



UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

Mat. Luis Manuel Rivera Gutiérrez

SOLUCIÓN NUMÉRICA DE LA ECUACIÓN DE ADVECCIÓN EMPLEANDO MALLAS ESTRUCTURADAS SOBRE REGIONES PLANAS IRREGULARES UTILIZANDO UN ESQUEMA DE DIFERENCIAS FINITAS.

TESIS

QUE PARA OBTENER EL TÍTULO DE

Licenciado en Ciencias Físico Matemáticas

PRESENTA

Gerardo Tinoco Guerrero

ASESORES:

Doctor en Matemáticas José Gerardo Tinoco Ruiz

Doctor en Cs. Físico Matemáticas Francisco Javier Domínguez Mota

Morelia Michoacán., Abril 2014

A mis padres,

que a lo largo de la vida me han dado todo.

Agradecimientos

A todos y a nadie, a quien estuvo y a quien faltó...

Agradecimientos 2.0

¿De verdad me creen tan mala onda como para dejar esos agradecimientos?

A lo largo de toda mi trayectoria académica he contado con el apoyo de muchas personas, algo por lo que estoy profundamente agradecido. Pero me gustaría dar un agradecimiento especial a mi asesores los Dr. José Gerardo Tinoco Ruiz y Francisco Javier Dominguez Mota. Ambos me han tenido mucha paciencia a lo largo de estos últimos años, y gracias a ellos es que se ha podido desarrollar este trabajo. También me han permitido conocer muchos lugares del mundo y han contribuido en mi crecimiento, tanto profesional como personal.

A la Universidad Michoacana de San Nicolás de Hidalgo y la Facultad de Ciencias Físico Matemáticas “Mat. Luis Manuel Rivera Gutiérrez”. No importa a donde llegue después, no importa donde estudie más tarde, siempre seré Nicolaíta de Corazón.

Gracias al Dr. Mario César Suárez Arriaga y al M.I. Cuauhtémoc Rivera Loaiza, quienes a lo largo de la carrera me han compartido su amistad, sus libros, sus consejos, y alguna que otra aplicación para el celular (sí Temo, estoy hablando de ti).

A mis profesoras, maestras y cómplices de crímenes Azucena Chávez González y Karina Mariela Figueroa Mora. Me enseñaron muchas cosas, de matemáticas, de computación y de la vidas. Sus enseñanzas viajarán conmigo hasta donde llegue.

El más grande de mis agradecimientos es para las personas a quienes he dedicado esta tesis: Mis padres, Ma. de Lourdes Guerrero Magaña y José Gerardo Tinoco Ruiz. A lo largo de la vida me han dado todo lo que necesito para se-

guir adelante, y muchas cosas que no necesitaba, pero que quería, también. El agradecimiento más grande es para ustedes, que a lo largo de toda mi vida han estado apoyando en cada etapa, consintiéndome y regañándome, según lo merezca. Sin ustedes nunca habría podido llegar hasta donde he llegado.

A mis hermanos Narciso Salvador y Luis Manuel Tinoco Guerrero, que aunque a veces nos sacamos canas unos con otros, quiero agradecerles por ser una gran apoyo para mi; ayudándome desde cosas como comprar un refresco hasta ayudarme con la programación de algunas cosas.

A mis otros 2 hermanos: Jorge Armando Tinoco Michel y David Iván Tinoco Ruiz. Aunque no son mis hermanos realmente los quiero como tales, han estado ahí conmigo desde que me acuerdo, siempre regalándome buenos momentos en el billar o con el chino (ustedes entienden).

Si ha habido personas quienes siempre me han consentido y cuidado, que logran parecer un millón de personas siendo solo cuatro, esos serían mis abuelos. Tanto los que aún están conmigo como los que partieron, han sido como mis padres a lo largo de toda mi vida, me han cuidado, enseñado, educado, amado y más. NUNCA podré pagar todo lo que han hecho por mi, ya que siempre han sido de mis grandes apoyos.

En el tiempo que he vivido, en el tiempo que he estudiado he conocido a mucha gente, pero hay algunos que merecen una mención aquí por su apoyo de siempre.

A quien fue mi pareja durante la mayor parte de este viaje: Fernanda, que aunque a la fecha no seguimos juntos supo aguantarme y me apoyarme a lo largo de la elaboración de este trabajo. Gracias de todo corazón.

A mis amigos Audomaro, Ricardo, Luis Antonio y Alejandro, tardes largas de cafés y/o cerveza, burlándonos de todo y de todos, incluyendo mi trabajo de tesis. Gracias por escuchar siempre mis problemas, contarme los suyos y siempre apoyarnos en cada momento, desde mucho antes de iniciar este trabajo.

A mis amigas Ma. Márgara de la Virgen del Perpetuo Socorro Gonzales de Gonzales y Gonzales, Marianita, Sammy, Sir Kathe, Fabiola, Andrea, Xiomara y Tanya, todas y cada una de ustedes me han dado mucha felicidad en muchos momentos, sobre todo cuando les cuento mis problemas amorosos y siempre me dan la razón a mi, las adoro por eso.

A mis compañeros, colegas y amigos Yunix, Belem, Erika, Geraldine, Moni-Moon, Pablo V., Pablo M., Fernando, Sanzón. Hemos compartido con todos más de una taza de café, hemos compartido chismes, sueños, congresos, viajes y, por muy difícil que sea de creer, hasta trabajábamos a veces.

[Lucy no me olvidé de ti, pero quedamos que te mencionaríamos en párrafos

separados. No queremos que se mezcle por ahí la maldad.]

Aparte de mis estudios básicos he mantenido algunos cursos extra curriculares, he aprendido otro idioma y esto me ha abierto muchas puertas. Gracias Alejandra Bravo Pérez y José Serrato Mendoza; los mejores maestros que se puede tener para aprender el Francés. Me abrieron las puertas de una nueva cultura, eso es algo que simplemente no se puede olvidar.

Por último, pero no por eso menos importante, quiero agradecer al ya famoso Don Pepe, de los tacos de birria 1×1 . Ir a almorzar ahí hacía que mis días mejoraran, me curaron más de una gripa y más de una tristeza. Gracias por todo su esfuerzo y trabajo.

Quiero agradecer a la academia, por permitirme ganar este premio cuando Leonardo Di Caprio no lo ha podido ganar. Simple y sencillamente no pude evitar el comentario.

No podía olvidar a las personas que nunca leerán esto, pero gracias a ellos decidí llegar hasta aquí: Al señor Paul Zaloom (alias Beakman) que gracias a el gané un extraño gusto por la ciencia; al Dr. Emmet Brown quien sigo esperando aparezca en su DeLorean; al Dr. House quien sin darse cuenta ni conocerme me ayudó a superar algunos problemas personales; a los doctores Sheldon Cooper y Leonard Hofstadter que, a pesar de ser físicos, me dieron muchos momentos de risa y gracias a ellos aprendí algunas cosas de su área.

Resumen

La ecuación de advección es un bloque muy importante en el proceso de modelado de problemas relacionados con aguas someras en lagos y estanques, sin embargo, a la fecha existen pocos esquemas de solución que sean fácilmente aplicables en regiones no rectangulares, y esto representa una seria limitación para éstos.

En los años recientes se han diseñado varios esquemas basados en diferencias finitas para la aproximación de diferentes Ecuaciones Diferenciales Parciales (EDP), en regiones planas que cumplan con ser no rectangulares y asimétricas.

Para poder trabajar con este tipo de regiones es necesario contar con mallas que cumplan con ciertas características geométricas deseables: que la distribución de las áreas de las celdas que las forman sea lo más uniforme posible, que los lados de cada celda sean lo más ortogonales posibles o que la variación de las líneas coordenadas no tenga cambios bruscos.

En este trabajo hacemos una breve revisión de los métodos de generación numérica de mallas, se revisan los métodos diseñados por Barrera-Sánchez y Tinoco-Ruiz [1, 4, 28] y se menciona el método diseñado por Domínguez-Mota [11] para la generación de mallas en diferentes regiones, que van desde regiones con geometría sencilla hasta regiones que son aproximaciones a lugares geográficos reales.

Tras hacer esta revisión, se presentan algunos de los métodos existentes de la solución de la ecuación de advección en $1 + 1D$ y en $2 + 1D$; Se proponen tres esquemas de diferencias para aproximar dicha ecuación en $2 + 1D$, se explica la manera como se diseñan éstos y se hace un estudio de la calidad de las aproximaciones logradas con dichos esquemas, mostrando la buena exactitud que se obtiene, sobre todo con el esquema de 6 puntos.

Palabras Clave: *diferencias finitas, regiones irregulares, ecuación de advección, métodos numéricos.*

Abstract

The advection equation is a very important block in the process of modeling shallow-water problems in lakes and ponds, however, to this day there are very few schemes of solution that can be easily applied in non-rectangular regions, and this is a serious limitation for the method.

In recent years many finite difference schemes have been designed for the numerical approximation of different Partial Differential Equations, in plane irregular regions that comply with be non-rectangular and non-symmetrical.

To be able to work with these kind of regions it is necessary to count with grids with certain desired geometrical characteristics: The distribution of the cell areas, which forms the grid, is as uniform as possible; the sides of every cell must also be as ortogonal as possible, and the variation of the coordinate lines doesn't have abrupt changes.

In this work we do a brief revision of numerical grid generation methods, we show the methods designed by Barrera-Sánchez and Tinoco-Ruiz [1, 4, 28] and we mention the method designed by Dominguez-Mota [11] for the numerical grid generation at different regions; from regions with simply geometry to regiones which are approximation to real geographical locations.

After this revision, we present the existent methods for the solution of the advection equations in $1 + 1D$ and $2 + 1D$; we present three finite-difference schemes to approximate the advection equation in $2 + 1D$, and we explain the way how they were designed, also we make a study of the quality of the approximations accomplished with this schemes, showing the precision we obtain, specially with the 6 points scheme.

Keywords: *finite difference, irregular regions, advection equations, numerical methods.*

Prefacio

En la literatura se pueden encontrar gran variedad trabajos que tratan sobre la solución de ecuaciones diferenciales parciales, ya sea de manera exacta por métodos analíticos, o de manera aproximada por métodos numéricos. Desafortunadamente no existen soluciones analíticas para todas las ecuaciones diferenciales parciales, y es ahí donde radica la importancia de los métodos numéricos.

Existen muchos fenómenos en la naturaleza que se pueden describir y aproximar por medio de ecuaciones diferenciales, tanto parciales como ordinarias, pero por desgracia los métodos comunes de aproximación están basados en geometrías regulares o muy sencillas, por lo que su aplicación es limitada cuando se trata de regiones más complicadas, como las que existen en la naturaleza. Buscando contar con esquemas que modelen los fenómenos para los cuales no se cuenta con la solución analítica de sus ecuaciones diferenciales se utilizan aproximaciones que sean lo suficientemente cercanas a la realidad. Así es como surgen los métodos numéricos, que nos sirven para contar con esquemas de aproximación a las soluciones de estos problemas.

Se han diseñado muchos métodos para aproximar la solución de una ecuación diferencial parcial, los más usados son los de diferencias finitas, elementos finitos, métodos espectrales, volúmenes finitos y elementos de frontera, entre otros similares. En este trabajo sólo nos centraremos en el método de diferencias finitas, el cual consiste en discretizar el dominio dado en donde se quiere aproximar la solución de cierta ecuación diferencial parcial u ordinaria, reemplazando las derivadas de las ecuaciones diferenciales por una combinación lineal de valores de la función en cada uno de los nodos de la malla. Como resultado, obtenemos un sistema de ecuaciones algebraicas, usualmente grande, que se resuelve en lugar de las ecuaciones diferenciales. La solución a este sistema es una aproximación de la solución de la ecuación diferencial en cada nodo.

Una de las mayores motivaciones para el uso de los métodos de aproximación es el hecho de que, gracias a ellos, el problema original puede convertirse en un problema que resulta más sencillo de resolver.

En este trabajo se utiliza un esquema basado en diferencias finitas. Los esquemas de diferencias finitas son relativamente fáciles de formular debido a que no

se necesitan matemáticas muy avanzadas para hacerlo, únicamente es necesario conocer sobre series de Taylor y Álgebra Lineal. El esquema se usa para obtener una aproximación a la solución del problema de advección en regiones que no cuentan con una frontera con forma regular, como un rectángulo, sino que su frontera cuenta con una complejidad mayor.

En el primer capítulo se explica de manera básica los conceptos a tratar durante la tesis, se muestra sobre las ecuaciones diferenciales y los diferentes métodos existentes a la fecha para encontrar aproximaciones a la solución de dichas ecuaciones. Se hace un estudio somero sobre los esquemas de diferencias finitas y se da una ligera introducción a los diferentes tipos de Ecuaciones Diferenciales Parciales.

En el capítulo dos se realiza una descripción panorámica de generación numérica de mallas, se muestran algunos de los diferentes métodos de generación, tanto continua como discreta, y se agrega una rápida comparación de los diferentes métodos.

El tercer capítulo trata sobre el problema central de esta tesis, la aproximación a la solución del problema de advección. Se muestran los diferentes esquemas existentes para su aproximación en $1+1D$ y se explica su generalización a $2+1D$. Se muestran los esquemas propuestos en este trabajo junto con su utilización para el problema de advección.

El capítulo cuatro contiene los resultados numéricos de las diferentes pruebas realizadas con cada uno de los esquemas descritos en el capítulo tres. También se agrega, a manera de conclusiones, un breve estudio sobre la calidad de las aproximaciones obtenidas.

Índice general

Resumen	XIII
Abstract	XV
Prefacio	XVII
1. Esquemas de Diferencias Finitas en Regiones Rectangulares	1
1.1. Ecuaciones Diferenciales Parciales (EDP)	1
1.1.1. Condiciones Iniciales y de Frontera	2
1.2. Diferencias Finitas en una Dimensión	3
1.2.1. Aproximación por Serie de Taylor	4
1.3. Esquemas de Diferencias en Regiones Bidimensionales Rectangulares	6
1.3.1. Consistencia	10
1.3.2. Estabilidad	12
1.3.3. Convergencia	12
2. Generación numérica de mallas	17
2.1. ¿Qué es una Malla?	17
2.2. Generación Numérica de Mallas Estructuradas	18
2.2.1. El Método de Interpolación Transfinita	21
2.3. Generación Variacional Continua	22
2.3.1. Introducción	24
2.3.2. Funcional de Longitud	25
2.3.3. Funcional de Área	27
2.3.4. Funcional de Ortogonalidad	29
2.3.5. Combinación de Funcionales	29
2.3.6. Funcional de Suavidad	31
2.4. Generación Variacional Discreta	35
2.4.1. Funcional Discreto de Longitud	36
2.4.2. Funcional Clásico de Área en su Versión Discreta	37
2.4.3. Funcional Discreto de Ortogonalidad	38
2.4.4. Funcionales Combinados	40
2.5. Funcionales de Suavidad y Área adaptivos de Tinoco	43

2.5.1. El Funcional de Suavidad Adaptiva	46
2.5.2. El Funcional de Área Adaptiva	46
2.5.3. Resultados	47
2.5.4. Combinación de Funcionales	50
2.6. Funcional Quasi-Armónico de Domínguez	54
2.7. Conclusiones	54
3. Ecuación de Advección	57
3.1. Advección en $1 + 1D$	57
3.1.1. El método Upwind en $1 + 1D$	59
3.1.2. El esquema FTCS en $1 + 1D$	60
3.1.3. El Esquema de Lax-Friedrichs	61
3.1.4. El Esquema Leapfrog en $1 + 1D$	63
3.1.5. El Esquema de Lax-Wendroff en $1 + 1D$	64
3.2. Advección en $2 + 1D$	65
3.2.1. El Caso Rectangular	66
3.2.2. El Caso en Regiones no Rectangulares	67
3.3. Advección en $2 + 1D$: Esquema de 4 Puntos	70
3.4. Advección en $2 + 1D$: Esquema de 6 Puntos	72
4. Resultados y conclusiones	75
4.1. Resultados	75
4.1.1. Esquemas en $1 + 1D$	75
4.1.2. Esquemas en $2 + 1D$: Lax-Wendroff	85
4.1.3. Esquemas en $2 + 1D$: Esquema Explícito de 4 Puntos	89
4.1.4. Esquemas en $2 + 1D$: Esquema Implícito de 4 Puntos	90
4.1.5. Esquemas en $2 + 1D$: Esquema de 6 Puntos	97
4.2. Conclusiones	110
Bibliografía	111

Índice de figuras

1.1. Mallado uniforme en el intervalo $[a, b] \times [c, T]$	7
1.2. Mallado uniforme en el intervalo $[0, 1]$	8
1.3. Mallado uniforme en el dominio espacio-tiempo	9
2.1. Malla uniforme en 1 dimensión.	18
2.2. Malla no uniforme en 1 dimensión.	18
2.3. Malla uniforme en 2 dimensiones.	19
2.4. Malla no uniforme en 2 dimensiones.	19
2.5. Transformación $U_2 \rightarrow \Omega$	20
2.6. Region dada por cuatro ecuaciones paramétricas.	22
2.7. Mallas generadas con TFI para las regiones m13 y m19.	23
2.8. Mallas generadas con TFI para las regiones ENG y HAB.	23
2.9. Mallas generadas usando el funcional continuo de longitud para las regiones m13 y m19.	26
2.10. Mallas generadas usando el funcional continuo de longitud para las regiones ENG y HAB.	27
2.11. Mallas generadas usando el funcional continuo de área para CUA y TRA.	28
2.12. Mallas generadas usando el funcional continuo de área para CUA y TRA.	28
2.13. Mallas generadas usando el funcional continuo de área-longitud para M13 y M19	30
2.14. Mallas generadas usando el funcional continuo de área-ortogonalidad para M13 y M19.	32
2.15. Mallas generadas usando el funcional continuo de área-ortogonalidad para ENG y HAB.	32
2.16. Mallas generadas usando el funcional continuo de suavidad para las regiones m13 y m19.	34
2.17. Mallas generadas usando el funcional continuo de suavidad para las regiones ENG y HAB.	34
2.18. Formación de los triángulos relevantes.	36
2.19. Mallas generadas usando el funcional discreto de longitud para M13 y M19.	37

