UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CENCIAS

EL MÉTODO DE DIFERENCIAS FINITAS CON FUNCIONES DE BASE RADIAL PARA LA SOLUCIÓN AL PROBLEMA DE LA CAVIDAD CON TAPA DESLIZANTE

T E S I S

QUE PARA OBTENER EL TíTULO DE:

MATEMÁTICO

P R E S E N T A:

JOSÉ RODRIGO ROJO GARCÍA

Director de tesis:

DR. PEDRO GONZÁLEZ CASANOVA HENRÍQUEZ

Ciudad Universitaria, CDMX 2018

1.Datos del alumno
Apellido paterno
Apellido materno
Nombres
Teléfono
Universidad Nacional Autónoma de México
Facultad de Ciencias
Carrera
Número de cuenta

2. Datos del tutor
Grado
Nombre
Apellido paterno
Apellido materno

3. Datos del sinodal 1
Grado
Nombres
Apellido paterno
Apellido materno

4. Datos del sinodal 2
Grado
Nombres
Apellido paterno
Apellido materno

5. Datos del sinodal 3
Grado
Nombres
Apellido paterno
Apellido materno

6. Datos del sinodal 4
Grado
Nombre
Apellido paterno
Apellido materno

7.Datos del trabajo escrito.
Título
Número de páginas
Año

1. Datos del alumno
Rojo
García
José Rodrigo
63 82 72 57
Universidad Nacional Autónoma de México
Facultad de Ciencias
Matemáticas
305256218

2. Datos del tutor
Dr
Pedro
González Casanova
Henríquez

3. Datos del sinodal 1
Dr
Ursula Xiomara
Iturrarán
Viveros

4. Datos del sinodal 2
Dr
Luis Miguel
De la Cruz
Salas

5. Datos del sinodal 3
Dr
Daniel Alejandro
Cervantes
Cabrera

6. Datos del sinodal 4
Dr
Josué
Tago
Pacheco

7. Datos del trabajo escrito

El método de diferencias finitas con funciones de base radial para la solución al problema de la cavidad con tapa deslizante.

113 p.

2018

Dedicatoria

Dedicada cariñosamente a mi familia y todas esas personas que creeyeron en esta aventura.

Agradecimientos

Agradezco el soporte dado por el proyecto de PAPIIT número IN102116 de la UNAM.

A la UNAM por darme la oportunidad de estudiar la carrera simultanea y sobre todo a la Facultad de Ciencias, donde he pasado una de las mejores épocas de mi vida.

A mi tutor el Dr. Pedro González Casanova Henríquez, quien me brindó la oportunidad de trabajar con él. Sus enseñanzas y su infinita paciencia ayudaron a encaminar este proyecto de manera adecuada.

A todos mis sinodales, quienes con sus comentarios han ayudado a pulir este trabajo. También agradezco sus ánimos y consejos personales para encaminar mi futuro.

A mi familia y a mis amigos entrañables, pues me apoyaron incondicionalmente a pesar de lo complejo que era el panorama de una segunda licenciatura en ese momento.

Siglas y Abreviaturas

ACBF	Precondicionador de Aproximación de Funciones de Base Cardinal (por sus siglas en inglés *Approximated Cardinal Basis Functions Preconditioner*)
BIGSTAB	Método del Gradiente Biconjugado Estabilizado (en inglés *Biconjugate Gradient Stabilized*)
BVP	Problema de Valores en la Frontera (por sus siglas en inglés *Boundary Value Problem*)
KAC	Colocación Asimétrica de Kansa (por sus siglas en inglés *Kansa Asymmetric Collocation*)
KSC	Colocación Simétrica de Kansa (por sus siglas en inglés *Kansa Symmetric Collocation*)
DDM	Método de descomposición de dominio (por sus siglas en inglés *Domain Descomposition Method*)
DQ	Cuadratura Diferencial (por sus siglas en inglés *Differential Quadrature*)
DQ-RBF	Método de Cuadratura Diferencial con Funciones de Base Radial
FD	Diferencias Finitas (por sus siglas en inglés *Finite Differences*)
FEM	Método de Elemento Finito (por sus siglas en inglés *Finite Element Method*)
FVM	Método de Volumen Finito (por sus siglas en inglés *Finite Volume Method*)
GA	Kernel Gaussiano (en inglés *Gaussian Kernel*)
GMRES	Método del Residuo Mínimo Generalizado (en inglés *Generalized Minimal Residual*)
IMQ	Kernel Inverso Multicuádrico (en inglés *Inverse Multiquadric Kernel*)

LDC	Problema de la tapa de deslizante (por sus siglas en inglés *Lid Driven Cavity*)
MQ	Kernel Multicuádrico (en inglés *Multiquadric Kernel*)
NASA	Administración Nacional de la Aeronáutica y del Espacio (por sus siglas en inglés *National Aeronautics and Space Administration*)
PDE	Ecuación Diferencial Parcial (por sus siglas en inglés *Partial Differential Equation*)
PHS	Splines Poliharmónicos (en inglés *Polyharmonic Splines*)
RBF	Función de Base Radial (por sus siglas en inglés *Radial Basis Function*)
RBF-FD	Método de Diferencias Finitas con Funciones de Base Radial
SOR	Método de Sobrerelajación Sucesiva (en inglés *Successive Over Relaxation*)
SS	Splines Suaves (en inglés *Smooth Splines*)
TDMA	Algoritmo para Matrices Tridiagonales (en inglés *Tridiagonal Matrix Algorithm*)

Resumen

El presente trabajo tiene como objetivo probar el método de Diferencias Finitas (FD) basadas en la teoría de Funciones de Base Radial (RBFs) para resolver las ecuaciones de Navier Stokes, a saber, el problema de la cavidad con tapa deslizante (LDC). Los métodos basados en RBFs, recientemente han sido ampliamente usados en la solución de Ecuaciones Diferenciales Parciales (PDE) ya que tienen propiedades que las hacen muy atratactivas, como por ejemplo el hecho de que no requieren la construción de mallas, diversas variantes han sido desarrolladas a través del tiempo y en años recientes ha surgido la combinación de RBFs con FD.

Uno de los principales problemas en el uso de los métodos basados en RBFs, es que los sistemas lineales resultantes llamados de Gram, tiene un alto condicionamiento. La variante del método que usaremos genera sistemas locales que permiten mejorar el condicionamiento del problema. Otra forma de mejorar el condicionamiento es aumentar la precisión de máquina, adicionalmente el empleo de un mapeo unitario, similar al que se emplea en la teoría del método de Elemento Finito (FEM), ha permitido mejorar el criterio en la elección del parámetro de forma. Como veremos, existen varios tipos de funciones o kerneles radiales que poseen distintas características, en esta tesis, sin embargo, sólo usaremos el kernel Inverso Multicuádrico (IMQ), debido a que esta función tiene propiedades importantes cuando variamos el llamado parámetro de forma.

En este trabajo nos proponemos comparar el número de condición local contra el parámetro de forma tanto en precisión doble, como cuadruple, con y sin mapeo unitario, cada uno de estos casos para diferente número de nodos locales. Con los resultados obtenidos pudimos elegir un parámetro de forma adecuado para cada caso y resolvimos el problema LDC para cada uno de los casos, comparando los resultados con los obtenidos por Ghia [20] y Cruz [6] quienes usaron FD y RBFs respectivamente. Usando las ventajas generales que nos da cada uno de ellos, los resultados sugirieron que es importante que los sistemas locales se resulevan con precisión cuádruple y mapeo unitario con a los más 9 nodos locales.

Índice general

Índice de tablas

Índice de figuras

Capítulo 1

Introducción

Durante años, una de las principales herramientas en la solución de PDEs ha sido el uso de análisis numérico, donde uno de los métodos más usados es el de FD. Este método ha sido ampliamente estudiado, debido a que en comparación de otros métodos como Volumen Finito (FVM) y FEM es relativamente más sencillo y tiene un costo computacional bajo, sin embargo, FD tiene la límitante de que sólo pueden usarse nodos equiespaciadas en coordenadas cartesianas o curvilineas.

Generalmente cuando se pretende estudiar un método númerico más sofisticado como FEM, no solo se recomienda comparar los resultados con la solución analítica (si es que se conoce), si no resolver también con FD (en los casos donde sea posible) para poder hacer una comparación válida.

Algunos problemas clásicos en la solución de PDEs por métodos numéricos han sido aquellos referentes a la dinámica de fluidos, un ejemplo clásico es el conocido como el problema LDC, de donde algunos trabajos destacables son los de Ghia [20] y Ertuk [8], [9]. Este problema describe el movimiento de un fluido sobre un recipiente cuya tapa es una banda transportadora que mueve la superficie del fluido de manera constante.

Las PDEs que gobiernan el problema LDC se caracterizan por su relativa complejidad en comparación con algunas PDEs clásicas como Poisson u Onda ya que se trata de un sistema fuertemente acoplado, sin embargo, también puede considerarse un problema sencillo en comparación con el sistema definido por las Ecuaciones de Navier Stokes en 3 dimensiones o problemas de electromagnetismo, (consulte por ejemplo Larsson [33] capítulos 12 y 13). Es por esta razón que su solución sirve de parámetro en la solución de PDEs más complejas.

Por otro lado, cuando se necesita resolver PDEs sobre nodos que no se encuentran en un arreglo uniformemente distribuido, uno de los métodos más estudiado ha sido FEM, sin embargo,

recientemente el desarrollo de la teoría de RBFs ha abierto un nuevo campo de estudio, ya que una de las principales ventajas que muestra sobre FEM es su convergencia espectral.

Las RBFs evolucionaron del método multicuádrico propuesto de manera heurística en 1971 por el geodesista Rolland Hardy en la interpolación de datos topográficos (véase Hardy [26]), así como la interpolación basada en Splines de capa delgada propuesta por los ingenieros Harder y Desmarais de la MacNeal-Schwendler Corporation y de la NASA respectivamente, para el diseño de aeronaves (véase Harder y Desmarais [25]), tiempo después fue generalizado como el método de interpolación por RBF y del cual se han formalizado y estudiado diversos aspectos teóricos que más adelante se mencionarán. Entre las aplicaciones más destacables además de la interpolación de datos, se encuentran: problemas de mínimos cuadrados (consulte Hardy [27]), problemas de redes neuronales sintéticas (puede ver Chen et al. [5]) y por supuesto solución a PDEs (véase Sarra y Kansa [40] capítulo 3).

Recientemente la combinación de FD con RBF (RBF-FD) descrita por Tolstykh [45] ha supuesto una mejora en los inconvenientes clásicos de la teoría de las RBFs, como es el caso del mal condicionamiento.

Capítulo 2

Conceptos Básicos

2.1. Normas de matrices

Gran parte de los problemas que se estudian en análisis numérico se reducen a un sistema de ecuaciones lineales cuya solución generalmente nos permite obtener una solución aproximada a nuestro problema original, es por ello que es de suma importancia caracterizar algunas propiedades que guardan las matrices y los vectores involucrados en dicho sistema. Es necesario hacer énfasis que aunque en este trabajo sólo se trabajarán con matrices y vectores en el campo de los reales, algunas definiciones se pueden generalizar al campo de los complejos (para ver una descripción más detallada puede consultar Saad [37] capítulo 1).

Definición 2.1.1. Sea el vector $x \in \mathbb{R}^n$, su norma p se define como 2.1

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p} \tag{2.1}$$

para todo $p \in [1, \infty)$, mientras que en el caso $p = \infty$ la norma está dada por 2.2,

$$\|x\|_p = \max_{1 \leq i \leq n} |x_i| \tag{2.2}$$

es fácil probar que estas normas están bien definidas, es decir cumplen con las propiedades básicas de una norma.

Algo interesante es que apartir de estas normas es posible inducir una norma para las matri-

ces, la cual está dada por la definición 2.1.2.

Definición 2.1.2. Sea una matriz $A \in \mathbb{R}^{n\times m}$, definimos la función $\|\cdot\|_p : \mathbb{R}^{n\times m} \to [0,\infty)$ dada por la expresión 2.3

$$\|A\|_p = \max_{x\in\mathbb{R}^m, x\neq 0} \frac{\|Ax\|_p}{\|x\|_p} \tag{2.3}$$

esta definición tiene como consecuencia la siguiente proposición.

Proposición 2.1.1. *A La función de la Definición 2.1.2, es una norma.*

Demostración. Es fácil probar que dado un escalar real c y otra matriz $B \in \mathbb{R}^{n\times m}$ la función hereda de la norma vectorial las primeras cuatro propiedades listadas abajo.

1. $\|A\|_p \geq 0$

2. $\|cA\|_p = \mid c \mid \|A\|_p$

3. $\|A+B\|_p \leq \|A\|_p + \|B\|_p$

4. Si $A = 0$ entonces $\|A\|_p = 0$

5. Ahora suponga que $\|A\|_p = 0$ y que $A \neq 0$, entonces existen i_0, j_0 tales que $A_{i_0,j_0} \neq 0$. Sea también $\widehat{x} \in \mathbb{R}^m$ tal que $\widehat{x}_j = \delta_{j,j_0}$, entonces tenemos que

$$\|A\|_p = \max_{x\in\mathbb{R}^m, x\neq 0} \frac{\|Ax\|_p}{\|x\|_p} \geq \frac{\|A\widehat{x}\|_p}{\|\widehat{x}\|_p} = \|A_{*j}\|_p \geq \mid A_{i_0,j_0} \mid > 0 \tag{2.4}$$

   lo cual es una contradicción, por tanto $A = 0$.

□

A partir de esta norma se desprenden algunas propiedades interesantes, las cuales son enlistadas en la proposición 2.1.2.

Proposición 2.1.2. *Sean $A \in \mathbb{R}^{n\times m}$, $B \in \mathbb{R}^{m\times l}$ y $y \in \mathbb{R}^m$, entonces se cumple lo siguiente*

1. *Definición equivalente*

$$\|A\|_p = \max_{x\in\mathbb{R}^m, \|x\|=1} \|Ax\|_p \tag{2.5}$$

2. *Propiedad de consistencia vectorial*

$$\|Ay\|_p \leq \|A\|_p \|y\|_p \tag{2.6}$$

3. *Propiedad de consistencia matricial*

$$\|AB\|_p \leq \|A\|_p \|B\|_p \tag{2.7}$$

Demostración. De la definición y usando las propiedades de la norma vectorial tenemos lo siguiente

1.

$$\max_{x\in\mathbb{R}^m, x\neq 0} \frac{\|Ax\|_p}{\|x\|_p} = \max_{x\in\mathbb{R}^m, x\neq 0} \|x\|_p \frac{\left\|A\frac{x}{\|x\|_p}\right\|_p}{\|x\|_p} = \max_{x\in\mathbb{R}^m, x\neq 0} \left\|A\frac{x}{\|x\|_p}\right\|_p = \max_{z\in\mathbb{R}^m, \|z\|=1} \|Az\|_p \tag{2.8}$$

2. Caso 1 $\quad y = 0$

   Se cumple la igualdad en este caso

   Caso 2 $\quad y \neq 0$

$$\|A\|_p = \max_{x\in\mathbb{R}^m, x\neq 0} \frac{\|Ax\|_p}{\|x\|_p} \geq \frac{\|Ay\|_p}{\|y\|_p} \tag{2.9}$$

   de la desigualdad anterior tenemos que $\|Ay\|_p \leq \|A\|_p \|y\|_p$

3. Sea $y \in \mathbb{R}^m$ unitario tal que $\|ABy\|_p = \|AB\|_p$, usando la propiedad de consistencia vectorial tenemos que

$$\|ABy\|_p \leq \|A\|_p \|By\|_p \tag{2.10}$$

aplicando nuevamente esta propiedad tenemos que

$$\|A\|_p \|By\|_p \leq \|A\|_p \|B\|_p \|y\|_p \tag{2.11}$$

dado que y es unitario se satisface la propiedad.

□

En el caso en el que B es ortogonal ($B^T = B^{-1}$) se da la igualdad con $p = 2$, la cual se establece con la proposición 2.1.3

Proposición 2.1.3. *Sea Q ortogonal, entonces para $p = 2$ se tiene la igualdad 2.12, más aún $\|Q\|_2 = 1$*

$$\|AQ\|_2 = \|A\|_2 \tag{2.12}$$

Demostración. Sea y unitario tal que $\|Qy\|_2 = \|Q\|_2$, tenemos entonces que,

$$\|Q\|_2^2 = \|Qy\|_2^2 = (Qy)^T(Qy) = y^T(Q^TQ)y = \|y\|_2^2 = 1 \tag{2.13}$$

usando la consistencia y el hecho de que la norma de Q es unitaria tenemos que

$$\|AQ\|_2 \leq \|A\|_2 \|Q\|_2 = \|A\|_2 \tag{2.14}$$

por otro lado, usando el hecho de que Q^T también es ortogonal, tenemos la siguiente desigualdad.

$$\|A\|_2 = \left\|(AQ)Q^T\right\|_2 \leq \|AQ\|_2 \left\|Q^T\right\|_2 = \|AQ\|_2 \tag{2.15}$$

Combinando ambas desigualdades tenemos la igualdad deseada.

□

De manera análoga es posible probar que dada Q ortogonal, se cumple la igualdad $\|AQ\|_2 = \|A\|_2$, lo cual nos lleva a la siguiente proposición 2.1.4.

Proposición 2.1.4. *Sea A simétrica, entonces se satisface la ecuación 2.16*

$$\|A\|_2 = |\lambda_{max}| =: \varrho(A) \tag{2.16}$$

Demostración. Dado que A es simétrica, existe una matriz Q ortogonal de eigenvectores tal que $A = Q^T diag(\lambda_1, \lambda_2, \cdots, \lambda_n)Q$, por lo que

$$\begin{aligned} \|A\|_2 &= \left\|Q^T diag(\lambda_1, \lambda_2, \cdots, \lambda_n)Q\right\|_2 \\ &= \left\|Q^T diag(\lambda_1, \lambda_2, \cdots, \lambda_n)\right\|_2 \\ &= \|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)\|_2 \end{aligned} \tag{2.17}$$

sea z el eigenvector unitario asociado al radio espectral, entonces

$$\|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)z\|_2 = \|\lambda_{max}z\|_2 = \varrho(A) \tag{2.18}$$

por lo que

$$\|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)\|_2 \geq \varrho(A). \tag{2.19}$$

Sea ahora $x = (x_1, x_2, \cdots, x_n)^T \in \mathbb{R}^n$ unitario, entonces tenemos la siguiente estimación

$$\|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)x\|_2^2 = \sum_{i=1}^{n} \lambda_i^2 x_i^2 \leq \varrho(A)^2 \sum_{i=1}^{n} x_i^2 = \varrho(A)^2 \tag{2.20}$$

de donde deducimos que $\varrho(A)$ es cota superior de $\|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)\|_2^2$, y podemos concluir que

$$\varrho(A) \leq \|diag(\lambda_1, \lambda_2, \cdots, \lambda_n)\|_2 \leq \varrho(A) \tag{2.21}$$

□

Finalmente como corolario, es posible calcular la norma de A^{-1} si A es simétrica.

Corolario 2.1.5. *Sea A simétrica e invertible, entonces la norma de A^{-1} está dada por la ecuación 2.22*

$$\left\|A^{-1}\right\|_2 = \frac{1}{|\lambda_{min}|} \tag{2.22}$$

Demostración. Por ser A simétrica, está se puede representar como $A = Q^T diag(\lambda_1, \lambda_2, \cdots, \lambda_n)Q$. Por lo que al calcular la inversa se tiene la ecuación 2.23

$$A^{-1} = Q^T diag(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \cdots, \frac{1}{\lambda_n})Q \tag{2.23}$$

usando la proposición 2.1.3 tenemos,

$$\left\|A^{-1}\right\|_2 = \left\| diag(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \cdots, \frac{1}{\lambda_n}) \right\|_2 \tag{2.24}$$

Finalmente, al ser la matriz diagonal simétrica es posible usar la proposición 2.1.4 de donde se concluye el valor de la norma.

□

2.2. Matrices definidas positivas

Uno de los problemas principales del álgebra lineal numérica es la solución al sistema de ecuaciones lineales (cuando son matrices cuadradas).

$$Ax = b \tag{2.25}$$

donde la solución a dicho sistema existe y es única si y sólo si $det(A) \neq 0$, existen también otras caracterizaciones de las cuales una clásica es la siguiente afirmación.

Proposición 2.2.1. *El sistema 2.25 tiene solución única si y sólo si todos los eigenvalores λ_i son diferentes de cero.*

Demostración. Para ver una prueba, puede consultar Golan [22] página 129. □

Dado que en general el cálculo tanto del determinante como de los eigenvalores es complicado, se puede hacer uso de propiedades de las matrices para establecer su invertibilidad.

Definición 2.2.1. Sea una matriz $A \in \mathbb{R}^{n\times n}$ simétrica, se dice que es simétrica definida positiva si para todo $x \in \mathbb{R}^n$ no nulo se cumple la desigualdad 2.26

$$x^T Ax > 0 \tag{2.26}$$

La importancia de este tipo de matrices es que son no singulares, tal y como lo establece la proposición 2.2.2.

Proposición 2.2.2. *Sea una matriz $A \in \mathbb{R}^{n\times n}$ simétrica definida positiva, entonces todos sus eigenvalores son positivos y por tanto es no singular.*

Demostración. Sea y un eigenvector de A asociado a su eigenvalor λ, por definición se tiene que

$$y^T(Ay) = y^T(\lambda y) = \lambda \|y\|_2^2 \tag{2.27}$$

dado que A es definida positiva, entonces $\lambda > 0$.

□

Las matrices simétricas definidas positivas son comunes en diversos métodos numéricos, lo que da lugar a que los problemas a resolver estén bien planteados si se conoce dicha propiedad desde un principio.

A pesar de poder asegurar la no singularidad en aritmética real, existen matrices que son singulares en aritmética de punto flotante, dicho inconveniente sirvió de inspiración para establecer una propiedad que indique que tan cercana a singular puede ser una matriz. Una forma de medir lo anterior es através de perturbaciones en la solución al sistema de ecuaciones 2.25 con el número de condición.

2.3. Número de condición

Dado que en aritmética de punto flotante sólo se trabaja con un número finito de dígitos, difícilmente es posible encontrar una solución exacta al sistema de ecuaciones 2.25 con algún método numérico. Es por ello que si la solución exacta es x, la calculada la llamaremos x_c.

Definición 2.3.1. .

1. El error del sistema 2.25 está dadó por 2.28

$$e = x - x_c \tag{2.28}$$

2. El residual del sistema 2.25 se define como 2.29

$$r = b - Ax_c \tag{2.29}$$

Definición 2.3.2. Sea una matriz $A \in \mathbb{R}^{n\times n}$ no singular, entonces su número de condición se define por la expresión 2.30.

$$\kappa_p(A) = \|A\|_p \left\|A^{-1}\right\|_p \tag{2.30}$$

Adicionalmente podemos decir que para matrices simétricas, es posible calcular el número de condición con los eigenvalores de la matriz.

Proposición 2.3.1. *Sea A una matriz no singular simétrica, entonces el número de condición con la norma $\|\cdot\|_2$ está dado por*

$$\kappa_2(A) = \frac{|\lambda_{max}|}{|\lambda_{min}|} \tag{2.31}$$

Demostración. De la proposición 2.1.4 y el corolario 2.1.5 tenemos que.

$$\begin{aligned} \kappa_2(A) &= \|A\|_2 \left\|A^{-1}\right\|_2 \\ &= |\lambda_{max}| \frac{1}{|\lambda_{min}|} \\ &= \frac{|\lambda_{max}|}{|\lambda_{min}|} \end{aligned} \tag{2.32}$$

□

Si bien, nos hemos enfocado en $p = 2$, la equivalencia de las normas en dimensión finita implica que los números de condición son equivalentes y por tanto éstos tienen aproximadamente los mismos órdenes de magnitud. En la práctica es común usar $p = 1$ y ∞ ya que su cálculo es relativamente sencillo (para ver más detalles consulte Saad [37] página 9).

Para entender como influye el número de condición en la solución al sistema 2.25, se tiene la siguiente estimación para el error relativo.

Proposición 2.3.2. *Sea A una matriz no singular y sea b un vector no nulo, entonces se cumple la desigualdad 2.33*

$$\frac{\|e\|_p}{\|x\|_p} \leq \kappa_p(A)\frac{\|r\|_p}{\|b\|_p} \tag{2.33}$$

Demostración. Sabemos que $Ae = r$, por lo que al despejar el error éste está dado por $e = A^{-1}r$, usando la propiedad 2.6 tenemos la desigualdad 2.34

$$\|e\|_p = \left\|A^{-1}r\right\|_p \leq \left\|A^{-1}\right\|_p \|r\|_p \tag{2.34}$$

por otro lado, de 2.25 es clara la desigualdad $\|b\|_p = \|Ax\|_p \leq \|A\|_p \|x\|_p$, y de ésta última tenemos que

$$\frac{1}{\|x\|_p} \leq \frac{\|A\|_p}{\|b\|_p} \tag{2.35}$$

combinando las desigualdades 2.34 y 2.35 se tiene la estimación deseada.

□

La proposición 2.3.2 nos dice que mientras más pequeño sea el número de condición entonces el error relativo es pequeño, mientras que si el número de condición es muy grande no se puede asegurar nada sobre el error relativo. Sin embargo existe una aproximación heurística dada por Golub en [23] página 138,

$$\frac{\|e\|_p}{\|x\|_p} \approx 10^{-d}\kappa_p(A) \tag{2.36}$$

donde d es el número de digitos decimales usados en la precisión de la maquina, para el caso de la precisión doble y cuadruple este valor es de 16 y 32 respectivamente, por lo que si el número de condición es del orden de 10^q con $q > d$, entonces el sistema 2.25 tendría una solución totalmente erronea y por tanto singular en aritmética de punto flotante.

Los métodos para resolver los sistemas de ecuaciones lineales pueden dividirse en dos categorias: aquellos que son directos y los iterativos, los directos son aquellos que se basan en

algoritmos con un número fijo de operaciones, por ejemplo: Eliminación Gaussiana, Descomposición LU, Descomposición de Cholesky y el algoritmo de Thomas entre otros, mientras que los iterativos como lo dice su nombre, van aproximando la solución mediante iteraciones las cuales se detienen hasta que alcancen una tolerancia o un número máximo de iteraciones, algunos ejemplos son: Jacobi, Gauss Seidel, el método de Sobrerelajación Sucesiva (SOR), Gradiente Conjugado, el método del Gradiente Biconjugado Estabilizado (BIGSTAB) y el método del Residuo Mínimo Generalizado (GMRES), entre otros.

En el caso de los métodos directos, la ventaja principal es que el error relativo sólo depende de las operaciones en aritmética de punto flotante y en consecuencia del número de condición de la matriz, sin embargo, generalmente estos métodos son costosos, dando complejidades operacionales del orden de $\mathcal{O}(N^3)$ y en algunos casos se necesita almacenamiento en memoria del orden $\mathcal{O}(N^2)$. Por otro lado los métodos iterativos pueden tener complejidades que dependen del número de iteraciones m, el número de elementos diferentes de cero N_z y la dimensión de la matriz N, por ejemplo, GMRES realiza $(m + 3 + 1/m)N + N_z$ multiplicaciones y necesita un almacenamiento de $(m+2)N$ (para ver más detalles puede consultar a Saad [38]). El problema principal de los métodos iterativos es que el número de iteraciones necesarios para alcanzar la tolerancia (si es que se alcanza) se incrementa cuando el número de condición es muy alto, además de que el error de truncamiento asociado al número de iteraciones se añade al ya propio error asociado al número de condición.

Si bien en varios métodos numéricos como FD, FVM y algunos problemas de interpolación es preferible usar métodos iterativos por el buen condicionamiento del sistema, en otros podría ser preferible usar descomposición LU si la matriz es pequeña (con $N \leq 100$).

2.4. Diferencias Finitas

A pesar de que actualmente existe una teoría bastante consolidada sobre la existencia y unicidad de algunos tipos de ecuaciones diferenciales tanto oridanarias como parciales, en muchos casos no es posible establecer la solución de manera explícita. Por ello, se han desarrollado diferentes métodos numéricos que han ayudado a establecer un solución aproximada en un número finito de nodos.

El método de diferencias finitas se basa en aproximar las derivadas de una función por medio de evaluaciones de ella en diferentes nodos, por ejemplo, sea $f : \Omega \subseteq \mathbb{R} \to \mathbb{R}$ una función suficientemente suave y supongamos que queremos calcular la primera derivada. Haciendo uso de la expansión en Serie de Taylor tenemos que la traslación de x hacia adelante en $0 < h < 1$ unidades, se puede expresar por las ecuación 2.37

$$f(x+h) = f(x) + h\frac{df}{dx}(x) + \mathcal{O}(h^2) \tag{2.37}$$

de la expresión 2.37 se tiene que la derivada se puede calcular por las expresión 2.38.

$$\frac{df}{dx}(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h) \tag{2.38}$$

donde $\mathcal{O}(h)$ es el error de truncamiento (orden h).

