

# **67.58 INTRODUCCIÓN AL MÉTODO DE LOS ELEMENTOS FINITOS**

## **TRABAJO PRÁCTICO 7 (ESPECIAL)**

**SIMULACIÓN NUMÉRICA DE LA DISTRIBUCIÓN DE TEMPERATURAS EN UN INTERCAMBIADOR  
DE CALOR**



**Estudiante**

**FREDY ANDRÉS MERCADO NAVARRO**

**Pasaporte: 98'773.532**

**Maestría en Simulación Numérica y Control**

**Cuatrimestre: II-2011**

**15 de Marzo**

**Universidad de Buenos Aires  
Ciudad Autónoma de Buenos Aires  
Argentina  
2012**

**INDICE DE CONTENIDOS**

<b>1. INTRODUCCIÓN .....</b>	<b>4</b>
<b>2. PLANTEAMIENTO DEL PROBLEMA .....</b>	<b>5</b>
<b>3. DISCRETIZACIÓN POR EL MÉTODO DE ELEMENTOS FINITOS .....</b>	<b>6</b>
3.1    FORMULACIÓN PARA ELEMENTOS TRIANGULARES DE TRES NODOS.....	6
3.2    FORMULACIÓN PARA ELEMENTOS CUADRANGULARES DE CUATRO NODOS.....	7
<b>4. INTEGRACIÓN NUMÉRICA .....</b>	<b>9</b>
<b>5. DISCRETIZACIÓN DEL DOMINIO CON GMSH .....</b>	<b>10</b>
5.1    MALLA TRIANGULAR NO ESTRUCTURADA .....	10
5.2    MALLA TRIANGULAR ESTRUCTURADA .....	11
5.3    MALLA CUADRANGULAR NO ESTRUCTURADA .....	12
5.4    MALLA CUADRANGULAR ESTRUCTURADA .....	12
<b>6. IMPLEMENTACIÓN .....</b>	<b>14</b>
6.1    PRE-PROCESAMIENTO .....	14
6.2    PROCESAMIENTO .....	14
6.3    POST-PROCESAMIENTO .....	14
<b>7. RESULTADOS Y ANÁLISIS .....</b>	<b>15</b>
7.1    SOLUCIÓN CON MALLAS ESTRUCTURADAS Y NO ESTRUCTURADAS .....	15
7.2    SOLUCIÓN CON VARIACIÓN DE LA CONDUCTIVIDAD DE LA ZONA <b>C2</b> .....	15
7.3    SOLUCIÓN CON VARIACIÓN DEL TAMAÑO DE LA MALLA .....	15
7.4    ANÁLISIS DE RESULTADOS.....	15
7.4.1 <i>Malla triangular – No estructurada</i> .....	16
7.4.2 <i>Malla triangular – Estructurada</i> .....	16
7.4.3 <i>Malla cuadrangular – No estructurada</i> .....	17
7.4.4 <i>Malla cuadrangular - Estructurada</i> .....	17
7.4.5 <i>Malla triangular vs. Malla cuadrangular</i> .....	18
7.4.6 <i>Malla estructurada vs. Malla no estructurada</i> .....	18
7.4.7 <i>Comportamiento de la solución con la variación de C2</i> .....	19
7.4.8 <i>Malla gruesa vs. Malla fina</i> .....	21
<b>8. CONCLUSIONES GENERALES Y OBSERVACIONES.....</b>	<b>25</b>

## INDICE DE TABLAS

TABLA 1. PESOS Y PUNTOS DE GAUSS PARA INTEGRACIÓN DE 2x2 .....	9
TABLA 2. DIFERENCIA DE SOLUCIONES ENTRE MALLAS ESTRUCTURADAS Y NO ESTRUCTURADAS. $Lc \sim 0.25$ .....	19
TABLA 3. PORCENTAJE DE DIFERENCIA ENTRE FORMULACIONES PARA DIFERENTE CONDUCTIVIDAD <b>C2</b> .....	21
TABLA 4. COMPORTAMIENTO DE LA SOLUCIÓN CON EL CAMBIO DEL TAMAÑO DE LA MALLA.....	23
TABLA 5. VARIACIÓN DE ALGUNOS PARÁMETROS CON EL CAMBIO DEL TAMAÑO DE LA MALLA.....	24

## INDICE DE FIGURAS

FIGURA 1. DOMINIO DEL PROBLEMA. ....	5
FIGURA 2. CONVENCIÓN PARA NUMERACIÓN DE NODOS LOCALES. ....	7
FIGURA 3. CONVENCIÓN PARA NUMERACIÓN DE NODOS LOCALES. ....	8
FIGURA 4. GEOMETRÍA PARA MALLAS NO ESTRUCTURADAS.....	10
FIGURA 5. MALLA TRIANGULAR NO ESTRUCTURADA. ....	11
FIGURA 6. GEOMETRÍA PARA MALLAS ESTRUCTURADAS.....	11
FIGURA 7. MALLA TRIANGULAR ESTRUCTURADA. ....	12
FIGURA 8. MALLA CUADRANGULAR NO ESTRUCTURADA. ....	12
FIGURA 9. MALLA CUADRANGULAR ESTRUCTURADA. ....	13
FIGURA 10. SOLUCIÓN CON MALLA TRIANGULAR NO ESTRUCTURADA.....	16
FIGURA 11. SOLUCIÓN CON MALLA TRIANGULAR ESTRUCTURADA.....	16
FIGURA 12. SOLUCIÓN CON MALLA CUADRANGULAR NO ESTRUCTURADA. ....	17
FIGURA 13. SOLUCIÓN CON MALLA CUADRANGULAR ESTRUCTURADA. ....	17
FIGURA 14. NODOS COMUNES A TODAS LAS DISCRETIZACIONES. ....	18
FIGURA 15. SOLUCIONES CON MALLA TRIANGULAR ESTRUCTURADA CON VARIACIÓN DE LA CONDUCTIVIDAD <b>C2</b> .....	19
FIGURA 16. PERFILES DE TEMPERATURA SOBRE LÍNEA DE SIMETRÍA.....	20
FIGURA 17. SOLUCIONES CON MALLA CUADRANGULAR ESTRUCTURADA CON VARIACIÓN DE LA CONDUCTIVIDAD <b>C2</b> .....	20
FIGURA 18. SOLUCIONES CON VARIACIÓN DEL TAMAÑO DE LA MALLA. ....	22
FIGURA 19. SOLUCIÓN VS. TAMAÑO DE MALLA PARA MALLA TRIANGULAR NO ESTRUCTURADA. ....	23
FIGURA 20. TAMAÑO DE MALLA VS. TIEMPO DE CÁLCULO DE CICLOS ELEMENTALES PARA MALLA TRIANGULAR.....	24

## 1. INTRODUCCIÓN

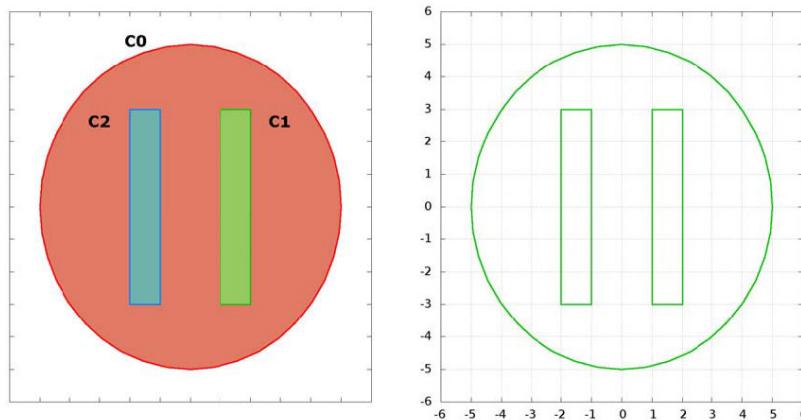
Durante el desarrollo de este trabajo se solucionará numéricamente el problema de la distribución de temperaturas en un intercambiador de calor bidimensional empleando la formulación del Método de los Elementos Finitos. El problema se resolverá en régimen estacionario teniendo en cuenta solo la transferencia de calor por difusión (conducción).

Para lograr este objetivo se discretizará el dominio bidimensional del intercambiador de calor en elementos triangulares y cuadrangulares mediante el software libre GMSH y se desarrollará un código computacional en Matlab.

Se analizará posteriormente la influencia de las formulaciones triangular y cuadrangular sobre las soluciones y también de factores como la estructura de la malla, el tamaño, los valores de conductividad al interior del dominio entre otros. Por último, como valor agregado, se analizarán algunos datos que comprueban la consistencia del método de los Elementos Finitos y la variación de los tiempos de cálculo con el tamaño de la malla.

## 2. PLANTEAMIENTO DEL PROBLEMA

$C_1$  y  $C_2$  son dos conductores térmicos dentro de un recinto  $C_0$ . El primero de ellos se mantiene a una temperatura constante de  $T_1 = 100 \text{ }^{\circ}\text{C}$  y el otro tiene una conductividad térmica  $k_2$  que resulta 5 veces mayor que  $k_0$ . Se asume que la temperatura en la frontera de  $C_0$  es  $T_0 = 20 \text{ }^{\circ}\text{C}$ .