2.20. Mallas generadas usando el funcional discreto de longitud para ENG y HAB.	38
2.21. Mallas generadas usando el funcional discreto de área para M13 y M19.	39
2.22. Mallas generadas usando el funcional discreto de área para ENG y HAB.	39
2.23. Triángulo con vértices c, a, b y ángulo Θ	40
2.24. Mallas generadas usando el funcional discreto de área-longitud para M13 y M19.	42
2.25. Mallas generadas usando el funcional discreto de área-longitud para ENG y HAB.	42
2.26. Mallas generadas usando el funcional discreto de área-Ortogonalidad para M13 y M19.	44
2.27. Mallas generadas usando el funcional discreto de área-Ortogonalidad para ENG y HAB.	44
2.28. Trayectoria hacia una malla convexa.	46
2.29. Mallas generadas usando el funcional suavidad para M13 y M19.	48
2.30. Mallas generadas usando el funcional de suavidad para ENG y HAB.	48
2.31. Mallas generadas usando el funcional de área para M13 y M19.	49
2.32. Mallas generadas usando el funcional de área para ENG y HAB.	49
2.33. Mallas generadas usando el funcional adaptivo de área-longitud para M13 y M19.	51
2.34. Mallas generadas usando el funcional adaptivo de área-longitud para ENG y HAB.	52
2.35. Mallas generadas usando el funcional Suavidad-Área $F_{SA} = \tau F_k + (1 - \tau)F_{A^2}$ para M13 y M19.	53
2.36. Mallas generadas usando el funcional Suavidad-Área $F_{SA} = \tau F_k + (1 - \tau)F_{A^2}$ para ENG y HAB.	53
2.37. Mallas generadas usando el funcional $H_\omega(G)$	55
2.38. Mallas generadas usando el funcional $H_\omega(G)$	56
3.1. Un ejemplo de un stencil tipo Upwind	59
3.2. Un ejemplo de un stencil para el esquema de Lax-Friedrichs	62
3.3. Un ejemplo de un stencil para el esquema de Leapfrog.	64
3.4. Distribución arbitraria de p_0 y sus vecinos.	68
3.5. Stencil tipo “upwind”.	71
3.6. Diferentes estenciles del esquema de 6 puntos.	73
4.1. Comparación de resultados método upwind en $t = 5$	78
4.2. Comparación de resultados método FTCS en $t = 5$	79
4.3. Comparación de resultados método Lax-Friedrichs en $t = 5$	81
4.4. Comparación de resultados método Lax-Friedrichs en $t = 5$	83
4.5. Comparación de resultados método Leapfrog en $t = 5$	84
4.6. Comparación de resultados método de Lax-Wendroff en $t = 5$	86
4.7. Dominio Ω rectangular con $\partial\Omega = S_1 \cup S_2$	87

4.8. Comparación de resultados método de Lax-Wendroff en $2 + 1D$.	88
4.9. Dominio Ω irregular con $\partial\Omega = S_1 \cup S_2$.	90
4.10. Comparación de resultados del esquema de 4 puntos en el rectángulo $[0, 1] \times [0, 1]$.	92
4.11. Comparación de resultados del esquema de 4 puntos en CAB.	93
4.12. Comparación de resultados del esquema de 4 puntos en HA.	94
4.13. Comparación de resultados del esquema de 4 puntos en MIC.	95
4.14. Comparación de resultados del esquema de 4 puntos en UC.	96
4.15. Comparación de resultados del esquema de 4 puntos implícito en el rectángulo $[0, 1] \times [0, 1]$.	98
4.16. Comparación de resultados del esquema de 4 puntos en CAB.	100
4.17. Comparación de resultados del esquema de 4 puntos en HA.	101
4.18. Comparación de resultados del esquema de 4 puntos en MIC.	102
4.19. Comparación de resultados del esquema de 4 puntos en UC.	103
4.20. Comparación de resultados del esquema de 6 puntos en el rectángulo $[0, 1] \times [0, 1]$.	104
4.21. Comparación de resultados del esquema de 4 puntos en CAB.	106
4.22. Comparación de resultados del esquema de 4 puntos en HA.	107
4.23. Comparación de resultados del esquema de 4 puntos en MIC.	108
4.24. Comparación de resultados del esquema de 6 puntos en UC.	109

Capítulo 1

Esquemas de Diferencias Finitas en Regiones Rectangulares

El propósito de este capítulo es dar una breve introducción a los esquemas de diferencias finitas, en una y dos dimensiones, en regiones simples. Esto también servirá para introducir los conceptos y la notación básica usada a lo largo del presente trabajo.

1.1. Ecuaciones Diferenciales Parciales (EDP)

Una ecuación en derivadas parciales es una relación entre una función de dos o más variables y algunas de sus derivadas parciales. El *orden de la EDP* se define como el mayor orden de las derivadas que están presentes en la ecuación. Las ecuaciones diferenciales que simulan fenómenos físicos comúnmente se separan en grupos dependiendo del tipo de las relaciones existentes entre las derivadas que se involucran.

Una EDP lineal de segundo orden tiene la forma:

$$Au_{xx}^2 + Bu_{xy}^2 + Cu_{yy}^2 + Du_x + Eu_y + F = 0 \quad (1.1)$$

en donde A , B , C , D , E y F son coeficientes que pueden depender tanto de x como de y pero no de la función. Las ecuaciones diferenciales parciales se pueden clasificar en *hiperbólicas*, *parabólicas* y *elípticas*; la manera más usada para saber a que grupo pertenece una EDP consiste en obtener el determinante de una matriz cuyos coeficientes son los coeficientes A , B y C de su forma (1.1) [25].

Una ecuación es *hiperbólica* si

$$Z = \det \begin{vmatrix} A & B \\ B & C \end{vmatrix} < 0,$$

es *parabólica* si

$$Z = \det \begin{vmatrix} A & B \\ B & C \end{vmatrix} = 0,$$

es *elíptica* si

$$Z = \det \begin{vmatrix} A & B \\ B & C \end{vmatrix} > 0.$$

Las ecuaciones hiperbólicas modelan el transporte y movimiento de alguna cantidad física, ya sea perturbaciones o flujo de fluidos. Uno de los ejemplos más utilizados es la ecuación de onda en una dimensión

$$\frac{\partial^2 \varphi}{\partial t^2} = c^2 \frac{\partial^2 \varphi}{\partial x^2},$$

en donde, si ordenamos los términos, podemos ver que $A = 1$, $B = 0$, $C = -c^2$; al obtener el determinante de la matriz generada por estos coeficientes podemos notar que $Z < 0$.

Por su parte, las ecuaciones parabólicas describen fenómenos que dependen del tiempo y que llevan a un estado de equilibrio. Uno de los ejemplos más encontrados en la literatura es la ecuación que describe el calentamiento de una barra a lo largo de uno de sus ejes

$$\frac{\partial \varphi}{\partial t} = \alpha \frac{\partial^2 \varphi}{\partial x^2},$$

en la cual $A = 0$, $B = 0$ y $C = \alpha$, y podemos ver que $Z = 0$.

Por último las ecuaciones elípticas están asociadas con un estado especial del sistema el cual corresponde a un estado de mínima energía. Por ejemplo, en la ecuación de Poisson

$$\frac{\partial^2 \varphi}{\partial x^2} + \frac{\partial^2 \varphi}{\partial y^2} = f$$

en la cual $A = C = 1$ y $B = 0$, por lo que $Z = 1$.

1.1.1. Condiciones Iniciales y de Frontera

Un factor muy importante al momento de resolver diferentes EDP's son las llamadas condiciones iniciales y de frontera. En el caso en que trabajamos con Ecuaciones Diferenciales Ordinarias (EDO's) podemos recurrir al teorema fundamental de existencia y unicidad, el cual nos garantiza que existe una solución a nuestro problema y que, además, es única si se da un conjunto de condiciones que la función debe de cumplir. Ya que sabemos que la solución general viene dada por una familia de funciones parametrizadas constantes, nos podemos

apoyar de las condiciones de frontera para especificar dichas constantes.

A diferencia de las EDO's, cuando se trata de una EDP no existe un teorema fundamental con el que podamos justificar la existencia y unicidad en general de las soluciones. En su lugar existen diversas teorías independientes que son usadas para cada uno de los tipos de EDP's con los cuales se trabaja comúnmente, ya sean lineales como la ecuación de Laplace, la ecuación de calor o la ecuación de onda; o no lineales como la ecuación de Burgers o la ecuación de Monge-Ampère.

Las condiciones que tendremos (o necesitaremos) dependerán del tipo de ecuación con la que trabajemos, en el caso de las EDP's parabólicas e hiperbólicas se suelen dar condiciones iniciales y de frontera; un ejemplo de esto es la distribución inicial de temperaturas y los valores de ésta en los extremos de una barra en el caso de la ecuación de calor. Mientras que en el caso de las ecuaciones elípticas se acostumbra dar condiciones principalmente de frontera.

Existen fundamentalmente 2 tipos de condiciones en la frontera: se dice que las condiciones de frontera son de tipo *Dirichlet* si se da el valor de la función en la frontera; de *Neumann* si se dan los valores de la derivada en la frontera y, por último, cuando se da una combinación lineal de ambos valores en la frontera, las condiciones se llamarán de *Robin*.

1.2. Diferencias Finitas en una Dimensión

Cuando trabajamos con Diferencias Finitas, la idea principal es reemplazar las derivadas de las ecuaciones diferenciales por combinaciones lineales de los valores de la función. Para hacer esto recordemos la definición de derivada en una dimensión por medio del uso de límites: Consideremos la función $u(x)$ definida en el intervalo abierto (a, b) , entonces la definición de la derivada de u en x es

$$\begin{aligned}\frac{du}{dx} &\equiv \lim_{h \rightarrow 0} \frac{u(x+h) - u(x)}{h}, \\ &\equiv \lim_{h \rightarrow 0} \frac{u(x+h) - u(x)}{(x+h) - x}.\end{aligned}\quad (1.2)$$

De esta definición podemos obtener “fácilmente” un esquema de diferencias finitas en $1D$ siguiendo algunos sencillos pasos. Primero, dividimos el intervalo $[a, b]$, en donde está definida la función, en n subintervalos, creando así una malla con $n+1$ nodos $a = x_0, x_1, \dots, x_n = b$. Aplicando esta discretización del dominio a (1.2) tenemos, si $x_{i+1} - x_i$ no es muy grande, para cada nodo interior

$$\frac{du}{dx}(x_i) \approx \frac{u(x_{i+1}) - u(x_i)}{(x_{i+1}) - x_i},$$

si reescribimos $u_{i+1} = u(x_{i+1})$ y $u(x_i) = u_i$ obtenemos

$$\frac{du}{dx}(x_i) \approx \frac{u_{i+1} - u_i}{x_{i+1} - x_i}.$$

Así hemos obtenido una expresión, basada en diferencias finitas, que aproxima a la derivada de u en x_i . A esta expresión se le conoce como *diferencias hacia delante*, debido a que involucra a los puntos i e $i + 1$. De una manera similar podemos llegar a las expresiones

$$\frac{du}{dx}(x_i) = \frac{u_i - u_{i-1}}{x_i - x_{i-1}},$$

que se conoce como *diferencias hacia atrás*, y

$$\frac{du}{dx}(x_i) = \frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}},$$

que es la expresión de las llamadas *diferencias centradas*.

1.2.1. Aproximación por Serie de Taylor

Los esquemas en diferencias finitas pueden ser obtenidos, de una manera más rigurosa, usando como herramienta la *expansión en serie de Taylor* para expandir $u(x)$.

Teorema 1 (Taylor) *Sea $u(x)$ una función continua en el intervalo $[a, b]$ la cual tiene derivadas continuas hasta de orden N sobre (a, b) ; dado un punto x_0 en (a, b) , entonces para cualquier otro punto x en (a, b)*

$$u(x) = u(x_0) + \frac{du(x_0)}{dx}(x - x_0) + \frac{d^2u(x_0)}{dx^2} \frac{(x - x_0)^2}{2!} + \dots + \frac{d^{N-1}u(x_0)}{dx^{N-1}} \frac{(x - x_0)^{N-1}}{(N-1)!} + R^N$$

en donde

$$R^N = \frac{d^N u(\xi)}{dx^N} \frac{(x - x_0)^N}{(N)!}, \quad \xi \in (a, b)$$

es el residuo [29].

Asumiendo que $u(x)$ cumple con las hipótesis del teorema de Taylor, usamos su expansión alrededor del punto x_{i+1} en el intervalo $[a, b]$ para obtener

$$u_{i+1} = u_i + \frac{du}{dx}(x_i)(x_{i+1} - x_i) + \mathcal{O}(\Delta x_i^2),$$

donde $\mathcal{O}(\Delta x^2)$ es el residuo de orden 2, y $\Delta x_i = x_{i+1} - x_i$. De ahí obtenemos la expresión

$$\frac{du}{dx}(x_i) = \frac{u_{i+1} - u_i}{x_{i+1} - x_i} + \mathcal{O}(\Delta x_i^2), \quad (1.3)$$

que resulta ser la expresión de diferencias finitas, diferencias hacia adelante para ser más preciso; para la aproximación de du_i/dx .

Haciendo un proceso similar al anterior, podemos obtener la expresión en diferencias hacia atrás, basta con tomar la expansión alrededor del punto x_{i-1}

$$u_{i-1} = u_i + \frac{du}{dx}(x_i)(x_{i-1} - x_i) + \mathcal{O}(\Delta x_i^2),$$

y seguir un proceso similar para llegar a

$$\frac{du}{dx}(x_i) = \frac{u_{i-1} - u_i}{x_{i-1} - x_i} + \mathcal{O}(\Delta x_i^3). \quad (1.4)$$

El caso de las diferencias centradas es un poco más laborioso, ya que para obtenerlas primero se tienen que tener las expresiones, hasta orden 2, para u_{i+1} y u_{i-1} :

$$\begin{aligned} u_{i+1} &= u_i + \frac{du}{dx}(x_i)(x_{i+1} - x_i) + \frac{1}{2} \frac{d^2u}{dx^2}(x_i)(x_{i+1} - x_i)^2 + \mathcal{O}(\Delta x_i^3) \\ u_{i-1} &= u_i + \frac{du}{dx}(x_i)(x_{i-1} - x_i) + \frac{1}{2} \frac{d^2u}{dx^2}(x_i)(x_{i-1} - x_i)^2 + \mathcal{O}(\Delta x_i^3) \end{aligned}$$

y después se tiene que restar la segunda de la primera

$$u_{i+1} - u_{i-1} = \frac{du}{dx}(x_i)(x_{i+1} - x_{i-1}) + \frac{1}{2} \frac{\partial^2 u}{\partial x^2}(x_i)[(x_{i+1} - x_i)^2 - (x_{i-1} - x_i)^2] + \mathcal{O}(\Delta x_i^3)$$

para así llegar a la expresión para las diferencias centradas

$$\frac{du_i}{dx} = \frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} + \mathcal{O}(\Delta x_i^3). \quad (1.5)$$

Nótese que en el caso de una malla uniforme, donde

$$x_{i+1} - x_i = x_i - x_{i-1}$$

las ecuaciones (1.3), (1.4) y (1.5) se reducen a:

$$\frac{du}{dx}(x_i) = \frac{u_{i+1} - u_i}{\Delta x} + \mathcal{O}(\Delta x), \quad (1.6)$$

$$\frac{du}{dx}(x_i) = \frac{u_{i-1} - u_i}{\Delta x} + \mathcal{O}(\Delta x), \quad (1.7)$$

$$\frac{du}{dx}(x_i) = \frac{u_{i+1} - u_{i-1}}{2\Delta x} + \mathcal{O}(\Delta x^2), \quad (1.8)$$

y en el último caso se obtiene un grado de aproximación mayor.

6 Esquemas de Diferencias en Regiones Bidimensionales Rectangulares

Así como se definió el esquema de diferencias para aproximar el valor de la derivada en u en x , también se puede hacer fácilmente un esquema para aproximar d^2u/dx^2 . Sumando las expansiones en serie de Taylor de u en el punto x_i

$$u(x_{i+1}) \approx u(x_i) + \frac{du(x_i)}{dx}(x_{i+1} - x_i) + \frac{1}{2!} \frac{d^2u(x_i)}{dx^2}(x_{i+1} - x_i)^2 + \mathcal{O}(\Delta x^3),$$

y

$$u(x_{i-1}) \approx u(x_i) + \frac{du(x_i)}{dx}(x_{i-1} - x_i) + \frac{1}{2!} \frac{d^2u(x_i)}{dx^2}(x_{i-1} - x_i)^2 + \mathcal{O}(\Delta x^3),$$

y recordando que, en el caso de mallas uniformes, $x_{i+1} - x_i = \Delta x$ y $x_{i-1} - x_i = -\Delta x$ podemos obtener la expresión

$$u(x_{i+1}) + u(x_{i-1}) \approx 2u(x_i) + \frac{d^2u(x_i)}{dx^2}(\Delta x^2) + \mathcal{O}(\Delta x^4),$$

si hacemos las operaciones pertinentes para despejar $d^2u(x_i)/dx^2$, podemos reescribir la ecuación anterior como

$$\frac{d^2u(x_i)}{dx^2} = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1}))}{(\Delta x)^2} + \mathcal{O}(\Delta x^2)$$

que es la expresión en diferencias finitas para la aproximación de d^2u/dx^2 .

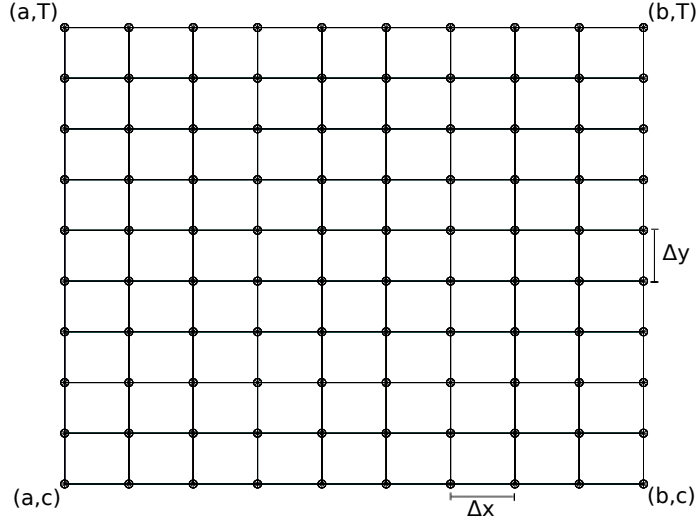
1.3. Esquemas de Diferencias en Regiones Bidimensionales Rectangulares

Una vez que hemos establecido los esquemas de diferencias en una dimensión podemos empezar a preguntarnos sobre el caso bidimensional. Los esquemas de diferencias en dos dimensiones pueden obtenerse por analogía de los de una dimensión, habida cuenta que las derivadas parciales implican una derivada en una dirección.