De manera análoga se puede hacer la expansión en Serie de Taylor de $f(x-h)$, la cual está dada por 2.39

$$f(x-h) = f(x) - h\frac{df}{dx}(x) + \mathcal{O}(h^2) \tag{2.39}$$

como consecuencia la derivada se puede también expresar por 2.40

$$\frac{df}{dx}(x) = \frac{f(x) - f(x-h)}{h} + \mathcal{O}(h) \tag{2.40}$$

Para obtener un error de truncamiento menor a los anteriores, es posible usar más términos en la expansión por Serie de Taylor, dando las expresiones 2.41 y 2.42.

$$f(x+h) = f(x) + h\frac{df}{dx}(x) + \frac{h^2}{2}\frac{d^2f}{dx^2}(x) + \mathcal{O}(h^3) \tag{2.41}$$

$$f(x-h) = f(x) - h\frac{df}{dx}(x) + \frac{h^2}{2}\frac{d^2f}{dx^2}(x) + \mathcal{O}(h^3) \tag{2.42}$$

restando 2.42 y 2.41 y dividiendo sobre h, tenemos que la primera derivada está dada por 2.43

$$\frac{df}{dx}(x) = \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2) \tag{2.43}$$

Supongamos ahora que se quiere aproximar la primera derivada en el punto x_i con valores discretos de la función $f_i = f(x_i)$ en una malla uniforme de N puntos, entonces es posible definir el paso $h = x_{i+h} - x_i$ para todo $i = 1, 2, \cdots, N-1$, el cual al usar la expresión 2.41 da lugar al esquema de diferencias finitas **hacia adelante** de para la aproximación de la primera derivada. Dicho esquema está dado por la ecuación 2.44

Hacia adelante

$$\frac{df}{dx}(x_i) \approx \frac{f_{i+1} - f_i}{h} = \begin{pmatrix} -\frac{1}{h} & \frac{1}{h} \end{pmatrix} \begin{pmatrix} f_i & f_{i+1} \end{pmatrix}^T \tag{2.44}$$

de manera análoga se pueden definir los esquemas de diferencias finitas **hacia atrás** y **centrales**, los cuales están dados respectivamente por las expresiones 2.45 y 2.46.

Hacia atrás

$$\frac{df}{dx}(x_i) \approx \frac{f_i - f_{i-1}}{h} = \begin{pmatrix} -\frac{1}{h} & \frac{1}{h} \end{pmatrix} \begin{pmatrix} f_{i-1} & f_i \end{pmatrix}^T \tag{2.45}$$

Centrales

$$\frac{df}{dx}(x_i) \approx \frac{f_{i+1} - f_{i-1}}{2h} = \begin{pmatrix} -\frac{1}{2h} & 0 & \frac{1}{2h} \end{pmatrix} \begin{pmatrix} f_{i-1} & f_i & f_{i+1} \end{pmatrix}^T \tag{2.46}$$

En el caso de la segunda derivada es sencillo probar que al promediar la segunda derivada en las expresiones 2.41 y 2.42, el esquema **central** tiene un error de truncamiento $\mathcal{O}(h^2)$ y cuya aproximación está dada por la expresión 2.47,

$$\frac{d^2f}{dx^2}(x_i) \approx \begin{pmatrix} \frac{1}{h^2} & \frac{-2}{h^2} & \frac{1}{h^2} \end{pmatrix} \begin{pmatrix} f_{i-1} & f_i & f_{i+1} \end{pmatrix}^T \tag{2.47}$$

en general la derivada en un punto u_i puede aproximarse como combinación lineal de todos los valores $\{u_j\}_{j=1}^{j=N}$ con los pesos $\{\omega_{ij}\}_{j=1}^{j=N}$ como 2.48,

$$\frac{du}{dx}(x_i) \approx \sum_{j=1}^{N} \omega_{ij} u_j = \begin{pmatrix} \omega_{i1} & \cdots & \omega_{iN} \end{pmatrix} \begin{pmatrix} u_1 & \cdots & u_N \end{pmatrix}^T \tag{2.48}$$

calcular las derivadas en todos los nodos es posible con el producto matriz vector 2.49.

$$\begin{pmatrix} \frac{du}{dx}(x_1) \\ \vdots \\ \frac{du}{dx}(x_N) \end{pmatrix} \approx \begin{pmatrix} \omega_{11} & \cdots & \omega_{1N} \\ \vdots & \ddots & \vdots \\ \omega_{N1} & \cdots & \omega_{NN} \end{pmatrix} \begin{pmatrix} u_1 \\ \vdots \\ u_N \end{pmatrix} \tag{2.49}$$

Elegir los pesos, es un tema de estudio bastante discutido, por ejemplo, existen algunos trabajos como los de Fornberg [14], [15] en los que se exponen a detalle algoritmos eficientes para calcular los pesos de diferencias finitas así como tablas para elegirlos desde un principio en diferentes tipos de mallas.

La forma de expresar los pesos en una matriz se puede generalizar a un operador diferencial $\mathfrak{D}$, por lo que es posible escribirlo de manera lineal por la aproximación 2.50

$$\mathfrak{D}U \approx WU \tag{2.50}$$

Una forma de calcular los pesos en una dimensión es con el método de los coeficientes indeterminados, el cual consiste en aproximar los pesos de W correspondientes al nodo central x_ξ a partir de una base polinomial con sus correspondientes nodos vecinos $\{x_{\sigma_k}\}_{k=1}^{N_\xi}$ en una malla. En el caso de una dimensión resulta el sistema lineal 2.51, donde los coeficientes γ_{σ_k} son los pesos correspondientes al nodo central y a la matriz de dicho sistema se le conoce como matriz de Vandermonde, la cual se denota como $V_{N_\xi}(x_{\sigma_1}, x_{\sigma_2}, \cdots, x_{\sigma_{N_\xi}})$ o de manera abreviada como V_{N_ξ}. Es importante mencionar que la matriz de Vandermonde también es usada para construir interpoladores polinómicos.

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_{\sigma_1} & x_{\sigma_2} & \cdots & x_{\sigma_{N_\xi}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{\sigma_1}^{N_\xi-1} & x_{\sigma_2}^{N_\xi-1} & \cdots & x_{\sigma_{N_\xi}}^{N_\xi-1} \end{pmatrix} \begin{pmatrix} \gamma_{\sigma_1} \\ \gamma_{\sigma_2} \\ \vdots \\ \gamma_{\sigma_{N_\xi}} \end{pmatrix} = \begin{pmatrix} \mathfrak{D}1|_{x_\xi} \\ \mathfrak{D}x|_{x_\xi} \\ \vdots \\ \mathfrak{D}x^{N_\xi-1}|_{x_\xi}) \end{pmatrix} \tag{2.51}$$

El sistema ya mencionado siempre tiene solución única, pues la matriz de Vandermonde tiene determinante no nulo, lo cual se probará a continuación.

Teorema 2.4.1. *Sean los nodos $\{x_i\}_{i=1}^N \subset \mathbb{R}$ y sea $V_N(x_1, x_2, \cdots, x_N)$ su matriz de Vandermonde, entonces el determinante está dado por*

$$det(V_N) = \prod_{1\le i<j\le N} (x_i - x_j), \tag{2.52}$$

más aún, si todos los nodos son distintos entonces es invertible.

Demostración. Primero observemos que si al renglon $i+1$ le restamos el renglón i multiplicado por x_N, el determinante no cambia. En efecto, usando el hecho de que dos renglones repetidos implican un determinante nulo, el determinante de Vandermonde se puede escribir por medio de la expresión 2.54.

$$
\begin{aligned}
det(V_N) &= det(V_N) + 0 \\
&= det\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_1^i & x_2^i & \cdots & x_N^i \\ x_1^{i+1} & x_2^{i+1} & \cdots & x_N^{i+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{N-1} & x_2^{N-1} & \cdots & x_N^{N-1} \end{pmatrix} - x_N det\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_1^i & x_2^i & \cdots & x_N^i \\ x_1^i & x_2^i & \cdots & x_N^i \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{N-1} & x_2^{N-1} & \cdots & x_N^{N-1} \end{pmatrix}
\end{aligned} \tag{2.53}
$$

Usando las propiedades multilineales del determinante, multiplicamos el escalar x_N por el i-ésimo renglón del segundo determinante de la expresión 2.52 y sumamos con el determinante de Vandermonde, de donde tenemos 2.54.

$$
\begin{aligned}
det(V_N) &= det\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{i+1} - x_1^i x_N & x_2^{i+1} - x_2^i x_N & \cdots & x_N^{i+1} - x_N^i x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{N-1} & x_2^{N-1} & \cdots & x_N^{N-1} \end{pmatrix} \\
&= det\begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_N \\ \vdots & \vdots & \ddots & \vdots \\ x_1^i(x_1 - x_N) & x_2^i(x_2 - x_N) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{N-1} & x_2^{N-1} & \cdots & x_N^{N-1} \end{pmatrix}
\end{aligned} \tag{2.54}
$$

Usando este mismo argumento podemos probar que la matriz de Vandermonde se puede calcular como

$$det(V_N) = det \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 - x_N & x_2 - x_N & x_3 - x_N & \cdots & 0 \\ x_1(x_1 - x_N) & x_2(x_2 - x_N) & x_3(x_3 - x_N) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^{N-2}(x_1 - x_N) & x_2^{N-2}(x_2 - x_N) & x_3^{N-2}(x_3 - x_N) & \cdots & 0 \end{pmatrix} \tag{2.55}$$

Ahora procederemos por inducción, queda claro que para el caso $N = 1$ no hay nada que hacer y para $N = 2$, la matriz de Vandermonde satisface la igualdad 2.52. Supongamos que es cierto para $N - 1$, entonces

$$\begin{aligned} det(V_N) &= det \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 - x_N & x_2 - x_N & x_3 - x_N & \cdots & 0 \\ x_1(x_1 - x_N) & x_2(x_2 - x_N) & x_3(x_3 - x_N) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^{N-2}(x_1 - x_N) & x_2^{N-2}(x_2 - x_N) & x_3^{N-2}(x_3 - x_N) & \cdots & 0 \end{pmatrix} \\ &= det \begin{pmatrix} x_1 - x_N & x_2 - x_N & x_3 - x_N & \cdots & x_{N-1} - x_N \\ x_1(x_1 - x_N) & x_2(x_2 - x_N) & x_3(x_3 - x_N) & \cdots & x_{N-1}(x_{N-1} - x_N) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^{N-2}(x_1 - x_N) & x_2^{N-2}(x_2 - x_N) & x_3^{N-2}(x_3 - x_N) & \cdots & x_{N-1}^{N-2}(x_{N-1} - x_N) \end{pmatrix} \\ &= det(V_{N-1}(x_1, \cdots, x_{N-1})) \prod_{i=1}^{N-1}(x_i - x_N) \end{aligned} \tag{2.56}$$

Pero por hipótesis de inducción tenemos que

$$\begin{aligned} det(V_{N-1}(x_1, \cdots, x_{N-1})) \prod_{i=1}^{N-1}(x_i - x_N) &= \prod_{1 \le i < j \le N-1}(x_i - x_j) \prod_{i=1}^{N-1}(x_i - x_N) \\ &= \prod_{1 \le i < j \le N}(x_i - x_j) \end{aligned} \tag{2.57}$$

que era lo que se deseaba probar.

□

Ya calculados los coeficientes γ_{σ_k} se tiene que para todo $\xi = 1, 2, \ldots, N$ los pesos son redefinidos con la función de Kronecker como en la expresión 2.58,

$$\omega_{\xi j} = \gamma_{\sigma_k} \delta_{\sigma_k j} \tag{2.58}$$

claramente el método proporciona un menor error conforme aumenta el número de nodos vecinos alrededor del nodo central, sin embargo en dicho caso la matriz es muy mal condicionada, por lo que en la práctica algunos problemas clásicos como la solución a la Ecuación de Poisson usan un esquema **central** de pocos nodos para un problema de valores en la frontera tipo Dirichlet, dando lugar así a matrices tridiagonales con un buen número de condición y con propiedades especiales (simétricas, definidas positivas y diagonal dominantes) cuya solución es calculada facilmente con el algoritmo de Thomas o bien por métodos iterativos.

Es importante señalar que en más de una dimensión no se puede hacer el cálculo de los pesos por medio de la matriz de Vandermonde, para ello consideremos los siguientes contraejemplos.

En dos dimensiones, el interpolante lineal define la ecuación de un plano en $\mathbb{R}^3$, por lo que se necesitan al menos 3 puntos no colineales para definir un plano. La matriz de Vandermonde en este caso está dada por la expresión 2.59

$$V_N = \begin{pmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix}, \tag{2.59}$$

supongamos ahora que los puntos pasan por la recta $ax + by + c = 0$, entonces tenemos la igualdad 2.60.

$$\begin{pmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{pmatrix}^T \begin{pmatrix} c \\ a \\ b \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.60}$$

Dado que los coeficientes a, b y c no pueden ser simultaneamente cero, la matriz V_N^T tiene un eigenvalor cero y por tanto es singular, pero como V_N tiene los mismos eigenvalores, entonces V_N también lo es.

Otro ejemplo importante es la interpolación cuadrática en 2D, en este caso se necesitan 6 puntos para construir la matriz de Vandermonde, la cual se puede escribir como 2.61

$$V_N = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ y_1 & y_2 & y_3 & y_4 & y_5 & y_6 \\ x_1^2 & x_2^2 & x_3^2 & x_4^2 & x_5^2 & x_6^2 \\ x_1y_1 & x_2y_2 & x_3y_3 & x_4y_5 & x_5y_6 & x_6y_6 \\ y_1^2 & y_2^2 & y_3^2 & y_4^2 & y_5^2 & y_6^2 \end{pmatrix}, \tag{2.61}$$

supongamos en este caso que los 6 puntos pasan por la cónica $ax^2+bxy+cy^2+dx+ey+f=0$, entonces tenemos el sistema 2.62

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ y_1 & y_2 & y_3 & y_4 & y_5 & y_6 \\ x_1^2 & x_2^2 & x_3^2 & x_4^2 & x_5^2 & x_6^2 \\ x_1y_1 & x_2y_2 & x_3y_3 & x_4y_4 & x_5y_5 & x_6y_6 \\ y_1^2 & y_2^2 & y_3^2 & y_4^2 & y_5^2 & y_6^2 \end{pmatrix}^T \begin{pmatrix} f \\ d \\ e \\ a \\ b \\ c \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.62}$$

Dado que los coeficientes de la cónica no pueden ser todos simultaneamente cero, entonces de manera análoga al ejemplo anterior tenemos que V_N^T es singular y V_N también lo es.

Es por ello que si se requieren calcular los pesos en 2D usando el método de los coeficientes indeterminados se deben elegir los nodos de manera que no pasen por una cónica, lo cual es muy complicado. En estos casos, para calcular los pesos de un operador diferencial por FD se usan productos tensoriales. Algunos ejemplos clásicos son el Laplaciano a 5 y 9 puntos.

2.5. Algoritmo de Thomas

El algoritmo para Sistemas Tridiagonales (TDMA), también conocido como algoritmo de Thomas, es un método directo basado en eliminación gaussiana y que es ampliamente usado en la resolución de sistemas lineales tridiagonales, es decir aquellos que son de la forma 2.63

$$Ay = z \tag{2.63}$$

donde

$$A = \begin{pmatrix} a_1 & c_1 & 0 & 0 & \cdots & 0 \\ b_2 & a_2 & c_2 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & b_{N-2} & a_{N-2} & c_{N-2} & 0 \\ \vdots & \ddots & \ddots & b_{N-1} & a_{N-1} & c_{N-1} \\ 0 & \cdots & 0 & 0 & b_N & a_N \end{pmatrix} \tag{2.64}$$

El proceso para resolver dicho sistema está dado por el algoritmo 1

Algoritmo 1 Algoritmo TDMA

Entrada: A, z.

1: $w = a_1$, $y_1 = \dfrac{z_1}{w}$

Salida: Solución al sistema $Ay = z$.

2: **para** $i = 2, 3, ..., N$ **hacer**
3: $\quad v_i = \dfrac{c_i - 1}{w}$
4: $\quad w = a_i - b_i v_i$
5: $\quad y_i = \dfrac{z_i - b_i y_{i-1}}{w}$
6: **fin para**
7: **para** $j = N - 1, N - 2, ..., 1$ **hacer**
8: $\quad y_j = y_j - v_{j+1} y_{j+1}$
9: **fin para**

Para que dicho algoritmo funcione de manera adecuada, la matriz tridiagonal además de ser invertible debe ser diagonal dominante, lo cual se define a continuación.

Definición 2.5.1. Decimos que una matriz A es:

1. Diagonal dominante si

$$|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|, \qquad \forall i \in \{1, 2, \cdots, N\} \tag{2.65}$$

2. Estrictamente diagonal dominante si

$$|a_{ii}| > \sum_{i \neq j} |a_{ij}|, \qquad \forall i \in \{1, 2, \cdots, N\} \tag{2.66}$$

Algunos teoremas importantes que se desprenden de la diagonal dominancia y estan relacionados con el TDMA son los siguientes.

Teorema 2.5.1 (Levy-Desplanques)**.** *Sea A una matriz estrictamente diagonal dominante, entonces A es invertible.*

Demostración. Procederemos por reducción al absurdo, supongamos que A es singular, entonces existe un eigenvalor nulo $\lambda = 0$ con su eigenvector asociado $x = (x_1, x_2, \ldots, x_N)^T$. Definimos a q como el subíndice para el cual $|x_q| = max\{|x_j|\}_{j=1}^{N}$, claramente este elemento x_q es no nulo por ser x eigenvector de A y para cualquier subíndice j cumple la desigualdad $\frac{|x_j|}{|x_q|} \leq 1$.

Por otro lado, como A es estrictamente diagonal dominante, entonces al multiplicar el renglon q de A por x tenemos la igualdad 2.67

$$\sum_{j=1}^{N} a_{qj}x_j = 0 \tag{2.67}$$

de donde al despejar el término $a_{qq}x_q$ llegamos a 2.68

$$\begin{aligned} |a_{qq}x_q| &= |a_{qq}||x_q| \\ &= |\sum_{q\neq j} a_{qj}x_j| \\ &\leq \sum_{q\neq j} |a_{qj}x_j| \\ &\leq \sum_{q\neq j} |a_{qj}||x_j| \end{aligned} \tag{2.68}$$

Dividiendo entre $|x_q|$, obtenemos la desigualdad 2.69,

$$|a_{qq}| \leq \sum_{q\neq j} |a_{qj}| \tag{2.69}$$

lo cual es una contradicción, ya que A es estrictamente diagonal dominante.

□

Teorema 2.5.2. *Sea A una matriz tridiagonal, invertible y diagonal dominante, entonces el sistema $Ay = z$ puede resolverse con el TDMA.*

Demostración. Puede consultar la prueba en Dahlquist y Björck [7]. □

Para ilustrar como funciona este método en el caso de diferencias finitas usaremos el Ejemplo 1.

Ejemplo 1. Sea el problema de valores en la frontera (BVP) dado por la ecuación de Poisson en una dimensión 2.70.

$$\begin{aligned} \nabla^2 u(x) &= f(x) \qquad 0 < x < 1 \\ u(x) &= a \qquad x = 0 \\ u(x) &= b \qquad x = 1 \end{aligned} \tag{2.70}$$

Suponiendo una malla equiespaciada de $N+2$ nodos y paso h, entonces, el esquema **central** tiene como discretización el sistema 2.71.

$$\begin{aligned} a - 2u_1 + u_2 &= h^2 f_1 \\ u_{i-1} - 2u_i + u_{i+1} &= h^2 f_i \qquad 1 < i < N \\ u_{N-1} - 2u_N + b &= h^2 f_N \end{aligned} \tag{2.71}$$

En este caso, la solución al BVP está dada por U en el sistema 2.72

$$WU = F \tag{2.72}$$

o bien en forma explicita por 2.73

$$\begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & -1 & 2 & -1 & 0 \\ \vdots & \ddots & \ddots & -1 & 2 & -1 \\ 0 & \cdots & 0 & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{pmatrix} = \begin{pmatrix} h^2 f_1 - a \\ h^2 f_2 \\ \vdots \\ h^2 f_{N-2} \\ h^2 f_{N-1} \\ h^2 f_N - b \end{pmatrix}. \tag{2.73}$$

La matriz W es no singular, ya que sus eigenvalores están dados por 2.74 los cuales nunca son nulos y su número de condición está dado por la expresión 2.75 (para más detalles consulte Allaire [1] página 87),

$$\lambda_i = 4sin^2\left(\frac{i\pi}{2}\frac{1}{N+1}\right) \tag{2.74}$$

$$\kappa_2(W) = \frac{sin^2(\frac{\pi}{2}\frac{N}{N+1})}{sin^2(\frac{\pi}{2}\frac{1}{N+1})} \approx \frac{4(N+1)^2}{\pi^2} \tag{2.75}$$

en este caso el sistema se puede resolver con TDMA pues claramente es diagonal dominante irreducible, alcanzando una complejidad $\mathcal{O}(N)$ mientras que en los métodos iterativos se puede alcanzar una aproximación adecuada con pocas iteraciones, ya que aunque el número de condición crece conforme aumenta N, en una dimensión se tiene una excelente aproximación para $N = 10^3$.

2.6. El método SOR

Como ya se mencionó antes uno de los métodos iterativos más usados en diferencias finitas es el conocido como SOR y el cual proporciona una convergencia adecuada con un número menor de iteraciones en comparación con otros métodos. Para ver como funciona SOR se empezará definiendo algunos conceptos previos.

Definición 2.6.1. Sea A una matriz no singular y sean M y N dos matrices tales que M es no singular y A se puede descomponer como 2.76

$$A = M - N \tag{2.76}$$

entonces, un método iterativo basado en descomposición es aquel dado por la sucesión 2.77

$$x_{k+1} = M^{-1}Nx_k + M^{-1}b \tag{2.77}$$

donde x_0 es una solución inicial propuesta.

El método conocido como Jacobi se basa en descomponer la matriz A con $M = D$ y $N = D-A$, mientras que Gauss-Seidel usa la descomposición de la forma $M = D - E$ y $N = F$, donde D, E y F son la diagonal principal, la parte triangular inferior y la parte triangular superior respectivamente. En el caso de SOR se necesita de manera adicional un parámetro conocido como parámetro de relajación θ, el cual define el siguiente método iterativo.

Definición 2.6.2. Sea $\omega \in \mathbb{R}^+$. El método iterativo conocido como SOR está dado por la descomposición 2.78

$$M = \frac{D}{\theta} - E, \quad N = \frac{1-\theta}{\theta}D + F \tag{2.78}$$

Claramente no es conveniente la aplicación directa de la expresión 2.77 para poder calcular x_k pues se necesita el cálculo de la inversa de M, el cual es costoso en términos de operaciones. Una manera equivalente de expresar la iteración dada por SOR es con el algoritmo 2,

Algoritmo 2 Método SOR

Entrada: x_0, θ, k_{max}, tol.

1: $k := 0$

Salida: Solución aproximada al sistema $Ax = b$.

2: **mientras** $k < k_{max}$ y $error > tol$ **hacer**

3: $x_i^{(k+1)} = (1-\theta)x_i^{(k)} + \frac{\theta}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij}x_j^{(k+1)} - \sum_{j>i} a_{ij}x_j^{(k)}\right)$, (Paso de SOR)

4: $error = \|x_{k+1} - x_k\|$

5: $k = k + 1$

6: **fin mientras**

donde la convergencia se puede asegurar con el siguiente teorema.

Teorema 2.6.1. *Sea A una matriz simétrica definida positiva. Entonces el método SOR converge para $\theta \in (0, 2)$.*

Demostración. Para ver detalles de la prueba, véase Teorema 8.2.1 de Allaire [1] y Teorema 4.10 de Saad [37].

□

Ejemplo 2. El sistema de ecuaciones 2.73, converge con SOR pues claramente es simétrica y por la ecuación 2.74 es definida positiva. A pesar de que en más dimensiones la matriz global W de este ejemplo sigue siendo simétrica definida positiva y diagonal dominante, ésta ya no es tridiagonal, si no tridiagonal por bloques y en estos casos se debe usar una generalización del TDMA por bloques. Es aquí cuando los métodos iterativos como SOR resultan más convenientes.

Capítulo 3

Funciones de Base Radial

3.1. Interpolación con RBFs

Definición 3.1.1. Una función de base radial ϕ es un campo escalar $\phi : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}$ tal que el mapeo es de la forma $x \mapsto \phi(\|x\|)$, donde $\|\cdot\|$ es la norma euclidiana, en otras palabras las RBFs sólo dependen de la distancia (radio) de sus elementos.

Sea $\{f_k\}_{k=1}^{k=N}$ un conjunto de valores reales de la función f generados por el conjunto de nodos aleatorios $\chi = \{x_k\}_{k=1}^{k=N}$ en $\mathbb{R}^n$, una función interpoladora se define por la expresión 3.1.

$$s(x) = \sum_{k=1}^{N} \lambda_k \phi\left(\|x - x_k\|\right) \tag{3.1}$$

donde λ_k son coeficientes de variable real.

Si definimos $\phi_{ij} = \phi\left(\|x_i - x_j\|\right)$ para $i, j = 1, 2, \ldots, N$, entonces los coeficientes λ_k pueden ser obtenidos resolviendo el sistema lineal 3.2,

$$\begin{pmatrix} \phi_{11} & \cdots & \phi_{1N} \\ \vdots & \ddots & \vdots \\ \phi_{N1} & \cdots & \phi_{NN} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_N \end{pmatrix} = \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix}. \tag{3.2}$$

donde podemos escribirlo de manera compacta como 3.3, a la matriz Φ se le conoce como matriz de Gram.

$$\Phi\Lambda = \Gamma \tag{3.3}$$

Los tipos más importantes de RBFs también llamados kernels pueden ser de diversas categorías, dos de las más respresentativas son: la de las RBFs Suaves a Trozos y la de las RBFs Infinitamente Suaves, las cuales se pueden observar en la TABLA 1. Algo interesante acerca de los kernels es que algunos dependen de un parámetro de forma c y cuya correcta elección es un tema que ha sido extensamente estudiado y sigue siendo un problema abierto, algunas propuestas han sido reportadas por ejemplo en [17], [18] y [36].

Tabla 3.1.1: RBFs-Kernels

RBF	**Kernel**
Suaves a trozos	
Splines Poliharmónicos (PHS)	$(-1)^{m+1}r^{2m}log(r)$
Splines Suaves (SS)	$(-1)^{m+1}r^{2m+1}$
Infinitamente suaves	
Multicuádrico (MQ)	$\phi(r,c) = (r^2+c^2)^{1/2}$
Multicuádrico Inverso (IMQ)	$\phi(r,c) = (r^2+c^2)^{-1/2}$
Gaussiano (GA)	$\phi(r,c) = exp(-(rc)^2)$

Para que el problema de interpolación con RBFs quede bien planteado, es necesario probar que la matriz de Gram asociada a un kernel ϕ es no singular, sin embargo probar dicha afirmación resulta no ser sencillo en todos los casos. Dado que para este trabajo sólo se utilizó el kernel IMQ, la prueba se realizará en la siguiente sección sólo para dicho kernel.

3.2. Matrices de Gram estrictamente definidas positivas

El objetivo de esta sección es probar que la matriz Φ asociada a IMQ es estrictamente definida positiva, para ello, es necesario introducir algunos conceptos previos sobre funciones definidas positivas.

Definición 3.2.1. Sea $\phi : \mathbb{R}^n \to \mathbb{C}$ una función continua, decimos que es definida positiva si se cumple la desigualdad 3.4,

$$\sum_{k=1}^{N}\sum_{l=1}^{N} y_k\overline{y_l}\phi\left(x_k - x_l\right) \geq 0 \tag{3.4}$$

para cualquier conjunto $\{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^n$ de puntos distintos y para cualquier $Y =$

$(y_1, y_2, \cdots, y_N)^T \in \mathbb{C}^n$. Decimos que es estrictamente definida positiva si la igualdad 3.4 se cumple sólo para $Y = 0$.

Observemos que en el caso de un kernel radial si este es una función estrictamente definida positiva, entonces, esto quiere decir que su matriz de Gram es estrictamente definida positiva. Sin embargo, como veremos más adelante no todos los kerneles radiales son estrictamente definidos positivos, en estos caso es necesario generalizar el concepto tal y como sigue.

Definición 3.2.2. Sea $\phi : \mathbb{R}^n \to \mathbb{R}$ una función continua y par, decimos que es condicionalemente definida positiva de orden m si se cumple la desigualdad 3.5,

$$\sum_{k=1}^{N}\sum_{l=1}^{N} y_k y_l \phi\left(x_k - x_l\right) \geq 0 \tag{3.5}$$

para cualquier conjunto $\{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^n$ de puntos distintos y para cualquier $Y = (y_1, y_2, \cdots, y_N)^T \in \mathbb{R}^n$ no nulo, que a su vez que satisfaga las condiciones de momento 3.6.

$$\sum_{k=1}^{N} y_k p(x_k) = 0 \tag{3.6}$$

Donde p es cualquier polinomio de grado menor a m, analogamente decimos que que es estrictamente condicionalmente definida positiva si la igualdad 3.5 se cumple sólo para $Y = 0$.

Respecto a los kerneles radiales, tenemos que si son estrictamente condicionalmente definidos positivos, entonces son estrictamente definidos positivos en el subespacio de los vectores $Y \in \mathbb{R}^n$ que satisfacen la igualdad 3.7.

$$\sum_{k=1}^{N} y_k p_l(x_k) = 0, \qquad 1 \leq l \leq Q = dim\mathbb{R}_{m-1}(\mathbb{R}^n) \tag{3.7}$$

Donde los interpolantes son de la forma 3.8

$$s(x) = \sum_{k=1}^{N} \lambda_k \phi\left(\|x - x_k\|\right) + \sum_{l=1}^{Q} \alpha_l p_l(x), \tag{3.8}$$

y el sistema de ecuaciones asociado al sistema está dado por 3.9

$$\begin{pmatrix} \Phi & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \Lambda \\ \Upsilon \end{pmatrix} = \begin{pmatrix} \Gamma \\ 0 \end{pmatrix} \tag{3.9}$$

Nuestra primer afirmación será que el kernel IMQ es estrictamente definido positivo, para ello se empezará probando que dicha afirmación es verdadera para el kernel GA, de donde se parte como primer lema.

