

SUBSECRETARÍA DE EDUCACIÓN SUPERIOR
DIRECCIÓN GENERAL DE EDUCACIÓN
SUPERIOR TECNOLÓGICA
INSTITUTO TECNOLÓGICO DE TUXTLA
GUTIÉRREZ



SECRETARÍA DE
EDUCACIÓN PÚBLICA



SEP

TRABAJO PROFESIONAL

COMO REQUISITO PARA OBTENER EL TÍTULO DE:

INGENIERO MECÁNICO

QUE PRESENTA:

JUAN LUIS PÉREZ RUÍZ

CON EL TEMA:

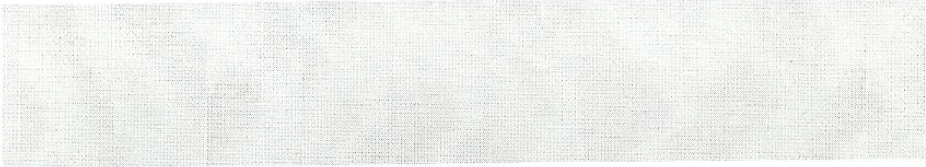
**“SOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE
DIFUSIÓN DE CALOR USANDO EL MÉTODO
DE PETROV-GALERKIN ”**

MEDIANTE :

**OPCION I
(TESIS PROFESIONAL)**

TUXTLA GUTIÉRREZ, CHIAPAS

ENERO 2012



SUBSECRETARÍA DE EDUCACIÓN SUPERIOR
DIRECCIÓN GENERAL DE EDUCACIÓN SUPERIOR TECNOLÓGICA
INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ

SECRETARÍA DE
EDUCACIÓN PÚBLICA

DIRECCIÓN
SUBDIRECCIÓN ACADÉMICA
DIVISIÓN DE ESTUDIOS PROFESIONALES
Tuxtla Gutiérrez, Chiapas., 02/FEBRERO/2012

OFICIO NUM. DEP-CT-020/2012

C. JUAN LUIS PÉREZ RUIZ
PASANTE DE LA CARRERA DE INGENIERÍA MECÁNICA
EGRESADO DEL INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ
P R E S E N T E.

Habiendo recibido la comunicación de su trabajo profesional por parte de los CC.ING. APOLINAR PÉREZ LÓPEZ, M.I. APOLINAR PÉREZ LÓPEZ, M.en C. IGNACIO ARRIJOA CÁRDENAS, M.I. LORENZO MARCIANO VÁZQUEZ e ING. ROBERTO CIFUENTES VILLAFUERTE, en el sentido que se encuentra satisfactorio el contenido del mismo como prueba escrita, **AUTORIZO** a Usted a que se proceda a la impresión del mencionado Trabajo denominado:

"SOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE DIFUSIÓN DE CALOR UTILIZANDO EL MÉTODO DE PETROV-GALERKIN"

Registrado mediante la opción:
I (TESIS PROFESIONAL)

ATENTAMENTE
"CIENCIA Y TECNOLOGÍA CON SENTIDO HUMANO"

ING. ROBERTO CIFUENTES VILLAFUERTE
JEFE DE LA DIVISIÓN DE ESTUDIOS PROFESIONALES

Vo. Bo.

M. en C. JOSÉ LUIS MENDEZ NAVARRO
DIRECTOR

C.c.p.- Departamento de Servicios Escolares
C.c.p.- Expediente
I'JLMN/I'RCV/L'ORC

Secretaría de Edu. Pública
Instituto Tecnológico
de Tuxtla Gutiérrez
Div. de Est. Profesionales
Carretera Panamericana Km.1080, C.P. 29050, Apartado Postal 599
Teléfonos: (961) 61 5-03-80 (961) 61 5-04-61 Fax: (961) 61 5-16-87
<http://www.ittg.edu.mx>



Alcance del Sistema: Proceso Educativo

ABSTRACT

This thesis presents the numerical solution of the two dimensional heat diffusion equation using the local Petrov-Galerkin method. This equation is numerically solved by discretizing its weak form, formulated in a cartesian coordinate system with local subdomains of integration. The Moving Least Squares (MLS) method is used to generate the shape functions. The test functions are the same weight functions used in the MLS process needed in the local Petrov-Galerkin method. The weight function is a fourth order spline with compact support.

The theoretical aspects and the numerical implementation of the local Petrov-Galerkin method are applied to physical problems in order to know the temperature distribution in rectangular or square geometries, using homogeneous boundary conditions.

Three different cases in two dimensional domains are presented:

- (1) Laplace equation with Dirichlet and Neumann conditions
- (2) Poisson equation with Dirichlet conditions
- (3) Heat diffusion equation with Dirichlet conditions.

AGRADECIMIENTOS

A Dios por su misericordia y eterno amor.

A mis padres: Profr. Juan Luis Pérez Urbina y Profra. María Fanny Ruíz Álvarez, quienes han estado conmigo en todo momento, apoyándome económica, moral y espiritualmente a lo largo de mi carrera, y sobretodo, dándome amor y cariño.

A mis hermanas Lic. Eunice Julieta Pérez Ruíz y Dr. Citlally Liliana Pérez Ruíz que siempre están ahí para ayudarme y aconsejarme.

A mis abuelitos por sus bendiciones, consejos y preocupaciones que han forjado mi vida.

A mis profesores de la carrera de Ingeniería Mecánica por la preparación académica, profesional y ética que me brindaron.

Al M.I Apolinar Pérez López por el asesoramiento y apoyo continuo en la realización de este trabajo.

Un agradecimiento adicional al Ing. Rafael Solís Velázquez por ser pieza fundamental en mi formación educacional.

A las instituciones que han sido eslabones en mi formación académica: Esc. Prim. Gral. Francisco Villa, Esc. Sec. Técnica. No. 09, Preparatoria No.1 del Estado y a mi alma máter, el Instituto Tecnológico de Tuxtla Gutiérrez.

ÍNDICE GENERAL

	Página
RESUMEN	IX
LISTA DE SÍMBOLOS	XI
CAPÍTULO 1: INTRODUCCIÓN	1
CAPÍTULO 2: FUNDAMENTO TEÓRICO DEL MÉTODO LOCAL DE PETROV-GALERKIN	5
2.1 Método de los residuos pesados	5
2.2 Mínimos Cuadrados Móviles	6
2.2.1 Funciones de peso.....	13
2.3 Aplicación del método local de Petrov-Galerkin: ec. de difusión de calor.....	15
2.3.1 La forma débil	16
2.3.2 Discretización de la forma débil	18
2.3.3 Discretización temporal.....	20
2.4 Implementación numérica del método local de Petrov-Galerkin.....	21
CAPÍTULO 3: APLICACIÓN NUMÉRICA DEL MÉTODO LOCAL DE PETROV-GALERKIN	23
3.1 El Método del Elemento Finito	23
3.2 Caso 1: Ecuación de Laplace	27
3.2.1 Condiciones de Dirichlet	28
3.2.2 Condiciones de Neumann.....	29

3.2.3 Ecuación de Laplace en estado transitorio	30
3.3 Caso 2: Ecuación de Poisson	31
3.4 Caso 3: Ecuación de difusión de calor.....	33
CAPÍTULO 4: RESULTADOS	35
4.1 Resultados numéricos de la ecuación de Poisson unidimensional usando el M.E.F	35
4.2 Resultados numéricos de la ecuación de Laplace con condiciones frontera de Dirichlet y Neumann	37
4.2.1 Condiciones de Dirichlet	37
4.2.2 Condiciones de Neumann.....	38
4.2.3 Ecuación de Laplace en estado transitorio	40
4.3 Resultados numéricos de la ecuación de Poisson	42
4.4 Resultados numéricos de la ecuación de difusión de calor	44
CAPÍTULO 5: CONCLUSIONES	49
APÉNDICES	
A FORMA DÉBIL DE LAS ECUACIONES DE LAPLACE Y POISSON	51
A.1 Forma débil de la ecuación de Laplace	51
A.2 Forma débil de la ecuación de Poisson.....	53
B CUADRATURA DE GAUSS-LOBBATO-LEGENDRE	55
C SOLIDWORKS FLOW SIMULATION	63
D PROGRAMACIÓN DEL MÉTODO LOCAL DE PETROV-GALERKIN USANDO MAPLE	65
BIBLIOGRAFÍA	75

ÍNDICE DE FIGURAS

CAPÍTULO 2:

2.1 Dominio y subdominios de un problema.....	7
2.2 La función de aproximación u^h	9
2.3 Funciones de interpolación basadas en el método de los MLS.....	12
2.4 Primeras derivadas de las funciones de interpolación basadas en el método MLS	12
2.5 Función de peso “spline” de cuarto orden para $d_{\max}=1.01$	14
2.6 Función de peso “spline” de cuarto orden para $d_{\max}=1.5$	14
2.7 Formas que pueden tener los subdominios.....	15

CAPÍTULO 3:

3.1 Estudio de barra calentada con el M.E.F.....	23
3.2 Funciones de expansión para cada elemento.	24
3.3 Problema físico de la ec. de Laplace con condiciones de Dirichlet.	28
3.4 Problema físico de la ec. de Laplace con condiciones de Neumann.....	29
3.5 Problema físico de la ecuación de Laplace en estado transitorio	31
3.6 Problema físico de la ecuación de Poisson.....	32
3.7 Problema físico de la ecuación de difusión de calor.....	34

CAPÍTULO 4:

4.1 Resultado numérico del M.E.F para $n=50$	36
4.2 Resultado numérico del M.E.F para $n=100$	36
4.3 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Laplace con condiciones de Dirichlet).....	37
4.4 Distribución de temperaturas generada por la solución analítica (ecuación de Laplace bidimensional con condiciones de Dirichlet).....	38

4.5 Distribución de temperaturas obtenida en 2D y 3D con el método Petrov-Galerkin (ec. Laplace con condiciones de Dirichlet) para 51 x 51 nodos.....	38
4.6 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Laplace con condiciones de Neumann)	39
4.7 Distribución de temperaturas generada por la solución analítica (ecuación de Laplace bidimensional con condiciones de Neumann)	39
4.8 Distribución de temperaturas obtenida en 2D y 3D con el método de Petrov-Galerkin (ec. de Laplace con condiciones de Neumann) para 51 x 51 nodos.	40
4.9 Distribución de temperaturas en estado transitorio para 50 x 50 nodos (ecuación de Laplace bidimensional).....	42
4.10 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Poisson con condiciones de Dirichlet).....	43
4.11 Solución analítica de la ec. Poisson bidimensional.	44
4.12 Solución numérica de la ec. Poisson bidimensional para 51 x 51 nodos	44
4.13 Solución numérica de la ec. de difusión de calor para 51 x 51 nodos	47

APÉNDICE B:

B.1 Desarrollo del método de integración de Gauss-Lobbato-Legendre usando dos puntos a partir del método trapezoidal.....	56
B.2 Derivación del método de integración de GLL	57

APÉNDICE C:

C.1 Distribución de temperaturas en °C de una placa rectangular (ecuación de Laplace con condiciones de Dirichlet) usando SolidWorks Flow Simulation	64
C.2 Distribución de temperaturas en °C de una placa rectangular con generación de calor (ec. de Poisson con condiciones de Dirichlet) usando SolidWorks Flow Simulation	64

RESUMEN

Esta tesis trata sobre la simulación numérica de la ecuación de difusión de calor bidimensional empleando el método local de Petrov-Galerkin. Esta ecuación se resuelve en su forma débil en un sistema de coordenadas cartesianas con subdominios locales de integración.

El esquema de interpolación por mínimos cuadrados móviles se usa para generar las funciones de expansión. La función de peso utilizada en el método de los residuales pesados es una función spline de cuarto orden, la misma que se emplea para generar las funciones de expansión basadas en mínimos cuadrados móviles.

Los aspectos teóricos y la implementación numérica del método local de Petrov-Galerkin se describen para resolver problemas físicos donde se desea conocer la distribución de temperaturas en geometrías rectangulares o cuadradas usando condiciones de frontera homogéneas.

Se presentan tres diferentes casos en dominios bidimensionales donde se emplea el método:

- (1) Ecuación de Laplace con condiciones de Dirichlet y Neumann.
- (2) Ecuación de Poisson con condiciones de Dirichlet.
- (3) Ecuación de difusión de calor con condiciones de Dirichlet.

LISTA DE SÍMBOLOS

Alfabeto español

d_{max}	parámetro de suavizado[m]
h	coeficiente de transferencia de calor [$W/m^2 \cdot K$]
k	conductividad térmica [$W/m \cdot K$]
N	número total de nodos
\mathbf{n}	vector normal unitario []
Q'	fuente de calor [W/m^3]
\bar{q}	flujo de calor por unidad de superficie [W/m^2]
\mathbf{T}	vector de temperaturas desconocidas [K]
\bar{T}	campo de temperaturas dado [K]
T	temperatura [K]
T_0	campo de temperaturas en $t = 0$ [K]
t	tiempo [s]
$\mathbf{x} = (x_1, x_2, x_3)$	coordenadas espaciales [m]

Símbolos griegos

α	difusividad térmica [m^2/s]
Φ	función de expansión
Ψ	función de prueba
Ω	dominio o subdominio del problema
Γ	frontera del dominio o subdominio del problema

Superíndices

'	derivada temporal
h	aproximación de una función
T	transpuesta de una matriz o vector

Subíndices

i	nodo piloto, renglón de una matriz
j	nodos vecinos, columnas de una matriz
$(\cdot)_{,k} = \partial(\cdot) / \partial x_k$	derivación espacial

Abreviaturas

EDP	Ecuación Diferencial Parcial
GLL	Gauss-Lobatto-Legendre
MDF	Método de las Diferencias Finitas
MEF	Método del Elemento Finito
MVF	Método del Volumen Finito
MLS	Moving Least Squares
WRM	Weighted Residual Method

CAPÍTULO 1

INTRODUCCIÓN

Una de las tareas fundamentales del ingeniero consiste en el análisis, esto es la predicción cuantitativa del comportamiento de un sistema o un proceso para proceder a su diseño eficiente o para cumplir con especificaciones de producción.

Los problemas de ingeniería se estudian con modelos matemáticos que representan situaciones o problemas físicos en los que intervienen equilibrio de masas, fuerzas o energía, estos modelos son ecuaciones diferenciales con condiciones de contorno iniciales determinadas.

Hay muchos problemas prácticos en ingeniería en los cuales no se puede obtener la solución exacta, esto se debe a la complejidad natural de las ecuaciones diferenciales o a las dificultades que pueden ocurrir con las condiciones de contorno o iniciales.

Para tratar este tipo de problemas usamos las aproximaciones numéricas. En contraste a la solución analítica, que muestra el comportamiento exacto de un sistema en cualquier punto del mismo, las soluciones numéricas aproximan la solución exacta solo en puntos discretos, llamados nodos.

El surgimiento de las computadoras ha tenido un gran impacto en la ingeniería y en los diferentes campos de ésta para la solución de problemas físicos aplicados. El método o código presentado en este trabajo (método local de Petrov-Galerkin) nos ayuda a resolver problemas de transferencia de calor y presentar los resultados mediante la simulación numérica por computadora.

Estos problemas son presentados en diferentes casos, primero, la solución de la ecuación de Laplace con condiciones de Dirichlet y Neumann; segundo, la solución de la ecuación de Poisson con condiciones de Dirichlet. Hasta este punto,

estos casos nos permiten validar el código computacional ya que existen soluciones analíticas disponibles para esos problemas cuando las condiciones de frontera son homogéneas o no homogéneas. Finalmente, el último caso es dar solución a la ecuación de difusión de calor (2D) que es una ecuación diferencial parcial (EDP) de segundo orden que describe la distribución del calor (o variaciones de la temperatura) en una región a lo largo del transcurso del tiempo con las condiciones de frontera antes mencionadas. Las regiones de trabajo en los casos serán geometrías rectangulares o cuadradas.

Existen otros métodos computacionales, tales como el método de los elementos finitos (MEF), el método del volumen finito (MVF), el método de las diferencias finitas (MDF), entre otros, y que también pueden usarse en este tipo de ecuaciones, pero muestran ciertas desventajas debido a las funciones de prueba que usan.

Este trabajo está organizado de la siguiente manera: En el capítulo 2 se presenta el procedimiento matemático necesario para poder aplicar la metodología del Método Local de Petrov-Galerkin a cualquier ecuación diferencial dada. A fin de introducir al lector algunos de los conceptos más importantes que se manejarán a lo largo de este trabajo, en la sección 2.1 se presenta una breve introducción al método de los residuos pesados del que pueden deducirse los métodos numéricos existentes (método de los elementos finitos, volúmenes finitos, elementos espectrales, etc.) y desde luego el método usado en este trabajo. El procedimiento para generar las funciones de expansión basadas en el método de interpolación por mínimos cuadrados móviles, se presenta en la sección 2.2. En la sección 2.3 se presenta la aplicación del Método Local de Petrov-Galerkin para resolver la ecuación de difusión de calor en dos dimensiones y en estado transitorio, utilizando coordenadas cartesianas. Aunque esta metodología puede aplicarse a cualquier EDP la ecuación de difusión nos permite mostrar la forma en que puede llevarse a cabo la discretización temporal y como se satisfacen las condiciones de frontera. El capítulo 2 concluye con la sección 2.4 presentando a manera de resumen los pasos a seguir para desarrollar un código computacional que resuelva numéricamente una ecuación diferencial parcial.

En el capítulo 3 se trabaja con el Método del Elemento Finito, esto con el fin de programar un código computacional sencillo y aprender a discretizar la ecuación de Poisson unidimensional y estacionario. Después, dentro del mismo capítulo se estudia la aplicación numérica del Método Local de Petrov-Galerkin para resolver las ecuaciones de Laplace, Poisson y difusión de calor aplicadas a problemas físicos. La forma débil de cada ecuación se presenta en forma bidimensional (ver apéndice A).

En el capítulo 4 se presentan los resultados que se obtuvieron utilizando el código numérico de Petrov-Galerkin. Se presentan primero, los resultados de la distribución de temperaturas en la barra calentada con generación de calor (Poisson unidimensional) obtenida con el M.E.F. Luego se muestran los resultados de la distribución de temperaturas en la sección transversal de una barra (ecuación de Laplace) con condiciones de frontera homogéneas de Dirichlet y Neumann, además de resolver el problema de la placa calentada súbitamente en sus extremos y ver el campo de temperaturas a través del tiempo. Posteriormente, se presentan los resultados del campo de temperaturas en las paredes de un horno con generación de calor (ecuación de Poisson bidimensional). Finalmente se resuelve el problema de un receptor solar con generación de calor pero en estado transitorio (ecuación de difusión de calor).

La sección 5 finaliza esta tesis presentando las conclusiones y los trabajos a futuro en el área de la simulación numérica utilizando métodos libres de malla.

En los apéndices se presentan aspectos matemáticos muy importantes a tomar en cuenta para implementar el Método de Petrov-Galerkin. Se da un ejemplo del código numérico del método para la resolver la ec. de Laplace, además de una sencilla comparación de los resultados obtenidos en este trabajo con otro programa como lo es SolidWorks que es una gran herramienta para la simulación de problemas de análisis térmico.

CAPÍTULO 2

FUNDAMENTOS TEÓRICOS DEL MÉTODO LOCAL DE PETROV-GALERKIN

En este capítulo se presentan los fundamentos matemáticos del Método Local de Petrov-Galerkin.

Con la finalidad de introducir algunos conceptos importantes que se manejan a lo largo de este trabajo, en la sección 2.1 se presenta brevemente el método de los residuos pesados (Fries and Matthies, 2004).

En la sección 2.2 se presenta el procedimiento estándar para generar las funciones de interpolación por mínimos cuadrados móviles, las cuales son utilizadas en el método de Petrov-Galerkin como funciones de expansión.

La sección 2.3 presenta la aplicación de la metodología del Método Local de Petrov-Galerkin en la solución de la ecuación de difusión de calor en estado transitorio (2D). El capítulo concluye con la sección 2.4 presentando un algoritmo general para escribir un código computacional empleando dicho método.

2.1. Método de los residuos pesados

La mayoría de los métodos numéricos pueden deducirse a partir del método de los residuos pesados (WRM, *Weighted Residual Method*). En términos generales, el objetivo del WRM es resolver numéricamente una ecuación diferencial parcial dada:

$$\mathcal{L}u(x) = f(x) \tag{2.1}$$

en donde \mathcal{L} es cualquier operador diferencial, $f(x)$ es un término fuente y $u(x)$ son las funciones que satisfacen la ecuación.

Se comienza proponiendo una función de aproximación $u^h(x) \approx u(x)$, la cual se expresa como la sumatoria de un conjunto de *funciones de expansión* Φ (también llamadas funciones de forma) y parámetros nodales desconocidos \mathbf{u} (incógnitas), esto es:

$$u^h(x) = \Phi^T(x)\mathbf{u} = \sum_{j=1}^N \Phi_j(x)u_j \quad (2.2)$$

en donde N es el número de términos de la sumatoria, de tal manera que con este pueda obtenerse una buena aproximación. Al reemplazar u^h por u en la EDP dada (ec. 2.1), obtenemos un error residual $r = \mathcal{L}u^h - f$, ya que en general la aproximación no satisface exactamente la ecuación. El residual r es “pesado” por medio de un conjunto de *funciones de prueba* Ψ , haciendo que ambos conjuntos de funciones sean ortogonales, esto es:

$$\int \Psi r d\Omega = \int \Psi (\mathcal{L}u^h - f) d\Omega = 0 \quad (2.3)$$

La expresión anterior es la forma discreta de la EDP y debe ser evaluada considerando las condiciones de frontera dadas. De la ec. (2.3), puede obtenerse un sistema de ecuaciones del tipo $\mathbf{A}\mathbf{u} = \mathbf{b}$ que debe resolverse para determinar las incógnitas \mathbf{u} , como se explicará en la sec. 2.3 de forma detallada.

Dependiendo de la elección de las funciones de prueba Ψ usadas en el WRM pueden producirse diferentes métodos numéricos, tales como colocación, volumen finito, método de momentos, entre otros (Karniadakis and Sherwin, 1999). Para el método que nos interesa, el Método de Petrov-Galerkin, las funciones de prueba Ψ son diferentes a las funciones de expansión Φ . En este trabajo, las funciones de interpolación por mínimos cuadrados móviles se emplean como funciones de expansión Φ y las funciones de prueba Ψ son las mismas funciones de peso w que se emplean para obtener las funciones Φ (ver sec.2.2).

2.2. Mínimos cuadrados móviles

El método de los mínimos cuadrados móviles (MLS, Moving Least Squares) se desarrolló para interpolar datos aleatorios con exactitud (Lancaster and Salkauskas, 1981). Los métodos MLS son ampliamente usados para generar las funciones de expansión ya que son fáciles de implementar en problemas n -

dimensionales y estos métodos son los utilizados en el método Petrov-Galerkin para este propósito. Existen diversos procedimientos para generar las funciones de expansión basadas en los métodos MLS. En esta sección las funciones de expansión Φ se obtienen minimizando una norma discreta L_2 (Fries and Matthies, 2004).

Suponga una función $u(\mathbf{x})$ definida en un dominio Ω , con frontera Γ . Dentro de este dominio, colocamos un conjunto de nodos $\mathbf{x}_i, i = 1, \dots, N$. Deseamos obtener una aproximación local u^h de la función u en algún punto \mathbf{x} , por lo que en su vecindad definimos un subdominio Ω_x , ambos localizados completamente dentro del dominio del problema Ω (ver fig. 2.1).