Sea una región rectangular (ver figura 1.1) definida en $[a, b] \times [c, T]$, sobre la cual construimos los nodos interiores como

$$\begin{aligned} x_i &= a + i\Delta x \\ y_j &= c + j\Delta y. \end{aligned}$$

donde h y k son el tamaño de paso horizontal y vertical respectivamente. Tomando en cuenta las ecuaciones (1.6 - 1.8), podemos escribir una expresión para cada una de las derivadas parciales

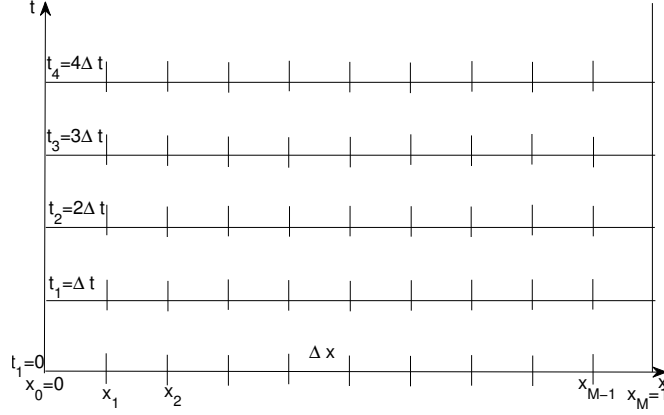
Figura 1.1: Mallado uniforme en el intervalo $[a, b] \times [c, T]$

$$\begin{aligned}
 \frac{\partial u}{\partial x}(x_i, y_j) &= \frac{u(x_{i+1}, y_j) - u(x_{i-1}, y_j)}{2\Delta x} + \mathcal{O}(\Delta x^2) \\
 \frac{\partial u}{\partial y}(x_i, y_j) &= \frac{u(x_i, y_{j+1}) - u(x_i, y_{j-1})}{2\Delta y} + \mathcal{O}(\Delta y^2) \\
 \frac{\partial^2 u}{\partial x^2}(x_i, y_j) &= \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j))}{\Delta x^2} + \mathcal{O}(\Delta x^2) \\
 \frac{\partial^2 u}{\partial y^2}(x_i, y_j) &= \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1}))}{\Delta y^2} + \mathcal{O}(\Delta y^2)
 \end{aligned}$$

Como nos encontramos ahora con un problema en dos dimensiones, dentro de las aproximaciones que debemos de obtener se encuentra aquella con el término cruzado $\partial^2 u / \partial x \partial y$, es fácil ver que una expresión para ello es

$$\frac{\partial^2 u}{\partial x \partial y} = \frac{u(x_{i+1}, y_{j+1}) - u(x_{i-1}, y_{j+1}) + u(x_{i-1}, y_{j-1}) - u(x_{i+1}, y_{j-1})}{4\Delta x \Delta y}.$$

Hasta el momento tenemos ya algunos esquemas en diferencias finitas que aproximan a la solución de cierta ecuación diferencial, usando una expansión en serie de Taylor, en donde al hacer la aproximación a la derivada tenemos un cierto orden. A continuación aplicaremos alguno de estos esquemas para resolver un ejemplo sencillo que nos ayude a entender mejor la aplicación de dichos esquemas.

Figura 1.2: Mallado uniforme en el intervalo $[0, 1]$ **Ejemplo 1**

Consideremos la ecuación de difusión o de calor, con una variable en el espacio y una en el tiempo

$$v_t(x, t) = \nu v_{xx}(x, t), \quad x \in (0, 1), \quad (1.9)$$

$$v(x, 0) = f(x), \quad x \in [0, 1], \quad (1.10)$$

$$v(0, t) = a(t), \quad v(1, t) = b(t), \quad t \geq 0. \quad (1.11)$$

Aquí a ν se le conoce como la constante de difusividad. La ecuación (1.10) es la condición inicial y (1.11) es la condición de frontera. Nuestro objetivo es escribir un esquema en diferencias finitas asociado a dicha ecuación. Un estudio sobre la solución analítica de este problema puede ser encontrado en [30].

Antes de comenzar con el proceso de aproximación es importante introducir cierta notación: u será la función que aproxime a v y dependerá de x y t ; denotaremos como u_m^n la función definida en el punto $(m\Delta x, n\Delta t)$ o, en aras de la simplicidad, el punto (m, n) , con $m = 0, 1, 2, \dots, M$ y $n = 0, 1, 2, \dots, N$.

Por conveniencia vamos a usar un mallado uniforme, con espacio entre cada nodo espacial de $\Delta x = 1/M$ (véase la figura 1.2).

De la misma manera que hemos discretizado el intervalo en el espacio podemos discretizar el tiempo con un mallado uniforme con separación Δt entre los nodos temporales. Haciendo este proceso obtenemos un mallado que va a aproximar el dominio espacio-tiempo como se muestra en la figura (1.3).

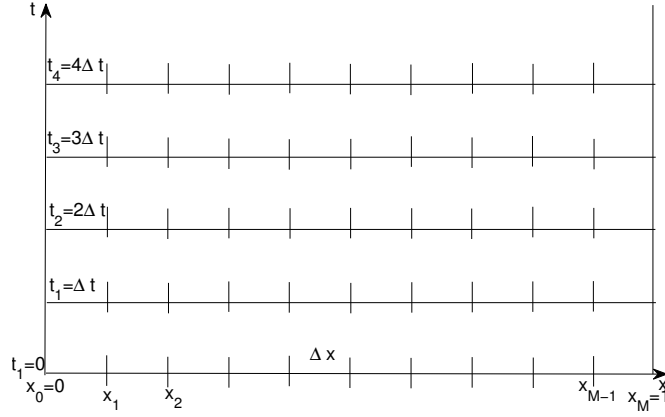


Figura 1.3: Mallado uniforme en el dominio espacio-tiempo

El siguiente paso es el de aproximar la solución de la ecuación (1.9) utilizando algún esquema de diferencias finitas, con los cuales ya contamos. En este caso utilizaremos *diferencias hacia delante* para hacer la discretización en el tiempo, para obtener

$$\frac{\partial v}{\partial t}(m\Delta x, n\Delta t) = v_t \approx \frac{u_m^{n+1} - u_m^n}{\Delta t}, \quad (1.12)$$

después utilizamos la expresión de la segunda derivada para aproximar v_{xx} , esto nos arroja

$$\frac{\partial^2 v}{\partial x^2}(m\Delta x, n\Delta t) = v_{xx} \approx \frac{u_{m+1}^n - 2u_m^n + u_{m-1}^n}{\Delta x^2}. \quad (1.13)$$

Podemos combinar las ecuaciones (1.12) y (1.13), para así aproximar la ecuación diferencial parcial (1.9); esto es, obtendremos la expresión

$$\frac{u_m^{n+1} - u_m^n}{\Delta t} = \nu \frac{u_{m+1}^n - 2u_m^n + u_{m-1}^n}{\Delta x^2}$$

que, tras despejar u_m^{n+1} , puede ser escrita como

$$u_m^{n+1} = u_m^n + \nu \frac{\Delta t}{\Delta x^2} (u_{m+1}^n - 2u_m^n + u_{m-1}^n). \quad (1.14)$$

La ecuación (1.14) es el esquema en diferencias finitas para la aproximación de la ecuación diferencial parcial descrita en (1.9). Ahora, las condiciones iniciales y de frontera están dadas por

$$u_k^0 = f(k\Delta x) \quad k = 0, \dots, M \quad (1.15)$$

$$u_0^{n+1} = a((n+1)\Delta t) \quad n = 0, \dots, N \quad (1.16)$$

$$u_M^{n+1} = b((n+1)\Delta t) \quad n = 0, \dots, N \quad (1.17)$$

Entonces nuestro objetivo es obtener una buena aproximación a la solución de la ecuación diferencial parcial en $(0, 1) \times (0, 1)$ resolviendo el problema discreto dado por (1.14-1.17). El qué tan buena será nuestra aproximación dependerá de la elección de Δx y Δt ; debemos ser muy cuidadosos al momento de elegirlos, ya que éstos serán quienes determinen el comportamiento de la aproximación a la solución. A continuación analizaremos con más detalle el cuidado de esta elección, y haremos una revisión somera de los conceptos de consistencia, estabilidad y convergencia.

1.3.1. Consistencia

Si queremos aproximar una ecuación diferencial parcial por medio de un esquema de diferencias finitas podremos utilizar la expansión en serie de Taylor alrededor de ciertos puntos para llegar a expresiones que logren esta aproximación. En particular, podemos hacer la expansión alrededor de $(n\Delta t, k\Delta x)$, y obtenemos

$$\begin{aligned} v_k^{n+1} &= v((n+1)\Delta t, k\Delta x) = v(n\Delta t, k\Delta x) + \frac{\partial v}{\partial t}(n\Delta t, k\Delta x) \frac{\Delta t}{1!} \\ &\quad + \frac{\partial^2 v}{\partial t^2}(n\Delta t, k\Delta x) \frac{\Delta t^2}{2!} + \dots \end{aligned}$$

Nótese que solo se escriben los primeros tres términos de la expansión. Si hacemos un poco de álgebra, podemos reescribir esta ecuación como

$$\frac{v_k^{n+1} - v_k^n}{\Delta t} = \frac{\partial v}{\partial t}(n\Delta t, k\Delta x) + \frac{\Delta t}{2} \left(\frac{\partial^2 v}{\partial t^2} \right)_k + \dots,$$

que a su vez, utilizando una manera más general en términos de $\mathcal{O}$, si v es lo suficientemente suave puede escribirse como

$$\frac{\partial v}{\partial t}(n\Delta t, k\Delta x) = \frac{v_k^{n+1} - v_k^n}{\Delta t} + \mathcal{O}(\Delta t).$$

Si queremos averiguar el orden del esquema de diferencias que aproxima la segunda derivada, utilizamos la expansión en serie de Taylor de (1.13)

$$\begin{aligned} v_{m+1}^n &= v(n\Delta t, m\Delta x) + \frac{\partial v}{\partial x}(n\Delta t, m\Delta x) \frac{\Delta x^2}{1!} + \frac{\partial^2 v}{\partial x^2}(n\Delta t, m\Delta x) \frac{\Delta x^2}{2!} + \\ &\quad \frac{\partial^3 v}{\partial x^3}(n\Delta t, m\Delta x) \frac{\Delta x^3}{3!} + \mathcal{O}(\Delta x^4), \\ v_{m-1}^n &= v(n\Delta t, m\Delta x) - \frac{\partial v}{\partial x}(n\Delta t, m\Delta x) \frac{\Delta x^2}{1!} + \frac{\partial^2 v}{\partial x^2}(n\Delta t, m\Delta x) \frac{\Delta x^2}{2!} - \\ &\quad \frac{\partial^3 v}{\partial x^3}(n\Delta t, m\Delta x) \frac{\Delta x^3}{3!} + \mathcal{O}(\Delta x^4). \end{aligned}$$

Sumando estas dos expansiones, y haciendo un poco de álgebra, podemos llegar al esquema que aproxima a v_{xx}

$$\frac{\partial^2 v}{\partial x^2} = \frac{v_{m+1}^n - 2v_m^n + v_{m-1}^n}{\Delta x^2} + \mathcal{O}(\Delta x^2). \quad (1.18)$$

Entonces el problema (1.9) puede escribirse como

$$\frac{v_m^{n+1} - v_m^n}{\Delta t} - \frac{\nu}{\Delta x^2} (v_{m+1}^n - 2v_m^n + v_{m-1}^n) + \mathcal{O}(\Delta t) + \mathcal{O}(\Delta x^2) = v_t(n\Delta t, m\Delta x) - \nu v_{xx}(n\Delta t, m\Delta x) = \quad (1.19)$$

Podemos ver que la ecuación (1.19) muestra qué tan bueno es el esquema en diferencias para aproximar a la ecuación original. Debemos resaltar que esto no nos dice directamente cómo la solución del esquema en diferencias se aproxima a la solución de la ecuación diferencial parcial.

Definición 1.1 *Se dice que el esquema en diferencias $L_m^n u_m^n = G_m^n$ es consistente punto a punto con la ecuación diferencial $LU = F$ si existe alguna $\phi(x, t)$, suave y continua, tal que*

$$(L\phi - F)|_k^n - [L_m^n \phi(n\Delta t, m\Delta x) - G_m^n] \rightarrow 0 \quad (1.20)$$

cuando $\Delta x, \Delta t$ y $((n+1)\Delta t, m\Delta x) \rightarrow 0$

Una vez definida la consistencia punto a punto, podemos ver que el esquema descrito en (1.19) es consistente punto a punto con la ecuación diferencial parcial original.

Podemos elegir ϕ como la solución v de la ecuación diferencial parcial, entonces la expresión en (1.20) la escribimos

$$L_m^n u_m^n - G_m^n \rightarrow 0 \quad \text{cuando} \quad \Delta x, \Delta t \rightarrow 0.$$

Si escribimos el esquema en diferencias como

$$\mathbf{u}^{n+1} = Q\mathbf{u}^n + \Delta t \mathbf{G}^n \quad (1.21)$$

de donde $\mathbf{u}^n = (\dots, u_{-1}^n, u_0^n, u_1^n, \dots)^T$, $\mathbf{G}^n = (\dots, u_{-1}^n, u_0^n, u_1^n, \dots)^T$ y Q es un operador que actúa en el espacio apropiado, entonces podemos escribir una definición de consistencia más fuerte.

Definición 1.2 *El esquema (1.21) es consistente con la ecuación diferencial parcial con la norma $\|\cdot\|$, si la solución de la ecuación diferencial parcial v satisface*

$$\mathbf{v}^{n+1} = Q\mathbf{u}^n + \Delta t \mathbf{G}^n + \Delta t \boldsymbol{\tau}^n,$$

y

$$\|\boldsymbol{\tau}^n\| \rightarrow 0$$

cuando $\Delta x, \Delta t \rightarrow 0$, donde $\mathbf{u}^n$ denota el vector cuya k -ésima componente es $v(n\Delta t, m\Delta x)$.

Hay que notar que la diferencia entre la consistencia punto a punto, y la consistencia con la norma, es que la consistencia con la norma obliga a todos los componentes del vector $\boldsymbol{\tau}^n$ a converger a cero de una manera uniforme.

1.3.2. Estabilidad

Una interpretación intuitiva de la estabilidad es que, si el esquema en diferencias de un problema con condiciones iniciales tiene errores pequeños de entrada, entonces tendremos errores pequeños en la aproximación final.

La estabilidad de un esquema con dos niveles de tiempo se explica en términos de una ecuación de la forma

$$u^{n+1} = Qu_n \quad n \geq 0 \quad (1.22)$$

Definición 1.3 Se dice que un esquema de diferencias con la forma (1.22) es estable con respecto a la norma $\|\cdot\|$ si existen constantes positivas Δx , Δt , α y β tales que

$$\|u^{n+1}\| \leq Ke^{\beta t} \|u^0\|, \quad (1.23)$$

para $0 \leq t \leq (n+1)\Delta t$, $0 < \Delta x \leq x_0$ y $0 < \Delta t \leq \Delta t_0$.

Hay que notar que, al igual que para la consistencia, la definición de estabilidad está dada en términos de una norma que puede variar dependiente del problema. Pero, independientemente de la norma, la solución puede crecer con el tiempo pero no con el número de pasos de tiempo. Es importante hacer ver que existe una gran variedad de definiciones de estabilidad, por ejemplo una definición común es aquella que no permite un crecimiento exponencial, entonces la desigualdad (1.23) es remplazada por

$$\|u^{n+1}\| \leq K \|u^0\|,$$

aquí el valor de K es quien dirá si el esquema es consistente o no. Si $0 < K < 1$ entonces el esquema será *incondicionalmente estable*. Si $K = 1$ entonces el esquema es *condicionalmente estable*, y por último si $K > 1$, entonces el esquema resulta ser *incondicionalmente inestable*.

1.3.3. Convergencia

Los esquemas con los que hemos estado tratando son muy usados, ya que presentan resultados satisfactorios a la hora de aproximar la solución de ciertas ecuaciones diferenciales en diferentes dominios. Esto se debe a que estamos obteniendo la solución del sistema de diferencias que surge de hacer una aproximación a la solución de la ecuación diferencial. Entonces ahora nos interesa garantizar la convergencia de estos esquemas, para poder describir buenos resultados de las aproximaciones generadas. No es fácil de conseguir esto en general, y en, muchas ocasiones, el proceso para conseguirlo llega a ser muy largo y laborioso.

Consideremos la siguiente EDP

$$Lv = F, \quad (1.24)$$

con v y F funciones que están definidas en $\mathbb{R}$ y con condición inicial $v(x, 0) = f(x)$. Supongamos que tenemos una aproximación a la solución, obtenido por medio de algún esquema de diferencias finitas, que llamaremos L_m^n . Para este problema, u_m^n define una malla uniforme con Δx y Δt distancias entre nodos, y satisface la condición inicial $u_m^0 = f(m\Delta x) \in [a, b]$. Denotemos a v como la solución exacta de nuestro problema inicial.

Definición 1.4 *Un esquema de diferencias $L_m^n u_k^n = G_m^n$ que aproxima a la ecuación diferencial parcial $Lv = F$ es un esquema que converge punto a punto para algún x y t , mientras $(m\Delta x, (n+1)\Delta t)$ converge a (x, t) , u_m^n converge a $v(x, t)$ cuando Δx y Δt tienden a cero.*

Para ejemplificar mejor esto retomaremos el problema del ejemplo 1, junto como su esquema en diferencias, el cual reescribiremos como

$$u_m^{n+1} = (1 - 2r)u_m^n + r(u_{m+1}^n + u_{m-1}^n),$$

en donde $r = \nu\Delta t/\Delta x^2$.