Lema 3.2.1. *El kernel GA es estrictamente definido positiva.*

Demostración. La Transformada de Fourier de la función $\phi(x) = e^{-c^2\|x\|^2}$ está dada por la expresión 3.10, (véase Folland [12] página 244).

$$\widehat{\phi}(\omega) = \frac{1}{2^{n/2}c^n} e^{-\|\omega\|^2/(4c^2)} \tag{3.10}$$

y por tanto el kernel GA se puede expresar por medio de la Transformada Inversa de Fourier de 3.10 y la cual está dada por 3.11

$$e^{-c^2\|x\|^2} = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} \frac{1}{2^{n/2}c^n} e^{-\|\omega\|^2/(4c^2)} e^{i\langle x,\omega\rangle} d\omega \tag{3.11}$$

Sea $Y = (y_1, y_2, \cdots, y_N)^T \in \mathbb{R}^N$ no nulo, usando la identidad 3.11 se tiene que la forma cuadrática de su matriz de Gram se puede expresar como 3.12

$$\begin{aligned}
Y^T \Phi Y &= \sum_{k=1}^{N}\sum_{l=1}^{N} y_k y_l \phi\left(\|x_k - x_l\|\right) \\
&= \sum_{k=1}^{N}\sum_{l=1}^{N} y_k y_l e^{-c^2\|x_k - x_l\|^2} \\
&= \sum_{k=1}^{N}\sum_{l=1}^{N} y_k y_l \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} \frac{1}{2^{n/2}c^n} e^{-\|\omega\|^2/(4c^2)} e^{i\langle x_k - x_l,\omega\rangle} d\omega \\
&= \frac{1}{(2c)^n \pi^{n/2}} \int_{\mathbb{R}^n} e^{-\|\omega\|^2/(4c^2)} \left(\sum_{k=1}^{N}\sum_{l=1}^{N} y_k y_l e^{i\langle x_k - x_l,\omega\rangle} \right) d\omega \\
&= \frac{1}{(2c)^n \pi^{n/2}} \int_{\mathbb{R}^n} e^{-\|\omega\|^2/(4c^2)} \left(\sum_{k=1}^{N} y_k e^{i\langle x_k,\omega\rangle} \right) \overline{\left(\sum_{l=1}^{N} y_l e^{i\langle x_l,\omega\rangle} \right)} d\omega \\
&= \frac{1}{(2c)^n \pi^{n/2}} \int_{\mathbb{R}^n} e^{-\|\omega\|^2/(4c^2)} \left\| \sum_{k=1}^{N} y_k e^{i\langle x_k,\omega\rangle} \right\|^2 d\omega \geq 0
\end{aligned} \tag{3.12}$$

Para demostrar que la expresión 3.12 es estrictamente positiva, se tiene que probar que el término $\left\|\sum_{k=1}^{N} y_k e^{i\langle x_k,\omega\rangle}\right\| > 0$, lo cual se hará por reducción al absurdo.

Supongamos que $\left\|\sum_{k=1}^{N} y_k e^{i\langle x_k,\omega\rangle}\right\| = 0$, esto implica que el la suma dentro de la norma es nula, dado que las funciones $\left\{e^{i\langle x_k,\omega\rangle}\right\}_{k=1}^{N}$ son linealmente independientes, entonces, todas las componentes y_k son cero, lo cual es una contradicción pues $Y \neq 0$ por hipótesis.

□

Una alternativa más sofisticada a este lema está dada por el Teorema de Bochner, el cual enunciamos a continuación.

Teorema 3.2.2 (Bochner)**.** *Sea $\phi : C(\mathbb{R}^n) \to \mathbb{R}$, entonces ϕ es una función estrictamente definida positiva si y sólo si es la transformada de Fourier de una función $W \in L_1(\mathbb{R}^n)$ no negativa y acotada, es decir, el kernel radial se puede escribir como 3.13.*

$$\phi(x) = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} W(\omega) e^{i\langle x,\omega\rangle} d\omega. \tag{3.13}$$

Demostración. Puede consultar la prueba en [46] página 70.

□

De aquí podemos observar que el Lema 3.2.1 es un caso particular del Teorema de Bochner, en el que $W = \widehat{\phi}(\omega)$, esto debido a que la transformada de Fourier del kernel GA es acotada, no negativa y está en $L_1(\mathbb{R}^n)$. Para que una función (en este caso un kernel) tenga transformada de Fourier, ésta debe al menos estar en $L_1(\mathbb{R}^n)$, desafortundamente varios kerneles no cumplen con esta propiedad.

En general no toda función tiene transformada Fourier, o bien, ésta no cumple con las hipótesis del Teorema de Bochner porque es condicionalmente positiva definida y no estrictamente positiva definida, en estos casos es posible usar la transformada de Fourier Generalizada, la cual se define sobre los Espacios de Schwarz (vea el Apéndice A). Esta generalización de la transformada de Fourier nos lleva al Teorema de Madych y Nelson (ver Wendland [46] página 103), el cual no veremos en este trabajo, sin embargo generaliza el Teorema de Bochner a funciones condicionalmente definidas positivas que no están en $L_1(\mathbb{R}^n)$.

El enfoque anterior está basado en la teoría de los espacios de Hilbert, otro enfoque, que depende del concepto de monotonicidad es el siguiente.

Definición 3.2.3. Una función $g : [0,\infty) \to \mathbb{R}$ tal que $g \in C[0,\infty) \bigcap C(0,\infty)$ se dice que es completamente monotónica o completamente monótona en $[0,\infty)$ si cumple para $r > 0$ la desigualdad 3.14.

$$(-1)^k g^{(k)}(r) \geq 0, \qquad k = 0, 1, 2, \ldots \tag{3.14}$$

Teorema 3.2.3 (Representación de Bernstein-Widder)**.** *Una función g es completamente monotónica en $[0, \infty)$ si y sólo g está definida por la transformada de Laplace de una medida μ no decreciente en $[0, \infty)$ y tal que $d\mu(t) \geq 0$, dicha transformada de Laplace está dada por la expresión 3.15*

$$g(r) = \int_0^\infty e^{-rt} d\mu(t) \tag{3.15}$$

Demostración. Para ver la prueba consulte Widder [47] página 160.

□

A continuación enunciamos el Teorema de Schoenberg [42], dicho teorema es la clave para el kernel IMQ.

Teorema 3.2.4 (Schoenberg)**.** *Sea g una función completamente monotónica no constante en $[0, \infty)$, entonces el kernel definido por $\phi(r) = g(r^2)$ es estrictamente definido positivo.*

Demostración. Sea $Y = (y_1, y_2, \cdots, y_N)^T \in \mathbb{R}^N$ no nulo, entonces su forma cuadrática está dada por la expresión 3.16

$$Y^T \Phi Y = \sum_{i=1}^N \sum_{j=1}^N y_i y_j g\left(\|x_i - x_j\|^2\right) \tag{3.16}$$

por el Teorema de Representación de Bernstein - Widder, tenemos que existe una medida μ tal que la forma cuadrática está definida por 3.17

$$\sum_{i=1}^N \sum_{j=1}^N y_i y_j g\left(\|x_i - x_j\|^2\right) = \sum_{i=1}^N \sum_{j=1}^N y_i y_j \int_0^\infty e^{-t\|x_i - x_j\|^2} d\mu(t) \tag{3.17}$$

Dado que se trata de sumas finitas, es posible intercambiar los operadores, dando lugar a la integral de la forma cuadrática asociada al kernel Gaussiano, es decir, tenemos la expresión 3.18

$$\sum_{i=1}^N \sum_{j=1}^N y_i y_j g\left(\|x_i - x_j\|^2\right) = \int_0^\infty \sum_{i=1}^N \sum_{j=1}^N y_i y_j e^{-t\|x_i - x_j\|^2} d\mu(t) \tag{3.18}$$

pero por el Lema 3.18, el kernel GA es estrictamente definido positivo lo cual implica que el kernel ϕ es estrictamente definido positivo.

□

Con el Teorema de Schoenberg 3.2.4 se concluye el resultado esperado para el kernel IMQ con el siguiente corolario.

Corolario 3.2.5. *La matriz de Gram asociada al kernel IMQ es estrictamente definida positiva y por tanto no singular.*

Demostración. Sea la función $g(r) = (r+c^2)^{-1/2}$, es fácil probar por inducción que su $k-$ésima derivada está dada por 3.19

$$g^{(k)}(r) = (-1)^k \frac{1 \cdot 3 \cdot 5 \cdots (2k-1)}{2^k} (r + c^2)^{-\frac{2k+1}{2}}, \tag{3.19}$$

y como consecuencia $(-1)^k g^{(k)}(r) \geq 0$.

Dado que $g(r)$ es completamente monotónica y el kernel de IMQ resulta ser $\phi(r) = g(r^2)$, entonces Φ es estrictamente definida positiva.

□

Por último mencionaremos que en el caso de las funciones condicionalmente definidas positivas, el Teorema análogo al de Schoenberg es el de Michelli y el cual enunciamos a continuación.

Teorema 3.2.6 (Michelli)**.** *Sea $g \in C[0,\infty) \cap C^\infty[0,\infty)$, y sea la función $\phi(r) = g(r^2)$, entonces ϕ es condicionalmente definida positiva de orden m si se cumple que $(-1)^m g^{(m)}(r)$ es completamente monotónica.*

Demostración. Consultar Wendland [46] página 113. □

3.3. Error de aproximación en RBFs

Cuando se desea hacer una interpolación o una aproximación, el error está dado en términos de parámetros que relacionan el espacio entre los nodos o bien el número de nodos, en dichos casos generalmente es necesario usar nodos equiespaciados en una malla cartesiana. Dado que el conjunto χ de nodos no necesariamente debe estar en una malla sino que su distribución en

general es aleatoria, al usar RBFs, la forma de medir el error será con el parámetro conocido como distancia de llenado (*fill distance*).

Definición 3.3.1. Llamaremos distancia de llenado al parámetro dado por la expresión 3.20

$$h_{\chi,\Omega} = \sup_{x\in\Omega} \min_{x_j\in\chi} \|x - x_j\|_2 \tag{3.20}$$

como iterpretación geométrica se tiene que es el radio de la máxima bola que no contiene ningún nodo de χ.

Definición 3.3.2. Sea f en un espacio de funciones $\mathcal{F}$ subespacio de $L_p(\Omega)$, se dice que una función de interpolación o de aproximación s tiene convergencia L_p de orden k, si existe una constante $C > 0$ tal que para todo $f \in \mathcal{F}$ se tiene la estimación 3.21.

$$\|f - s\|_{L_p(\Omega)} \leq C h^k_{\chi,\Omega} \|f\|_{\mathcal{F}} \tag{3.21}$$

Recordemos que las normas en L_p con $p \in [1, \infty)$ están dadas por 3.22,

$$\|f\|_{L_p(\Omega)} = \left(\int_\Omega |f|^p dx \right)^{1/p}, \qquad , 1 \leq p < \infty \tag{3.22}$$

y para $p = \infty$ tenemos 3.23

$$\|f\|_{L_\infty(\Omega)} := ess \sup_{x\in\Omega} |f| \tag{3.23}$$

es decir, el supremo de todas las cotas superiores para $|f|$ en Ω excepto en un conjunto con medida de Lebesgue cero. Algunos ejemplos comunes de espacios $\mathcal{F}$ son los espacios $L_p(\Omega)$ y sus subespacios como los de Sobolev $W^{k,p}(\Omega)$, entre otros.

Ejemplo 3. La interpolación lineal a pedazos para una dimensión en $\Omega = [a, b]$ tiene convergencia en L_2 de orden 2, es decir se cumple 3.24 (véase Atkinson y Han [2] sección 3.1.3).

$$\|f - s\|_{L_2(\Omega)} \leq C h^2_{\chi,\Omega} \|f''\|_{L_2(\Omega)} \tag{3.24}$$

A pesar de que en principio podría ser una buena estimación el tener una convergencia de este tipo, con orden k muy alto y distancia de llenado muy pequeña, algunos kernels de RBFs superan dicha convergencia lo cual supone superioridad sobre otros métodos, dicha convergencia se conoce como espectral y se define a continuación.

Definición 3.3.3. Se dice que la función de aproximación s tiene convergencia L_∞ espectral, si existe una constante $C > 0$ y una constante $0 < \lambda < 1$ tal que para todo $f \in \mathcal{F}$ se tiene la cota 3.25.

$$\|f - s\|_{L_\infty(\Omega)} \leq C\lambda^{-1/h^k_{\chi,\Omega}} \|f\|_{\mathcal{F}} \tag{3.25}$$

En ocasiones la convergencia espectral se puede escribir en términos de la función exponencial como $\|f - s\|_{L_p(\Omega)} \leq Ce^{-\widehat{C}/h^k_{\chi,\Omega}} \|f\|_{\mathcal{F}}$, donde C y $\widehat{C}$ son constantes positivas.

Respecto a la teoría de las RBFs, cada kernel ϕ tiene su propio espacio $\mathcal{F}$ de funciones llamado **espacio Hilbertiano nativo** y el cual se denota como $\mathcal{N}_\phi(\mathbb{R}^n)$, en consecuencia la cota de error en cada kernel depende de su espacio nativo, es decir,

$$\|f - s\|_{L_\infty(\Omega)} \leq CF(h_{\chi,\mathbb{R}^n}) \|f\|_{\mathcal{N}_\phi} \tag{3.26}$$

sin embargo, este punto excede los límites de este trabajo. Pese a que el estudio de los espacios nativos no lo trataremos en este trabajo, señalamos que $\mathcal{N}_\phi$, se deriva del teorema de Madych y Nelson y está definido como el espacio $L_2(\mathbb{R}^n, \mu)$ donde μ es la transformada generalizada de Fourier del kernel ϕ, (para más detalles consulte [46]).

La tabla 3.3.1 contiene las cotas de error para algunos de los kerneles ya mencionados, para ver la discusión completa sobre los espacios nativos y las cotas de error $F(h_{\chi,\Omega})$, véase Wendland [46] y Fasshauer [10] sección 15.1.1. De esta tabla queda claro que al haber varios kernels con convergencia espectral, el uso de RBFs es más conveniente que otros métodos de interpolación.

3.4. Principio de Incertidumbre

La relación que existe entre el error de interpolación y el número de condición fue dada por Schaback en [41], para ello estableció una cota inferior para el radio espectral de Φ^{-1} dicha cota está en función de un parámetro conocido como distancia mínima de separación, el cual está dado por la definición 3.4.1,

Tabla 3.3.1: RBFs-Errores

RBF	$F(h_{\chi,\Omega})$
Suaves a trozos	
Splines Poliharmónicos (PHS)	$h_{\chi,\Omega}^{2m}$
Splines Suaves (SS)	$h_{\chi,\Omega}^{2m+1}$
Infinitamente suaves	
Multicuádrico (MQ)	$e^{-\widehat{C}/h_{\chi,\Omega}^{k}}$
Multicuádrico Inverso (IMQ)	$e^{-\widehat{C}/h_{\chi,\Omega}^{k}}$
Gaussiano (GA)	$e^{-\widehat{C}\lvert log(h_{\chi,\Omega})\rvert/h_{\chi,\Omega}^{k}}$

Definición 3.4.1. Llamaremos distancia mínima de separación al parámetro q dado por

$$q = \frac{1}{2} \min_{i \neq j} \|x_i - x_j\|_2 \tag{3.27}$$

con dicho parámetro tenemos que la cota inferior para el kernel IMQ viene dada por la proposición 3.4.1

Proposición 3.4.1. *El radio espectral de la inversa de la matriz de Gram, está acotado de la siguiente manera*

$$\frac{1}{\varrho(\Phi^{-1})} \geq \frac{exp(-12.76nc/q)}{q} \tag{3.28}$$

Demostración. Para ver la prueba puede consultar Schaback [41] página o Buhmann [4] página 146. □

Por otro lado, es posible dar una cota superior para el radio espectral de Φ a partir de la equivalencia de las normas matriciales, dicha cota está dada por la proposición 3.4.2

.

Proposición 3.4.2. *El radio espectral de alguna matriz A se puede acotar por*

$$\varrho(A) \leq N \|A\|_\infty \tag{3.29}$$

Demostración. Usando el hecho de que $\|A\|_2 = \varrho(A)$ y que se cumplen las estimaciones $\|A\|_2 \leq N^{1/2} \|A\|_1$ y $\|A\|_1 \leq N^{1/2} \|A\|_\infty$ (véase Proposition 3.1.4 de Allaire [1]), se tiene la cota deseada. □

Como consecuencia al aplicar la cota dada por la proposición 3.4.2 a la matriz de Gram, tenemos que la cota superior para el radio espectral de Φ no es siginificativa en el número de condición, por lo que la estabilidad está mayotitariamente dada por el inverso de la expresión 3.4.1. Debido a que $\varrho(\Phi^{-1})$ está acotado por una exponencial con argumento positivo, para tener un buen condicionamiento es necesario que q sea muy grande, lo cual implica que $h_{\chi,\Omega}$ debe ser muy grande y por tanto el error de aproximación es también grande. Al mismo tiempo tenemos que si deseamos un error bastante pequeño es necesario que $h_{\chi,\Omega}$ sea pequeño y como consecuencia q es pequeño, lo cual implica un mal condicionamiento. Dado que no se puede tener un buen condicionamiento y un error pequeño en la aproximación por RBFs de manera simultanea, en análogía a la mecánica cuántica con la relación que existe entre velocidad y posición se le llamó a este hecho **principio de incertidumbre de Schaback**.

3.5. RBFs en la solución de PDEs

Una aplicación típica de las RBFs es la solución de PDEs tanto dependientes como no dependientes del tiempo. La primer propuesta fue hecha por Kansa en [28] y en [29], en dichos trabajos se aproximaron derivadas espaciales y se resolvieron ecuaciones diferenciales parciales de tipo elíptico y parabólico. En el caso de las no dependientes del tiempo, el problema de valores en la frontera por resolver es de la forma 3.30

$$\begin{aligned} \mathfrak{L}u &= g(x) \quad x \in \Omega \\ \mathfrak{B}u &= h(x) \quad x \in \partial\Omega \end{aligned} \tag{3.30}$$

donde $\mathfrak{L}$ y $\mathfrak{B}$ son operadores diferenciales, los cuales defienen la PDE para el interior y la frontera respectivamente.

La discretización en términos de RBFs, requiere introducir un conjunto de nodos interiores $\{x_i\}_{i=1}^{i=N_I}$, los cuales pueden estar distribuidos de forma no uniforme (incluso aleatoriamente) con sus respectivos valores para la función $\{g_i\}_{i=1}^{i=N_I}$ y otro conjunto de nodos frontera $\{x_k\}_{k=1+N_I}^{k=N}$ con los valores $\{h_k\}_{k=1+N_I}^{k=N}$, la idea es similar a la del interpolador ya que la solución es propuesta como combinación lineal de los kernels como en 3.1, entonces aplicar los operadores diferenciales y evaluar en los nodos de colocación para encontrar los coeficientes de la combinación lineal (también llamada ansatz radial), por esa razón $\mathfrak{L}$ y $\mathfrak{B}$ deben ser operadores lineales.

La solución aproximada al problema 3.30 está forzada por las ecuaciones 3.31 y 3.32,

$$\sum_{k=1}^{N} \lambda_k \mathfrak{L}\phi_{ik} = g_i, \qquad i = 1, \ldots, N_I \tag{3.31}$$

$$\sum_{k=1}^{N} \lambda_k \mathfrak{B}\phi_{ik} = h_i, \qquad i = N_I + 1, \ldots, N \tag{3.32}$$

el correspondiente sistema lineal es dado por 3.33 y éste es comunmente conocido como Colocación Asimétrica de Kansa (KAC).

$$\left(\begin{array}{ccc|ccc} \mathfrak{L}\phi_{11} & \cdots & \mathfrak{L}\phi_{1N_I} & \mathfrak{L}\phi_{1(N_I+1)} & \cdots & \mathfrak{L}\phi_{1N} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{L}\phi_{N_I1} & \cdots & \mathfrak{L}\phi_{N_IN_I} & \mathfrak{L}\phi_{N_I(N_I+1)} & \cdots & \mathfrak{L}\phi_{N_IN} \\ \hline \mathfrak{B}\phi_{(N_I+1)1} & \cdots & \mathfrak{B}\phi_{(N_I+1)N_I} & \mathfrak{B}\phi_{(N_I+1)(N_I+1)} & \cdots & \mathfrak{B}\phi_{(N_I+1)N} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{B}\phi_{N1} & \cdots & \mathfrak{B}\phi_{NN_I} & \mathfrak{B}\phi_{N(N_I+1)} & \cdots & \mathfrak{B}\phi_{NN} \end{array}\right) \left(\begin{array}{c} \lambda_1 \\ \vdots \\ \lambda_{N_I} \\ \hline \lambda_{N_I+1} \\ \vdots \\ \lambda_N \end{array}\right) = \left(\begin{array}{c} g_1 \\ \vdots \\ g_{N_I} \\ \hline h_{N_I+1} \\ \vdots \\ h_N \end{array}\right). \tag{3.33}$$

El problema principal de la KAC es que no hay una prueba de la invertibilidad de la matriz involucrada, sin embargo diversos experimentos han probado resultados exitosos en la solución de problemas de valores en la frontera, vea por ejemplo [28], [29], [30] y [40]. Por otro lado, el problema del mal condicionamiento se extiende a la KAC por lo que algunas técnicas para mejorar este problema han sido desarrrolladas, como es el caso del precondicionamiento proximal dado por Brown et al. [3] y métodos de descomposición de dominio (DDM) (véase por ejemplo Gónzalez et al. [24]). Una vez el sistema es resuelto, la solución es escrita como 3.1 y donde dicha ecuación resuelve el problema para todo nodo interior en el dominio, incluso si ese nodo no fue usado en 3.33.

Tiempo después surge lo que se conoce como la Colocación Simétrica de Kansa (KSC), cuya ventaja principal sobre la KAC es que en dicho esquema la matriz del sistema lineal es simétrica y además es posible probar su invertibilidad.

3.6. Funciones de Base Radial - Diferencias Finitas (RBF-FD)

Una propuesta para solucionar el problema del mal condicionamiento tanto en la KAC como en la KSC fue reportada por [45] y en la cual se combina la idea de usar RBFs para calcular los pesos como en FD, a este método se le conoce de manera abreviada como (RBF-FD), esta idea

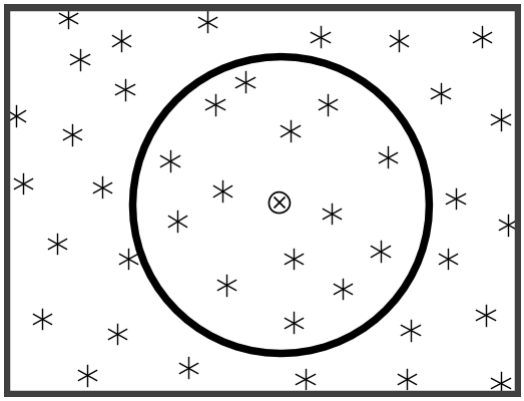

Figura 3.6.1: Distribución de los nodos alrededor de x_c

.

fue adapatada por Martin y Fornberg [34] para ser usada en problemas de exploración sísmica. Así mismo existe una versión muy parecida a RBF-FD propuesta por Shu et al. [43] y la cual es conocida como Cuadratura Diferencial con RBFs (DQ-RBF), ésta consiste en aproximar los pesos usando el kernel MQ.

Con el fin de explicar como funciona RBF-FD se selecciona un conjunto de nodos $\{x_i\}_{i=1}^{i=N_I}$ en el interior del dominio Ω, la idea es aproximar el operador diferencial $\mathfrak{L}$ evaluado en el nodo central x_c a través de pesos con algunos nodos cercanos $I = \{x_{\sigma_k}\}_{k=1}^{k=N_c}$ como en la Figura 3.6.1. En analogía al método de los coeficientes indeterminados [34], los pesos son calculados con el sistema lineal 2.51 pero en vez de usar polinomios se usan kernels de RBF con matrices locales de Gram como en el sistema lineal 3.34,

$$\begin{pmatrix} \phi_{\sigma_1\sigma_1} & \phi_{\sigma_1\sigma_2} & \cdots & \phi_{\sigma_1\sigma_{N_c}} \\ \phi_{\sigma_2\sigma_1} & \phi_{\sigma_2\sigma_2} & \cdots & \phi_{\sigma_2\sigma_{N_c}} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{\sigma_{N_c}\sigma_1} & \phi_{\sigma_{N_c}\sigma_2} & \cdots & \phi_{\sigma_{N_c}\sigma_{N_c}} \end{pmatrix} \begin{pmatrix} \gamma_{\sigma_1} \\ \gamma_{\sigma_2} \\ \vdots \\ \gamma_{\sigma_{N_c}} \end{pmatrix} = \begin{pmatrix} \mathfrak{L}\phi_{\sigma_1 c}|_{x_c} \\ \mathfrak{L}\phi_{\sigma_2 c}|_{x_c} \\ \vdots \\ \mathfrak{L}\phi_{\sigma_{N_c} c}|_{x_c} \end{pmatrix} \tag{3.34}$$

el cual puede ser abreviado como en la ecuación 3.35. Este sistema, como hemos visto es invertible para el kernel IMQ.

$$\Phi_c \Gamma = (\mathfrak{L}\Phi)_c \tag{3.35}$$

Una vez calculados todos los pesos, es posible construir la matriz global de la discretización de $\mathfrak{L}$ dando lugar a la aproximación 3.36.

$$\mathfrak{L}U \approx W_\phi U \tag{3.36}$$

Para el caso en que se resuelva un problema como el dado por el sistema 2.72 a aquellos nodos que se encuentran en la frontera se les asigna un único peso en el mismo nodo central y este tiene como valor 1, es decir si $x_c \in \overline{\partial\Omega}$, entonces el único peso diferente de cero está dado por $\gamma_c = 1$. Sin embargo, cuando usamos condiciones de frontera de tipo Neumann puede ser complicado implementar las condiciones de frontera de esta manera y por ello aunque se usen nodos distribuidos de forma no uniforme, se deja un borde cartesiano para poder implementarlas con diferencias finitas clásicas.

Diversas ventajas se desprenden al usar RBF-FD sobre otros métodos, por ejemplo: al calcular los pesos con el parámetro de forma adecuado la matriz de Gram tiene un número de condición mucho más bajo que si se usaran todos los nodos vecinos, esto sin duda reduce el error de punto flotante en los pesos asociado al número de condición tal y como se discutió en el capítulo anterior. A pesar de que el número de condición pueda ser mucho más bajo usando la matriz de Gram de manera local, es posible que éste sea muy cercano al máximo aceptado por la precisión doble o cuadruple según sea el caso, por ello no es recomendable usar métodos iterativos, por lo que los métodos directos como la descomposición LU o de Cholesky pueden funcionar de manera eficiente al sólo usar un número pequeño de nodos vecinos.

Por otro lado, la matriz global W_ϕ por construcción es una matriz no densa lo cual la hace relativamente bien condicionada, en este caso si es recomendable usar métodos iterativos como GMRES si se está resolviendo el problema 2.72 o bien métodos directos con almacenamiento disperso por ejemplo la Descomposición LU Gilbert-Peierls [21]. No obstante algunas PDEs dependientes del tiempo también tienen ventajas al tener una matriz global dispersa, éste es el caso del método de lineas, en el que la aplicación de los métodos de Runge-Kutta implica operaciones matriz vector que son mucho más eficientes al estar almacenados los pesos de manera dispersa.

3.7. Mapeo Unitario

Como ya se ha mencionado anteriormente, la elección del parámetro de forma c es un problema que hasta la fecha sigue siendo debatido, por lo que la elección de este depende de varios factores como lo es el kernel seleccionado, el tamaño de h y la distribución de los nodos.

El trabajo reportado por Shu et al [43] usa DQ-RBF sobre diferentes PDEs de prueba, dicho método es muy similar al ya mencionado RBF-FD y consiste en calcular los pesos con un conjunto de nodos dispersos alrededor de un nodo central usando el kernel MQ como en la figura 3.6.1. El inconveniente en este caso era que para cada vecindad de nodos alrededor de uno central se necesitaría elegir un parámetro óptimo, Shu resolvió dicho problema y uniformizó dicho

parámetro haciendo un mapeo como el definidio a continuación.

Definición 3.7.1. Sea I el conjunto de nodos vecinos alrededor del nodo central x_c y sea $x \in I$, llamaremos mapeo unitario a la transformación dada por 3.37

$$\overline{x} = \frac{x}{D} \tag{3.37}$$

donde

$$D = 2 \max_{x_{\sigma_k} \in I} \|x_c - x_{\sigma_k}\| \tag{3.38}$$

Con dicho mapeo se puede asegurar que los nodos están encerrados en una bola cerrada de diámetro unitario, en este caso las derivadas se pueden calcular usando la regla de la cadena, es decir, dada una función $f : \Omega \subseteq \mathbb{R}^n \to \mathbb{R}$ y dado $y = (y_1, y_2, \cdots, y_n) \in \Omega$, las derivadas parciales están dadas por 3.39

$$\frac{\partial f}{\partial y_i} = \frac{\partial f}{\partial \overline{y_i}} \frac{\partial \overline{y_i}}{\partial y_i} = \frac{1}{D} \frac{\partial f}{\partial \overline{y_i}}, \tag{3.39}$$

si suponemos que los pesos están calculados con DQ-RBF, las derivadas parciales con el mapeo unitario estarían dadas por 3.40,

$$\frac{\partial f}{\partial \overline{y_i}} \approx \sum_{k=1}^{k=N_c} \overline{\omega_k} f_k \tag{3.40}$$

usando la expresión 3.39 tendríamos que las derivadas parciales en nuestro espacio de interés están dadas por la expresión 3.41

$$\frac{\partial f}{\partial y_i} \approx \frac{1}{D} \sum_{k=1}^{k=N_c} \overline{\omega_k} f_k \tag{3.41}$$

de manera análoga es posible calcular los pesos para derivadas de órdenes más altos.