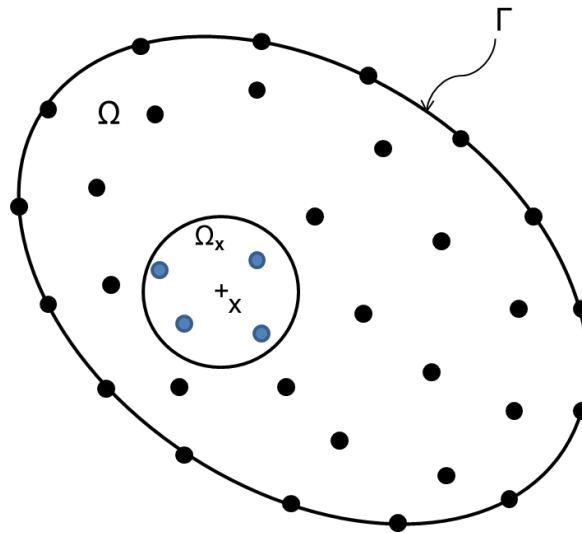


Figura 2.1 El punto \mathbf{x} y su vecindad, definida por el subdominio Ω_x . El conjunto de nodos del problema se muestran en puntos negros y los nodos en la vecindad del punto \mathbf{x} se muestran en azul. El dominio global se representa por Ω cerrado por su frontera Γ (2D).

Los métodos MLS definen la aproximación local de u alrededor de \mathbf{x} como:

$$u^h(\mathbf{x}) = \sum_{i=1}^m p_i(\mathbf{x}) a_i(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \mathbf{a}(\mathbf{x}) \quad (2.4)$$

donde m es el número de términos del polinomio base, $p_i(\mathbf{x})$ son funciones monomiales y $a_i(\mathbf{x})$ son sus coeficientes, los cuales están en función de las coordenadas espaciales $\mathbf{x} = (x_1, x_2, x_3)$. Las bases más frecuentemente utilizadas son las bases lineales:

$$\mathbf{p}^T(\mathbf{x}) = [1, x_1] \text{ para 1D ; } m = 2 \quad (2.5)$$

$$\mathbf{p}^T(\mathbf{x}) = [1, x_1, x_2] \text{ para 2D ; } m = 3 \quad (2.6)$$

y las bases cuadráticas:

$$\mathbf{p}^T(\mathbf{x}) = [1, x_1, x_1^2] \text{ para 1D ; } m = 3 \quad (2.7)$$

$$\mathbf{p}^T(\mathbf{x}) = [1, x_1, x_2, x_1^2, x_1x_2, x_2^2] \text{ para 2D ; } m = 6 \quad (2.8)$$

Para obtener los coeficientes $\mathbf{a}(\mathbf{x})$ construimos una funcional con norma L_2 entre la aproximación local y la función:

$$\begin{aligned} J(\mathbf{x}) &= \sum_{j=1}^n w_j(\mathbf{x}) [u^h(\mathbf{x}) - u(\mathbf{x}_j)]^2 \\ &= \sum_{j=1}^n w_j(\mathbf{x}) [\mathbf{p}^T(\mathbf{x}_j) \mathbf{a}(\mathbf{x}) - u_j]^2 \end{aligned} \quad (2.9)$$

donde \mathbf{x}_j son las posiciones de los n nodos que se encuentran alrededor de la vecindad del punto \mathbf{x} , esto es, dentro del subdominio $\Omega_{\mathbf{x}}$; $w_j(\mathbf{x})$ son funciones de peso de soporte compacto asociada a cada nodo (que se discuten más adelante en ésta misma sección) y u_j son los valores de la función u evaluada en cada nodo (ver fig. 2.2 para el caso 1D).

La ecuación (2.9) puede escribirse en forma matricial como:

$$J(\mathbf{x}) = (\mathbf{Pa} - \mathbf{u})^T \mathbf{w}(\mathbf{x}) (\mathbf{Pa} - \mathbf{u}) \quad (2.10)$$

donde:

$$\mathbf{u}^T = (u_1, u_2, \dots, u_n) \quad (2.11)$$

$$\mathbf{P} = \begin{bmatrix} \rho_1(\mathbf{x}_1) & \rho_2(\mathbf{x}_1) & \dots & \rho_m(\mathbf{x}_1) \\ \rho_1(\mathbf{x}_2) & \rho_2(\mathbf{x}_2) & \dots & \rho_m(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_1(\mathbf{x}_n) & \rho_2(\mathbf{x}_n) & \dots & \rho_m(\mathbf{x}_n) \end{bmatrix} \quad (2.12)$$

$$\mathbf{w}(\mathbf{x}) = \begin{bmatrix} w_1(\mathbf{x}) & 0 & \dots & 0 \\ 0 & w_2(\mathbf{x}) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_n(\mathbf{x}) \end{bmatrix} \quad (2.13)$$

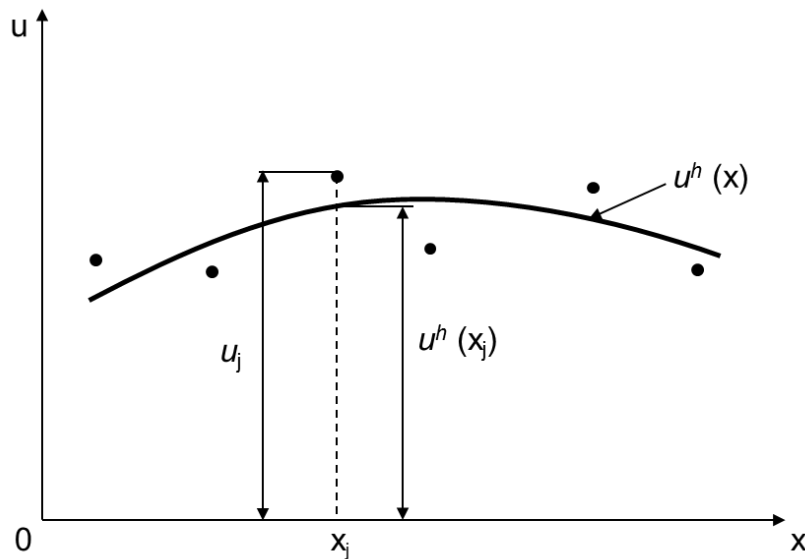


Figura 2.2 La función de aproximación $u^h(\mathbf{x})$ y los parámetros nodales u_j en la aproximación por mínimos cuadrados móviles (1D).

Para encontrar los coeficientes $\mathbf{a}(\mathbf{x})$, calculamos las derivadas de $J(\mathbf{x})$ con respecto a $\mathbf{a}(\mathbf{x})$ y las igualamos a cero, esto es:

$$\frac{\partial J(\mathbf{x})}{\partial \mathbf{a}} = 0 \quad (2.14)$$

Aplicamos este procedimiento a la ecuación (2.9) y obtenemos:

$$\sum_{j=1}^n w_j(\mathbf{x}) \mathbf{p}(\mathbf{x}_j) [\mathbf{p}^T(\mathbf{x}_j) \mathbf{a}(\mathbf{x}) - u_j] = 0$$

ó

$$\sum_{j=1}^n w_j(\mathbf{x}) \mathbf{p}(\mathbf{x}_j) \mathbf{p}^T(\mathbf{x}_j) \mathbf{a}(\mathbf{x}) = \sum_{j=1}^n w_j(\mathbf{x}) \mathbf{p}(\mathbf{x}_j) u_j \quad (2.15)$$

que puede reescribirse en forma matricial como:

$$\mathbf{M}(\mathbf{x})\mathbf{a}(\mathbf{x}) = \mathbf{B}(\mathbf{x})\mathbf{u} \quad (2.16)$$

donde $\mathbf{M}(\mathbf{x})$ es llamada *matriz de momento* y está definida como:

$$\mathbf{M}(\mathbf{x}) = \mathbf{P}^T \mathbf{w}(\mathbf{x}) \mathbf{P} \quad (2.17)$$

y

$$\mathbf{B}(\mathbf{x}) = \mathbf{P}^T \mathbf{w}(\mathbf{x}) \quad (2.18)$$

Por lo tanto de la ec. (2.16):

$$\mathbf{a}(\mathbf{x}) = [\mathbf{M}(\mathbf{x})]^{-1} \mathbf{B}(\mathbf{x}) \mathbf{u} \quad (2.19)$$

y reemplazando esta última en la ecuación (2.4) tenemos:

$$u^h(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) [\mathbf{M}(\mathbf{x})]^{-1} \mathbf{B}(\mathbf{x}) \mathbf{u} \quad (2.20)$$

Finalmente, la aproximación u^h puede ser expresada de manera semejante a la ec. (2.2):

$$u^h(\mathbf{x}) = \sum_{j=1}^n \Phi_j(\mathbf{x}) u_j = \Phi^T(\mathbf{x}) \mathbf{u} \quad (2.21)$$

de donde obtenemos las funciones de expansión MLS:

$$\Phi^T(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) [\mathbf{M}(\mathbf{x})]^{-1} \mathbf{B}(\mathbf{x}) \quad (2.22)$$

Para evaluar estas funciones deben definirse puntos en donde se desea conocer su valor empleando un código numérico para invertir la matriz $\mathbf{M}(\mathbf{x})$. Más adelante se verá que estos puntos corresponden a los puntos de integración por cuadratura de Gauss-Lobatto-Legendre. En la literatura, esto es conocido como “evaluación digital de una función de expansión”, ya que no conocemos su forma explícita como las usadas en el método del elemento finito (Bathe and Wilson, 1976).

Las funciones de expansión Φ solo están definidas cuando la matriz $\mathbf{M}(\mathbf{x})$ es no singular. Una condición para ello, es que al menos m funciones de peso sean diferentes de cero para cada punto $\mathbf{x} \in \Omega_x$ y que los nodos en Ω_x no esten distribuidos con algún arreglo especial, por ejemplo, en una línea recta (Fries and Matthies, 2004).

Las primeras derivadas parciales de las funciones de expansión MLS se obtienen con la regla del producto (Fries and Matthies, 2004):

$$\Phi_{,k}^T(\mathbf{x}) = \mathbf{p}_{,k}^T \mathbf{M}^{-1} \mathbf{B} + \mathbf{p}^T \mathbf{M}_{,k}^{-1} \mathbf{B} + \mathbf{p}^T \mathbf{M}^{-1} \mathbf{B}_{,k} \quad (2.23)$$

en donde $\mathbf{M}_{,k}^{-1} = -\mathbf{M}^{-1} \mathbf{M}_{,k} \mathbf{M}^{-1}$ y los índices precedidos por una coma representan derivadas parciales, esto es, $(\)_{,k} = \partial(\) / \partial x_k$.

La suavidad de las funciones de expansión $\Phi(\mathbf{x})$ está determinada por las funciones base $\mathbf{p}(\mathbf{x})$ y las funciones de peso w_j que se elijan (ver subsec. 2.2.1).

La fig. 2.3 muestra las funciones de expansión MLS en un dominio unidimensional $\Omega = [0, 1]$ con 11 nodos uniformemente distribuidos. La función de peso que se empleó tiene un parámetro de suavizado $d_{\max} = 2.3\Delta x = 0.23$ (ver subsec. 2.2.1). La fig. 2.4 muestra la primera derivada de las funciones de expansión MLS. Con respecto a estas gráficas podemos hacer las siguientes observaciones:

1. La línea roja en la fig. 2.3 es la sumatoria de las funciones de expansión $\sum_i \Phi_i(\mathbf{x})$ y ésta es igual a 1 en todo el dominio. Cuando esto ocurre se dice que las funciones $\{\Phi_i\}$ forman una partición de unidad (*Partition Unity*, PU). Las primeras y las segundas derivadas de las funciones de expansión MLS, forman una partición de nulidad (*Partition of Nullities*, PNs), o sea, $\sum_i \Phi_{i,x}(\mathbf{x}) = \sum_i \Phi_{i,xx}(\mathbf{x}) = 0$. Esto es muy importante porque es una condición que debe cumplirse en cualquier punto x en donde sean evaluadas.
2. Las funciones de expansión MLS son suaves y pueden ser consideradas como polinomios para propósitos de integración de la forma débil de la EDP a resolver. Sin embargo, las segundas derivadas pierden esa característica, no siendo así para las primeras derivadas que siguen siendo suaves.
3. En cada nodo existe por lo menos una función de expansión MLS cuyo valor no es la unidad, por lo que se dice que estas funciones son no interpolantes o que no poseen la propiedad de la delta de Kronecker, como ocurre con las funciones de expansión utilizadas en el método de los elementos finitos. Esto hace que en ocasiones sea complicado el tratamiento las condiciones de frontera.

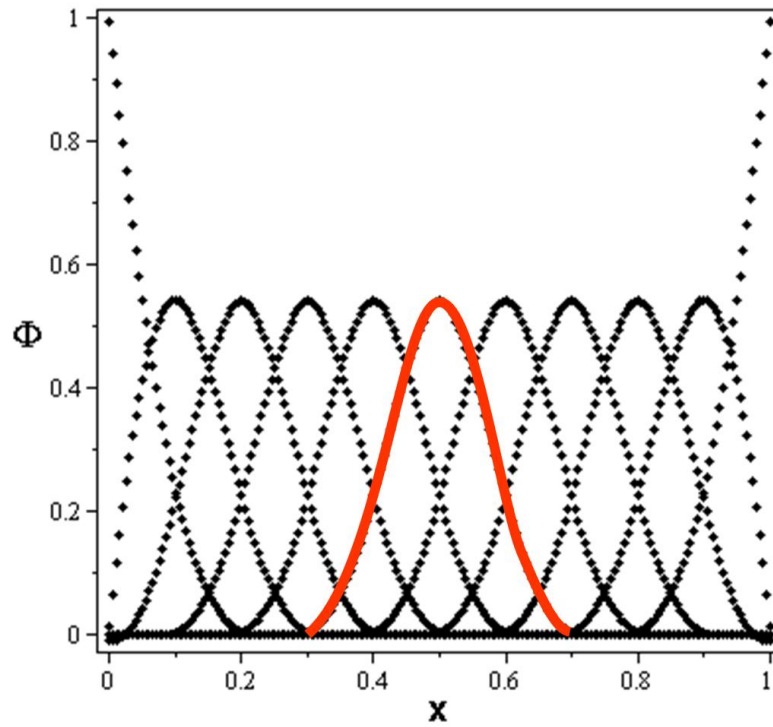


Figura 2.3 Funciones de interpolación basadas en el método de los MLS.

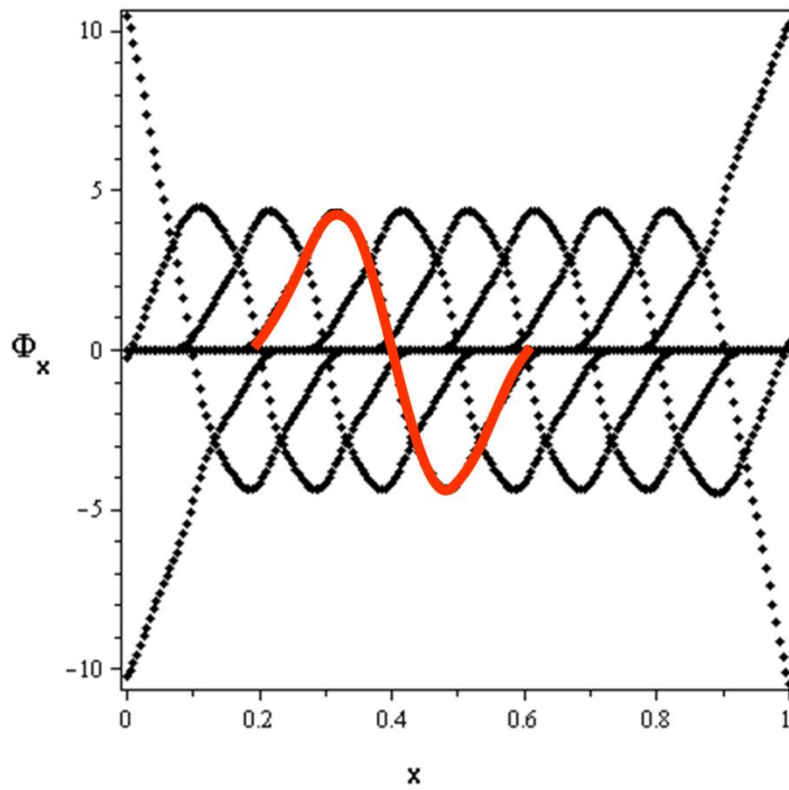


Figura 2.4 Primeras derivadas de las funciones de interpolación basadas en el método MLS.

2.2.1. Funciones de peso

La selección correcta de la función de peso tiene un papel muy importante en la implementación numérica del Método Local de Petrov-Galerkin entre otras cosas porque es ésta quien le da el carácter local al método, como lo indica su propio nombre. Además, por medio de ella, podemos seleccionar la forma de los subdominios de integración de la forma débil y se emplea tanto para la construcción de las funciones de expansión Φ al igual que se emplea como función de peso Ψ en el proceso de residuos pesados para obtener el Método Local de Petrov-Galerkin.

Existe una extensa lista de funciones de peso que se usan comúnmente, pero la usada en este trabajo será una función de peso “spline” de cuarto orden. Primeramente se presentará en una dimensión y posteriormente la manera en que ésta función puede aplicarse al caso bidimensional.

La función de peso “spline” de cuarto orden en una dimensión está dada por:

$$w_j(x) = \begin{cases} 1 - 6d^2 + 8d^3 - 3d^4 & d \leq 1 \\ 0 & d > 1 \end{cases} \quad (2.24)$$

donde:

$d = |(x-x_i)| / d_{\max}$ es la distancia que va del punto de interés x a la posición del nodo x_j dividida por el parámetro de suavizado d_{\max} . Este último es el que define el subdominio para el cual $w_j \neq 0$ y por ello las funciones de peso son llamadas funciones de soporte compacto. Las figuras 2.5 y 2.6 muestran las curvas correspondientes a la ec. (2.24) para diferentes parámetros de suavizado.

Para obtener funciones de peso n -dimensionales podemos definir funciones del tipo $w(\|\mathbf{x}\|)$ en donde los dominios son de forma circular (2D) y esférica (3D) o bien definir un producto tensorial:

$$w_j(\mathbf{x}) = \prod_{i=1}^n w_j(x_i) \quad (2.25)$$

del que resulta un dominio de forma rectangular (2D) que son los dominios utilizados en este trabajo (ver fig. 2.7) o de un paralelepípedo (3D). Es importante señalar que en el método local de Petrov-Galerkin cada nodo tiene una función de peso asociada, la cual determina la “influencia” o “peso” del nodo x_i dentro de Ω . En este trabajo el subdominio asociado a cada nodo es designado por Ω_{Q_i} y su frontera por Γ_{Q_i} . En este subdominio es donde se fijan los puntos de cuadratura para integrar la forma débil de la EDP (subsecc. 2.3.1).

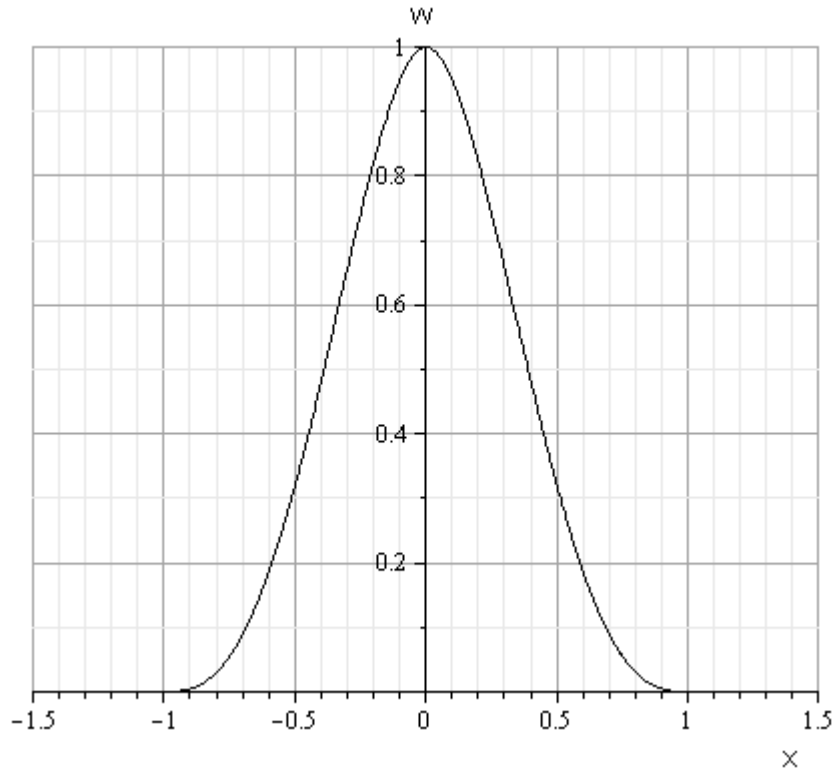


Figura 2.5 Función de peso “spline” de cuarto orden de soporte compacto para $d_{\max}=1.01$

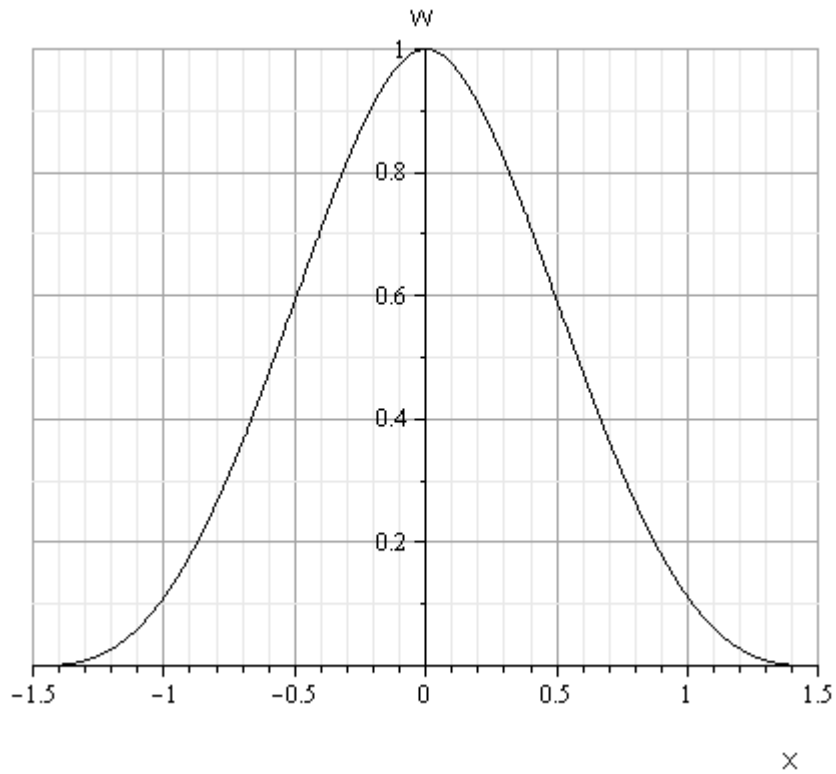


Figura 2.6 Función de peso “spline” de cuarto orden de soporte compacto para $d_{\max}=1.5$

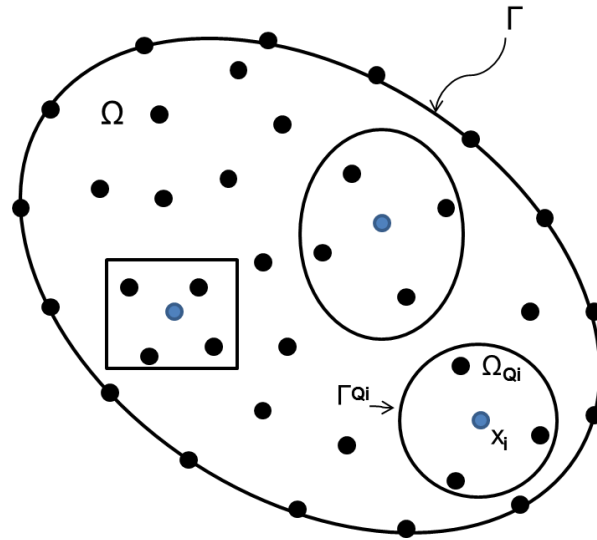


Figura 2.7 Formas que pueden tener los subdominios de las funciones de peso asociadas a cada nodo x_i : rectangular, circular o elíptico (2D). El nodo de referencia x_i se muestra en azul.

