangulares
 - *t* → espesor de la estructura bidimensional.
 - *tipof* → vector que contiene el tipo de cada una de las fuerzas que están aplicadas en la estructura. Los tipos de fuerzas son
 - 1. Para cargas puntuales aplicadas en los nudos
 - 2. Para cargas repartidas entre dos nudos
 - *nodof* → matriz que contiene los nodos en los que se aplica cada fuerza. Contendrá dos valores en cada fila para cada tipo de fuerza, siendo 0 el valor de la segunda columna para cargas puntuales.
 - *fuerza* → matriz que contiene el valor de las fuerzas aplicadas. Para fuerzas puntuales tendrá el valor en x en la primera columna y el valor en y en la segunda. Para cargas repartidas el valor de la fuerza normal a la recta que une los nodos (carga en nodo 1 en la primera columna y carga en nodo 2 en la segunda) y el valor de la fuerza tangencial a la recta que une los nodos (carga en nodo 1 en la tercera columna y carga en nodo 4 en la cuarta columna).
 - *xcon* → vector que contiene los grados de libertad en los que se conoce su desplazamiento.
 - *Nxcon* → vector que contiene el valor de los desplazamientos conocidos
 - *consti* → variable que indica el tipo de problema que se encuentra la estructura. Puede tomar los valores
 - 1. Para tensión plana
 - 2. Para deformación plana
 - *poiss* → indica el coeficiente de Poisson del material usado en la estructura.
 - *Eng* → indica el modulo de Young del material usado en la estructura.
- Variables del programa
 - *tiempo* → vector que contiene los tiempos de ejecución del programa.
 - *K* → matriz completa de rigidez de la estructura.
 - *F* → vector completo de fuerzas de la estructura.

- $D \rightarrow$ matriz que contiene los desplazamientos nodales de la estructura. Contiene los desplazamientos en x en la primera columna y los desplazamientos en y en la segunda.
- $RD \rightarrow$ matriz que contiene las reacciones en los apoyos de la estructura. Contiene las reacciones en x en la primera columna y en y en la segunda.
- $T \rightarrow$ matriz que contiene las tensiones nodales alisadas de la estructura. Tensiones en el eje x en la primera columna, tensiones en y en la segunda, tensiones en z , si las hubiese, en la tercera y tensiones tangenciales en la cuarta.
- $Err \rightarrow$ vector que contiene la norma energética del error de cada elemento.
- $TVM \rightarrow$ Vector de tensiones de von Mises de cada nodo.
- $elem \rightarrow$ vector que contiene los nodos que corresponden a un elemento.
- $coorel \rightarrow$ matriz que contiene las coordenadas de los nodos de un elemento.
- $gradol \rightarrow$ vector que contiene los grados de libertad globales que pertenecen a los nodos de un elemento.
- $Kel \rightarrow$ matriz de rigidez del elemento.
- $POS1 \rightarrow$ coordenadas ξ de los puntos de la cuadratura.
- $POS2 \rightarrow$ coordenadas η de los puntos de la cuadratura.
- $PES \rightarrow$ pesos de los puntos de la cuadratura.
- $D \rightarrow$ matriz constitutiva del elemento.
- $N \rightarrow$ función de forma del elemento en coordenadas isoperimétricas.
- $DN \rightarrow$ derivada de la función de forma del elemento.
- $Jacob \rightarrow$ matriz del jacobiano del elemento.
- $Detjac \rightarrow$ determinante del jacobiano del elemento.
- $Invjac \rightarrow$ matriz inversa del jacobiano del elemento.
- $Be \rightarrow$ matriz de deformación del elemento
- $Dvolum \rightarrow$ diferencial de volumen del elemento
- $Numf \rightarrow$ número de fuerzas al que está sometida la estructura.
- $Nodofi \rightarrow$ vector que indica el/los nodos donde se aplica una fuerza.
- $Fuerzai \rightarrow$ matriz que indica el valor de una fuerza.
- $FO \rightarrow$ vector de fuerza equivalente en nodos de una fuerza dada.
- $Dvolu \rightarrow$ diferencial de volumen.
- $PG \rightarrow$ vector que contiene los valores de las fuerzas aplicados en los puntos de la cuadratura.
- $coorF \rightarrow$ coordenadas de los nodos entre los que se aplica la fuerza.
- $Jacob \rightarrow$ jacobiano del elemento.
- $PX \rightarrow$ valor en el eje x de la carga distribuida.
- $PY \rightarrow$ valor en el eje y de la carga distribuida.

- *Gdlnul* → vector de grados de libertad donde el desplazamiento es nulo.
 - *GdlDes* → vector de grados de libertad donde el desplazamiento es desconocido
 - *Gdlnonul* → vector de grados de libertad donde el desplazamiento no es nulo
 - *DDes* → vector que contiene los desplazamientos nodales desconocidos.
 - *FD* → vector que contiene las fuerzas nodales desconocidas.
 - *Nten* → número de tensiones en cada nodo.
 - *Delem* → matriz de desplazamientos nodales de un elemento.
 - *Te* → matriz de tensiones en los puntos de la cuadratura de un elemento.
 - *N1* → matriz de la funciones de forma para la extrapolación.
 - *POS12* → coordenadas ξ de los nodos para la extrapolación.
 - *POS22* → coordenadas η de los nodos para la extrapolación.
 - *conT* → vector que contiene el número de elementos que concurren en cada nodo.
 - *Ta* → Matriz que contiene las tensiones alisadas en los nodos de un elemento.
 - *Td* → matriz que contiene las tensiones discretas en los nodos de un elemento.
- Contadores de la función
 - *contel* → contador sobre cada elemento de la discretización.
 - *contdim* → contador sobre cada nodo del elemento.
 - *contg* → contador de los puntos de la cuadratura
 - *contg1* → contador de los puntos de la cuadratura para elementos cuadrilaterales.
 - *contf* → contador sobre cada una de las fuerzas aplicadas.
 - *contdes* → contador de los desplazamientos nodales desconocidos.
 - *contn* → contador de los nodos de la estructura

CAPÍTULO IV

Resultados y análisis

Contenido

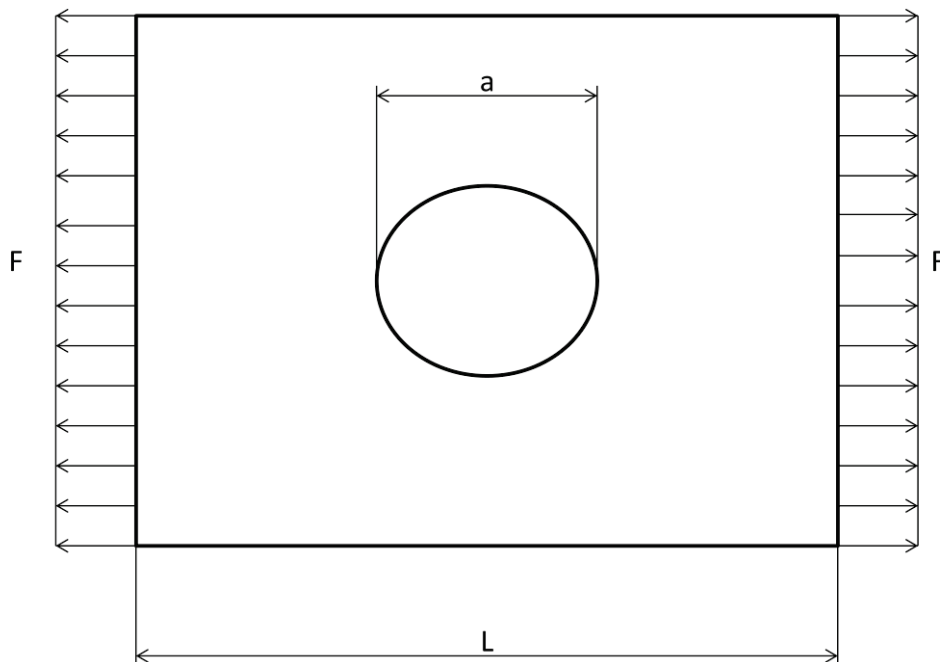
4. Resultados y análisis	43
4.1. Resultados del programa <i>MEFBID</i>	44
4.2. Resultados analíticos del problema	62
4.3. Resultados de otros programas.....	62
4.4. Análisis de los resultados	64
4.4.1. Análisis de los tiempos de cálculo.....	65
4.4.2. Análisis de los elementos utilizados	66
4.4.3. Análisis de la norma de energía	67
4.4.4. Análisis de la convergencia	68
4.5. Otro resultado	68

4. Resultados y análisis

Para poder valorar el modelo programado, es necesario evaluar unos resultados sobre un problema tipo del que se conozca la solución exacta del problema exacto. También es necesario comparar los resultados obtenidos con *MEFBID* con los que se obtienen con otro software comercial con el fin de comparar la solución con una discretización similar. Para ello se estudiara el siguiente problema:

Supóngase una placa plana cuadrada con un orificio central, figura 4.1, sometida a tracción en estado tensional plano.

Figura 4.1. Problema para el análisis de los resultados del programa. Placa plana cuadrada con orificio central sometida a tracción.



El material utilizado es acero, con $E = 210 \cdot 10^3 \text{ N/mm}^2$ y $\nu = 0.3$. Las dimensiones y el sistema de cargas al que está sometido se muestran en la figura, siendo:

$$L = 2000 \text{ mm}$$

$$a = 400 \text{ mm}$$

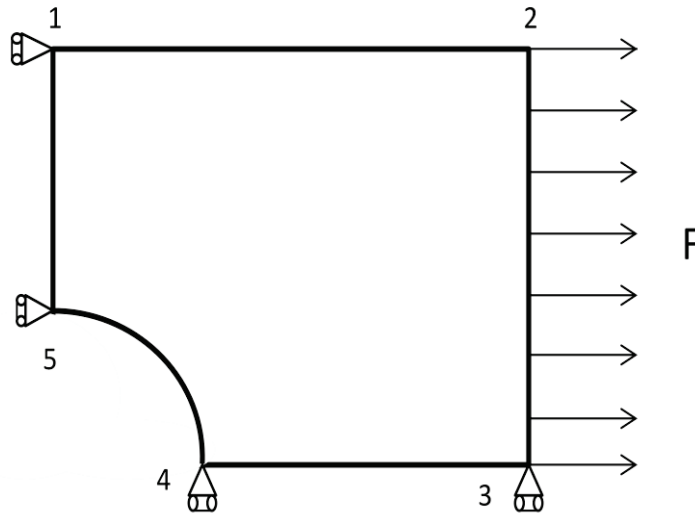
$$F = 500 \text{ N/mm}$$

$$e = 1 \text{ mm, espesor de la placa}$$

Como se observa, la placa tiene simetría con respecto a los dos ejes centrales, por lo que bastara con estudiar un cuarto de la placa, teniendo en cuenta

las restricciones correspondientes. Con ello, el modelo a estudiar se muestra en la figura 4.2, donde se indican los nodos que serán objeto de estudio.

Figura 4.2. Simplificación del problema, una vez aplicada la simetría y las restricciones.



La solución analítica a este problema fue obtenida por G. Kirsch y puede hallarse en una amplia bibliografía. El campo de tensiones en coordenadas polares para el problema es [Timoshenko et Goodier]:

$$\sigma_r = \frac{S}{2} \left(1 - \frac{a^2}{r^2} \right) + \frac{S}{2} \left(1 + \frac{3a^4}{r^4} - \frac{4a^2}{r^2} \right) \cos(2\theta)$$

$$\sigma_\theta = \frac{S}{2} \left(1 + \frac{a^2}{r^2} \right) - \frac{S}{2} \left(1 + \frac{3a^4}{r^4} \right) \cos(2\theta)$$

$$\tau_{r\theta} = -\frac{S}{2} \left(1 - \frac{3a^4}{r^4} + \frac{2a^2}{r^2} \right) \sin(2\theta)$$

Siendo θ y r las coordenadas polares del punto y S la tensión en la sección transversal sin orificio.

4.1. Resultados del programa *MEFBID*

Para poder realizar un análisis correcto del programa se realizan diferentes mallados de la estructura con los dos elementos que el programa puede analizar. Se realizaran los siguientes cálculos de la estructura:

– Elementos triangulares:

1. 16 nodos, 20 elementos y una cuadratura de 1 punto de Gauss.

2. 51 nodos, 80 elementos y una cuadratura de 1 punto de Gauss.
 3. 51 nodos, 80 elementos y una cuadratura de 3 puntos de Gauss.
 4. 181 nodos, 320 elementos y una cuadratura de 1 punto de Gauss.
 5. 681 nodos, 1280 elementos y una cuadratura de 1 punto de Gauss.
- Elementos cuadrilaterales.
6. 28 nodos, 18 elementos y una cuadratura de 1 x 1 puntos de Gauss.
 7. 28 nodos, 18 elementos y una cuadratura de 2 x 2 puntos de Gauss.
 8. 28 nodos, 18 elementos y una cuadratura de 2 x 2 puntos de Gauss para integrales y 1 x 1 puntos para tensiones.
 9. 34 nodos, 23 elementos y una cuadratura de 1 x 1 puntos de Gauss.
 10. 34 nodos, 23 elementos y una cuadratura de 2 x 2 puntos de Gauss.
 11. 34 nodos, 23 elementos y una cuadratura de 2 x 2 puntos de Gauss para integrales y 1 x 1 puntos para tensiones.
 12. 111 nodos, 90 elementos y una cuadratura de 2 x 2 puntos de Gauss.

Para los casos de elementos triangulares se ha utilizado el toolbox de Matlab `pdetool` para realizar el mallado de la estructura. Para los casos 6, 7 y 8 de elementos cuadriláteros se ha realizado un mallado manual y para el resto de los casos el mallado se ha importado desde COMSOL.

A continuación se muestran los resultados obtenidos para los diferentes casos. El programa calcula los valores de desplazamiento, reacciones, tensiones y la tensión de von Mises en todos los nodos de la estructura pero aquí se expondrán los valores obtenidos para los puntos más característicos de la estructura mostrados en la figura 4.2. Se muestran también los tiempos de cálculo de alguno de los casos y las gráficas obtenidas para cada uno de ellos.

El código utilizado para la realización de las gráficas no se incluye en el texto, ya que no es muy útil su realización en el entorno de Matlab, siendo el usuario del programa el responsable de mostrar los resultados de una forma clara.

- Elementos triangulares:
 - **Caso 1:** 16 nodos, 20 elementos y Gauss igual a 1.

En la tabla 4.1 y las figuras 4.3 - 4.8 se muestran los resultados obtenidos para este caso.

Tabla 4.1. Resultados en los nodos para el caso 1 con mallado de 20 elementos triangulares. Desplazamientos en mm y tensiones en N/mm².

Nodo	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-1,9199	507,0666	20,7169	53,2880	505,5291
2	2,1666	-0,0463	530,3448	3,2067	-17,6024	529,6271
3	4,3111	0,0000	450,8485	175,3672	-98,1138	428,7574
4	3,4758	0,0000	244,8132	-251,2983	-139,4840	492,9227
5	0,0000	-1,4769	1149,8085	98,9620	-62,1546	1108,8982

Figura 4.3. Gráfica de los elementos y su deformada(x10) del caso 1 de elementos triangulares. Cotas en mm.

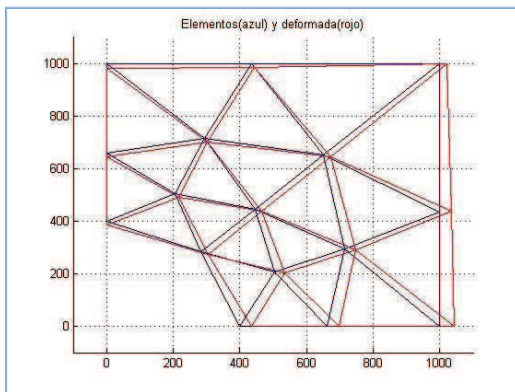


Figura 4.4. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 1 de elementos triangulares. Cotas en mm y tensiones en N/mm².

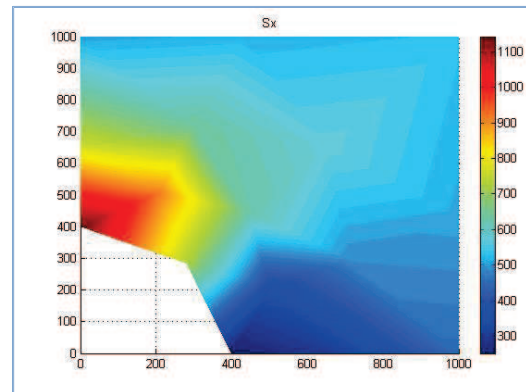


Figura 4.5. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 1 de elementos triangulares. Cotas en mm y tensiones en N/mm².

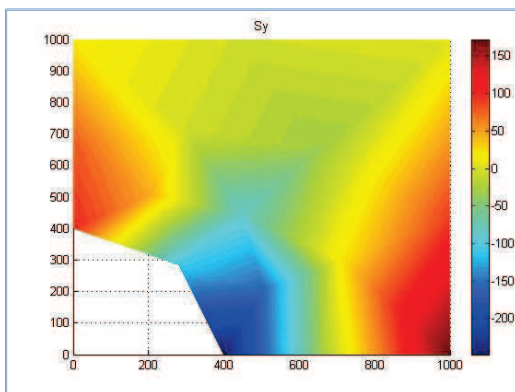


Figura 4.6. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 1 de elementos triangulares. Cotas en mm y tensiones en N/mm².