Definamos z_m^n como

$$z_k^n = u_k^n - v(m\Delta x, n\Delta t).$$

Si ahora regresamos a la ecuación diferencial parcial original, y la escribimos como

$$\begin{aligned} v_t(m\Delta x, n\Delta t) - \nu v_{xx}(m\Delta x, n\Delta t) = \\ \frac{v_m^{n+1} - v_m^n}{\Delta t} - \frac{\nu}{\Delta x^2}(v_{m+1}^n - 2v_m^n + v_{m-1}^n) + \mathcal{O}(\Delta t) + \mathcal{O}(\Delta x^2) \end{aligned} \quad (1.25)$$

y ponemos la solución exacta v del lado izquierdo de la ecuación anterior obtenemos

$$0 = \frac{v_m^{n+1} - v_m^n}{\Delta t} - \frac{\nu}{\Delta x^2}(v_{m+1}^n - 2v_m^n + v_{m-1}^n) + \mathcal{O}(\Delta t) + \mathcal{O}(\Delta x^2),$$

de donde podemos despejar v_m^{n+1} para obtener

$$v_m^{n+1} = (1 - 2r)v_m^n + r(v_{m+1}^n + v_{m-1}^n) + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta t\Delta x^2),$$

y finalmente, restando la ecuación anterior de la ecuación (1.25), tendremos

$$z_m^{n+1} = (1 - 2r)z_m^n + r(z_{m+1}^n + z_{m-1}^n) + \mathcal{O}(\Delta t^2) + \mathcal{O}(\Delta t\Delta x^2). \quad (1.26)$$

Si $0 < r \leq 1/2$, los coeficientes del lado derecho de la ecuación (1.26) son positivos y, además,

$$\begin{aligned} |z_m^{n+1}| &\leq (1 - 2r)|z_m^n| + r(|z_{m+1}^n| + |z_{m-1}^n|) + A(\Delta t^2 + \Delta t\Delta x^2) \\ &\leq Z^n + A(\Delta t^2 + \Delta t\Delta x^2), \end{aligned}$$

14 Esquemas de Diferencias en Regiones Bidimensionales Rectangulares

donde A es una constante asociada con el término $\mathcal{O}$ y $Z^n = \sup_k \{|z_k^n|\}$. Luego, si tomamos el $\sup$ sobre k del lado izquierdo, tenemos

$$Z^{n+1} \leq Z^n + A(\Delta t^2 + \Delta t \Delta x^2).$$

Notemos que el supremo del lado derecho de la ecuación anterior incluye los términos Δt y Δx . Aplicando esta condición un número finito de veces

$$\begin{aligned} Z^{n+1} &\leq Z^n + A(\Delta t^2 + \Delta t \Delta x^2) \\ &\leq Z^{n-1} + 2A(\Delta t^2 + \Delta t \Delta x^2) \\ &\quad \cdot \\ &\quad \cdot \\ &\quad \cdot \\ &\leq Z^0 + (n+1)A(\Delta t^2 + \Delta t \Delta x^2). \end{aligned}$$

De ahí que si $Z^0 = 0$, $|u_m^{n+1} - v(m\Delta x, (n+1)\Delta t)| \leq Z^{n+1}$ y $(n+1)\Delta t \rightarrow T$.

$$\begin{aligned} |u_m^{n+1} - v(m\Delta x, (n+1)\Delta t)| &\leq (n+1)\Delta t A(\Delta t + \Delta x^2) \\ &\rightarrow 0 \quad \text{cuando} \quad \Delta t, \Delta x \rightarrow 0. \end{aligned}$$

Así que para algún x y t , cuando Δt y Δx se aproximan a cero y

$$(m\Delta x, (n+1)\Delta t) \rightarrow (x, t),$$

entonces u_m^n se aproxima a $v(x, t)$. Hay que señalar que, el término $n+1$ en la expresión anterior podría ser un problema para nuestra convergencia por lo que es importante que $(n+1)\Delta t \rightarrow T$. Con esto hemos logrado probar que $Z^n \rightarrow 0$, con $z_m^n = u_m^n - v(m\Delta x, n\Delta t)$.

En general es muy difícil probar la convergencia punto a punto, y no suele ser tan útil como un tipo de convergencia uniforme, en vez de eso utilizamos una definición de convergencia en términos de una norma de la diferencia entre la solución exacta y la aproximación obtenida.

Si queremos probar la convergencia en términos de la rapidez con la que la aproximación converge a la solución exacta, debemos definir la *convergencia de orden* (p, q) como sigue

Definición 1.5 *Un esquema de diferencias $L_k^n u_k^n$ que aproxima a la ecuación diferencial parcial $Lu = F$, es un esquema convergente de orden (p, q) si para t , cuando $(n+1)\Delta t \rightarrow T$*

$$\|\mathbf{u}^{n+1} - \mathbf{v}^{n+1}\| = \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta t^q)$$

cuando $\Delta x, \Delta y \rightarrow 0$

En este caso $\mathcal{O}$ dependerá de t . Usando esta definición estamos en condiciones de discutir el rango de convergencia de $\mathbf{u}^n$ a $\mathbf{v}^n$.

No es nuestra intención profundizar más en los conceptos mencionados en este capítulo, bastaba con dar una introducción a ellos para así poder introducir también la notación que se utiliza a lo largo del presente trabajo. En particular a lo largo de este capítulo se habló de diferentes particiones, o mallas, de las cuales no se introdujo el tema; esto se hará en el capítulo 2.

Capítulo 2

Generación numérica de mallas

Existen 2 métodos ampliamente usados en la solución de problemas que involucran ecuaciones diferenciales parciales, estos son el método de diferencias finitas (MDF) y el método de elementos finitos (MEF). Ambos métodos cuentan con sus ventajas y desventajas con respecto del otro. Algunas personas prefieren el método de elementos finitos porque se puede adaptar fácilmente a regiones no regulares, sin embargo el trasfondo teórico y las estructuras de datos requeridas para su uso pueden ser difíciles de manejar.

Por otro lado, el método de diferencias finitas cuenta con una base teórica muy simple que no requiere conocimientos avanzados de matemáticas; sin embargo, su aplicación en regiones irregulares no es trivial.

Desde hace algún tiempo se ha propuesto usar cambios de variables para transformar un problema sobre una región irregular en un problema sobre el cuadrado unitario. Para poder aplicar este método es necesario construir una función suave y biyectiva entre la región original de trabajo y el cuadrado unitario. Esto es esencialmente lo que se le conoce como generación numérica de mallas sobre regiones planas irregulares.

2.1. ¿Qué es una Malla?

Cuando se resuelve numéricamente un problema continuo de modelado, frecuentemente es necesario hacer una discretización sobre el dominio en el que se trabaja, esto es, hacer una malla que represente el dominio donde se desea modelar el problema. Esto se hace seleccionando un conjunto finito de puntos que reflejen la geometría de la región.

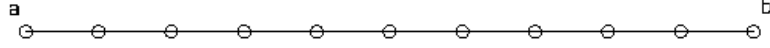


Figura 2.1: Malla uniforme en 1 dimensión.

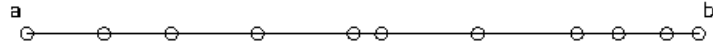


Figura 2.2: Malla no uniforme en 1 dimensión.

Si queremos resolver problemas en una dimensión en el intervalo $[a, b]$, podemos generar 2 tipos de mallas, mallas uniformes y mallas no uniformes. Mientras que las primeras consisten en partir el intervalo $[a, b]$ en una cantidad n de subintervalos de igual longitud (figura 2.1), las segundas consisten en partir el mismo intervalo pero de manera que los n subintervalos no tengan todos la misma longitud (figura 2.2).

Para problemas en 2 dimensiones, podemos considerar dos tipos de regiones. Las primeras son dominios simples, como rectángulos, $[0, a] \times [0, b]$ en donde se puede hacer fácilmente una discretización dividiendo la región en rectángulos regulares más pequeños; esta discretización, al igual que en el caso en 1 dimensión, se puede hacer de manera uniforme (figura 2.1) o no uniforme (figura 2.2).

El segundo tipo de regiones, son dominios que no cuentan con una geometría regular y sencilla de discretizar como en el caso de las mallas sobre dominios rectangulares. Sobre la generación de mallas en este tipo de regiones se hablará en este capítulo.

2.2. Generación Numérica de Mallas Estructuradas

Una malla $\mathbf{x}(\xi, \eta) = (x(\xi, \eta), y(\xi, \eta))^t$ sobre una región $\Omega \in R^2$ es una función continua $\mathbf{x} : U_2 \rightarrow \Omega$ (figura 2.2), donde U_2 es el cuadrado unitario $[0, 1] \times [0, 1]$. Si tomamos una línea coordenada ξ o η constante en U_2 su imagen en Ω será una curva, estas curvas son llamadas curvas coordenadas.

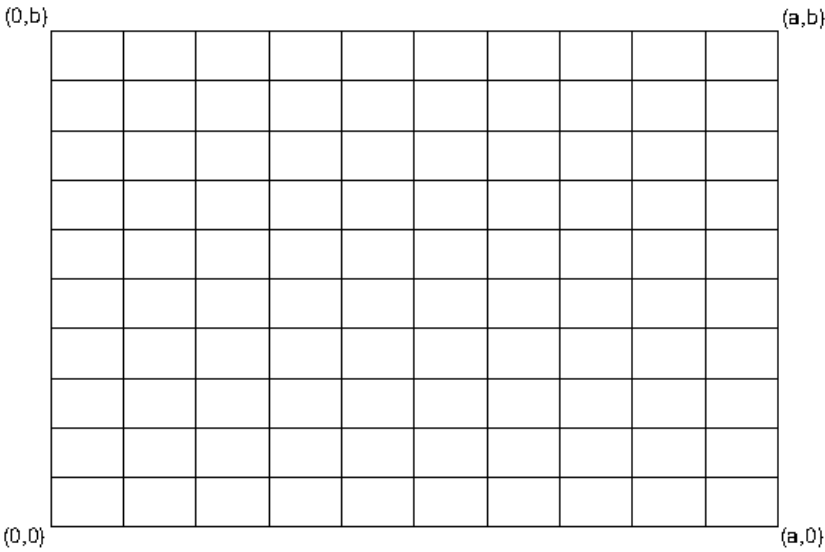


Figura 2.3: Malla uniforme en 2 dimensiones.

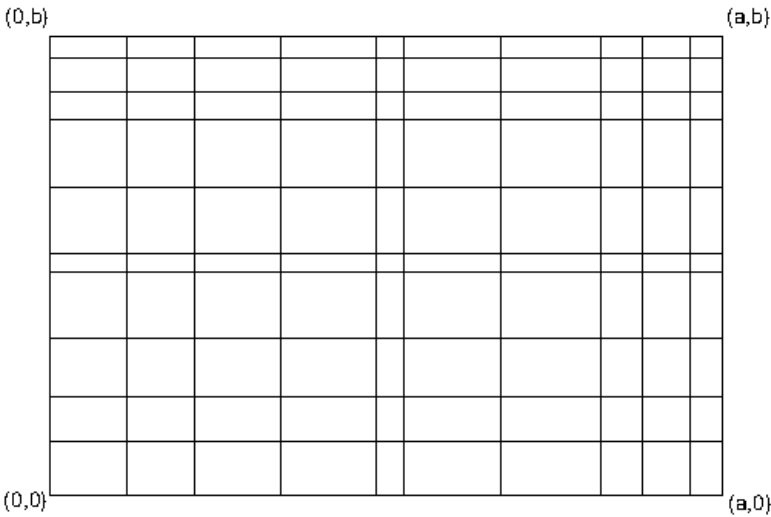


Figura 2.4: Malla no uniforme en 2 dimensiones.

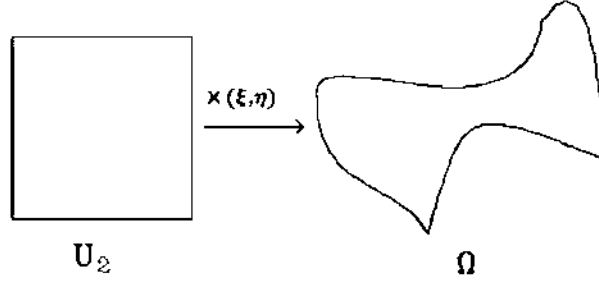


Figura 2.5: Transformación $U_2 \rightarrow \Omega$.

Si en U_2 tomamos una cuadrícula, o en otras palabras, un conjunto de líneas coordenadas en ξ y η , la función $\mathbf{x}(\xi, \eta)$ proporcionará un conjunto de curvas coordenadas que formará un sistema de coordenadas en Ω , de ahí viene el nombre de malla o red.

Una cualidad fundamental para resolver ecuaciones diferenciales de la función $\mathbf{x}$ es que cumpla con $\mathbf{x}(\partial U_2) = \partial \Omega$, si es así diremos que se ajustan a la frontera, o que conforman a la misma.

De entre las posibles mallas que pueden ser generadas para la región Ω nos interesarán las mallas que cumplan con las siguientes características:

1. Que todos los diferentes puntos de U_2 tengan como imagen puntos diferentes en Ω ; esto quiere decir que se requiere que $\mathbf{x}$ sea 1-1.
2. Que $\mathbf{x}$ llene a Ω ; es decir, que $\mathbf{x}$ sea un mapeo sobre.
3. Es deseable que la imagen de una curva suave en U_2 sea una curva suave en Ω ; esto es, que las funciones $\xi(\eta)$ y $\eta(\xi)$ tengan derivadas continuas al ser consideradas como funciones de ξ y η separadamente.

Las mallas que cumplen con las primeras 2 características son llamadas “mallas convexas”; estas son el tipo de mallas que se utilizan en gran cantidad de trabajos sobre solución numérica de ecuaciones diferenciales parciales, debido a que las mallas que no cumplen estas características, llamadas “no convexas” o “dobladitas” no son útiles para este fin porque no representan una transformación de coordenadas.

La malla que cumple con las 3 características es llamada un difeomorfismo entre U_2 y Ω . Entonces podemos plantear, de manera más especializada, el problema de la generación de mallas como:

Dado un difeomorfismo

$$x : \partial(U_2) \rightarrow \partial\Omega$$

extenderlo a un difeomorfismo

$$x : U_2 \rightarrow \Omega$$

Hasta hace no mucho tiempo, los métodos para generar este tipo de mallas eran básicamente de 2 tipos: (1) construcción por interpolación algebraica, (2) solución numérica de ecuaciones diferenciales parciales. El primer tipo de procedimientos van desde ser muy sencillos, con una simple normalización de las curvas frontera, hasta la interpolación transfinita de las curvas frontera. Por su parte, en los métodos por solución numérica de ecuaciones diferenciales, las ecuaciones pueden ser elípticas, parabólicas o hiperbólicas. Aún cuando cada uno de estos procedimientos tiene sus ventajas, estos se pueden resumir diciendo que la generación algebraica es más rápida, pero las mallas obtenidas por ecuaciones diferenciales son más suaves.

2.2.1. El Método de Interpolación Transfinita

Como ya fue mencionado anteriormente, los métodos más simples usados para la generación de mallas son los algebraicos basados en interpolación. Estos métodos tienen la ventaja de ser de fácil implementación y muy rápidos en comparación con los métodos variaciones continuos y discretos que se verán en la secciones siguientes. Tienen, sin embargo, la desventaja de que transmiten la falta de suavidad de la frontera hacia el interior y, para regiones no convexas, frecuentemente forman mallas dobladas; esto quiere decir que forman mapeos que no son 1-1. No obstante, debido a su simplicidad, estos métodos son usados extensivamente en diferentes áreas, y las mallas que producen son usadas como punto de partida por los métodos que se estudiarán más adelante, es conveniente revisar al menos la versión más usada de ellos: el método de interpolación transfinita (TFI). Para leer más sobre estos métodos algebraicos el lector puede consultar [13], [17], [18], [19], [24], [26] o [27].

Para generar una malla sobre una región Ω es necesario conocer las imágenes de los lados del cuadrado unitario; o dicho de otra forma, se requiere la parametrización de las cuatro partes en las cuales la frontera de Ω es dividida y que son imágenes correspondientes de cuatro segmentos de la frontera de U_2 . Si suponemos que la descripción de la frontera de Ω está dada por cuatro ecuaciones paramétricas

$$\begin{aligned} x_b(\xi), \quad x_t(\xi) & \quad 0 < \xi < 1, \\ x_l(\eta), \quad x_r(\eta) & \quad 0 < \eta < 1, \end{aligned}$$

donde los subíndices b, r, t, l representan los lados abajo, derecha, arriba, izquierda (por sus nombres en inglés bottom, right, top, left)(figura 2.6). La orien-

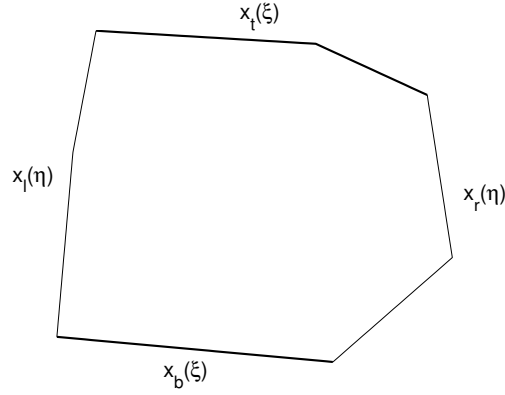


Figura 2.6: Region dada por cuatro ecuaciones paramétricas.

tación debe de conservarse y, para que x sea continua, se debe cumplir que en las esquinas correspondientes los mapeos coincidan; eso quiere decir que

$$x_b(0) = x_l(0) \quad x_b(1) = x_r(0) \quad x_r(1) = x_t(1) \quad x_t(0) = x_l(1).$$

La fórmula básica de la interpolación transfinita usa estos cuatro mapeos para generar una malla sobre Ω :

$$\begin{aligned} x(\xi, \eta) = & (1 - \eta)x_b(\xi) + \eta x_t(\xi) + (1 - \xi)x_l(\eta) + \xi x_r(\eta) - [\xi \eta x_t(1) + \\ & \xi(1 - \eta)x_b(1) + \eta(1 - \xi)x_l(0) + (1 - \xi)(1 - \eta)x_b(0)] \end{aligned}$$

Se muestran mallas generadas por interpolación transfinita para cuatro regiones no convexas (fig. 2.7) y para dos regiones que son aproximaciones a regiones geográficas reales (fig. 2.8). Las primeras dos, llamadas m13 y m19, han sido extensamente usadas por varios autores para ejemplificar las características de las diferentes técnicas de generación de mallas. Las otras dos son regiones “difíciles”, en el sentido de que no se logra convexidad de las mallas correspondientes cuando se usan métodos clásicos. Estas mallas se obtuvieron tomando la malla uniforme de 41 por 41 puntos por lado del cuadrado unitario, y serán usadas a lo largo del presente trabajo.

Ninguna de estas mallas es convexa; en la tabla 2.1 se muestra el número de celdas no convexas correspondiente a cada una de ellas.