2.3. Aplicación del Método de Petrov-Galerkin: Ecuación de difusión de calor.

En esta sección se presenta el procedimiento para aplicar el método de Petrov-Galerkin para resolver la ecuación de difusión de calor en estado transitorio en dos dimensiones, empleando coordenadas cartesianas. Este procedimiento es aplicable a cualquier ecuación diferencial parcial, sin embargo, esta ecuación nos permite mostrar cómo se lleva a cabo la discretización temporal, que podría aplicarse en otros trabajos futuros tales como las ecuaciones de Navier-Stokes para flujos en estado transitorio.

Como se mencionó anteriormente, éste método puede formularse a partir del método de los residuos pesados, por lo tanto seguiremos con detalle los pasos mencionados en la sec. 2.1. La ecuación de difusión de calor en el sistema de coordenadas cartesianas rectangulares cuando las propiedades del material son constantes, puede escribirse como (Cervantes, 1999):

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{Q'}{k} = \frac{1}{\alpha} \cdot \frac{\partial T}{\partial t} \quad (2.26)$$

donde $T \equiv T(x,y,t)$ es el campo de temperaturas (K); α es la difusividad térmica (m^2/s); k es la conductividad térmica ($W/m \cdot K$); $Q' = Q'(x,y,t)$ es un término fuente (W/m^3) y t es el tiempo (s). Se tienen las siguientes condiciones de frontera:

$$T = \bar{T} \text{ en } \Gamma \text{ para } t > 0 \quad (2.27)$$

$$-k \nabla T \cdot \mathbf{n} = \bar{q} \text{ en } \Gamma \text{ para } t > 0 \quad (2.28)$$

donde Γ es una superficie en donde se especifica un campo de temperaturas o un flujo de calor por unidad de superficie y \mathbf{n} es el vector normal unitario a la superficie Γ . La condición inicial está dada por:

$$T = T_0 \text{ para } t = 0 \text{ en } \Omega \quad (2.29)$$

donde T_0 es el campo inicial de temperaturas y Ω es el dominio del problema.

2.3.1. La forma débil

A continuación se obtiene la forma débil de la ecuación de difusión de calor. Comenzamos proponiendo una aproximación al campo de temperaturas, $T \approx T^h = T^h(x,y,t)$ y la reemplazamos en la ecuación principal para obtener un error residual r dado por:

$$r = \frac{1}{\alpha} \cdot \frac{\partial T^h}{\partial t} - \left(\frac{\partial^2 T^h}{\partial x^2} + \frac{\partial^2 T^h}{\partial y^2} \right) - \frac{Q'}{k} \quad (2.30)$$

En el método Petrov-Galerkin se definen subdominios locales de cuadratura Ω_{Qi} asociados a cada nodo que se encuentre completamente dentro del dominio Ω , es decir, no tomamos en cuenta los nodos que están en la frontera Γ . La integral del residual pesado, en cada uno de los subdominios de cuadratura es igualada a cero, esto es (WRM, sec. 2.1):

$$\int_{\Omega_{Qi}} r \Psi_i \, d\Omega_{Qi} = 0 \quad (2.31)$$

donde: Ψ_i es una *función de prueba* (sec. 2.1) que sólo depende de las coordenadas espaciales y que es diferente de cero solamente dentro de Ω_{Qi} . Sustituyendo la ec. (2.30) en la ec. (2.31), obtenemos:

$$\frac{1}{\alpha} \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial t} \Psi_i d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial y^2} \Psi_i d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{Q'}{k} \Psi_i d\Omega_{Qi} = 0$$

A continuación integramos por partes el segundo término de la expresión anterior:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i d\Omega_{Qi} = \int_{\Omega_{Qi}} \frac{\partial}{\partial x} \left(\frac{\partial T^h}{\partial x} \Psi_i \right) d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} d\Omega_{Qi}$$

Y aplicamos el teorema de Gauss:

$$\int_V \frac{\partial}{\partial x} \lambda dV = \int_S \lambda n_1 dS, \quad \text{donde: } \lambda = \frac{\partial T^h}{\partial x} \Psi_i$$

Al primer término del lado derecho de la igualdad para obtener:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i d\Omega_{Qi} = \int_{\Gamma_{Qi}} \frac{\partial T^h}{\partial x} n_1 \Psi_i d\Gamma_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} d\Omega_{Qi} \quad (2.32)$$

donde Γ_{Qi} es la frontera del dominio local de cuadratura Ω_{Qi} y n_1 es la componente en la dirección x_1 del vector normal unitario \mathbf{n} a la superficie Γ_{Qi} . Aplicamos el mismo procedimiento al tercer término para obtener *la forma débil de la ecuación de difusión de calor*:

$$\int_{\Omega_{Qi}} \left(\frac{1}{\alpha} \frac{\partial T^h}{\partial t} \Psi_i + \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} - \frac{Q'}{k} \Psi_i \right) d\Omega_{Qi} - \int_{\Gamma_{Qi}} \left(\frac{\partial T^h}{\partial x} n_1 \Psi_i + \frac{\partial T^h}{\partial y} n_2 \Psi_i \right) d\Gamma_{Qi} = 0 \quad (2.33)$$

Ésta expresión es válida para los nodos que se encuentran completamente dentro del dominio del problema Ω y como puede verse, el método de Petrov-Galerkin establece una ecuación para cada nodo. Las integrales se evalúan numéricamente utilizando cuadratura de Gauss-Lobato-Legendre (GLL).

2.3.2. Discretización de la forma débil

Para la discretización de la forma débil de la ecuación de difusión, la función de aproximación T^h se expresa como la combinación lineal de las funciones de expansión Φ_j que son funciones únicamente de las coordenadas espaciales y de los valores nodales de las temperaturas a encontrar T_j las cuales dependerán únicamente del tiempo, esto es:

$$T^h(x, y, t) = \sum_{j=1}^n T_j(t) \Phi_j(x, y) \quad (2.34)$$

En donde n es el número de nodos vecinos a algún nodo dado i . De ésta expresión vemos que podemos escribir las derivadas temporales como:

$$\frac{\partial T^h}{\partial t} = \sum_{j=1}^n \frac{dT_j}{dt} \Phi_j \quad (2.35)$$

y las derivadas espaciales:

$$\frac{\partial T^h}{\partial x} = \sum_{j=1}^n T_j \frac{\partial \Phi_j}{\partial x} \quad \text{y} \quad \frac{\partial T^h}{\partial y} = \sum_{j=1}^n T_j \frac{\partial \Phi_j}{\partial y} \quad (2.36)$$

En este trabajo, la función de prueba asociada a cada nodo que se utiliza para la forma débil, es la misma función de peso w_i (ec. 2.24) que se utilizó para generar las funciones de expansión Φ vistas en la sección anterior, esto es:

$$\Psi_i = w_i = w_i(x, y) \quad (2.37)$$

Ahora sustituimos las ecs. (2.34)– (2.37) en la forma débil (2.33) para obtener:

$$\int_{\Omega_{Q_i}} \left(\sum_{j=1}^n \frac{1}{\alpha} \frac{dT_j}{dt} \Phi_j - \frac{Q'}{k} \right) w_i d\Omega_{Q_i}$$

$$\begin{aligned}
 & + \int_{\Omega_{Qi}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} \frac{\partial w_i}{\partial x} + T_j \frac{\partial \Phi_j}{\partial y} \frac{\partial w_i}{\partial y} \right) d\Omega_{Qi} \\
 & - \int_{\Gamma_{Qi}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} n_1 w_i + T_j \frac{\partial \Phi_j}{\partial y} n_2 w_i \right) d\Gamma_{Qi} = 0 \quad (2.38)
 \end{aligned}$$

Colocando las sumatorias fuera de las integrales y reordenando términos podemos reescribir la expresión anterior en forma de un sistema de ecuaciones lineales, que es válido para todos los nodos que están completamente dentro del dominio Ω . La forma matricial de la ec. (2.38) es:

$$\mathbf{CT}' + \mathbf{KT} = \mathbf{f} \quad (2.39)$$

en donde las matrices y vectores se definen como sigue. La matriz \mathbf{K} llamada en la literatura *matriz de rigideces* contiene los términos que son coeficientes de las derivadas espaciales, sus elementos son:

$$\begin{aligned}
 K_{ij} = & \int_{\Omega_{Qi}} \left[\frac{\partial \Phi_j}{\partial x} \frac{\partial w_i}{\partial x} + \frac{\partial \Phi_j}{\partial y} \frac{\partial w_i}{\partial y} \right] d\Omega_{Qi} \\
 & - \int_{\Gamma_{Qi}} \left[\frac{\partial \Phi_j}{\partial x} n_1 w_i + \frac{\partial \Phi_j}{\partial y} n_2 w_i \right] d\Gamma_{Qi} \quad (2.40)
 \end{aligned}$$

La matriz de amortiguamiento \mathbf{C} contiene los coeficientes de las derivadas temporales, sus elementos son:

$$C_{ij} = \frac{1}{\alpha} \int_{\Omega_{Qi}} \Phi_j w_i d\Omega_{Qi} \quad (2.41)$$

El vector de cargas \mathbf{f} contiene la integral del término fuente:

$$f_i = \frac{1}{k} \int_{\Omega_{Qi}} \Phi' w_i d\Omega_{Qi} \quad (2.42)$$

De igual forma, los vectores incógnita \mathbf{T}' y \mathbf{T} están dados por:

$$T_j' = \frac{dT_j(t)}{dt}, \quad T_j = T_j(t) \quad (2.43)$$

respectivamente.

Para obtener el sistema lineal que habrá de resolverse es necesario llevar a cabo la discretización temporal de la forma débil.

2.3.3. Discretización temporal

La derivada con respecto al tiempo es discretizado usando el esquema de Crank-Nicholson, como se describe en (Sterk, 2005). En este esquema se supone que los valores nodales de las temperaturas T_j son una función lineal del tiempo. Si usamos la notación vectorial, la razón de cambio del campo de temperaturas entre dos instantes de tiempo (t) y ($t + \Delta t$), puede escribirse como:

$$\mathbf{T}' = \frac{d\mathbf{T}^{(t+\Delta t/2)}}{dt} = \frac{\mathbf{T}^{(t+\Delta t)} - \mathbf{T}^{(t)}}{\Delta t} \quad (2.44)$$

Si sustituimos la ec. (2.44) en la ec. (2.39) en el tiempo (t), tenemos:

$$\mathbf{C} \left[\frac{\mathbf{T}^{(t+\Delta t)} - \mathbf{T}^{(t)}}{\Delta t} \right] + \mathbf{K}\mathbf{T}^{(t)} = \mathbf{f} \quad (2.45)$$

y para el tiempo ($t + \Delta t$):

$$\mathbf{C} \left[\frac{\mathbf{T}^{(t+\Delta t)} - \mathbf{T}^{(t)}}{\Delta t} \right] + \mathbf{K}\mathbf{T}^{(t+\Delta t)} = \mathbf{f} \quad (2.46)$$

Sumando las ecs. (2.45) y (2.46), y reordenando los términos, obtenemos el sistema final que habrá de resolverse para cada paso de tiempo:

$$\mathbf{A}\mathbf{T}^{(t+\Delta t)} = \mathbf{F}^{(t)} \quad (2.47)$$

donde: $\mathbf{A} = 2\mathbf{C} + \Delta t \mathbf{K}$ y $\mathbf{F}^{(t)} = (2\mathbf{C} - \Delta t \mathbf{K})\mathbf{T}^{(t)} + 2\Delta t \mathbf{f}$

Como se mencionó, este sistema es válido únicamente para los nodos que se encuentran completamente dentro del dominio del problema. A este sistema habrá que agregarle las ecuaciones de los nodos que están en la frontera. De esta forma **A** tiene un tamaño de $(N \times N)$, **T** de $(N \times 1)$ y **F** de $(N \times 1)$, en donde N es el número total de nodos en que se ha discretizado el dominio.

2.4. Implementación numérica del método local de Petrov-Galerkin.

El propósito de esta sección es presentar el procedimiento que se utilizó para desarrollar el código en Maple para resolver numéricamente la ecuación de difusión de calor en estado transitorio (2D) y que puede utilizarse para resolver cualquier otra EDP. De hecho, este mismo procedimiento es el que se aplica para resolver otras ecuaciones tales como las de Navier-Stokes (2D) para un flujo incompresible.

La implementación numérica del método puede llevarse a cabo de acuerdo al siguiente procedimiento:

1. Colocar cierto número de nodos N dentro del dominio físico Ω y sobre la frontera Γ . Se deben seleccionar las funciones base \mathbf{p}^T y la función de peso w_i para construir las funciones de expansión Φ .
2. Determinar el número de nodos vecinos n para cada uno de los nodos y de igual manera determinar los sub-dominios locales de integración Ω_{Q_i} para cada nodo.
3. Determinar el número de pasos de tiempo y el Δt del problema.
4. Hacer un ciclo sobre el número de pasos de tiempo.
 - Hacer un ciclo sobre los nodos que están completamente dentro de Ω
 - a) Determinar el número de puntos de cuadratura x_Q de GLL dentro del subdominio Ω_{Q_i} .
 - b) Hacer un ciclo sobre el número de puntos de cuadratura.
 - Generar las funciones MLS para cada nodo dentro del subdominio Ω_{Q_i} evaluadas en el punto x_Q .
 - Encontrar las derivadas de todas las funciones asociadas a cada nodo dentro de Ω_{Q_i} .
 - Formar la matriz **A** y los vectores **T** y **F**.
 - c) Terminar el ciclo sobre los puntos dentro de Ω_{Q_i} .

- d) Determinar el número de puntos de cuadratura x_Q de GLL sobre la frontera Γ_{Q_i} .
 - e) Hacer un ciclo sobre el número de puntos de cuadratura.
 - Generar las funciones MLS para cada nodo dentro del subdominio Ω_{Q_i} evaluadas en el punto x_Q .
 - Encontrar las derivadas de todas las funciones asociadas a cada nodo dentro de Ω_{Q_i} , además de los vectores normales unitarios a la superficie Γ_{Q_i} en cada punto x_Q .
 - Evaluar numéricamente las integrales de superficie que aparecen en la forma débil.
 - Sumar las contribuciones de estas integrales a la matriz **A**.
 - f) Terminar el ciclo sobre los nodos que se encuentran en Γ_{Q_i} .
- Terminar el ciclo para los nodos que están completamente en el dominio Ω .
 - Hacer un ciclo para los nodos que se encuentra sobre la frontera Γ y generar sus ecuaciones para agregarlas a la matriz **A**.
 - Resolver el sistema lineal $\mathbf{AT} = \mathbf{F}$ para el tiempo t .
 - Ir al siguiente paso de tiempo haciendo $t = (t + \Delta t)$.
5. Terminar el ciclo sobre el número de pasos de tiempo.

CAPÍTULO 3

APLICACIÓN NÚMERICA DEL MÉTODO LOCAL DE PETROV-GALERKIN

3.1. El Método del Elemento Finito

Para poder entender la base de las simulaciones numéricas y cómo discretizar una ecuación, se trabajó con el Método de los Elementos Finitos (M.E.F) mediante un código computacional antes de comenzar con el Método Local de Petrov-Galerkin

El problema físico se delimitó a calcular la distribución de temperaturas en una barra rectangular delgada de longitud 10 cm con calentamiento usando la ecuación de Poisson (ec. 3.1) en una dimensión y estado estacionario como se muestra en la figura 3.1. Las condiciones frontera son $T(0, t) = 40$ y $T(10, t) = 200$ además de una fuente de calor uniforme de $f(x)=10$.

$$\frac{\partial^2 T}{\partial x^2} = -f(x) \quad (3.1)$$

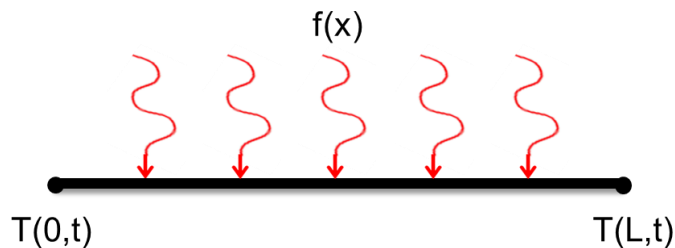


Figura 3.1 Barra larga y delgada sujeta a condiciones de frontera fijas y a una fuente continua de calor $f(x)$ a lo largo de su eje.

La solución analítica del problema para las condiciones de frontera dadas es:

$$T = -5x^2 + 66x + 40 \quad (3.2)$$

Los pasos del M.E.F que se siguieron son (Chapra and Canale, 2003):

I. Discretización: Este paso consiste en dividir el dominio de la solución en elementos finitos. Los puntos de intersección de las líneas que forman los lados de los elementos se conocen como *nodos*, y los mismos lados se denominan *líneas o planos nodales*.

II. Ecuaciones de los elementos: Para aproximar la solución de cada elemento se proponen funciones de aproximación T^h que se expresan por lo general como una combinación lineal de otras funciones llamadas “funciones de interpolación” o “funciones de expansión” representadas por ϕ . La figura 3.2 muestra la forma de las funciones de expansión de la ec (3.3) para cinco nodos.

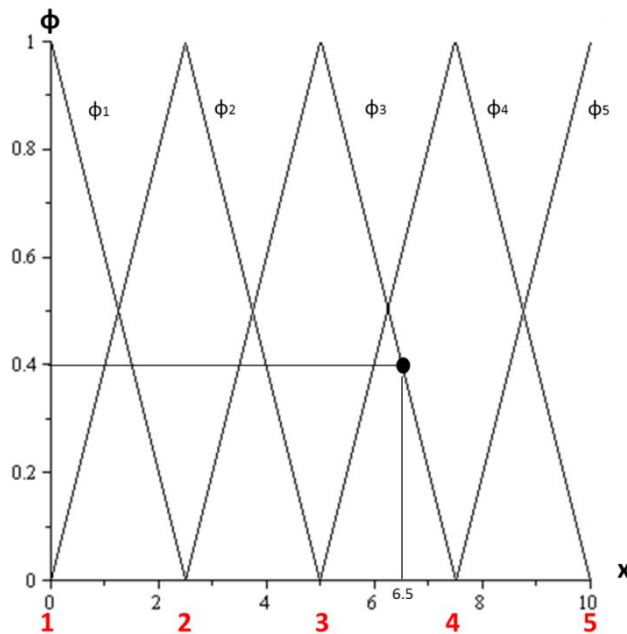


Figura 3.2 Funciones de expansión para cada elemento.

$$\Phi_j = \frac{(x - x_{j-1})}{(x_j - x_{j-1})} \quad y \quad \Phi_j = \frac{(x_{j+1} - x)}{(x_{j+1} - x_j)} \quad (3.3)$$

Se tiene la ecuación de Poisson:

$$\frac{\partial^2 T}{\partial x^2} + 10 = 0 \quad (3.4)$$

el residuo es:

$$r = \frac{\partial T^h}{\partial x^2} + 10 \quad (3.5)$$

El residuo (3.5) es “pesado” utilizando el método de los residuos pesados y se fuerza a que:

$$\int_{\Omega_i} r \Psi d\Omega_i = 0 \quad (3.6)$$

donde Ω es el dominio del problema, en nuestro ejemplo, $\Omega=[0,10]$. La función Ψ es la llamada “función de peso” o “función de prueba”.

- Método de colocación
- Método del subdominio
- Método de mínimos cuadrados
- Método de Galerkin $\Psi=\Phi$ (el más utilizado)

} Ψ

Entonces se usa el método de Galerkin para los pesos residuales y se tiene:

$$\Psi = \Phi \quad (3.7)$$

donde Φ es la función de expansión o de interpolación. además:

$$T^h = \sum_{j=1}^N \Phi_j T_j \quad (3.8)$$

Sustituyendo el valor del residuo “r” en (ec. 3.6):

$$\int_{\Omega_i} \left(\frac{\partial^2 T^h}{\partial x^2} + 10 \right) \Psi dx = 0 \quad (3.9)$$

Aplicando integración por partes en (ec. 3.9), usando la definición de T^h (ec.3.8) y el método de Galerkin (ec. 3.7), al final se obtiene:

$$\Phi_i \sum_{j=1}^N \Phi_j T_j \Big|_{\Omega_i} - \int_{\Omega_i} \sum_{j=1}^N \Phi_j T_j - \Phi_i dx = -10 \int_{\Omega_i} \Phi_i dx \quad (3.10)$$

donde:

i = nodo en estudio

N = número de nodos vecinos del nodo en estudio

j = valor de los nodos vecinos

Φ' = derivada de la función de expansión

T = es el valor de temperatura que toma la función en los nodos

Ω_i = dominio (o subdominio) del elemento en estudio

Para poder resolver la integral del producto de las derivadas de Φ que aparece en la ecuación 3.10, se hace uso del método del trapecio.

III. Ensamble y condiciones de frontera: Una vez obtenidas las ecuaciones de los elementos individuales (ec 3.10), éstas deben unirse o ensamblarse para caracterizar el comportamiento de todo el sistema. El proceso de ensamble está regido por el concepto de continuidad. Es decir, las soluciones de elementos contiguos se acoplan, de manera que los valores de las incógnitas (y algunas veces las derivadas) en sus nodos comunes sean equivalentes. Así, la solución total será continua.

El sistema completo se expresa de forma matricial como:

$$[K]\{T\} = \{F\} \tag{3.11}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ k_{21} & k_{22} & k_{23} & 0 & 0 \\ 0 & k_{32} & k_{33} & k_{34} & 0 \\ 0 & 0 & k_{43} & k_{44} & k_{45} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{bmatrix}$$

donde $[K]$ = una propiedad del elemento o matriz de rigidez, $\{T\}$ =vector columna de las incógnitas en los nodos y $\{F\}$ =vector columna determinado por el efecto de cualquier influencia externa aplicada a los nodos (condiciones de frontera y término fuente).

IV. Solución de las ecuaciones ensambladas: Para dar solución al problema se invierte la matriz K y se resuelve para T , donde cada valor será el valor de las temperaturas en los nodos.

$$\mathbf{T} = \mathbf{K}^{-1} \cdot \mathbf{F} \quad (3.12)$$

V. Verificar los resultados y representarlos en gráficas: Se puede usar algún software, en este caso Maple, para poder visualizar los datos numéricos obtenidos.

3.2. Caso 1: Ecuación de Laplace

El método local de Petrov-Galerkin es aplicado a problemas físicos bidimensionales separados en tres casos diferentes. Los dos primeros casos nos servirán para probar el código computacional que se usará en la ecuación de difusión de calor esto debido a que las ecuaciones en estado estacionario poseen soluciones analíticas fáciles de verificar y de comparar con el resultado numérico del método.

Para verificar que el código computacional de Petrov- Galerkin es correcto, se empieza solucionando numéricamente la ecuación de Laplace bidimensional que es una EDP en estado estacionario y sin término fuente.