En el trabajo de Shu et al. [43] se resuelve la ecuación de Poisson variando el parámetro de forma c para diferente número de nodos locales N_c con nodos dispersos, de las conclusiones obtenidas de dicha prueba se tiene que cuando N_c es muy grande, el parámetro de forma sólo se puede escoger en un rango pequeño pues el error aumenta abruptamente, por lo que sugiere a lo más usar 16 nodos locales.

Capítulo 4

Ecuaciones de Navier Stokes (El problema de la cavidad con tapa deslizante)

Las ecuaciones de Navier-Stokes desde su surgimiento han sido ampliamente estudiadas desde diferentes puntos de vista, en los que se encuentran el aspecto físico, el matemático desde la teoría de las ecuaciones diferenciales parciales y el numérico. Su importancia radica en que las aplicaciones son inmensas al modelar la dinámica de fluidos, a continuación se hará la deducción de dichas ecuaciones.

4.1. Fundamentos físicos

Estudiar la dinámica de fluidos puede ser tan complicado como se quiera, por lo que es necesario hacer algunas consideraciones físicas para simplificar el problema, para empezar, vamos a trabajar sólo con Fluidos Newtonianos, los cuales son aquellos en los que la viscosidad μ es constante respecto el tiempo, es decir $\frac{\partial \mu}{\partial t} = 0$. Por otro lado para estudiar un fluido es posible hacerlo desde dos perspectivas, la primera es conocida como Formulación Lagrangiana y consiste en estudiar el movimiento de una parcela de fluido individual en el espacio y tiempo, la segunda es conocida como Formulación Euleriana y estudia el movimiento de las parcelas de fluido a través de un plano fijo en función del tiempo y del espacio, es de ésta última de la cual partimos.

Empezaremos definiendo el vector velocidad como 4.1

$$\overline{q} = (u, v, w), \tag{4.1}$$

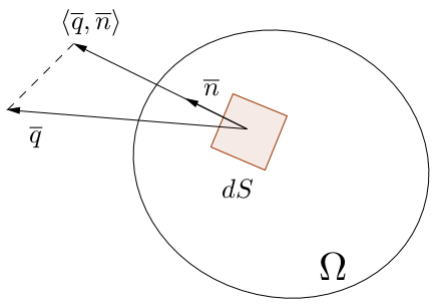

Figura 4.1.1: Volumen de control

.

donde $q_1 := u$, $q_2 := v$ y $q_3 := w$ son las componentes de velocidad en x, y y z respectivamente. Ahora, definamos el volumen de control Ω como el de la Figura 4.1.1, si asumimos que el fluido sobre Ω satisface la ecuación de conservación de masa, entonces para un fluido con densidad ρ se tiene la expresión integral dada por 4.2

$$\frac{\partial}{\partial t}\int_{\Omega} \rho dV + \int_{\partial\Omega} \rho \langle \overline{q}, \overline{n} \rangle ds = 0, \tag{4.2}$$

la cual indica que el cambio en la masa del fluido respecto al tiempo es igual a la masa transportada a través de la superficie.

Sabemos que el teorema de la divergencia está dado por la expresión 4.3

$$\int_{\Omega} \nabla \cdot \overline{q} dV = \int_{\partial\Omega} \langle \overline{q}, \overline{n} \rangle ds, \tag{4.3}$$

al aplicar dicho teorema sobre el término vectorial $\rho\overline{q}$, tenemos la expresión

$$\int_{\Omega} \nabla \cdot \rho\overline{q} dV = \int_{\partial\Omega} \rho \langle \overline{q}, \overline{n} \rangle ds, \tag{4.4}$$

la cual a su vez al sustituir en la ecuación de conservación de masa 4.2 nos dá la integral 4.5

$$\int_{\Omega} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot \rho\overline{q} \right) dV = 0. \tag{4.5}$$

Dado que se trata de un volumen de control arbitrario Ω y los integrandos son funciones continuas, llegamos a la ecuación 4.6

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho\overline{q} = 0, \tag{4.6}$$

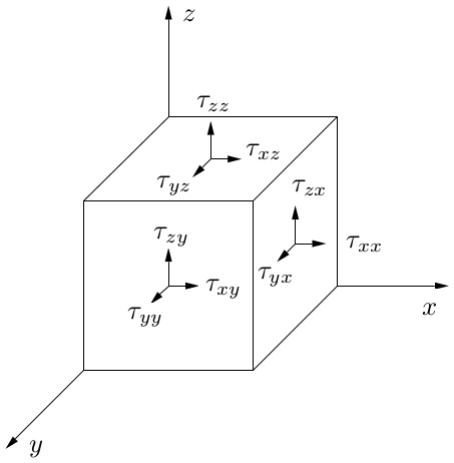

Figura 4.1.2: Tensiones

.

dicha ecuación es conocida como ecuación de continuidad y la cual puede escribirse también como 4.7

$$\frac{\partial \rho}{\partial t} + \overline{q}\nabla\rho + \rho\nabla \cdot \overline{q} = 0. \tag{4.7}$$

Por otro lado, un requisito indispensable es que también deben cumplirse las ecuaciones de conservación de momento, las cuales están dadas por 4.8

$$\frac{\partial m\overline{q}}{\partial t} = \sum \overline{F}, \tag{4.8}$$

donde $\overline{F}$ son las fuerzas involucradas en el sistema. Para simplificar el problema, consideremos un cubo microscópico como el de la figura 4.1.2, sobre dicho cubo interactuan fuerzas de cuerpo y fuerzas de contacto, en el caso de las fuerzas de cuerpo tenemos la fuerza por unidad de masa y la cual denotamos $\overline{f} = (f_1, f_2, f_3)$, mientras que las fuerzas de contacto sobre las paredes se calculan con las tensiones τ_{ij} por unidad de área, lo que nos lleva a la expresión 4.9.

$$\left(\sum \overline{F}\right)_i = \int_\Omega \rho f_i dV + \int_{\partial\Omega} \sum_{j=1}^{3} n_j \tau_{ij} ds \tag{4.9}$$

Al mismo tiempo tenemos que la expresión del lado izquierdo de 4.9 se puede expresar en forma integral para el volumen de control como 4.10

$$\frac{\partial m\overline{q}}{\partial t} = \frac{\partial}{\partial t}\int_\Omega \rho\overline{q} dV + \int_{\partial\Omega} \rho\overline{q}\langle\overline{q}, \overline{n}\rangle ds \tag{4.10}$$

de donde tenemos la expresión 4.11

$$\int_{\Omega} \rho f_i dV + \int_{\partial\Omega} \sum_{j=1}^{3} n_j \tau_{ij} ds = \frac{\partial}{\partial t} \int_{\Omega} \rho q_i dV + \int_{\partial\Omega} \rho q_i \langle \overline{q}, \overline{n} \rangle ds \tag{4.11}$$

que al usar el teorema de la divergencia sobre las integrales de superficie da como resultado la expresión 4.12

$$\int_{\Omega} \left(\frac{\partial}{\partial t} (\rho q_i) + \nabla \cdot (\rho q_i \overline{q}) - \rho f_i - \sum_{j=1}^{3} \frac{\partial}{\partial x_i} \tau_{ij} \right) dV = 0 \tag{4.12}$$

y donde nuevamente al ser arbitrario el volumen de control sobre una función continua se tiene que el integrando es nulo. Dicho integrando al ser desarrollado da como resultado la expresión 4.13

$$\left[\frac{\partial}{\partial t} q_i + \overline{q} \nabla q_i \right] \rho = \rho f_i + \sum_{j=1}^{3} \frac{\partial}{\partial x_i} \tau_{ij} \tag{4.13}$$

que al considerar un fluido newtoniano las componentes de tensión están dadas por 4.14 (véase Katz [31] página 38),

$$\tau_{ij} = \left(-p - \frac{2}{3} \mu \sum_{k=1}^{3} \frac{\partial}{\partial x_k} q_k \right) \delta_{ij} + \mu \left(\frac{\partial q_i}{\partial x_j} + \frac{\partial q_j}{\partial x_i} \right) \tag{4.14}$$

al sustitur en 4.13 finalmente tenemos las ecuaciones de Navier-Stokes 4.15

$$\left[\frac{\partial}{\partial t} q_i + \overline{q} \cdot \nabla q_i \right] \rho = \rho f_i - \frac{\partial}{\partial x_i} \left(p + \frac{2}{3} \mu \nabla \cdot \overline{q} \right) + \sum_{j=1}^{3} \frac{\partial}{\partial x_j} \mu \left(\frac{\partial q_i}{\partial x_j} + \frac{\partial q_j}{\partial x_i} \right) \tag{4.15}$$

Una manera de estandarizar el problema es volverlo adimensional, para ello se pueden definir la transformación, dada por los mapeos 4.16, 4.17 y 4.18

$$x^* = \frac{x}{L}, \qquad y^* = \frac{y}{L}, \qquad z^* = \frac{z}{L} \tag{4.16}$$

$$u^* = \frac{u}{V}, \qquad v^* = \frac{v}{V}, \qquad w^* = \frac{w}{V} \tag{4.17}$$

$$t^* = \frac{t}{T}, \qquad p^* = \frac{p}{p_0}, \qquad f^* = \frac{\overline{f}}{f_0} \tag{4.18}$$

donde:

1. V es un volumen de referencia.
2. L es una longitud de referencia.
3. T es un tiempo de referencia.
4. p_0 es una presión de referencia.
5. f_0 es una aceleración producida por las fuerzas de cuerpo (usualmente la constante gravitatoria terrestre g).

Suponiendo un fluido con viscosidad constante e incompresible, es decir con densidad constante (o equivalentemente $\nabla \cdot \overline{q} = 0$) y usando la regla de la cadena es posible escribir las ecuaciones de Navier-Stokes como 4.19, 4.20 y 4.21,

$$C_T \frac{\partial u^*}{\partial t} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} + w^* \frac{\partial u^*}{\partial z^*} = \frac{1}{F_r^2} f_1^* - E_u \frac{\partial p}{\partial x} + \frac{1}{Re}\left(\frac{\partial^2 u^*}{\partial x^{*2}} + \frac{\partial^2 u^*}{\partial y^{*2}} + \frac{\partial^2 u^*}{\partial z^{*2}}\right) \tag{4.19}$$

$$C_T \frac{\partial v^*}{\partial t} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} + w^* \frac{\partial v^*}{\partial z^*} = \frac{1}{F_r^2} f_2^* - E_u \frac{\partial p}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}} + \frac{\partial^2 v^*}{\partial z^{*2}}\right) \tag{4.20}$$

$$C_T \frac{\partial w^*}{\partial t} + u^* \frac{\partial w^*}{\partial x^*} + v^* \frac{\partial w^*}{\partial y^*} + w^* \frac{\partial w^*}{\partial z^*} = \frac{1}{F_r^2} f_3^* - E_u \frac{\partial p}{\partial z} + \frac{1}{Re}\left(\frac{\partial^2 w^*}{\partial x^{*2}} + \frac{\partial^2 w^*}{\partial y^{*2}} + \frac{\partial^2 w^*}{\partial z^{*2}}\right) \tag{4.21}$$

de donde tenemos las siguientes constantes adimensionales. La primera es $C_T := \frac{L}{TV}$, esta constante controla los términos temporales en el sistema. La segunda constante está definida por $F_r := \frac{V}{\sqrt{Lf_0}}$ y se le conoce como número de **Froude**, el cuadrado de esta constante determina la razón que existe entre las fuerzas de inercia sobre las fuerzas de cuerpo. En tercer lugar está $E_u := \frac{p_0}{\rho V^2}$ conocida como número de **Euler**, dicho número representa la razón entre la presión y las fuerzas de inercia. Finalmente está la constante $Re := \frac{\rho V L}{\mu}$ conocida como número de **Reynolds**, la importancia de éste radica en que puede determinar regimen laminar o turbulencia en cualquier problema de fluidos.

Las suposiciones que haremos de nuestro fluido son las siguientes:

1. El problema es bidimensional o bien puede ser modelado de esta manera.
2. La única fuerza de cuerpo involucrada es la gravitacional ($f_0 = g$) y las fuerzas de inercia son mucho mayores que las fuerzas de cuerpo, es decir, el efecto de la fuerza gravitatoria es mucho menor que el de las fuerzas de cuerpo, de aquí tenemos que el térmnino $\frac{1}{F_r^2} \approx 0$.
3. Las fuerzas de inercia producen un efecto mucho mayor al de la presión en el sistema, por tanto la constante de Euler puede ser despreciable.
4. El tiempo de referencia T no puede ser extremadamente pequeño, ya que si asumimos el tiempo de referencia como el tiempo medio del proceso, entonces este debe ser prolongado o de lo contrario sería casi instantaneo. De las dos suposiciones anteriores tenemos que V es grande en magnitud mientras que L es relativamente pequeño, por lo que de esta última suposición la constante $C_T \approx 0$.

Como consecuencia de esto, el sistema 4.19, 4.20 y 4.21 se reduce a dos ecuaciones que no dependen del tiempo, a esto se le conoce como un problema de flujo estable que sólo depende del parámetro Re. Una vez hecho lo anterior y con el fin de simplificar todavía más el sistema de ecuaciones 4.19, 4.20 y 4.21, es posible introducir algunas variables físicas como son la vorticidad, la cual está definida como $\overline{\xi} = \nabla \times \overline{q}$ y cuyo significado físico es la capacidad de giro del fluido. Dado que se trata de un fluido modelado en dos dimensiones, dicha vorticidad sólo tiene una componente ξ en la dirección z, es decir

$$\overline{\xi} = \left(0, 0, \frac{\partial v^*}{\partial x^*} - \frac{\partial u^*}{\partial y^*}\right). \tag{4.22}$$

Por otro lado tenemos la función de flujo ψ, la cual representa las posibles curvas sobre las cuales una partícula dentro del fluido podría viajar. Por definición las componenetes de velocidad siempre son tangentes a dichas trayectorias, por lo que se define dicha función en forma diferencial con el sistema 4.23

$$u^* = \frac{\partial \psi}{\partial y^*}, \qquad v^* = -\frac{\partial \psi}{\partial x^*}. \tag{4.23}$$

Sustituyendo las expresiones 4.22 y 4.23 en el sistema de Navier-Stokes 4.19, 4.20, 4.21 con las consideraciones ya mencionadas, se tiene finalmente el sistema acoplado dado por las ecuaciones 4.24 y 4.25

$$\nabla^2 \psi = -\xi \tag{4.24}$$

$$\nabla^2 \xi = u^* \frac{\partial \xi}{\partial x^*} + v^* \frac{\partial \xi}{\partial y^*}. \tag{4.25}$$

Finalmente, es necesario decir que a apartir de este momento evitaremos la notación $*$ con fines prácticos (para los detalles referentes a esta sección puede consultar Katz [31] capítulo 2 y Pozrikidis [35] capítulo 6).

4.2. El problema de la cavidad con tapa deslizante

El problema de la cavidad con tapa deslizante ampliamente conocido en inglés como Lid Driven Cavity (LDC), consiste en resolver las ecuaciones de Navier-Stokes dadas por el sistema 4.24 y 4.25, para un flujo estable sobre el dominio rectangular: $\overline{\Omega} = [0,1] \times [0,1]$ y cuyas condiciones de frontera están representadas en la figura 4.2.1,

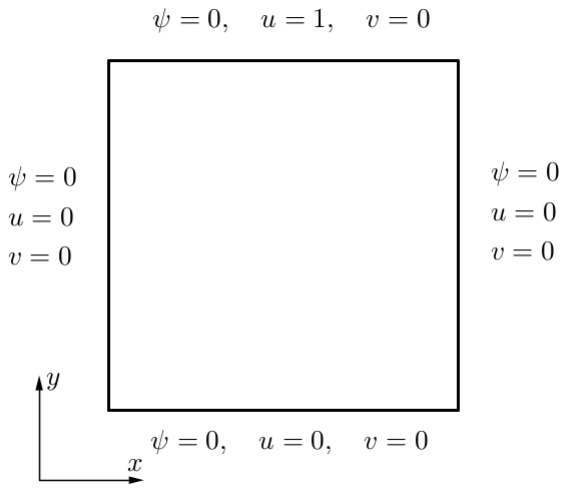

Figura 4.2.1: LDC

.

este problema puede reproducirse experimentalmente en un laboratorio con el dispositivo dado en la figura 4.2.2, en éste se tiene un recipiente en 3 dimensiones con un fluido donde la tapa superior no es más que una banda transportadora que tiene contacto con el fluido y la cual es movida por dos ruedas que giran en sentido antihorario.

El problema ha sido resuelto generalmente usando diferencias finitas sobre una malla cartesiano en diversos trabajos, una dificultad presentada por este problema es que el sistema de ecuaciones es fuertemente acoplado, por lo que es necesario usar algún método numérico que pueda dar una solución iterativa tanto de ψ como de ξ.

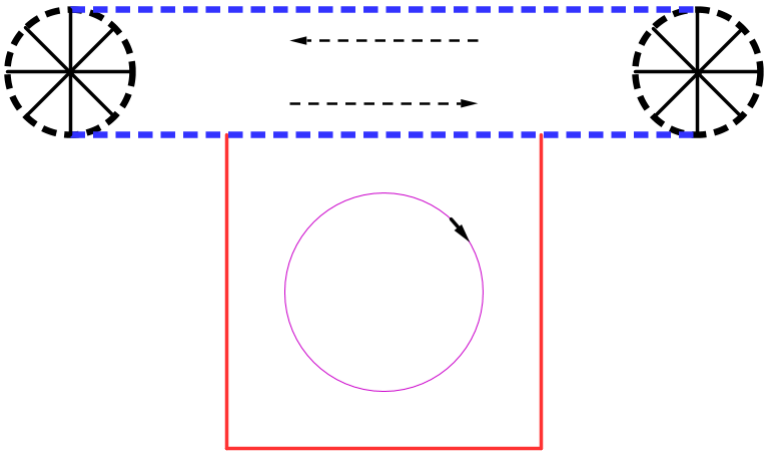

Figura 4.2.2: Dispositivo físico para el LDC

.

En la mayoría de los casos la discretización clásica en diferencias finitas del problema está dada por el Laplaciano a 5 puntos y derivadas centradas para el caso de los operadores $\frac{\partial}{\partial y}$ y $\frac{\partial}{\partial x}$, dando lugar así al esquema numérico 4.26 y 4.27

$$0 = \frac{\psi_{i-1,j} - 2\psi_{i,j} + \psi_{i+1,j}}{\Delta x^2} + \frac{\psi_{i,j-1} - 2\psi_{i,j} + \psi_{i,j+1}}{\Delta y^2} + \xi_{i,j} \tag{4.26}$$

$$\begin{aligned} 0 = & \frac{\xi_{i-1,j} - 2\xi_{i,j} + \xi_{i+1,j}}{\Delta x^2} + \frac{\xi_{i,j-1} - 2\xi_{i,j} + \xi_{i,j+1}}{\Delta y^2} \\ & - Re\frac{\psi_{i,j+1} - \psi_{i,j-1}}{2\Delta y}\frac{\xi_{i+1,j} - \xi_{i-1,j}}{2\Delta x} + Re\frac{\psi_{i+1,j} - \psi_{i-1,j}}{2\Delta x}\frac{\xi_{i,j+1} - \xi_{i,j-1}}{2\Delta y}, \end{aligned} \tag{4.27}$$

mientras que existen diferentes alternativas para el método iterativo. Un trabajo clásico es el presentado por Ghia [20], en el cual el problema se resuelve con un método iterativo conocido como multigrid, por otro lado tenemos el trabajo de Erturk [9] el cual usa la adición de un término llamado pseudoderivada temporal dando lugar así al sistema 4.28 y 4.29

$$\nabla^2\psi + \xi = \frac{\partial\psi}{\partial t} \tag{4.28}$$

$$\nabla^2\xi - u\frac{\partial\xi}{\partial x} - v\frac{\partial\xi}{\partial y} = \frac{\partial\xi}{\partial t}, \tag{4.29}$$

la adición de este término no altera el sistema original para un problema de flujo estable, pues la función de flujo y la vorticidad no dependen del tiempo por lo que sólo se adhiere un cero a cada

ecuación. La forma en la que trabaja este método es proponer las soluciones iniciales ψ^0 y ξ^0 con la cuales la discretización 4.28 y 4.29 transforma el sistema en un problema de valor inicial para ecuaciones diferenciales ordinarias, en este caso Ertuk propone usar un esquema implícito para asegurar estabilidad, sin embargo esta elección implica la solución de los sistemas lineales asociadas al Laplaciano en cada iteración por lo que puede ser muy costoso computacionalmente hablando.

Otra alternativa también usada por Erturk [8] es utilizar SOR como el método iterativo del sistema acoplado, en dicho trabajo Erturk también propone usar la formula de Jensen 4.30 (descrita en Fletcher [11] página 374) para la vorticidad,

$$\xi_0 = \frac{-4\psi_1 + 0.5\psi_2}{\Delta h^2} - \frac{3U}{\Delta h}, \tag{4.30}$$

en esta fórmula ξ_0 es la vorticidad sobre las paredes de la cavidad, ψ_1 es la función de flujo en el nodo adyacente a la pared y ψ_2 es adyacente al nodo ψ_1. El valor de U es la velocidad sobre las paredes de la cavidad, resultando ser $U = 1$ en la parte superior y cero en otro caso, finalmente Δh es Δx o Δy según sea la dirección de los nodos adyacentes.

El algoritmo que describe este proceso es el siguiente

Algoritmo 3 Problema LDC con FD (SOR)

Entrada: ξ^0, ψ^0, u^0, v^0, U, θ, k_{max}, tol.

1: Calcula los pesos para ∇^2, $\frac{\partial}{\partial x}$ y $\frac{\partial}{\partial y}$ con FD.

2: $k := 0$

Salida: Solución aproximada ξ y ψ.

3: **mientras** $k < k_{max}$ y $error > tol$ **hacer**

4: Calcular un paso de SOR en $\nabla^2\psi^{k+1} = -\xi^k$

5: Calcular las velocidades con FD $u^{k+1} = \frac{\partial\psi^{k+1}}{\partial y}$, $v^{k+1} = -\frac{\partial\psi^{k+1}}{\partial x}$

6: Calcular la expresión $\eta^{k+1} = u^{k+1}\frac{\partial\xi^k}{\partial x} + v^{k+1}\frac{\partial\xi^k}{\partial y}$

7: Condiciones de frontera con la fórmula de Jensen

8: Calcular un paso de SOR en $\nabla^2\xi^{k+1} = Re\,\eta^{k+1}$

9: $k = k + 1$

10: $error = \text{máx}\{\frac{\left\|\xi^k - \xi^{k-1}\right\|_2}{\left\|\xi^k\right\|_2}, \frac{\left\|\psi^k - \psi^{k-1}\right\|_2}{\left\|\psi^k\right\|_2}\}$

11: **fin mientras**

El número de Reynolds juega un papel fundamental en la solución del problema, por ejemplo, Koseff y Street [44] probaron de manera experimental que para $Re > 1000$ el LDC no puede ser físicamente modelado de manera bidimensional, es decir, en una cavidad no se puede usar la simplificación de Navier-Stokes en 2 dimensiones, más aún, dicho problema tampoco puede ser considerado como un problema de flujo estable.

En dicho trabajo también se probó de manera experimental que si el número de Reynolds se encuentra en el rango de $6000 \leq Re \leq 8000$, el fluido empieza a dejar de ser laminar y presenta rasgos de turbulencia. Para el caso de $Re = 8000$, Fortin et. al [19] detectaron una bifurcación de Hopf a través del cálculo numérico de eigenvalores, mientras que Koseff [32] probó turbulencia para $Re = 10000$ de manera experimental. Erturk [8] menciona que aunque el fluido no pueda ser modelado físicamente, numéricamente si es posible resolver el problema para $Re > 1000$ de manera bidimensional y como problema de flujo estable, en estos casos se considera que el problema es de un **fluido ficticio** [1], por otro lado la estabilidad en los casos donde Re es muy grande sólo se puede asegurar haciendo las mallas cada vez más finas y por tanto la convergencia puede depender de cada vez más iteraciones.

En el trabajo de Ghia [20] se resolvió el problema para Re = 100, 400, 1000, 3200, 5000, 7500 y 10000 en una malla de 129×129 nodos y generando tablas del perfil horizontal de v y del perfil vertical de u que pasan por el centro de la cavidad, en el caso del trabajo de Erturk [8], este usa una malla de 1025×1025 para Re = 1000, 2500, 5000, 7500, 10000, 12500, 15000, 17500 y 20000 obteniendo resultados favorables y donde grafica algunos perfiles para u y para v.

Por último, un trabajo importante es el de Cruz [6] en el cual se resuelve el LDC con KAC, esto para una malla de 51×51 para Re = 100, 400 y 1000. El kernel que se usó en dicho trabajo fue MQ con el parámetro de forma $c = 1/\sqrt{N}$, así como el precondicionador llamado Approximated Cardinal Basis Functions Preconditioner (ACBF) propuesto por Brown et al. [3] con 50 nodos vecinos. Dicho experimento tuvo resultados favorables, los cuales se pudieron corroborar al comparar los perfiles de u y v con los reportados por Ghia [20].

[1] Un fluido ficticio es aquel cuyo comportamiento puede ser descrito por medio de un modelo matemático (como una ecuación diferencial), sin embargo, existe evidencia en laboratorio de que no es posible físicamente.

Capítulo 5

Metodología

5.1. Selección de pesos

El método iterativo por usar en la solución al LDC será SOR, lo cual como ya se mencionó se debe a que puede ser menos costoso computacionalmente que la adición de pseudoderivadas y adicionalmente ha reportado buenos resultados. En este trabajó se usará un arreglo de nodos distribuidos uniformemente de manera que la matriz global W_ϕ tenga una estructura simétrica, esto con la finalidad de comparar los resultados con los reportados por Ghia [20] y Cruz [6]. Cabe decir que existen algunos trabajos como el de Shu [43] en el cual se puede usar SOR para resolver PDEs con DQ-RBF sobre una distribución de nodos aleatorios.

Una forma de construir W_ϕ de manera simétrica, es usar una configuración de nodos en la que cada nodo local (con excepción del nodo central) tiene un par a la misma distancia del nodo central. De manera simple escogeremos las configuraciones de nodos locales con N_c = 9, 25 y 49 tal y como se muestran en la figura 5.1.1, sin embargo, no es posible usar 25 o 49 nodos locales sobre los puntos que están más a la orilla. Para resolver este problema en el caso que se deseen usar más de 9 nodos locales, es necesario dividir la distribución de los nodos en tres categorías como en la figura 5.1.2, por ejemplo, considere la esquina superior izquierda y supongamos que queremos usar 49 nodos locales, en este caso a los nodos consecuentes a los nodos frontera les corresponden los pesos correspondientes a 9 nodos locales, a los adyacentes a estos últimos les asignamos los pesos de 25 nodos locales y finalmente los que restan se calculan con 49 nodos locales.

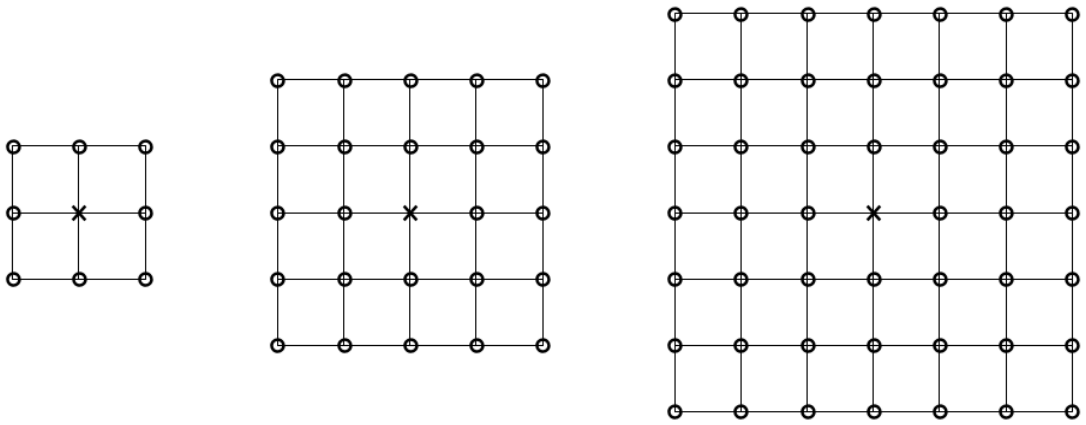

Figura 5.1.1: Nodos locales

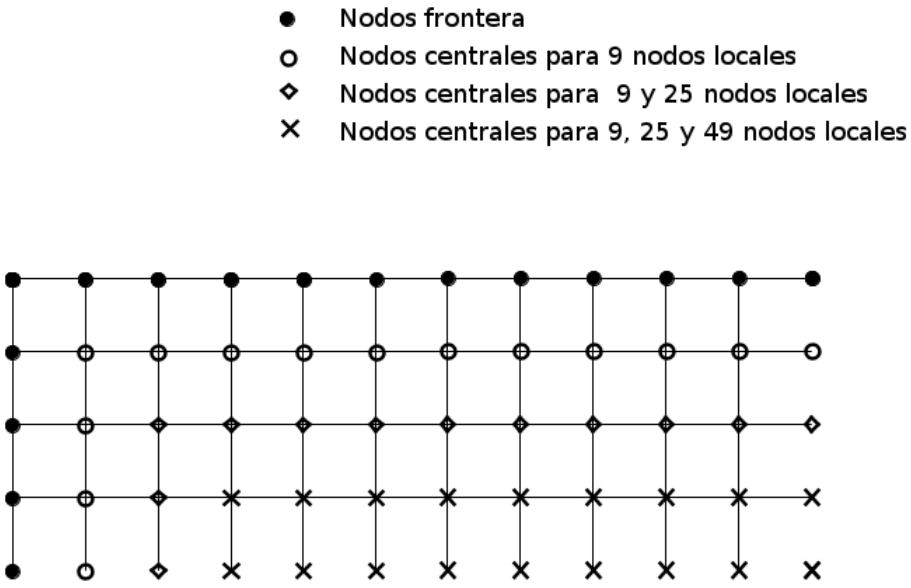

Figura 5.1.2: Categorías de los nodos

Para calcular los pesos en los nodos interiores sólo es necesario hacerlo una vez al igual que en FD clásico para los operadores ∇^2, $\frac{\partial}{\partial x}$ y $\frac{\partial}{\partial y}$, por simplicidad escogemos como nodo central al origen para todos los casos, ya que una traslación es una isometría en $\mathbb{R}^n$ y dado que los kernels son dependientes de la norma, entonces no se altera la estructura de la matriz de Gram. Para resolver el sistema 3.3, se usará la descomposición LU y aprovechando dicha descomposición se calculará la inversa de la matriz de Gram, la cual a su vez servirá para calcular el número de condición $\kappa_\infty(A)$. A pesar de que esta forma de calcular el número de condición puede ser costosa en comparación con otros algoritmos de aproximación (veáse por ejemplo la sección 5.3.4 de Allaire [1]), a lo más éste se calcula 3 veces por cada parámetro de forma para unos cuántos nodos locales, además de que se debe ser muy preciso al delimitar en que casos el número de condición sobrepasa la precisión de la máquina. Una vez obtenidos los pesos locales se pueden construir las respectivas matrices globales de cada operador, las cuales se almacenan en el formato ralo CSR (para ver detalles de esto, puede consultar [37] sección 3.4). Cabe decir que todos los códigos necesarios para este trabajo se realizarán en **Fortran 95** y de los cuales se pueden consultar algunos en el Apéndice B.