2.3. Generación Variacional Continua

En la sección anterior se mencionó que los métodos para generar mallas eran básicamente de dos tipos: La construcción por interpolación algebraica y la solución numérica de ecuaciones diferenciales parciales. En esa misma sección se

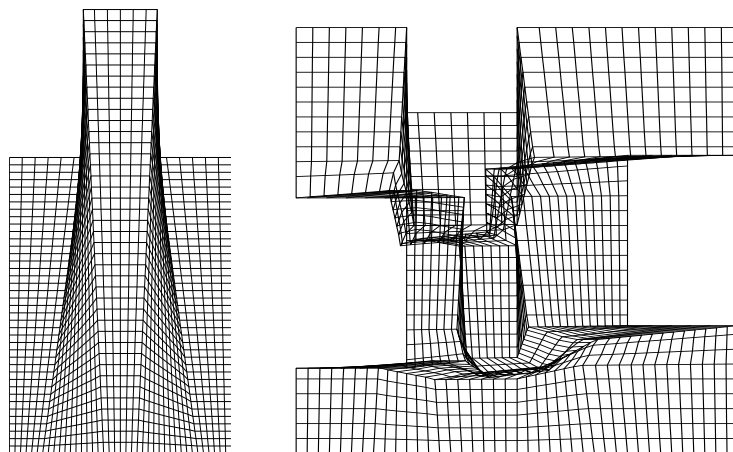


Figura 2.7: Mallas generadas con TFI para las regiones m13 y m19.

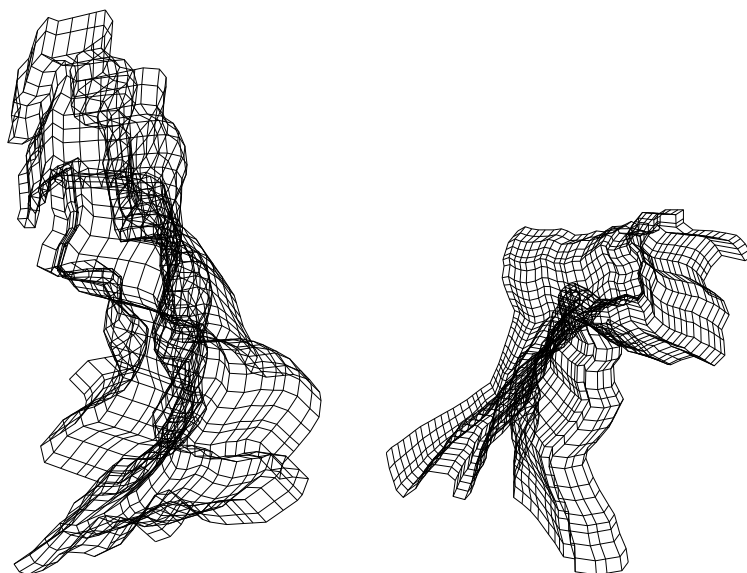


Figura 2.8: Mallas generadas con TFI para las regiones ENG y HAB.

Malla	Celdas no convexas
M13	88
M19	253
ENG	575
HAB	394

Tabla 2.1: Número de celdas no convexas de las mallas generadas por TFI

comentó un método perteneciente al primero grupo. Esta sección está dedicada a hablar de diferentes métodos mediante la solución numérica de ecuaciones diferenciales parciales.

2.3.1. Introducción

Aun cuando la noción de funcional es mucho más amplia, para los fines de este trabajo, se entenderá por un funcional continuo a una función de la forma

$$I(\mathbf{x}) = \int_0^1 \int_0^1 L(\xi, \eta, x, y, x_\xi, x_\eta, y_\xi, y_\eta) d\eta d\xi$$

que a cada malla

$$\mathbf{x} : U_2 \rightarrow \Omega$$

le asocia un número real. Para el presente trabajo, los funcionales que serán considerados tienen una forma mucho más simple, del estilo

$$I(\mathbf{x}) = \int_0^1 \int_0^1 L(x_\xi, x_\eta, y_\xi, y_\eta) d\eta d\xi$$

donde L es una función que contendrá información acerca de ciertas cantidades geométricas que se desea controlar con el objeto de generar mallas adecuadas y que conformen a la frontera de Ω .

El problema de la generación variaciones continua de mallas es encontrar $\mathbf{x}^*$ que conforme la frontera y que haga mínimo el valor de $I(\mathbf{x})$; además $\mathbf{x}$ debe de tener un valor predeterminado sobre la frontera de U_2 . Esto quiere decir que necesitamos calcular una $\mathbf{x}^*$ tal que

$$I(\mathbf{x}^*) = \min_{\mathbf{x}} I(\mathbf{x})$$

Entonces, usando cálculo de variaciones ([14], [16]), obtenemos que $\mathbf{x}^*$ debe de ser solución del problema de valores a la frontera

$$\frac{d}{d\xi} \left(\frac{\partial L}{\partial x_\xi} \right) + \frac{d}{d\eta} \left(\frac{\partial L}{\partial x_\eta} \right) = 0$$

$$\frac{d}{d\xi} \left(\frac{\partial L}{\partial y_\xi} \right) + \frac{d}{d\eta} \left(\frac{\partial L}{\partial y_\eta} \right) = 0$$

con $\mathbf{x}(\partial U_2)$ dado; el cual consta de un sistema de ecuaciones diferenciales parciales, comúnmente acoplado y no lineal, y de condiciones a la frontera dadas por el valor de la malla en la frontera. Estas condiciones son llamadas en conjunto *Ecuaciones de Euler-Lagrange*. Comúnmente este problema de valores a la frontera se resuelve numéricamente usando algún esquema de diferencias finitas y algoritmos iterativos, tomando la malla generada por TFI como una aproximación inicial.

2.3.2. Funcional de Longitud

El funcional clásico de Longitud Φ_l en su forma más simple se obtiene haciendo

$$\lambda L(x_\xi, x_\eta, y_\xi, y_\eta) = x_\xi^2 + x_\eta^2 + y_\xi^2 + y_\eta^2.$$

Al obtener el mínimo del funcional

$$\Phi_l(\mathbf{x}) = \int_0^1 \int_0^1 \lambda d\eta d\xi$$

se está tratando de obtener una malla que tenga la característica de que las longitudes de las líneas coordenadas sean los más cercadas posibles a un mismo valor. De hecho, este funcional se puede escribir como

$$\begin{aligned} \Phi(\mathbf{x}_l) &= \int_0^1 \int_0^1 [(x_\xi - y_\eta)^2 + (x_\eta + y_\xi)^2] d\eta d\xi + 2 \int_0^1 \int_0^1 (x_\xi y_\eta - x_\eta y_\xi) d\eta d\xi \\ &= \int_0^1 \int_0^1 [(x_\xi - y_\eta)^2 + (x_\eta + y_\xi)^2] d\eta d\xi + 2 \int_0^1 \int_0^1 J d\eta d\xi \end{aligned}$$

donde

$$J = x_\xi y_\eta - x_\eta y_\xi$$

es llamado el jacobiano de la transformación $\mathbf{x}$. Como $\int_0^1 \int_0^1 J d\eta d\xi = Area(\Omega)$, el minimizar $\Phi_l(\mathbf{x})$ es lo mismo que minimizar

$$\int_0^1 \int_0^1 [(x_\xi - y_\eta)^2 + (x_\eta + y_\xi)^2] d\eta d\xi$$

esto quiere decir que el mínimo de $\Phi(\mathbf{x})$ se consigue en una transformación que satisface las ecuaciones de Cauchy-Riemann

$$x_\xi = y_\eta \quad x_\eta = -y_\xi$$

en el sentido de los cuadrados mínimos. Las ecuaciones de Euler-Lagrange para Φ_l son

$$x_{\xi\xi} + x_{\eta\eta} = 0$$

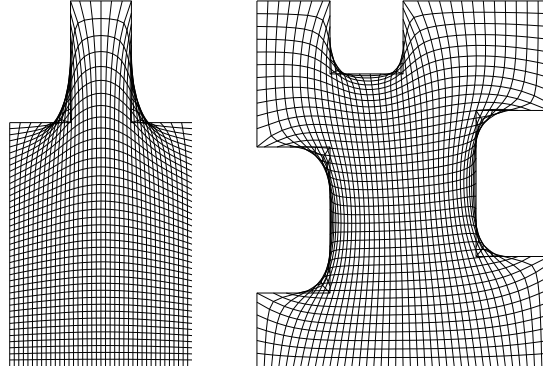


Figura 2.9: Mallas generadas usando el funcional continuo de longitud para las regiones m13 y m19.

Malla	Celdas no convexas
M13	42
M19	78
ENG	245
HAB	94

Tabla 2.2: Número de celdas no convexas usando el funcional continuo de longitud

$$y_{\xi\xi} + y_{\eta\eta} = 0$$

que no son otra cosa más que las ecuaciones de Laplace para los componentes de la transformación. En este caso particular las ecuaciones de Euler-Lagrange son elípticas, lineales y no acopladas, y es el caso más simple que se presentará. El hecho de que un sistema sea elíptico garantiza que sus soluciones tienen derivadas continuas de más alto orden que las condiciones de frontera; si elegimos condiciones de frontera “suaves” las soluciones serán “suaves”.

Por desgracia las mallas producidas por este funcional no son útiles, ya que para la mayoría de las regiones no convexas se generan mallas dobladas; esto es, no son $1-1$; más sin embargo son muy suaves. En las figuras (2.9 y 2.10) se presentan mallas generadas minimizando este funcional, utilizando el algoritmo presentado en [24], para las 4 regiones de prueba. El número de celdas no convexas se muestra en la tabla (2.2).

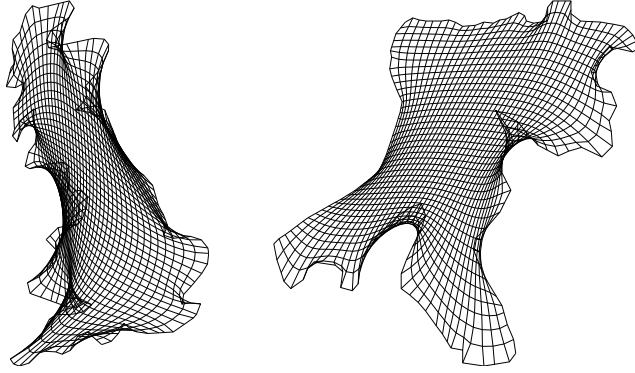


Figura 2.10: Mallas generadas usando el funcional continuo de longitud para las regiones ENG y HAB.

2.3.3. Funcional de Área

El funcional continuo de área Φ_A , en su versión más simple, está diseñado para producir mallas tales que la variación del jacobiano sea lo menor posible en el sentido de cuadrados mínimos; y se escribe

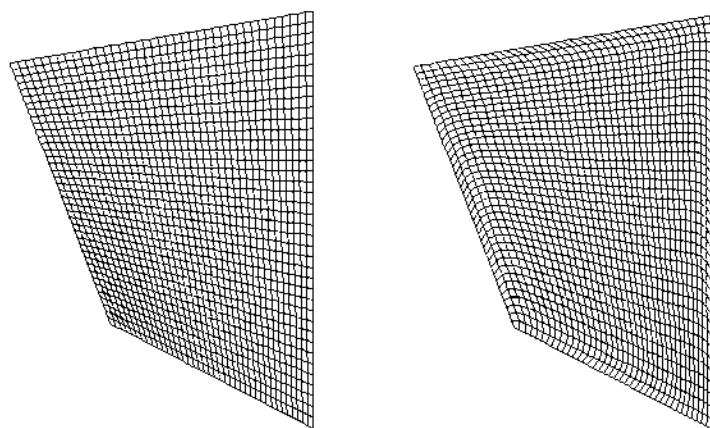
$$\begin{aligned}\Phi_A(\mathbf{x}) &= \int_0^1 \int_0^1 (x_\xi y_\eta - x_\eta y_\xi)^2 d\eta d\xi \\ &= \int_0^1 \int_0^1 J^2 d\eta d\xi\end{aligned}$$

Las ecuaciones de Euler-Lagrange para este funcional son más complicadas que las correspondientes al funcional de longitud,

$$\begin{aligned}y_\eta^2 x_{\xi\xi} - x_\eta y_\eta y_{\xi\xi} - 2y_\xi y_\eta x_{\xi\eta} + (x_\xi y_\eta + x_\eta y_\xi) y_{\xi\eta} - y_\xi^2 x_{\eta\eta} - x_\xi y_\eta y_{\eta\eta} &= 0 \\ -x_\eta y_\eta x_{\xi\xi} + x_\xi^2 y_{\xi\xi} + (x_\xi y_\eta + x_\eta y_\xi) y_{\xi\eta} - 2x_\xi x_\eta y_{\xi\eta} - x_\xi y_\xi x_{\eta\eta} + x_\xi^2 y_{\eta\eta} &= 0\end{aligned}$$

y además son un sistema cuasilineal, acoplado y no elíptico. No hay hasta la fecha resultados de existencia o unicidad de las soluciones de este sistema. Las mallas producidas con este funcional no son suaves y, salvo en casos simples, son dobladas.

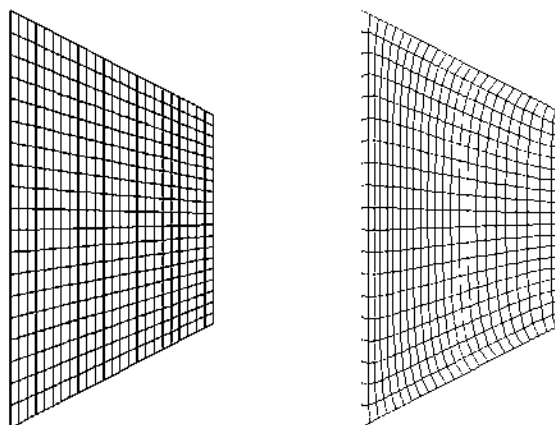
El algoritmo de solución descrito en [24], utilizado para la optimización de las mallas, no convergió para ninguna de las mallas de prueba. Por esta razón, y con fines ilustrativos, se muestran algunas mallas producidas por el funcional de área en dos regiones simples (véase figuras 2.11 y 2.12), en ambos casos, las mallas iniciales ya son convexas, así lo son las mallas óptimas y se puede notar que las celdas son muy uniformes en área.



Malla inicial generada con TFI

Malla óptima

Figura 2.11: Mallas generadas usando el funcional continuo de área para CUA y TRA.



Malla inicial generada con TFI

Malla óptima

Figura 2.12: Mallas generadas usando el funcional continuo de área para CUA y TRA.

2.3.4. Funcional de Ortogonalidad

Si lo que queremos es obtener una malla con líneas coordenadas lo más ortogonales posible entonces utilizamos el funcional de ortogonalidad. Esto es equivalente a pedir que las tangentes a dichas líneas coordenadas sean ortogonales en cada punto, cosa que podemos describir por la relación

$$O = \mathbf{x}_\xi^t \mathbf{x}_\eta = 0$$

Para la mayoría de las regiones no existe una malla que tenga esta característica; se trata entonces de calcular una lo más cercana posible a la ortogonalidad. Una manera de lograr esto es calcular una malla que satisfaga las condiciones de ortogonalidad en el sentido de los cuadrados mínimos; esto quiere decir que se calcula el mínimo de

$$\begin{aligned} \Phi_O(\mathbf{x}) &= \int_0^1 \int_0^1 (\mathbf{x}_\xi^t \mathbf{x}_\eta)^2 d\eta d\xi \\ &= \int_0^1 \int_0^1 (x_\xi x_\eta + y_\xi y_\eta)^2 d\eta d\xi \end{aligned}$$

cuyas ecuaciones de Euler-Lagrange son:

$$\begin{aligned} x_\eta^2 x_{\xi\xi} + x_\eta y_\eta y_{\xi\xi} + (4x_\xi x_\eta + 2y_\xi y_\eta) x_{\xi\eta} + (x_\xi y_\eta + x_\eta y_\xi) y_{\xi\eta} \\ + x_\xi^2 x_{\eta\eta} + x_\xi y_\xi y_{\eta\eta} = 0 \end{aligned}$$

$$\begin{aligned} x_\eta y_\eta x_{\xi\xi} + y_\eta^2 y_{\xi\xi} + (x_\xi y_\eta + x_\eta y_\xi) x_{\xi\eta} + (4y_\xi y_\eta + 2x_\xi x_\eta) y_{\xi\eta} \\ + x_\xi y_\xi x_{\eta\eta} + y_\xi^2 y_{\eta\eta} = 0 \end{aligned}$$

las cuales son nuevamente cuasilineales, acopladas y no elípticas. Muy frecuentemente su proceso de solución no converge, y cuando lo hace las mallas generadas son, a menudo, severamente dobladas. Es por esta razón que el usar este funcional de manera aislada no es recomendable.

2.3.5. Combinación de Funcionales

Knupp y Steinberg [24] introducen el uso de combinaciones convexas de los funcionales anteriores con el objetivo de lograr mallas que gocen, en cierta medida, de las ventajas que ofrece cada funcional por separado. La forma general de esta combinación es

$$\Phi(\mathbf{x}) = \sigma_l \Phi_l(\mathbf{x}) + \sigma_A \Phi_A(\mathbf{x}) + \sigma_O \Phi_O(\mathbf{x})$$

en donde $\sigma_l, \sigma_A, \sigma_O \geq 0$ y $\sigma_l + \sigma_A + \sigma_O = 1$. La elección de los parámetros σ_l, σ_A y σ_O depende fuertemente de la experiencia personal y ésta es una gran limitante para el uso de esta combinación.

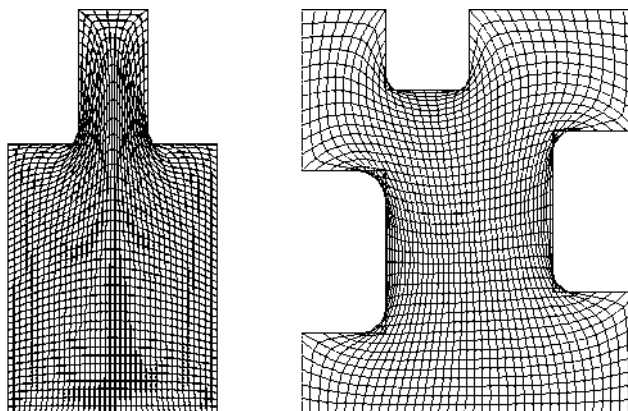


Figura 2.13: Mallas generadas usando el funcional continuo de área-longitud para M13 y M19

Malla	Celdas no convexas	σ_A
M13	14	.80
M19	44	.50

Tabla 2.3: Número de celdas no convexas usando el funcional continuo de área-longitud

Funcionales de Área-Longitud

Las combinaciones con $\sigma_O = 0$, que generan los llamados funcionales de área-longitud son muy usados, ya que para regiones no convexas y geoméricamente no muy complicadas producen mallas convexas (no dobladas) y aceptablemente suaves; sin embargo en la mayoría de las regiones no simples la convexidad no logra ser alcanzada. Para las regiones de prueba *HAB* y *ENG*, el algoritmo de solución de las ecuaciones diferenciales, mostrado en [24], no convergió. Se muestran entonces, en la figura 2.13, las mallas obtenidas con un funcional de área-longitud para las regiones *M13* y *M19*. La tabla (2.3) muestra el número de celdas no convexas para estos ejemplos, así como los valores de σ_A correspondientes.