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (3.13)$$

Se obtiene la forma débil de la ecuación usando los métodos WRM y el método de Petrov-Galerkin como se vio en la sección 2.3.1:

$$\int_{\Omega_{Q_i}} \left(\frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} \right) d\Omega_{Q_i} - \int_{\Gamma_{Q_i}} \left(\frac{\partial T^h}{\partial x} n_1 \Psi_i + \frac{\partial T^h}{\partial y} n_2 \Psi_i \right) d\Gamma_{Q_i} = 0 \quad (3.14)$$

Discretizando la forma débil se obtiene la ecuación de la cual se obtendrán la matriz de rigideces \mathbf{K} que contiene los términos que son coeficientes de las derivadas espaciales; el vector de cargas \mathbf{F} y el vector incógnita \mathbf{T} :

$$\int_{\Omega_{Q_i}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} \frac{\partial w_i}{\partial x} + T_j \frac{\partial \Phi_j}{\partial y} \frac{\partial w_i}{\partial y} \right) d\Omega_{Q_i}$$

$$-\int_{\Gamma_{Q_i}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} n_1 w_i + T_j \frac{\partial \Phi_j}{\partial y} n_2 w_i \right) d\Gamma_{Q_i} = 0 \quad (3.15)$$

3.2.1. Condiciones de Dirichlet

Para este caso se estudiará una barra larga de sección transversal rectangular que tiene 0.6 m de ancho, 0.9 m de largo, y una conductividad térmica de 1 W/m·K. Uno de sus anchos está sometido a un proceso de convección con aire y se encuentra a 100 °C. El resto de los lados se mantiene a 50 °C. Se encontrará la distribución de temperaturas en toda la sección transversal.

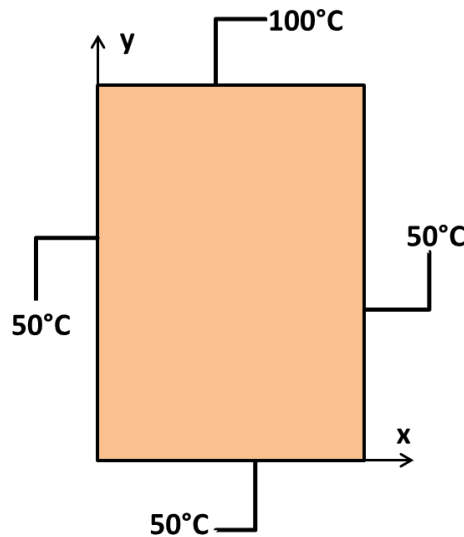


Figura 3.3 Problema físico de la ecuación de Laplace con condiciones de Dirichlet.

Las condiciones de frontera del problema son:

$$T(0,y)=50 \quad T(0.6,y)=50 \quad (3.16)$$

$$T(x,0)=50 \quad T(x,0.9)=100 \quad (3.17)$$

El dominio Ω está determinado por las dimensiones de la sección transversal de la barra. Para una mejor visualización en los resultados se trabajará arriba de los 40x40 nodos los cuales formaran los subdominios locales de cuadratura Ω_{Q_i} .

Antes de que el código computacional calcule la distribución de temperaturas en la sección, se necesitan datos tales como el número de nodos a trabajar, condiciones de frontera, número de puntos de integración por cuadratura de Gauss-

Lobatto-Legendre la cual depende del valor que asignemos para mayor precisión en la evaluación de las funciones de expansión Φ , entre otros. Para ver el algoritmo y los pasos que se deben de seguir para obtener los resultados ver sección 2.4.

Los resultados de la solución numérica se pueden comparar con la solución analítica de la ecuación de Laplace en estado estacionario bidimensional con condiciones de Dirichlet (Zill,2006).

$$u(x, y) = \sum_{n=1}^{\infty} A_n \sinh \frac{n\pi}{a} y \sin \frac{n\pi}{a} x \quad (3.18)$$

donde: $A_n = \frac{2}{a \sinh \frac{n\pi b}{a}} \int_0^a f(x) \sin \frac{n\pi}{a} x dx$; a es el valor final del dominio en x, b es el valor final del dominio en y $f(x)$ es la temperatura en alguna de las fronteras.

3.2.2. Condiciones de Neumann

Un problema parecido se resuelve pero con condiciones de Neumann.

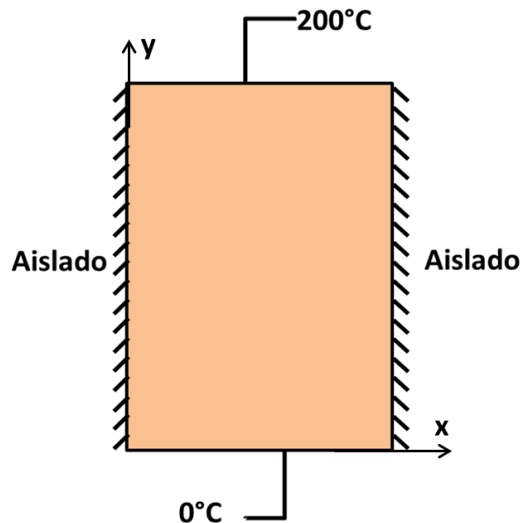


Figura 3.4 Problema físico de la ecuación de Laplace con condiciones de Neumann.

Las condiciones de frontera del problema son:

$$\left. \frac{\partial T}{\partial x} \right|_{x=0} = 0 \quad \left. \frac{\partial T}{\partial x} \right|_{x=0.6} = 0 \quad (3.19)$$

$$T(x,0)=0 \quad T(x,0.9)=200 \quad (3.20)$$

Para resolver las condiciones frontera con aislamiento se puede tomar la condición 3.19 como una función de aproximación:

$$\left. \frac{\partial T}{\partial x} \right|_{x=0.6} = 0 \approx T^h = \sum_{j=1}^N \Phi_j(x, y) \cdot T_j \quad (3.21)$$

Los resultados de la solución numérica también se pueden comparar con la solución analítica de la ecuación de Laplace bidimensional con condiciones de Neumann (Zill, 2006).

$$u(x, y) = A_0 y + \sum_{n=1}^{\infty} A_n \sinh \frac{n\pi}{a} y \cos \frac{n\pi}{a} x \quad (3.22)$$

donde:

$$A_0 = \frac{1}{ab} \int_0^a f(x) dx \quad y \quad A_n = \frac{2}{a \sinh \frac{n\pi b}{a}} \int_0^a f(x) \cos \frac{n\pi}{a} x dx$$

3.2.3. Ecuación de Laplace en estado transitorio

Como ya se ha trabajado en problemas estacionarios y se ha comprobado las soluciones analíticas, se resolverá la ecuación de Laplace de forma transitoria con condiciones de Dirichlet. Así la ecuación 3.13 se convierte en:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = \frac{1}{\alpha} \cdot \frac{\partial T}{\partial t} \quad (3.23)$$

El ejemplo será una placa de cobre de 1 m de longitud y 0.50 m de ancho que está inicialmente a una temperatura uniforme de 20 °C. Las caras de los extremos se someten de pronto a una intensa rapidez de calentamiento que las eleva a una temperatura de 500 °C. Al mismo tiempo, una de las caras se somete a calentamiento por un flujo de gas con una temperatura de 500 °C y un coeficiente de transferencia de calor de 100 W/m²·K. Se determinará la temperatura en el punto central de la placa 8 segundos después de la aplicación súbita del calor.

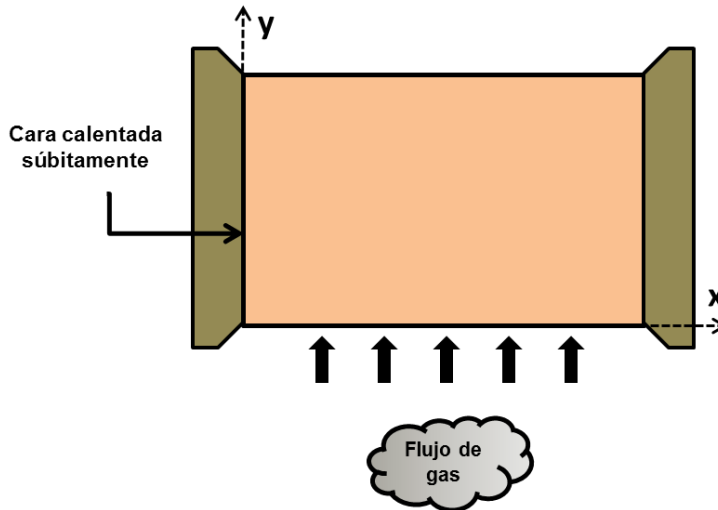


Figura 3.5 Problema físico de la ecuación de Laplace en estado transitorio.

Las condiciones de frontera del problema son:

$$T(0,y)=500 \quad T(1,y)=500 \quad (3.24)$$

$$T(x,0)=500 \quad T(x,0.5)=20 \quad (3.25)$$

3.3. Caso 2: Ecuación de Poisson

Aunque el caso anterior nos ayudaba a validar el código de Petrov-Galerkin no poseía término fuente, por lo que se resolverá un ejemplo con la ecuación de Poisson bidimensional que es una versión de la ecuación de difusión de calor pero en estado estacionario.

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = -\frac{Q'}{k} \quad (3.26)$$

Se obtiene la forma débil de la ecuación 3.26 (sección 2.3.1):

$$\int_{\Omega_{Qi}} \left(\frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} - \frac{Q'}{k} \Psi_i \right) d\Omega_{Qi} - \int_{\Gamma_{Qi}} \left(\frac{\partial T^h}{\partial x} n_1 \Psi_i + \frac{\partial T^h}{\partial y} n_2 \Psi_i \right) d\Gamma_{Qi} = 0 \quad (3.27)$$

y discretizando esta forma débil se obtiene:

$$\int_{\Omega_Q} -\frac{Q'}{k} w_i d\Omega_{Qi} + \int_{\Omega_{Qi}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} \frac{\partial w_i}{\partial x} + T_j \frac{\partial \Phi_j}{\partial y} \frac{\partial w_i}{\partial y} \right) d\Omega_{Qi} - \int_{\Gamma_{Qi}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} n_1 w_i + T_j \frac{\partial \Phi_j}{\partial y} n_2 w_i \right) d\Gamma_{Qi} = 0 \quad (3.28)$$

El problema para este caso es un análisis estacionario de la transferencia de calor a través de la pared de un horno de concreto ($k=1 \text{ W/m}\cdot\text{K}$) de 6m de largo y 4m de ancho como lo muestra la figura 3.6. Dentro del horno se generan 100 W/m^3 donde existe aire caliente. Las paredes exteriores del horno se encuentran a una temperatura de 30°C . Se determinará la distribución de temperaturas a través de la pared del horno considerando las propiedades térmicas del mismo.

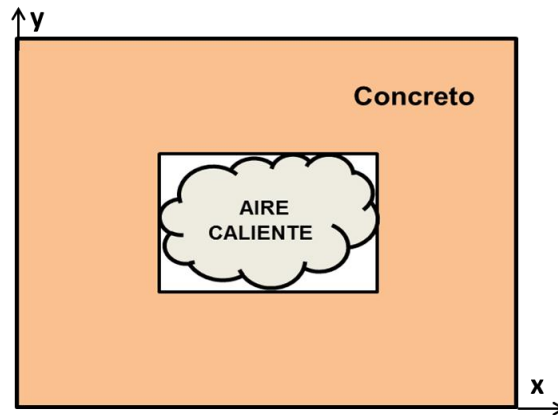


Figura 3.6 Problema físico de la ecuación de Poisson.

Las condiciones de frontera del problema son:

$$T(0,y)=30 \quad T(6,y)=30 \quad (3.29)$$

$$T(x,0)=30 \quad T(x,4)=30 \quad (3.30)$$

La solución analítica de la ecuación de Poisson bidimensional con condiciones de Dirichlet se presenta (Carslaw,1959):

$$u(x, y) = \frac{Q(a^2 - x^2)}{2k} - \frac{16Qa^2}{k\pi^3} \cdot \sum_{n=0}^{\infty} \frac{(-1)^n \cos\left[\frac{(2n+1)\pi x}{2a}\right] \cosh\left[\frac{(2n+1)\pi y}{2a}\right]}{(2n+1)^3 \cosh\left[\frac{(2n+1)\pi b}{2a}\right]} \quad (3.31)$$

donde: Q es el término fuente; a y b son los dominios en x y y; k es la conductividad térmica del material.

3.4. Caso 3: Ecuación de difusión de calor

Para este último caso se dará solución a la ecuación de difusión de calor (2D) que es una ecuación diferencial parcial (EDP) de segundo orden que describe la distribución del calor (o variaciones de la temperatura) en una región a lo largo del transcurso del tiempo con las condiciones de frontera de Dirichlet.

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{Q'}{k} = \frac{1}{\alpha} \cdot \frac{\partial T}{\partial t} \quad (3.32)$$

La forma débil de la ecuación de difusión de calor se vio de manera desarrollada en la sección 2.3.1

$$\int_{\Omega_{Qi}} \left(\frac{1}{\alpha} \frac{\partial T^h}{\partial t} \Psi_i + \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} - \frac{Q'}{k} \Psi_i \right) d\Omega_{Qi} - \int_{\Gamma_{Qi}} \left(\frac{\partial T^h}{\partial x} n_1 \Psi_i + \frac{\partial T^h}{\partial y} n_2 \Psi_i \right) d\Gamma_{Qi} = 0 \quad (3.33)$$

y discretizando la forma débil se tiene:

$$\int_{\Omega_{Qi}} \left(\sum_{j=1}^n \frac{1}{\alpha} \frac{dT_j}{dt} \Phi_j - \frac{Q'}{k} \right) w_i d\Omega_{Qi}$$

$$\begin{aligned}
 & + \int_{\Omega_{Q_i}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} \frac{\partial w_i}{\partial x} + T_j \frac{\partial \Phi_j}{\partial y} \frac{\partial w_i}{\partial y} \right) d\Omega_{Q_i} \\
 & - \int_{\Gamma_{Q_i}} \sum_{j=1}^n \left(T_j \frac{\partial \Phi_j}{\partial x} n_1 w_i + T_j \frac{\partial \Phi_j}{\partial y} n_2 w_i \right) d\Gamma_{Q_i} = 0 \quad (3.34)
 \end{aligned}$$

Para aplicar el programa computacional se resolverá un problema de un receptor solar de cobre ($k=372.1 \text{ W/m}\cdot\text{K}$) en forma de placa cuadrada de 8m de lado que recibe energía de un heliostato, aportando una generación de calor de 5000 W/m^3 . Las caras del receptor se encuentran enfriadas a una temperatura de 40°C . Se simulará el comportamiento de las temperaturas en el dominio debido a la transferencia de calor transcurrido un tiempo.

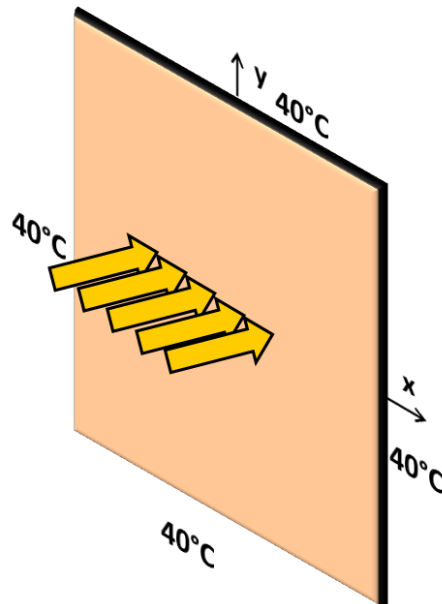


Figura 3.7 Receptor con generación de calor debido a la energía solar recibida de un heliostato

Este problema físico se puede resolver de manera numérica con el algoritmo de la ecuación de difusión de calor que se planteó en el capítulo 2.

CAPÍTULO 4

RESULTADOS

En este capítulo se presentan los resultados obtenidos utilizando el método local de Petrov-Galerkin para la solución de la ecuación de Laplace, Poisson y difusión de calor con condiciones de frontera homogéneas. Antes de trabajar con el método de Petrov-Galerkin, se determinó la distribución de temperaturas de una barra calentada (ecuación de Poisson unidimensional y en estado estacionario) usando el Método del Elemento Finito. Los problemas y los resultados numéricos del Método Local de Petrov-Galerkin fueron presentados en tres casos; en el primer caso, se determinó la distribución de temperaturas en la sección transversal de una barra (condiciones de Dirichlet y Neumann) y también se solucionó el problema de una placa de cobre en estado transitorio; en el segundo caso, se determinó la variación de temperatura a través de las paredes de un horno de concreto; el tercer caso trató de un receptor solar con generación de calor y su distribución de temperaturas a lo largo del tiempo.

4.1. Resultados numéricos de la ecuación de Poisson unidimensional usando el M.E.F

Los resultados obtenidos del problema de la barra calentada (ecuación de Poisson en estado estacionario y unidimensional) sirvieron para entender la base de las simulaciones numéricas y cómo discretizar una ecuación. Las figuras 4.1 y 4.2 muestran las temperaturas ($^{\circ}\text{C}$) a lo largo de la barra calentada con condiciones de frontera constantes y con generación de calor. Se puede observar que la solución analítica de la ecuación de Poisson unidimensional (línea azul continua) y la solución numérica del M.E.F (puntos rojos) son casi iguales, ya sea que se trabajen para pocos o muchos nodos. El código numérico del M.E.F se basó en los pasos que se vieron en la sección 3.1.

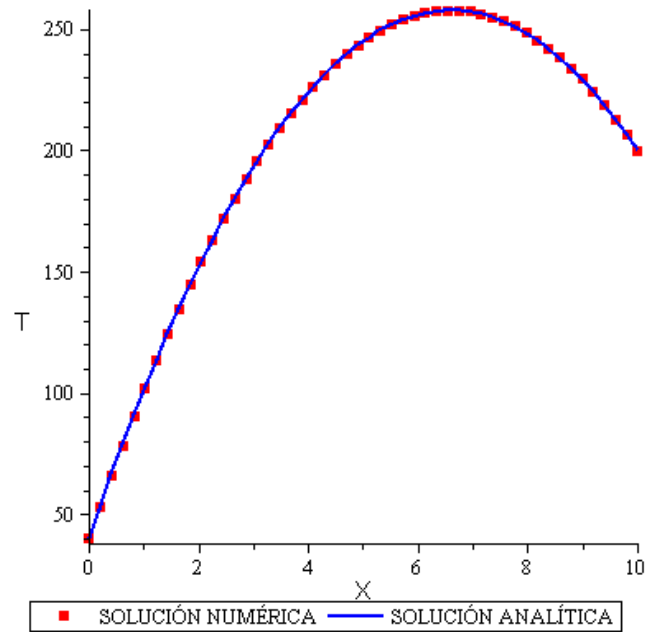


Figura 4.1 Distribución de temperaturas de la barra calentada (ecuación de Poisson unidimensional) usando el M.E.F para 50 nodos.

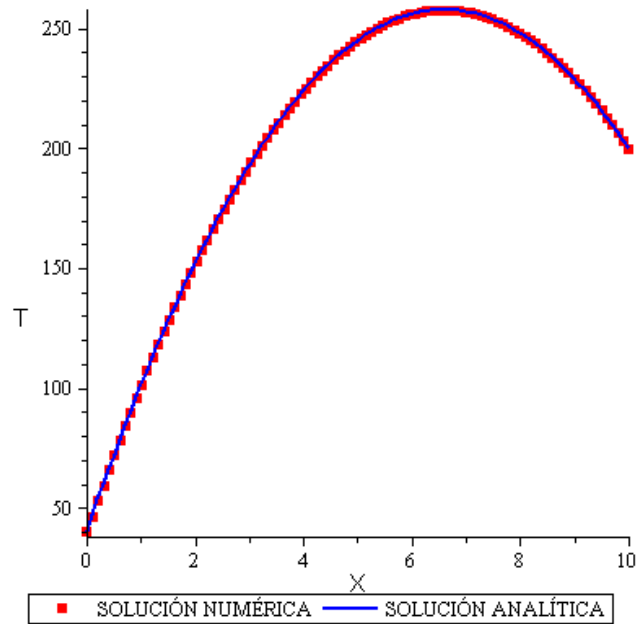


Figura 4.2 Distribución de temperaturas de la barra calentada (ecuación de Poisson unidimensional) usando el M.E.F para 100 nodos.

4.2. Resultados numéricos de la ecuación de Laplace con condiciones frontera de Dirichlet y Neumann.

4.2.1. Condiciones de frontera de Dirichlet

El problema para este caso, fue el de la sección transversal de una barra con una cara calentada a 100°C y las otras a 50°C . Se usó el código computacional del método de Petrov-Galerkin introduciendo los datos necesarios. Con esto se logra conocer las temperaturas ($^{\circ}\text{C}$) en puntos específicos de la sección estudiada. La figura 4.3 muestra la distribución de temperaturas en un dominio unidimensional para $x=0.3$ m. En ésta figura se puede apreciar la comparación que se tiene entre la solución numerica generada por el programa computacional de Petrov-Galerkin y la solución analitica de la ecuación de Laplace unidimensional, todo esto con el fin de verificar los resultados y validar el código computacional del problema.

La figura 4.4 muestra el campo de temperaturas del problema generado por la solución analitica pero en forma bidimensional (sección 3.1.2). En la figura 4.5 se observan los resultados numéricos de las temperaturas de la sección transversal obtenidos por el Método Local de Petrov-Galerkin

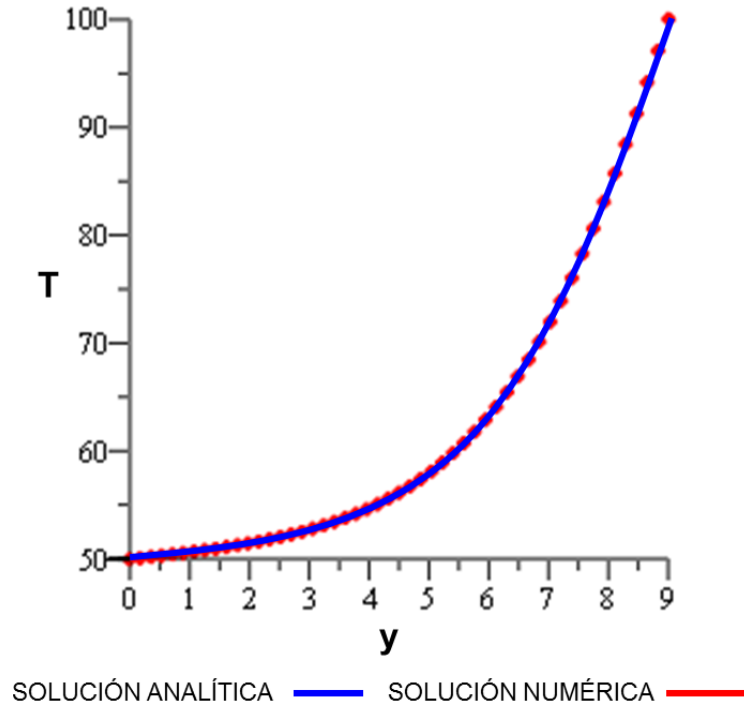


Figura 4.3 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Laplace con condiciones de Dirichlet).