**Figura 1.** Dominio del problema.

El objetivo del problema es conocer  $T(x, y)$  en el recinto  $\Omega$ , donde  $\Omega$  es el interior de  $C_0$ , sin tener en cuenta a  $C_1$ . Los puntos a desarrollar a lo largo del trabajo son los siguientes:

- Discretizar el problema utilizando el método de los Elementos Finitos.
- Plantear la discretización para el dominio presentado utilizando el generador de mallas GMSH. Resolverlo para mallas estructuradas y no estructuradas utilizando elementos triangulares (de tres nodos) y cuadrangulares (de cuatro nodos). Implementar la solución en un código computacional.
- Obtener y graficar la distribución de temperaturas para los cuatro tipos de resolución.
- Resolver el problema con el código numérico evaluando la sensibilidad del problema al valor de la conductividad térmica de  $C_2$ .
- Para una de las discretizaciones, remallar para obtener mallas de distintas precisiones y utilizar el código ya desarrollado para obtener nuevas soluciones.
- Resolver el mismo problema pero considerando que  $C_2$  es adiabático, es decir, imponiendo como condición de borde, en la frontera de  $C_2$ , que el gradiente normal de temperatura es igual a cero (condición de Neumann homogénea).

### 3. DISCRETIZACIÓN POR EL MÉTODO DE ELEMENTOS FINITOS

En esta sección se incluirá la formulación matemática del método de los Elementos Finitos aplicada a un problema de conducción (difusión) de calor bidimensional, lineal y en estado estacionario.

La ecuación a resolver por Elementos Finitos es:

$$K^k \hat{T} = Q$$

Donde  $K^k$  es la matriz de conductividad (también llamada matriz de coeficientes o matriz de rigidez),  $\hat{T}$  es el vector de temperaturas nodales y  $Q$  es el vector de calores nodales externos. Los términos de la ecuación que aplican tanto para elementos triangulares como cuadrangulares están definidos de la siguiente manera:

$$K^k = \sum_m \int_{V^m} B^{(m)T} k^{(m)} B^{(m)} dV^{(m)}$$

$$\hat{T} = [\hat{T}_1 \quad \hat{T}_2 \quad \dots \quad \hat{T}_n]^T$$

$$Q = [Q_1 \quad Q_2 \quad \dots \quad Q_n]^T$$

Siendo  $n$  el número de nodos totales de la discretización.

$$k^{(m)} = \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix}, \quad dV = \det(J) dr ds$$

Para la matriz  $k$  las conductividades son iguales no importa la dirección (material isotrópico). Por ello:

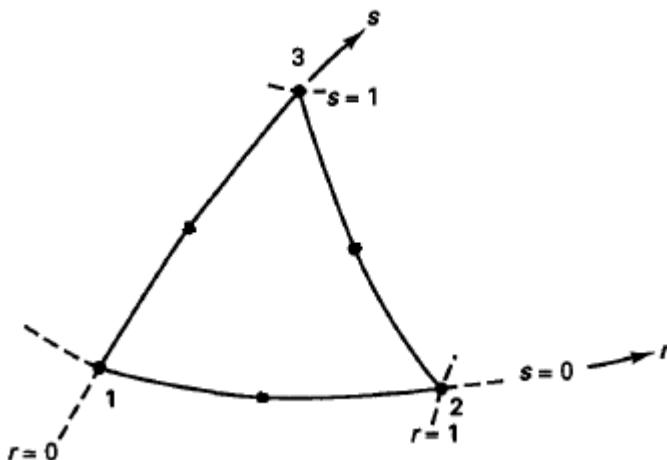
$$k^{(m)} = \begin{bmatrix} k & 0 \\ 0 & k \end{bmatrix}$$

#### 3.1 Formulación para elementos triangulares de tres nodos

A continuación se desarrollará en forma general, la formulación para el método de Elementos Finitos cuyo dominio ha sido discretizado como una malla de elementos triangulares. Las funciones de forma estarán expresadas en términos de las coordenadas naturales  $r$  y  $s$ , las cuales estarán definidas  $0 \leq r \leq 1$  y  $0 \leq s \leq 1$ .

$$J = \begin{bmatrix} \frac{\partial h_1}{\partial r} & \frac{\partial h_2}{\partial r} & \frac{\partial h_3}{\partial r} \\ \frac{\partial h_1}{\partial s} & \frac{\partial h_2}{\partial s} & \frac{\partial h_3}{\partial s} \end{bmatrix} \cdot \begin{bmatrix} \hat{x}_1 & \hat{y}_1 \\ \hat{x}_2 & \hat{y}_2 \\ \hat{x}_3 & \hat{y}_3 \end{bmatrix}, \quad B = \nabla H = J^{-1} \cdot \begin{bmatrix} \frac{\partial h_1}{\partial r} & \frac{\partial h_2}{\partial r} & \frac{\partial h_3}{\partial r} \\ \frac{\partial h_1}{\partial s} & \frac{\partial h_2}{\partial s} & \frac{\partial h_3}{\partial s} \end{bmatrix}$$

Dado que todos los elementos son triángulos con 1 nodo en cada vértice, las funciones de forma para todos los elementos son las mismas. La numeración de los nodos se desarrolló en base a la Figura 2:



**Figura 2.** Convención para numeración de nodos locales.

Las funciones de forma y sus derivadas están definidas como sigue:

Para $h_1$	Para $h_2$	Para $h_3$
$h_1 = 1 - r - s$ $\frac{\partial h_1}{\partial r} = -1$ $\frac{\partial h_1}{\partial s} = -1$	$h_2 = r$ $\frac{\partial h_2}{\partial r} = 1$ $\frac{\partial h_2}{\partial s} = 0$	$h_3 = s$ $\frac{\partial h_3}{\partial r} = 0$ $\frac{\partial h_3}{\partial s} = 1$

### 3.2 Formulación para elementos cuadrangulares de cuatro nodos

La formulación para los elementos cuadrangulares es muy similar a la formulación triangular. Solo se debe tener en cuenta que el número de nodos cambia de 3 a 4, lo que implica que se deben reformular las funciones de forma. El resultado es:

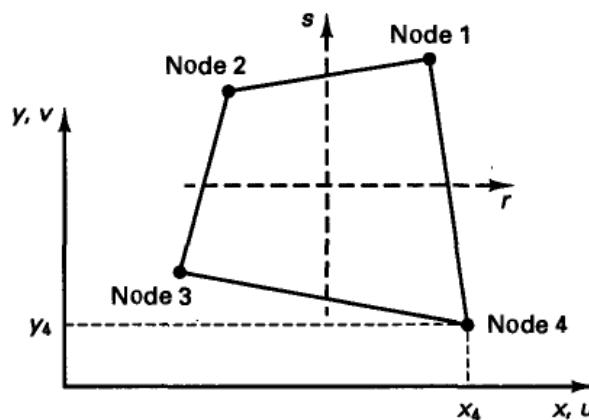
#### Para el caso cuadrilátero

$$J = \begin{bmatrix} \frac{\partial h_1}{\partial r} & \frac{\partial h_2}{\partial r} & \frac{\partial h_3}{\partial r} & \frac{\partial h_4}{\partial r} \\ \frac{\partial h_1}{\partial s} & \frac{\partial h_2}{\partial s} & \frac{\partial h_3}{\partial s} & \frac{\partial h_4}{\partial s} \end{bmatrix} \cdot \begin{bmatrix} \hat{x}_1 & \hat{y}_1 \\ \hat{x}_2 & \hat{y}_2 \\ \hat{x}_3 & \hat{y}_3 \\ \hat{x}_4 & \hat{y}_4 \end{bmatrix}, \quad B = \nabla H = J^{-1} \cdot \begin{bmatrix} \frac{\partial h_1}{\partial r} & \frac{\partial h_2}{\partial r} & \frac{\partial h_3}{\partial r} & \frac{\partial h_4}{\partial r} \\ \frac{\partial h_1}{\partial h_1} & \frac{\partial h_2}{\partial h_2} & \frac{\partial h_3}{\partial h_3} & \frac{\partial h_4}{\partial h_4} \\ \frac{\partial h_1}{\partial s} & \frac{\partial h_2}{\partial s} & \frac{\partial h_3}{\partial s} & \frac{\partial h_4}{\partial s} \end{bmatrix}$$

Para la matriz  $k$  las conductividades son iguales no importa la dirección. Por ello:

$$k^{(m)} = \begin{bmatrix} k & 0 \\ 0 & k \end{bmatrix}$$

Dado que todos los elementos son cuadriláteros con 1 nodo en cada vértice, las funciones de forma para los elementos son las mismas. La numeración de los nodos se desarrolló en base a la Figura 3:



**Figura 3.** Convención para numeración de nodos locales.

Las funciones de forma y sus derivadas están definidas como sigue:

Para $h_1$	Para $h_2$
$h_1 = \frac{1}{4}(1+r)(1+s)$ $\frac{\partial h_1}{\partial r} = \frac{1}{4}(1+s)$ $\frac{\partial h_1}{\partial s} = \frac{1}{4}(1+r)$	$h_2 = \frac{1}{4}(1-r)(1+s)$ $\frac{\partial h_2}{\partial r} = -\frac{1}{4}(1+s)$ $\frac{\partial h_2}{\partial s} = \frac{1}{4}(1-r)$
Para $h_3$	Para $h_4$
$h_3 = \frac{1}{4}(1-r)(1-s)$ $\frac{\partial h_3}{\partial r} = -\frac{1}{4}(1-s)$ $\frac{\partial h_3}{\partial s} = -\frac{1}{4}(1-r)$	$h_4 = \frac{1}{4}(1+r)(1-s)$ $\frac{\partial h_4}{\partial r} = \frac{1}{4}(1-s)$ $\frac{\partial h_4}{\partial s} = -\frac{1}{4}(1+r)$

#### 4. INTEGRACIÓN NUMÉRICA

##### **Para elementos triangulares**

El orden del polinomio de las funciones de forma de los elementos triangulares es 1, de modo que todas sus derivadas con respecto de las coordenadas  $r$  y  $s$  son constantes y el producto de matrices que conforman a la función  $F$  por el determinante del Jacobiano resulta en una matriz de términos constantes de 3x3. Esto implica que:

$$\int_0^1 \int_0^1 F(C) drds = F(C)$$

Donde:

$$F(C) = B^T K^k B \det(J)$$

Por último no sobra recordar que este es un caso especial que aplica para un elemento triangular de 3 nodos y con funciones de forma definidas entre 0 y 1. Esto último implica que el elemento finito utilizado para la solución con elementos triangulares no es isoparamétrico.

##### **Para elementos cuadrangulares**

Se empleó la metodología de Gauss-Legendre para la integración numérica de cada elemento. La definimos como:

$$\int_{-1}^1 \int_{-1}^1 F(r, s) drds = \sum_{i,j} \alpha_i \alpha_j F(r_i, s_j)$$

Donde:

$\alpha_i, \alpha_j$ : pesos que corresponden a cada evaluación de  $r_i$  y  $s_j$ , respectivamente.

$r_i, s_j$ : coordenadas naturales de los puntos de Gauss donde se evaluará la función  $F$  para la integración numérica.

Se empleó el método de Gauss de 2x2 puntos de integración para el elemento cuadrilátero de 4 nodos debido a que dos evaluaciones por coordenada son suficientes para integrar en forma exacta un polinomio de grado 3 o menor, el cual es el caso. La teoría que soporta eso se puede consultar en la referencia [1].

Teniendo en cuenta lo anterior, los pesos y las coordenadas de los puntos de Gauss están ilustrados en la Tabla 1.

**Tabla 1.** Pesos y Puntos de Gauss para integración de 2x2

<b><math>n</math></b>	<b><math>r_i</math> y <math>s_i</math></b>	<b><math>\alpha_i</math> y <math>\alpha_j</math></b>
<b>2</b>	$1/\sqrt{3}$	1
	$-1/\sqrt{3}$	1

## 5. DISCRETIZACIÓN DEL DOMINIO CON GMSH

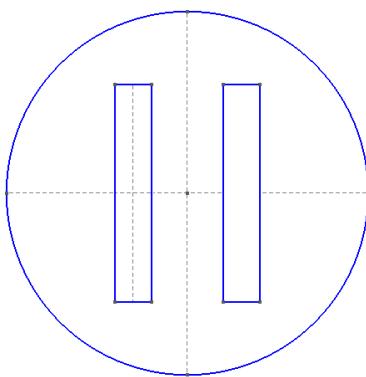
El dominio bidimensional del intercambiador de calor será discretizado por medio del software libre GMSH.

El primer paso para la generación de los mallados es la creación de la geometría. A partir de la geometría se genera la malla. Para todos los mallados se utilizó una longitud característica aplicada a los puntos igual a 1. Posteriormente, a través del comando Refine by Splitting, en el entorno Mesh, refinamos la malla hasta el tamaño deseado.

Luego de refinar las mallas modificando la longitud característica, se descubrió que el uso del comando Refine by Splitting conserva los patrones de la malla original, ya que crea los nuevos elementos dividiendo en dos cada lado del elemento, mientras que, generar la malla desde un inicio con una longitud característica pequeña resultó en que las mallas no estructuradas eran creadas con un nivel de aleatoriedad mayor, por lo cual fue preferible asignar una  $Lc=1$  y refinar automáticamente el número de veces deseado.

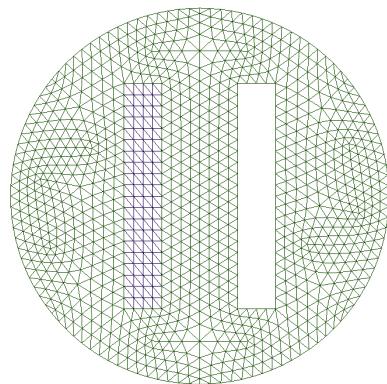
### 5.1 Malla triangular no estructurada

La Figura 4 ilustra la geometría utilizada para las mallas no estructuradas.



**Figura 4.** Geometría para mallas no estructuradas.

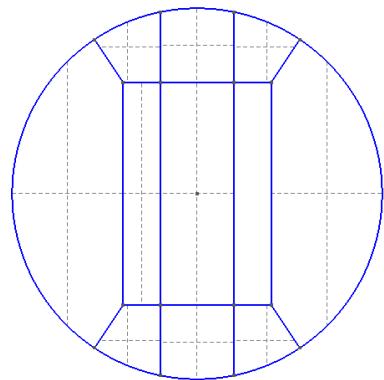
Se introdujo como parámetro una longitud característica nodal de 1. Antes de mallar, seleccionamos Tools>Options>Subdivision algorithm>None. De esta manera, la malla a generar será triangular y no estructurada por defecto. Luego, teniendo la geometría lista, en el módulo Mesh seleccionamos la opción 2D. El resultado es una malla triangular no estructurada de elementos con  $Lc=1$ , ver Figura 18a. Mientras más pequeños sean los elementos mejor será la aproximación de la solución. Por ello, refinamos dos veces, logrando una malla satisfactoria con una  $Lc=0.25$ . El resultado del mallado se ilustra en la Figura 5.



**Figura 5.** Malla triangular no estructurada.

## 5.2 Malla triangular estructurada

Para las mallas estructuradas, la cantidad de nodos sobre cada línea fue especificada en el algoritmo del GMSH, de modo que todas las mallas iniciales son producto de un trabajo de discretización artesanal. La malla original no fue refinada dado que el tamaño de los elementos obtenido equivale a una  $L_c \sim 0.25$ . La geometría utilizada para los mallados estructurados se ilustra en la Figura 6.

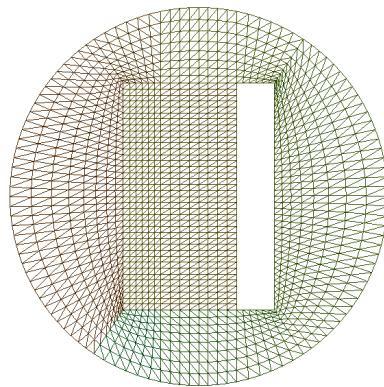


**Figura 6.** Geometría para mallas estructuradas.

Para generar la malla estructurada buscamos generar una distribución de nodos más fina cerca de las zonas  $C_1$  y  $C_2$ . Ver Figura 1. Para lograr este propósito se utilizó la siguiente línea de código:

```
Transfinite Line {17, 18, 19, 20, 22, 23, 24, 25} = 10 Using Progression 1.2;
```

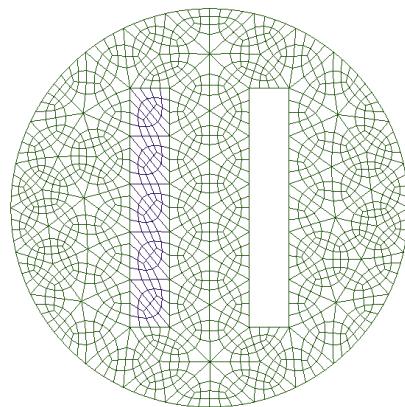
Donde los números entre corchetes indican el número de las líneas, el número 10 la cantidad de nodos sobre las líneas y los números al final de los códigos indican el valor de la progresión o la cantidad en que crece el espacio entre nodos a medida que crece la distancia desde el punto inicial. El resultado final de la malla triangular estructurada se ilustra en la Figura 7. Nótese que la distancia entre nodos aumenta a medida que los nodos se alejan de las zonas  $C_1$  y  $C_2$ , a excepción de la zona ubicada en medio de ellas.



**Figura 7.** Malla triangular estructurada.

### 5.3 Malla cuadrangular no estructurada

La longitud característica seleccionada es  $Lc=1$ . Para lograr el mallado cuadrangular no estructurado se modificó en GMSH la siguiente secuencia: Options>Mesh>General>Subdivision algorithm>All Quads. De este modo los elementos serán elementos cuadrangulares. Utilizando la misma geometría de la Figura 4 y ejecutando el comando 2D en el módulo Mesh obtenemos la malla ilustrada en la Figura 8.