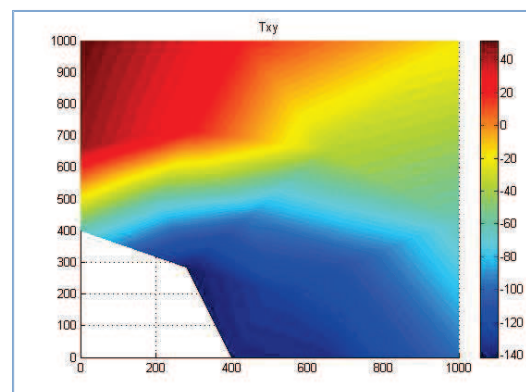


Figura 4.7. Grafica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 1 de elementos triangulares. Cotas en mm y tensiones en N/mm².

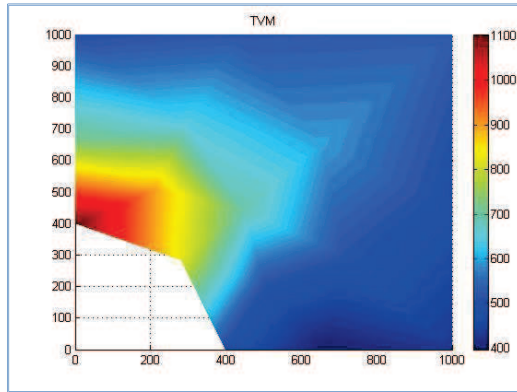
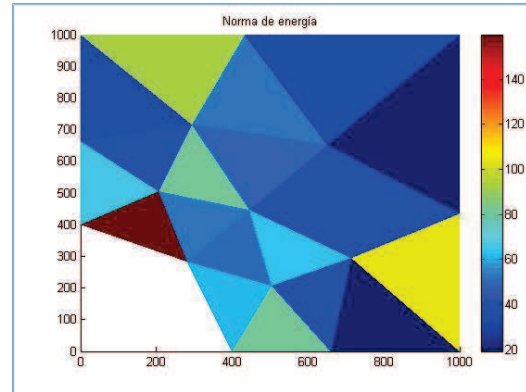


Figura 4.8. Grafica de la norma de energía de los elementos para el caso 1 de elementos triangulares.



– **Caso 2:** 51 nodos, 80 elementos y Gauss igual a 1.

En la tabla 4.2 se muestran los valores obtenidos en los nodos para este caso.

Tabla 4.2. Resultados en los nodos para el caso 2 con mallado de 80 elementos triangulares. Desplazamientos en mm y tensiones en N/mm².

Nodo	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,5923	326,7760	24,0836	46,2311	325,4299
2	1,8780	0,3044	509,3666	2,1617	-4,7259	508,3551
3	5,1149	0,0000	462,4936	412,5112	-62,6733	452,8420
4	4,3759	0,0000	61,9926	-676,6375	-27,2192	711,2317
5	0,0000	-2,1744	1726,4441	134,9328	-130,6316	1678,4088

Para estudiar los tiempos de cálculo de las distintas funciones del programa, en la tabla 4.3 se muestran estos.

Tabla 4.3. Tiempos de cálculo de las funciones del programa para el caso 2.

Cálculo	Matriz K	Vector F	Desplazamientos	Tensiones	Error	Ten von Mises
Tiempo (s)	0,0310	0,0000	0,0020	0,0160	0,0700	0,0010

Por último, en las figuras 4.9 - 4.14 se muestran las gráficas obtenidas para el caso 2:

Figura 4.9. Gráfica de los elementos y su deformada(x10) del caso 2 de elementos triangulares. Cotas en mm.

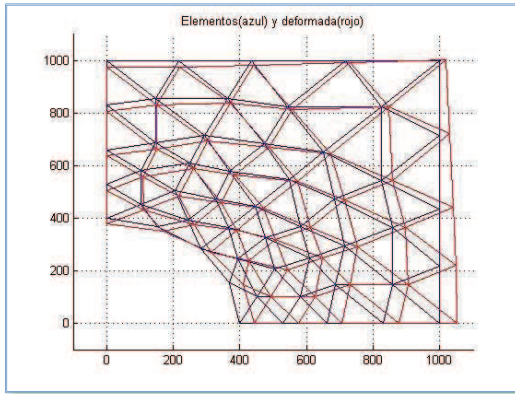


Figura 4.10. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 2 de elementos triangulares. Cotas en mm y tensiones en N/mm².

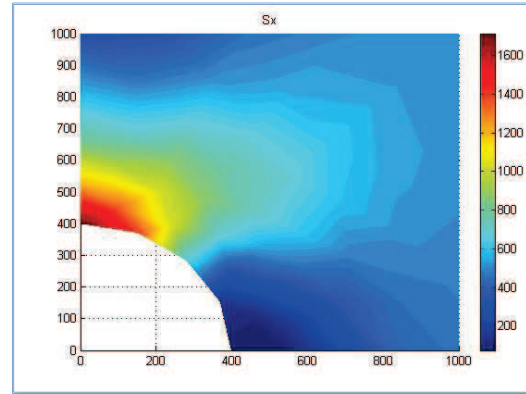


Figura 4.11. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 2 de elementos triangulares. Cotas en mm y tensiones en N/mm².

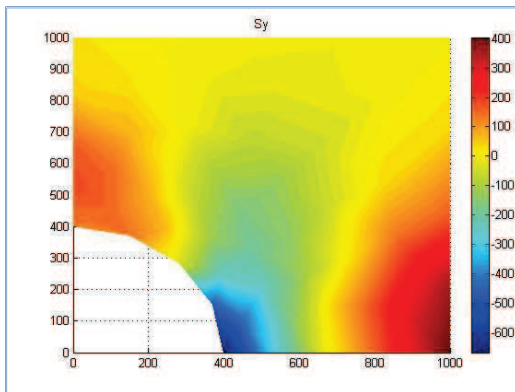


Figura 4.12. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 2 de elementos triangulares. Cotas en mm y tensiones en N/mm².

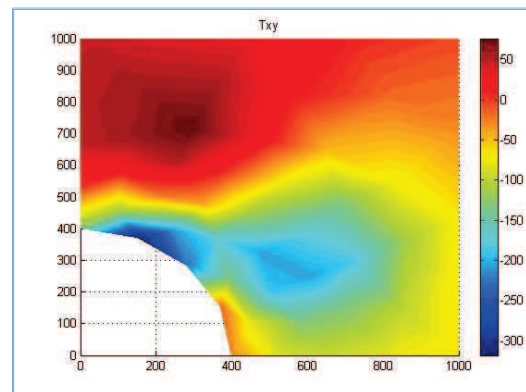


Figura 4.13. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 2 de elementos triangulares. Cotas en mm y tensiones en N/mm².

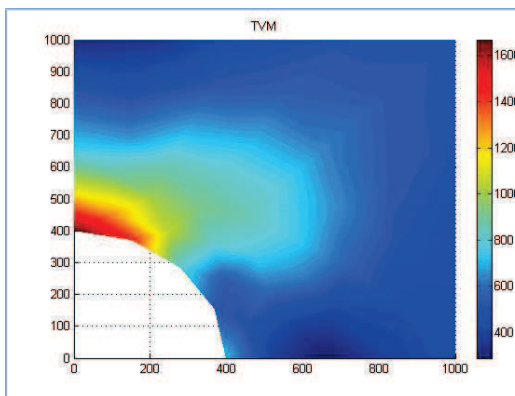
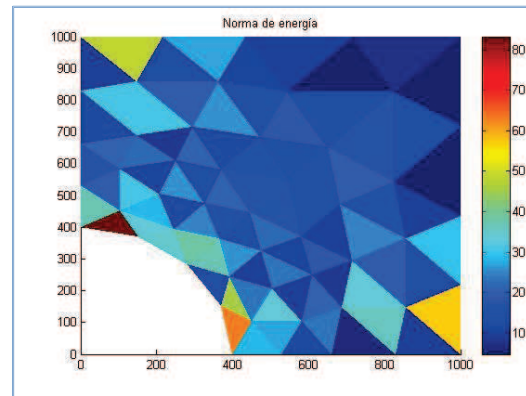


Figura 4.14. Gráfica de la norma de energía de los elementos para el caso 2 de elementos triangulares.



- **Caso 3:** 51 nodos, 80 elementos y Gauss igual a 3.

En la tabla 4.4 se muestran los valores obtenidos en los nodos para el caso 3.

Tabla 4.4. Resultados en los nodos para el caso 3 con mallado de 80 elementos triangulares. Desplazamientos en mm y tensiones en N/mm²

Nodo	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,5923	326,7760	24,0836	46,2311	325,4299
2	1,8780	0,3044	509,3666	2,1617	-4,7259	508,3551
3	5,1149	0,0000	462,4936	412,5112	-62,6733	452,8420
4	4,3759	0,0000	61,9926	-676,6375	-27,2192	711,2317
5	0,0000	-2,1744	1726,4441	134,9328	-130,6316	1678,4088

Para estudiar los tiempos de cálculo de las distintas funciones del programa, en la tabla 4.5 se muestran estos.

Tabla 4.5. Tiempos de cálculo de las funciones del programa para el caso 3.

Cálculo	Matriz K	Vector F	Desplazamientos	Tensiones	Error	Ten von Mises
Tiempo (s)	0,0600	0,0010	0,0010	0,0480	0,0980	0,0000

- **Caso 4:** 181 nodos, 320 elementos y Gauss igual a 1.

En la tabla 4.6 y las figuras 4.15 – 4.20 se muestran los resultados obtenidos para el caso 4.

Tabla 4.6. Resultados en los nodos para el caso 4 con mallado de 320 elementos triangulares. Desplazamientos en mm y tensiones N/mm²

Nodo	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,8659	188,1199	11,0401	23,4243	187,2971
2	1,7530	0,4499	500,2652	-0,0282	0,4918	500,2800
3	5,4291	0,0000	484,7062	577,2531	-29,0084	539,3399
4	4,7294	0,0000	6,7965	-988,1151	25,0455	992,4793
5	0,0000	-2,4842	2089,1261	93,2372	-117,8868	2054,2757

Figura 4.15. Gráfica de los elementos y su deformada(x10) del caso 4 de elementos triangulares. Cotas en mm.

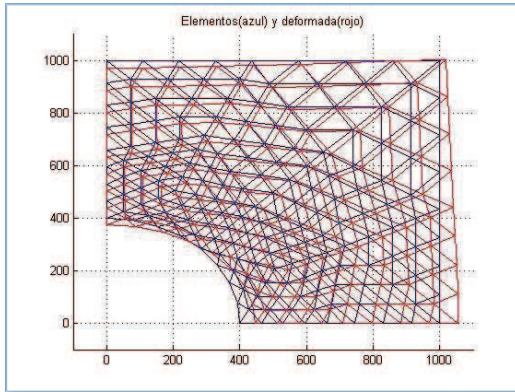


Figura 4.16. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 4 de elementos triangulares. Cotas en mm y tensiones en N/mm².

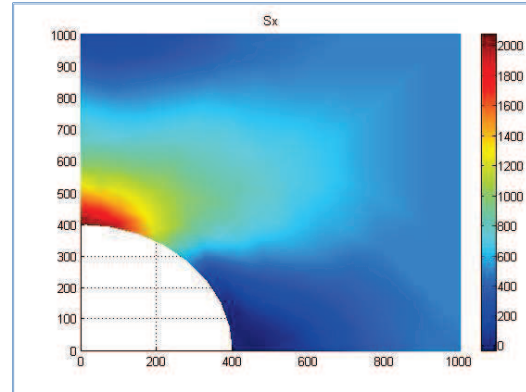


Figura 4.17. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 4 de elementos triangulares. Cotas en mm y tensiones en N/mm².

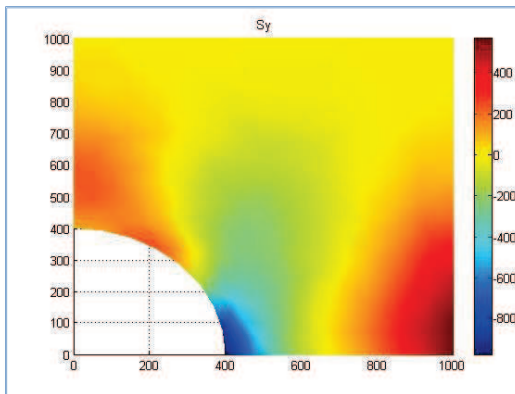


Figura 4.18. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 4 de elementos triangulares. Cotas en mm y tensiones en N/mm².

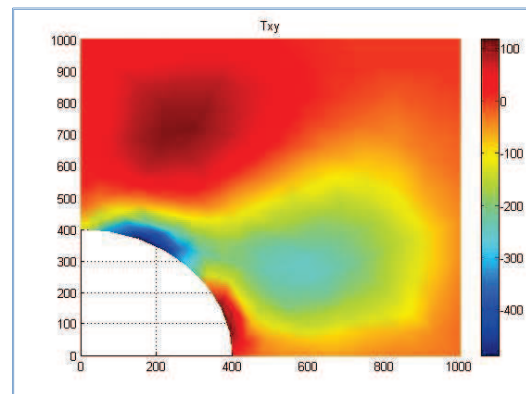


Figura 4.19. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 4 de elementos triangulares. Cotas en mm y tensiones en N/mm².

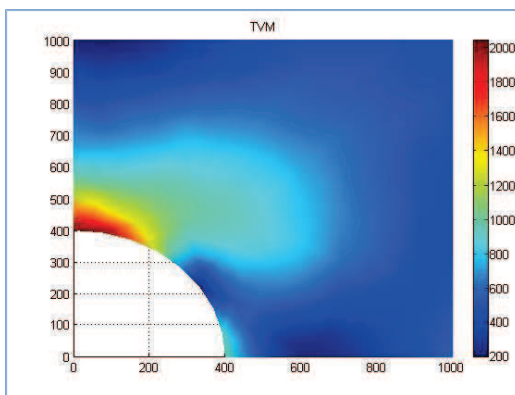
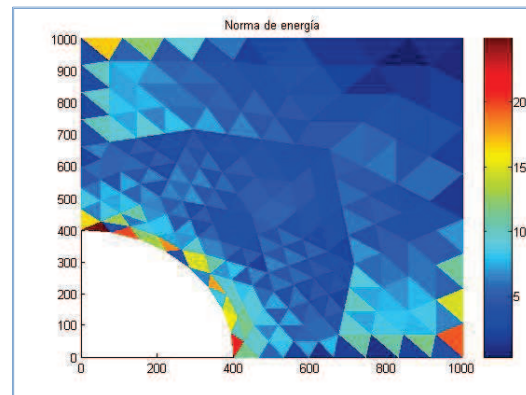


Figura 4.20. Gráfica de la norma de energía de los elementos para el caso 4 de elementos triangulares.



- **Caso 5:** 681 nodos, 1280 elementos y Gauss igual a 1.

En la tabla 4.7 se muestran los valores obtenidos en los nodos para el caso 5.

Tabla 4.7. Resultados en los nodos para el caso 5 con mallado de 1280 elementos triangulares. Desplazamientos en mm y tensiones en N/mm²

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,9438	117,9000	3,6496	10,5838	117,5563
2	1,7160	0,4929	499,4077	-0,5400	0,7211	499,6795
3	5,5176	0,0000	495,1675	659,7461	-12,7821	595,1984
4	4,8315	0,0000	-2,5229	-1138,0791	26,5275	1137,7479
5	0,0000	-2,5804	2257,8123	52,4036	-76,3624	2235,9872

Para estudiar los tiempos de cálculo de las distintas funciones del programa, en la tabla 4.8 se muestran estos.

Tabla 4.8. Tiempos de cálculo de las funciones del programa para el caso 5.

Cálculo	Matriz K	Vector F	Desplazamientos	Tensiones	Error	Ten von Mises
Tiempo (s)	0,0310	0,0000	0,0020	0,0160	0,0700	0,0010

Por último, en las figuras 4.21 – 4.22, se muestran las graficas de los resultados obtenidos para el caso 5:

Figura 4.21. Gráfica de los elementos y su deformada(x10) del caso 5 de elementos triangulares. Cotas en mm.

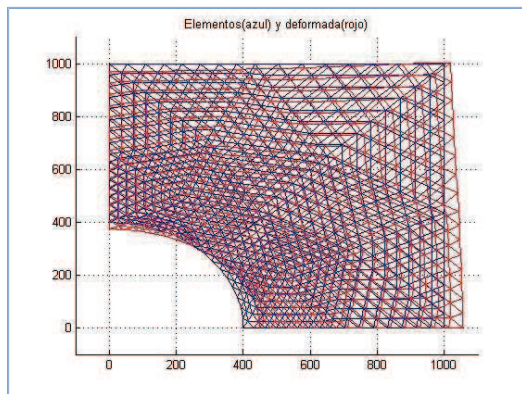


Figura 4.22. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 5 de elementos triangulares. Cotas en mm y tensiones en N/mm².