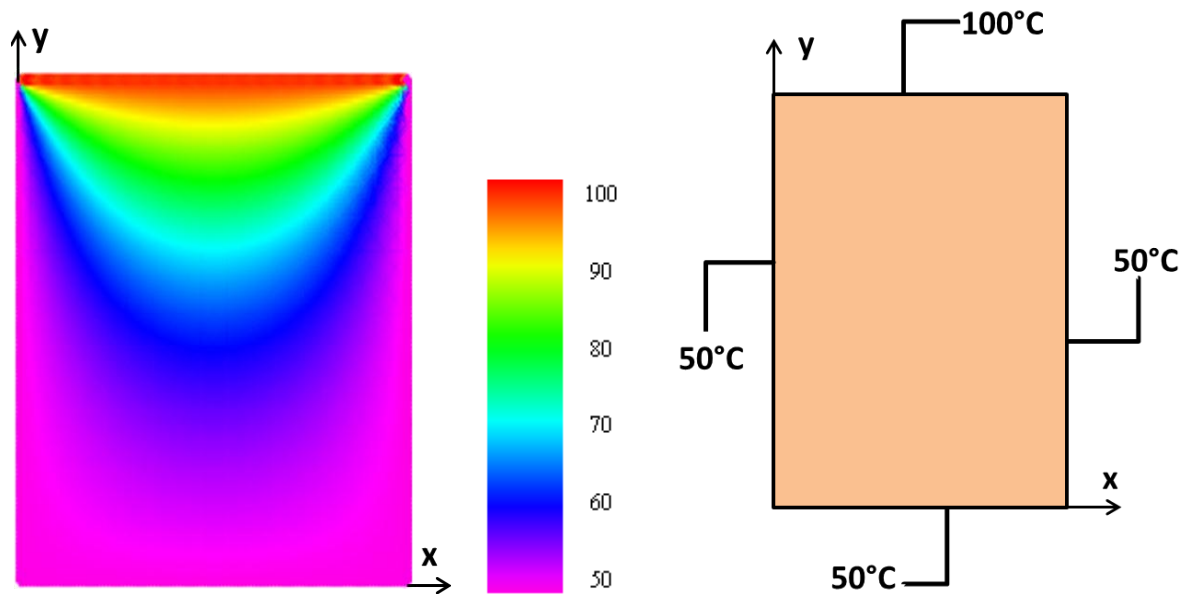


Figura 4.4 Distribución de temperaturas del problema generada por la solución analítica (ecuación de Laplace bidimensional con condiciones de Dirichlet).

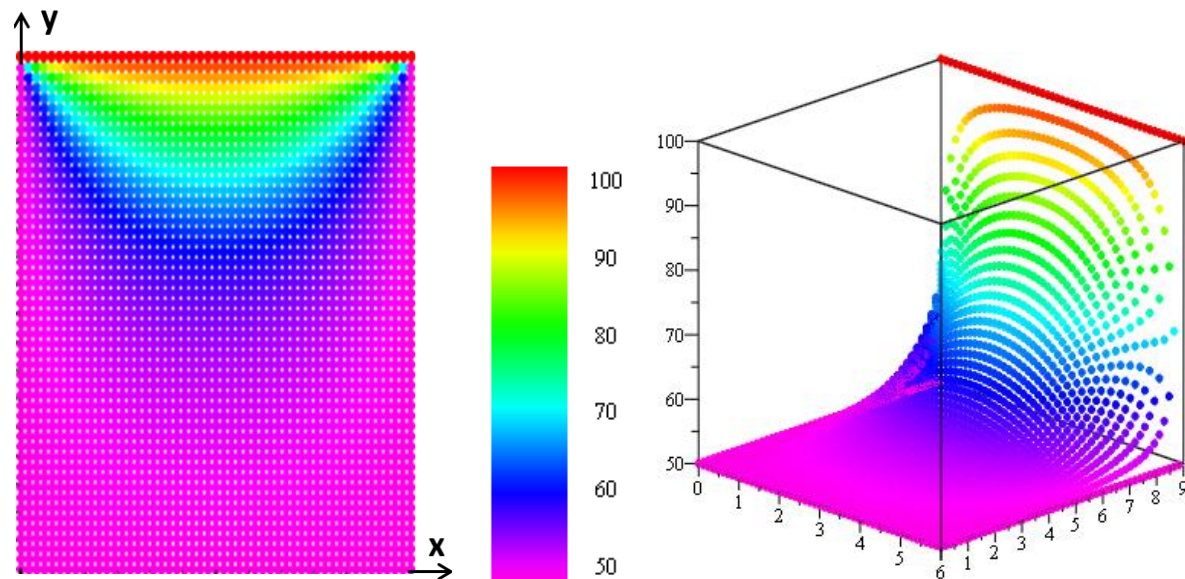


Figura 4.5 Distribución de temperaturas del problema obtenida en 2D Y 3D con el método local de Petrov-Galerkin (ec. de Laplace con condiciones de Dirichlet) para 51 x 51 nodos.

4.2.2. Condiciones de frontera de Neumann

Dentro del caso se resuelve el mismo problema pero con diferentes condiciones de frontera. Las caras laterales están aisladas, mientras que las otras están a 0°C y a 200°C. En la grafica 4.6 se tienen los resultados numéricos y analíticos en un dominio unidimensional para $x=0.3m$, lo cual nos ayuda a

4.2 RESULTADOS NUMÉRICOS DE LA ECUACIÓN DE LAPLACE CON CONDICIONES DE FRONTERA DE DIRICHLET Y NEUMANN

comprobar que los valores de las temperaturas ($^{\circ}\text{C}$) obtenidas con el código numérico de Petrov-Galerkin son muy aproximados a las temperaturas exactas. La solución analítica de la ecuación de Laplace bidimensional con condiciones de Neumann se muestra en la figura 4.7. En la figura 4.8 se observa la distribución de temperaturas de la sección trabajando con 51×51 nodos.

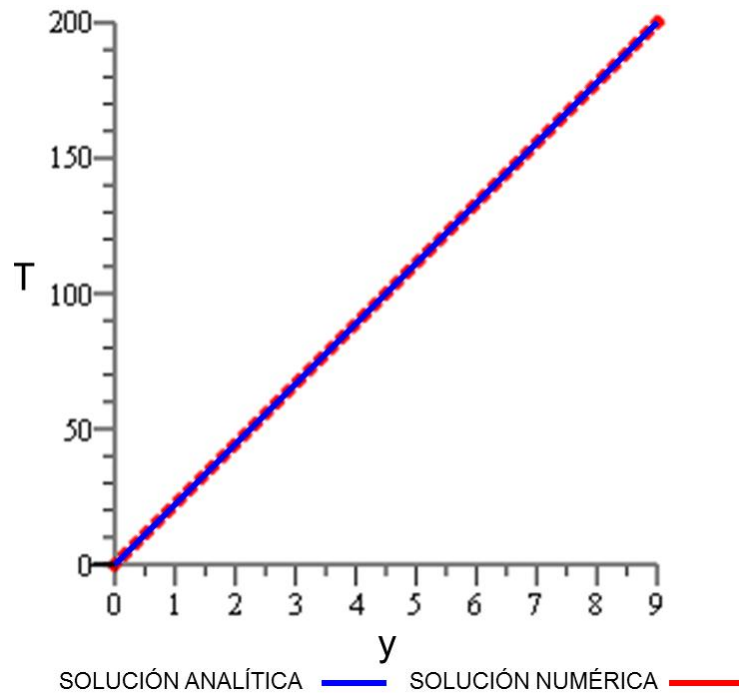


Figura 4.6 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Laplace con condiciones de Neumann).

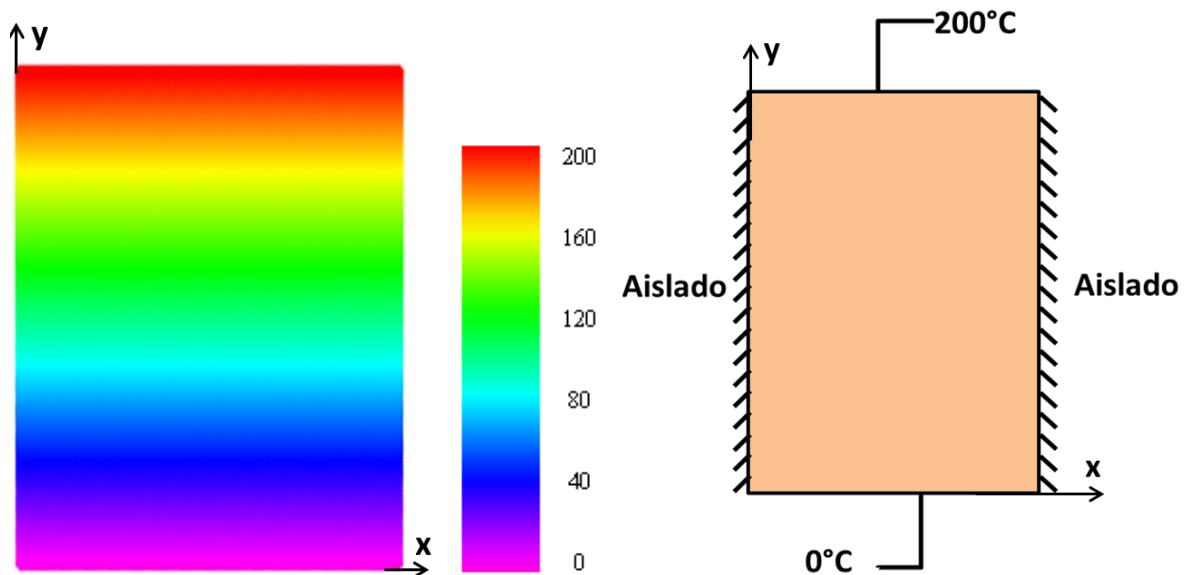


Figura 4.7 Distribución de temperaturas del problema generada por la solución analítica (ecuación de Laplace bidimensional con condiciones de Neumann).

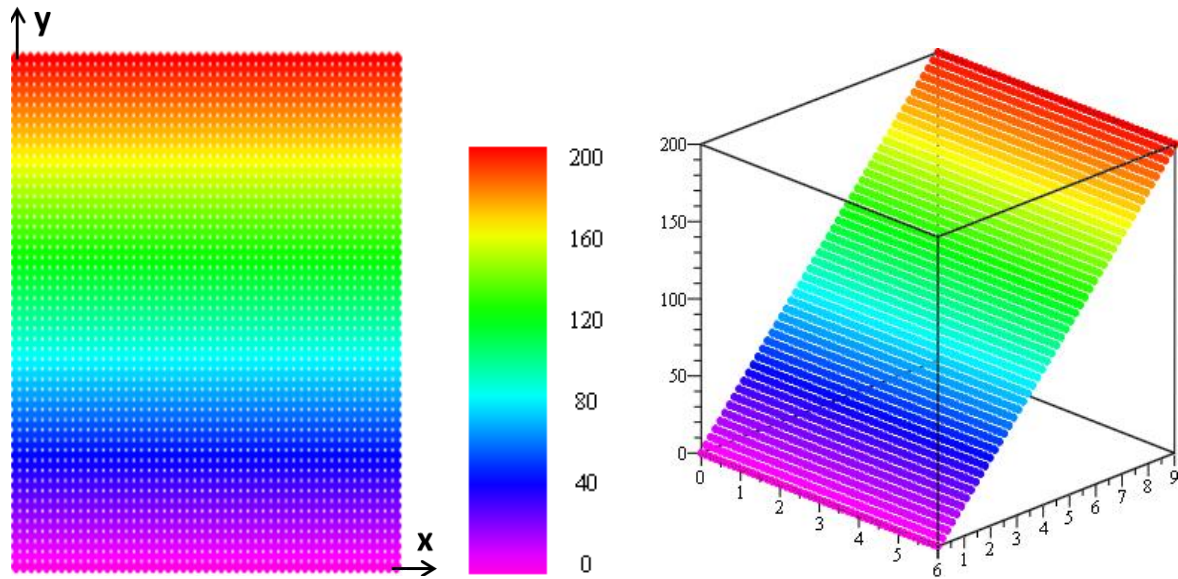
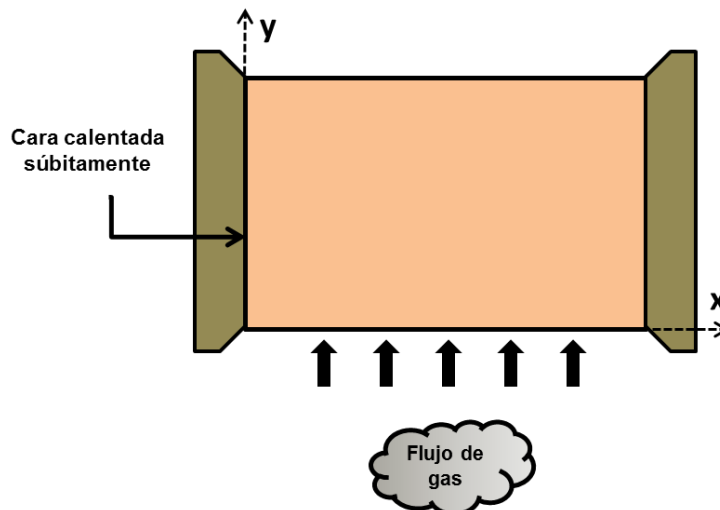


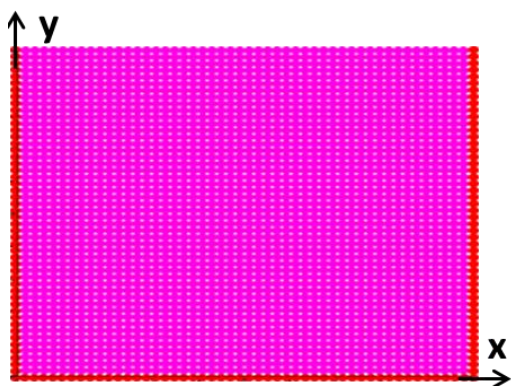
Figura 4.8 Distribución de temperaturas del problema obtenida en 2D Y 3D con el método local de Petrov-Galerkin (Ecuación de Laplace con condiciones de Neumann) para 51 x 51 nodos.

4.2.3. Ecuación de Laplace en estado transitorio

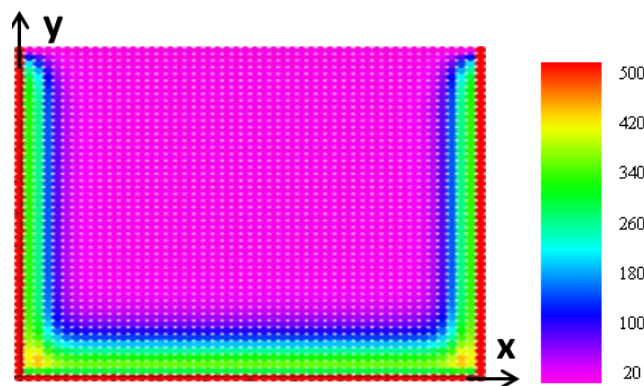
La ecuación de Laplace se resolvió en estado transitorio antes de tratar la ecuación de difusión de calor. El objetivo es encontrar la temperatura en el centro de la placa rectangular de cobre 8 segundos después de aplicar un calentamiento súbito en las caras laterales y un flujo de gas caliente en la cara inferior. La figura 4.9 muestra el campo de temperaturas en estado transitorio donde los tiempos están expresados en segundos y las temperaturas en °C.



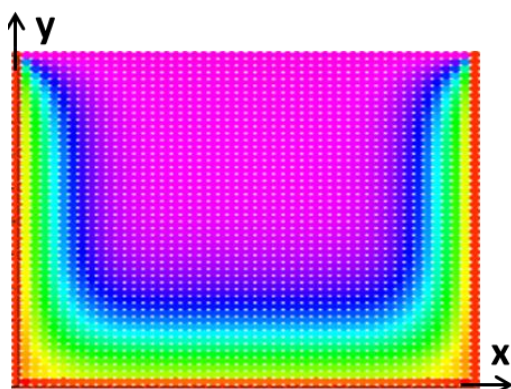
4.2 RESULTADOS NUMÉRICOS DE LA ECUACIÓN DE LAPLACE CON CONDICIONES DE FRONTERA DE DIRICHLET Y NEUMANN



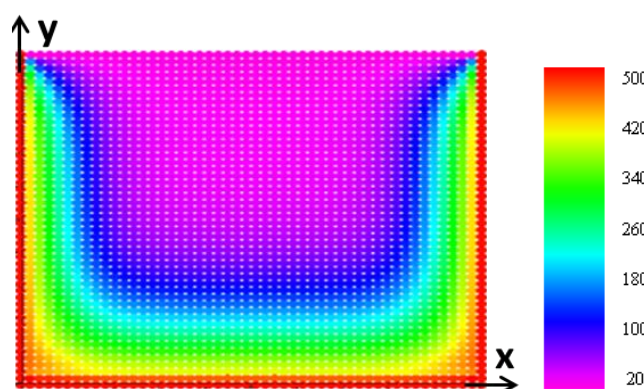
t=0 seg.



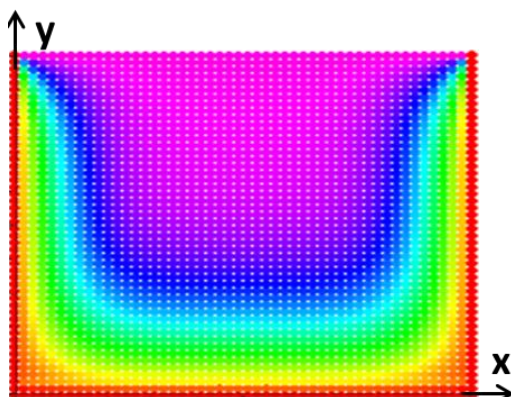
t=0.5 seg.



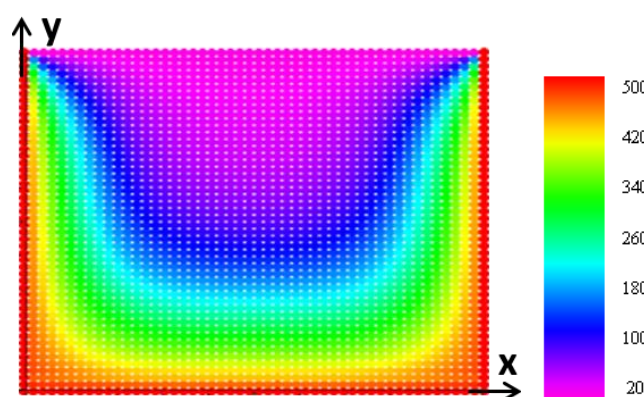
t=1 seg.



t=1.5 seg.



t=2 seg.



t=3 seg.

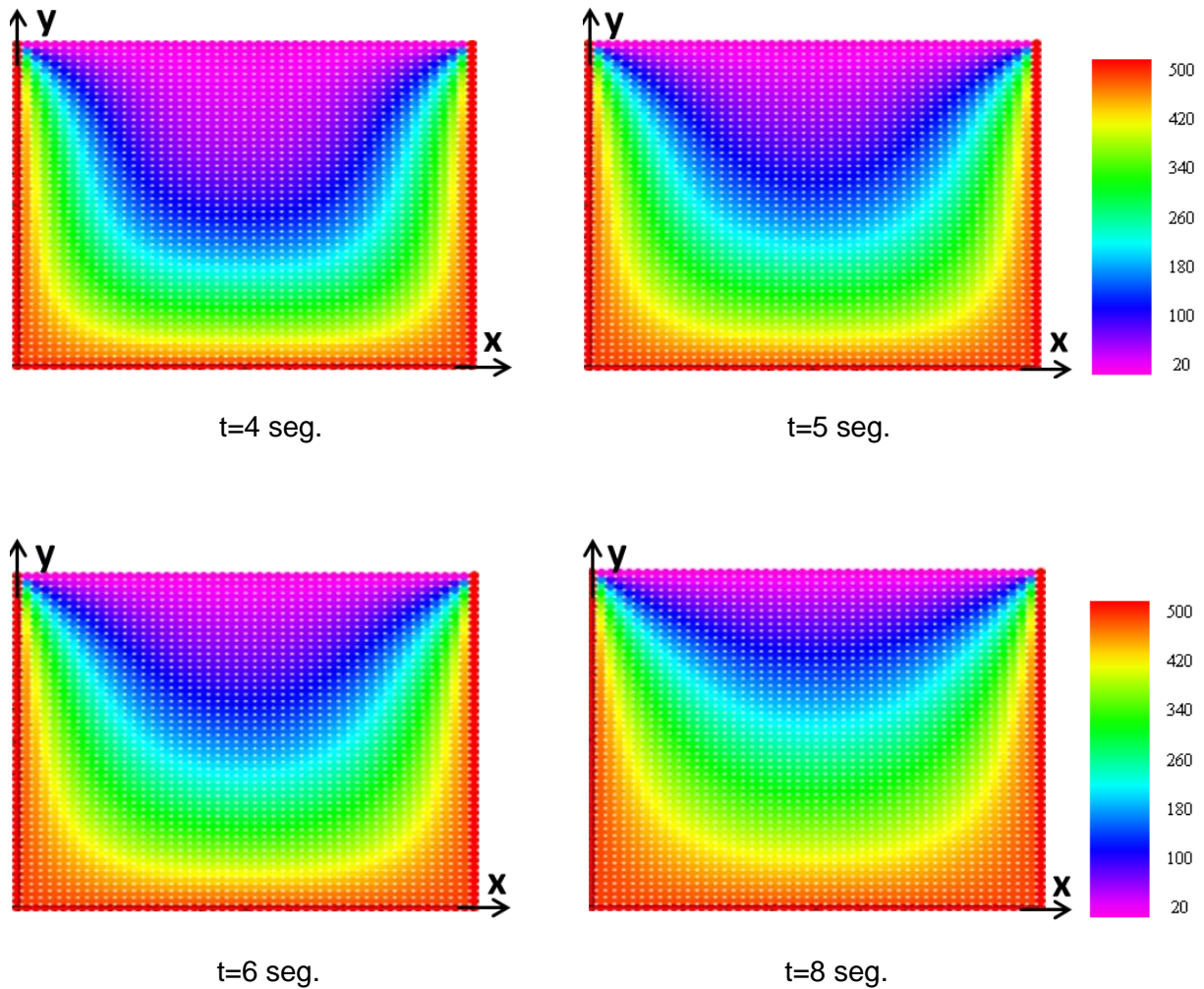


Figura 4.9 Distribución de temperaturas del problema en estado transitorio para 50 x 50 nodos (ecuación de Laplace bidimensional).

La temperatura del centro de la placa 8 segundos después de aplicar las condiciones de frontera antes mencionadas es aproximadamente de 220 °C.

4.3. Resultados numéricos de la ecuación de Poisson

A diferencia del primer caso, en este problema existe un término fuente que describe la generación de calor dentro de un horno de concreto donde existe aire caliente. Las paredes exteriores se encuentran a una temperatura de 30°C. Con ayuda del Método Local de Petrov-Galerkin se puede conocer aproximadamente la distribución de temperaturas a través de las paredes del horno. Para comprobar los

4.3 RESULTADOS NUMÉRICOS DE LA ECUACIÓN DE POISSON

valores numéricos obtenidos, se comparan con la solución analítica de la ecuación de Poisson unidimensional (figura 4.10). Los resultados del código computacional de Petrov-Galerkin (figura 4.12) se presentan de un modo diferente a la solución analítica de la ecuación de Poisson bidimensional que se vio en la sección 3.2. (figura 4.11), esto debido a que en el problema se impone la temperatura dentro del horno (aproximadamente de 641°C), en la región donde se encuentra el aire caliente.