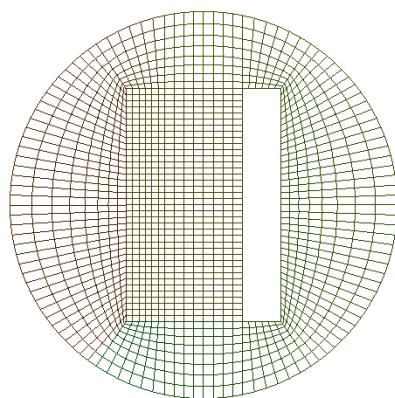
**Figura 8.** Malla cuadrangular no estructurada.

### 5.4 Malla cuadrangular estructurada

El proceso para la generación de la malla cuadrangular estructurada sigue el mismo algoritmo que para la malla triangular estructurada. La diferencia radica en incluir en el código de la geometría la siguiente línea:

```
Recombine Surface {36, 38, 40, 42, 28, 30, 32, 44, 46, 34};
```

Esto último produce la combinación de 2 elementos triangulares en un solo elemento cuadrangular. El resultado del mallado se observa en la Figura 9.



**Figura 9.** Malla cuadrangular estructurada.

Nótese que la malla cuadrangular estructurada es la malla triangular estructurada sin las líneas diagonales en cada elemento cuadrilateral.

## 6. IMPLEMENTACIÓN

La solución numérica del problema fue desarrollada a través de la implementación de un código computacional desarrollado en el programa *Matlab*. Además se utilizó la aplicación GMSH para el pre- y post-procesamiento.

### 6.1 Pre-procesamiento

El mismo se realizó utilizando la aplicación GMSH (<http://geuz.org/gmsh/>). A través de su interfaz gráfica se dibujó la geometría y se elaboraron las mallas para cada caso a analizar. Se produce un archivo en formato ASCII con extensión .msh, el cual contiene la información de las coordenadas de cada nodo e información de la conectividad entre los elementos.

### 6.2 Procesamiento

Los datos suministrados por el archivo ASCII son separados en tres matrices. Una matriz de coordenadas nodales, seguida por una matriz de nodos con condiciones de borde definidas y una matriz de conectividad.

Posterior a la lectura de los datos se procede a procesarlos de tal forma que podamos aplicar las ecuaciones de elementos finitos para la solución numérica del problema. Para ello, se resolvió un sistema de ecuaciones de  $N \times N$ , donde  $N$  es el número total de nodos de la malla. Tenemos entonces un sistema de ecuaciones lineales de la forma,

$$\mathbf{K}\hat{\mathbf{T}} = \mathbf{b}$$

donde  $\mathbf{K} [N \times N]$  es la matriz global de coeficientes,  $\hat{\mathbf{T}} [N \times 1]$  es el vector de incógnitas (temperaturas nodales) y  $\mathbf{b} [N \times 1]$  es un vector que contiene información de flujos de calor calculados a partir de las temperaturas especificadas como condiciones de borde  $C_0$  y  $C_1$ . La matriz global de coeficientes se obtiene aplicando la formulación a cada elemento. Luego, dentro del ciclo elemental cada matriz se suma a la matriz global con la información otorgada por la matriz de conectividad. Posterior a esto se introducen las condiciones de borde, donde se modifican las matrices  $\mathbf{K}$  y  $\mathbf{b}$ .

Dado que nuestro problema es lineal, Matlab utiliza uno de sus algoritmos de solución para hallar  $\hat{\mathbf{T}} = \mathbf{K} \backslash \mathbf{b}$  (en lenguaje de Matlab).

### 6.3 Post-procesamiento

El código de Matlab genera un archivo en formato ASCII con las temperaturas de cada nodo. En este punto se utiliza el módulo Post-processing de GMSH para graficar la distribución de temperaturas. Dicho módulo utiliza las temperaturas nodales para graficar la distribución de temperaturas sobre todo el dominio interpolando (linealmente en este caso) los valores interiores de cada elemento a partir de sus valores nodales.

## 7. RESULTADOS Y ANÁLISIS

En los primeros numerales se introducirán los resultados de la simulación numérica. Por último se analizarán los resultados respectivos para cada caso.

### 7.1 Solución con mallas estructuradas y no estructuradas

En esta sección se ilustran los resultados de la simulación numérica con los cuatro mallados solicitados por el planteamiento del problema. Se especifican el tipo, número de elementos, y número de nodos totales. El valor de la conductividad térmica de  $C_2$  permaneció constante siendo 5 veces el valor de la conductividad térmica  $C_0$ .

La distribución de temperaturas para cada tipo de mallado se ilustra en la Figura 10, Figura 11, Figura 12 y Figura 13.

### 7.2 Solución con variación de la conductividad de la zona $C_2$

La Figura 15 y Figura 17 ilustran la distribución de temperaturas para cuatro valores de conductividad de la zona  $C_2$ . Los valores seleccionados son 10, 1 y 0.1 veces la conductividad de  $C_0$ . Solo se incluyen resultados para las mallas estructuradas debido a que los resultados para las mallas no estructuradas son muy similares.

### 7.3 Solución con variación del tamaño de la malla

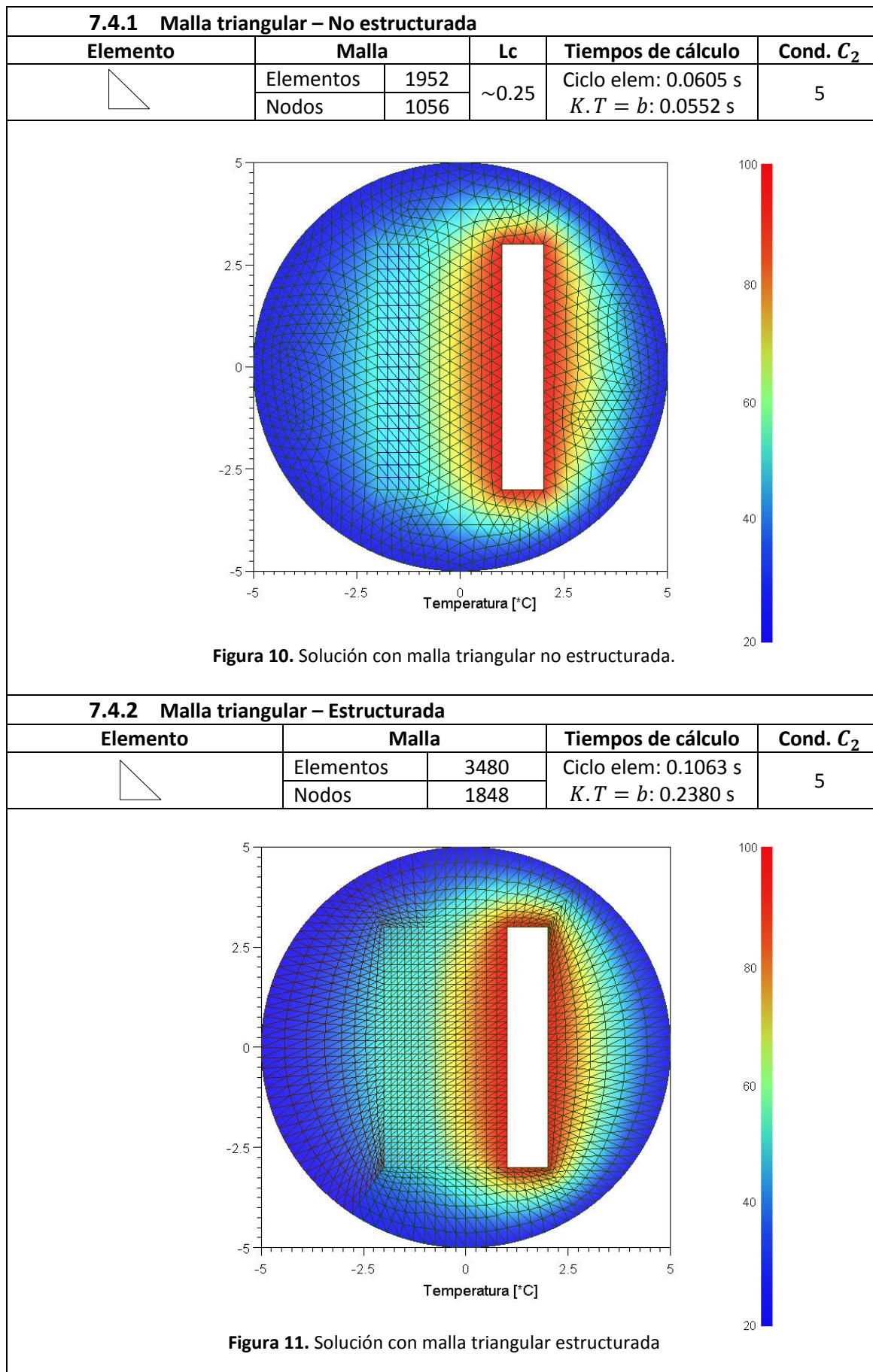
La Figura 18 muestra la comparación entre los resultados de 4 mallados de elementos triangulares de primer orden. Se inició con una malla con longitud característica aplicada en los nodos igual a 1. Luego se refinó mediante el comando Refine by Splitting, el cual divide en dos cada lado para formar nuevos elementos. Así, el lado de un elemento de la malla más fina mide aproximadamente 1/8 del lado de los elementos de la discretización inicial.