La primer gráfica propuesta consiste en gráficar el máximo de los números de condición entre 9, 25 y 49 nodos locales contra el parámetro de forma, esto nos indica los parámetros que son incompatibles con la precisión de la máquina. Estos resultados se muestran en la Figura 6.1.1, la cual se analizará en el próximo capítulo.

5.2. Solución al problema de la cavidad con tapa deslizante usando RBF-FD

Una vez realizadas las gráficas mencionadas, es posible elegir un parámetro de forma para poder usarse en el cálculo de los pesos. Como requisito, el número de condición de dicho parámetro debe ser grande (por el principio de incertidumbre) sin sobrepasar la precisión de la máquina en cada caso. Ya elegido el parámetro de forma adecuada para cada uno de los casos, se calcularán los pesos con RBF-FD y el algoritmo que determinará la solución al problema es el 4.

Algoritmo 4 LDC RBF-FD (SOR)

Entrada: ξ^0, ψ^0, u^0, v^0, U, θ, k_{max}, tol.

1: Calcula los pesos para ∇^2, $\frac{\partial}{\partial x}$ y $\frac{\partial}{\partial y}$ con RBF-FD.

2: $k := 0$

Salida: Solución aproximada ξ y ψ.

3: **mientras** $k < k_{max}$ y $error > tol$ **hacer**

4: Calcular un paso de SOR en $\nabla^2\psi^{k+1} = -\xi^k$

5: Calcular las velocidades con FD $u^{k+1} = \frac{\partial\psi^{k+1}}{\partial y}$, $v^{k+1} = -\frac{\partial\psi^{k+1}}{\partial x}$

6: Calcular la expresión $\eta^{k+1} = u^{k+1}\frac{\partial\xi^k}{\partial x} + v^{k+1}\frac{\partial\xi^k}{\partial y}$

7: Condiciones de frontera con la fórmula de Jensen

8: Calcular un paso de SOR en $\nabla^2\xi^{k+1} = Re\,\eta^{k+1}$

9: $k = k + 1$

10: $error = \text{máx}\{\frac{\left\|\xi^k - \xi^{k-1}\right\|_2}{\left\|\xi^k\right\|_2}, \frac{\left\|\psi^k - \psi^{k-1}\right\|_2}{\left\|\psi^k\right\|_2}\}$

11: **fin mientras**

Este problema será resuelto para una distribución uniforme de nodos de 50×50 y una de 100×100, dicha prueba se realizará para los números de Reynolds $Re = 100$, 400 y 1000 en cada caso una tolerancia de 10^{-10} para el error en SOR y un máximo de iteraciones de 100000. Una vez hecho esto se graficarán los perfiles de velocidad que pasan por el centro de la cavidad, tal y como se muestran en la Figura 5.2.1. Dichos resultados se compararán con los obtenidos por Ghia [20], adicionalmente calcularemos u_{min}, v_{min} y v_{max} en cada prueba, luego se comparará con los resultados obtenidos por de la Cruz en [6].

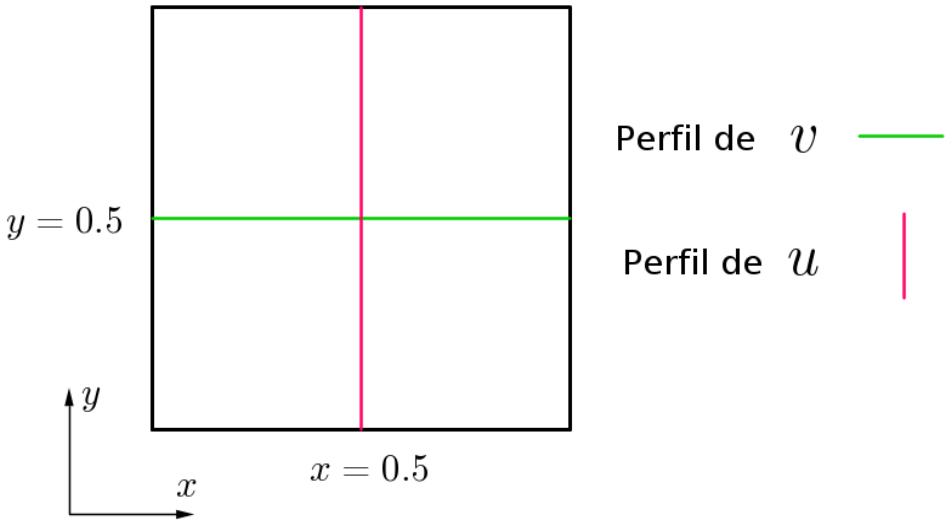

Figura 5.2.1: Perfiles para u y v

.

Capítulo 6

Resultados

6.1. Números de condición

La Figura 6.1.1 contiene los resultados de graficar el parámetro de forma c contra el número de condición en precisión doble, cuádruple, con y sin mapeo unitario. Para el caso de la precisión doble, tenemos que en general los números de condición aumentan considerablemente conforme c aumenta, en este caso el intervalo de c donde crece abruptamente es muy pequeño. Por ejemplo, para 9 nodos locales tanto en arreglos de 50×50 con los de 100×100 tenemos que c crece rápidamente desde 10^{-2} hasta 1 aproximadamente. Como es de esperarse el número de condición crece más abruptamente mientras más nodos se usen en la matriz de Gram, más aún, al aumentar el número total de nodos (al usar 100×100) el parámetro q es más pequeño y por tanto como lo establece el principio de incertidumbre de la sección 2.5, el número de condición crece más que en el caso de 50×50.

En el caso de la precisión doble con mapeo unitario, el intervalo de c en el que el número de condición crece abruptamente es mucho más amplio, pues va aproximadamente desde 10^{-1} hasta casi 10^2 para c. Otro detalle importante es que las gráficas de los números de condición con el mismo número de nodos locales se encuentran sobrepuestas, esto se debe a que al usar el mapeo unitario el parámetro q es el mismo independientemente del número de nodos totales.

Para las gráficas de precisión cuadruple con y sin mapeo, observamos que los números de condición siguen el mismo comportamiento que su análogo en precisión doble, sin embargo, el intervalo de c en el que número de condición crece abruptamente es mucho más amplio, pues en el caso de 9 nodos locales éste va aproximadamente de 10^{-2} hasta 10^1 sin usar mapeo, y al usar el mapeo el intervalo soprepasa el valor de 10^2.

Con los resultados ya mencionados, fue posible elegir el parámetro de forma adecuado para

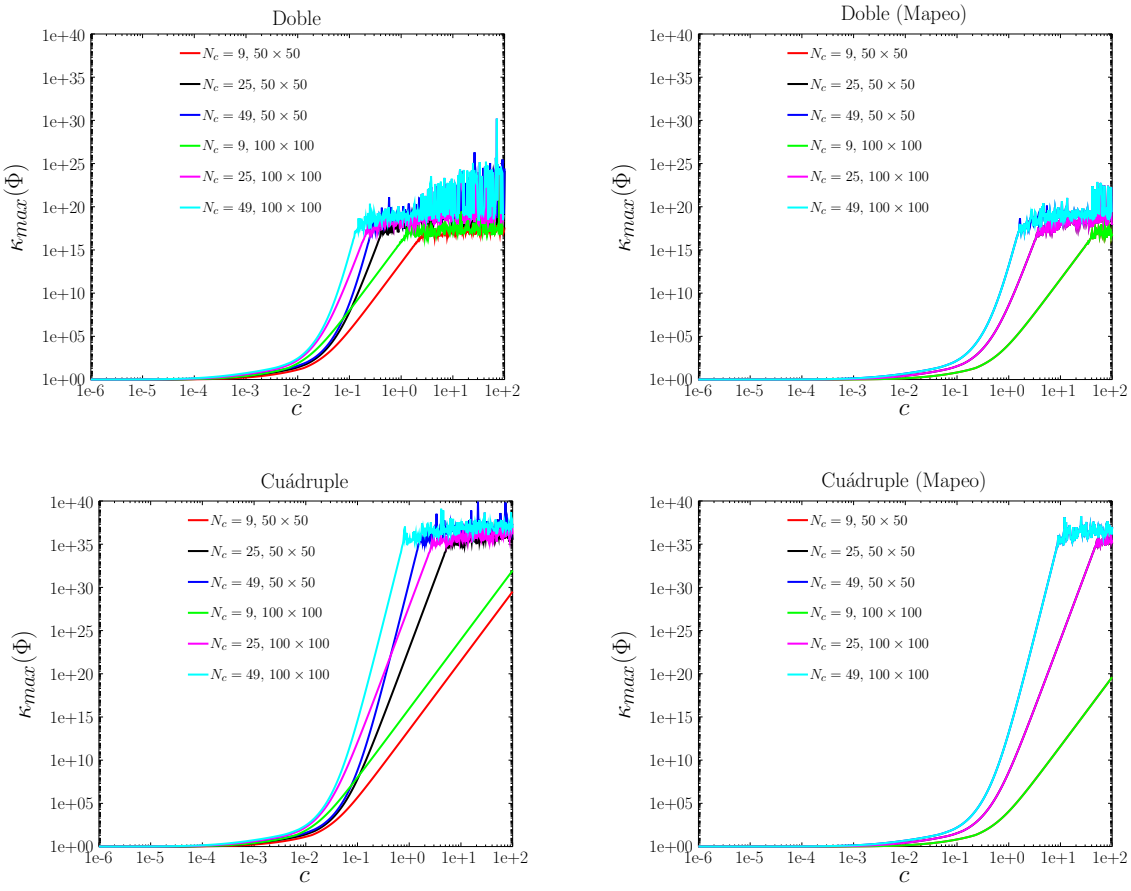

Figura 6.1.1: Números de condición para precisión doble, cuádruple con y sin mapeo unitario.

cada caso, dicho parámetro debía cumplir que el número de condición fuera alto sin importar la precisión a la que se calcularan los pesos. Esto tanto para cada una de las posibilidades de 9, 25 y 49 nodos locales, los mapeos, así como para ambos casos de 50×50 y 100×100.

Las tablas 6.1.1 y 6.1.2 clasifican los experimentos hechos para 50×50 y 100×100 respectivamente con el parámetro de forma elegido en cada caso, una vez hecho el experiemento en estas mismas tablas se recopila la información de salida obtenida como es el caso de las iteraciones hechas con SOR, y el error en éste.

Tabla 6.1.1: Experimentos en arreglos de 50×50

Experimento	N_x	N_y	N_c	Re	c	iter (SOR)	error (SOR)	Precisión	Mapeo
E1	50	50	9	100	0.8	3899	9.9462E-11	doble	No
E2	50	50	25	100	0.2	4275	9.9702E-11	doble	No
E3	50	50	49	100	0.16	5159	9.9638E-11	doble	No
E4	50	50	9	400	0.8	14870	9.9901E-11	doble	No
E5	50	50	25	400	0.2	29343	1.0000E-10	doble	No
E6	50	50	49	400	0.16	35089	9.9961E-11	doble	No
E7	50	50	9	1000	0.8	87235	9.9987E-11	doble	No
E8	50	50	25	1000	0.2	100000	9.5905E-09	doble	No
E9	50	50	49	1000	0.16	100000	5.1113E-08	doble	No
E10	50	50	9	100	13	3899	9.9944E-11	doble	Si
E11	50	50	25	100	1.8	4229	9.9797E-11	doble	Si
E12	50	50	49	100	0.85	5170	9.9684E-11	doble	Si
E13	50	50	9	400	13	14871	9.9972E-11	doble	Si
E14	50	50	25	400	1.8	29039	9.9978E-11	doble	Si
E15	50	50	49	400	0.85	35154	9.9982E-11	doble	Si
E16	50	50	9	1000	13	87245	9.9982E-11	doble	Si
E17	50	50	25	1000	1.8	100000	8.3208E-09	doble	Si
E18	50	50	49	1000	0.85	100000	5.1489E-08	doble	Si
E19	50	50	9	100	100	3897	9.9539E-11	cuádruple	No
E20	50	50	25	100	3	3255	9.9758E-11	cuádruple	No
E21	50	50	49	100	1.3	5336	9.9395E-11	cuádruple	No
E22	50	50	9	400	100	14864	9.9933E-11	cuádruple	No
E23	50	50	25	400	3	22417	9.9942E-11	cuádruple	No
E24	50	50	49	400	1.3	36266	9.9973E-11	cuádruple	No
E25	50	50	9	1000	100	87203	9.9984E-11	cuádruple	No
E26	50	50	25	1000	2	100000	2.0226E-09	cuádruple	No
E27	50	50	49	1000	1	100000	5.7027E-08	cuádruple	No
E28	50	50	9	100	6.5	3897	9.9548E-11	cuádruple	Si
E29	50	50	25	100	0.35	3282	9.9693E-11	cuádruple	Si
E30	50	50	49	100	0.225	5336	9.9814E-11	cuádruple	Si
E31	50	50	9	400	6.5	14864	9.9936E-11	cuádruple	Si
E32	50	50	25	400	0.35	22620	9.9998E-11	cuádruple	Si
E33	50	50	49	400	0.225	36270	9.9973E-11	cuádruple	Si
E34	50	50	9	1000	6.5	87203	9.9987E-11	cuádruple	Si
E35	50	50	25	1000	0.28	100000	5.2004E-09	cuádruple	Si
E36	50	50	49	1000	0.2	100000	5.1971E-08	cuádruple	Si

Tabla 6.1.2: Experimentos en arreglos de 100×100

Experimento	N_x	N_y	N_c	Re	c	iter (SOR)	error (SOR)	Precisión	Mapeo
E37	100	100	9	100	0.38	15300	9.9826E-11	doble	No
E38	100	100	25	100	0.12	15684	9.9991E-11	doble	No
E39	100	100	49	100	0.085	20138	9.9922E-11	doble	No
E40	100	100	9	400	0.38	55026	9.9989E-11	doble	No
E41	100	100	25	400	0.12	100000	1.1710E-10	doble	No
E42	100	100	49	400	0.085	100000	2.3364E-09	doble	No
E43	100	100	9	1000	0.38	100000	2.1077E-06	doble	No
E44	100	100	25	1000	0.12	100000	4.0683E-06	doble	No
E45	100	100	49	1000	0.085	100000	4.4463E-06	doble	No
E46	100	100	9	100	13	15300	9.9831E-11	doble	Si
E47	100	100	25	100	1.8	16491	9.9914E-11	doble	Si
E48	100	100	49	100	0.85	20200	9.9937E-11	doble	Si
E49	100	100	9	400	13	55025	9.9996E-11	doble	Si
E50	100	100	25	400	1.8	100000	2.2563E-10	doble	Si
E51	100	100	49	400	0.85	100000	2.3789E-09	doble	Si
E52	100	100	9	1000	13	100000	2.1077E-06	doble	Si
E53	100	100	25	1000	1.8	100000	4.1620E-06	doble	Si
E54	100	100	49	1000	0.85	100000	4.4512E-06	doble	Si
E55	100	100	9	100	4	15288	9.9987E-11	cuádruple	No
E56	100	100	25	100	0.13	15179	9.9868E-11	cuádruple	No
E57	100	100	49	100	0.09	20176	9.9957E-11	cuádruple	No
E58	100	100	9	400	4	54998	9.9986E-11	cuádruple	No
E59	100	100	25	400	0.13	98039	9.9996E-11	cuádruple	No
E60	100	100	49	400	0.09	100000	2.3851E-09	cuádruple	No
E61	100	100	9	1000	4	100000	2.1032E-06	cuádruple	No
E62	100	100	25	1000	0.13	100000	3.9949E-06	cuádruple	No
E63	100	100	49	1000	0.9	100000	4.4482E-06	cuádruple	No
E64	100	100	9	100	100	15289	9.9802E-11	cuádruple	Si
E65	100	100	25	100	3	12762	9.9990E-11	cuádruple	Si
E66	100	100	49	100	1.3	20786	9.9907E-11	cuádruple	Si
E67	100	100	9	400	100	54998	9.9991E-11	cuádruple	Si
E68	100	100	25	400	3	83002	9.9986E-11	cuádruple	Si
E69	100	100	49	400	1.3	100000	3.1694E-09	cuádruple	Si
E70	100	100	9	1000	100	100000	2.1032E-06	cuádruple	Si
E71	100	100	25	1000	2	100000	4.1042E-06	cuádruple	Si
E72	100	100	49	1000	1	100000	4.4474E-06	cuádruple	Si

6.2. Líneas de flujo

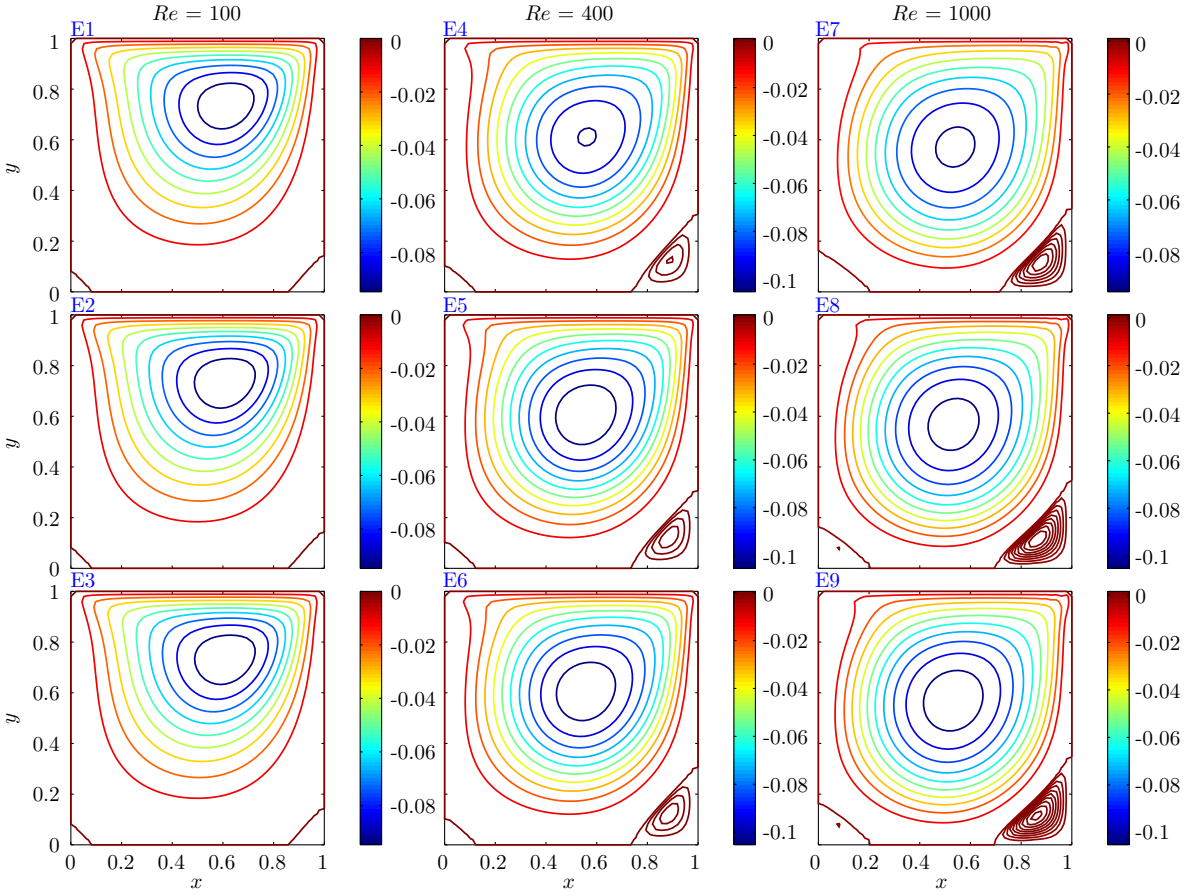

Figura 6.2.1: Función de líneas de flujo con precisión doble.

En la Figura 6.2.1 tenemos los resultados obtenidos sobre la función de líneas de flujo con precisión doble, siguiendo la clasificación de las tablas 6.1.1 y 6.1.2 observamos que no hay una diferencia significativa en las gráficas con el mismo número de Reynolds salvo en la E8 y en la E9 donde se intensifica el número de isolíneas en el vórtice inferior derecho. Esto se debe a que al aumentar el número de Reynolds el número de iteraciones en SOR no fue suficiente para alcanzar la tolerancia ya mencionada.

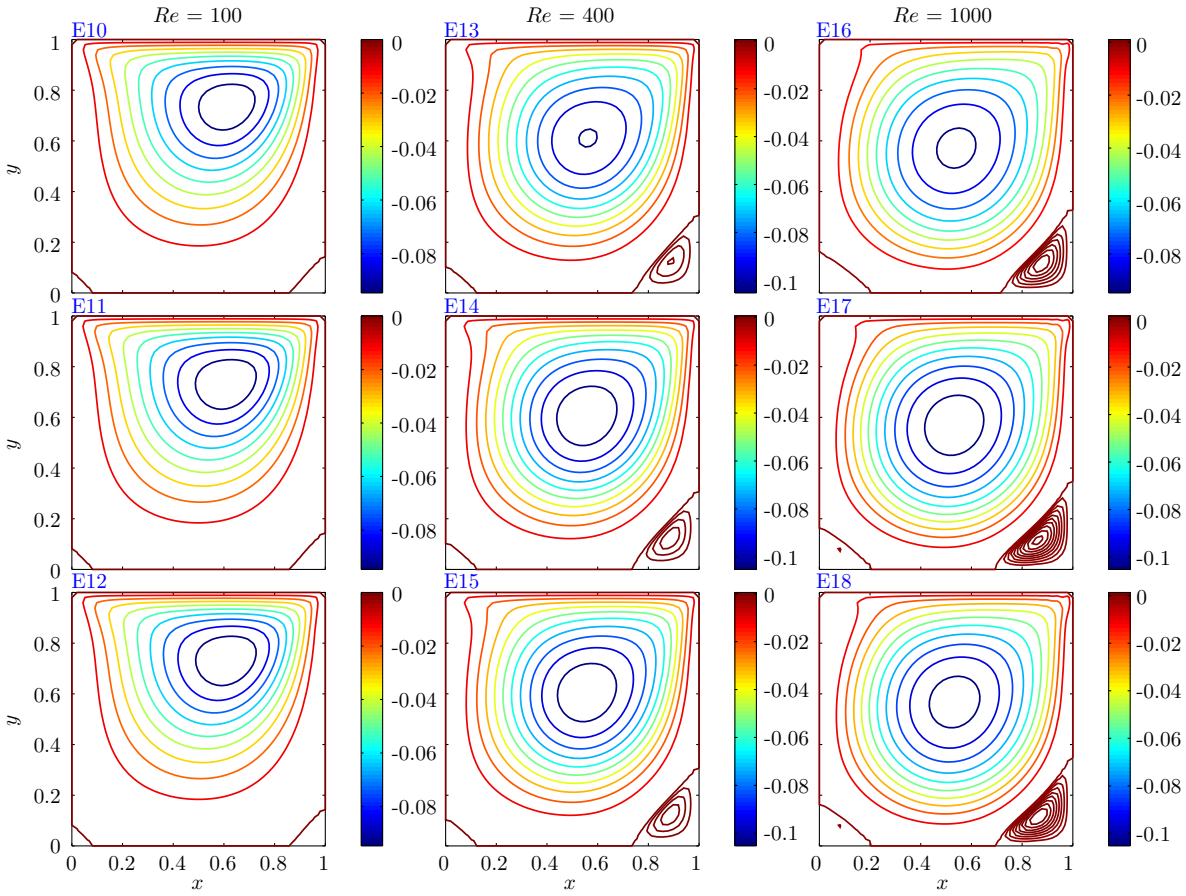

Figura 6.2.2: Función de líneas de flujo con precisión doble y mapeo unitario.

La Figura 6.2.2 contiene los resultados obtenidos sobre la función de líneas de flujo con precisión doble y aplicando el mapeo unitario, en este caso tampoco hay una diferencia significativa en las gráficas con el mismo número de Reynolds, nuevamente es en las gráficas E17 y E18 donde se intensifica el número de isolíneas en el vórtice inferior derecho. En ambos casos no se alcanzó la tolerancia para SOR, sin embargo los órdenes de magnitud son muy cercanos a la tolerancia especificada.

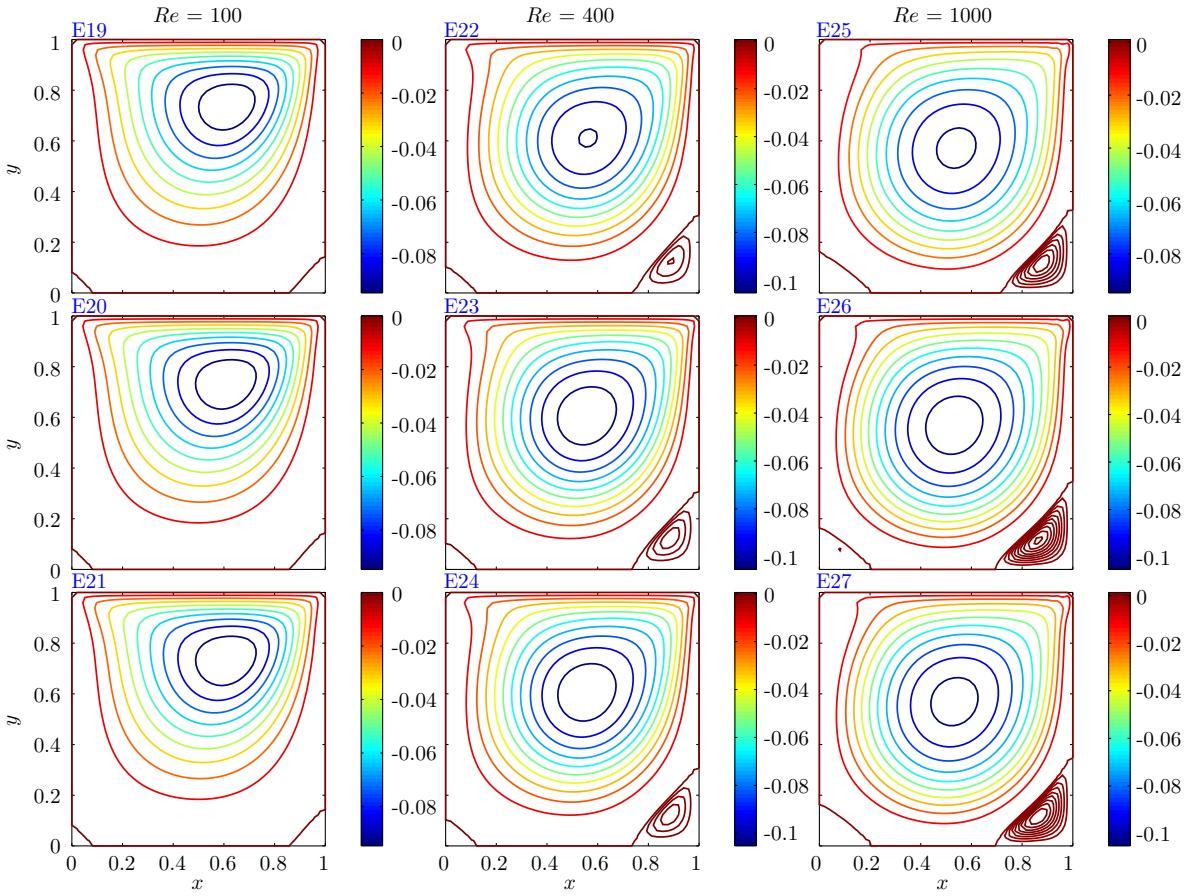

Figura 6.2.3: Función de líneas de flujo con precisión cuádruple.