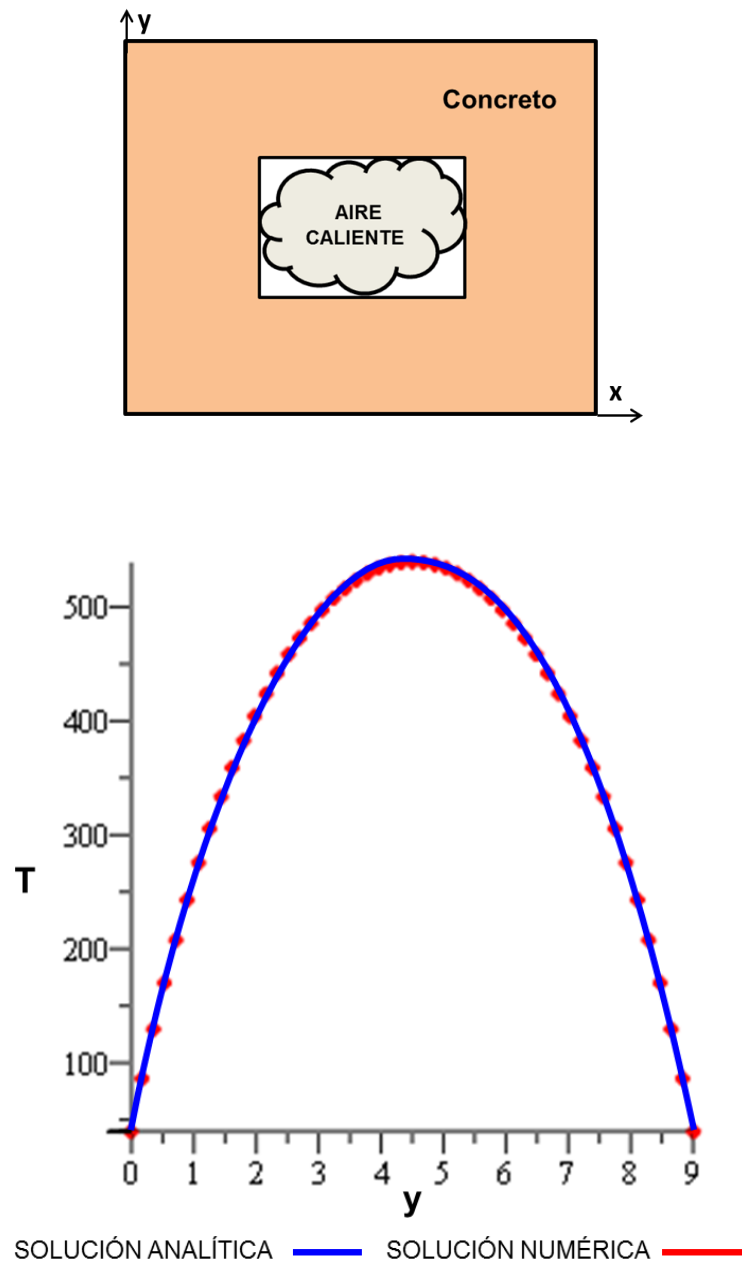


Figura 4.10 Distribución de temperaturas en un dominio unidimensional del problema (ecuación de Poisson con condiciones de Dirichlet).

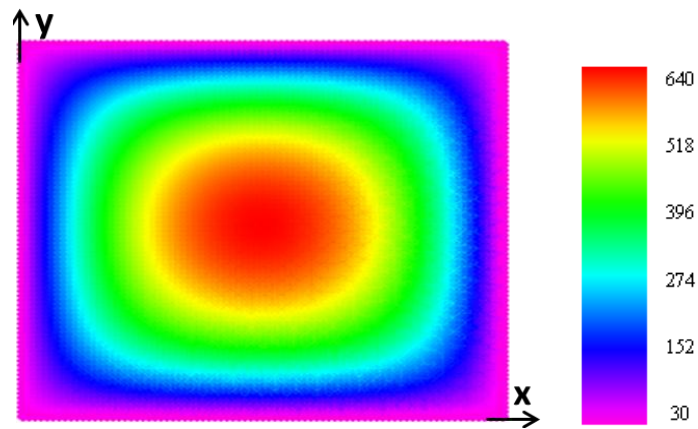


Figura 4.11 Solución analítica de la ecuación de Poisson bidimensional.

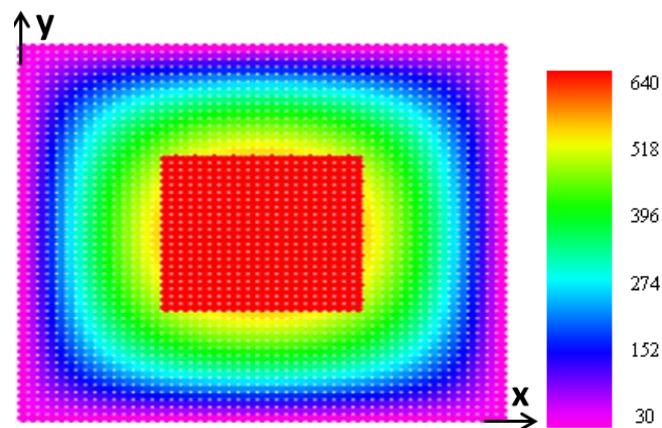
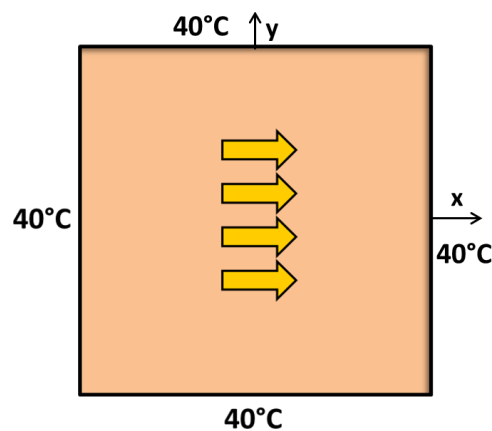
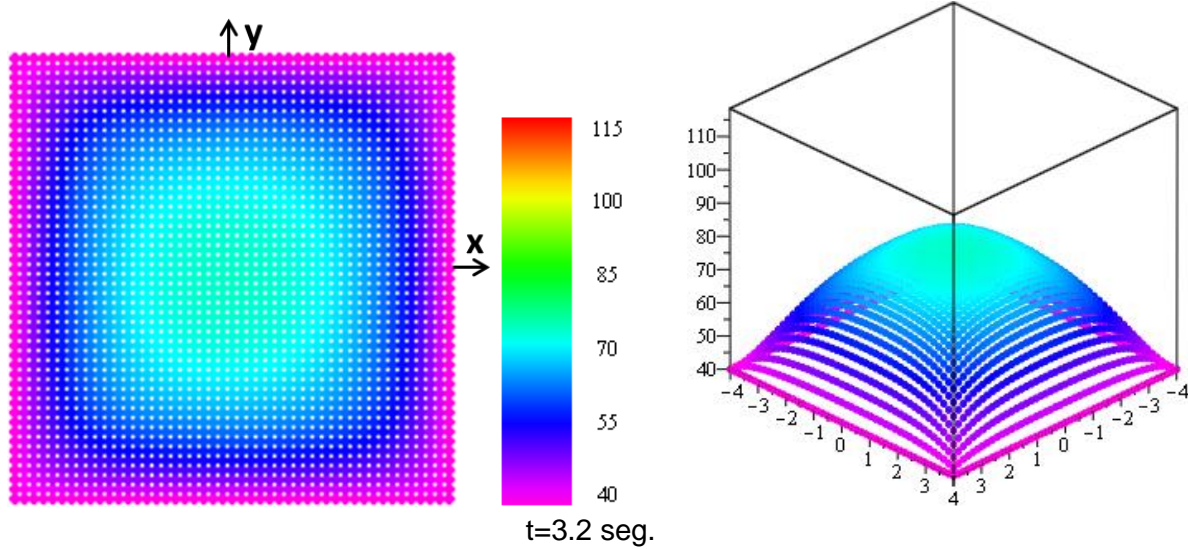
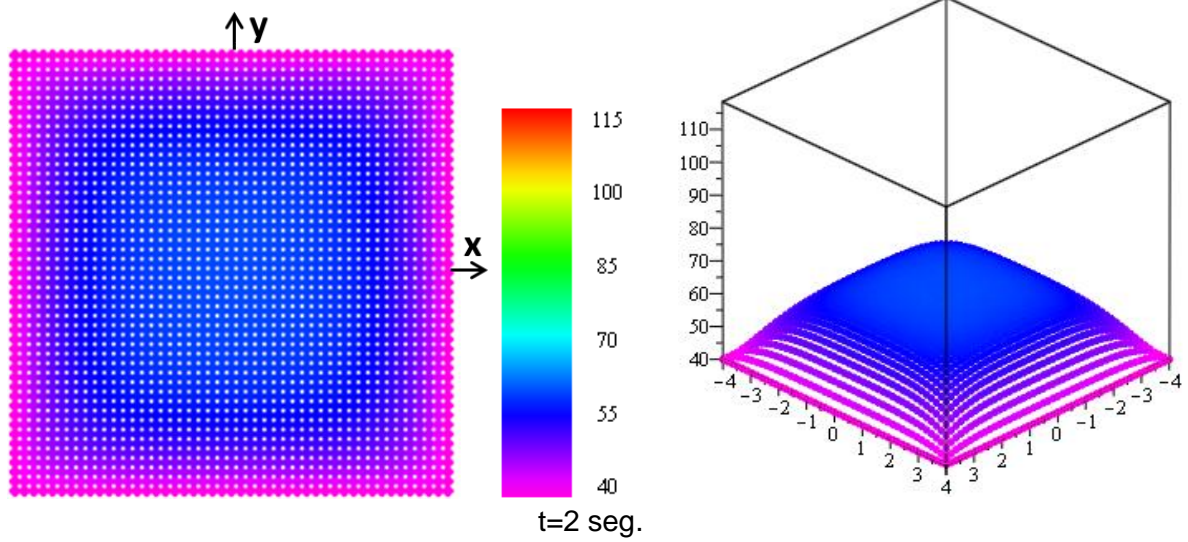
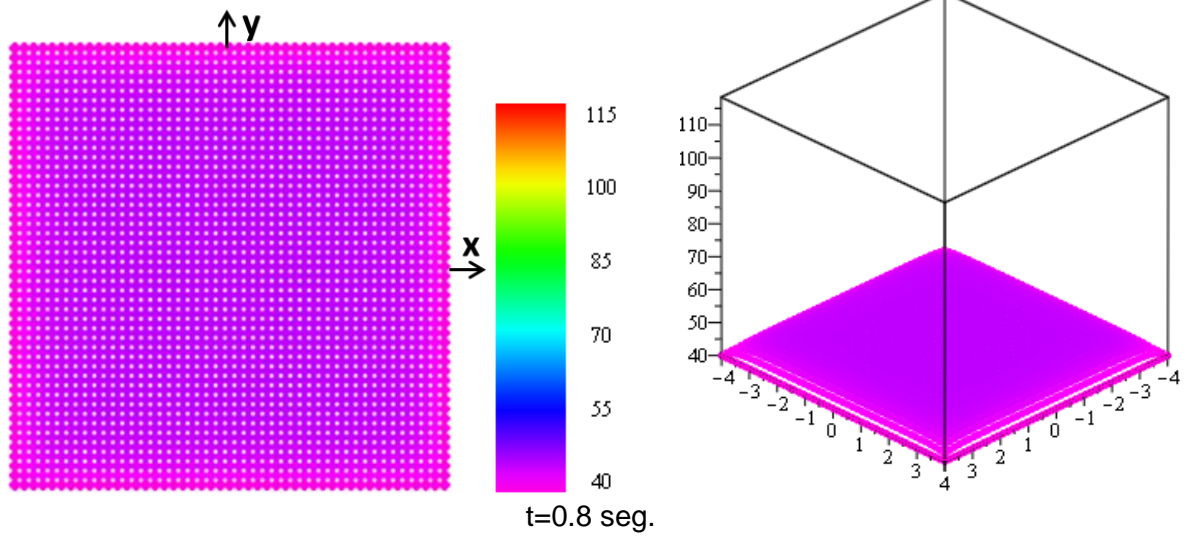


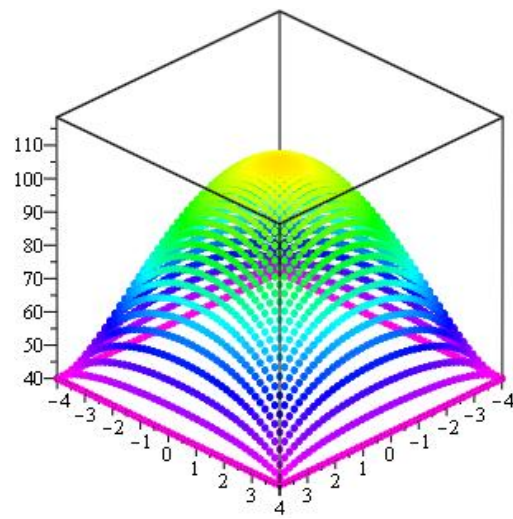
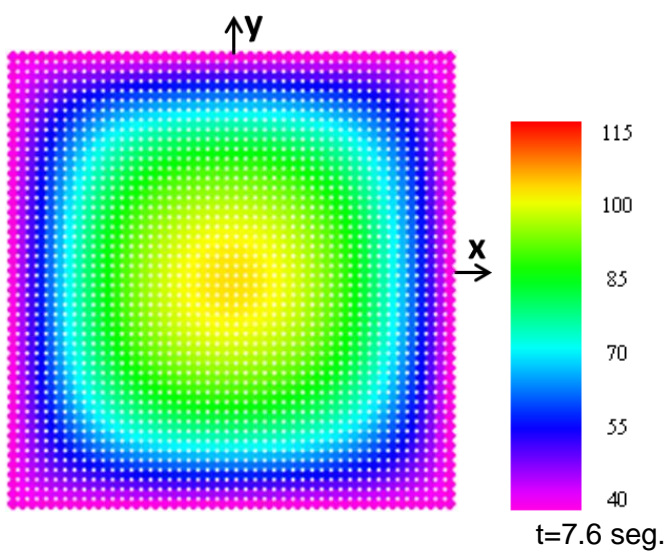
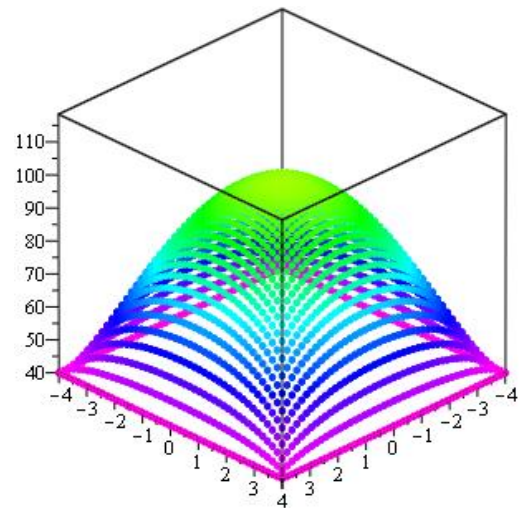
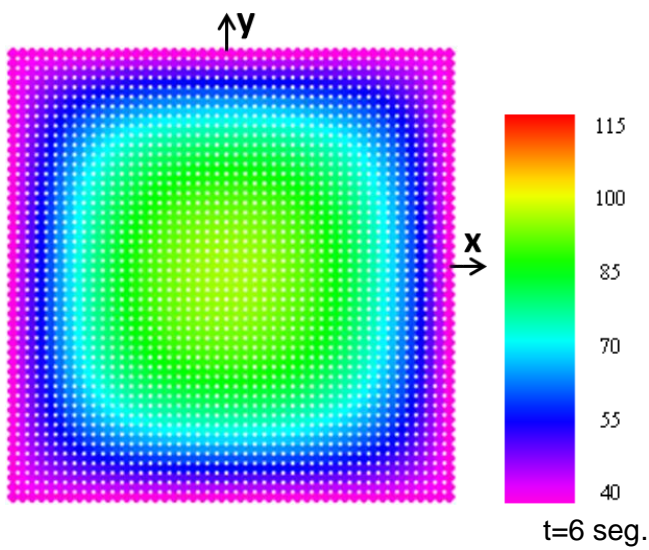
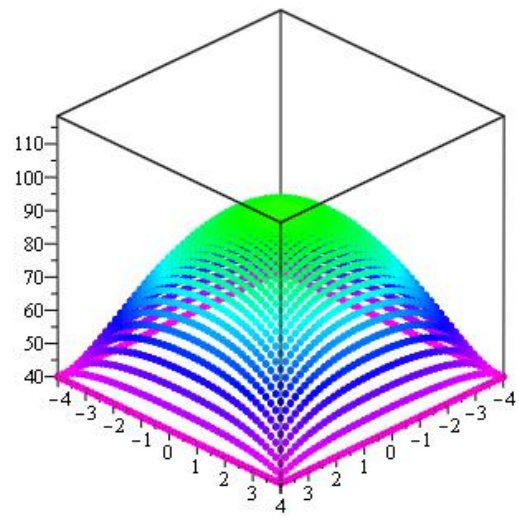
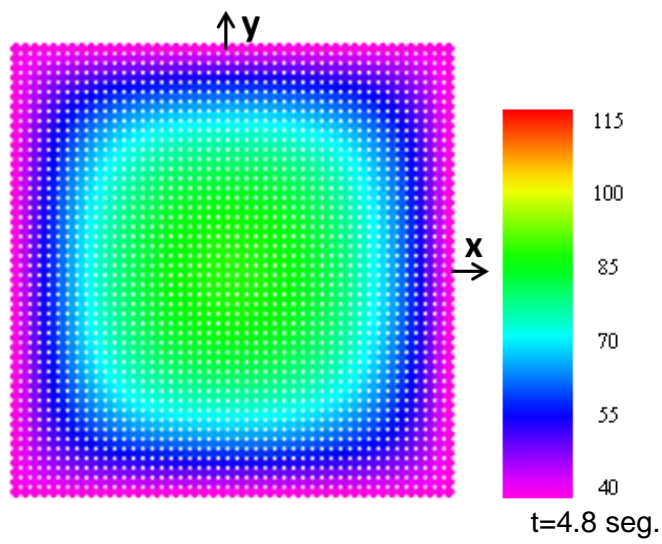
Figura 4.12 Solución numérica de la ec. de Poisson bidimensional para 51 x 51 nodos.

4.4. Resultados numéricos de la ecuación de difusión de calor

En el último caso, se da solución a la ecuación de difusión de calor usando el código de Petrov-Galerkin. La figura 4.13 muestra el comportamiento del campo de temperaturas del receptor solar de forma bidimensional y tridimensional a través del tiempo hasta alcanzar el estado estable.







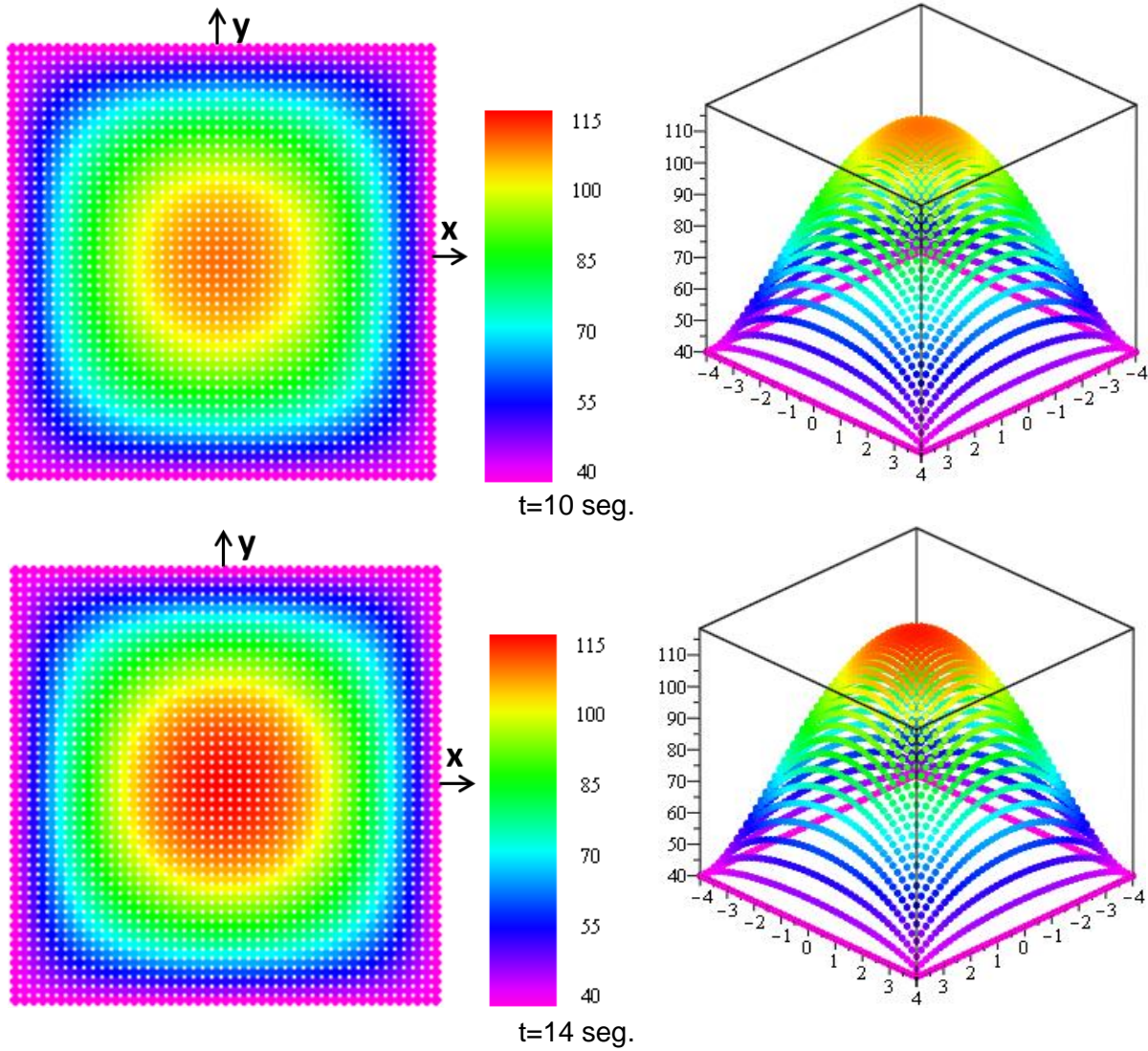


Figura 4.13 Solución numérica del problema (ecuación de difusión de calor) para 51 x 51 nodos.

CAPÍTULO 5

CONCLUSIONES

En este trabajo se ha presentado la teoría necesaria para la implementación numérica del Método Local de Petrov-Galerkin para problemas de difusión de calor. Para mostrar esta metodología, se trabajó en tres casos bidimensionales diferentes: la ecuación de Laplace, la ecuación de Poisson y la ecuación de difusión de calor en estado transitorio y en los cuales se resolvieron algunos problemas de transferencia de calor. El objetivo fue el de conocer la distribución de temperaturas en los distintos problemas físicos. Los resultados numéricos que se presentaron se aproximan bastante con las soluciones analíticas disponibles.

Sabemos que las soluciones exactas o ideales no existen en los problemas físicos reales de la ingeniería, es por eso de la importancia de los resultados que se obtienen de manera numérica, porque aunque exista discrepancia de lo exacto, muestran los valores más reales y aproximados. Se pudo comprobar que el método de Petrov-Galerkin puede acercarse a la solución analítica de una ecuación diferencial de manera muy aproximada.

Una de las cuestiones importantes a tomar en cuenta en éste método es lo complicado que resultan los procedimientos para resolver las ecuaciones debido a la dependencia de una malla, que se traduce en un alto costo computacional. La generación de las mallas computacionales, en ocasiones consume hasta un 70% del tiempo total de cómputo para resolver el problema (Muñoz, 2005). Es por esa razón que los problemas físicos que se vieron en el capítulo 3 se resolvieron trabajando con 50 x 50 nodos o 51 x 51 nodos, porque con más nodos el tiempo para obtener los valores era demasiado grande. También hay que considerar las características del ordenador con el que se trabajó, entre ellas: modelo Aser Aspire 5745, procesador Intel® Core™ i3 CPU M 350 @2.27GHz, memoria Ram de 2 GB, sistema operativo de 64 bits, etc.