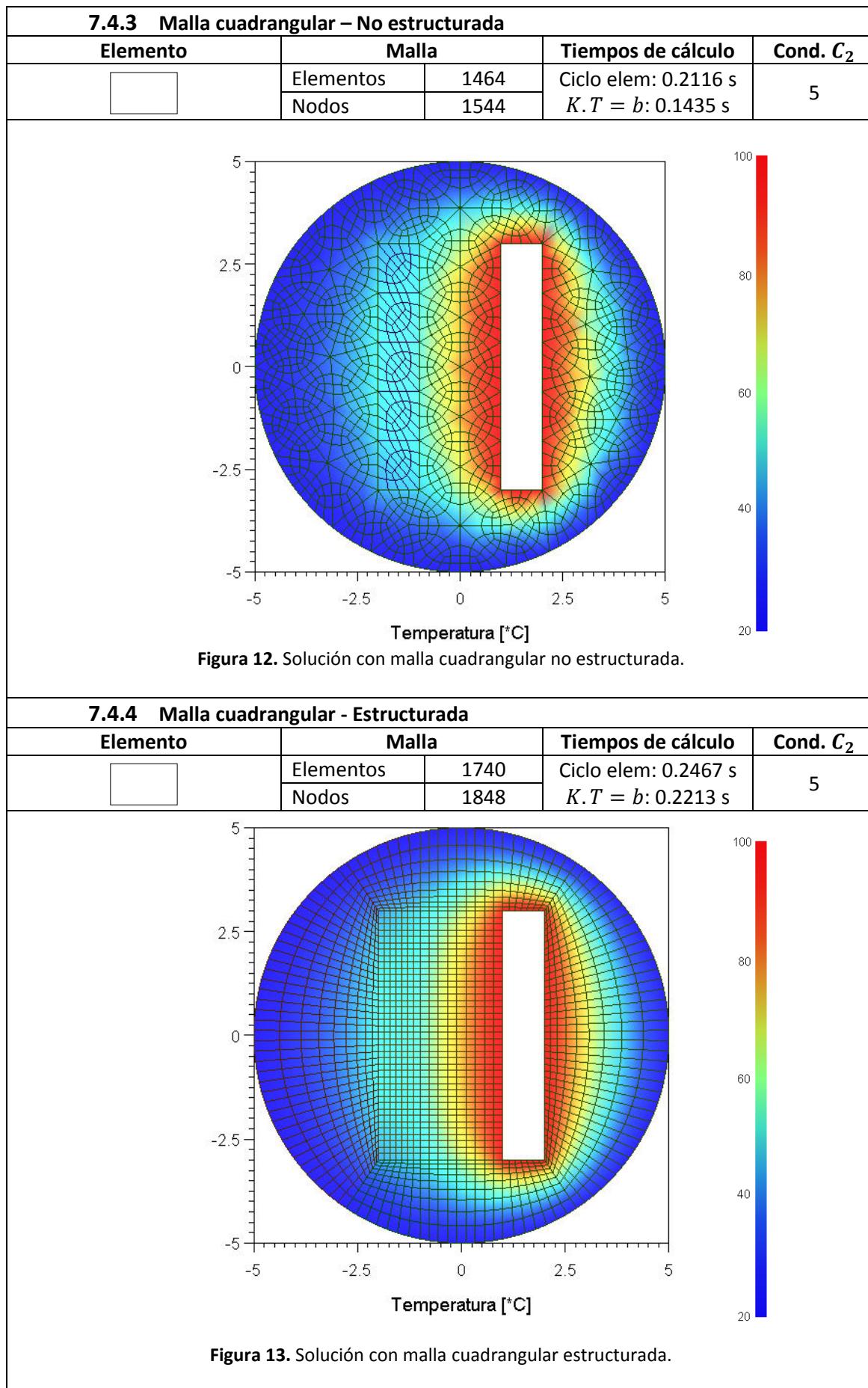
Las figuras muestran a la izquierda la discretización y a la derecha la distribución de temperaturas. Se grafican tonos rojos para las temperaturas altas y azules para las bajas.

### 7.4 Análisis de Resultados

Para comparar los resultados disponemos de las mismas soluciones halladas a través del método de Volúmenes Finitos. Además de la semejanza de los resultados de ambas formulaciones, emplearemos los siguientes criterios para justificar los resultados:

- Ninguna de las temperaturas nódales supera los 100°C ni desciende por debajo de 20°C (condiciones de Dirichlet del problema).
- Los gráficos de temperatura muestran gradientes de temperatura lógicos. Las temperaturas descienden a medida que los nodos se alejan de la condición de borde más caliente. Las pendientes nunca se hacen positivas, siempre son negativas.
- Los resultados, al igual que la geometría del intercambiador, son simétricos. Numéricamente existen diferencias, pero éstas disminuyen con el refinamiento de la malla.





- d) En la zona con conductividad  $C_2$  las temperaturas descienden con una pendiente menor. Esto obedece la física del problema visto desde el punto de vista de la ley de Fourier, donde a mayor conductividad, menor debe ser la diferencia de temperaturas para conservar el mismo flujo de calor. Esto se ilustrará mejor más adelante.
- e) La solución del problema por la discretización triangular y cuadrangular otorgan resultados muy similares.

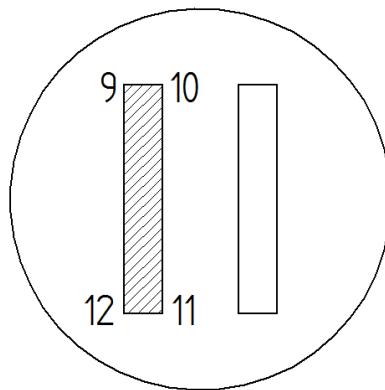
#### 7.4.5 Malla triangular vs. Malla cuadrangular

Una diferencia importante entre las dos discretizaciones es el tiempo de cálculo del ciclo elemental. Comparando los tiempos se observa que el tiempo invertido en los ciclos elementales de las mallas cuadrangulares es alrededor de 2.3 veces mayor que las triangulares.

Lo anterior es solo un reflejo de la cantidad de operaciones que involucra la formulación cuadrangular, dado que posee un nodo adicional y requiere integración numérica. Las evaluaciones de la integración numérica no fueron necesarias para la formulación triangular debido a que los elementos utilizados para esta formulación no son Isoparamétricos y esto permite que las integrales puedan calcularse en forma analítica (ver capítulo 4, Integración Numérica).

#### 7.4.6 Malla estructurada vs. Malla no estructurada

Para comparar los resultados se tomarán cuatro nodos que son comunes para los cuatro mallados. Estos nodos corresponden a los vértices de la superficie  $C_2$ .



**Figura 14.** Nodos comunes a todas las discretizaciones.

En la Tabla 2 se observa una comparación entre soluciones en nodos que son comunes a todas las mallas. Entre las mallas triangulares la diferencia máxima es de 0.0689%, mientras que entre mallas cuadrangulares de -3.6107%. Es necesario recordar que las integrales de la formulación triangular fueron resueltas analíticamente, lo cual implica una ventaja numérica que reduce el error (Diferencia). Queda como labor pendiente comparar los mismos resultados de la Tabla 2 para elementos cuadrilaterales utilizando una integración de Gauss con un orden de precisión mayor dado que los errores son hasta un orden de magnitud mayores en comparación con los triangulares.

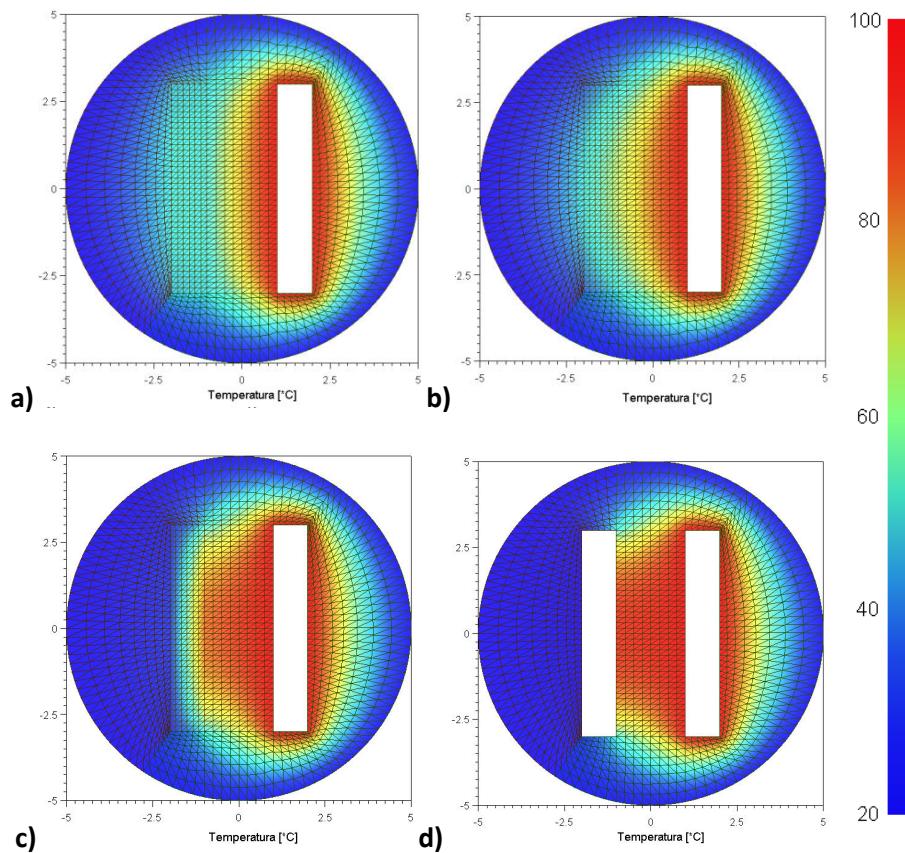
**Tabla 2.** Diferencia de soluciones entre mallas estructuradas y no estructuradas.  $Lc \sim 0.25$ .