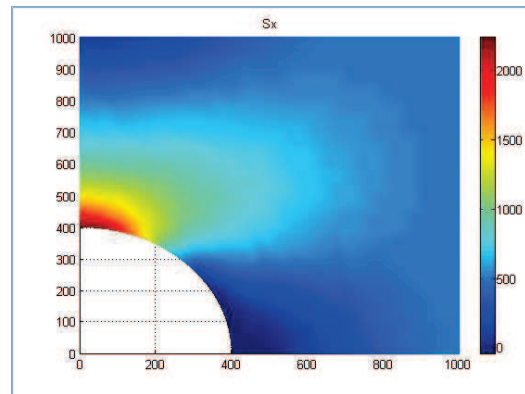


Figura 4.23. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 5 de elementos triangulares. Cotas en mm y tensiones en N/mm^2 .

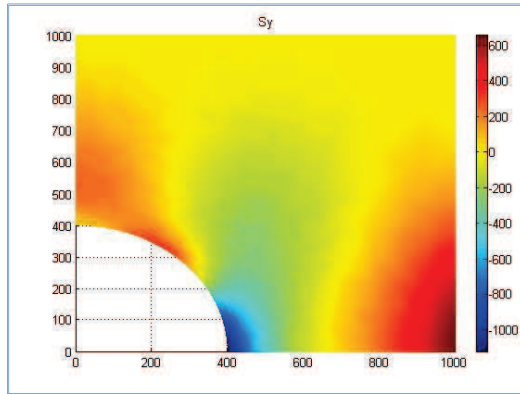


Figura 4.24. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 5 de elementos triangulares. Cotas en mm y tensiones en N/mm^2 .

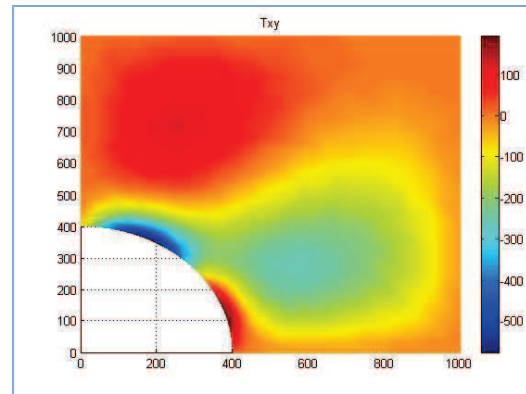


Figura 4.25. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 5 de elementos triangulares. Cotas en mm y tensiones en N/mm^2 .

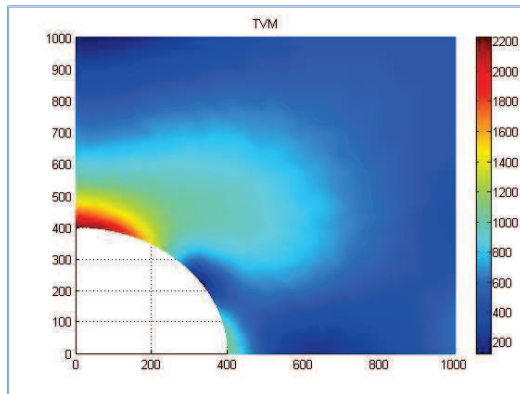
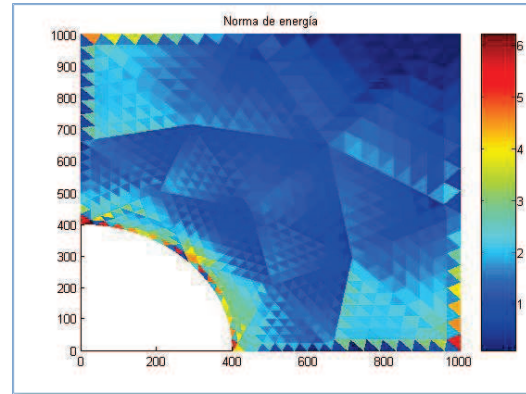


Figura 4.26. Gráfica de la norma de energía de los elementos para el caso 5 de elementos triangulares.



- Elementos cuadrilaterales:
- **Caso 6:** 28 nodos, 18 elementos y Gauss igual a 1 x 1.

En la tabla 4.9 se muestran los valores obtenidos en los nodos para el caso 1.

Tabla 4.9. Resultados en los nodos para el caso 6 con mallado de 18 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm^2

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-14,4471	160,4469	63,7680	63,7680	178,2618
2	6,7937	-8,9392	471,0035	-18,5577	23,1972	482,2280
3	0,7021	0,0000	478,8505	438,1736	-16,9196	460,7961
4	5,5557	0,0000	59,3431	-712,1175	-118,6862	771,4606
5	0,0000	0,9007	1539,2493	63,7680	-63,7680	1512,4150

Por último, en las figuras 4.27 – 4.32 se muestran las gráficas obtenidas para el caso 6:

Figura 4.27. Gráfica de los elementos y su deformada del caso 6 de elementos cuadrangulares. Cotas en mm.

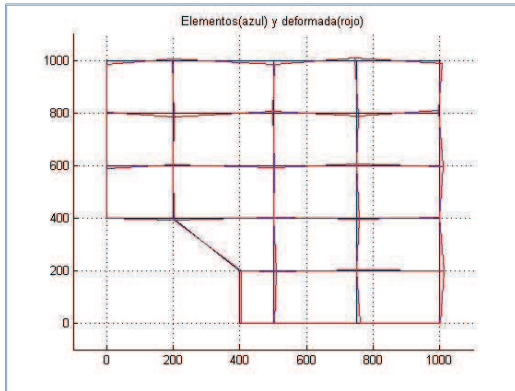


Figura 4.28. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 6 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

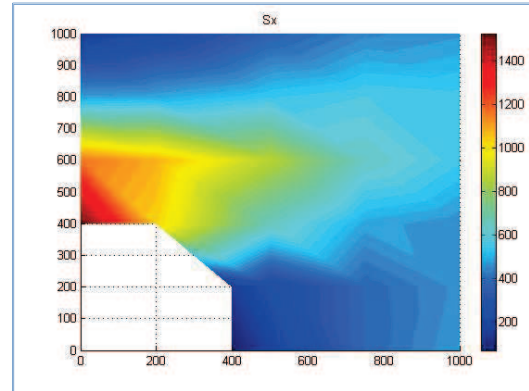


Figura 4.29. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 6 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

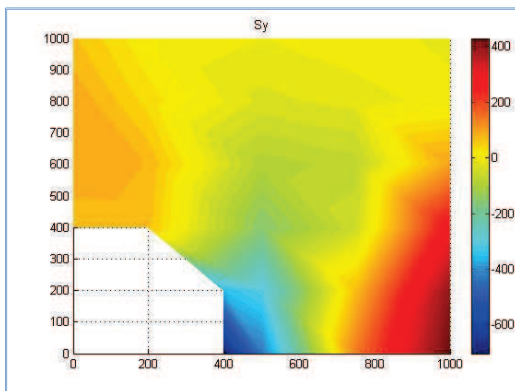


Figura 4.30. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 6 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

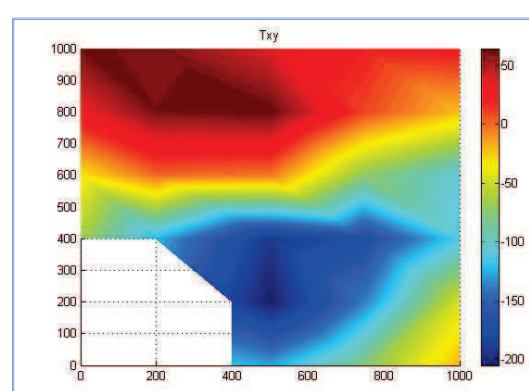


Figura 4.31. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 6 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

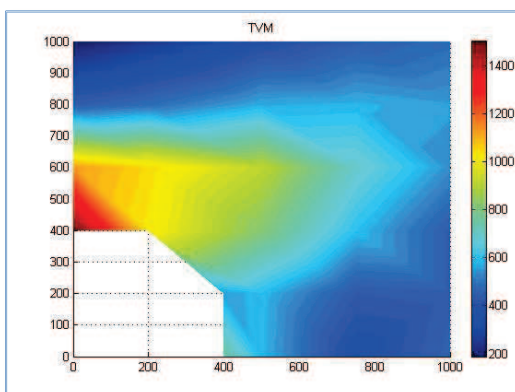
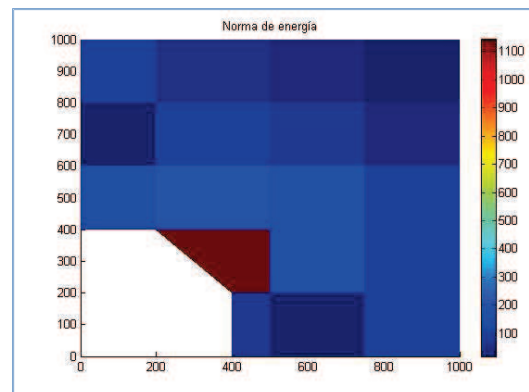


Figura 4.32. Gráfica de la norma de energía de los elementos para el caso 6 de elementos cuadrangulares.



- **Caso 7:** 28 nodos, 18 elementos y Gauss igual a 2 x 2.

En la tabla 4.10 y las figuras 4.33 – 4.38 se muestran los resultados obtenidos para el caso 7.

Tabla 4.10. Resultados en los nodos para el caso 7 con mallado de 18 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm²

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,8757	123,7278	-77,8024	-106,6953	255,2249
2	1,7312	0,5274	518,2130	-11,1759	8,2894	524,0871
3	5,5585	0,0000	296,5273	118,9937	-203,0631	436,4710
4	4,9045	0,0000	-275,5644	-1270,6788	246,9621	1234,2517
5	0,0000	-2,3573	832,8116	-196,6506	274,9618	1059,6375

Figura 4.33. Gráfica de los elementos y su deformada(x10) del caso 7 de elementos cuadrangulares. Cotas en mm.

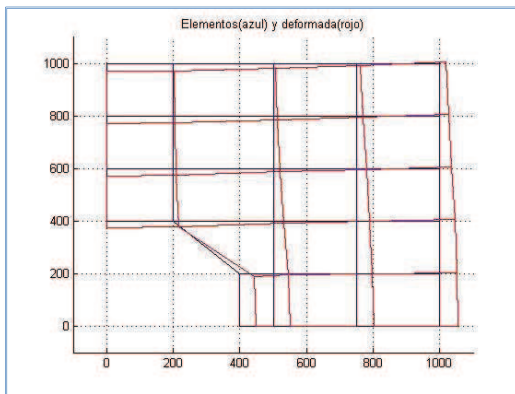


Figura 4.34. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 7 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

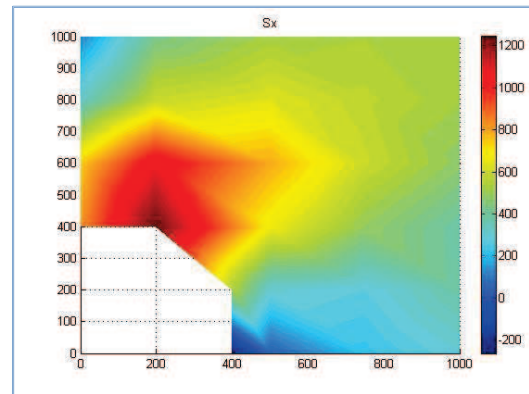


Figura 4.35. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 7 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

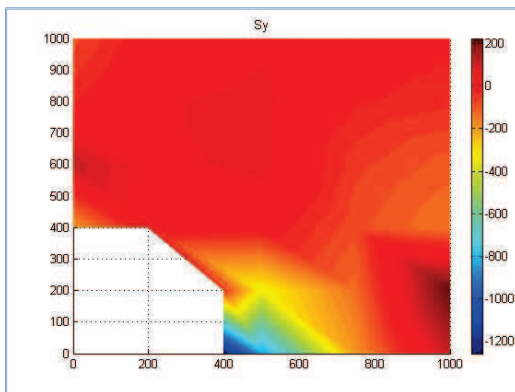


Figura 4.36. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) del caso 7 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

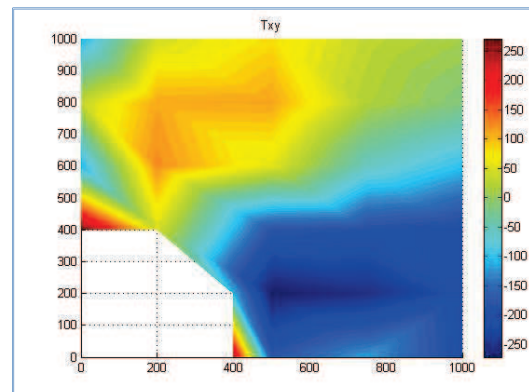


Figura 4.37. Grafica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 7 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

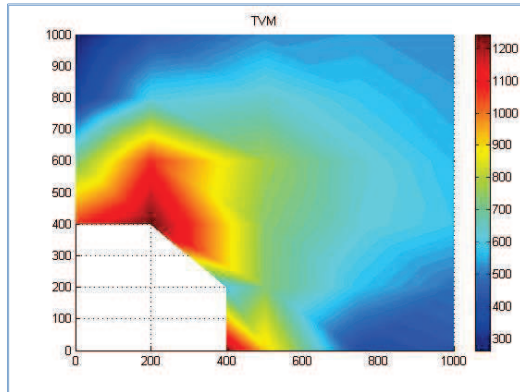
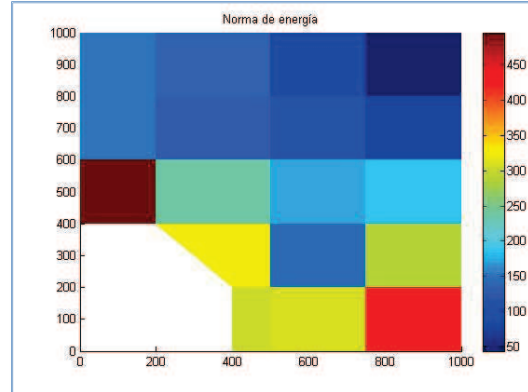


Figura 4.38. Grafica de la norma de energía de los elementos para el caso 7 de elementos cuadrangulares.



- **Caso 8:** 28 nodos, 18 elementos y Gauss igual a 2 x 2 para integración y 1 x 1 para el cálculo de tensiones.

En la tabla 4.11 y las figuras 4.39 – 4.44 se muestran los resultados obtenidos para el caso 8.

Tabla 4.11. Resultados en los nodos para el caso 8 con mallado de 18 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,8757	344,8893	19,2836	28,0593	339,1633
2	1,7312	0,5274	507,8374	-2,6770	-0,4109	509,1817
3	5,5585	0,0000	437,5852	387,3830	-65,7761	430,1312
4	4,9045	0,0000	-27,7337	-688,6960	-2,8798	675,2749
5	0,0000	-2,3573	1416,2280	123,3151	-81,9341	1366,1646

Figura 4.39. Gráfica de los elementos y su deformada(x10) del caso 8 de elementos cuadrangulares. Cotas en mm.

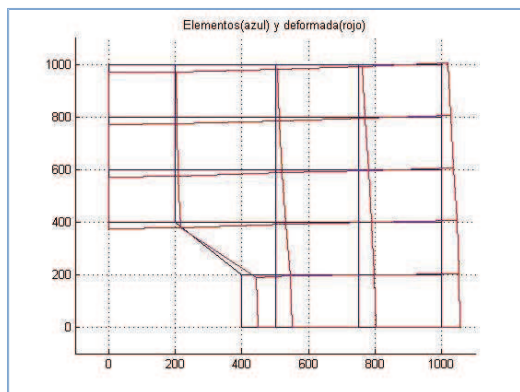


Figura 4.40. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 8 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

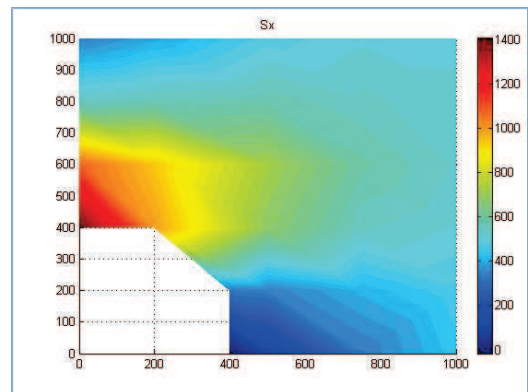


Figura 4.41. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 8 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

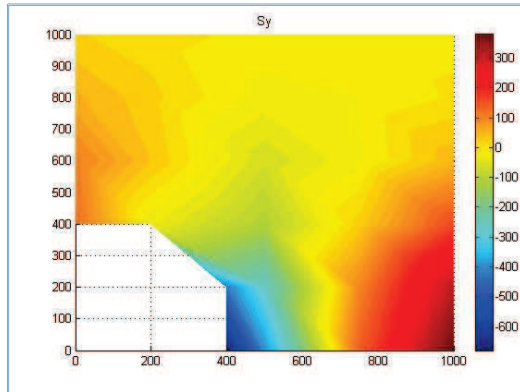


Figura 4.42. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 8 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

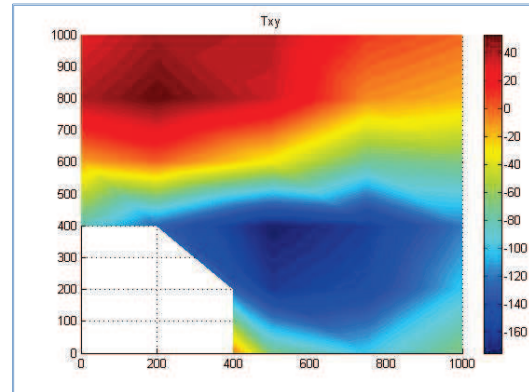


Figura 4.43. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 8 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

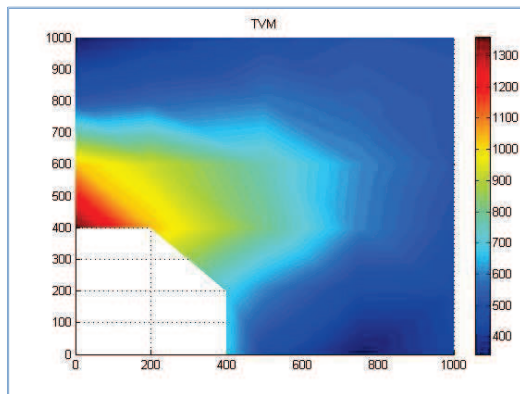
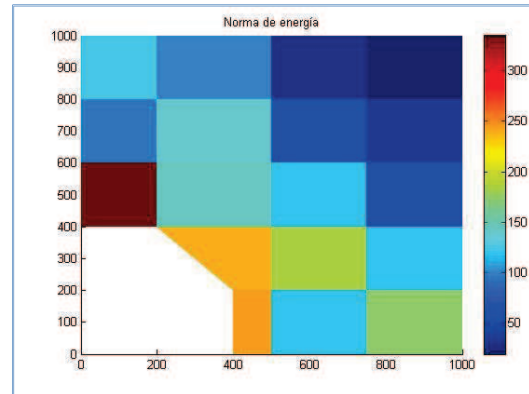


Figura 4.44. Gráfica de la norma de energía de los elementos para el caso 8 de elementos cuadrangulares.