Funcionales de Área-Ortogonalidad

Un generador que sí es automático, o sea que no depende de los parámetros, se obtiene utilizando los valores

$$\sigma_l = 0 \quad \sigma_A = 1/2 \quad \sigma_O = 1/2$$

Malla	Celdas no convexas
M13	2
M19	2
ENG	80
HAB	12

Tabla 2.4: Número de celdas no convexas usando el funcional continuo de área-ortogonalidad

que da lugar al llamado funcional de área-ortogonalidad de Knupp. Éste tiene la forma

$$\begin{aligned}
\Phi_{AO} &= \frac{1}{2} \int_0^1 \int_0^1 [J^2 + O^2] d\eta d\xi \\
&= \frac{1}{2} \int_0^1 \int_0^1 \left[(x_\xi y_\eta - x_\eta y_\xi)^2 + (x_\xi x_\eta + y_\xi y_\eta)^2 \right] d\eta d\xi \\
&= \frac{1}{2} \int_0^1 \int_0^1 (x_\xi^2 y_\eta^2 + x_\eta^2 y_\xi^2 + x_\xi^2 x_\eta^2 + y_\xi^2 y_\eta^2) d\eta d\xi \\
&= \frac{1}{2} \int_0^1 \int_0^1 [(x_\xi^2 + y_\xi^2) (x_\eta^2 + y_\eta^2)] d\eta d\xi
\end{aligned}$$

esto es, L se puede interpretar como producto de los cuadrados de los segmentos "verticales" y "horizontales".

Las ecuaciones de Euler-Lagrange correspondientes para este funcional son

$$\begin{aligned}
(x_\eta^2 + y_\eta^2)x_{\xi\xi} + 4x_\xi x_\eta x_{\xi\eta} + 2(x_\xi y_\eta + x_\eta y_\xi)y_{\xi\eta} + (x_\xi^2 + y_\xi^2)x_{\eta\eta} &= 0 \\
(x_\eta^2 + y_\eta^2)y_{\xi\xi} + 2(x_\xi y_\eta + x_\eta y_\xi)x_{\xi\eta} + 4y_\xi y_\eta y_{\xi\eta} + (x_\xi^2 + y_\xi^2)y_{\eta\eta} &= 0
\end{aligned}$$

las cuales no siempre son elípticas.

Las mallas producidas por el funcional de área-ortogonalidad son suaves y convexas, si la región es relativamente siempre, se tiene una variación relativamente baja del jacobiano y son casi ortogonales. En la figuras (2.14 y 2.15) se muestran algunas mallas optimas logradas con este funcional. En la tabla (2.4) se muestra el número de celdas no convexas para cada una de ellas.

2.3.6. Funcional de Suavidad

El funcional de longitud definido para la transformación inversa

$$\xi(\mathbf{x}) = \begin{pmatrix} \xi(x, y) \\ \eta(x, y) \end{pmatrix}$$

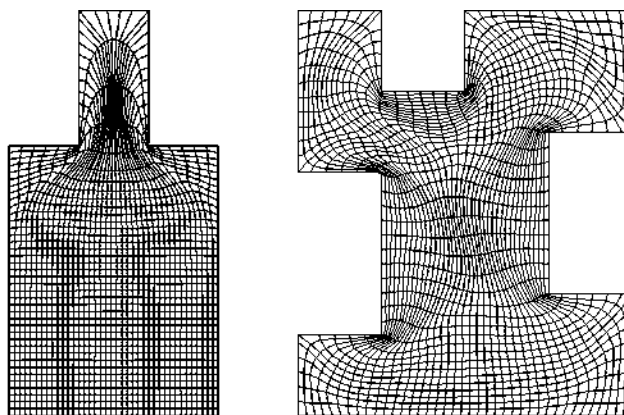


Figura 2.14: Mallas generadas usando el funcional continuo de área-ortogonalidad para M13 y M19.

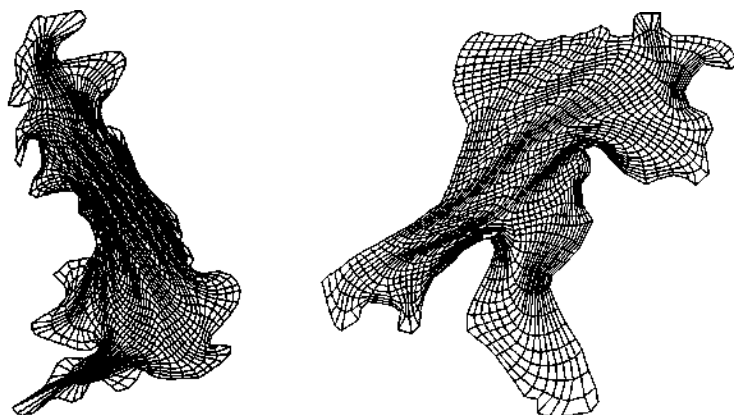


Figura 2.15: Mallas generadas usando el funcional continuo de área-ortogonalidad para ENG y HAB.

$$\xi : \Omega \rightarrow U_2$$

está dado por

$$\int \int_{\Omega} (\xi_x^2 + \xi_y^2 + \eta_x^2 + \eta_y^2) dy dx$$

el cual, al hacer el cambio de coordenadas hacia el espacio lógico, se escribe como

$$\Phi_s(\mathbf{x}) = \int_0^1 \int_0^1 \frac{x_\xi^2 + x_\eta^2 + y_\xi^2 + y_\eta^2}{J} d\eta d\xi$$

Este funcional es el funcional de suavidad de Winslow, también referido como el funcional Thomas-Thames-Mastin homogéneo (TTM). Las ecuaciones de Euler-Lagrange para este funcional son las ecuaciones de Laplace para la transformación inversa

$$\begin{aligned} \xi_{xx} + \xi_{yy} &= 0 \\ \eta_{xx} + \eta_{yy} &= 0 \end{aligned}$$

que, a pesar de su simplicidad, no son fácilmente resueltas de esta manera. En lugar de éstas, se resuelven las correspondientes al mapeo directo, las cuales son cuasilineales y acopladas

$$\begin{aligned} g_{22}x_{\xi\xi} - 2g_{12}x_{\xi\eta} + g_{11}x_{\eta\eta} &= 0 \\ g_{22}y_{\xi\xi} - 2g_{12}y_{\xi\eta} + g_{11}y_{\eta\eta} &= 0 \end{aligned}$$

donde

$$\begin{aligned} g_{11} &= x_\xi^2 + y_\xi^2 \\ g_{12} &= x_\xi x_\eta + y_\xi y_\eta \\ g_{22} &= x_\eta^2 + y_\eta^2 \end{aligned}$$

Este funcional, al igual que los anteriores, tiene la desventaja de que no siempre converge a una malla convexa. Sin embargo debe hacerse la aclaración que el uso de éste funcional usualmente se restringe a mejorar, en lo que a suavidad se refiere, una malla que se ha optimizado por otros métodos. En este sentido solo se realizan pocas iteraciones del proceso de este optimización con este funcional. En las figuras (2.16 y 2.17) se muestran algunas mallas generadas con este funcional. El número de celdas no convexas para cada una de las mallas se muestra en la tabla (2.5).

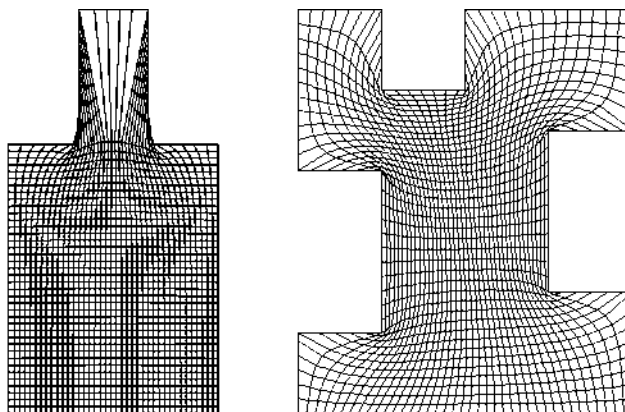


Figura 2.16: Mallas generadas usando el funcional continuo de suavidad para las regiones m13 y m19.

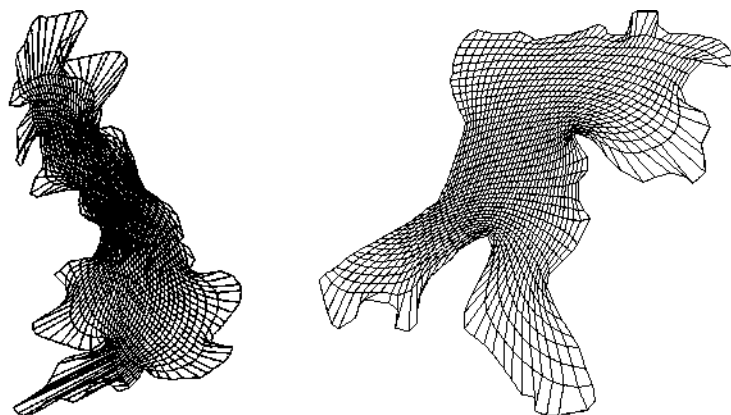


Figura 2.17: Mallas generadas usando el funcional continuo de suavidad para las regiones ENG y HAB.

Malla	Celdas no convexas
M13	6
M19	11
ENG	64
HAB	21

Tabla 2.5: Número de celdas no convexas usando el funcional continuo de suavidad.

2.4. Generación Variacional Discreta

En los métodos variacionales continuos la generación de mallas se realiza resolviendo las ecuaciones de Euler-Lagrange correspondientes al funcional a minimizar. Esta solución se realiza de manera numérica eligiendo una malla sobre el cuadrado unitario; es decir,

$$(\xi_i, \eta_j) \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n$$

y aplicando un esquema de diferencias finitas para obtener aproximaciones $P_{i,j}$ a los valores de $\mathbf{x}(\psi_i, \eta_j)$.

Varios autores (entre ellos Barrera [1], [4], Ivanenko [21], [22] y Castillo [5], [6], [7], [8], [9]) proponen realizar primeramente una discretización del funcional a minimizar, y después resolver el correspondiente problema de optimización multivariada. En otras palabras, el funcional continuo

$$\Phi(\mathbf{x}) = \int_0^1 \int_0^1 L(x_\xi, x_\eta, y_\xi, y_\eta) d\eta d\xi$$

es reemplazado por una función de $m \times n$ variables

$$\{P_{i,j}\} \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n$$

$$F(P_{1,1}, P_{1,2}, \dots, P_{i,n}, P_{2,1}, P_{2,2}, \dots, P_{2,n}, \dots, P_{m,1}, P_{m,2}, \dots, P_{m,n}) = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} f_{i,j}$$

donde $f_{i,j}$ es una aproximación al valor de L en la celda de $[0, 1] \times [0, 1]$ definida por los puntos

$$(\xi_{i-1}, \eta_{j-1}), (\xi_{i-1}, \eta_j), (\xi_i, \eta_j), (\xi_i, \eta_{j-1}).$$

Los puntos

$$\{P_{1,j}\}, \{P_{m,j}\} \quad j = 1, 2, \dots, n \quad \{P_{i,1}\}, \{P_{i,n}\} \quad i = 1, 2, \dots, m$$

son los valores correspondientes del mapeo $\mathbf{x}$ en los puntos de la frontera del cuadrado unitario

$$\begin{aligned} &\{\xi_1, \eta_j\}, \quad \{\xi_m, \eta_j\}, \quad j = 1, 2, \dots, n \\ &\{\xi_i, \eta_1\}, \quad \{\xi_i, \eta_n\}, \quad i = 1, 2, \dots, m \end{aligned}$$

que están fijos y los puntos interiores

$$\{P_{i,j}\} \quad i = 2, \dots, m-1 \quad j = 2, \dots, n-1$$

que son las imágenes bajo el mapeo buscado $\mathbf{x}$ de los puntos del cuadrado unitario

$$\{(\xi_i, \eta_j)\} \quad i = 2, \dots, m-1 \quad j = 2, \dots, n-1$$

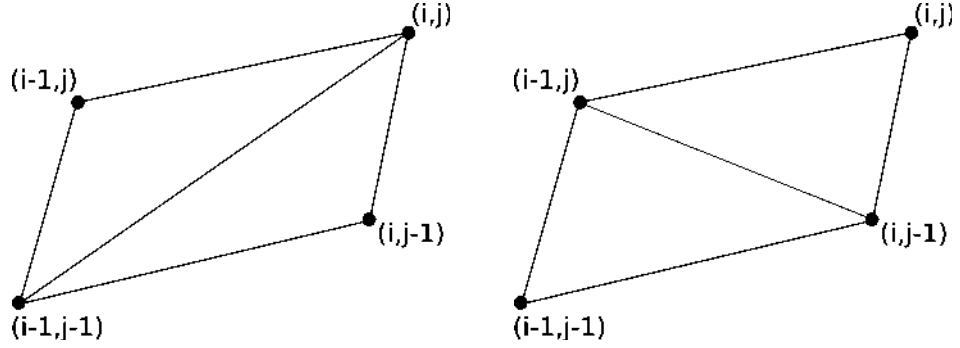


Figura 2.18: Formación de los triángulos relevantes.

se calculan de tal forma que hagan mínimo a F .

Esta minimización de F se realiza partiendo de una malla inicial, generalmente generada por TFI. Se usa un método de optimización multivariada de gran escala, por ejemplo *L-BFGS* ó *Newton Truncado*. Cabe mencionar que aunque los métodos discretos clásicos se obtienen a partir de la discretización del funcional continuo correspondiente, también se puede pensar en generar directamente funcionales discretos; o en otras palabras, proponer una cantidad geométrica que se quiera minimizar sobre cada celda de la malla.

2.4.1. Funcional Discreto de Longitud

La forma más simple del funcional discreto de longitud es

$$F_l = \frac{1}{2N\bar{\alpha}} \sum_{q=1}^N l(\Delta_q) = \frac{1}{2N\bar{\alpha}} \sum_{q=1}^N l_q$$

donde $N = 4(m-1)(n-1)$ es el número de triángulos relevantes en la malla (Figura 2.18), m y n representan el tamaño de la malla y $\bar{\alpha}$ es el doble del área promedio de todos los triángulos. Este funcional es usado debido a los buenos resultados que otorga al ser combinada con otros funcionales que comentaremos más adelante. Una versión más general es propuesta en [1] y [4].

F_l es un funcional cuadrático, lo que conlleva a que su optimización sea muy eficiente. También tiene la propiedad de que su hessiana es definida positiva en todas partes [4], lo cual garantiza la existencia de un único mínimo.

En las figuras (2.19) y (2.20) se muestran las mallas generadas para las regiones de prueba minimizando este funcional, tomando como malla inicial una generada por el método TFI en su versión discreta. En la tabla (2.6) se muestran el número

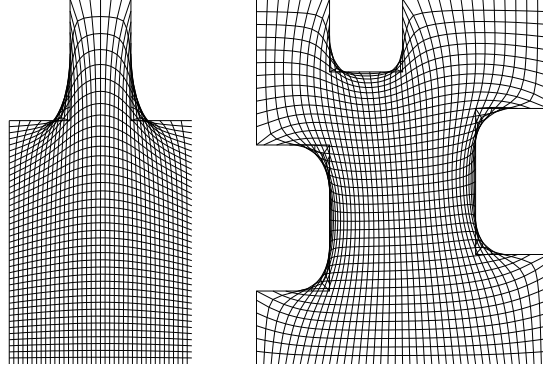


Figura 2.19: Mallas generadas usando el funcional discreto de longitud para M13 y M19.

Malla	Celdas no convexas	Iteraciones
M13	42	5
M19	81	5
ENG	252	1
HAB	91	5

Tabla 2.6: Número de celdas no convexas usando el funcional discreto de longitud.

de celdas no convexas y la cantidad de iteraciones necesarias para convergencia numérica.

2.4.2. Funcional Clásico de Área en su Versión Discreta

De acuerdo a la formulación anterior, el funcional clásico de área, en su versión discreta, es

$$F_{A^2} = \frac{1}{N\bar{\alpha}^2} \sum_{q=1}^N \alpha_q^2.$$

Sean Ω, m, n fijos, este funcional tiene las siguientes propiedades:

1. Si existe una malla G tal que los triángulos de todas las celdas tienen la misma área, entonces G es un punto crítico para el funcional de área.
2. La Hessiana del funcional de área valuada en G es semipositiva definida.

La demostración de estas propiedades se puede consultar en [4]. Una forma más general de estos resultados se expone en [28]. Las propiedades anteriores garantizan que, en caso de existir una malla ideal para el funcional de área, su

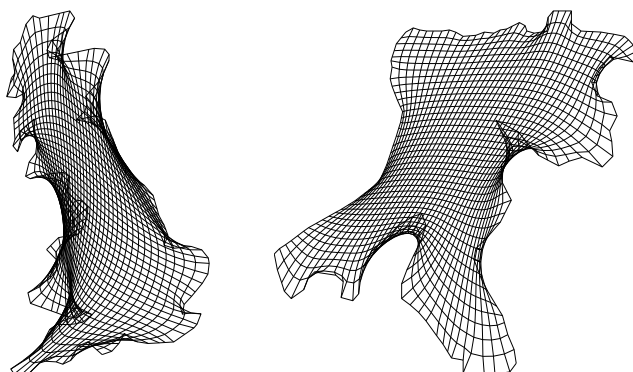


Figura 2.20: Mallas generadas usando el funcional discreto de longitud para ENG y HAB.

Malla	Celdas no convexas	Iteraciones
M13	0	43
M19	1	108
ENG	31	180
HAB	6	126

Tabla 2.7: Número de celdas no convexas usando el funcional discreto de área.

minimización conduce a ella; en caso de no existir dicha malla ideal, el mínimo estará en una malla que sea lo más cercana posible a tener triángulos de igual área. La existencia de una malla ideal para este funcional de área depende fuertemente de la frontera de la región y de la distribución de los puntos sobre dicha frontera. Una malla ideal G^* para el funcional de área es aquella en la cual las áreas de todos los triángulos son iguales.