	Temperatura °C					
	Triangular			Cuadrangular		
	Estruct.	No estruct.	Error	Estruct.	No estruct.	Error
<b>Nodo 9</b>	42.496040	42.454249	0.0522%	40.026224	42.914798	-3.6107%
<b>Nodo 10</b>	47.087388	47.095751	-0.0105%	47.272894	46.358643	1.1428%
<b>Nodo 11</b>	47.072786	47.094860	-0.0276%	46.658099	45.848318	1.0122%
<b>Nodo 12</b>	42.396550	42.451701	-0.0689%	42.266199	42.036144	0.2876%

Las soluciones para los nodos 9, 12 y 10, 11, deberían ser muy similares teniendo en cuenta la simetría de la geometría y de las condiciones de borde. Numéricamente, los resultados expuestos en la Tabla 2 apoyan esta tendencia.

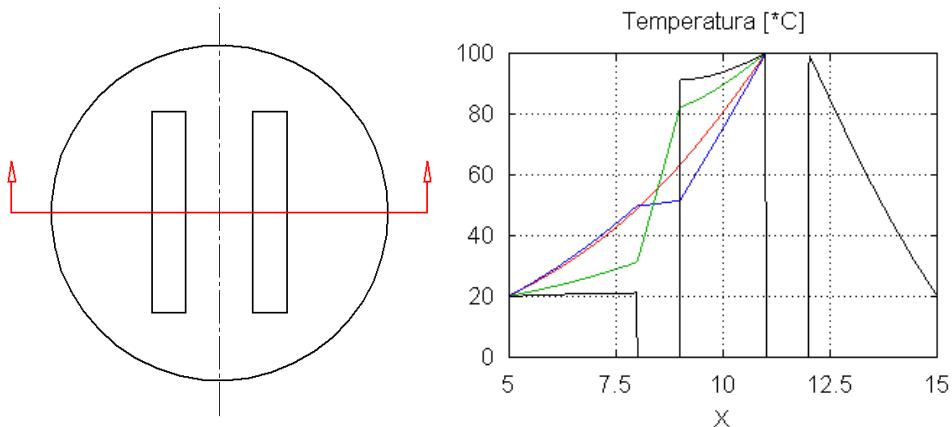
#### 7.4.7 Comportamiento de la solución con la variación de $C_2$

A continuación compararemos los resultados de las soluciones numéricas para los tipos de elemento. Nuestro interés es observar el comportamiento del campo de temperaturas con el cambio de la conductividad térmica de la zona  $C_2$ .

**Figura 15.** Soluciones con malla triangular estructurada con variación de la conductividad  $C_2$ .

a)  $C_2 = 10C_0$ . b)  $C_2 = C_0$ . c)  $C_2 = 0.1C_0$ . d)  $C_2 = 0$ .

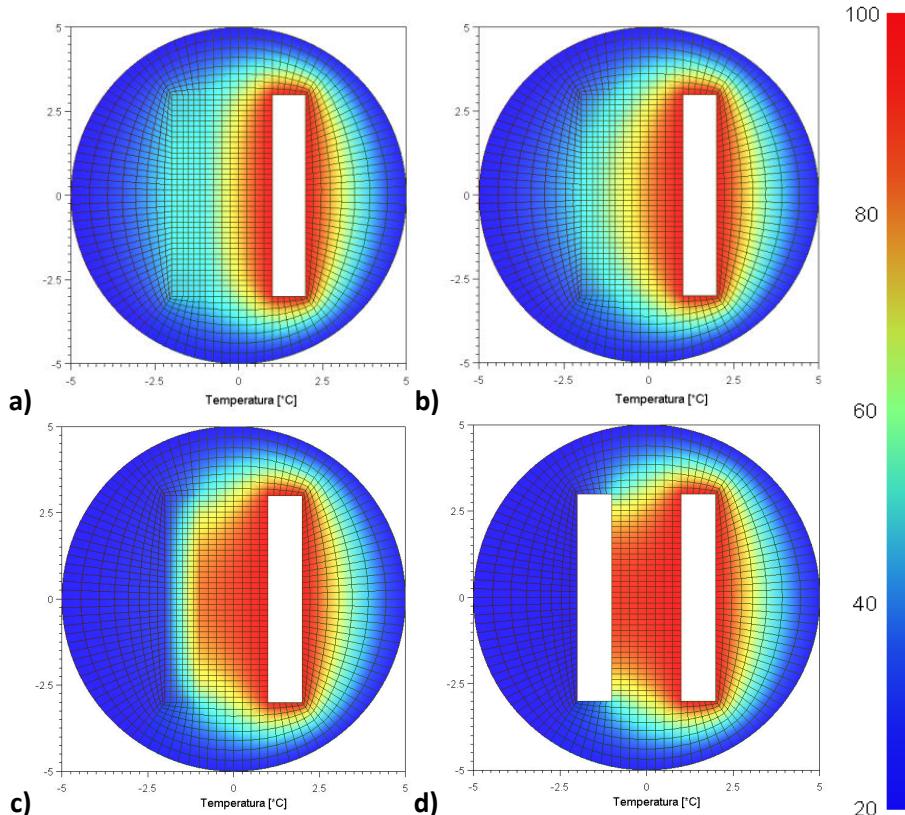
En la Figura 16 se observan las curvas de temperatura sobre la línea de simetría (línea roja) para los diferentes valores de conductividad. De la coordenada  $X=10$  hacia la derecha la curva de color negro se superpone sobre las demás, por ello no se aprecian, dado que las curvas azul roja y verde, hacia la derecha de  $C_1$ , tienen todas soluciones muy similares. Las curvas de esta figura fueron generadas con resultados triangulares y estructurados. No se grafican curvas para la malla cuadrangular dado que son muy similares a las de la Figura 16.



**Figura 16.** Perfiles de temperatura sobre línea de simetría.

$$- C_2 = 10C_0 \quad - C_2 = C_0 \quad - C_2 = 0.1C_0 \quad - C_2 = 0$$

La Figura 16 es producto del Post-Procesamiento de los datos a través del GMSH. En la Tabla 3 se observa la diferencia máxima entre la solución de la formulación triangular y la cuadrangular (vectores de temperatura) para diferentes valores de  $C_2$ . Realizar esta comparación fue posible gracias a que la numeración de los nodos coincide por completo.