– **Caso 9:** 34 nodos, 23 elementos y Gauss igual a 1 x 1.

En la tabla 4.12 y las figuras 4.45 – 4.50 se muestran los resultados obtenidos para el caso 9.

Tabla 4.12. Resultados en los nodos para el caso 9 con mallado de 23 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm^2 .

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-3,1264	252,3076	36,5552	30,4627	241,9839
2	1,4599	0,6162	503,4896	5,0251	-4,1875	501,0485
3	5,7384	0,0000	452,7466	450,3152	-47,2534	458,8935
4	4,1162	0,0000	63,8664	-469,8985	-50,1753	512,2957
5	0,0000	-2,0001	1370,1666	187,3495	-147,1899	1311,7740

Figura 4.45. Gráfica de los elementos y su deformada(x10) del caso 9 de elementos cuadrangulares. Cotas en mm.

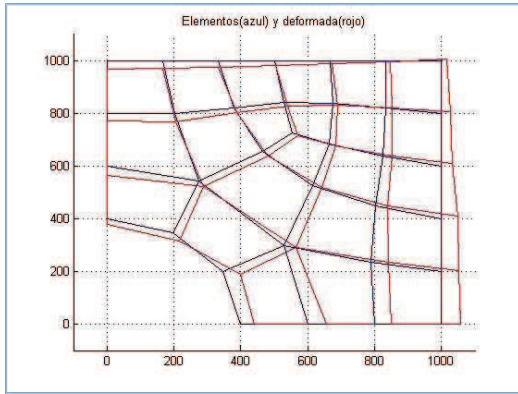


Figura 4.46. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 9 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

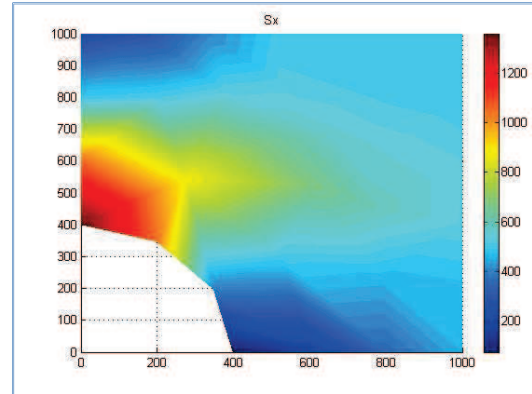


Figura 4.47. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 9 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

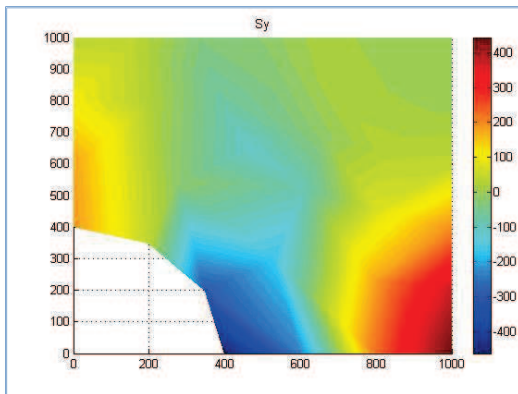


Figura 4.48. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 9 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

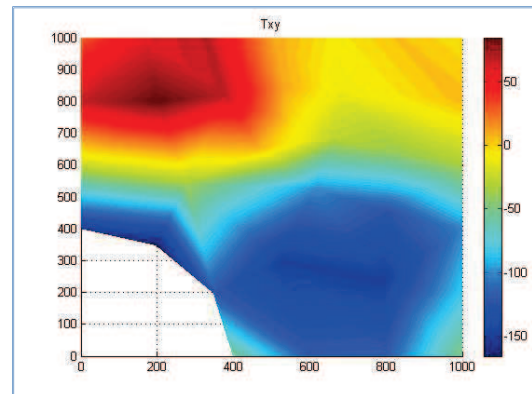


Figura 4.49. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 9 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

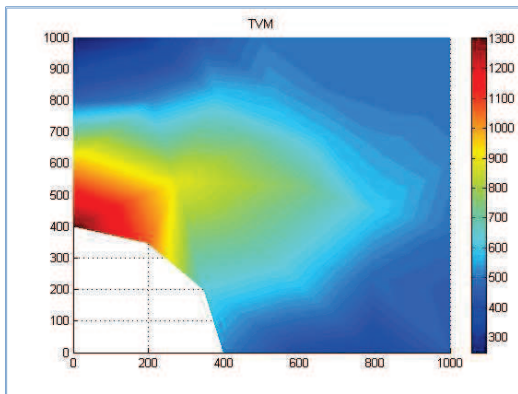
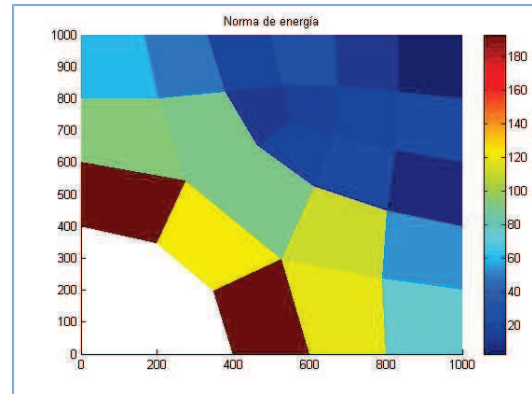


Figura 4.50. Gráfica de la norma de energía de los elementos para el caso 9 de elementos cuadrangulares.



- **Caso 10:** 34 nodos, 23 elementos y Gauss igual a 2 x 2.

En la tabla 4.13 y las figuras 4.51 – 4.56 se muestran los resultados obtenidos para el caso 10.

Tabla 4.13. Resultados en los nodos para el caso 10 con mallado de 23 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm²

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,3928	644,6983	62,0511	29,0022	618,0659
2	1,8980	0,2668	507,7958	5,2197	3,7504	505,2480
3	4,8848	0,0000	314,4849	-8,1310	-169,6902	433,4838
4	4,1290	0,0000	297,4126	270,0529	-42,6330	294,1401
5	0,0000	-1,9842	449,4840	-195,0120	219,6162	687,3239

Figura 4.51. Gráfica de los elementos y su deformada(x10) del caso 10 de elementos cuadrangulares. Cotas en mm.

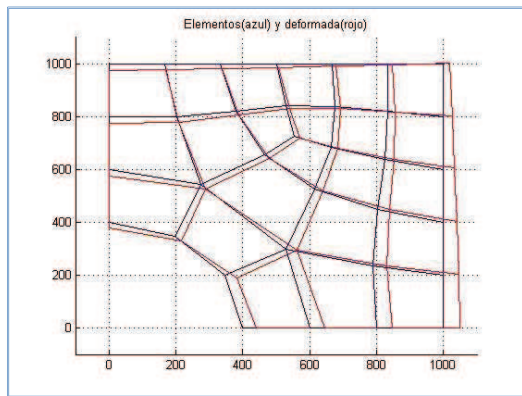


Figura 4.52. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 10 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

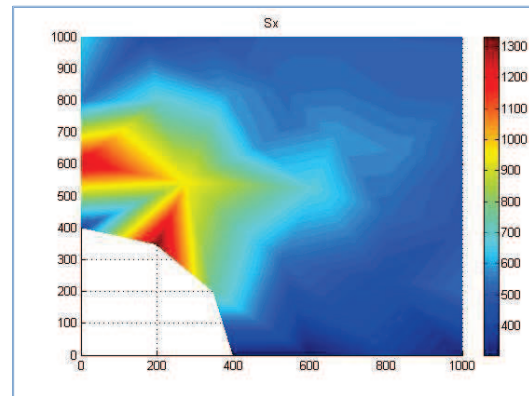


Figura 4.53. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 10 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

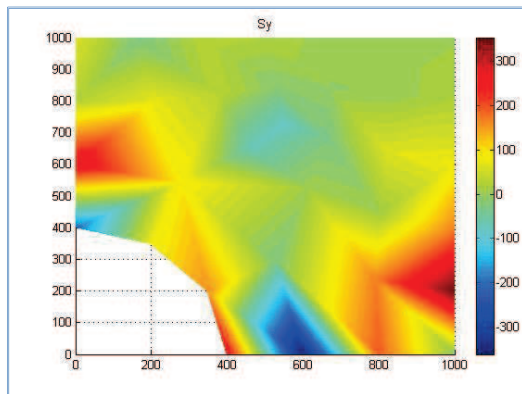


Figura 4.54. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 10 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

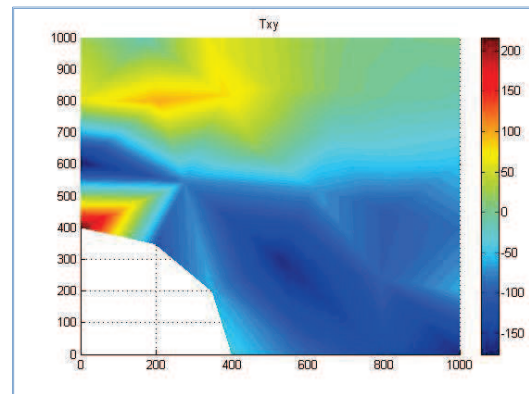


Figura 4.55. Grafica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 10 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

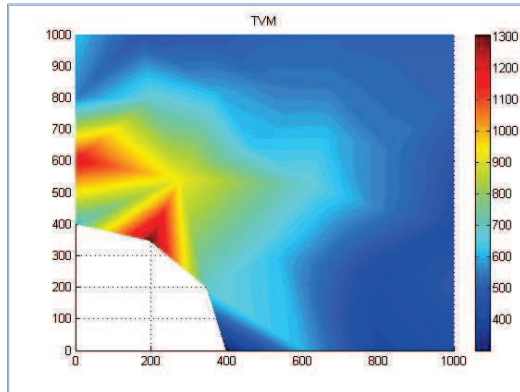
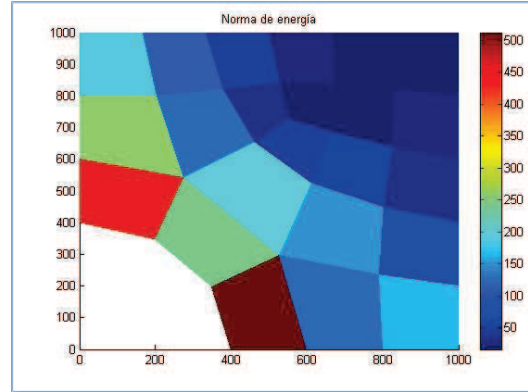


Figura 4.56. Grafica de la norma de energía de los elementos para el caso 10 de elementos cuadrangulares.



- **Caso 11:** 34 nodos, 23 elementos y Gauss igual a 2 x 2 para el cálculo de integrales y 1 x 1 para las tensiones.

En la tabla 4.14 y las figuras 4.57 – 4.62 se muestran los resultados obtenidos para el caso 11.

Tabla 4.14. Resultados en los nodos para el caso 11 con mallado de 23 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,3928	380,2700	15,6692	39,4327	378,8893
2	1,8980	0,2668	501,5145	0,4307	0,3842	501,2998
3	4,8848	0,0000	476,6305	325,6923	-60,7671	434,8507
4	4,1290	0,0000	127,4642	-354,6072	-106,4797	470,3263
5	0,0000	-1,9842	1250,7439	146,0503	-85,7684	1193,7705

Figura 4.57. Gráfica de los elementos y su deformada(x10) del caso 11 de elementos cuadrangulares. Cotas en mm.

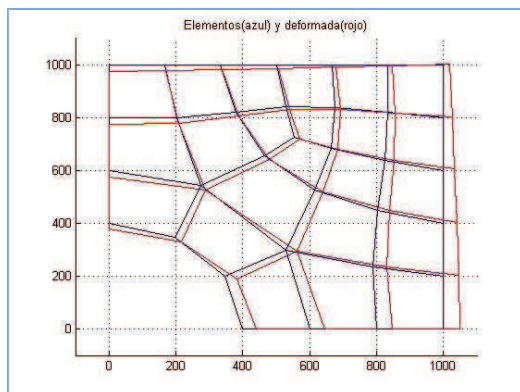


Figura 4.58. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 11 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

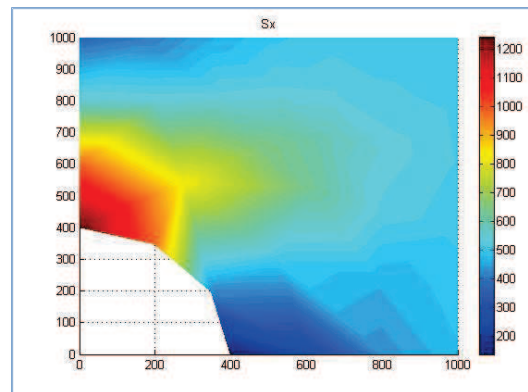


Figura 4.59. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 11 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

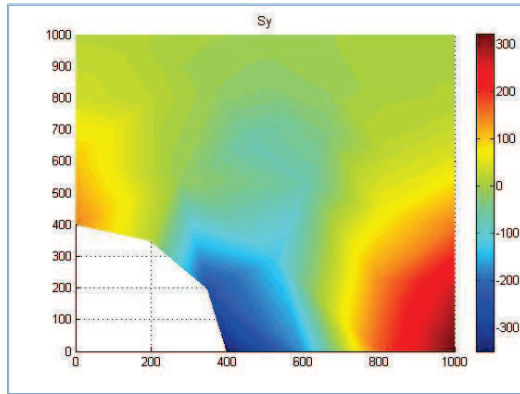


Figura 4.60. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 11 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

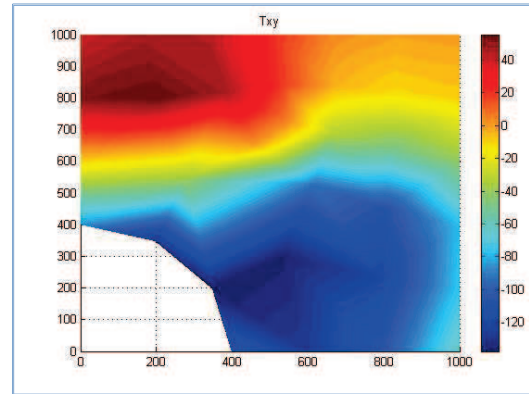


Figura 4.61. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 11 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm².

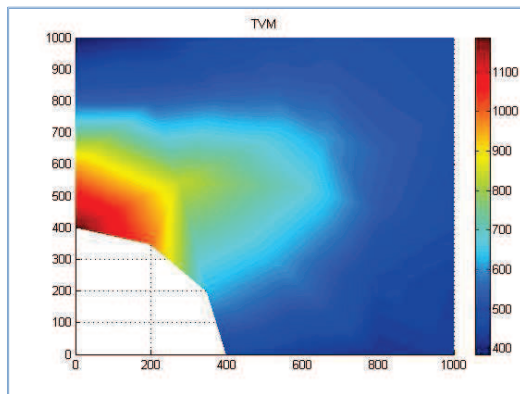
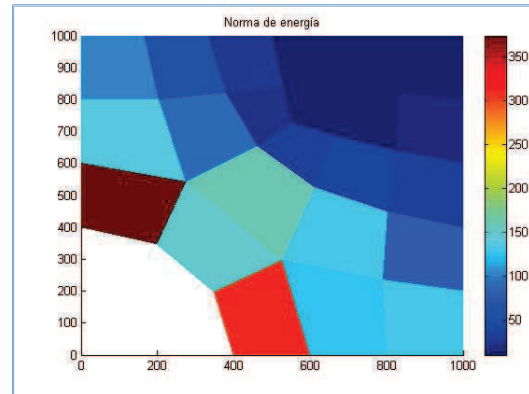


Figura 4.62. Gráfica de la norma de energía de los elementos para el caso 11 de elementos cuadrangulares.