Para la Figura 6.2.3 tenemos los resultados obtenidos sobre la función de líneas de flujo con precisión cuádruple, nuevamente tenemos una diferencia poco significativa en las gráficas con el mismo número de Reynolds, y al igual que en los casos de precisión doble es en las figuras E26 y E27 donde no se alcanza la tolerancia para SOR.

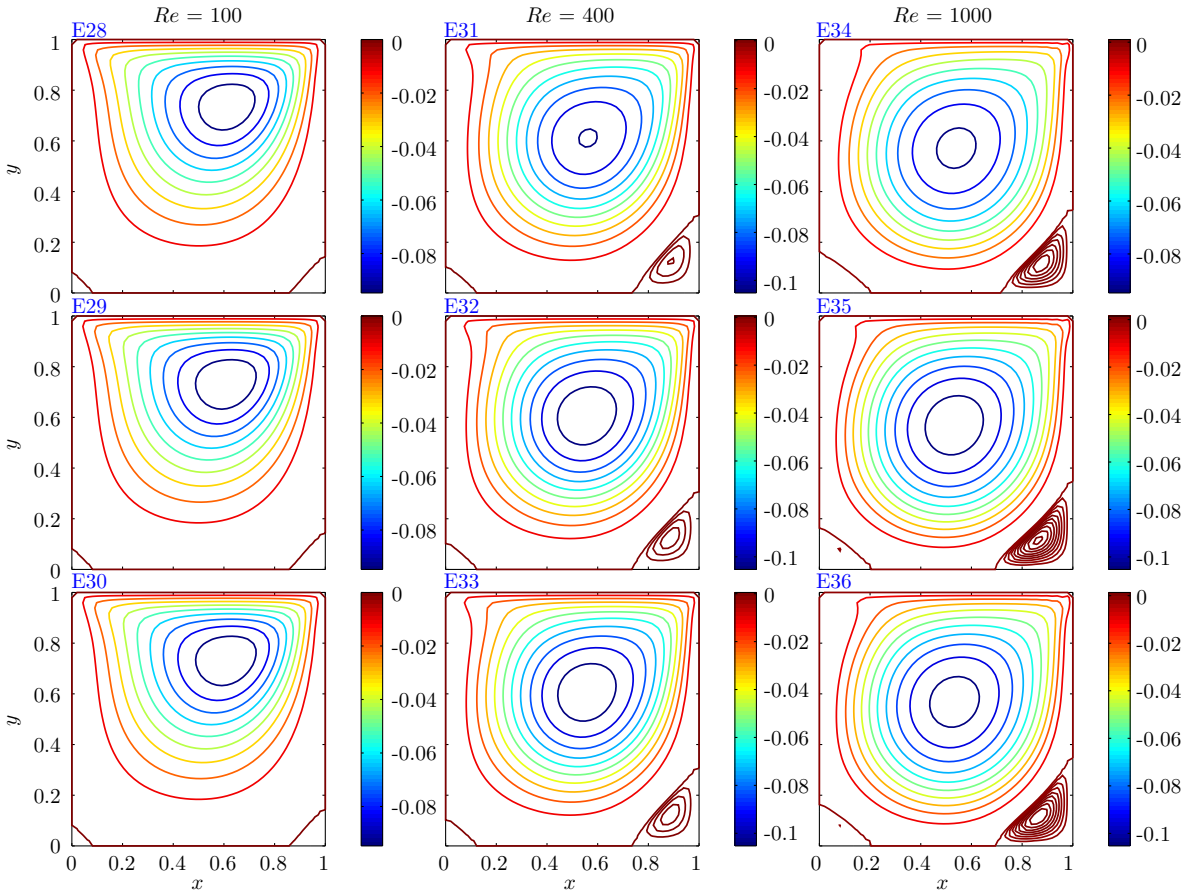

Figura 6.2.4: Función de líneas de flujo con precisión cuádruple y mapeo unitario.

En el caso de la Figura 6.2.4 se encuentran los resultados obtenidos sobre la función de líneas de flujo con precisión cuádruple y aplicando el mapeo unitario, también sucede en este caso que hay poca diferencia entre las figuras con el mismo número de Reynolds, sin embargo al no alcanzar la tolerancia especificada para SOR se intensifican las isolíneas de las figuras E35 y E36.

6.3. Vorticidad

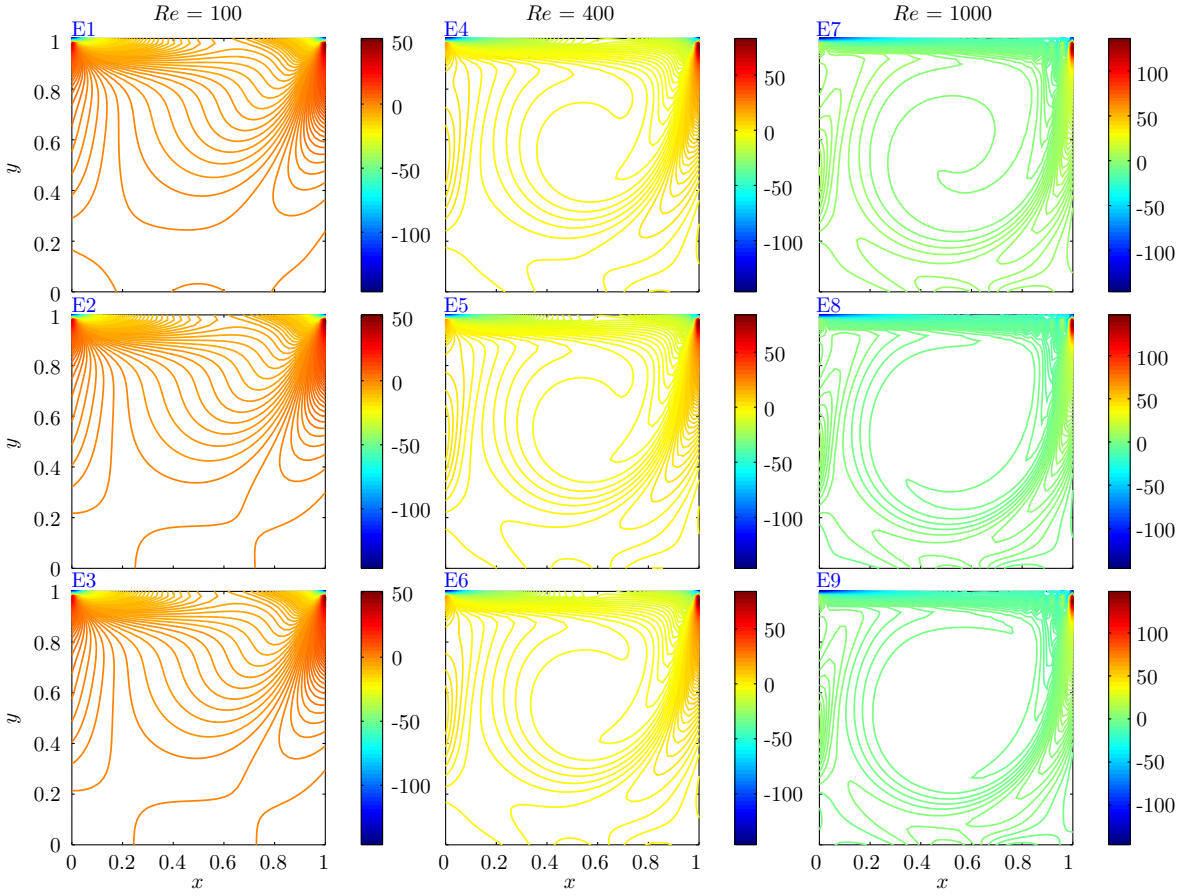

Figura 6.3.1: Vorticidad con precisión doble.

Para la vorticidad con precisión doble 6.3.1 es posible observar diferencias más marcadas entre las figuras con el mismo número de Reynolds. Por ejemplo, para el número de $Re = 100$ tenemos que las figuras E2 y E3 son muy parecidas entre si, sin embargo, E1 tiene marcadas diferencias en su parte inferior. Este mismo patrón se repite para $Re = 400$, mientras que para $Re = 1000$ se observan diferentes detalles en cada una de las figuras.

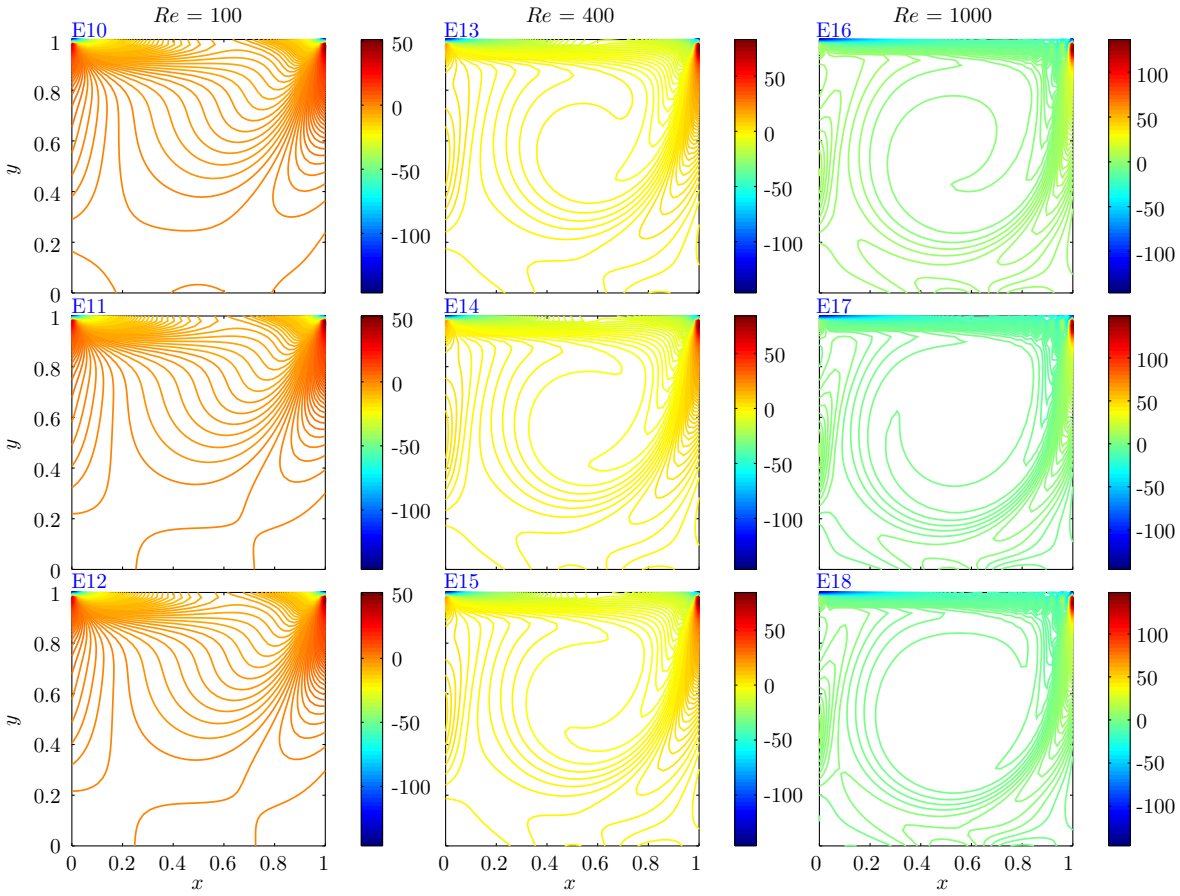

Figura 6.3.2: Vorticidad con precisión doble y mapeo unitario.

En la vorticidad con precisión doble y mapeo unitario cuyos resultados están en la Figura 6.3.2 tenemos un comportamiento muy similar al de los experimentos hechos con precisión doble, es claro que para $Re = 100$ y para $Re = 400$ hay un comportamiento muy similar entre las figuras E11 y E12, así como las figuras E14 y E15. En el caso de $Re = 1000$ se nota un comportamiento diferente entre si para los tres tipos de nodos locales, esto se debe seguramente a que no se alcanzó la tolerancia de SOR.

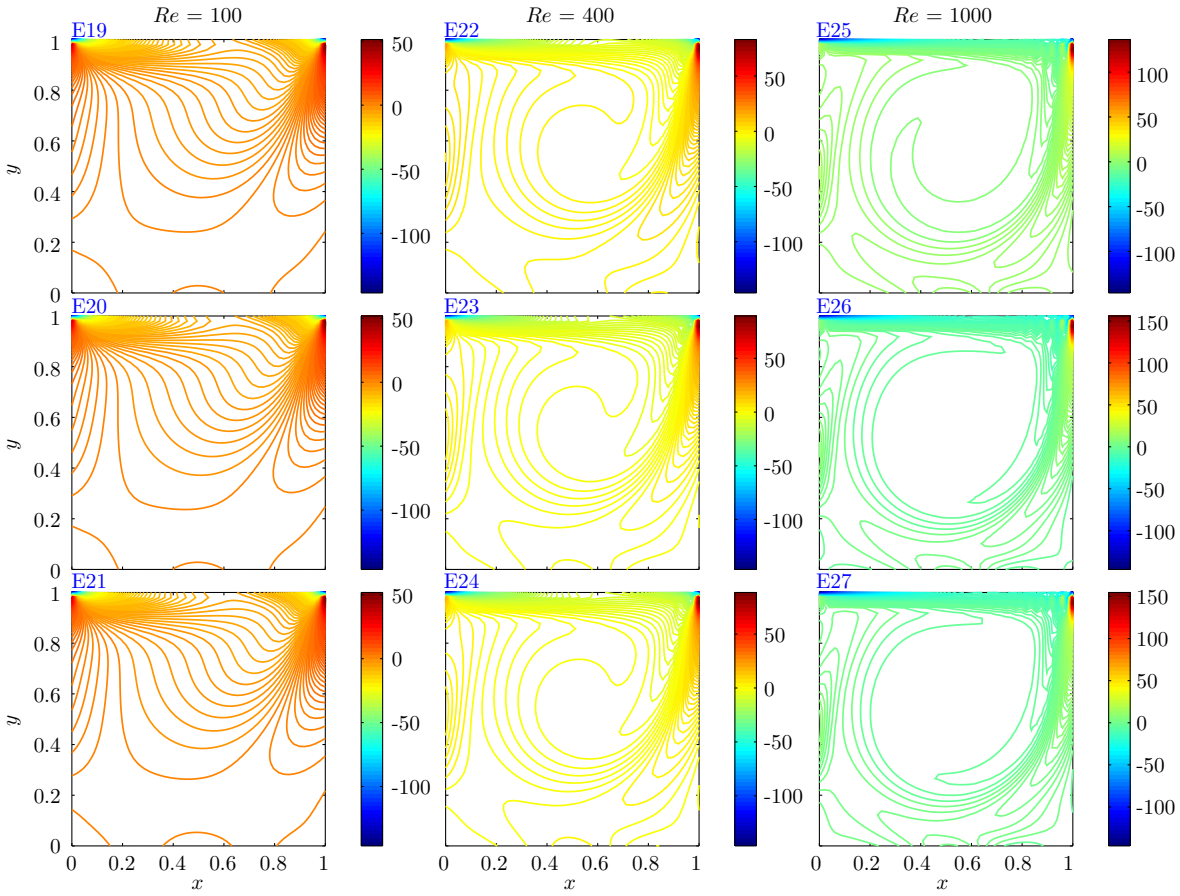

Figura 6.3.3: Vorticidad con precisión cuádruple.

La figura 6.3.3 contiene los experiemtos hechos con precisión cuádruple para la vorticidad, en este caso es posible observar que en $Re = 100$ las figuras son bastante parecidas entre si. Por otro lado en $Re = 400$ podemos observar que las figuras E23 y E24 son casi idénticas, mientras que al compararlas con la E22 se puede apreciar una diferencia muy sutil en la parte inferior izquierda. Finalmente en el caso de $Re = 1000$ se notan pequeñas diferencias entre todas, sin embargo, es claro que la E26 y E27 tienen un error mayor que la E25 pues no alcanzaron la tolerancia de SOR.

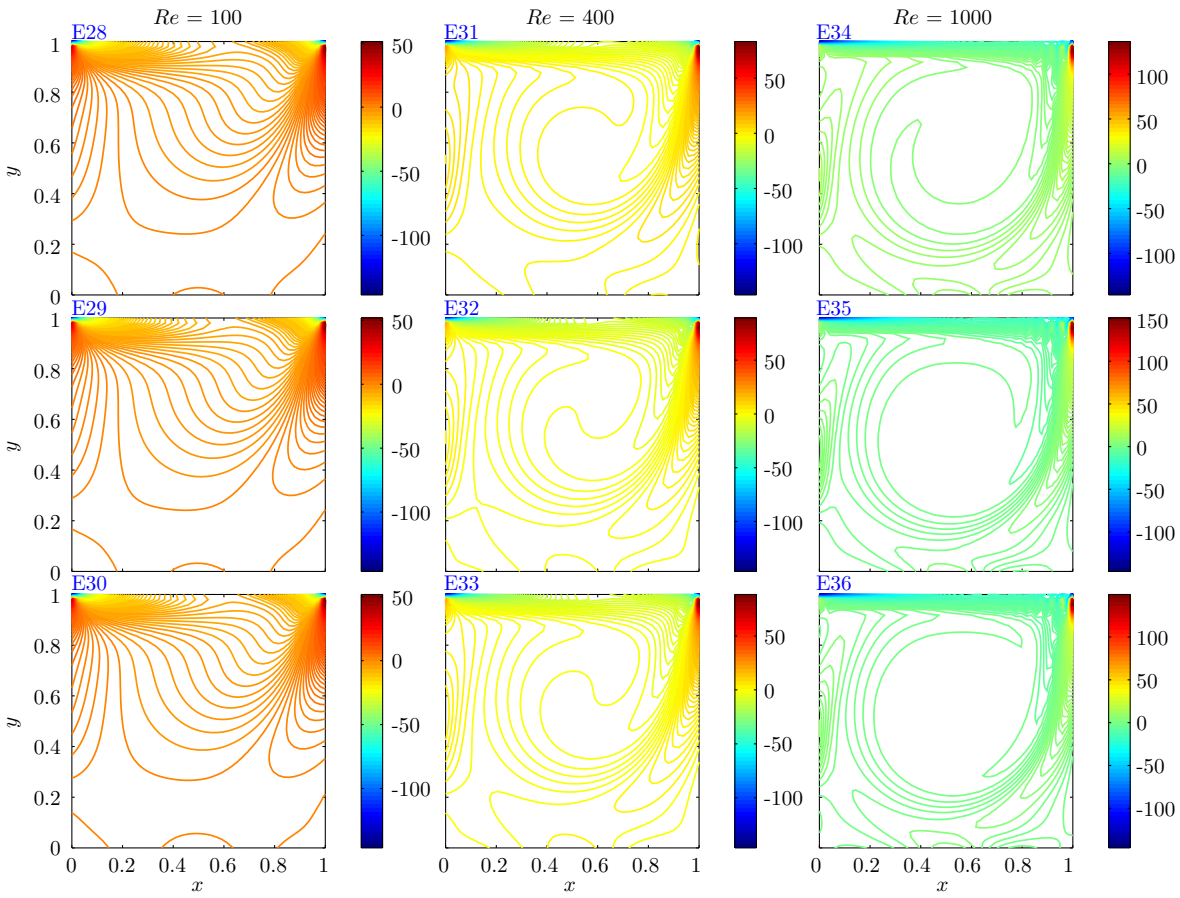

Figura 6.3.4: Vorticidad con precisión cuádruple y mapeo unitario.

En el último caso dado por la vorticidad con precisión cuádruple y mapeo unitario 6.3.4, tenemos resultados muy parecidos al caso en el que sólo se aumentaba la precisión pues para $Re = 100$ todas las figuras son parecidas entre sí. Y para los demás casos todas las figuras tienen diferencias muy sutiles entre si. Por ejemplo, para la gráfica E32 y E35 se observan líneas de contorno poco suaves en su extremo inferior izquierdo. Al igual que en todos los casos anteriores, es de esperarse que las figuras E35 y E36 son las que tienen un mayor error al no alcanzar la tolerancia de SOR.

6.4. Perfiles

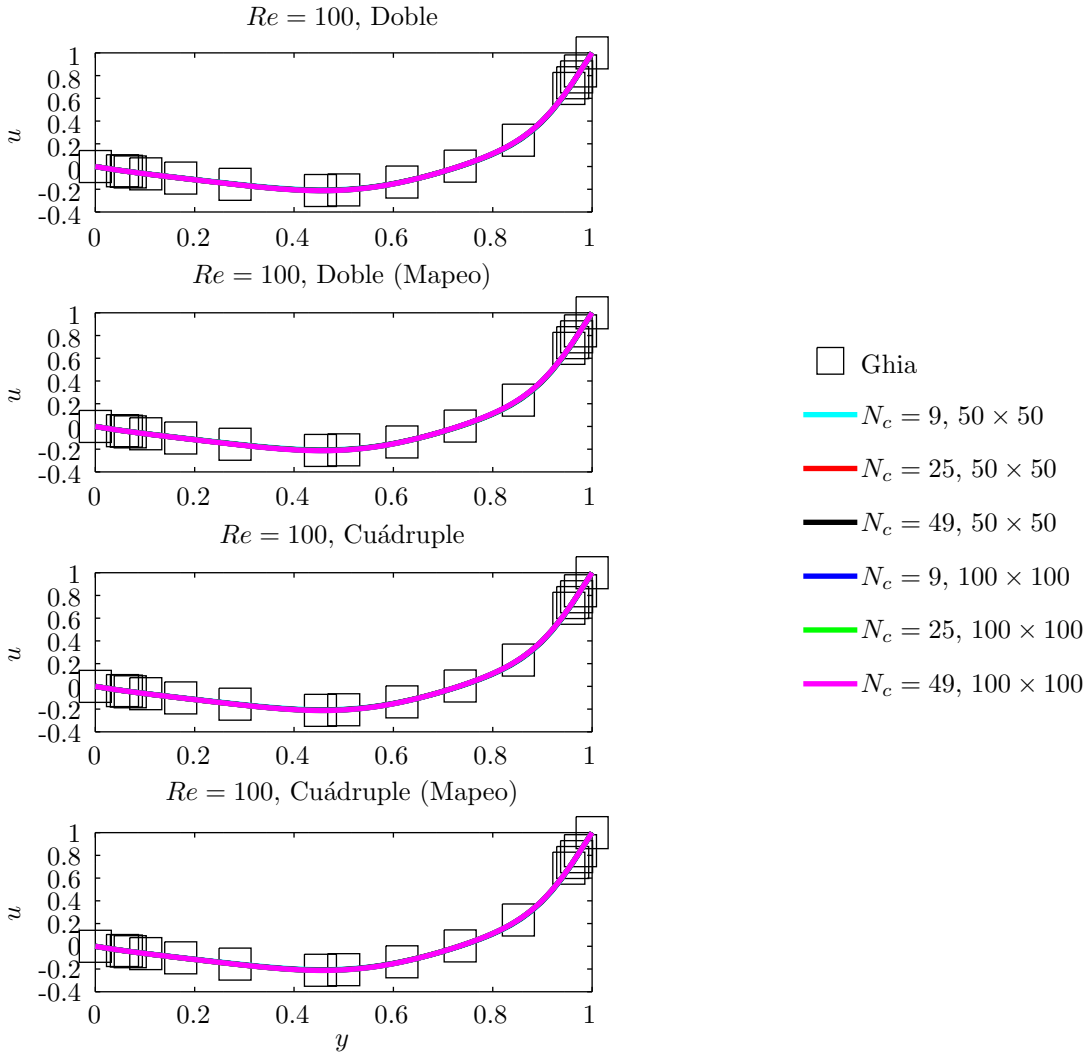

Figura 6.4.1: Perfiles de velocidades u atravesando el centro de la cavidad, $Re = 100$

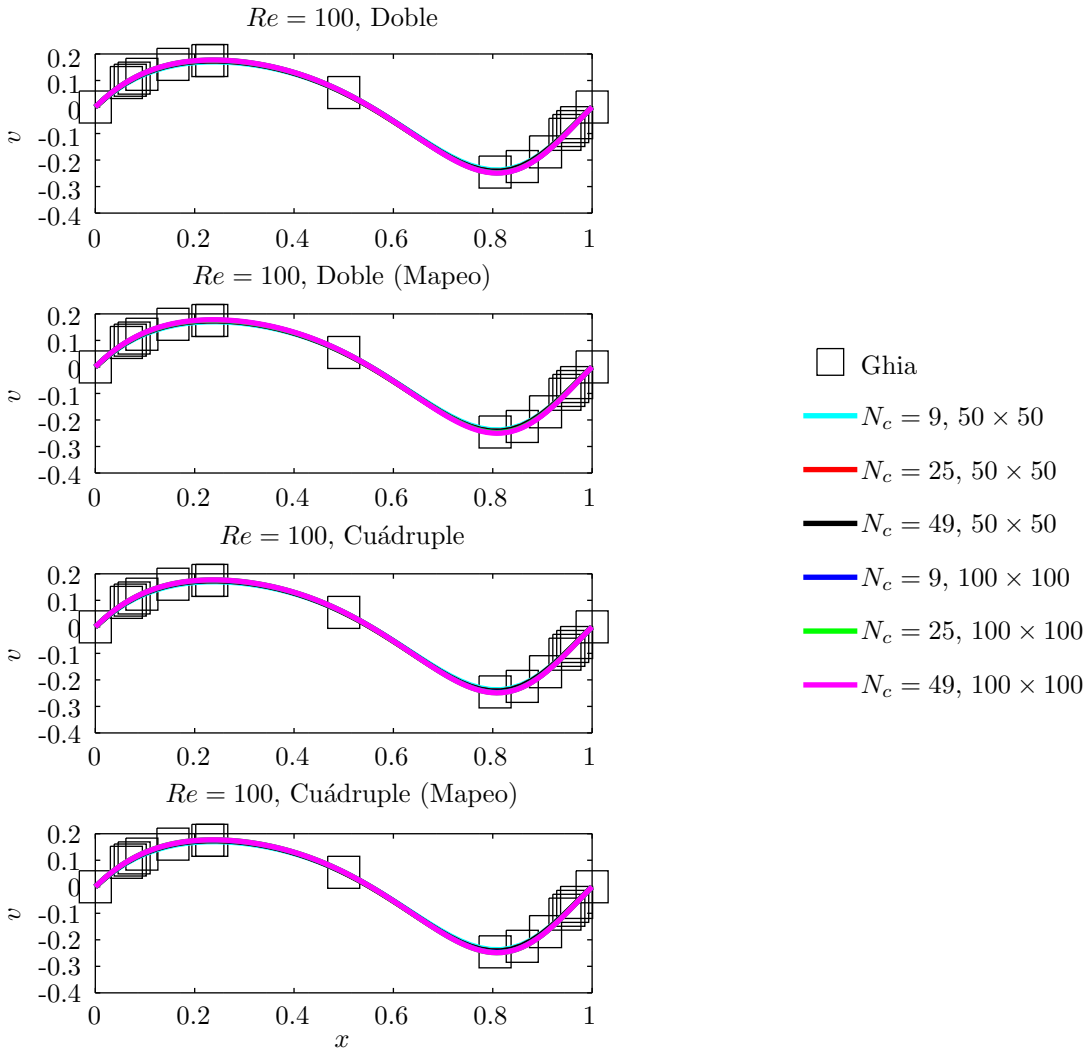

Figura 6.4.2: Perfiles de velocidades v atravesando el centro de la cavidad, $Re = 100$

En las Figuras 6.4.1 y 6.4.2 tenemos los perfiles de velocidades de u y de v a través del centro de la cavidad para $Re = 100$, dichos perfiles fueron comparados con los datos publicados por Ghia, claramente podemos ver que todos los datos experimentos hechos con RBF-FD se ajustan a los datos de Ghia y además se encuentran encimados entre sí, es decir, para fines prácticos tenemos un error despreciable en todos los casos, sin importar la precisión de la máquina.