**Figura 17.** Soluciones con malla cuadrangular estructurada con variación de la conductividad  $C_2$ .

**Tabla 3.** Porcentaje de diferencia entre formulaciones para diferente conductividad  $C_2$ 

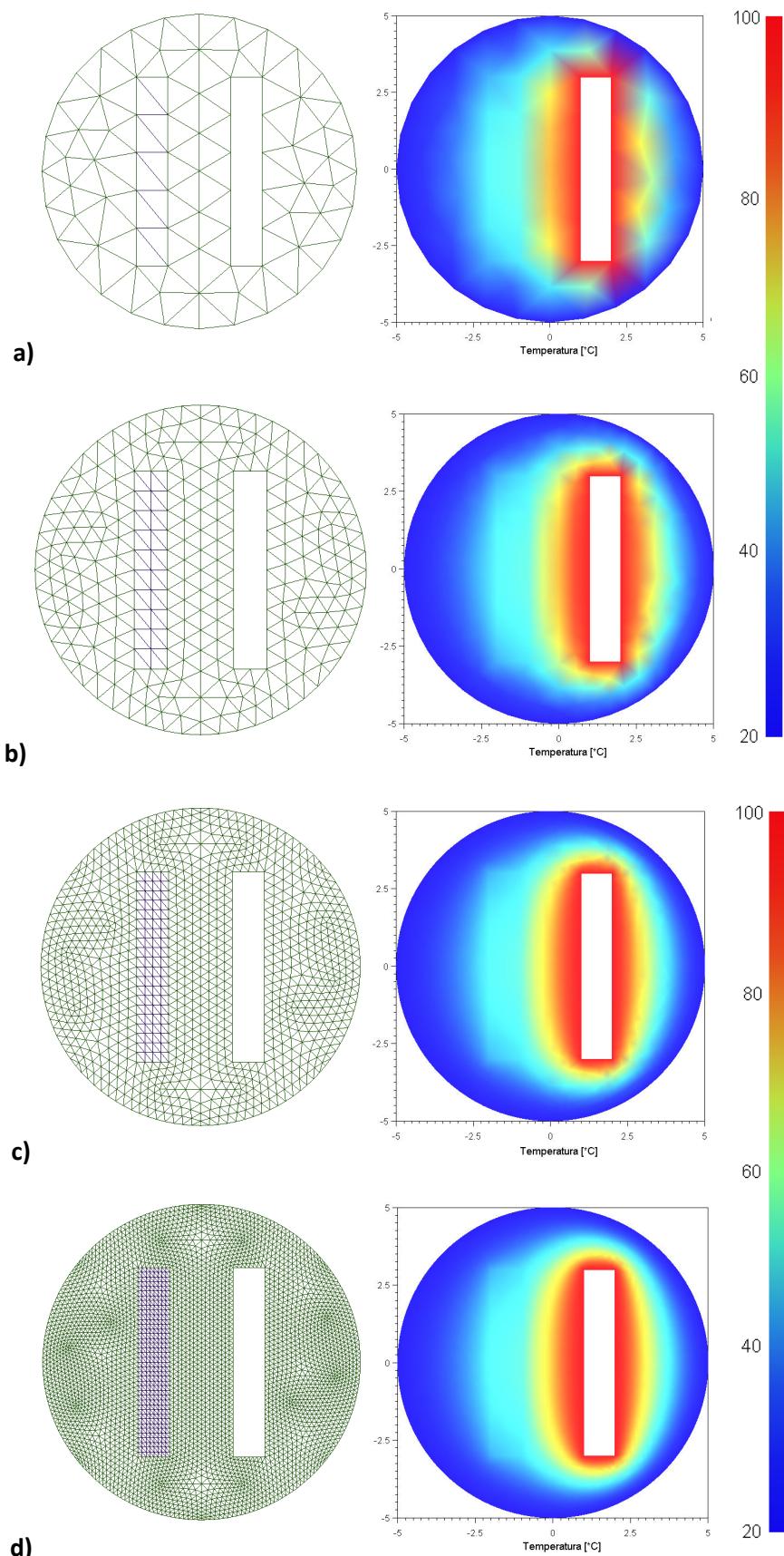
$C_2$	Error máximo absoluto °C $(T_{triangular} - T_{cuadrangular})$	Error máximo relativo $[(T_{tri} - T_{cuad})/80] * 100$
$10C_0$	3.0261	3.7826%
$5C_0$	2.4698	3.0873%
$C_0$	2.4113	3.0141%
$0.1C_0$	2.4051	3.0063%
<b>0</b>	2.6829	3.3537%

De la Tabla 3 podemos concluir que el error máximo entre las soluciones de las dos formulaciones permanece entre 3 y 3.8% para el mismo tamaño de malla ( $L_c \sim 0.25$ ).

#### 7.4.8 Malla gruesa vs. Malla fina

Se seleccionó el mallado triangular no estructurado para el análisis. En la Figura 18 observamos las diferentes distribuciones de temperatura a medida que la malla es refinada. Se observa que a medida que el tamaño de los elementos decrece la distribución del campo de temperaturas nodales permite que las interpolaciones lineales entre los valores nodales de cada elemento aproximen mejor la solución de la ecuación diferencial.

Aún faltando algunos análisis, estos resultados gráficos indican que el método aparentemente es consistente en lo que respecta a la formulación triangular.



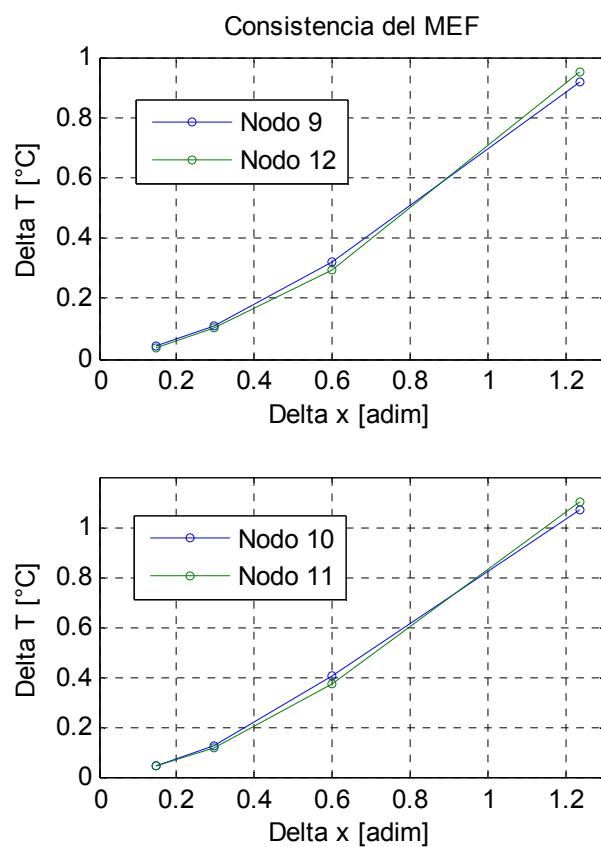
**Figura 18.** Soluciones con variación del tamaño de la malla.

**a)** Malla sin refinar.  $Lc=1$ . **b)**  $Lc=0.5$ . **c)**  $Lc=0.25$ . **d)**  $Lc=0.125$ .

**Tabla 4.** Comportamiento de la solución con el cambio del tamaño de la malla.

Temperaturas: Malla triangular – No estructurada					
$x$	2.4203	1.1863	0.5897	0.2943	0.1470
Lc	2	1	0.5	0.25	0.125
<b>Nodo 9</b>	43.802775	42.886877	42.563585	42.454249	42.413515
<b>Nodo 10</b>	48.681529	47.617441	47.217009	47.095751	47.050322
<b>Nodo 11</b>	48.681529	47.582315	47.212365	47.094860	47.049960
<b>Nodo 12</b>	43.802775	42.853995	42.556644	42.451701	42.412173

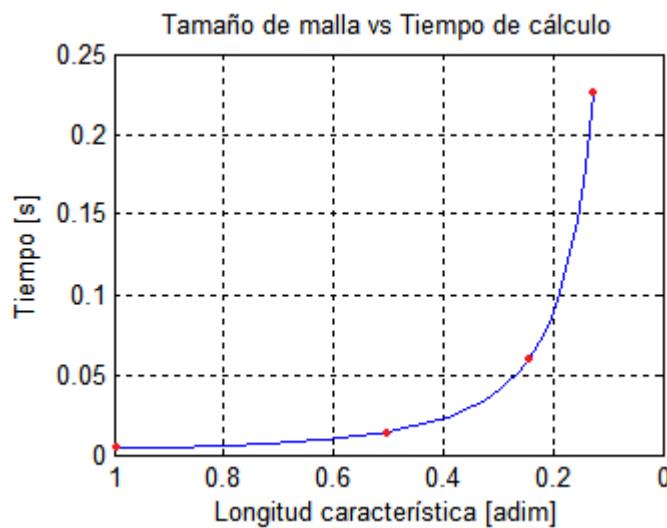
Analizando los resultados numéricos de la Tabla 4, ilustrados en la Figura 19, nos muestran la tendencia consistente del método de los Elementos Finitos a través del resultado de las curvas Delta x vs. Delta T. Así, cuando Delta x tiende a cero, Delta T también tiende a cero.

**Figura 19.** Solución vs. Tamaño de malla para malla triangular no estructurada.

En la Tabla 5 se presenta el cambio de algunas variables con el tamaño de la malla. Estos datos corresponden a las simulaciones de la Figura 18. La variable  $\Delta x$  corresponde a la longitud promedio de los lados de todos los elementos de la malla.

**Tabla 5.** Variación de algunos parámetros con el cambio del tamaño de la malla.

Longitud característica	$\Delta x$	Tiempo calculo ciclo elemental	Tiempo cálculo $K \cdot \hat{T} = b$	Elementos	Nodos
1	1.1863	0.0049	0.0088	122	81
0.5	0.5897	0.0142	0.0151	488	284
0.25	0.2943	0.0570	0.0585	1952	1056
0.125	0.1470	0.2260	1.6706	7808	4064

**Figura 20.** Tamaño de malla vs. Tiempo de cálculo de ciclos elementales para malla triangular.

La Figura 20 contiene la curva de  $Lc$  vs. Tiempo de cálculo de ciclos elementales para la malla triangular refinada de la Figura 18. Un buen ajuste obtenido para los cuatro puntos ( $R^2=0.9999$ ) resultó con una regresión de la forma  $t_{calculo} = a(Lc)^b$ , donde  $a = 0.00365, b = -1.984$ .

Es posible entonces concluir que el tiempo de cálculo de todos los ciclos elementales de cada simulación varía en forma potencial con la reducción de la longitud característica.

## 8. CONCLUSIONES GENERALES Y OBSERVACIONES

Utilizando el software de mallado GMSH se dibujaron dos geometrías para el problema (una para malla estructurada y otra para no estructurada), y se generaron mallas de diferentes longitudes características y disposición de elementos con el propósito de estudiar el comportamiento de variables como los tiempos de procesamiento, errores numéricos, distribuciones de temperatura, etc.