- **Caso 12:** 111 nodos, 90 elementos y Gauss igual a 2 x 2 para el cálculo de integrales y 1 x 1 para las tensiones.

En la tabla 4.15 y las figuras 4.63 – 4.68 se muestran los resultados obtenidos para el caso 12.

Tabla 4.15. Resultados en los nodos para el caso 12 con mallado de 90 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}	σ_{eq}
1	0,0000	-2,8215	221,7525	7,1109	11,5473	219,1983
2	1,7541	0,4494	499,8265	-0,1938	0,7340	499,9250
3	5,3848	0,0000	490,6419	532,9453	-20,0927	514,2821
4	4,6699	0,0000	-27,2654	-749,0694	15,1367	736,2826
5	0,0000	-2,4497	1755,0305	156,8863	-116,2452	1694,0909

Figura 4.63. Gráfica de los elementos y su deformada($\times 10$) del caso 12 de elementos cuadrangulares. Cotas en mm.

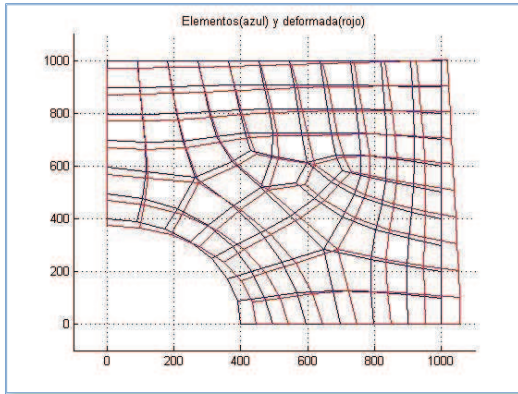


Figura 4.64. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el caso 12 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

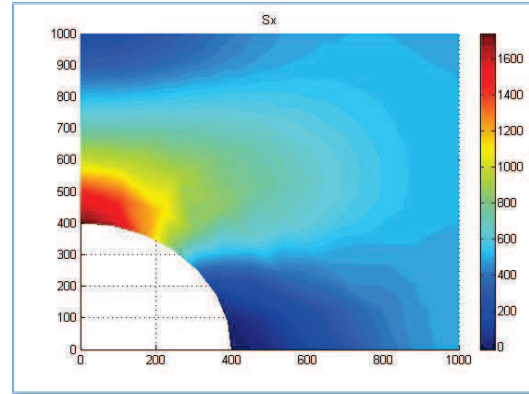


Figura 4.65. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el caso 12 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

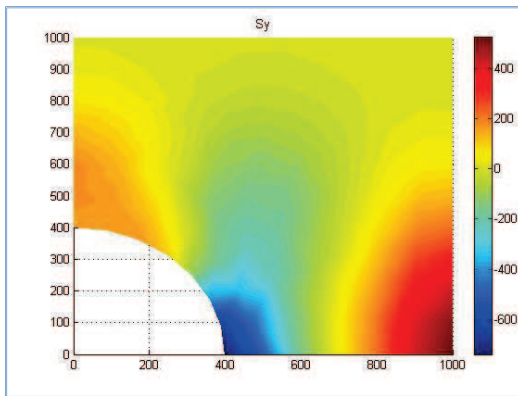


Figura 4.66. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el caso 12 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

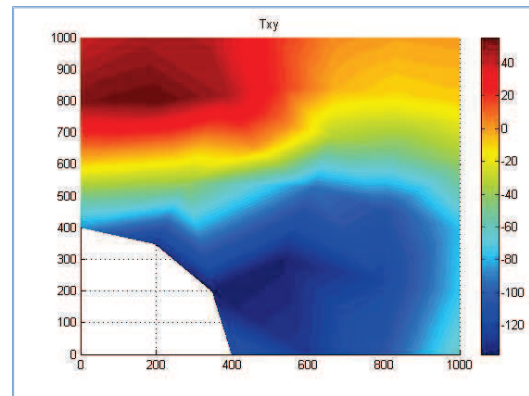


Figura 4.67. Gráfica de las tensiones de von Mises (σ_{eq}) en los nodos para el caso 12 de elementos cuadrangulares. Cotas en mm y tensiones en N/mm^2 .

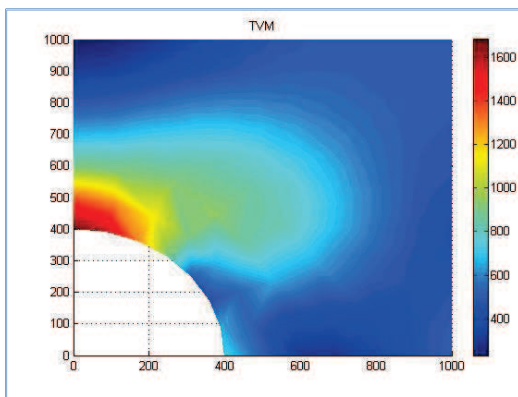
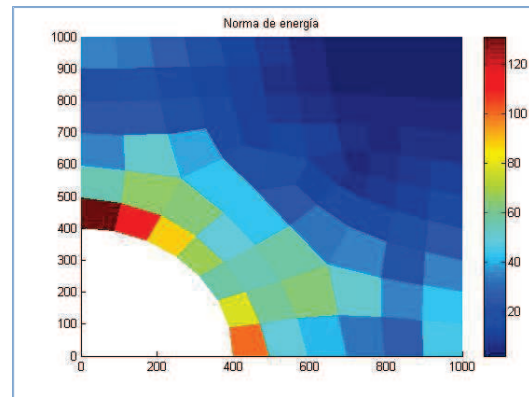


Figura 4.68. Gráfica de la norma de energía de los elementos para el caso 12 de elementos cuadrangulares.



4.2. Resultados analíticos del problema

Para poder estudiar el comportamiento de los resultados obtenidos es necesario obtener la solución analítica del problema. Esta solución analítica, en coordenadas polares, se expuso en 4.1

$$\sigma_r = \frac{S}{2} \left(1 - \frac{a^2}{r^2} \right) + \frac{S}{2} \left(1 + \frac{3a^4}{r^4} - \frac{4a^2}{r^2} \right) \cos(2\theta)$$

$$\sigma_\theta = \frac{S}{2} \left(1 + \frac{a^2}{r^2} \right) - \frac{S}{2} \left(1 + \frac{3a^4}{r^4} \right) \cos(2\theta)$$

$$\tau_{r\theta} = -\frac{S}{2} \left(1 - \frac{3a^4}{r^4} + \frac{2a^2}{r^2} \right) \sin(2\theta)$$

Siendo θ y r las coordenadas polares del punto y S la tensión en la sección transversal sin orificio.

Si evaluamos estas funciones en los nodos de estudio y realizamos el cambio de ejes necesario para transformar la solución en polares a la solución en los ejes cartesianos del problema se pueden obtener la solución analítica del problema estudiado.

4.3. Resultados de otros programas

Con el fin de poder evaluar el correcto funcionamiento del programa, se deben comparar los resultados no solo con la solución exacta al problema, sino con los resultados que se obtienen de otros programas. Para ello se obtendrán los resultados obtenidos por el programa MEFI (programa de la Universidad Politécnica de Cartagena) y el programa COMSOL.

Con el programa MEFI se obtendrán los resultados de la discretización mediante elementos triangulares y con el programa COMSOL se obtendrán los resultados de la discretización mediante elementos cuadrangulares.

Los resultados obtenidos con el programa MEFI para dos mallados distintos, el primero con una malla de 16 nodos y 20 elementos y un segundo mallado de 648 nodos y 1229 elementos. Los resultados obtenidos se muestran en las tablas 4.16 y 4.17:

Tabla 4.16. Resultados del programa MEF para una malla de 16 nodos y 20 elementos triangulares. Desplazamientos en mm y tensiones en N/mm²

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}
1	0,0000	-1,9641	471,3000	38,5530	59,8770
2	2,1073	-0,0217	523,8000	12,2770	-17,1700
3	4,3298	0,0000	454,3100	186,5500	-97,8740
4	3,4620	0,0000	287,1200	-223,0900	-145,2200
5	0,0000	-1,5118	1114,3000	81,2040	-41,5270

Tabla 4.17. Resultados del programa MEF para una malla de 648 nodos y 1229 elementos triangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}
1	0,0000	-2,9179	120,3200	8,0042	8,1404
2	1,7219	0,4859	499,5700	-0,3981	0,5750
3	5,4877	0,0000	488,7700	655,2600	-11,8340
4	4,8029	0,0000	-28,1090	-1018,7000	19,5020
5	0,0000	-2,5657	2121,4000	101,7300	-82,6880

Los resultados obtenidos con el programa COMSOL para dos mallados distintos, el primero con una malla de 34 nodos y 23 elementos y un segundo mallado de 111 nodos y 90 elementos. Los resultados obtenidos se muestran en las tablas 4.18 y 4.19:

Tabla 4.18. Resultados del programa COMSOL para una malla de 34 nodos y 23 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}
1	0,0000	-2,3888	216,0011	11,2093	72,8656
2	1,9009	0,2644	496,0682	-2,9414	-1,6899
3	4,8819	0,0000	471,8397	500,5887	-106,4330
4	4,1262	0,0000	0,4797	-879,9814	-238,2388
5	0,0000	-1,9892	2059,6833	309,0177	111,1999

Tabla 4.19. Resultados del programa COMSOL para una malla de 111 nodos y 90 elementos cuadrangulares. Desplazamientos en mm y tensiones en N/mm².

Nodos	u_x	u_y	σ_x	σ_y	τ_{xy}
1	0,0000	-2,8215	96,8002	-21,8880	48,2024
2	1,7541	0,4494	498,2645	-1,8676	-0,1465
3	5,3848	0,0000	517,7132	677,6175	-65,7967
4	4,6699	0,0000	-123,0436	-1187,9834	-101,6636
5	0,0000	-2,4497	2359,9557	290,5019	45,3168

4.4. Análisis de los resultados

Para evaluar el correcto funcionamiento del programa *MEFBID* se compararan los resultados obtenidos en los diferentes casos con los obtenidos con los diferentes programas. Ya que para elementos triangulares nos e ha utilizado el mismo mallado, los valores conseguidos por *MEFBID* deben ser cercanos a los obtenidos con otro programa. Para elementos cuadrangulares si se ha utilizado el mismo mallado para el cálculo de desplazamientos y tensiones, aunque, al ser COMSOL un programa comercial, el cálculo de las tensiones se realiza con un método diferente al utilizado en *MEFBID*.

En primer lugar se comparan los desplazamientos calculados por los diferentes mallados con el fin de observar si el cálculo de la matriz de rigidez y la resolución del sistema de ecuaciones es correcto. En las tabla 4.20 – 4.23 se observan los desplazamientos obtenidos para elementos. Donde el subíndice i indica el número de elementos de la malla y el superíndice j indica el programa utilizado (1 para *MEFBID*, 2 para MEF1 y 3 para COMSOL).

Tabla 4.20. Desplazamientos (mm) nodales de elementos triangulares. Comparación 1

Nodo	u_{20x}^1	u_{20x}^2	u_{20y}^1	u_{20y}^2
1	0,0000	0,0000	-1,9199	-1,9641
2	2,1666	2,1073	-0,0463	-0,0217
3	4,3111	4,3298	0,0000	0,0000
4	3,4758	3,4620	0,0000	0,0000
5	0,0000	0,0000	-1,4769	-1,5118

Tabla 4.21. Desplazamientos (mm) nodales de elementos triangulares. Comparación 2

Nodos	u_{1280x}^1	u_{1229x}^2	u_{1280y}^1	u_{1229y}^2
1	0,0000	0,0000	-2,9438	-2,9179
2	1,7160	1,7219	0,4929	0,4859
3	5,5176	5,4877	0,0000	0,0000
4	4,8315	4,8029	0,0000	0,0000
5	0,0000	0,0000	-2,5804	-2,5657

Tabla 4.22. Desplazamientos (mm) nodales de elementos cuadrilaterales. Comparación 1.

Nodos	u_{23x}^1	u_{23x}^3	u_{23y}^1	u_{23y}^3
1	0,0000	0,0000	-2,3928	-2,3888
2	1,8980	1,9009	0,2668	0,2644
3	4,8848	4,8819	0,0000	0,0000
4	4,1290	4,1262	0,0000	0,0000
5	0,0000	0,0000	-1,9842	-1,9892

Tabla 4.23. Desplazamientos (mm) nodales de elementos cuadrilaterales. Comparación 2.

Nodos	u_{90x}^1	u_{90x}^3	u_{90y}^1	u_{90y}^3
1	0,0000	0,0000	-2,8215	-2,8215
2	1,7541	1,7541	0,4494	0,4494
3	5,3848	5,3848	0,0000	0,0000
4	4,6699	4,6699	0,0000	0,0000
5	0,0000	0,0000	-2,4497	-2,4497

Como se puede observar, los resultados de desplazamientos calculados por *MEFBID* son casi exactamente iguales que los calculados en otros programas. Es decir, se puede tomar como valido el programa para el cálculo de la matriz de rigidez y la solución del sistema de ecuaciones y, por tanto, el cálculo de los desplazamientos nodales.

A continuación se comparan las tensiones calculadas en los nodos. Para ello, compararemos las tensiones normales en el eje x calculadas por los programas comerciales con las calculadas en el programa *MEFBID*. Las tensiones a comparar se encuentran en las tablas 4.24 para elementos triangulares y en la tabla 4.25 para elementos cuadrangulares.

Tabla 4.24. Tensiones (N/mm^2) nodales de los elementos triangulares. Comparación 1

Nodo	σ_{20x}^1	σ_{20x}^2	σ_{1229x}^1	σ_{1229x}^2
1	507,07	471,30	117,90	120,32
2	530,34	523,80	499,41	499,57
3	450,85	454,31	495,17	488,77
4	244,81	287,12	-2,52	-28,11
5	1149,81	1114,30	2257,81	2121,40

Tabla 4.25. Tensiones (N/mm^2) nodales de los elementos cuadrangulares. Comparación 2.

Nodos	σ_{23x}^1	σ_{23x}^3	σ_{90x}^1	σ_{90x}^3
1	380,27	216,00	221,75	96,80
2	501,52	496,07	499,83	498,26
3	476,63	471,84	490,64	517,71
4	127,46	0,48	-27,27	-123,04
5	1250,74	2059,68	1755,03	2359,96

Comparando los resultados, las tensiones obtenidas por *MEFBID* en elementos triangulares, tabla 4.24, son muy aproximadas a las calculadas por MEF1. Por el contrario, en elementos cuadrilaterales, tabla 4.25, las tensiones obtenidas en los nodos no se acercan a las calculadas en el programa comercial. Esto se debe a que COMSOL utiliza un método mejorado para el cálculo de tensiones, por lo que los resultados obtenidos en elementos cuadrilaterales son bastante mejores a los calculados por un alisado local de las tensiones.

Dejando aparte este hecho, se puede llegar a la conclusión que el programa *MEFBID* realiza un cálculo correcto de los desplazamientos nodales y de las tensiones nodales, además de todos los cálculos que para la obtención de estos son necesarios.

Además de comparar los datos con otros programas, el programa puede servir como base para analizar algunos de los estudios realizados en elementos finitos y en sus programas de cálculos. En los siguientes apartados se realizarán dichos análisis, apoyándonos en los resultados obtenidos por el programa *MEFBID* en los distintos casos.

4.4.1 Análisis de los tiempos de cálculo.

Como se puede observar en la tabla 4.8, donde se recogen los tiempos que el ordenador tarda en realizar los cálculos más importantes para la resolución del problema con una malla de 1280 elementos, hay grandes diferencias entre ellos. Esto nos indica en que partes del código hay que realizar un mayor esfuerzo en programar correctamente el método.

Si se consigue mejorar estas partes del código, principalmente en el cálculo de la matriz de rigidez, se puede ahorrar importantes tiempos de cálculo, ya que el cálculo de las tensiones y el cálculo de la norma de energía de los elementos forma parte del postproceso del programa. Como se puede observar, la resolución del sistema de ecuaciones ocupa una pequeña parte del tiempo total consumido por el programa, algo que sin duda se debe al alto potencial que Matlab ofrece en cuanto a la resolución de estos problemas cuando se trabaja matricialmente.

4.4.2 Análisis de los elementos utilizados.

Al utilizar elementos triangulares y cuadrangulares lineales se consigue simplificar la programación del MEF, aunque con ello se consigue una aproximación algo inexacta a la solución exacta del problema, a no ser que se utilicen mallados muy tupidos, aumentando con ello el tiempo de cálculo del programa.

Aun teniendo en cuenta este hecho, se puede conseguir una solución cercana a la exacta en tiempos razonables si se eligen correctamente los puntos de Gauss y el elemento utilizado.

Con el elemento triangular de tres nodos se consigue acercar considerablemente la geometría de estructuras no rectas, como se puede observar en la discretización del orificio del problema. Además, como se puede observar en los datos obtenidos en los casos 2 y 3 para un mallado de 80 elementos triangulares, los resultados obtenidos para una cuadratura de 1 y de 3 puntos son iguales, siendo más reducido el tiempo de cálculo cuando se utiliza 1 solo punto de Gauss. Esto es debido a que la matriz de deformación B de un elemento triangular lineal es constante en todo el elemento, por lo que toma el mismo valor independientemente del punto en el que se calcule.