Existen otras alternativas para resolver problemas más complejos. Una de ellas es usar los métodos libres de mallas. En la actualidad existen desacuerdos aun en la terminología que debe manejarse e incluso en los nombres de los métodos por la gran diversidad de métodos libres de malla que se han desarrollado. Sin embargo, las ventajas que ofrecen son evidentes ya que reducen el tiempo de cómputo al no generar el mallado computacional. Con respecto a las desventajas, la mayor es la gran cantidad de memoria que se requiere para invertir las matrices, que se espera se reduzcan en un futuro con los avances en los algoritmos numéricos para invertirlas y con la ayuda del cómputo paralelo, ya que los métodos libres de malla son altamente paralelizables. Su futuro es prometedor ya que eliminan la dependencia del mallado y por lo tanto se tendrían ventajas al abordar problemas más complicados como los de flujo con fronteras móviles, flujo con superficie libre, cambio de fase, etc. que podrían incluso resolverse en tres dimensiones.

Entre las recomendaciones que se da al lector es que antes de empezar cualquier programación se debe tener cuidado y entender las bases del método de Petrov-Galerkin, esto debido a que suele haber errores de cálculo o mal sintaxis de fórmulas dentro del proceso del método, ocasionando errores demasiado grandes, eso sin tomar en cuenta que el mismo método genera un error que hace que los valores no sean iguales a la solución exacta.

También se tiene que tomar en cuenta que cada lenguaje de programación es diferente y cada uno tiene características muy específicas. Entre los más importantes esta C++ y Fortran. Maple posee muchos comandos y paquetes parecidos a C++, pero también muchos diferentes. Es básico que se conozca el entorno de Maple antes de programar porque aunque se domine el método, sin la correcta sintaxis y funcionamiento de los elementos en Maple, no se podrán ver los resultados deseados.

El objetivo central de éste trabajo fue el de validar el Método Local de Petrov-Galerkin como herramienta para la simulación de campos de temperaturas utilizando variables primitivas para resolver la ecuación de difusión de calor, esperando que sirva como punto de partida para el estudio de problemas más complejos en el futuro.

APÉNDICE A

FORMA DÉBIL DE LAS ECUACIONES DE LAPLACE Y POISSON

A continuación se presentan los desarrollos para obtener la forma débil de las ecuaciones de Laplace y Poisson bidimensional. En términos generales se seguirá el procedimiento que se utilizó con la ecuación de difusión de calor en la sección 2.3.

- Ecuación de Laplace:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (\text{A.1})$$

- Ecuación de Poisson:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = -\frac{Q'}{k} \quad (\text{A.2})$$

donde $T \equiv T(x,y,t)$ es el campo de temperaturas (K); k es la conductividad térmica (W/m·K); $Q' = Q'(x,y,t)$ es un término fuente (W/m³).

A.1. Forma débil de la ecuación de Laplace

Primero se obtiene el residual pesado a partir de la ec. (A.1) La integral del residual pesado, en cada uno de los subdominios de cuadratura es igualada a cero, esto es (WRM, sec. 2.1):

$$r = - \left(\frac{\partial^2 T^h}{\partial x^2} + \frac{\partial^2 T^h}{\partial y^2} \right) \quad (\text{A.3})$$

$$-\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial y^2} \Psi_i \, d\Omega_{Qi} = 0 \quad (\text{A.4})$$

A continuación integramos por partes el primer término de la expresión anterior:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} = \int_{\Omega_{Qi}} \frac{\partial}{\partial x} \left(\frac{\partial T^h}{\partial x} \Psi_i \right) \, d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} \, d\Omega_{Qi} \quad (\text{A.5})$$

y aplicamos el Teorema de Gauss:

$$\int_V \frac{\partial}{\partial x} \lambda \, dV = \int_S \lambda n_x \, dS, \quad \text{donde: } \lambda = \frac{\partial T^h}{\partial x} \Psi_i$$

al primer término del lado derecho de la igualdad para obtener:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} = \int_{\Gamma_{Qi}} \frac{\partial T^h}{\partial x} n_x \Psi_i \, d\Gamma_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} \, d\Omega_{Qi} \quad (\text{A.6})$$

donde Γ_{Qi} es la frontera del dominio local de cuadratura Ω_{Qi} y n_x es la componente en la dirección x_1 del vector normal unitario \mathbf{n} a la superficie Γ_{Qi} . Aplicamos el mismo procedimiento al segundo término de (ec A.4) para obtener *la forma débil de la ecuación Laplace*:

$$\int_{\Omega_{Qi}} \left(\frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} \right) \, d\Omega_{Qi} - \int_{\Gamma_{Qi}} \left(\frac{\partial T^h}{\partial x} n_x \Psi_i + \frac{\partial T^h}{\partial y} n_y \Psi_i \right) \, d\Gamma_{Qi} = 0 \quad (\text{A.7})$$

A.2. Forma débil de la ecuación de Poisson

Primero se obtiene el residual pesado a partir de la ec. (A.2) La integral del residual pesado, en cada uno de los subdominios de cuadratura es igualada a cero, esto es (WRM, sec. 2.1):

$$r = - \left(\frac{\partial^2 T^h}{\partial x^2} + \frac{\partial^2 T^h}{\partial y^2} \right) - \frac{Q'}{k} \quad (\text{A.8})$$

$$- \int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial y^2} \Psi_i \, d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{Q'}{k} \Psi_i \, d\Omega_{Qi} = 0 \quad (\text{A.9})$$

A continuación integramos por partes el primer término de la expresión anterior:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} = \int_{\Omega_{Qi}} \frac{\partial}{\partial x} \left(\frac{\partial T^h}{\partial x} \Psi_i \right) \, d\Omega_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} \, d\Omega_{Qi} \quad (\text{A.10})$$

y aplicando el Teorema de Gauss:

$$\int_V \frac{\partial}{\partial x} \lambda \, dV = \int_S \lambda n_1 \, dS, \quad \text{donde: } \lambda = \frac{\partial T^h}{\partial x} \Psi_i$$

al primer término del lado derecho de la igualdad para obtener:

$$\int_{\Omega_{Qi}} \frac{\partial^2 T^h}{\partial x^2} \Psi_i \, d\Omega_{Qi} = \int_{\Gamma_{Qi}} \frac{\partial T^h}{\partial x} n_1 \Psi_i \, d\Gamma_{Qi} - \int_{\Omega_{Qi}} \frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} \, d\Omega_{Qi} \quad (\text{A.11})$$

donde Γ_{Qi} es la frontera del dominio local de cuadratura Ω_{Qi} y n_1 es la componente en la dirección x_1 del vector normal unitario \mathbf{n} a la superficie Γ_{Qi} . Aplicamos el mismo procedimiento al segundo término de (ec A.9) para obtener *la forma débil de la ecuación Poisson*:

$$\int_{\Omega_{Q_i}} \left(\frac{\partial T^h}{\partial x} \frac{\partial \Psi_i}{\partial x} + \frac{\partial T^h}{\partial y} \frac{\partial \Psi_i}{\partial y} - \frac{Q'}{k} \Psi_i \right) d\Omega_{Q_i} - \int_{\Gamma_{Q_i}} \left(\frac{\partial T^h}{\partial x} n_1 \Psi_i + \frac{\partial T^h}{\partial y} n_2 \Psi_i \right) d\Gamma_{Q_i} = 0 \quad (\text{A.12})$$

APÉNDICE B

CUADRATURA DE GAUSS-LOBBATO- LEGENDRE (GLL)

Dentro de la integración numérica existen muchos métodos para calcular el valor numérico de la integral definida. El método de cuadratura de Gauss-Lobato-Legendre es uno de los más precisos ya que permite disminuir el error cambiando la localización de los puntos sobre la curva de integración $f(x)$.

En la figura B.1 se tiene la curva de la función $f(x)$ que se desea integrar entre los límites a y b . La parte (a) de la figura muestra como se integraría usando un trapecioide: uniendo el punto A de coordenadas $(a, f(a))$ con el punto B $(b, f(b))$ mediante un segmento de recta $p(x)$. Esto forma un trapecioide de altura $h=(b-a)$, cuya área es:

$$T = \frac{h}{2} [f(a) + f(b)] \quad (\text{B.1})$$

y que podría escribirse como:

$$T = w_1 f(a) + w_2 f(b) \quad (\text{B.2})$$

donde T es el área del trapecioide calculada que aproxima el área bajo la curva $f(x)$;
 $w_1=w_2=h/2$;

Por otro lado, en la aplicación de la cuadratura de Gauss-Lobato-Legendre, en lugar de tomar los dos puntos A y B en los extremos del intervalo, se escogen dos puntos interiores C y D (véase la parte b de la figura B.1).

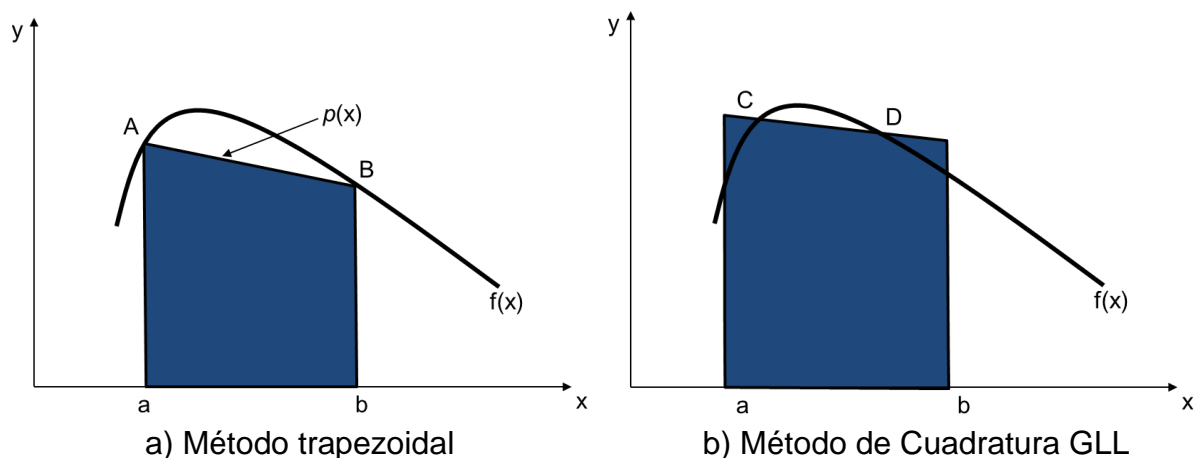


Figura B.1 Desarrollo del método de integración de Gauss-Lobatto-Legendre (parte b) usando dos puntos a partir del método trapezoidal (parte a)

Se traza una línea recta por estos dos puntos, se extiende hasta los extremos del intervalo y se forma el trapezoide sombreado. Parte del trapezoide queda por encima de la curva y parte por abajo. Si se escogen adecuadamente los puntos C y D, cabe igualar las dos zonas de modo que el área del trapezoide sea igual al área bajo la curva, el cálculo del área del trapezoide resultante da la integral exacta. El método de cuadratura de GLL consiste esencialmente en seleccionar los puntos C y D adecuados.

Se considera primero, sin que esto implique perder generalidad, que se desea integrar la función mostrada en la figura B.2 entre los límites -1 y +1. Los puntos C y D se escogen sobre la curva y se forma el trapezoide con vértices E,F,G,H. Sean las coordenadas del punto C $(z_1, f(z_1))$ y las del punto D $(z_2, f(z_2))$. Debido a la fórmula trapezoidal (ec B.1), en GLL se propone desarrollar una fórmula tipo:

$$A = w_1 F(z_1) + w_2 F(z_2) \quad (\text{B.3})$$

Ya que esto simplificaría relativamente el cálculo del área. El problema planteado de esta manera, consiste en encontrar los valores de z_1 , z_2 , w_1 y w_2 . Entonces hay cuatro parámetros por determinar y, por tanto, cuatro condiciones que se pueden imponer. Estas se eligen de manera que el método dé resultados exactos cuando la función por integrar sea alguna de las cuatro siguientes o combinaciones lineales de ellas.

$$F(z)=1; F(z)=z; F(z)=z^2; F(z)=z^3$$

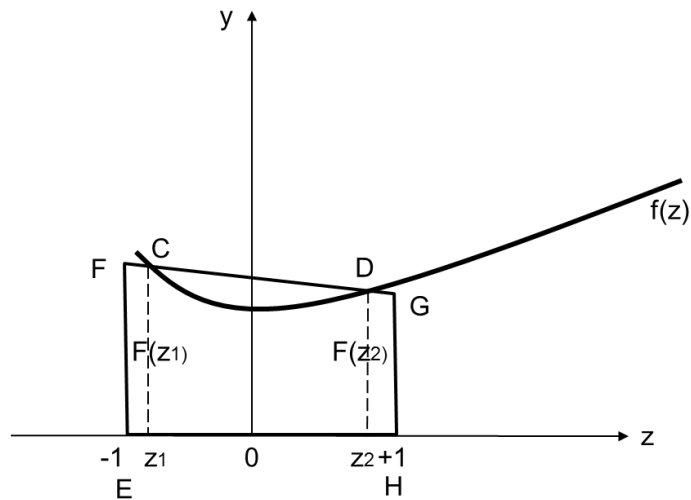


Figura B.2 Derivación del método de integración de GLL.

Los valores exactos de integrar estas cuatro funciones entre -1 y +1 son:

$$I_1 = \int_{-1}^1 1 \, dz = z \Big|_{-1}^1 = 1 - 1(-1) = 2$$

$$I_2 = \int_{-1}^1 z \, dz = \frac{z^2}{2} \Big|_{-1}^1 = \frac{1^2}{2} - \frac{(-1)^2}{2} = 0$$

$$I_3 = \int_{-1}^1 z^2 \, dz = \frac{z^3}{3} \Big|_{-1}^1 = \frac{1^3}{3} - \frac{(-1)^3}{3} = \frac{2}{3}$$

$$I_4 = \int_{-1}^1 z^3 \, dz = \frac{z^4}{4} \Big|_{-1}^1 = \frac{1^4}{4} - \frac{(-1)^4}{4} = 0$$

Suponiendo que una ecuación de la forma B.3 funciona correctamente, se tendría el siguiente sistema de ecuaciones:

$$I_1 = w_1(1) + w_2(1) = 2$$

$$I_2 = w_1 z_1 + w_2 z_2 = 0$$

$$I_3 = w_1 z_1^2 + w_2 z_2^2 = \frac{2}{3}$$

$$I_4 = w_1 z_1^3 + w_2 z_2^3 = 0$$

De la primera ecuación se tiene que $w_1+w_2=2$; nótese que también que si

$$w_1 = w_2$$

y

$$z_1 = -z_2$$

Se satisface la segunda y la cuarta ecuación del sistema. Entonces se elige

$$w_1 = w_2 = 1$$

y

$$z_1 = -z_2$$

y al sustituir en la tercera ecuación se obtiene

$$z_1^2 + (-z_1)^2 = \frac{2}{3}$$

o bien

$$z_1^2 = \frac{1}{3}$$

donde

$$z_1 = \pm \frac{1}{\sqrt{3}} = \pm 0.57735\dots$$

y queda entonces:

$$z_1 = -0.57735\dots$$

$$z_2 = 0.57735\dots$$

Con lo que se tiene la fórmula

$$\int_{-1}^1 F(z) dz = w_1 F(z_1) + w_2 F(z_2) = F(-0.57735\dots) + F(+0.57735\dots) \quad (\mathbf{B.4})$$

Que, salvo porque se tiene que calcular el valor de la función en un valor irracional de z , es tan simple como la regla del método trapezoidal; además, trabaja perfectamente para funciones cúbicas, mientras que la regla trapezoidal lo hace sólo para líneas rectas.

Para integrar en un intervalo distinto de $[-1,1]$, se requiere un cambio de variable a fin de pasar del intervalo de integración general $[a, b]$ a $[-1,1]$ y así aplicar la ecuación B.4.

Para cambiar el intervalo de integración se aplica la fórmula:

$$z = \frac{2x - (a + b)}{(b - a)} \quad (\text{B.5})$$

El integrando $f(x) dx$ en términos de la nueva variable queda:

$$f(x) = F\left(\frac{(b - a)}{2}z + \frac{(a + b)}{2}\right)$$

y

$$dx = d\left(\frac{(b - a)}{2}z + \frac{(a + b)}{2}\right) = \frac{(b - a)}{2} dz$$

Por lo que la integral finalmente queda como:

$$\begin{aligned} \int_a^b f(x) dx &= \frac{b - a}{2} \int_{-1}^1 F\left(\frac{(b - a)}{2}z + \frac{(a + b)}{2}\right) dz \\ &= \frac{b - a}{2} \left[F\left(\frac{(b - a)}{2}(-0.57735...) + \frac{(a + b)}{2}\right) \right. \\ &\quad \left. + F\left(\frac{(b - a)}{2}(+0.57735...) + \frac{(a + b)}{2}\right) \right] \quad (\text{B.6}) \end{aligned}$$

Una cuestión importante es que el método de GLL puede extenderse a tres o más puntos; por ejemplo, si se escogen tres puntos no equidistantes en el segmento de la curva $f(z)$ comprendida entre -1 y 1 , se podría pasar una parábola por los tres como en la regla de Simpson, excepto en que dichos puntos se escogerían de modo que minimicen o anulen el error. Similarmente es factible elegir cuatro puntos y una curva cuadrática, cinco puntos y una curva cúbica, etc. En general el algoritmo tiene la forma:

$$\int_{-1}^1 F(z) dz = A = w_1 F(z_1) + w_2 F(z_2) + w_3 F(z_3) + \dots + w_n F(z_n) \quad (\text{B.7})$$

donde se han calculado los valores w_i y z_i por usar. La tabla B.1 da los valores hasta para seis puntos.

Con dos puntos, el método GLL está diseñado para obtener exactitud en polinomios cúbicos; con tres, se tendrá exactitud en polinomios de cuarto grado y así sucesivamente.

Los puntos y pesos dados en la tabla B.1 sirven para integrar sobre todo el intervalo de interés, o bien puede dividirse el intervalo en varios subintervalos y aplicar el método GLL a cada uno de ellos.

NUMERO DE PUNTOS	PESOS	PUNTOS
1	$w_1=2$	$z_1=0$
2	$w_1=w_2=1.0$	$-z_1=z_2=0.5773502692$
3	$w_2=0.88888\dots$ $w_1=w_3=0.55555\dots$	$-z_1=z_3=0.7745966692$ $z_2=0$
4	$w_1=w_4=0.3478548451$ $w_2=w_3=0.6521451549$	$-z_1=z_4=0.8611363116$ $-z_2=z_3=0.3399810436$
5	$w_1=w_5=0.2369268851$ $w_2=w_4=0.4786286705$ $w_3=0.56888\dots$	$-z_1=z_5=0.9061798459$ $-z_2=z_4=0.5384693101$ $z_3=0$
6	$w_1=w_6=0.1713244924$ $w_2=w_5=0.3607615730$ $w_3=w_4=0.4679139346$	$-z_1=z_6=0.9324695142$ $-z_2=z_5=0.6612093865$ $-z_3=z_4=0.2386191861$

Tabla B.1 Pesos y puntos usados en el método de Gauss-Lobatto-Legendre

La fórmula B.7 puede describirse en una forma más general y adecuada para programarla, así:

$$\int_a^b f(x) dx = \frac{b-a}{2} \sum_{i=1}^n w_i F\left[\frac{(b-a)z_i + a + b}{2}\right] \quad \text{(B.8)}$$

Igualmente puede emplearse la cuadratura de GLL para integrales dobles o triples. Así en el caso general

$$\int_c^d \int_a^b f(x, y) dx dy$$

primero se cambian los intervalos de x y y a $[-1,1]$ con las fórmulas

$$z = \frac{2x - (a + b)}{(b - a)} \quad y \quad u = \frac{2y - (c + d)}{(d - c)}$$

y asimismo se cambian dx , dy y $f(x,y)$ en términos de las nuevas variables z y u . para esto se despeja a x y después y .

$$x = \frac{(b - a)}{2}z + \frac{(a + b)}{2} \quad \text{de donde} \quad dx = \frac{(b - a)}{2} dz$$

$$y = \frac{(d - c)}{2}u + \frac{(c + d)}{2} \quad \text{de donde} \quad dy = \frac{(d - c)}{2} du$$

se sustituye:

$$\int_c^d \int_a^b f(x, y) dx dy = \frac{(b - a)(d - c)}{4} \cdot \int_{-1}^1 \int_{-1}^1 f \left[\frac{(b - a)}{2}z + \frac{(a + b)}{2}, \frac{(d - c)}{2}u + \frac{(c + d)}{2} \right] dz du \quad (\text{B.9})$$

Para mayor sencillez y facilidad de programación, es conveniente usar la fórmula:

$$\int_c^d \int_a^b f(x, y) dx dy = \frac{(b - a)(d - c)}{4} \cdot \sum_{j=1}^m w_j \sum_{i=1}^n w_i F \left[\frac{(b - a)}{2}z_i + \frac{(a + b)}{2}, \frac{(d - c)}{2}u_j + \frac{(c + d)}{2} \right] \quad (\text{B.10})$$

APÉNDICE C

SOLIDWORKS FLOW SIMULATION

Existen muchos programas que usan métodos numéricos para resolver ecuaciones diferenciales parciales aplicadas a problemas físicos. Solid Works es un software de diseño asistido por computadora para modelado mecánico y que utiliza el M.E.F (Método del Elemento Finito) para dar solución a problemas complejos de análisis estructural, térmico, estudios dinámicos, etc.

El M.E.F es una técnica numérica para analizar diseños de ingeniería. El M.E.F está aceptado como el método de análisis estándar debido a su generalidad y compatibilidad para ser implementado en computadoras. El M.E.F divide el modelo en numerosas piezas pequeñas de formas simples llamadas "elementos", que reemplazan eficazmente un problema complejo por muchos problemas simples que deben ser resueltos de manera simultánea.

En los estudios térmicos se calculan temperaturas, gradientes de temperatura y flujo del calor sobre la base de la generación de calor y condiciones de conducción, convección y radiación. Los estudios térmicos nos ayudan a evitar condiciones térmicas no deseadas, tales como el sobrecalentamiento y la fusión.

Para fines de comparación con el Método de Petrov-Galerkin se usó SolidWorks Flow Simulation para obtener la distribución de temperaturas en una placa rectangular (ecuación de Laplace con condiciones de Dirichlet). La figura C.1 muestra el campo de temperaturas bidimensional de la placa usando el M.E.F. En la figura C.2 se obtuvo la distribución de temperaturas para la misma placa pero con generación de calor (ecuación de Poisson).

Se puede ver que las graficas son muy parecidas a las obtenidas con el método de Petrov-Galerkin (Capítulo 4).

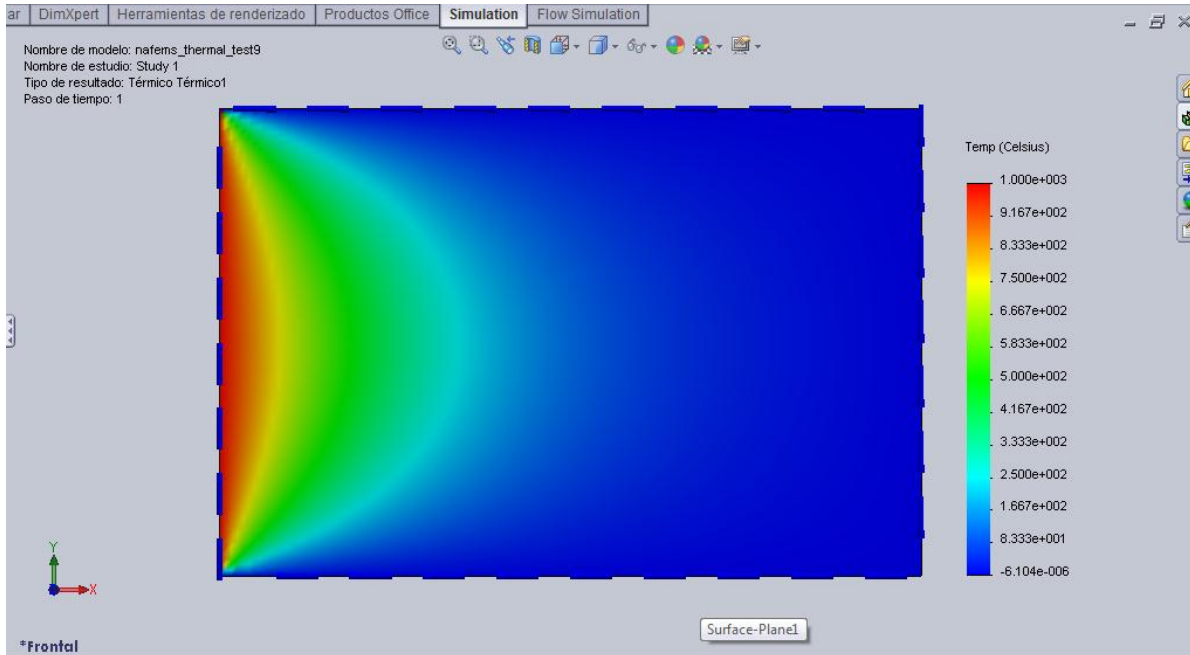


Figura C.1 Distribución de temperaturas en °C de una placa rectangular (ecuación de Laplace con condiciones de Dirichlet) usando SolidWorks Flow Simulation.