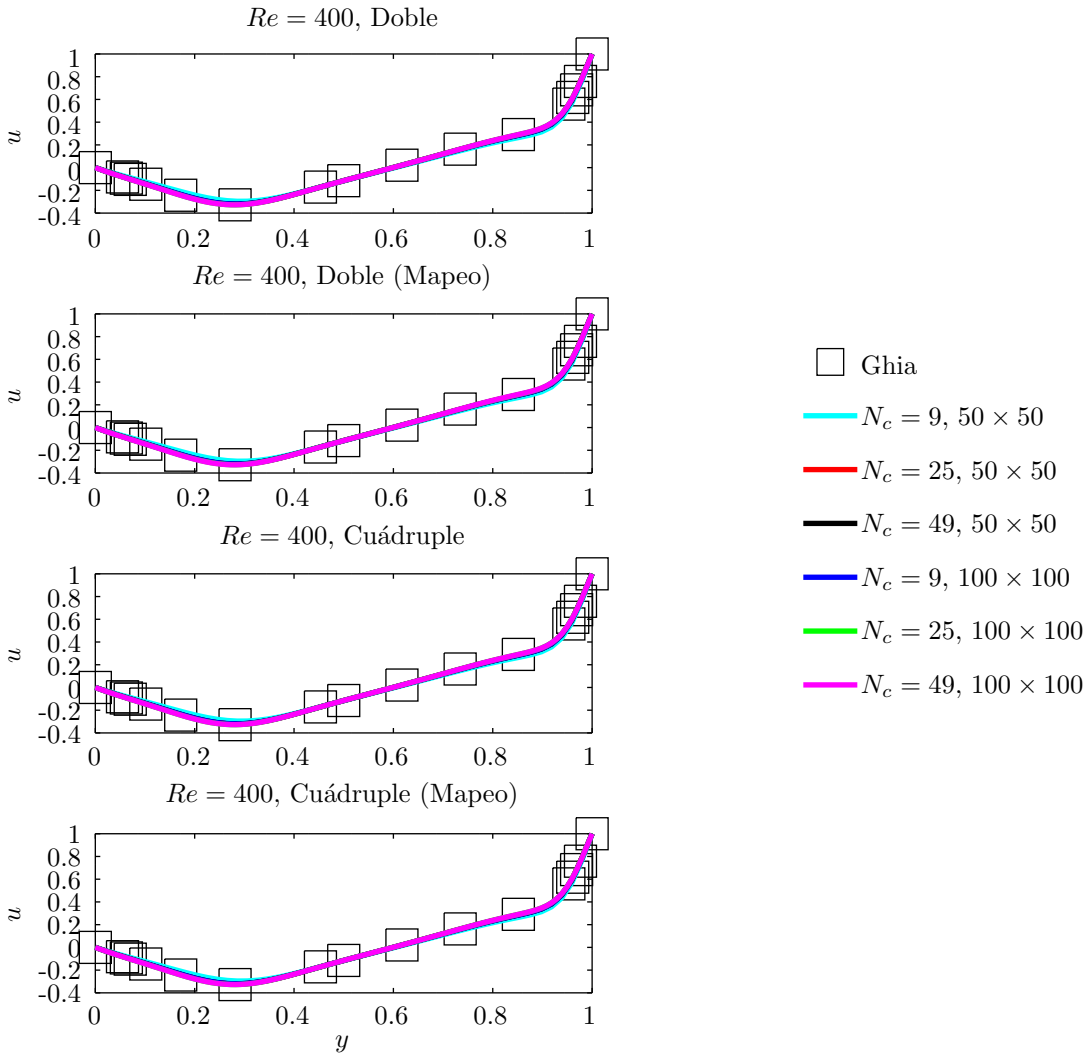

Figura 6.4.3: Perfiles de velocidades u atravesando el centro de la cavidad, $Re = 400$

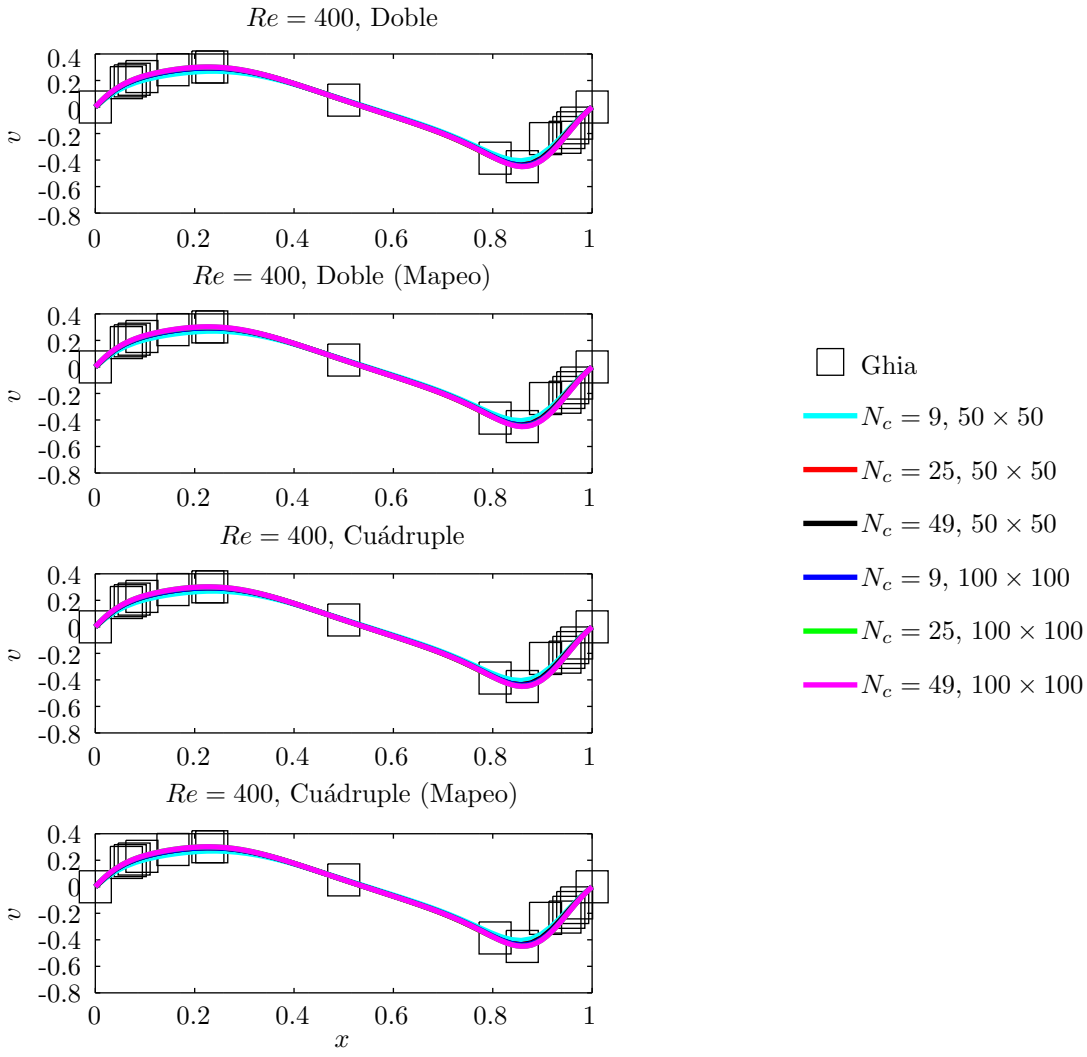

Figura 6.4.4: Perfiles de velocidades v atravesando el centro de la cavidad, $Re = 400$

La Figuras 6.4.3 y 6.4.4 muestran los perfiles de velocidades de u y de v a través del centro de la cavidad para $Re = 400$, en este caso podemos observar que el perfil de 9 nodos locales para el arreglo de 50 por 50 se desajusta ligeramente en todos los casos de RBF-FD, así como con los puntos dados por Ghia. Esto sucede en ambos perfiles u y v, así como los diferentes tipos de precisión.

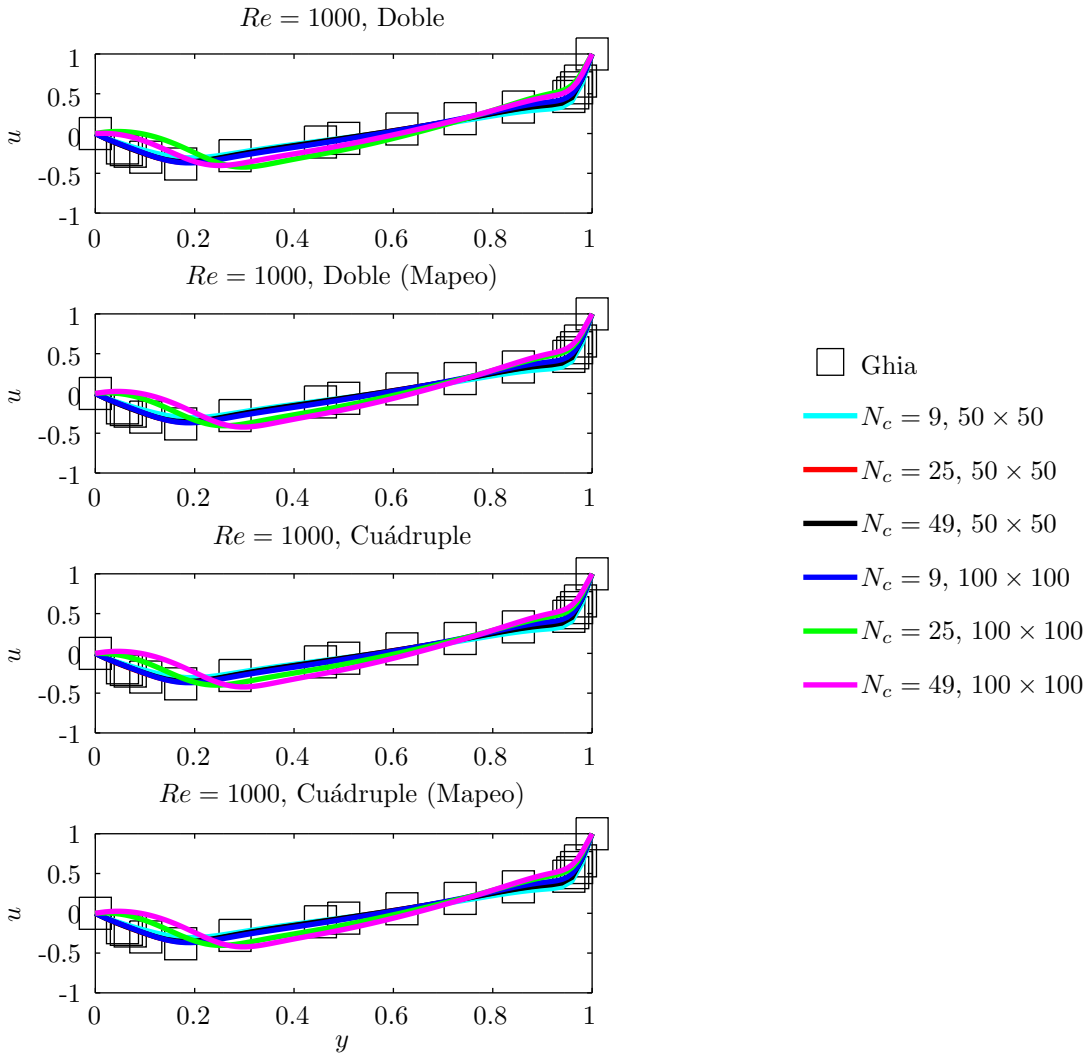

Figura 6.4.5: Perfiles de velocidades u atravesando el centro de la cavidad, $Re = 1000$

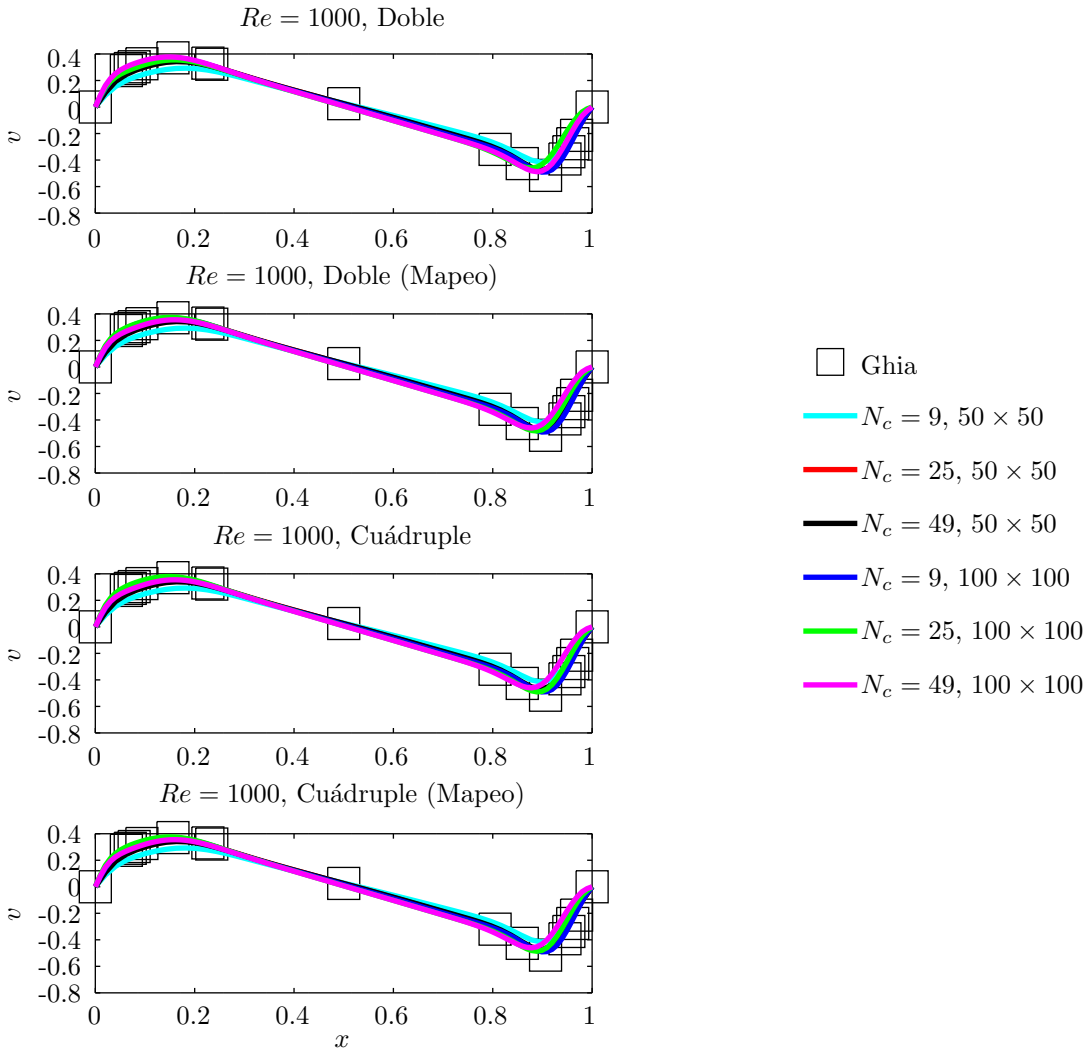

Figura 6.4.6: Perfiles de velocidades v atravesando el centro de la cavidad, $Re = 1000$

En el último caso, tenemos los perfiles de velocidad u y v a través del centro de la cavidad con $Re = 1000$ dados por las Figuras 6.4.5 y 6.4.6, para este caso vemos que el perfil de 9 nodos locales en el arreglo de 50 por 50 se desajusta muy ligeramente tanto a otros perfiles como a los datos dados por Ghia en todos los casos, mientras que los nodos de 25 y 49 nodos locales para el arreglo de 100 por 100 se desajustan abruptamente en el perfil de u.

Tabla 6.4.1: Valores máximos y mínimos de u y v para $Re = 100$

$Re = 100$	E1	E2	E3	E10	E11	E12	De la Cruz	Ghia
u_{min}	-0.20976	-0.21378	-0.21358	-0.20978	-0.2138	-0.21363	-0.22633	-0.2109
v_{min}	-0.24799	-0.25312	-0.25324	-0.24803	-0.25316	-0.25335	-0.27342	-0.24533
v_{max}	0.17527	0.17934	0.17923	0.17529	0.17936	0.17929	0.19566	0.17527

$Re = 100$	E19	E20	E21	E28	E29	E30	De la Cruz	Ghia
u_{min}	-0.20957	-0.21369	-0.21355	-0.20957	-0.21369	-0.21355	-0.22633	-0.2109
v_{min}	-0.24773	-0.25302	-0.25307	-0.24773	-0.25303	-0.25307	-0.27342	-0.24533
v_{max}	0.17511	0.17926	0.17918	0.17511	0.17926	0.17918	0.19566	0.17527

$Re = 100$	E37	E38	E39	E46	E47	E48	De la Cruz	Ghia
u_{min}	-0.21316	-0.21378	-0.21429	-0.21317	-0.21338	-0.21489	-0.22633	-0.2109
v_{min}	-0.25264	-0.25339	-0.25423	-0.25266	-0.25279	-0.25516	-0.27342	-0.24533
v_{max}	0.17863	0.1793	0.17985	0.17865	0.17887	0.18051	0.19566	0.17527

$Re = 100$	E55	E56	E57	E64	E65	E66	De la Cruz	Ghia
u_{min}	-0.21289	-0.21383	-0.21415	-0.2129	-0.21395	-0.214	-0.22633	-0.2109
v_{min}	-0.25226	-0.25349	-0.25403	-0.25226	-0.25368	-0.25378	-0.27342	-0.24533
v_{max}	0.1784	0.17936	0.17971	0.1784	0.1795	0.17955	0.19566	0.17527

Tabla 6.4.2: Errores porcentuales de los valores máximos y mínimos de u y v para $Re = 100$, se usó como punto de comparación el valor obtenido por Ghia

$Re = 100$	E1	E2	E3	E10	E11	E12	De la Cruz
u_{min}	0.54054	1.36558	1.27074	0.53106	1.37506	1.29445	7.31626
v_{min}	1.08425	3.17531	3.22423	1.10056	3.19162	3.26907	11.44988
u_{max}	0	2.32213	2.25937	0.01141	2.33354	2.2936	11.63348

$Re = 100$	E19	E20	E21	E28	E29	E30	De la Cruz
u_{min}	0.63063	1.3229	1.25652	0.63063	1.3229	1.25652	7.31626
v_{min}	0.97827	3.13455	3.15493	0.97827	3.13863	3.15493	11.44988
v_{max}	0.09129	2.27649	2.23084	0.09129	2.27649	2.23084	11.63348

$Re = 100$	E37	E38	E39	E46	E47	E48	De la Cruz
u_{min}	1.0716	1.36558	1.6074	1.07634	1.17591	1.89189	7.31626
v_{min}	2.97966	3.28537	3.62777	2.98781	3.0408	4.00685	11.44988
v_{max}	1.91704	2.29931	2.61311	1.92845	2.05397	2.98967	11.63348

$Re = 100$	E55	E56	E57	E64	E65	E66	De la Cruz
u_{min}	0.94358	1.38928	1.54101	0.94832	1.44618	1.46989	7.31626
v_{min}	2.82477	3.32613	3.54624	2.82477	3.40358	3.44434	11.44988
v_{max}	1.78582	2.33354	2.53323	1.78582	2.41342	2.44195	11.63348

En las tablas 6.4.1 y 6.4.2 tenemos que los resultados de 9 nodos locales son los que mejor se ajustan a los reportados por Ghia en el caso del arreglo de 50×50 nodos, es posible observar para este mismo caso que no hay gran diferencia entre usar o no el mapeo unitario para cada precisión. Al comparar las precisiones para 9 nodos locales, vemos que se ajustan mejor a los de Ghia para v_{min} solamente, pero es complicado afirmar que los resultados por Ghia son mejores que los de RBF-FD pues Ghia usa como método base FD.

Por otro lado, para estas mismas tablas tenemos que los demás resultados también se acercan a los obtenidos por Ghia pero con un error relativamente más significativo que el obtenido para 9 nodos locales, en este caso vemos que los resultados son muy parecidos entre sí con una sutil mejora al usar el cambio de precisión a cuádruple. Otro detalle importante es el hecho de que al aumentar el tamaño de arreglo a 100×100 los resultados también se alejan de manera muy ligera a los Ghia en comparación con el arreglo de 50×50 nodos.

Finalmente podemos decir que los resultados obtenidos por RBF-FD para $Re = 100$ se acercan más a los resultados reportados por Ghia que los de Cruz, esto posiblemente se deba a la elección del parámetro de forma en dicho experimento. En general podemos decir que todos los resultados son buenos, con cierta superioridad para el caso de 9 nodos locales en alguna de sus variantes.

Tabla 6.4.3: Valores máximos y mínimos de u y v para $Re = 400$

$Re = 400$	E4	E5	E6	E13	E14	E15	De la Cruz	Ghia
u_{min}	-0.2987	-0.32567	-0.3246	-0.29874	-0.32555	-0.32454	-0.34309	-0.32726
v_{min}	-0.41436	-0.44752	-0.44783	-0.41441	-0.44737	-0.44788	-0.45963	-0.44993
v_{max}	0.27338	0.30027	0.29949	0.27341	0.30015	0.29946	0.31744	0.30203

$Re = 400$	E22	E23	E24	E31	E32	E33	De la Cruz	Ghia
u_{min}	-0.29845	-0.32441	-0.32381	-0.29845	-0.32442	-0.32384	-0.34309	-0.32726
v_{min}	-0.41403	-0.44595	-0.44597	-0.41403	-0.44596	-0.44602	-0.45963	-0.44993
v_{max}	0.27314	0.29896	0.29853	0.27315	0.29897	0.29856	0.31744	0.30203

$Re = 400$	E40	E41	E42	E49	E50	E51	De la Cruz	Ghia
u_{min}	-0.32167	-0.32837	-0.32862	-0.32169	-0.328	-0.32928	-0.34309	-0.32726
v_{min}	-0.44463	-0.45325	-0.45368	-0.44466	-0.45287	-0.45444	-0.45963	-0.44993
v_{max}	0.29656	0.30342	0.30376	0.29659	0.30301	0.30451	0.31744	0.30203

$Re = 400$	E58	E59	E60	E67	E68	E69	De la Cruz	Ghia
u_{min}	-0.32128	-0.32835	-0.32845	-0.32128	-0.32835	-0.32829	-0.34309	-0.32726
v_{min}	-0.44411	-0.45325	-0.4535	-0.44411	-0.45321	-0.45327	-0.45963	-0.44993
v_{max}	0.29617	0.30342	0.30358	0.29618	0.30343	0.30339	0.31744	0.30203

Tabla 6.4.4: Errores porcentuales de los valores máximos y mínimos de u y v para $Re = 400$, se usó como punto de comparación el valor obtenido por Ghia

$Re = 400$	E4	E5	E6	E13	E14	E15	De la Cruz
u_{min}	8.72701	0.48585	0.81281	8.71478	0.52252	0.83114	4.83713
v_{min}	7.90567	0.53564	0.46674	7.89456	0.56898	0.45563	2.15589
v_{max}	1.81108	0.46022	0.57279	1.80115	0.32447	0.82111	5.10214

$Re = 400$	E22	E23	E24	E31	E32	E33	De la Cruz
u_{min}	8.8034	0.87087	1.05421	8.8034	0.86781	1.04504	4.83713
v_{min}	7.97902	0.88458	0.88014	7.97902	0.88236	0.86902	2.15589
v_{max}	1.9402	0.46022	0.51319	1.93689	0.46353	0.45029	5.10214

$Re = 400$	E40	E41	E42	E49	E50	E51	De la Cruz
u_{min}	1.70812	0.33918	0.41557	1.70201	0.22612	0.61725	4.83713
v_{min}	1.17796	0.73789	0.83346	1.17129	0.65343	1.00238	2.15589
v_{max}	1.81108	0.46022	0.57279	1.80115	0.32447	0.82111	5.10214

$Re = 400$	E58	E59	E60	E67	E68	E69	De la Cruz
u_{min}	1.82729	0.33307	0.36363	1.82729	0.33307	0.31473	4.83713
v_{min}	1.29353	0.73789	0.79346	1.29353	0.729	0.74234	2.15589
v_{max}	1.9402	0.46022	0.51319	1.93689	0.46353	0.45029	5.10214

Para las tablas 6.4.3 y 6.4.4 la cual contiene los experimentos de $Re = 400$, tenemos que los resultados con de 9 nodos locales son los que menos se ajustan a los reportados por Ghia en el arreglo de 50×50 nodos a diferencia del experimento de $Re = 100$, sin embargo, la diferencia en el caso de 9 nodos locales y los datos de Ghia es es bastante aceptable, pues el error es menor al 2 %. Al igual que en el caso anterior, para estas mismas tablas tenemos que hay poca diferencia entre los resultados obtenidos para los casos de 25 y 49 nodos locales.

En el caso del arreglo de 100×100 con 9 nodos locales hay un mayor acercamiento a los resultados de Ghia en comparación con el arreglo de 50×50, pero para los demás casos aumenta la diferencia, esto para ambas precisiones.

Con este número de Reynolds, podemos observar la diferencia entre los resultados obtenidos por Ghia y los de Cruz es menor que con $Re = 100$, ya que el error máximo es del orden de 5 %, en comparación con el de la tabla 6.4.2 donde éste es del orden de 11 %. Esto nos lleva a decir que en general los resultados son aceptables.

Tabla 6.4.5: Valores máximos y mínimos de u y v para $Re = 1000$

$Re = 1000$	E7	E8	E9	E16	E17	E18	De la Cruz	Ghia
u_{min}	-0.30615	-0.3635	-0.35741	-0.30617	-0.36315	-0.35724	-0.3856	-0.38289
v_{min}	-0.4226	-0.48592	-0.48068	-0.42262	-0.48553	-0.48073	-0.51525	-0.5155
v_{max}	0.29457	0.3512	0.34433	0.29459	0.35086	0.3441	0.37829	0.37095

$Re = 1000$	E25	E26	E27	E34	E35	E36	De la Cruz	Ghia
u_{min}	-0.30599	-0.36143	-0.35463	-0.30599	-0.3623	-0.35643	-0.3856	-0.38289
v_{min}	-0.4224	-0.48369	-0.47598	-0.4224	-0.48463	-0.47903	-0.51525	-0.5155
v_{max}	0.29442	0.34914	0.34171	0.29442	0.35003	0.34339	0.37829	0.37095

$Re = 1000$	E43	E44	E45	E52	E53	E54	De la Cruz	Ghia
u_{min}	-3.69E-01	-4.03E-01	-4.23E-01	-3.69E-01	-4.07E-01	-4.23E-01	-3.86E-01	-3.83E-01
v_{min}	-4.95E-01	-4.92E-01	-4.65E-01	-4.95E-01	-4.87E-01	-4.66E-01	-5.15E-01	-5.16E-01
v_{max}	3.71E-01	3.80E-01	3.59E-01	3.71E-01	3.76E-01	3.59E-01	3.78E-01	3.71E-01

$Re = 1000$	E61	E62	E63	E70	E71	E72	De la Cruz	Ghia
u_{min}	-3.69E-01	-4.00E-01	-4.23E-01	-3.69E-01	-4.04E-01	-4.23E-01	-3.86E-01	-3.83E-01
v_{min}	-4.94E-01	-4.95E-01	-4.65E-01	-4.94E-01	-4.90E-01	-4.65E-01	-5.15E-01	-5.16E-01
v_{max}	3.70E-01	3.82E-01	3.58E-01	3.70E-01	3.79E-01	3.59E-01	3.78E-01	3.71E-01

Tabla 6.4.6: Errores porcentuales de los valores máximos y mínimos de u y v para $Re = 1000$, se usó como punto de comparación el valor obtenido por Ghia

$Re = 1000$	E7	E8	E9	E16	E17	E18	De la Cruz
u_{min}	20.04231	5.06412	6.65465	20.03709	5.15553	6.69905	0.70778
v_{min}	18.02134	5.73812	6.75461	18.01746	5.81377	6.74491	0.0485
v_{max}	20.59038	5.32417	7.17617	20.58498	5.41582	7.23817	1.9787

$Re = 1000$	E25	E26	E27	E34	E35	E36	De la Cruz
u_{min}	20.0841	5.60474	7.38071	20.0841	5.37752	6.9106	0.70778
v_{min}	18.06014	6.17071	7.66634	18.06014	5.98836	7.07468	0.0485
v_{max}	20.63081	5.8795	7.88246	20.63081	5.63957	7.42957	1.9787

$Re = 1000$	E43	E44	E45	E52	E53	E54	De la Cruz
u_{min}	3.65535	5.22193	10.44386	3.65535	6.26632	10.44386	0.78329
v_{min}	4.06977	4.65116	9.88372	4.06977	5.62016	9.68992	0.1938
v_{max}	0	2.42588	3.2345	0	1.34771	3.2345	1.88679

$Re = 1000$	E61	E62	E63	E70	E71	E72	De la Cruz
u_{min}	3.65535	4.43864	10.44386	3.65535	5.48303	10.44386	0.78329
v_{min}	4.26357	4.06977	9.88372	4.26357	5.03876	9.88372	0.1938
v_{max}	0.26954	2.96496	3.50404	0.26954	2.15633	3.2345	1.88679

La tablas 6.4.5 y 6.4.6 nos muestran los resultados para $Re = 1000$, para el caso del arreglo de 50×50 tenemos que en 9 nodos locales la diferencia con los resultados de Ghia es significativa, mientras que para los demás nodos en el mismo arreglo esta diferencia no es tan grande que con 9 nodos locales. Por otro lado es necesario decir que fue en este caso cuando la tolerancia de SOR no fue alcanzada para el número de nodos mayor a 9.

En el caso del arreglo de 100×100 vemos una mejora significativa para todos los casos de nodos locales y es para 9 nodos locales donde la diferencia con los datos obtenidos con Ghia es despreciable, también ocurre aquí que la diferencia entre los resultados de Ghia y los de Cruz es pequeña, lo que implica que en este caso los resultados también se acercan a los de Cruz. Podemos decir en general que los resultados son aceptables a pesar de no haber alcanzado la tolerancia de SOR.

Capítulo 7

Conclusiones

En este trabajo se usó el método de diferencias finitas con funciones de base radial para resolver el problema de mecánica de fluidos conocido como el problema de la cavidad con tapa deslizante, para esto se utilizó el kernel Inverso Multicuádrico, pues con éste se tiene convergencia espectral. Uno de los aspectos estudiados fue el comportamiento del número de condición con respecto al parámetro de forma c, así como configuraciones de 9, 25 y 49 nodos locales uniformemente distribuidos tanto en precisión doble como cuádruple.

El primer aspecto a resaltar es que el cambio de precisión de doble a cuádruple aumentó el rango de posibilidades para la elección del parámetro de forma c, pues en este caso los parámetros de forma tienen asociado un número de condición mucho mayor que en precisión doble, los cuales son adecuados por el principio de incertidumbre.

Otra técnica que ayudó a la elección del parámetro de forma fue el uso del mapeo unitario, ya que los resultados muestran una amplificación del rango en el que c es adecuado, como consecuencia tenemos una mayor estabilidad a la hora de escoger éste el parámetro de forma. Por otro lado, el mapeo unitario también permitió que la elección del c no dependiera del espacio entre los nodos locales para cada una de las configuraciones de nodos dadas.