Con el elemento cuadrangular lineal se pueden acercar geometrías rectas, perdiendo mucha capacidad de discretizar elementos curvos con un mallado no excesivamente tupido. En este tipo de elementos es donde se hace más importante la elección del número de puntos de Gauss a utilizar.

Como se observa en el caso 6 en el que se obtienen las integrales con una cuadratura de 1×1 puntos, se produce una gran inexactitud en el cálculo de los desplazamientos nodales. Esto se debe a que este elemento, si se elige la cuadratura mínima, no se verifica el test de la parcela (Zienkiewicz et al, 2005) ya que no se mantiene la convergencia necesaria y da origen a mecanismos internos (modos de energías nulos), produciendo desplazamientos nodales incorrectos. Este hecho se puede observar en el caso 6, donde se producen desplazamientos nodales muy alejados de los calculados con diferentes cuadraturas.

Si se realiza una cuadratura de 2×2 puntos en el mismo mallado estos mecanismos internos desaparecen y se obtiene desplazamientos nodales correctos. Sin embargo, si se utiliza este número de puntos para obtención de las tensiones nodales, se obtiene (caso 7 y caso 10) una solución de tensiones inexacta y discontinua. Esta solución puede ser mejorada si se utilizan 1×1 puntos para la obtención de las tensiones alisadas, dando valores de tensiones mejorados, como se puede observar en los casos 8 y 11. Este hecho, comprobado por muchos autores (Zienkiewicz et al (2005)), se debe a que con una cuadratura de 1×1 puntos se consigue la solución óptima del campo de tensiones de un elemento cuadrangular lineal.

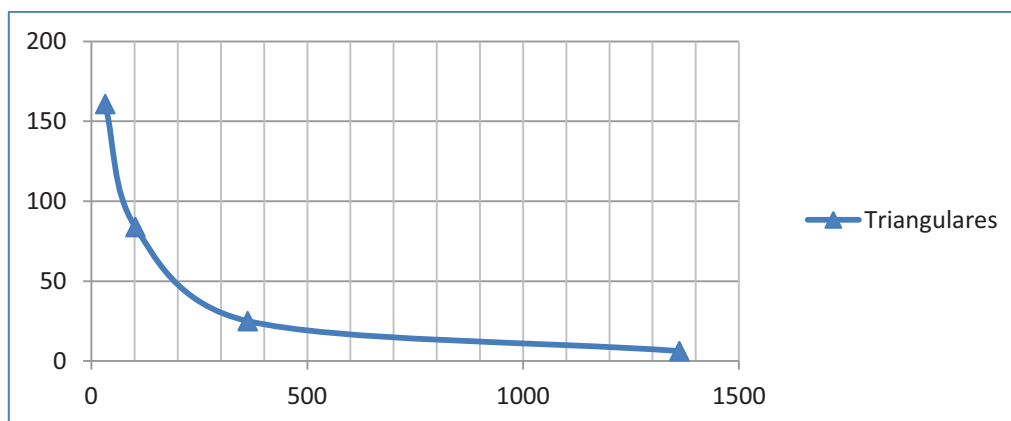
4.4.3. Análisis de la norma de energía

Además de las tensiones nodales alisadas, el programa calcula la norma de energía de cada elemento. Esta norma de energía da una idea del error que se comete en el cálculo de las tensiones, siendo mayor el error cuando mayor sea la norma de la energía.

Si se observan las figuras donde se indica la norma de energía, esta siempre es máxima cerca del nodo 5 de estudio, donde mayor tensión normal en el eje x se produce en la estructura y donde mayor tensión de von Mises se produce (y por lo tanto el punto crítico de la estructura).

Esta norma de energía disminuye conforme se va refinando la malla y, por lo que se verá más adelante, conforme se va acercando la solución del programa a la solución exacta del problema. En la figura 4.69 se observa la tendencia a disminuir la norma de energía máxima en los elementos dependiendo de los grados de libertad del mallado para elementos triangulares.

Figura 4.69. Tendencia del valor máximo de la norma de energía de los elementos en función del número de grados de libertad del mallado.

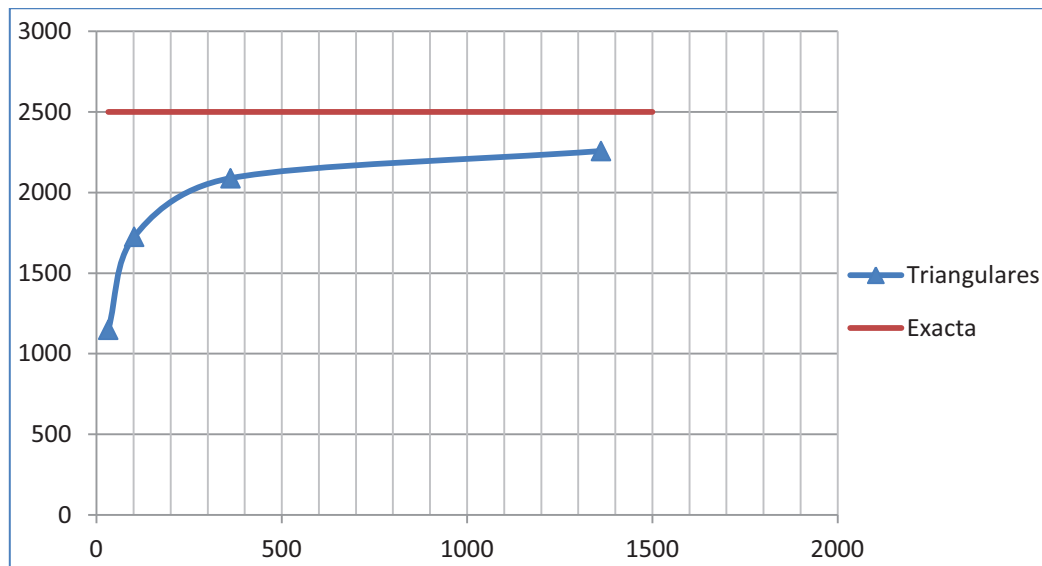


4.4.4. Análisis de la convergencia.

Un aspecto importante en el cálculo con el MEF de una estructura es la convergencia hacia la solución exacta cuando se refina la malla. Para ello, se deben de tener datos suficientes para poder observar claramente la tendencia o no de los valores obtenidos a la solución exacta. En el problema estudiado se realizará el estudio de la convergencia para elementos triangulares ya que no se tienen datos suficientes para mallados rectangulares y el estudio de éste no sería correcto.

La convergencia se estudia en tensión en el punto 5, por ser este punto el más solicitado en la estructura y en el que se produciría en primer lugar la rotura de esta. En la figura 4.70 Se muestra la convergencia de la tensión normal en el eje x de los distintos mallados obtenidos para elementos triangulares.

Figura 4.70. Convergencia de los valores de las tensiones normales al eje x en el punto 5 con distintos mallado.



Como se puede observar, a medida que se realiza un mallado mas tupido (mas grados de libertad en la discretización), se consigue una mayor aproximación a la solución exacta en el punto estudiado, siendo esto evaluable en el resto de nodos de la estructura.

4.5. Otro resultado

El problema anterior se trata de un problema de tensión plana. El programa permite el cálculo de problemas de deformación plana por lo que, para certificar el

correcto comportamiento del programa para este tipo de elasticidad plana se resolverá un problema tipo de deformación plana.

Supóngase un cilindro de pared gruesa, de $r = 5 \text{ mm}$ de radio interior y $R = 20 \text{ mm}$ de radio exterior bajo presión interior uniforme (10 N/mm). El problema se puede estudiar como un problema de deformación plana con un espesor de $e = 1 \text{ mm}$. Datos: $E = 1000 \text{ N/mm}^2$ y $\mu = 0,3$. Al ser un problema con simetría se puede estudiar un cuarto de la estructura.

Los resultados obtenidos con una malla de 301 nodos y 544 elementos se pueden observar en las figuras 4.71-4.77.

Figura 4.71. Gráfica de los elementos y su deformada($\times 10$) del cilindro de pared. Cotas en mm.

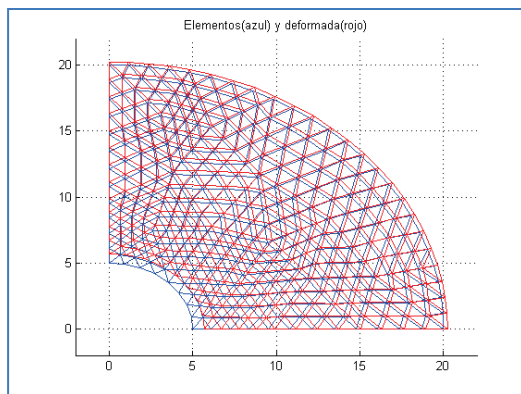


Figura 4.72. Gráfica de las tensiones normales alisadas en el eje x (σ_x) para el cilindro de pared gruesa. Cotas en mm y tensiones en N/mm^2 .

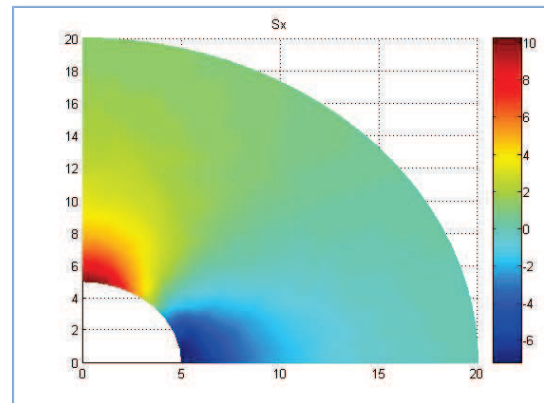


Figura 4.73. Gráfica de las tensiones normales alisadas en el eje y (σ_y) para el cilindro de pared gruesa. Cotas en mm y tensiones en N/mm^2 .

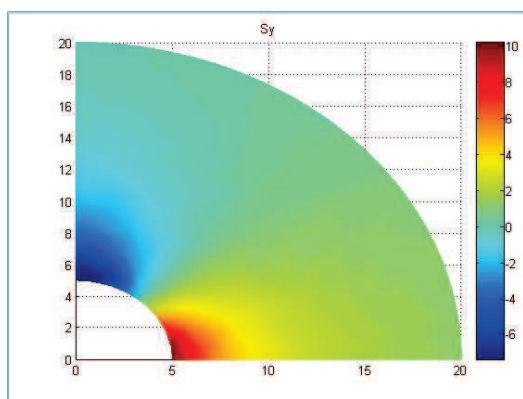


Figura 4.75. Gráfica de las tensiones tangenciales alisadas (τ_{xy}) para el cilindro de pared gruesa. Cotas en mm y tensiones en N/mm^2 .

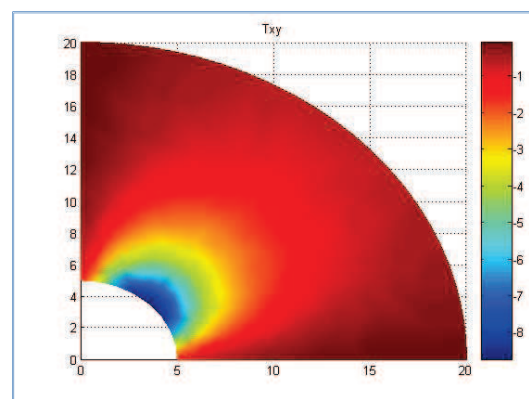


Figura 4.74. Grafica de las tensiones normales alisadas en el eje z (σ_z) para el cilindro de pared gruesa. Cotas en mm y tensiones en N/mm^2 .

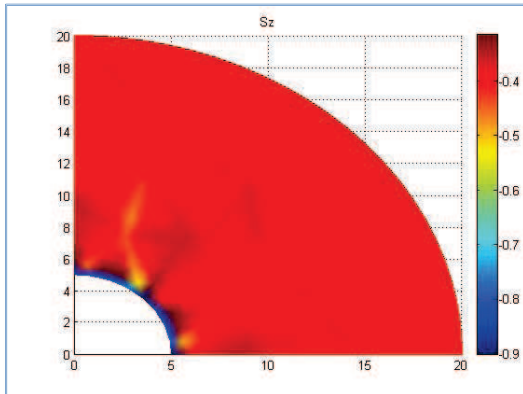


Figura 4.62. Grafica de la norma de energía de los elementos para el cilindro de pared gruesa

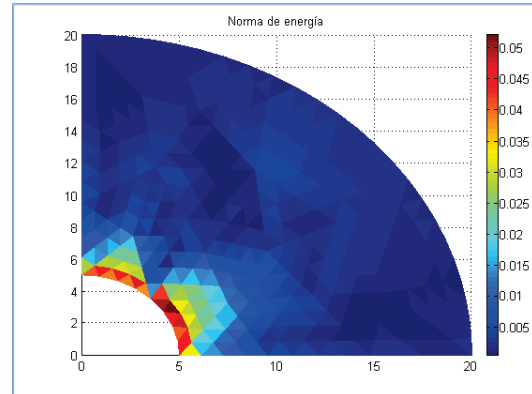
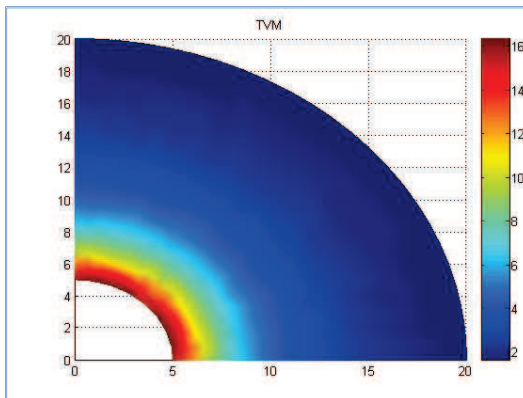


Figura 4.76. Grafica de las tensiones de von Mises (σ_{eq}) en los nodos para el cilindro de pared gruesa. Cotas en mm y tensiones en N/mm^2 .



Si comparamos estos resultados con los de la solución analítica del problema (Timoshenko, 1968), se llega a la conclusión que es correcta ya que es bastante cercana a ella.

CAPÍTULO V

Conclusiones

Contenido

5. Conclusiones.....	72
5.1. Extensión del programa <i>MEFBID</i>	73

5. Conclusiones

- El uso del programa *MEFBID* para el cálculo de estructuras de elasticidad plana mediante el MEF es correcto, siendo los valores de los desplazamientos obtenidos iguales a los que se obtendrían en programas comerciales y los valores de tensiones muy cercanos para elementos triangulares.
- Para una discretización en elementos triangulares de la estructura es suficiente considerar un solo punto para la cuadratura de Gauss-Legendre y para el cálculo óptimo de las tensiones en los nodos. Además, debido a que la matriz de deformación es constante en todo elemento, la formulación de este elemento y el cálculo de las matrices de éste se puede realizar directamente sin la necesidad de realizar la cuadratura.
- Para una discretización en elementos cuadrangulares de la estructura habría que diferenciar entre el número de puntos necesarios para el cálculo de la cuadratura y el número de puntos para el cálculo óptimo de las tensiones, siendo para un elemento lineal de cuatro nodos necesario una cuadratura de 2×2 puntos para el cálculo exacto de las integrales (evitando así que se creen mecanismos en la estructura) y 1×1 puntos para el cálculo de las tensiones nodales.
- En elementos triangulares, con un mallado uniforme en toda la estructura, se necesita una malla muy refinada de elementos para conseguir una solución cercana a la exacta en los puntos de mayor norma de energía.
- Para realizar un refinamiento óptimo del mallado de una estructura no es necesario realizar un remallado en toda la estructura. Si se utiliza la norma de la energía para localizar las zonas donde se produce mayor error para poder alisar en esos lugares el mallado se pueden mejores resultados.
- A diferencia de otros lenguajes de programación, Matlab ofrece la oportunidad de trabajar fácil y eficientemente con matrices independientemente del tamaño de éstas, teniendo la capacidad de resolver un sistema de ecuaciones

matricial en poco tiempo y dando al programador la oportunidad de dedicar más tiempo a la programación del MEF.

5.1. Extensión del programa *MEFBID*

Como cualquier programa de cálculo, la programación de *MEFBID* es susceptible de mejoras. En el presente capítulo se describirán algunas de las más inmediatas para mejorar el funcionamiento con la finalidad de que el programa sirva como base para futuros proyectos de programación del MEF.

Como se dijo en apartados anteriores, la entrada de datos se realiza introduciendo en una función los datos necesarios para la resolución del problema. Esto, en la práctica, es algo tedioso y complicado de trabajar, y que se puede facilitar con un preprocesador adecuado. Un ejemplo de ello es *GID*, el cual te permite, además de una entrada de datos eficiente, unas posibilidades gráficas mayores que *Matlab*.

Otra forma de mejorar el programa es aumentando los posibles elementos a utilizar en el programa, siendo los elementos triangulares y cuadrangulares cuadráticos los primeros que deberían ser formulados. Para ello bastaría con aumentar las funciones ya programadas. También se podrían modificar las líneas de códigos existentes para poder solucionar más tipos de problemas (problemas de tres dimensiones, sólidos de revolución, placas...).

Es importante también modificar las funciones de cálculo del vector de fuerzas para poder incluir fuerzas aplicadas en áreas, fuerzas aplicadas en volumen o para poder introducir el peso propio de la estructura (que es también una fuerza distribuida por unidad de volumen). También podría modificarse el cálculo de las fuerzas puntuales para que éstas no deban estar necesariamente aplicadas en un nodo.