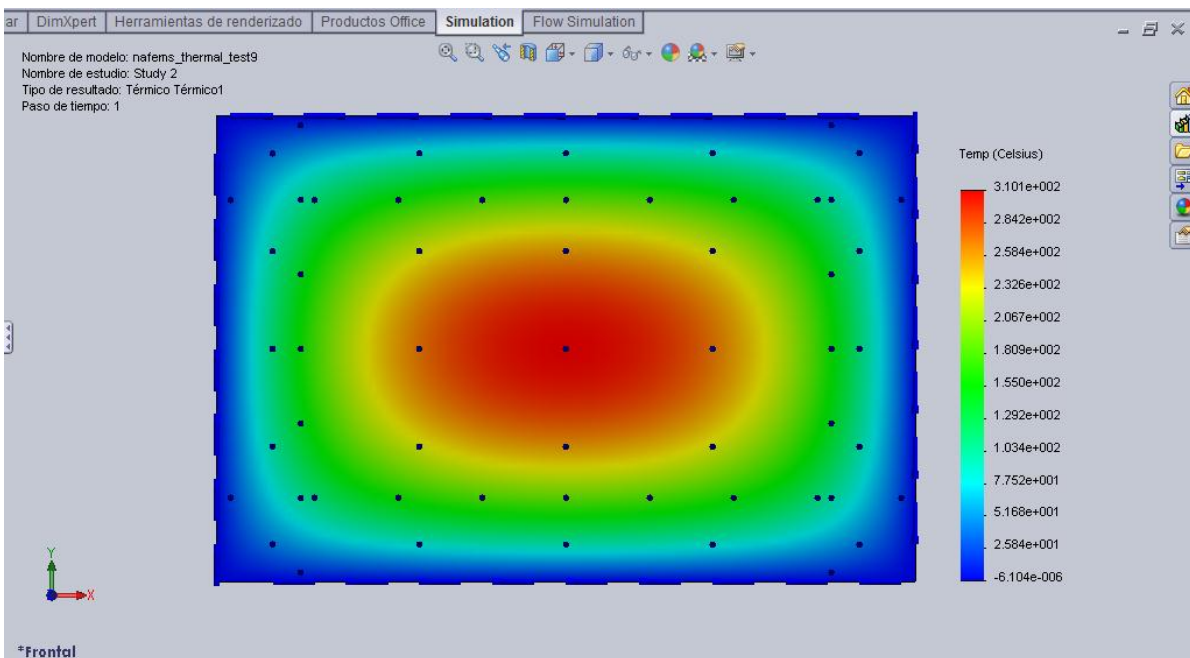


Figura C.2 Distribución de temperaturas en °C de una placa rectangular con generación de calor (ecuación de Poisson con condiciones de Dirichlet) usando SolidWorks Flow Simulation.

APÉNDICE D

PROGRAMACIÓN DEL MÉTODO LOCAL DE PETROV-GALERKIN EN LA ECUACIÓN DE LAPLACE USANDO MAPLE

A continuación se presenta la programación en Maple del método local de Petrov-Galerkin aplicado a la ecuación de Laplace con condiciones de Dirichlet. El problema de este caso se resolvió en la sección 3.1.1.



```
>restart:
>with (plots):
>with (LinearAlgebra):
>with (MTM):
>with (Statistics):
>PG:= proc ( );
local j, i, m, n, o, f, v, r, ñ;
global Lo, Lf, Wo, Wf, nx, ny, incx, incy, nt, fun, ρ, ω, nodo, no, nod,
nodovecinos, vecinosnodo, a, b, c, d, nox, noy, CONSTANTE, vn, l,
X, Y, dx, dy, wx, wy, wxy, qx, ddx, wodx, wdx, qy, ddy, wody, wdy,
Vxy, Wxy, pxy, ptxy, colxy, Pxy, PTxy, Bxy, Mxy, MIxy, Vdx, Wdx,
pdx, ptdx, Bdx, Mdx, MIdx, Φdx, laudx, Vdy, Wdy, pdy, ptdy, Bdy,
Mdy, MI dy, Φdy, laudy, puntox, puntoy, gaussx, gaussy, sumax,
sumay, A, P, vec, can, k, ka, ARR, MA, MI, pos, efe, FE, T, zetnum,
equis, yes, numerica;
printf ("SIMULACIÓN NUMÉRICA DE LA ECUACIÓN DE LAPLACE
CON CONDICIONES DE DIRICHLET \n");
```

```

#inicio del dominio en x
Lo:=0;
#fin del dominio en x
Lf:=6;
#inicio del dominio en y
Wo:=0;
#fin del dominio en y
Wf:=9;
#número de nodos en "x" y "y"
nx:=51;
ny:=51;
incx:= (Lf-Lo)/(nx-1);
incy:= (Wf-Wo)/(ny-1);
nt:= nx*ny;
#temperatura f(x) homogénea en alguna frontera del problema físico
fun:= 100;

#pesos y puntos de la cuadratura de Gauss-Lobato-Legendre (5
puntos)
ρ:= [-0.90617, -0.53846, 0, 0.53846, 0.90617];
ω:= [0.23692, 0.47862, 0.56888, 0.47862, 0.23692];

#"nodo" guarda la coordenada de c/nodo; "j","i" es la fila y columna
donde esta el nodo en la geometría cuadrada
for j from 1 to ny do
  for i from 1 to nx do
    nodoj,i:= [((i-2) incx + incx), ((j-2) incy + incy)];
  od;
od;

#"nodo" guarda la coordenada de c/nodo pero sin necesidad de
indicar su fila y su columna sino por el número de nodo; "m" es el
nodo estudiado
no:=seq(seq(nodoj,i, i=1..nx), j=1..ny);
for m from 1 to nt do
  nodm:=no[m];
od;

#"nodovecinos" da los vecinos del nodo "m" en estudio
for m from 1 to nt do
  if (m=1) then;
  nodovecinosm:= [ m, m+1, m+nx, m+nx+1 ];
  fi;
  if (m=nx) then;
  nodovecinosm:= [m-1, m, m+(nx-1), m+nx];
  fi;

```

```

if (m=(nt)-(nx-1)) then;
nodovecinosm:= [nt-((2nx)-1), nt-((2nx)-2), nt-(nx-1), nt-(nx-2)];
fi;
if (m=nt) then;
nodovecinosm:= [nt-(nx+1), nt-nx, nt-1, nt];
fi;
if (1<m<nx) then;
nodovecinosm:= [m-1, m, m+1, (m+nx)-1, (m+nx), m+nx+1];
fi;
if (nt-(nx-1)<m<nt) then;
nodovecinosm:= [m-nx-1, m-nx, m-nx+1, m-1, m, m+1];
fi;
od;
for m from (nx+1) by (nx) to ((nt)-((2nx)-1)) do
nodovecinosm:= [m-nx, m-(nx-1), m, m+1, m+nx, m+nx+1];
od;
for m from (nx+nx) by (nx) to (nt-(nx)) do
nodovecinosm:= [m-nx-1, m-nx, m-1, m, m+(nx-1), m+nx];
od;
for j from 1 to (ny-2) do
  for m from (j-nx+2) to ((j-nx)+(nx-1)) do;
  nodovecinosm:= [m-nx-1, m-nx, m-(nx-1), m-1, m, m+1, m+(nx-1),
  m+nx, m+nx+1];
  od;
od;

```

#"vecinosnodo" da los vecinos del nodo "m" en forma de coordenadas; "a", "b", "c" y "d" son los valores iniciales y finales de los subdominios de cada nodo en "x" y "y"

```

for j from 1 to ny do
  for i from 1 to nx do
    if (j=1) then;
      if (i=1) then;
        vecinosnodoj,i:= [nodo[j,i],nodo[j,i+1],nodo[j+1,i],nodo[j+1,i+1]];
        a[j,i]:=nodo[j,i][1];
        b[j,i]:=nodo[j+1,i+1][1];
        c[j,i]:=nodo[j,i][2];
        d[j,i]:=nodo[j+1,i+1][2];
      fi;
    fi;
    if (j=1) then;
      if (i=nx) then;
        vecinosnodoj,i:= [nodo[j,i-1],nodo[j,i],nodo[j+1,i-1],nodo[j+1,i]];
        a[j,i]:=nodo[j,i-1][1];
        b[j,i]:=nodo[j+1,i][1];
        c[j,i]:=nodo[j,i-1][2];
        d[j,i]:=nodo[j+1,i][2];
      fi;
    fi;
  od;

```

```

fi;
if (j=ny) then;
  if (i=1) then;
    vecinosnodoj,i:= [nodo[j-1,i],nodo[j-1,i+1],nodo[j,i],nodo[j,i+1]];
    a[j,i]:=nodo[j-1,i][1];
    b[j,i]:=nodo[j,i+1][1];
    c[j,i]:=nodo[j-1,i][2];
    d[j,i]:=nodo[j,i+1][2];
  fi;
fi;
if (j=ny) then;
  if (i=nx) then;
    vecinosnodoj,i:= [nodo[j-1,i-1],nodo[j-1,i],nodo[j,i-1],nodo[j,i]];
    a[j,i]:=nodo[j-1,i-1][1];
    b[j,i]:=nodo[j,i][1];
    c[j,i]:=nodo[j-1,i-1][2];
    d[j,i]:=nodo[j,i][2];
  fi;
fi;
if (j=1) then;
  if (1<i<nx) then;
    vecinosnodoj,i:= [nodo[j,i-1],nodo[j,i],nodo[j,i+1],nodo[j+1,i-1],
    nodo[j+1,i],nodo[j+1,i+1]];
    a[j,i]:=nodo[j,i-1][1];
    b[j,i]:=nodo[j+1,i+1][1];
    c[j,i]:=nodo[j,i-1][2];
    d[j,i]:=nodo[j+1,i+1][2];
  fi;
fi;
if (j=ny) then;
  if (1<i<nx) then;
    vecinosnodoj,i:= [nodo[j-1,i-1],nodo[j-1,i],nodo[j-1,i+1],
    nodo[j,i-1],nodo[j,i],nodo[j,i+1]];
    a[j,i]:=nodo[j-1,i-1][1];
    b[j,i]:=nodo[j,i+1][1];
    c[j,i]:=nodo[j-1,i-1][2];
    d[j,i]:=nodo[j,i+1][2];
  fi;
fi;
if (i=1) then;
  if (1<j<ny) then;
    vecinosnodoj,i:= [nodo[j-1,i],nodo[j-1,i+1],nodo[j,i],nodo[j,i+1],
    nodo[j+1,i],nodo[j+1,i+1]];
    a[j,i]:=nodo[j-1,i][1];
    b[j,i]:=nodo[j+1,i+1][1];
    c[j,i]:=nodo[j-1,i][2];
    d[j,i]:=nodo[j+1,i+1][2];
  fi;
fi;

```

```

    fi;
fi;
if (i=nx) then;
  if (1<j<ny) then;
    vecinosnodoj,i:= [nodo[j-1,i-1],nodo[j-1,i],nodo[j,i-1],nodo[j,i],
    nodo[j+1,i-1],nodo[j+1,i]];
    a[j,i]:=nodo[j-1,i-1][1];
    b[j,i]:=nodo[j+1,i][1];
    c[j,i]:=nodo[j-1,i-1][2];
    d[j,i]:=nodo[j+1,i][2];
  fi;
fi;
if (1<j<ny) then;
  if (1<i<nx) then;
    vecinosnodoj,i:= [ nodo[j-1,i-1],nodo[j-1,i],nodo[j-1,i+1],
    nodo[j,i-1],nodo[j,i],nodo[j,i+1],nodo[j+1,i-1],nodo[j+1,i],
    nodo[j+1,i+1]];
    a[j,i]:=nodo[j-1,i-1][1];
    b[j,i]:=nodo[j+1,i+1][1];
    c[j,i]:=nodo[j-1,i-1][2];
    d[j,i]:=nodo[j+1,i+1][2];
  fi;
fi;
od;
od;
#para formar la matriz P del método Petrov-Galerkin
nox:= seq(nodm[1],m=1..nt);
noy:= seq(nodm[2],m=1..nt);

#La constante es de la fórmula general de cuadratura GLL (integral
doble). La cuadratura sólo se evaluó en el subdominio del nodo
(nx+2) esto debido a que los demás nodos interiores poseen los
mismos valores
CONSTANTE:= (((b2,2+0.01)-(a2,2-0.01))·((d2,2+0.01)-(b2,2-0.01)))/4;

#"vn" guarda los vecinos del nodo (nx+2) y "l" los cuenta
vn:= nodovecinos(nx+2);
l:= Count(vn);

#evaluación del nodo (nx+2) y su subdominio con la cuadratura de
GLL (NODOS INTERNOS)
for ñ from 1 to l do
  for o from 1 to 5 do;
    for m from 1 to 5 do;
      # "X" y "Y" son las variables de integración
      X:= (((b2,2+0.01)-(a2,2-0.01))/2)·(ρo)+(((a2,2-0.01)+(b2,2+0.01))/2);
      Y:= (((d2,2+0.01)-(c2,2-0.01))/2)·(ρm)+(((c2,2-0.01)+(d2,2+0.01))/2);
      for j from 1 to nt do;

```

"d" distancia del punto de interés a la posición del nodo
dividida por el parámetro de suavizado "d_{max}"(esto se aplica
para "x" y "y")

dxj:= |(X- nodj[1])| / (1.01·incx);

dyj:= |(Y- nodj[2])| / (1.01·incy);

#función de peso "w" spline de cuarto orden para 2D

$$w_{xj} := \begin{cases} 1 - 6dx_j^2 + 8dx_j^3 - 3dx_j^4 & ; dx_j \leq 1; \\ 0 & ; dx_j > 1 \end{cases}$$

$$w_{yj} := \begin{cases} 1 - 6dy_j^2 + 8dy_j^3 - 3dy_j^4 & ; dy_j \leq 1; \\ 0 & ; dy_j > 1 \end{cases}$$

wxyj:= wxj. wyj;

#derivada de "w" con respecto a "x" cuidando el signo en la
derivada del valor absoluto con la variable "qx"

qx:= X-nodj[1];

ddxj:= |(qx)| / (1.01·incx);

$$wodxj := \begin{cases} -12ddx_j + 24ddx_j^2 - 12ddx_j^3 & ; ddx_j \leq 1; \\ 0 & ; ddx_j > 1 \end{cases}$$

wdxj:= wodx·wyj / (1.01·incx);

if (qx<0) then;

wdxj:= -wodx·wyj / (1.01·incx);

fi;

#derivada de "w" con respecto a "y" cuidando el signo en la
derivada del valor absoluto con la variable "qy"

qy:= Y-nodj[2];

ddyj:= |(qy)| / (1.01·incy);

$$wodyj := \begin{cases} -12ddy_j + 24ddy_j^2 - 12ddy_j^3 & ; ddy_j \leq 1; \\ 0 & ; ddy_j > 1 \end{cases}$$

wdyj:= wody·wxj / (1.01·incy);

if (qy<0) then;

wdyj:= -wody·wxj / (1.01·incy);

fi;

od;

#Mínimos Cuadrados Móviles para la generación de las
funciones de expansión Φ

Vxy:= Vector ([seq(wxyj, j = 1..nt)]);

Wxy:= Matrix (Vxy, shape = diagonal);

pxy:= Vector ([1, X, Y]);

ptxy:= transpose(pxy);

colxy:= seq(1, i = 1..nt);

Pxy:= <<colxy>>|<nox>>|<noy>>;

PTxy:= transpose (Pxy);

```

Bxy:= PTxy.Wxy;
Mxy:= PTxy.Wxy.Pxy;
Mlxy:= MatrixInverse (Mxy);

```

#Para las primeras derivadas en “x” de las funciones de expansión

```

Vdx:= Vector ([seq(wdxj, j = 1..nt)]);
Wdx:= Matrix (Vdx, shape = diagonal);
pdx:= Vector ([0,1,0]);
ptdx:= transpose (pdx);
Bdx:= PTxy.Wdx;
Mdx:= PTxy.Wdx.Pxy;
Mldx:= -Mlxy.Mdx.Mlxy;
Φdx:= ptdx.Mlxy.Bxy+ptxy.Mldx.Bxy+ptxy.Mlxy.Bdx;
laudx:= convert (Φdx, 'list');

```

#Para las primeras derivadas en “y” de las funciones de expansión

```

Vdy:= Vector ([seq(wdyj, j = 1..nt)]);
Wdy:= Matrix (Vdy,shape=diagonal);
pdy:= Vector ([0,0,1]);
ptdy:= transpose (pdy);
Bdy:= PTxy.Wdy;
Mdy:= PTxy.Wdy.Pxy;
Mldy:= -Mlxy.Mdy.Mlxy;
Φdy:= ptdy.Mlxy.Bxy+ptxy.Mldy.Bxy+ptxy.Mlxy.Bdy;
laudy:= convert (Φdy,'list');
puntoxo,m:= ω[o]· ω[m] (laudx[vn[ñ]]· wdx(nx+2));
puntoyo,m:= ω[o]· ω[m] (laudy[vn[ñ]]· wdy(nx+2));
od;

```

```

od;
gaussx:= [seq(seq(puntoxo,m, m=1..5),o=1..5)];
gaussy:= [seq(seq(puntoyo,m, m=1..5),o=1..5)];
sumax:= CONSTANTE (add(gaussx[i], i=1..25));
sumay:= CONSTANTE (add(gaussy[i], i=1..25));
A(nx+2),vn[ñ]:= sumax+sumay;
od;

```

```

P(nx+2):= seq(A(nx+2, vn[ñ]), ñ = 1..l);

```

```

for j from 1 to nt do
  vecj:= nodovecinosj;
od;
for i from 1 to nt do
  canj:= Count (nodovecinosj);
od;

```

#formación de la matriz de rigideces K (MA)

```

for j from 1 to (ny-2) do
  for m from (j·nx+2) to ((j·nx)+(nx-1)) do
    km:= [P(nx+2)];
  od;
od;
for j from 1 to nx do
  if (j = 1) then
    kj:= [1, seq(0, m = 2 .. canj)];
  fi;
  if (j = nx) then
    kj:= [0, 1, seq(0, m = 3 .. canj)];
  fi;
  if 1 < j < nx then
    kj:= [0, 1, seq(0, m = 3 .. canj)];
  fi;
od;
for j from (nt-(nx-1)) to nt do
  if j = (nt-(nx-1)) then
    kj:= [0, 0, 1, seq(0, m = 4 .. canj)];
  fi;
  if (j = nt) then
    kj:= [0, 0, 0, 1];
  fi;
  if (nt-(nx-1)) < j < nt then
    kj:= [0, 0, 0, 0, 1, seq(0, m = 6 .. canj)];
  fi;
od;
for j from (nx+1) by nx to (nt-((2·nx)-1)) do
  kj:= [0, 0, 1, seq(0, m = 4 .. canj)];
od;
for j from (nx+nx) by nx to (nt-nx) do
  kj:= [0, 0, 0, 1, seq(0, m = 5 .. canj)];
od;

```

ARR:= Array(1..nt, 1..nt);

```

for j from 1 to nt do
  for m from 1 to canj do
    ARR(j, nodovecinosjj[m]) := kj[m];
  od;
od;

```

MA:= convert(ARR, Matrix);
 MI:= MatrixInverse(MA);

#condiciones de frontera en vector de carga

```

for m from 1 to nt do
  if 1 ≤ m ≤ nx then
    posm:= 50;
  fi;
fi;

```

```

    fi;
    if (nt-(nx-1)) ≤ m ≤ nt then
        posm:= 100;
    fi;
od;
for m from (nx+1) by nx to (nt-((2·nx)-1)) do
    posm:= 50;
od;
for m from (nx+nx) by nx to (nt-nx) do
    posm:= 50;
od;
for j from 1 to (ny-2) do
    for m from (j·nx+2) to ((j·nx)+(nx-1)) do
        posm:= 0;
    od;
od;
efe:= seq([posm], m = 1..nt);
FE:= Matrix([efe]);
T:= MI.FE;
zetnum:= convert(T,'list');

#generando la base para la graficación
equis:= [seq(seq(k, k=Lo..Lf, incx), m=Wo..Wf, incy)];
yes:= [seq( seq(m, k=Wo..Wf, incy), m=Wo..Wf, incy)];

#graficación
numerica:=pointplot3d(equis,yes,zetnum,axes=FRAMED,symbol=dia
mond,shading=zhue,orientation=[-90,0]);

end proc:
>PG( ):

```


BIBLIOGRAFÍA

Bathe, K. and Wilson, E. (1976). *Numerical Methods in Finite Element Analysis*. Prentice Hall.

Carslaw, H.S and Jaeger, J.C (1959). *Conduction of heat in solids*. Oxford University Press.

Cervantes, J. (1999). *Fundamentos de Transferencia de Calor*. Ediciones Científicas Universitarias.

Fries, T. and Matthies, H. (2004). *Clasificación and Overview of Meshfree Methods*. Scientific Computing.

Karniadakis, G. and Sherwin, S. (1999). *Spectral/hp Element Methods for CFD*. Oxford University Press.

Lancaster, P. and Salkauskas, K. (1981). Surfaces Generated by Moving Least Squares Methods. *Mathematics of Computation*, 37:141–158.

Muñoz, J., González-Casanova, P., and Rodríguez, G. (2005). Métodos Libres de Malla para Ecuaciones Diferenciales Parciales Evolutivas. Technical Report CCC-05-001, Coordinación de Ciencias Computacionales INAOE, Puebla, México.

Nieves Hurtado, Antonio and Domínguez Sánchez, Federico C.(1995) *Métodos numéricos aplicados a la ingeniería*. Edit. CECSA.

Osizik, M. (1993). *Heat Conduction*. Edit. Jhon Wiley & Sons.Inc.

Reddy, J.N. (2005). *An Introduction to the Finite Element Method*, Mc Graw Hill.

Seegerlind, Larry J. (1985). *Applied Finite Element analysis*. Edit John Wiley & Sons, Inc.

Sterk, M., Robic, B., and Trobec, R. (2005). Mesh Free Method Applied to the Diffusion Equation. *Parallel Numerics*, 05:57–66.

Chapra, Steven and Canale, Raymond P. (2003) *Métodos Numéricos para Ingenieros*, Mc. Graw- Hill.

Zill, Dennis G. and Cullen, Michael R. (2008). *Matemáticas avanzadas para ingeniería Vol. 1 Ecuaciones diferenciales*. Edit. Mc. Graw-Hill.