Se comprobó la dependencia de la calidad de los resultados numéricos del diseño de la malla. Resulta conveniente refinar la malla en las zonas de interés o donde exista un gradiente de temperaturas mayor. Se puede concluir también que el tamaño de los elementos es fundamental para obtener mejores aproximaciones de la solución numérica, mejorando ésta conforme su tamaño tiende a cero.

Se verificó que el método de los Elementos Finitos es consistente, dado que el error tiende a cero conforme se hace más fina la malla.

Gracias a la simetría de la geometría y de las condiciones de borde, el problema de transferencia de calor tratado en este trabajo pudo haberse solucionado resolviendo numéricamente solo uno de los lados simétricos.

Finalmente, el método numérico basado en Elementos Finitos se implementó exitosamente a través de la elaboración de un algoritmo que resultó útil para determinar diferentes aproximaciones a la solución del problema de transferencia de calor.

### **REFERENCIAS BIBLIOGRÁFICAS**

1. BATHE, Klaus-Jurgen. Finite Element Procedures. Prentice-Hall. 1996. PP 465.

# ANEXOS

## RESUMEN DE PARÁMETROS DE MALLADO

Los siguientes parámetros fueron utilizados en cada operación de mallado a lo largo de todo el trabajo. Todos pueden ser observados siguiendo Options -> Mesh, en la interfaz de usuario de GMSH.

<b>Parámetros generales</b>	
Algoritmo de mallado 2D	MeshAdapt
Algoritmo de recombinación	Standard
Algoritmo de subdivisión	Ninguno
Algoritmo de remallado	Automático
Parametrización de remallado	Armónica
Smoothing steps	1
Longitud característica (inicial)	1
Orden del elemento	1

## ALGORITMO DE CÁLCULO COMPUTACIONAL

A continuación se presenta el algoritmo de cálculo elaborado en la versión R2008a de Matlab.

```
%INTRODUCCIÓN AL MÉTODO DE LOS ELEMENTOS FINITOS
%TRABAJO PRÁCTICO 7 - ESPECIAL
%Alumno: Fredy Andrés Mercado Navarro.
%Identificación: 98773532.
%Presentado al profesor: Dr. Sergio Ferro.

clear all
clc

%A continuación, el comando load carga
%Un archivo que contiene:
%1.Matriz de coordenadas de los nodos
%2.Matriz de nodos de los bordes
%3.Matriz de conectividad entre nodos y elementos

-----VARIABLES QUE SE PUEDEN MODIFICAR-----
formulacion=1; %1 para resolver con Triangulos
                 %2 para resolver con Cuadrangulos
load matrices_triest_fina_c2
Cond=5; %Valor/Múltiplo para definir Conductividad de C2
-----VARIABLES QUE SE PUEDEN MODIFICAR-----

nnodos=length(coord_0(:,1));
nelementos=length(conec(:,1));
ncoord_0=length(coord_0(1,:));
nfilasnodos_bc=length(nodos_bc(:,1));

coord=zeros(nnodos,6);
```

```

%Sumo la matriz coord_0 a coord

for j=1:ncolcoord_0
    for i=1:nodos
        coord(i,j)=coord_0(i,j);
    end
end

%Se definen las condiciones de borde
%de Dirichlet

bc1=20;
bc2=100;

for i=1:nfilasnodos_bc
    %Llena columna con temperaturas de bc1 - Exterior
    if nodos_bc(i,4)==1 %Cambiar si GMSH enumeró diferente
        for j=6:7
            coord(nodos_bc(i,j),5)=1;
            coord(nodos_bc(i,j),6)=bc1;
        end
    end
    %Llena columna con temperaturas de bc2 - Conductor Der.
    if nodos_bc(i,4)==2 %Cambiar si GMSH enumeró diferente
        for j=6:7
            coord(nodos_bc(i,j),5)=1;
            coord(nodos_bc(i,j),6)=bc2;
        end
    end
end

kglobal=zeros(nnodos,nnodos);

%Conductividad en x e y

Kxy=1;

%Matriz de conductividades

K=[Kxy 0;0 Kxy];

%CICLO ELEMENTAL

tic;

if formulacion==1 %ELEMENTO TRIANGULAR

    for k=1:nelementos
        %Derivadas de funciones de forma
        %Definidas entre 0 y 1
        %h1=1-r-s
        %h2=r
        %h3=s

        dh1dr=-1;
        dh1ds=-1;
        dh2dr=1;

```

```

dh2ds=0;
dh3dr=0;
dh3ds=1;

%Defino la siguiente matriz de derivadas

Mdhdrs=[dh1dr dh2dr dh3dr; ...
          dh1ds dh2ds dh3ds];

%Llamo las coordenadas de los nodos
%del elemento

x1=coord(conec(k,6),2);
x2=coord(conec(k,7),2);
x3=coord(conec(k,8),2);

y1=coord(conec(k,6),3);
y2=coord(conec(k,7),3);
y3=coord(conec(k,8),3);

%Defino la matriz de coordenadas
%como:

coordelem=[x1 y1;x2 y2;...
            x3 y3];

%Defino al Operador Jacobiano
%como:

J=Mdhdrs*coordelem;

%Operador Jacobiano inverso

invJ=inv(J);

%Matriz gradiente de H

B=invJ*Mdhdrs;

%Matriz de rigidez elemental

kelemental=B'*K*B*det(J);

%Imposición de conductividad de C2

if coneck(4)==4
    kelemental=Cond*kelemental;
end

%Llenado de matriz global

for j=1:3
    for i=1:3
        kglobal(conec(k,i+5),coneck(j+5))=...
            kglobal(conec(k,i+5),coneck(j+5))...
            +kelemental(i,j);
    end
end

```

```

end

elseif formulacion==2 %ELEMENTO CUADRANGULAR

%Puntos de Gauss
puntogauss=[1/sqrt(3) -1/sqrt(3) ];

%CICLO ELEMENTAL

tic;

for k=1:nELEMENTOS
    for j=1:2
        for i=1:2
            r=puntogauss(i);
            s=puntogauss(j);

            %Defino tamaño de matriz
            %de rigidez de cada elemento

            kelemental=zeros(4,4);

            %Derivadas de funciones de forma

            dh1dr=(1/4)*(1+s);
            dh1ds=(1/4)*(1+r);
            dh2dr=-(1/4)*(1+s);
            dh2ds=(1/4)*(1-r);
            dh3dr=-(1/4)*(1-s);
            dh3ds=-(1/4)*(1-r);
            dh4dr=(1/4)*(1-s);
            dh4ds=-(1/4)*(1+r);

            %Defino la siguiente matriz de derivadas

            Mdhdrs=[dh1dr dh2dr dh3dr dh4dr; ...
                      dh1ds dh2ds dh3ds dh4ds];

            %Llamo las coordenadas de los nodos
            %del elemento

            x1=coord(conec(k,6),2);
            x2=coord(conec(k,7),2);
            x3=coord(conec(k,8),2);
            x4=coord(conec(k,9),2);

            y1=coord(conec(k,6),3);
            y2=coord(conec(k,7),3);
            y3=coord(conec(k,8),3);
            y4=coord(conec(k,9),3);

            %Defino la matriz de coordenadas
            %como:

            coordelem=[x1 y1;x2 y2; ...
                       x3 y3;x4 y4];

            %Defino al Operador Jacobiano

```

```

%como:

J=Mdhdrs*coordelem;

%Operador Jacobiano inverso

invJ=inv(J);

%Matriz gradiente de H

B=invJ*Mdhdrs;

%Apunte correspondiente a la
%evaluación del punto de Gauss
%de coordenadas (r,s)

aportegauss=B'*K*B*det(J);

%Llenado de matriz de rigidez
%elemental con aportes

kelemental=kelemental+aportegauss;
end
end

%Imposición de conductividad de C2

if coneck(4)==4
    kelemental=Cond*kelemental;
end

%Llenado de matriz global

for j=1:4
    for i=1:4
        kglobal(conek(i+5),conek(j+5))=...
            kglobal(conek(i+5),conek(j+5))...
            +kelemental(i,j);
    end
end
end
end

tiempo=zeros(2);
tiempo(1)=toc;

%Multiplicamos y pasamos valores conocidos al vector b
%Llenamos el vector b con las temperaturas conocidas

b=zeros(nnodos,1);
for j=1:nnodos
    for i=1:nnodos
        if coord(i,5)~=1
            b(i,1)=b(i,1)-kglobal(i,j)*coord(j,6);
        else
            b(i,1)=coord(i,6);
        end
    end
end

```

```
end

for i=1:nodos
    if coord(i,5)==1
        kglobal(:,i)=0;
        kglobal(i,:)=0;
        kglobal(i,i)=1;
    end
end

coord;

tic;
T=kglobal\b;
tiempo(2)=toc;

nfilasT=length(T(:,1));
Tmsh=zeros(nfilasT,2);

for i=1:nfilasT
    Tmsh(i,1)=i;
    Tmsh(i,2)=T(i);
end

Tmsh;

fi=fopen('salida.txt', 'w');

for i=1:nfilasT
    for j=1:2
        fprintf(fi, '%f \r', Tmsh(i,j));
    end
    fprintf(fi, '\n');
end

fclose(fi)
```

--- Fin del Programa ---