Aunque se ha conseguido un programa que calcula en un tiempo razonable una estructura medianamente tupida (0,2 segundos para un mallado de 1362 grados de libertad), no se ha utilizado todo el potencial que *Matlab* ofrece, si bien el nivel de almacenaje necesario para la ejecución del programa es mínimo ya que, a diferencia de códigos realizados en otros lenguajes de programación, *Matlab* tiene la capacidad de realizar cálculos matriciales rápidamente, siendo menor el coste de recalcular una matriz que el de almacenarla. Si se utiliza el potencial que *Matlab* tiene, se podrían aminorar los tiempos de cálculo y permitiría trabajar con mayor número de grados de libertad en tiempos menores a los actuales.

Para poder estudiar la programación con claridad, se han reducido al mínimo los datos calculados una vez obtenidos los desplazamientos nodales. Además de las

tensiones nodales, tensión de von Mises y la norma de energía calculadas, se podrían programar las funciones necesarias para obtener más información del problema estudiado. Ejemplo de estos datos, muy importantes para el cálculo de estructuras, son las deformaciones unitarias (las cuales se tendrían que calcular de igual forma que las tensiones nodales, calculándolas en los puntos de Gauss y extrapolándola a los nodos), el error, en tanto por ciento, de la norma de energía relativa, tensiones principales en los nodos...

CAPÍTULO VI

Referencias

Contenido

6. Referencias	76
----------------------	----

5. Referencias

- CARMONA, V., FERNÁNDEZ, J.R.** 1998. Introducción a Matlab. Escuela Universitaria Politécnica. Universidad de Sevilla.
- CELIGÜETA, J.T.** 2008. Método de los elementos finitos para análisis estructural. Editorial UNICOPIA C.B. (3ª edición), San Sebastián.
- CUBILLOS, A.** 2007. Guía de Matlab para Ingeniería Mecánica. Universidad de Ibagué.
- Echevarría, R.** Breves apuntes para comenzar con Matlab. Departamento Ecuaciones Diferenciales y Análisis Numérico. Universidad de Sevilla.
- OÑATE, E.** 1995. Cálculo de estructuras por el Método de Elementos Finitos. Análisis estático lineal. Centro Internacional de Métodos Numéricos en Ingeniería (CINME), 2ª edición, Barcelona.
- PERAL, I.** 2004-2005. PDE-toolbox de Matlab (primeros auxilios).
- PIÑEIRO, C.** Apuntes de Matlab. Escuela Técnica Superior de Ingenieros. Universidad de Huelva.
- RESTREPO, J.L.** 1998. Error en la Solución de Problemas de Elasticidad con el Método de los Elementos Finitos. Revista Universitaria EAFIT. Julio-Agosto-Septiembre. 107-118.
- TIMOSHENKO, S., GOODIER, J.N.** 1968. Teoría de la elasticidad. Traducción al castellano de la obra Theory of Elasticity. Urmo, S.A. de Ediciones J. de Ajuriaguerra. Bilbao.
- VÁZQUEZ, M., LÓPEZ, E.** 2001. El Método de los Elementos Finitos aplicado al análisis estructural. Editorial Noela. Madrid.
- ZÁRATE, F., OÑATE, E.** 2006. MAT-fem. Learning Finite Elements with Matlab and GID. Centro Internacional de Métodos Numéricos en Ingeniería (CINME). Barcelona.
- ZIENCKIEWICZ, O.C., TAYLOR, R.L.** 2004. El Método de los Elementos Finitos, Volumen 1. Centro Internacional de Métodos Numéricos en Ingeniería (CINME), 5ª edición. Barcelona.
- ZIENCKIEWICZ, O.C., TAYLOR, R.L., ZHU, J.Z.** 2005. El Método de los Elementos Finitos, Volumen 1. Centro Internacional de Métodos Numéricos en Ingeniería (CINME), 6ª edición. Barcelona.

ANEXO I

Código fuente

Contenido

I. Código fuente.....	78
I.1. Funciones principales	78
I.2. Subfunciones	90

I. Código fuente

A continuación se muestran los códigos de las funciones del programa, empezando por el código de la función principal *MEFBID* y a continuación los códigos de las funciones que en este aparecen.

I.1. Funciones principales

➤ Función *MEFBID*

```
function MEFBID
%función principal del programa
clc
clear
tic,
tiempo=[];

%En primer lugar obtenemos los datos necesarios para la entrada del
%problema, introduciendo el nombre del archivo donde se encuentran.
Problema1

global dim nodos elementos conectividad coor gauss elemento t Tipof
nodof fuerza xcon Nxcon consti poiss Eng TITULO

%Llama a la función que calcula la matriz de rigidez de la estructura
K=KESTRUCT;
tiempo=[tiempo toc];

%Llama a la función que calcula el vector de fuerzas nodales
equivalentes
%de la estructura
F=FESTRUCT;
tiempo=[tiempo toc];

%Llama a la función que resuelve el sistema de ecuaciones  $K \cdot D = F$ 
[Des,RD]=SOLUC(K,F)
tiempo=[tiempo toc];

%llamamos la función que devuelve la tensión
T=TALIS(Des);
tiempo=[tiempo toc];

%llama la función que calcula y dibuja la norma del error
Err=ERROR(T,Des)';
tiempo=[tiempo toc];

%Llama la función que calcula y dibuja la tensión de Von Mises
TVM=TVONM(T);
```

➤ Función de entrada de datos

```

function problema1
%Función que devuelve los datos de entrada para una placa de 2000*2000 con
%un taladro central (simetría de la placa en ejes x e y). Elementos
%triangulares

global dim nodos elementos conectividad coor gauss elemento t Tipof nodof
fuerza xcon Nxcon consti poiss Eng TITULO

TITULO=('problema1');

%coor devuelve las coordenadas de los nodos
%valor x en la primera columna e y en la segunda
coor=[
    0          1.0000
    1.0000     1.0000
    1.0000     0
    0.4000     0
    0          0.4000
    0.2828     0.2828
    0.4348     1.0000
    1.0000     0.4348
    0.6609     0
    0          0.6609
    0.6511     0.6511
    0.5041     0.2068
    0.2068     0.5041
    0.2936     0.7141
    0.7141     0.2936
    0.4464     0.4464]*1000;

%conectividad devuelve la conectividad de los elementos
%nodo 1 en la primera columna, nodo 2 en la segunda y nodo 3 en la tercera
conectividad=[
    6      4     12
    7      1     14
    8      2     11
   10      5     13
    2      7     11
    9      3     15
    5      6     13
   11      7     14
    3      8     15
    1     10     14
    4      9     12
   12      9     15
   13      6     16
   15     11     16
    6     12     16
   10     13     14
    8     11     15
   14     13     16
   11     14     16
   12     15     16];

%dim devuelve el número de grados de libertad de los nodos
dim=2;

```

```

%elemento devuelve el tipo de elemento utilizado
%1 para elemento triangular y 2 para rectangular
elemento=1;

%nodos devuelve el número de nodos
nodos=length(coor);

%elementos devuelve el número de elementos
elementos=length(conectividad);

%gauss devuelve el número de puntos de gauss para integración numérica
gauss=1;

%t indica el espesor de la placa
t=1;

%Tipof es un vector que contiene indicaciones sobre el tipo de fuerza
%exterior
%1 para carga puntual y 2 para carga repartida
Tipof=[2 2];

%nodof en una matriz que contiene los nodos donde se aplica la fuerza
%[N 0]para fuerzas puntuales y [N1 N2] para repartidas
nodof=[3 8;8 2]';

%fuerza es una Matriz que contiene el valor de la fuerza
%[Fx Fy 0 0]para fuerza puntual y [N1 N2 T1 T2]para carga repartida
fuerza=[-500 -500 0 0;-500 -500 0 0];

%xcon es un vector que contiene los grados de libertad con desplazamiento
%conocido
xcon=[1 6 8 9 18 19]';

%Nxcon es un vector que contiene el valor de los desplazamientos conocidos
Nxcon=[0 0 0 0 0 0]';

%consti indica el tipo de problema
%1 para Tensión plana y 2 para Deformación plana
consti=1;

%poiss devuelve el coeficiente de Poisson
poiss=0.3;

%Eng devuelve el modulo de elasticidad
Eng=210*10^3;

```

➤ Función *KESTRUCT*

```

function K = KESTRUCT
%Calcula y ensambla la matriz de rigidez de la estructura

%En primer lugar obtenemos los datos necesarios
global dim nodos elementos conectividad coor elemento

%inicializa la matriz de rigidez de la estructura como matriz de ceros
K=zeros(dim*nodos);

```

```

%iteración en cada elemento
for contel=1:elementos

%Se obtendrá la matriz de rigidez dependiendo del tipo de elemento usado

%Matriz de rigidez de la estructura para elementos triangulares de 3 nodos
if elemento==1
    %indica los nodos que tiene el elemento
    elem=conectividad(contel,:);
    %indica las coordenadas de los nodos del elemento
    coorel=coor(elem,1:dim)
    %Vector de grados de libertad de los nodos
    gradol=[]
    for contdim=1:3
        gradol=[gradol dim*elem(contdim)-(dim-1):dim*elem(contdim)];
    end
    %obtiene la matriz de rigidez a partir de las distintas matrices de
    rigidez ampliadas de los elementos
    K(gradol,gradol)=K(gradol,gradol)+KELEMEN(coorel);
end

%Matriz de rigidez de la estructura para elementos cuadriláteros de 4 nodos
if elemento==2
    %indica los nodos que tiene el elemento
    elem=conectividad(contel,:);
    %indica las coordenadas de los nodos del elemento
    coorel=coor(elem,1:dim);
    %Vector de grados de libertad de los nodos
    gradol=[];
    for contdim=1:4
        gradol=[gradol dim*elem(contdim)-(dim-1):dim*elem(contdim)];
    end
    %obtiene la matriz de rigidez a partir de las distintas matrices de
    rigidez ampliadas de los elementos
    K(gradol,gradol)=K(gradol,gradol)+KELEMEN(coorel);
end
end

```

➤ Función *KELEMEN*

```

function Kel= KELEMEN (coorel)
%Función que calcula la matriz K del elemento

%En primer lugar obtenemos los datos necesarios
global gauss elemento t dim
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;

%llama la función que devuelve los puntos de integración de gauss
[POS1,POS2,PES]=GAUSS (gauss,elemento);
%obtiene la matriz constitutiva
[D]=DCONSTI;

%La matriz de rigidez dependerá del tipo de elemento

%Matriz de rigidez de elemento triangular de 3 nodos
if elemento==1

```

```

%inicializa la matriz de rigidez del elemento
Kel=zeros(3*dim);
%bucle en cada punto de integración
for contg=1:gauss
    %obtiene la función de forma y su derivada en el punto de
    integración del elemento
    [N,DN]=NFORMA(POS1(contg),POS2(contg));
    %obtiene en jacobiano, su determinante y la inversa
    [jacob,Detjac,Invjac]=JACOBI(coorel,DN);
    if Detjac==0
        error('MEFBID, el determinante del jacobiano de un elemento es
0');
    end
    %obtiene la matriz de deformación
    [Be]=BDEFOR(Invjac,DN);
    %obtiene el diferencial de volumen
    [dvolum]=DVOLUM(Detjac,PES,contg);
    %calcula la matriz de rigidez del elemento
    Kel=Kel+Be'*D*Be*dvolum;
end
end
%
%Matriz de rigidez de elemento rectangular de 4 nodos
if elemento==2
    %inicializa la matriz de rigidez del elemento
    Kel=zeros(4*dim);
    %bucle en cada punto de integración
    for contg=1:gauss
        for contg1=1:gauss
            %obtiene la función de forma y su derivada en el punto de
            integración del elemento
            [N,DN]=NFORMA(POS1(contg),POS1(contg1));
            %obtiene en jacobiano, su determinante y la inversa
            [jacob,Detjac,Invjac]=JACOBI(coorel,DN);
            if Detjac==0
                error('MEFBID, el determinante del jacobiano de un elemento
es 0');
            end
            return
        end
        %obtiene la matriz de deformación
        [Be]=BDEFOR(Invjac,DN);
        %obtiene el diferencial de volumen
        [dvolum]=DVOLUM(Detjac,PES,contg,contg1);
        %calcula la matriz de rigidez del elemento
        Kel=Kel+Be'*D*Be*dvolum;
    end
end
end
end

```

➤ Función *FESTRUCT*

```

function F = FESTRUCT
%función que devuelve el vector de fuerzas equivalentes en nudos
%En primer lugar obtenemos los datos necesarios
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon]=datos1;
global dim nodos Tipof nodof fuerza

%se calcula el número de fuerzas exteriores conocidas

```

```

Numf=length(Tipof);
%inicializa el vector de fuerzas nodales
F=zeros(dim*nodos,1);

%bucle sobre cada una de las fuerzas aplicadas
for contf=1:Numf
    %Fuerza puntual
    if Tipof(contf)==1
        %nodo donde se aplica la fuerza i
        nodofi=nodof(1,contf);
        %colocamos la fuerza como Fx y Fy
        fuerzai=fuerza(contf,1:2);

        %Fuerza distribuida en un lado
    elseif Tipof(contf)==2
        %nodos entre los que se aplican la fuerza distribuida i
        nodofi=nodof(1:2,contf);
        %colocamos las fuerzas normales en la columna 1 y las tangenciales
        en la columna 2
        fuerzai=[fuerza(contf,1:2); fuerza(contf,3:4)];
    end
    %llamamos a la función que nos calcula el vector de fuerzas
    equivalentes de 1 fuerza dada
    F=F+FEQUIV(nodofi,fuerzai,Tipof(contf));
end

```

➤ Función *FEQUIV*

```

function F0 =FEQUIV (nodofi,fuerzai,Tipof)
%función que devuelve el vector de fuerzas equivalentes de una fuerza dada

%En primer lugar obtenemos los datos necesarios
global dim nodos coor
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon]=datos1;

%inicializo el vector F
F0=zeros(dim*nodos,1);

%La fuerza equivalente dependerá del tipo de fuerza aplicada
%Fuerza puntual aplicada en el nodo nodof
if Tipof==1
    %Directamente se coloca la fuerza en los grados de libertad
    %correspondientes al nodo
    F0(dim*nodofi-1:dim*nodofi)=fuerzai;

%Fuerza repartida sobre un borde del elemento
else
    %coordenadas y pesos de los puntos de gauss para interpolación lineal
    %de 2 puntos
    [POS1,POS2,PES] = GAUSS(2,2);

    %bucle para los puntos de integración
    for contg=1:2
        %función de forma de elemento lineal de 2 puntos
        N=[(1-POS1(contg))*0.5 (1+POS1(contg))*0.5];
        %derivada de la función de forma
        DN=[-0.5 0.5];
    end
end

```



```

    %diferencial de volumen
    DVolu=PES(contg);
    %intensidad de carga repartida
    PG=N(1)*fuerzai(:,1)+N(2)*fuerzai(:,2);
    %coordenadas de los nodos
    coorF=coor(nodofi,1:dim);
    %jacobiano de los nodos
    jacob=DN*coorF;
    %fuerzas de superficie
    PX=jacob(1)*PG(2)-jacob(2)*PG(1);
    PY=jacob(1)*PG(1)+jacob(2)*PG(2);
    %cálculo de las componentes del vector de fuerza equivalentes
    F0(dim*nodofi(1)-1,1)=F0(dim*nodofi(1)-1,1)+PX*DVolu*N(1);
    F0(dim*nodofi(1),1)=F0(dim*nodofi(1),1)+PY*DVolu*N(1);
    F0(dim*nodofi(2)-1,1)=F0(dim*nodofi(2)-1,1)+PX*DVolu*N(2);
    F0(dim*nodofi(2),1)=F0(dim*nodofi(2),1)+PY*DVolu*N(2);
end
end

```

➤ Función SOLUC

```

function [Des,RD] = SOLUC(K,F)
%función que ensambla y resuelve el sistema de ecuaciones  $K*d=F$ 

%En primer lugar obtenemos los datos necesarios
global dim nodos coor xcon Nxcon

%inicializa el vector gdlnul, que contiene las posiciones donde el
%desplazamiento es conocido y nulo
gdlnul=[];

%desarrollo del vector gdlnul
for contdes=1:length(xcon);
    if Nxcon(contdes)==0
        gdlnul=[gdlnul;xcon(contdes)];
    end
end

%el vector gdlldes contiene los grados de libertad que los desplazamientos
son desconocidos
gdlldes=setdiff(1:dim*nodos,xcon);
%el vector gdlnonul contiene los grados de libertad con desplazamiento no
nulos
gdlnonul=setdiff(1:dim*nodos,gdlnul);
%Calculamos el vector de desplazamientos desconocidos por eliminación
%gaussiana
DDes=K(gdlnonul,gdlldes)\(F(gdlnonul)-K(gdlnonul,xcon)*Nxcon);
%calculamos el vector de fuerzas desconocidas
FD=K(gdlnul,gdlldes)*DDes+K(gdlnul,xcon)*Nxcon;
%calculamos las reacciones en los apoyos
RD=zeros(nodos*dim,1);
RD(gdlnul)=FD-F(gdlnul);
RD=[RD(1:2:nodos*dim) RD(2:2:nodos*dim)];
%inicializo el vector Des que contiene los desplazamientos de los nudos
Des=zeros(2*nodos,1);
%colocamos los desplazamientos conocidos en su posición