Los resultados en la comparación del parámetro de forma con los números de condición muestran que la cantidad de nodos locales adecuada con las configuraciones de nodos dadas es de $N_c = 9$, ya que el número de condición de su matriz de Gram tarda más en sobrepasar la precisión de la máquina tanto en doble como cuádruple y por ello el error puede ser menor.

Una vez que se eligió un parámetro de forma para cada uno de los casos, los resultados en la simulación del problema de la cavidad con tapa deslizante fueron favorables en todos ellos, sin embargo, una de las complicaciones presentadas fue que el número de iteraciones de SOR aumentaba conforme el número de Reynolds Re lo hacía. Para $Re = 1000$ tenemos que sólo en

algunos casos en los que se usó 9 nodos locales se pudo alcanzar la convergencia o se estaba muy cerca de la tolerancia, desafortundamente las técnicas como el empleo del mapeo unitario y el cambio de precisión no ayudaron de manera significativa en la convergencia de SOR para este valor de Re.

De todo lo anterior podemos concluir que para obtener un resultado favorable con Funciones de Base Radial - Diferencias Finitas es necesario usar siempre una metodología consistente en la aplicación del mapeo unitario y el cambio de precisión de doble a cuádruple. Respecto al número de nodos, debemos enfatizar que en este problema en particular recomendamos sólo 9 nodos, lo cual es congruente con los resultados reportados por Shu et al. [43]. En dicho trabajo, éste recomienda usar a lo más 16 nodos locales para que el error se estabilice en datos aleatorios, pero dadas las configuraciones que escogimos 9 es una opción adecuada.

Respecto a la determinación del valor del parámetro de forma para la solución de problemas en PDEs, es un tema sujeto a numerosas investigaciones. Por ejemplo Fornberg propone el algoritmo QR que en esencia es un cambio de base en el espacio discreto de RBF [16]. Otro enfoque es el que ha sido doptado por Kansa y Sarra [39] que proponen la utilización de precisión extendida de tipo mayor a la cuadruple, esto entre diferentes formulaciones que se han propuesto.

Bibliografía

[1] Allaire, G., Kaber, S. M., and Trabelsi, K. (2008). *Numerical linear algebra*, volume 55. Springer. 22, 24, 34, 52

[2] Atkinson, K. and Han, W. (2005). *Theoretical numerical analysis*, volume 39. Springer. 32

[3] Brown, D., Ling, L., Kansa, E., and Levesley, J. (2005). On approximate cardinal preconditioning methods for solving pdes with radial basis functions. *Engineering Analysis with Boundary Elements*, 29(4):343–353. 36, 49

[4] Buhmann, M. D. (2003). *Radial basis functions: theory and implementations*, volume 12. Cambridge university press. 34

[5] Chen, S., Cowan, C. F., and Grant, P. M. (1991). Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on neural networks*, 2(2):302–309. 2

[6] Cruz, L. M. D. L. (2008). Scat mobility grant report mq-rbf meshless method for solving cfd problems using an object-oriented approach. VI, 49, 50, 53

[7] Dahlquist, G. and Björck, Å. (2008). *Numerical Methods in Scientific Computing*. Number v. 2 in Numerical Methods in Scientific Computing. Society for Industrial and Applied Mathematics. 21

[8] Erturk, E. (2009). Discussions on driven cavity flow. *International Journal for Numerical Methods in Fluids*, 60(3):275–294. 1, 48, 49

[9] Erturk, E., Corke, T. C., and Gökçöl, C. (2005). Numerical solutions of 2-d steady incompressible driven cavity flow at high reynolds numbers. *International Journal for Numerical Methods in Fluids*, 48(7):747–774. 1, 47

[10] Fasshauer, G. E. (2007). *Meshfree approximation methods with MATLAB*, volume 6. World Scientific. 33

[11] Fletcher, C. (2012). *Computational techniques for fluid dynamics 2: Specific techniques for different flow categories*. Springer Science & Business Media. 48

[12] Folland, G. B. (1992). *Fourier analysis and its applications*, volume 4. American Mathematical Soc. 28

[13] Folland, G. B. (1995). *Introduction to partial differential equations*. Princeton university press. 86, 87, 88, 89

[14] Fornberg, B. (1988). Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, 51(184):699–706. 14

[15] Fornberg, B. (1998). Classroom note: Calculation of weights in finite difference formulas. *SIAM review*, 40(3):685–691. 14

[16] Fornberg, B., Larsson, E., and Flyer, N. (2011). Stable computations with gaussian radial basis functions. *SIAM Journal on Scientific Computing*, 33(2):869–892. 80

[17] Fornberg, B. and Piret, C. (2008). On choosing a radial basis function and a shape parameter when solving a convective pde on a sphere. *Journal of Computational Physics*, 227(5):2758–2780. 26

[18] Fornberg, B. and Wright, G. (2004). Stable computation of multiquadric interpolants for all values of the shape parameter. *Computers & Mathematics with Applications*, 48(5):853–867. 26

[19] Fortin, A., Jardak, M., Gervais, J., and Pierre, R. (1997). Localization of hopf bifurcations in fluid flow problems. *International Journal for Numerical Methods in Fluids*, 24(11):1185–1210. 49

[20] Ghia, U., Ghia, K. N., and Shin, C. (1982). High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, 48(3):387–411. VI, 1, 47, 49, 50, 53

[21] Gilbert, J. R. and Peierls, T. (1988). Sparse partial pivoting in time proportional to arithmetic operations. *SIAM Journal on Scientific and Statistical Computing*, 9(5):862–874. 38

[22] Golan, J. (1995). Foundations of linear algebra, volume 11 of kluwer texts in the mathematical sciences. 8

[23] Golub, G. and Van Loan, C. (2013). Matrix computations, 4th. *Johns Hopkins*. 11

[24] González-Casanova, P., Muñoz-Gómez, J. A., and Rodríguez-Gómez, G. (2009). Node adaptive domain decomposition method by radial basis functions. *Numerical Methods for Partial Differential Equations*, 25(6):1482–1501. 36

[25] Harder, R. L. and Desmarais, R. N. (1972). Interpolation using surface splines. *Journal of aircraft*, 9(2):189–191. 2

[26] Hardy, R. L. (1971). Multiquadric equations of topography and other irregular surfaces. *Journal of geophysical research*, 76(8):1905–1915. 2

[27] Hardy, R. L. (1977). Least squares prediction. *Photogrammetric Engineering and Remote Sensing*, 43(4). 2

[28] Kansa, E. J. (1990a). Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—i surface approximations and partial derivative estimates. *Computers & Mathematics with applications*, 19(8-9):127–145. 35, 36

[29] Kansa, E. J. (1990b). Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics—ii solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers & mathematics with applications*, 19(8):147–161. 35, 36

[30] Kansa, E. J., Aldredge, R. C., and Ling, L. (2009). Numerical simulation of two-dimensional combustion using mesh-free methods. *Engineering analysis with boundary elements*, 33(7):940–950. 36

[31] Katz, J. (2010). *Introductory fluid mechanics*. Cambridge University Press. 43, 46

[32] Koseff, J. and Street, R. (1984). The lid-driven cavity flow: a synthesis of qualitative and quantitative observations. *J. Fluids Eng*, 106(12):390–398. 49

[33] Larson, M. G. and Bengzon, F. (2013). *The finite element method: Theory, implementation, and applications*, volume 10. Springer Science & Business Media. 1

[34] Martin, B., Fornberg, B., and St-Cyr, A. (2014). Seismic modeling with radial basis function-generated finite differences (rbf-fd) (seismic modeling with rbf-fd). 37

[35] Pozrikidis, C. (2011). *Introduction to theoretical and computational fluid dynamics*. Oxford university press. 46

[36] Rippa, S. (1999). An algorithm for selecting a good value for the parameter c in radial basis function interpolation. *Advances in Computational Mathematics*, 11(2):193–210. 26

[37] Saad, Y. (2003). *Iterative methods for sparse linear systems*. SIAM. 3, 10, 24, 52

[38] Saad, Y. and Schultz, M. H. (1986). Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869. 12

[39] Sarra, S. A. (2011). Radial basis function approximation methods with extended precision floating point arithmetic. *Engineering Analysis with Boundary Elements*, 35(1):68–76. 80

[40] Sarra, S. A. and Kansa, E. J. (2009). Multiquadric radial basis function approximation methods for the numerical solution of partial differential equations. *Advances in Computational Mechanics*, 2(2). 2, 36

[41] Schaback, R. (1995). Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3(3):251–264. 33, 34

[42] Schoenberg, I. J. (1938). Metric spaces and completely monotone functions. *Annals of Mathematics*, pages 811–841. 30

[43] Shu, C., Ding, H., and Yeo, K. (2003). Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 192(7):941–954. 37, 38, 39, 50, 80

[44] Street, R. (1984). Visualization studies of a shear driven three-dimensional recirculating flow. *Journal of Fluids Engineering*, 106:21. 49

[45] Tolstykh, A. I. (2000). On using rbf-based differencing formulas for unstructured and mixed structured-unstructured grid calculations. In *Proceedings of the 16th IMACS World Congress*, volume 228, pages 4606–4624. 2, 36

[46] Wendland, H. (2004). Scattered data approximation. 29, 31, 33

[47] Widder, D. V. (1946). *The Laplace Transformation*. Princeton University Press. 30

[48] Zhang, F. (2011). *Matrix theory: basic results and techniques*. Springer Science & Business Media.

Apéndice A

Análisis Funcional

En este apéndice se introducirán algunas nociones de Espacios de Sobolev y su relación con la Transformada de Fourier.

A.1. Espacios de Sobolev

Empezaremos introduciendo el vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{N}^d$, a este lo llamaremos multi-índice de orden $|\alpha| = \Sigma_{i=1}^{d} \alpha_i$.

Definición A.1.1. Sea $x \in \Omega \subseteq \mathbb{R}^d$, y sea α un multi-índice, entonces definimos a la α-ésima potencia de x como A.1

$$x = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_d^{\alpha_d}. \tag{A.1}$$

En el caso de las funciones derivables en sentido clásico, es posible introducir la idea de derivada en términos de notación multi-índice.

Definición A.1.2. Sea $f \in C^k(\Omega)$, donde $\Omega \subseteq \mathbb{R}^d$ y sea α un multi-índice de orden $|\alpha| \leq k$ definimos la α-ésima derivada de f como

$$D^\alpha f(x) := \frac{\partial^{|\alpha|} f(x)}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_d^{\alpha_d}} = \partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \cdots \partial x_d^{\alpha_d} f(x). \tag{A.2}$$

En la teoría de los espacios de Sobolev es posible hacer una generalización de la derivada, para ello consideremos el conjunto $C_c^\infty(\mathbb{R}^d)$, este es el conjunto de las funciones infinitamente

diferenciables y tales que tienen soporte compacto en $\mathbb{R}^d$. A los elementos de este conjunto generalmente se les llama funciones de prueba.

Otro conjunto importante dentro de esta teoría es el de las funciones localmente integrables $L^1_{loc}(\Omega)$, es decir aquellas que pertenecen a $L_1(A)$, donde A es cualquier compacto de Ω.

Definición A.1.3. Sea α un multi-índice y sean $f, g \in L^1_{loc}(\Omega)$, decimos que g es la α-ésima derivada débil de f si para toda funcón de prueba ϕ satisface la expresión A.3

$$\int_\Omega f D^\alpha \phi dx = (-1)^{|\alpha|} \int_\Omega g\phi dx. \tag{A.3}$$

En este caso denotamos a la derivada débil como $D^\alpha f = g$.

Debemos hacer enfasis que en los casos en los que f tiene su α-ésima derivada en sentido clásico, esta coincide con la débil.

Definición A.1.4. Sean $k \in \mathbb{N}$ y $p \in [0, \infty]$, llamaremos espacio de Sobolev $W^{k,p}(\Omega)$ al conjunto dado por A.4

$$W^{k,p}(\Omega) = \{f \in L_p(\Omega) \mid D^\alpha f \in L_p(\Omega),\ |\alpha| \leq k\}. \tag{A.4}$$

Los elementos f de estos espacios pueden ser dotados con una norma, la cual está dada por

1. Para $p \in [1, \infty)$

$$\|f\|_{W^{k,p}(\Omega)} = \left(\sum_{|\alpha| \leq k} \| D^\alpha f \|^p_{L_p(\Omega)} \right)^{1/p}. \tag{A.5}$$

2. Y si $p = \infty$

$$\|f\|_{W^{k,p}(\Omega)} = \sum_{|\alpha| \leq k} \| D^\alpha f \|_{L_p(\Omega)}. \tag{A.6}$$

Un teorema que nos habla de la completes de los espacios de Sobolev es el siguiente.

Teorema A.1.1. *Los espacios $W^{k,p}(\Omega)$ son de Banach.*

Demostración. Puede consultar Folland [13] página 199. □

En el caso de $p = 2$, el producto interior definido por $\langle \cdot, \cdot \rangle_{L_2(\Omega)}$ induce el producto interior A.7

$$\langle f, g \rangle_{W^{k,2}(\Omega)} = \sum_{|\alpha| \leq k} \langle D^\alpha f, D^\alpha g \rangle_{L_2(\Omega)}, \tag{A.7}$$

por lo que $W^{k,2}(\Omega)$ es un espacio de Hilbert, el cual denotaremos por $W^{k,2}(\Omega) = H^k(\Omega)$.

A.2. Transformada de Fourier

Definición A.2.1. Sean $f \in L_1(\mathbb{R}^d)$, se le llama transformada de Fourier a la función

$$\mathcal{F} : L_1(\mathbb{R}^d) \to L_\infty(\mathbb{R}^d) \tag{A.8}$$

dada por

$$\mathcal{F}f(\xi) = \widehat{f}(\xi) := (2\pi)^{-d/2} \int_{\mathbb{R}^d} f(x) e^{-ix\cdot\xi} dx. \tag{A.9}$$

Definición A.2.2. Llamaremos espacio de Schwarz al espacio de las funciones dado por

$$\mathcal{S}(\mathbb{R}^d) := \{ f \in C^\infty(\mathbb{R}^d) \,|\, x^\beta D^\alpha f \in L_\infty(\mathbb{R}^d) \quad \forall \alpha, \beta \in \mathbb{N}^d \}. \tag{A.10}$$

Una característica importante de los espacios de Schwarz es la siguiente.

Proposición A.2.1.

$$\mathcal{S}(\mathbb{R}^d) \subseteq L_p(\mathbb{R}^d), \quad p \in [1, \infty] \tag{A.11}$$

Demostración. Puede consultar Folland [13] página 4. □

En base a esto, es claro que se puede redefinir la Transformada de Fourier en el espacio de Schwarz

$$\mathcal{F} : \mathcal{S}(\mathbb{R}^d) \to \mathcal{S}(\mathbb{R}^d), \tag{A.12}$$

la cual está bien definida, pues se cumplen las siguientes relaciones

$$\mathcal{F}(x^\beta f) = (i)^{|\beta|} D_\xi^\beta \widehat{f} \tag{A.13}$$

$$\mathcal{F}(D_x^\alpha f) = (i)^{|\alpha|}\xi^\alpha \widehat{f}. \tag{A.14}$$

Teorema A.2.2 (Plancherel)**.** *La transformada de Fourier $\mathcal{F} : \mathcal{S}(\mathbb{R}^d) \to \mathcal{S}(\mathbb{R}^d)$ es biyectiva y es un isomorfismo cuya inversa está dada por*

$$\mathcal{F}^* f(x) = (2\pi)^{-d/2} \int_{\mathbb{R}^d} f(\xi) e^{ix\cdot\xi} d\xi. \tag{A.15}$$

Demostración. Véase Folland [13] página 17. □

El espacio dual de $\mathcal{S}(\mathbb{R}^d)$ está dado por $\mathcal{S}(\mathbb{R}^d)^*$ y también se le conoce como el espacio de las distribuciones temperadas, y sobre este espacio se puede generalizar el concepto de derivada, es decir, para α un multi-índice se define la derivada α-ésima como

$$D^\alpha : \mathcal{S}(\mathbb{R}^d)^* \to \mathcal{S}(\mathbb{R}^d)^* \tag{A.16}$$

tal que para $T \in \mathcal{S}(\mathbb{R}^d)^*$ y f se cumple la relación

$$\langle D^\alpha T, f\rangle = (-1)^{|\alpha|}\langle D^\alpha T, f\rangle. \tag{A.17}$$

Para poder definir una generalización de la transformada de Fourier, definimos el conjunto

$$\mathcal{S}_m := \{\phi \in \mathcal{S} | \phi(x) = \mathcal{O}(\|x\|^m), \quad \|x\| \to 0,\, m \in \mathbb{N}\}. \tag{A.18}$$

donde la notación $\phi(x) = \mathcal{O}(\|x\|^m)$ indica que el término $\|\phi(x)\|/\|x\|^m$ es acotado en una vecindad de cero.

Definición A.2.3. Sea f una función continua que crece a lo más como un polinomio y sea $\phi \in \mathcal{S}_{2m}$, decimos que $\widehat{f}$ es la transformada de Fourier generalizada de f, si satisface la igualdad A.19

$$\int_{\mathbb{R}^d} f\widehat{\phi} dx = \int_{\mathbb{R}^d} \phi\widehat{f} dx. \tag{A.19}$$

Cabe mencionar que en el caso de las funciones continuas tales que $f \in L_1(\mathbb{R}^d) \cup L_2(\mathbb{R}^d)$, la transformada generalizada de Fourier coincide con la transformada de Fourier clásica de orden 0.

Finalmente podemos decir que una caracterización importante del espacio de Sobolev $H^k(\mathbb{R}^d)$ se puede dar en términos de la transformada de Fourier.

Proposición A.2.3. *Sean las normas $\|f\|_{H_2(\mathbb{R}^d)}$ y A.20, entonces $f \in H^k(\mathbb{R}^d)$ si y solo si $(1 + |\xi|^2)\widehat{f} \in L_2(\mathbb{R}^d)$ y dichas normas son equivalentes.*

$$\|f\|_s^2 := \int_{\mathbb{R}^d} |\widehat{f}(\xi)|^2 (1 + |\xi|^2)^s d\xi. \tag{A.20}$$

Demostración. Consulte Folland [13] 191. □

Lo cual nos lleva a la generalización de $H^k(\mathbb{R}^d)$ dada por el conjunto A.21

$$H^s(\mathbb{R}^d) := \{f \in \mathcal{S}(\mathbb{R}^d)^* | \widehat{f}, \text{ es función}, \|f\|_s < \infty\}. \tag{A.21}$$

Apéndice B

Códigos Fortran 95

B.1. Módulos para RBF

B.1.1. Módulo para el cálculo del kernel IMQ y sus derivadas

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Parameters

! c = Shape parameter
! Xi = Node in position i (not confuse with the i-component of X)
! Xj = Node in position j (not confuse with the j-component of X)
! k = Spatial component (Partial derivative is in this component)
! r = Distance between Xi and Xj

module RBF_kernels

contains

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!IMQ (Inverse Multiquadric)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function IMQ(r,c) result(phi)

real(kind = 16),intent(in)::r,c
```

```
real(kind = 16)::phi

    phi = 1.q0 / sqrt(r*r + c*c)

end function IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!der_xi_IMQ (Derivative Inverse Multiquadric)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function der_xi_IMQ(r,c,Xi,Xj,k) result(derxi)

integer,intent(in)::k
real(kind = 16),intent(in)::r,c
real(kind = 16),dimension(:),intent(in)::Xi,Xj
real(kind = 16)::dif, derxi

     dif = delta(Xi,Xj,k)
     derxi = IMQ(r,c)
     derxi = dif * derxi * derxi * derxi

end function der_xi_IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!der_2_xi_IMQ (Second order derivative of Inverse Multiquadric)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function der_2_xi_IMQ(r,c,Xi,Xj,k) result(derxi)

integer,intent(in)::k
real(kind = 16),intent(in)::r,c
real(kind = 16),dimension(:),intent(in)::Xi,Xj
real(kind = 16)::dif,derxi

dif = delta(Xi,Xj,k)
    derxi = r*r + c*c
    dif = (3.q0 * dif * dif) - derxi
    dif = dif/(derxi * derxi)
    derxi = IMQ(r,c)
```

```
    derxi = derxi * dif

end function der_2_xi_IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! delta
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function delta(Xi,Xj,k) result(dx)

integer,intent(in)::k
real(kind = 16),dimension(:),intent(in)::Xi,Xj
real(kind = 16)::dx

    dx = Xi(k) - Xj(k)

end function delta

end module
```

B.1.2. Módulo para el cálculo de la matriz de Gram

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!The functions are

! gram_IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Parameters

! c = Shape parameter
! X = Vector node
! r = Distance between Xi and Xj nodes
! N = Number of nodes

module gram_matrix_RBF

use RBF_kernels

contains

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!gram_IMQ
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function gram_IMQ(X,N,c) result(PHI)

integer,intent(in)::N
real(kind = 16),intent(in)::c
real(kind = 16),dimension(:,:),intent(in)::X
integer::i,j
real(kind = 16)::r,PHI(N,N)

    PHI(1,1) = 1.q0 / c

    do i = 2,N
        PHI(i,i) = 1.q0 / c
```

```
        do j = 1,i-1

        r = sqrt(dot_product(X(i,:) - X(j,:),X(i,:) - X(j,:)))
            PHI(i,j) = IMQ(r,c)
            PHI(j,i) = PHI(i,j)

        enddo
    enddo

end function gram_IMQ

end module
```

B.1.3. Módulo para el cálculo del vector de derivadas $(\mathfrak{L}\Phi)_c$

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Parameters

! c = Shape parameter
! M = Number of local nodes
! r = Distance between Xi and Xj
! X = Vector node
! X_central = Central node
! dim_x = Dimension (R^n)
! i0 = Component of derivation

module vector_RBF_FD

use RBF_kernels

contains

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Vector Dx_{i} Operator IMQ
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function vect_Dxi_IMQ(X,x_central,i0,M,c,dim_x) result(vect)

integer,intent(in)::M,dim_x,i0
real(kind = 16),intent(in)::c
real(kind = 16),dimension(:),intent(in)::x_central
real(kind = 16),dimension(:,:),intent(in)::X
integer::i,j
real(kind = 16)::r
real(kind = 16),dimension(:)::vect(M)

  !Initial value
  vect = 0.q0

  !Differential Operator
```

```
  do i = 1,M
    r = sqrt(dot_product(X(i,:) - x_central,X(i,:) - x_central))
    vect(i) = der_xi_IMQ(r,c,X(i,:),x_central,i0)
  enddo

end function vect_Dxi_IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Vector D2x_{i} Operator IMQ
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function vect_D2xi_IMQ(X,x_central,i0,M,c,dim_x) result(vect)

integer,intent(in)::M,dim_x,i0
real(kind = 16),intent(in)::c
real(kind = 16),dimension(:),intent(in)::x_central
real(kind = 16),dimension(:,:),intent(in)::X
integer::i,j
real(kind = 16)::r
real(kind = 16),dimension(:)::vect(M)

  !Initial value
  vect = 0.q0

  !Differential Operator
  do i = 1,M
    r = sqrt(dot_product(X(i,:) - x_central,X(i,:) - x_central))
    vect(i) = der_2_xi_IMQ(r,c,X(i,:),x_central,i0)
  enddo

end function vect_D2xi_IMQ

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Vector Lapalcian Operator IMQ
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function vect_laplacian_IMQ(X,x_central,M,c,dim_x) result(vect)

integer,intent(in)::M,dim_x
```

```
real(kind = 16),intent(in)::c
real(kind = 16),dimension(:),intent(in)::x_central
real(kind = 16),dimension(:,:),intent(in)::X
integer::i,j
real(kind = 16)::r
real(kind = 16),dimension(:)::vect(M)

  !Initial value
  vect = 0.q0

  !Differential Operator
  do i = 1,M
    r = sqrt(dot_product(X(i,:) - x_central,X(i,:) - x_central))
    do j = 1,dim_x
        vect(i) = vect(i) + der_2_xi_IMQ(r,c,X(i,:),x_central,j)
    enddo
  enddo

end function vect_laplacian_IMQ

end module
```

B.2. Módulos para sistemas lineales

B.2.1. Módulo para la solución a sistemas lineales por métodos directos

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Notes

! The system is of the form
! Ax = b
! And "N" is the size of the matrix A

module direct_methods

contains

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Thomas Algorithm (TDMA)
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function TDMA(A,b,N)  result(x)

integer,intent(in)::N
real(kind = 16),dimension(:),intent(in)::b(N)
real(kind = 16),dimension(:,:),intent(in)::A(N,3)
integer::i
real(kind = 16),dimension(:)::x(N),ci(N),di(N)

 !Coefficients
 ci(1) = A(1,3) / A(1,2)
 di(1) = b(1) / A(1,2)

 do i = 2,n-1
  ci(i) = A(i,3)/(A(i,2) - ci(i - 1)*A(i,1))
    di(i) = (b(i) - di(i - 1)*A(i,1))/(A(i,2) - ci(i - 1)*A(i,1))
 enddo

 di(N) = (b(N) - di(N - 1)*A(N,1))/(A(N,2) - ci(N - 1)*A(N,1))
```

```
 !Solution
 x(N) = di(N)

 do i = N-1,1,-1
  x(i) = di(i) - ci(i) * x(i + 1)
 enddo

end function TDMA


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! LU Descomposition Version (Gilbert-Peierls) Total
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
subroutine LU_descomposition_gilbert_peierls_total(A,N,LU)

integer,intent(in)::N
real(kind = 16),dimension(:,:),intent(in)::A(N,N)
real(kind = 16),dimension(:,:),intent(out)::LU(N,N)
integer::i,j
real(kind = 16),dimension(:)::s(N)

 !Initial data
 do i = 1,N
  LU(i,i) = 1.q0
 enddo

 !Step one
 s = A(:,1)
 LU(1,1) = s(1)
 LU(2:N,1) = s(2:N)/LU(1,1)

 !Step two
 do i = 2,N
  s = A(:,i)
  do j = 1,i-1
  s(j+1:N) = s(j+1:N) - LU(j+1:N,j)*s(j)
enddo
```

```
    LU(1:i,i) = s(1:i)
    LU(i+1:N,i) = s(i+1:N)/LU(i,i)
 enddo

end subroutine LU_descomposition_gilbert_peierls_total


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Inverse Matrix with LU Gilbert-Peierls Total
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
function inverse_LU_gilbert_peierls_total(A,N)    result(A_1)

integer,intent(in)::N
real(kind = 16),dimension(:,:),intent(in)::A(N,N)
integer::k
real(kind = 16),dimension(:)::x(N),y(N),e(N)
real(kind = 16),dimension(:,:)::LU_A(N,N),A_1(N,N)

 !LU descomposition matrices
 call LU_descomposition_gilbert_peierls_total(A,N,LU_A)

 !Initial data
 e = 0.q0

 do k = 1,N

  !Canonical vector
y = 0.q0
    e(k) = 1.q0

!Forward sustitution
y(k:N) = forward_sustitution_doolittle(LU_A(k:N,k:N), &
    & e(k:N),N - k + 1)

!Backward sustitution
A_1(:,k) = backward_sustitution(LU_A,y,N)
```

```
 enddo

end function inverse_LU_gilbert_peierls_total
```

B.2.2. Módulo para el paso de SOR con almacenamiento ralo (Formato CSR)

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Notes

! The storage of the matrix operator is CSR

! Nt = Dimension of the matrix
! Nnz = Number of entries no zero
! w = Parameter for SOR
! D_CSR = Elements of the matrix in format CSR
! I_CSR = Index of the matrix in format CSR
! point_CSR = Pointer of the matrix in format CSR
! index_diag = Pointer of the diagonal element in I_CSR

module SOR_step_LDC

contains


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! SOR Step
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
subroutine SOR_step_RBF_FD_lid_driven_cavity_2D_stationary(Nt,&
   & Nnz,w,point_CSR,I_CSR,D_CSR,index_diag,b,Y_prev,Y)

integer,intent(in)::Nt,Nnz
integer,dimension(:),intent(in)::point_CSR(Nt + 1),I_CSR(Nnz)
integer,dimension(:),intent(in)::index_diag(Nt)
real(kind = 8),intent(in)::w
real(kind = 8),dimension(:),intent(in)::D_CSR(Nnz),b(Nnz)
real(kind = 8),dimension(:),intent(in)::Y_prev(Nt)
real(kind = 8),dimension(:),intent(out)::Y(Nt)
integer::i,j,k1,k2,k3
real(kind = 8)::aux
```

```
    do i = 1,Nt
        k1 = point_CSR(i);    k2 = index_diag(i)
        k3 = point_CSR(i+1) - 1
        aux = dot_product(D_CSR(k1:k2 - 1),Y(I_CSR(k1:k2-1)))
        Y(i) = b(i) - aux
        aux = dot_product(D_CSR(k2 + 1:k3), &
         & Y_prev(I_CSR(k2 + 1:k3)))
        Y(i) = (Y(i) - aux)/D_CSR(k2)
  Y(i) = w*Y(i) + (1.0d0 - w)*Y_prev(i)
enddo

end subroutine

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Index for diagonal elements in SOR
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
subroutine pointers_SOR_step_RBF_FD_lid_driven_cavity_2D_ &
   & stationary(Nt,Nnz,point_CSR,I_CSR,index_diag)

integer,intent(in)::Nt,Nnz
integer,dimension(:),intent(in)::point_CSR(Nt + 1),I_CSR(Nnz)
integer,dimension(:),intent(out)::index_diag(Nt)
integer::i,j

    ! Process
    do i = 1,Nt
        do j = point_CSR(i),point_CSR(i + 1) - 1

            if (I_CSR(j) == i)   then
                index_diag(i) = j
                exit
            endif

        enddo
    enddo

end subroutine
```