El proceso de minimización del funcional F_{A^2} siempre converge. Los resultados obtenidos sobre las mallas de prueba minimizando este funcional en la tabla (2.7). Las imágenes mostradas en las figuras (2.21) y (2.22) son las mallas generadas mediante la optimización de este funcional. Podemos observar que las mallas están lejos de ser suaves, pero tienen menos celdas no convexas.

2.4.3. Funcional Discreto de Ortogonalidad

Aunque el uso de la versión discreta del funcional de ortogonalidad no conduce a buenas mallas cuando se usa de manera aislada para casi ninguna región, su

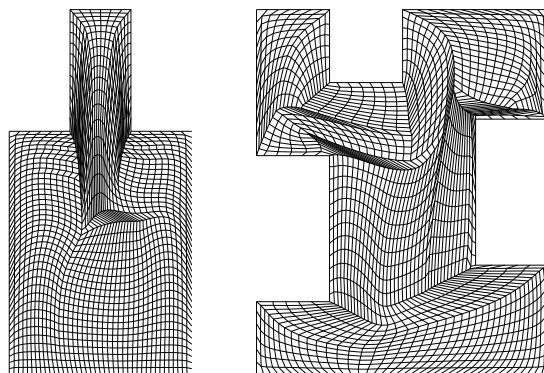


Figura 2.21: Mallas generadas usando el funcional discreto de área para M13 y M19.

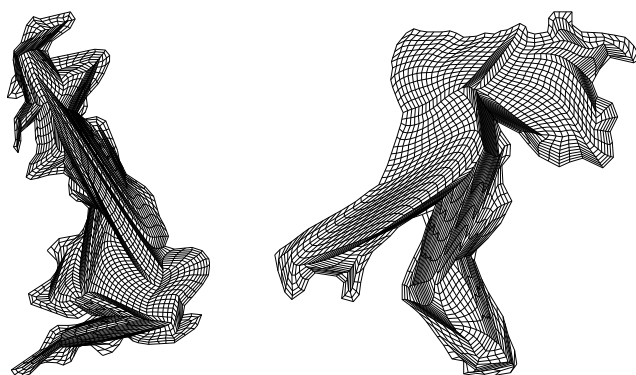


Figura 2.22: Mallas generadas usando el funcional discreto de área para ENG y HAB.

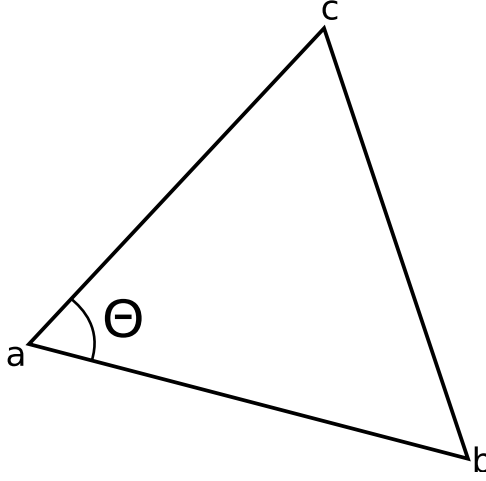


Figura 2.23: Triángulo con vértices c, a, b y ángulo Θ

combinación con otros funcionales sí lo hace. La forma que tiene es

$$F_{\mathcal{O}} = \frac{1}{N\bar{\alpha}^2} \sum_{q=1}^N \mathcal{O}(\Delta_q)^2 = \frac{1}{N\bar{\alpha}^2} \sum_{q=1}^N \mathcal{O}_q^2.$$

donde $\mathcal{O}_q^2$ es la discretización del integrando en el funcional continuo de ortogonalidad

$$o^2 = \|c - a\|^2 \|b - a\|^2 \cos^2 \theta.$$

Una malla ideal para el funcional de ortogonalidad es aquella en la que las celdas son rectángulos. Utilizando este funcional no se obtuvo ninguna malla convexa para las regiones de prueba, y además el número de iteraciones realizado fue alto.

2.4.4. Funcionales Combinados

También en el enfoque discreto es conveniente usar combinaciones convexas de diferentes funcionales para producir mallas más adecuadas. La forma general de esta combinación es

$$F = \tau_1 F_{A^2} + \tau_2 F_l + (1 - \tau_1 - \tau_2) F_o$$

donde $0 \leq \tau_1, \tau_2 \leq 1$ y $\tau_1 + \tau_2 \leq 1$. Los funcionales se usan en su forma normalizada.

Funcional de Área-Longitud

Cuando se hace $\tau_1 = \tau$, $\tau_2 = 1 - \tau$ obtenemos la familia de funcionales

$$F = \tau F_{A^2} + (1 - \tau) F_l \quad 0 \leq \tau \leq 1$$

Malla	τ	Celdas no convexas	Iteraciones
M13	.75	14	17
	.95	6	28
	.99	0	47
M19	.75	8	23
	.95	2	4
	.99	0	82
HAB	.75	42	30
	.95	13	62
	.99	11	126
ENG	.75	128	36
	.95	58	111
	.99	29	114

Tabla 2.8: Resultados para el Funcional de Area-Longitud

que con elecciones adecuadas de τ (usualmente con τ cercano a 1) dan lugar a mallas que son aceptables para la mayoría de las regiones sencillas. Es cuestión de experiencia el elegir el valor de τ y lograr buenas mallas en los casos en los que la región es complicada; esto hace que el usar los funcionales de área-longitud sea complicado aún para los expertos en generación de mallas. Se hicieron diferentes pruebas, utilizando diferentes valores de τ , los resultados se muestran en la tabla (2.8). En las figuras (2.24) y (2.25) se muestran también las gráficas correspondientes a $\tau = 0,95$ de las regiones de prueba. Se observa que éstas son más suaves que las correspondientes al funcional de área, pero tienen más celdas no convexas.

Funcional Discreto de Área-Ortogonalidad

Al hacer $\tau_2 = 0$ se obtiene la familia conocida como funcionales de Área-Ortogonalidad. La forma de estos funcionales es

$$F_{ao}^{(\tau)} = \tau_1 F_{A^2} + (1 - \tau_1) F_o.$$

Cuando se habla del *Funcional de Área-Ortogonalidad*, sin especificar el valor del parámetro τ , se hace referencia al miembro de esta familia correspondiente al valor $\tau = .5$. La forma que toma para un triángulo Δcab es

$$\begin{aligned}
F_{ao} &= \frac{1}{2}(\alpha_q^2 + o_q^2) \\
&= \frac{1}{2}(\|c - a\|^2 \|b - a\|^2 \cos^2 \theta + \|c - a\|^2 \|b - a\|^2 \sin^2 \theta) \\
&= \frac{1}{2}(\|c - a\|^2 \|b - a\|^2)
\end{aligned}$$

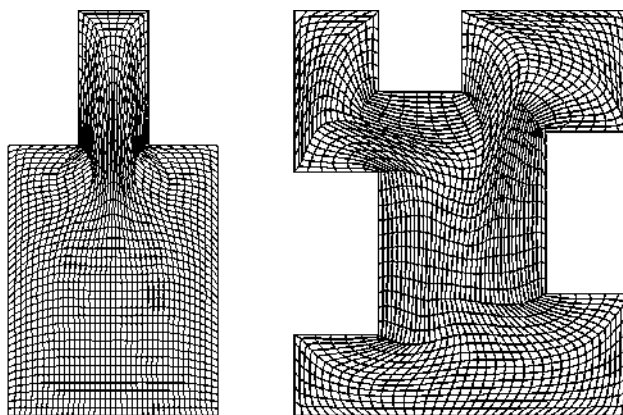


Figura 2.24: Mallas generadas usando el funcional discreto de área-longitud para M13 y M19.

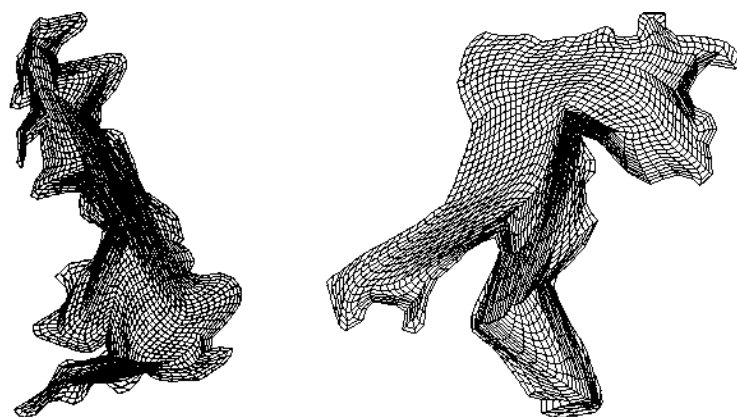


Figura 2.25: Mallas generadas usando el funcional discreto de área-longitud para ENG y HAB.

Malla	τ	Celdas no convexas	Iteraciones
M13	.50	2	33
	.75	2	24
	.95	0	54
M19	.50	2	21
	.75	1	30
	.95	2	48
HAB	.50	32	46
	.75	11	38
	.95	13	62
ENG	.50	115	60
	.75	66	79
	.95	58	194

Tabla 2.9: Resultados para el Funcional de Area-Ortogonalidad

A pesar de ser una combinación de funcionales que no producen mallas suaves, los funcionales de Área-Ortogonalidad dan lugar a mallas relativamente suaves, las áreas de los triángulos tienen poca variación y el número de iteraciones menor que para el funcional de área; sin embargo, en la mayoría de las regiones de prueba no se logra obtener mallas convexas. En la tabla (2.9) se muestran resultados obtenidos sobre las mallas de prueba para diferentes valores de τ . Las gráficas de las mallas que se muestran en las figuras (2.26) y (2.27) corresponden a este funcional.

2.5. Funcionales de Suavidad y Área adaptivos de Tinoco

El objetivo principal a conseguir en la generación de mallas sobre regiones planas irregulares es el que la malla sea convexa; una malla que no cumpla con este requisito es completamente inútil para los propósitos que se persiguen. Los métodos revisados hasta ahora no garantizan que el objetivo se logre; en muchos casos la malla óptima es no convexa. Sería deseable contar con un funcional tal que el proceso de su minimización conduca siempre a una malla convexa y que se tengan resultados que garanticen que eso siempre se logra.

Para precisar la idea de optimización de una malla, considérese Ω una región plana; $m, n \in \mathbb{N}$ y los correspondientes puntos frontera

$$\{P_{1,j}\}_{j=1,\dots,m} \quad \{P_{m,j}\}_{j=1,\dots,n} \quad \{P_{i,1}\}_{i=1,\dots,m} \quad \{P_{i,n}\}_{i=1,\dots,m}$$

también fijos.

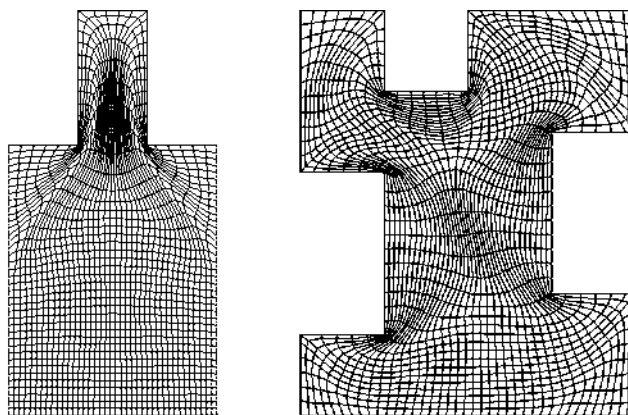


Figura 2.26: Mallas generadas usando el funcional discreto de área-Ortogonalidad para M13 y M19.

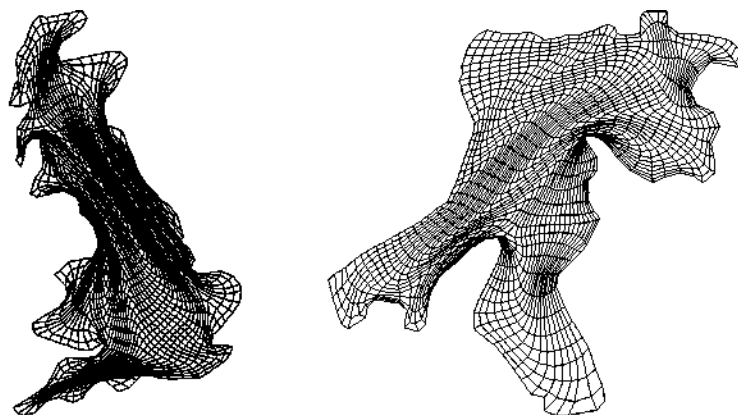


Figura 2.27: Mallas generadas usando el funcional discreto de área-Ortogonalidad para ENG y HAB.

En [28] se define, para una malla G sobre Ω , la cantidad

$$\alpha_-(G) = \min_{\Delta \in G} \{\alpha(\Delta)\}$$

la cual es el doble del mínimo de las áreas de los triángulos en la malla. Ahora definimos, para k real, el conjunto

$$D_k = \{G : G \in M \text{ y } \alpha_-(G) > -k\}$$

donde M es el conjunto de mallas definidas sobre Ω . Obviamente si $k < k'$, se da la contención $D_k \subset D_{k'}$, y además si

$$k \leq -\bar{\alpha} = -\frac{\text{Area}(\Omega)}{(m-1)(n-1)}$$

entonces $D_k = \emptyset$. Así pues, el conjunto de las k tales que

$$D_k \neq \emptyset$$

es un conjunto de números reales, no vacío y acotado inferiormente. Sea

$$k_- = \inf\{k : D_k \neq \emptyset\}$$

si $k_- < 0$ entonces el conjunto de las mallas convexas sobre Ω , D_0 , es no vacío.

El proceso de optimización de un funcional comienza con una malla G_0 tal que $\alpha_-(G_0) > k_0$; esto es $G_0 \in D_{k_0}$. Si las mallas producidas en cada paso de esta optimización son consideradas como los puntos de una trayectoria en el conjunto de mallas sobre Ω , se puede parafrasear nuestro objetivo de generación de mallas convexas como el de calcular una trayectoria

$$\Gamma : [-k_0, 0] \rightarrow M$$

tal que

$$\Gamma(-k_0) = G_0$$

y

$$\Gamma(0) = G \in D_0$$

si es que $D_0 \neq \emptyset$ (fig. 2.28).

En caso de que no exista una malla convexa sobre Ω sería deseable de cualquier manera que el punto dinas de dicha trayectoria fuera lo más cercano posible a la convexidad; esto es, se desea ser capaz de obtener una malla de D_k con k lo más pequeño posible.

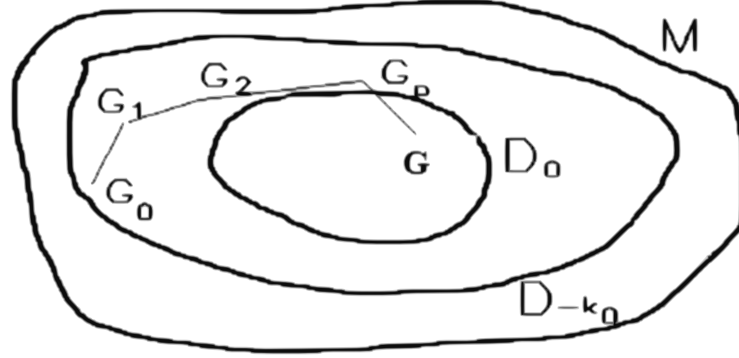


Figura 2.28: Trayectoria hacia una malla convexa.

2.5.1. El Funcional de Suavidad Adaptiva

Anteriormente fue definido el funcional discreto de suavidad como

$$\sum_{q=1}^N \frac{l_q}{\alpha_q}.$$

Una de las metas deseables es la de obtener funcionales en cuyo proceso de optimización no se requiera de condiciones de arranque muy estrictas y que puedan producir mallas suaves. Supóngase que se tiene una malla no convexa tal que el área de cada triángulo que contribuye al funcional es mayor que $-k/2$. Entonces la función

$$f_k = \frac{l}{k + \alpha}$$

toma valores positivos para cada triángulo en la malla. Esta función da lugar al funcional

$$F_s = \frac{1}{2N} \sum_{q=1}^N \frac{l}{k + \alpha}.$$

Minimizando este funcional se obtiene una malla que está más cerca de ser convexa que la original. Una adecuada actualización de k produce, en un número finito de pasos, una malla convexa si existe una [28].

2.5.2. El Funcional de Área Adaptiva

Al hacer una versión discreta del función área continuo explicado en [28] se obtiene el funcional discreto de área para una malla discreta G

$$F_{A^{-1}}(G) = \sum_{\Delta \in G} f_{A^{-1}}(\Delta) = \sum_{\Delta \in G} \frac{1}{\alpha(\Delta)}$$

Malla	Celdas no convexas	Iteraciones
M13	0	50
M19	0	55
HAB	0	85
ENG	0	210

Tabla 2.10: Resultados para el funcional de suavidad.

que, usando una numeración adecuada de triángulos puede ser escrito como

$$F_{A^{-1}}(G) = \sum_{q=1}^N \frac{1}{\alpha_q}.$$

Debemos notar que para que el proceso de optimización de $F_{A^{-1}}$ genere una malla convexa, se necesita que la malla inicial ya lo sea. Esta es una seria limitación para el uso de este funcional la cual, sin embargo, se podrá superar más adelante.

Modificación del Funcional

Motivados por el buen resultado obtenido en [28] al modificar el funcional de suavidad para ampliar su dominio de definición, se considerará el funcional discreto dado por

$$F_{A^{-1},k} = \frac{k + \bar{\alpha}}{N} \sum_{q=1}^N \frac{1}{\alpha_q + k}$$

el cual se puede optimizar si se parte de una malla G tal que $\alpha_-(G) > -k$ y producirá una malla óptima con la misma propiedad sobre α_- ; de hecho si $\hat{G}$ es la malla óptima y k se elige adecuadamente, $\alpha_-(\hat{G}) > \alpha_-(G)$.

2.5.3. Resultados

En las tablas (2.10) y (2.11) se muestran los resultados obtenidos con la optimización de los funcionales de suavidad y área respectivamente. Debe hacerse la aclaración que, en el caso del funcional de área, los resultados para las regiones de prueba se alcanzaron tomando como dato inicial la malla óptima para el funcional de longitud; esto fue necesario debido a que el proceso de optimización del funcional de área no convergió tomando como condición inicial la malla generada por TFI. Esto constituye una desventaja para este funcional la cual se puede superar usando combinaciones adecuadas con otros funcionales, como se explica más adelante. Las gráficas de las mallas óptimas correspondientes se muestran en las figuras (2.29), (2.30), (2.31) y (2.32).