```

```
Des(xcon)=Nxcon;
%colocamos los desplazamientos desconocidos calculados en su posición
Des(gdlDes)=DDes;
%colocamos los desplazamientos en x en la columna 1 y los desplazamientos
%en y en la columna 2
Des=[Des(1:2:dim*nodos,1) Des(2:2:dim*nodos,1)];
```

➤ Función *TALIS*

```
function [T]= TALIS (Des)
%Calcula y devuelve la matriz de tensiones de los elementos

%En primer lugar obtenemos los datos necesarios
global dim elementos conectividad coor elemento t consti poiss
gauss=1;
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;

%Matriz constitutiva de Tensión plana
if consti==1
    Nten=3;
%Matriz constitutiva de Deformación plana
elseif consti==2
    Nten=4;
end

%Matriz constitutiva
[D]=DCONSTI;
%inicializa la matriz de tensiones como matriz de ceros
T=zeros(length(coor),Nten);
%inicializa el contador de tensiones en nodos
conT=zeros(1,length(coor));
%calcula de los puntos de integración de gauss
[POS1,POS2,PES]=GAUSS(gauss,elemento);

%bucle sobre cada elemento
for contel=1:elementos

    %indica los nodos que tiene el elemento
    elem=conectividad(contel,:);
    %indica las coordenadas del elemento
    coorel(:,1:dim)=coor(elem,1:dim);
    %obtiene la deformación de los nodos del elemento
    Delem=Des(elem,1:dim);
    %inicializa la matriz de tensiones del elemento
    Te=[];
    %inicializa la matriz de función de forma de los nodos del elemento
    N1=[];

    %Vector de tensión de elemento triangular de 3 nodos
    if elemento==1
        %recoloca la deformación en los nodos
        Delem=[Delem(1,:)';Delem(2,:)';Delem(3,:)'];

        %bucle sobre cada punto de integración
        for contg=1:gauss
            %obtiene la función de forma y su derivada
            [N,DN]=NFORMA(POS1(contg),POS2(contg));
```

```

%obtiene el jacobiano, su determinante y su inversa
[jacob, Detjac, Invjac]=JACOB(coorel, DN);
%obtiene la matriz de deformación
[Be]=BDEFOR(Invjac, DN);
%Calcula y construye la matriz de tensiones en los puntos de
%gauss
Te=[Te D*Be*Delem];

end
%coordenadas de los nudos para la extrapolación

%funcion de forma de los nodos para la extrapolación
if gauss==1
    N1=eye(3);
    Te=[Te Te Te];
elseif gauss==3
    POS12=[-1/3 5/3 -1/3];
    POS22=[-1/3 -1/3 5/3];
    for contg=1:3
        N1=[N1; NFORMA(POS12(contg), POS22(contg))];
    end
end

end

end

%Vector de tensión de elemento cuadrilátero
if elemento==2

    %recoloca la deformación en los nodos
    Delem=[Delem(1,:)'; Delem(2,:)'; Delem(3,:)'; Delem(4,:)']

    %bucle sobre cada punto de integración
    for contg=1:gauss
        for contg1=1:gauss
            %obtiene la función de forma y su derivada
            [N, DN]=NFORMA(POS1(contg), POS1(contg1))
            %obtiene el jacobiano, su determinante y su inversa
            [jacob, Detjac, Invjac]=JACOB(coorel, DN)
            %obtiene la matriz de deformación
            [Be]=BDEFOR(Invjac, DN)
            %Calcula y construye la matriz de tensiones en los puntos
de gauss.
            Te=[Te D*Be*Delem]
        end
    end
end
%función de forma de los nodos para la extrapolación
if gauss==1
    N1=eye(4);
    Te=[Te Te Te Te];
elseif gauss==2
    POS12=[-1 1 -1 -1]/POS1(contg);
    POS22=[-1 -1 1 1]/POS1(contg1);
    for contg=1:4
        N1=[N1; NFORMA(POS12(contg), POS22(contg))];
    end
end
end
%coordenadas de los nudos para la extrapolación

```

```

end

%extrapolación de la matriz de tensiones en los puntos de gauss a
%los nodos
Te=N1*Te';
%Cálculo de la tensión en Z para deformación plana
if Nten==4
    Te=[Te(:,1:2) -poiss*(Te(:,1)+Te(:,2)) Te(:,3)]
end

%Sumamos la matriz de tensiones del elemento a la matriz de
%tensiones en el lugar correcto
T(elem,:)=T(elem,:)+Te;
%Sumamos al contador 1 en los nodos correspondiente
conT(elem)=conT(elem)+1;

end

%alisamos la matriz de tensiones en los nudos, dividiendo el valor de
%esta por el del contador correspondiente al nodo
for contn=1:length(coor)
    T(contn,:)=T(contn,:)/conT(contn);
end

```

➤ Función *ERROR*

```

function Err = ERROR(T,Des)
%función que calcula la norma energética del error cometido

%En primer lugar obtenemos los datos necesarios
global dim nodos elementos conectividad coor gauss elemento t Tipof nodof
fuerza xcon Nxcon consti poiss Eng

%inicializamos el vector de error de elemento
Err=zeros(1,elementos);

%bucle sobre cada elemento para el cálculo del error
for contel=1:elementos
    %obtiene las coordenadas de los puntos de gauss
    [POS1,POS2,PES]=GAUSS(gauss,elemento);
    %calculamos la matriz constitutiva
    [D]=DCONSTI;

    %Calcularemos el error dependiendo del tipo de elemento usado

    %Elemento triangular de 3 nodos
    if elemento==1

        %indica los nodos que tiene el elemento
        elem=conectividad(contel,1:3);
        %indica las coordenadas del elemento
        coorel(1:3,1:dim)=coor(elem,1:dim);
        %indica los desplazamientos de los nodos del elemento

```

```

Delem=Des(elem,1:dim);
%obtiene la matriz de tensión alisada en los nodos del elemento
Ta=T(elem,:);
%llama a la función que calcula la tensión en los nodos del
elemento
Td=TDISC(coorel,Delem);
%Se obtienen las tensiones en x e y y la tensión tangencial para
%elasticidad plana
if consti==2
    Ta=[Ta(1:2,:) ;Ta(4,:)];
    Td=[Td(1:2,:) ;Td(4,:)];
end
%bucle sobre cada punto de la integración numérica
for contg=1:gauss;
    [N,DN]=NFORMA(POS1(contg),POS2(contg));
    [jacob,Detjac,Invjac]=JACOB(coorel,DN);
    [dvolum]=DVOLUM(Detjac,PES,contg);
    Err(contel)=Err(contel)+sqrt((N*(Ta-Td)')*inv(D)*(N*(Ta-
Td)')'*dvolum));
end
end

if elemento==2
%indica los nodos que tiene el elemento
elem=conectividad(contel,1:4);
%indica las coordenadas del elemento
coorel(1:4,1:dim)=coor(elem,1:dim);
%indica los desplazamientos de los nodos del elemento
Delem=Des(elem,1:dim);
%obtiene la matriz de tension alisada en los nudos del elemento
Ta=T(elem,:);
%llama a la funcion que calcula la tension en los nodos del elemento
Td=TDISC(coorel,Delem);
if consti==2
    Ta=[Ta(1:2,:) ;Ta(4,:)];
    Td=[Td(1:2,:) ;Td(4,:)];
end
    for contg=1:gauss
        for contg1=1:gauss
            [N,DN]=NFORMA(POS1(contg),POS1(contg1));
            [jacob,Detjac,Invjac]=JACOB(coorel,DN);
            [dvolum]=DVOLUM(Detjac,PES,contg,contg1);
            Err(contel)=Err(contel)+sqrt(abs((N*(Ta-
Td)')*inv(D)*(N*(Ta-Td)')'*dvolum)));
        end
    end
end
end
end

```

➤ Función *TDISC*

```

function Td = TDISC (coork,Delem)
%función que calcula y devuelve la matriz de tensión discreta en los
%nodos del elemento

%En primer lugar obtenemos los datos necesarios
global nodos elemento t consti poiss
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;

```

```

%Dependiendo del tipo de problema habrá 3 o 4 tensiones en la placa
%Tension plana
if consti==1
    Nten=3;
%Deformación plana
elseif consti==2
    Nten=4;
end
%inicializa la matriz de tensiones
Td=[];

%matriz de tensión discreta de elementos triangulares
if elemento==1
    %coordenadas de los nodos
    POS1=[0 1 0];
    POS2=[0 0 1];
    %Deformación de los nodos
    Delem=[Delem(1,:)';Delem(2,:)';Delem(3,:)'];
    %bucle sobre cada nodo del elemento
    for contg=1:3
        %Función de forma del nodo j
        [N,DN]=NFORMA(POS1(contg),POS2(contg));
        %Jacobiano del nodo
        [jacob,Detjac,Invjac]=JACOB(coork,DN);
        %Matriz de deformación
        [Be]=BDEFOR(Invjac,DN);
        %Matriz constitutiva
        [D]=DCONSTI;
        %Cálculo de la matriz de tensiones
        Td=[Td D*Be*Delem];
    end
end

%Matriz de tensión de elementos cuadriláteros
if elemento==2
    %coordenadas de los nodos
    POS1=[-1 1 1 -1];
    POS2=[-1 -1 1 1];
    %Deformación de los nodos
    Delem=[Delem(1,:)';Delem(2,:)';Delem(3,:)';Delem(4,:)'];
    %Bucle sobre cada nodo del elemento
    for contg=1:4
        %Función de forma del nodo
        [N,DN]=NFORMA(POS1(contg),POS2(contg));
        %Matriz jacobiano del nodo
        [jacob,Detjac,Invjac]=JACOB(coork,DN);
        %Matriz de deformación del nodo
        [Be]=BDEFOR(Invjac,DN);
        %Matriz constitutiva
        [D]=DCONSTI;
        %Cálculo de la matriz de tensiones
        Td=[Td D*Be*Delem];
    end
end

%Cálculo de la tensión en Z en el caso de deformación plana
if Nten==4
    Td=[Td(1:2,:);-poiss*(Td(1,:)+Td(2,:));Td(3,:)];
end

```

➤ Función *TVONM*

```

function [TVM]= TVONM (T)
%Calcula y devuelve la matriz de tensiones de los elementos

%En primer lugar obtenemos los datos necesarios
global conectividad nodos coor consti
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;

%Número de tensiones para Tensión plana
if consti==1
    Nten=3;
%Número de tensiones para Deformación plana
elseif consti==2
    Nten=4;
end

%inicializa la vector de tensiones como vector de ceros
TVM=zeros(nodos,1);

%bucle sobre cada nodo
for contn=1:nodos

    %obtiene la deformación de los nodos del elemento
    Tel=T(contn,1:Nten);

    %Se calcula la tensión de Von Mises dependiendo del número de tensiones
    %del nodo
    %Tensión de Von Mises para Tensión plana
    if Nten==3
        TVM(contn)=sqrt(Tel(1)^2-Tel(1)*Tel(2)+Tel(2)^2+3*Tel(3)^2);
    %Tensión de Von Mises para Deformación plana
    else
        TVM(contn)=sqrt(Tel(1)^2+Tel(2)^2+Tel(3)^2-
        (Tel(1)*Tel(2)+Tel(2)*Tel(3)+Tel(3)*Tel(1))+3*Tel(4)^2);
    end

end
% Dibujamos la Tensión de Von Mises
if consti==2
    figure(7)
else
    figure(6)
end
end

```

I.2. Subfunciones➤ Subfunción *DCONSTI*

```

function D= DCONSTI
%Función que devuelve la matriz constitutiva

%En primer lugar obtenemos los datos necesarios

```

```
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;
global consti poiss Eng

%Matriz constitutiva de Tensión plana
if consti==1
    D=Eng/(1-poiss^2)*[1 poiss 0;poiss 1 0;0 0 (1-poiss)*0.5];
end

%Matriz constitutiva de Deformación plana
if consti==2
    D=Eng/((1+poiss)*(1-2*poiss))*[1-poiss poiss 0;poiss 1-poiss 0;0 0 (1-
2*poiss)*0.5];
End
```

➤ Subfunción GAUSS

```
function [POS1,POS2,PES] = GAUSS(gauss,elemento)
%Pesos y coordenadas de los puntos de integración de la cuadratura de Gauss

%En primer lugar obtenemos los datos necesarios
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;

%Las coordenadas y los pesos de los puntos de gauss dependerán del tipo de
%elemento y del número de puntos

%Para elementos triangulares
if elemento==1
    %Cuadratura de 1 punto
    if gauss==1
        %coordenadas
        POS1=[1/3];
        POS2=[1/3];
        %pesos
        PES=[0.5];
    end
    %Cuadratura de 3 puntos
    if gauss==3
        %coordenadas
        POS1=[1/6 2/3 1/6];
        POS2=[1/6 1/6 2/3];
        %pesos
        PES=[1/6 1/6 1/6];
    end
end

%Para elementos cuadriláteros y lados cargados
if elemento==2
    POS2=0;
    %Cuadratura de 1x1 puntos
    if gauss==1
        %coordenada
        POS1=0.0000000000;
        %pesos
        PES=2.0000000000;
    end
end
```



```

%Cuadratura de 2 o 2x2 puntos
if gauss==2
    %coordenadas
    POS1=[0.5773502692 -0.5773502692];
    %pesos
    PES=[1.0000000000 1.0000000000];
end

end

```

➤ Subfunción *NFORMA*

```

function [N,DN]= NFORMA(E1,E2)
%Función que devuelve la función de forma del elemento

%En primer lugar obtenemos los datos necesarios
global elemento

%Obtenemos la función de forma dependiendo del elemento usado

%Función de forma de elemento triangular de 3 nodos
if elemento==1
    %Funciones de forma del punto
    N1=1-E1-E2;
    N2=E1;
    N3=E2;
    %matriz de la función de forma
    N=[N1 N2 N3];
    %diferencial de la función de forma
    DN=[-1 1 0;-1 0 1];
end

%Función de forma de elemento cuadrilátero de 4 nodos
if elemento==2
    %Funciones de forma del punto
    N1=(1-E1)*(1-E2)*0.25;
    N2=(1+E1)*(1-E2)*0.25;
    N3=(1+E1)*(1+E2)*0.25;
    N4=(1-E1)*(1+E2)*0.25;
    %Matriz de la función de forma
    N=[N1 N2 N3 N4];
    %diferencial de la función de forma
    DN=[-(1-E2) (1-E2) (1+E2) -(1+E2);-(1-E1) -(1+E1) (1+E1) (1-E1)]*0.25;
end

```

➤ Subfunción *JACOB*

```

function [jacob,Detjac,Invjac] = JACOB(coorel,DN)
%Función que devuelve el jacobiano del elemento así como el determinante y
%la inversa de este

%Calcula el Jacobiano del elemento
jacob=DN*coorel;
%Calcula el determinante del jacobiano

```

```
Detjac=det(jacob);
%Calcula la inversa del jacobiano
Invjac=inv(jacob);
```

➤ Subfunción *DVOLUM*

```
function [dvolum]=DVOLUM (Detjac,PES,contg,contg1)
%Función que devuelve el diferencial de volumen

%En primer lugar obtenemos los datos necesarios
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;
global elemento t

%El diferencial de volumen se calcula dependiendo del elemento usado para
%la división en elementos finitos

%Diferencial de volumen de elemento triangular de 3 nodos
if elemento==1
    dvolum=Detjac*PES(contg)*t;
end

%Diferencial de volumen de elemento cuadrilátero de 4 nodos
if elemento==2
    dvolum=Detjac*PES(contg)*PES(contg1)*t;
end
```

➤ Subfunción *BDEFOR*

```
function Be =BDEFOR (Invjac,DN)
%Función que devuelve la matriz de deformación del elemento donde i indica
%los nodos de la estructura que ocupa el elemento

%En primer lugar obtenemos los datos necesarios
%[dim,nodos,elementos,conectividad,coor,gauss,elemento,t,Tipof,nodof,fuerza
,xcon,Nxcon,consti,poiss,E]=datos1;
global elemento

%Matriz unitaria ampliada
I=[1 0 0 0;0 0 0 1;0 1 1 0];
%Matriz inversa del jacobiano ampliada
Invamp=[Invjac zeros(2);zeros(2) Invjac];

%Matriz de deformación de elemento triangular de 3 nodos
if elemento==1
    %matriz ampliada de la derivada de la función de forma
    DNamp=[DN(1:2,1) zeros(2,1) DN(1:2,2) zeros(2,1) DN(1:2,3) zeros(2,1);
           zeros(2,1) DN(1:2,1) zeros(2,1) DN(1:2,2) zeros(2,1) DN(1:2,3)];
    %calculo de la matriz de deformacion
    Be=I*Invamp*DNamp;
end

%Matriz de deformación de elemento rectangular de 4 nodos
if elemento==2
    %matriz ampliada de la derivada de la función de forma
```

```
DNamp=[DN(1:2,1) zeros(2,1) DN(1:2,2) zeros(2,1) DN(1:2,3) zeros(2,1)
DN(1:2,4) zeros(2,1);
        zeros(2,1) DN(1:2,1) zeros(2,1) DN(1:2,2) zeros(2,1) DN(1:2,3)
zeros(2,1) DN(1:2,4)];
        %cálculo de la matriz de deformación
        Be=I*Invamp*DNamp;
end
```