Malla	Celdas no convexas	Iteraciones
M13	0	245
M19	0	107
HAB	0	281
ENG	0	500

Tabla 2.11: Resultados para el funcional de área.

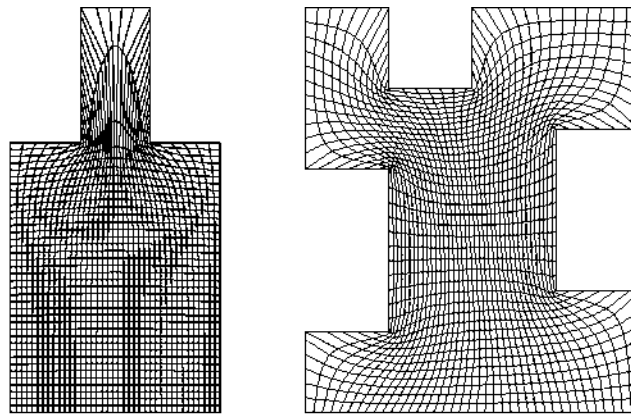


Figura 2.29: Mallas generadas usando el funcional suavidad para M13 y M19.

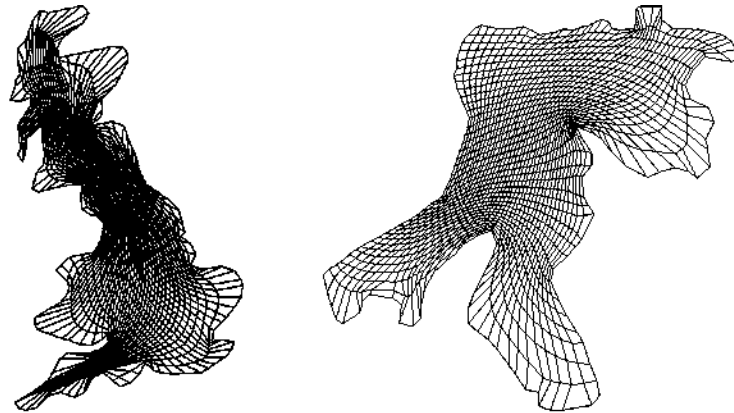


Figura 2.30: Mallas generadas usando el funcional de suavidad para ENG y HAB.

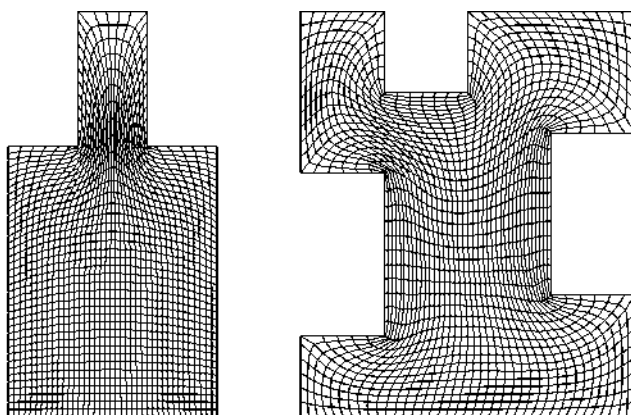


Figura 2.31: Mallas generadas usando el funcional de área para M13 y M19.

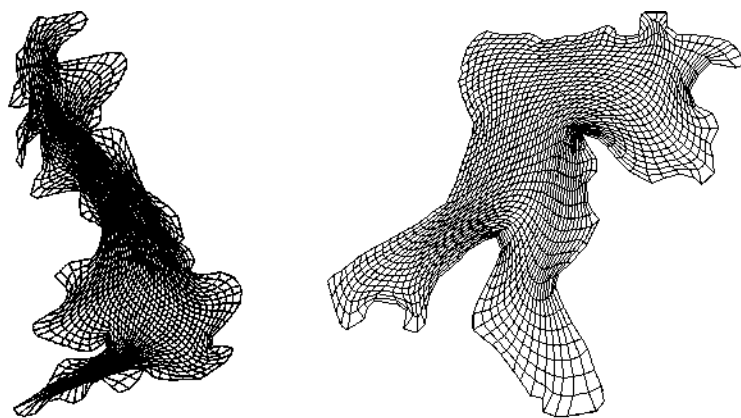


Figura 2.32: Mallas generadas usando el funcional de área para ENG y HAB.

2.5.4. Combinación de Funcionales

Los dos funcionales explicados en las secciones anteriores nos conducen a mallas óptimas con características muy diferentes. Por un lado, el funcional de suavidad nos proporciona una manera de obtener mallas muy suaves, pero el control de las áreas no es muy eficiente. Por otro lado el funcional de área produce un mejor control del tamaño de las celdas, pero distan mucho de proporcionar mallas óptimas suaves y el número de iteraciones necesarias para llegar al mínimo es muy grande.

Al igual que con los funcionales clásicos, se ha visto la conveniencia de combinar diferentes funcionales para tratar de conservar las ventajas de cada funcional por separado y disminuir sus inconvenientes. En este sentido, se presentan las combinaciones que han dado mejores resultados. En cada caso se usarán las formas normalizadas de los diferentes funcionales.

Combinaciones de Área con Longitud

Una manera de suavizar las mallas producidas con los funcionales de área es seguir la idea de usarlos en combinación con el funcional de longitud. Esto conlleva también a disminuir los tiempos necesarios para obtener convergencia.

El uso del funcional de área recíproca, además de los inconvenientes comunes a todos los funcionales de área ya mencionados, tiene la desventaja de que requiere, en muchos casos, que las mallas iniciales reciban un tratamiento con otros funcionales a fin de obtener convergencia. Sin embargo al combinarlo con el funcional de longitud, este inconveniente se ve superado en los casos estudiados. Esta combinación tiene la forma

$$F_{A^{-1}-l}^{(\tau)} = \tau F_{A^{-1}} + (1 - \tau) F_l \quad 0 \leq \tau \leq 1.$$

En la tabla 2.12 se muestran los resultados de optimizar algunos de estos funcionales partiendo de las mallas iniciales generadas con TFI. En particular se ha observado que el valor de $\tau = 0.75$ conduce a excelentes resultados. Las gráficas de las mallas que se muestran en las figuras 2.33 y 2.34 corresponden a este valor. Se siguió la estrategia TOPE para estos ejemplos.

Combinaciones de Suavidad con Área

Una manera de obtener mallas convexas y suaves que tengan una buena distribución del tamaño de las celdas es combinar el esquema adaptivo de los funcionales de suavidad con algún funcional de área. Ésta idea da buenos resultados para las combinaciones con el funcional clásico de área

$$F_{S-A}^{(\tau)} = \tau F_k + (1 - \tau) F_{A^2} \quad 0 \leq \tau \leq 1.$$

Los resultados de la optimización de las mallas iniciales de 41 por 41 usando los funcionales anteriores se presentan en la tabla (2.13). Las gráficas de las mallas que se presentan en las figuras (2.35) y (2.36) corresponden al valor de $\tau = 0.5$.

Malla	τ	Celdas no convexas	Iteraciones
M13	.50	0	37
	.75	0	32
	.95	0	70
M19	.50	0	69
	.75	0	49
	.95	0	60
HAB	.50	0	94
	.75	0	93
	.95	0	93
ENG	.50	0	164
	.75	0	184
	.95	0	266

Tabla 2.12: Resultados para el funcional adaptivo de área-longitud.

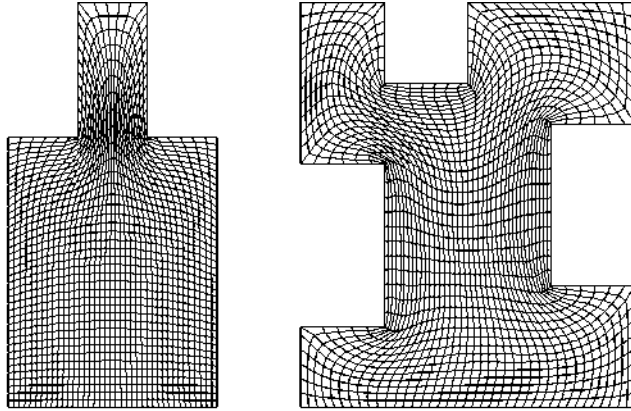


Figura 2.33: Mallas generadas usando el funcional adaptivo de área-longitud para M13 y M19.

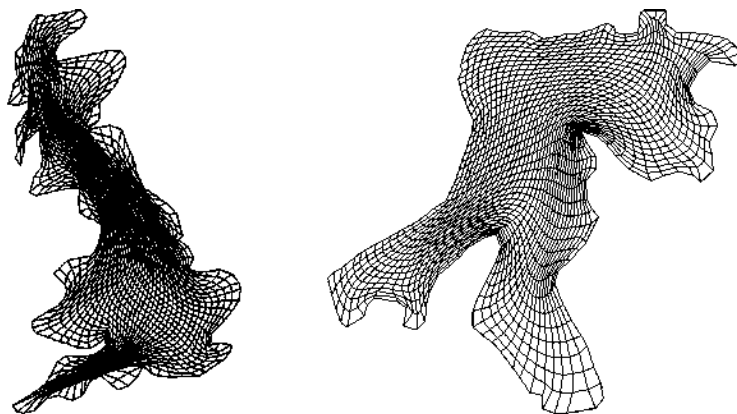


Figura 2.34: Mallas generadas usando el funcional adaptivo de área-longitud para ENG y HAB.

Malla	τ	Celdas no convexas	Iteraciones
M13	.25	0	63
	.50	0	54
	.75	0	46
M19	.25	0	62
	.50	0	59
	.75	0	57
HAB	.25	0	123
	.50	0	114
	.75	0	94
ENG	.25	0	217
	.50	0	207
	.75	0	203

Tabla 2.13: Resultados para el funcional Suavidad-Área $F_{SA} = \tau F_k + (1 - \tau) F_{A^2}$.

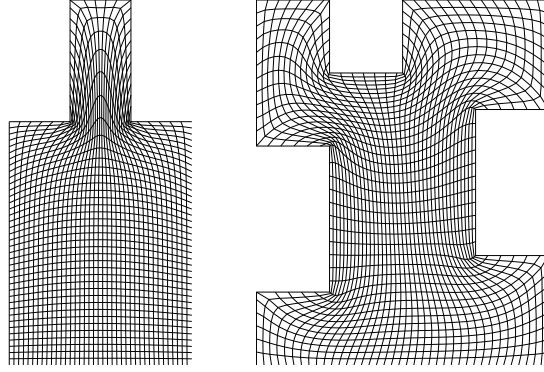


Figura 2.35: Mallas generadas usando el funcional Suavidad-Área $F_{SA} = \tau F_k + (1 - \tau)F_{A^2}$ para M13 y M19.

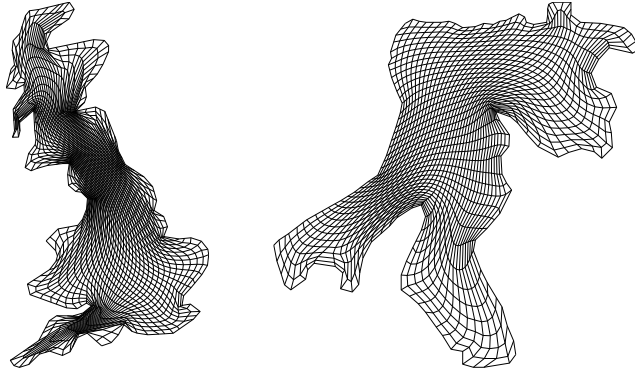


Figura 2.36: Mallas generadas usando el funcional Suavidad-Área $F_{SA} = \tau F_k + (1 - \tau)F_{A^2}$ para ENG y HAB.

2.6. Funcional Quasi-Armónico de Domínguez

Siguiendo las ideas mostradas hasta el momento Domínguez-Mota y Barrera-Sánchez posponen en [12] un nuevo funcional quasi-armónico que depende de un parámetro $\omega > 0$, basado en el funcional armónico de Ivanenko [23]

$$H(G) = \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^4 \frac{\lambda(\Delta_{i,j}^k)}{\alpha(\Delta_{i,j}^k)} = \sum_{q=1}^N \frac{\lambda(\Delta_q)}{\alpha(\Delta_q)},$$

donde m y n son el número horizontal y vertical de puntos de la malla, y k se usa para denotar los 4 triángulos formados por cada celda de la misma. En este caso N se utiliza para denotar el número total de triángulos de la malla, *i.e.* $N = 4mn$.

La idea principal es reemplazar α^{-1} en el funcional $H(G)$ de Ivanenko por una función capaz de actuar como una barrera continua y definiendo $f(\Delta q) = \lambda(\Delta q)\varphi_\omega(\alpha)$. Una opción para $\varphi_\omega(\alpha)$ es una función continua, convexa y estrictamente decreciente que tienda a $1/\alpha$ para $\alpha \geq \omega$. Entonces, se considera para un número real positivo ω

$$\varphi_\omega(\alpha) = \begin{cases} \frac{2\omega - \alpha}{\omega^2} & \text{si } \alpha < \omega \\ \frac{1}{\alpha} & \text{si } \alpha \geq \omega \end{cases}$$

y gracias a esta función se define el funcional H_ω como

$$H_\omega(G) = \sum_{q=1}^N \lambda(\Delta q)\varphi_\omega(\Delta q).$$

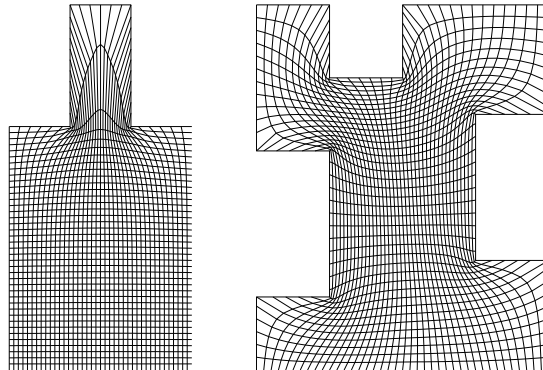
Un estudio más a fondo de este funcional, así como la implementación del mismo, pueden encontrarse en [12].

Este funcional produce mallas de muy buena calidad, para todas las regiones de prueba se pueden ver los resultados en la tabla (2.14); de igual manera en las figuras (2.37) y (2.38) podemos ver las gráficas de las mallas de las regiones de prueba con 41 puntos por lado generadas con este funcional. Podemos notar que para las regiones de prueba, el uso de éste funcional produce mallas convexas sin celdas con convexas, con buenas características de suavidad. En la práctica nos hemos dado cuenta que mezclar éste funcional con funcionales de área, además de producir mallas convexas, genera mallas con una buena distribución de áreas; esta combinación de funcionales será usada para generar todas las mallas del presente trabajo.

2.7. Conclusiones

Tras hacer un estudio breve sobre generación de mallas, los resultados mostrados nos confirman que las mallas con mejores características geométricas son

Malla	n	Celdas no convexas	Iteraciones
M13	21	0	17
	41	0	39
	81	0	143
M19	21	0	26
	41	0	37
	81	0	93
HAB	21	0	38
	41	0	80
	81	0	180
ENG	21	0	75
	41	0	166
	81	0	387

Tabla 2.14: Resultados para el funcional $H_\omega(G)$.Figura 2.37: Mallas generadas usando el funcional $H_\omega(G)$.

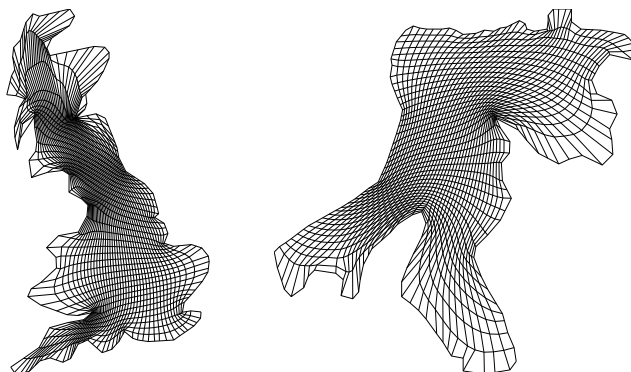


Figura 2.38: Mallas generadas usando el funcional $H_w(G)$.

aquellas generadas mediante la optimización de la combinación de 2 funcionales. En particular los funcionales de Suavidad-Área generan las mallas más suaves y con una distribución de área de las celdas más uniforme. Estas mallas serán utilizadas más adelante para resolver el problema principal de éste trabajo, que es la aproximación de la ecuación de advección.

Capítulo 3

Ecuación de Advección

La mecánica de fluidos gira en torno a la idea de partículas de un fluido que son transportadas en el tiempo y en el espacio; una de las ecuaciones importantes de esta disciplina es la *ecuación de advección*. La ecuación de advección es una ecuación diferencial parcial que muestra la variación de un campo escalar en un punto dado por el efecto de un campo vectorial. Por ejemplo el transporte de una sustancia contaminante que viaja por el caudal de un río. También en meteorología es usada para modelar el efecto del transporte de una propiedad atmosférica, como la temperatura o la humedad, por efecto del viento; al igual que en oceanografía se puede usar para modelar el transporte de la salinidad por efecto de las corrientes marinas. Se deriva de la *Ley de la conservación escalar* y del *Teorema de Gauss* cuando se toma el límite infinitesimal.

La ecuación de advección es, aunque no lo parezca, fácil de resolver analíticamente, pero su resolución numérica presenta una gran dificultad de solución, esto se debe a que es una ecuación diferencial parcial hiperbólica.

3.1. Advección en $1 + 1D$

Si una sustancia esta siendo transportada (o advectada) en un fluido a una velocidad constante a , entonces la función del flujo puede escribirse como

$$f(u) = au.$$

Entonces podemos saber el flujo de materia que pasa por el punto x simplemente multiplicando la densidad local $u(x, t)$ por la velocidad de transporte. Como el total de la masa que pasa entre x_1 y x_2 cambia únicamente debido al flujo en los puntos finales, podemos calcularla como

$$\frac{d}{dt} \int_{x_1}^{x_2} u(x, t) dx = f(u(x_1, t)) - f(u(x_2, t)), \quad (3.1)$$

el signo menos de esta ecuación viene del hecho de que f es, por definición, el flujo a la derecha.

Si asumimos que u y f son funciones *suaves*, i. e. tienen tantas derivadas como necesitemos, la ecuación (3.1) puede escribirse como

$$\int_{x_1}^{x_2} \left[\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) \right] dx = 0, \quad (3.2)$$

y debido a que la integral tiene que ser igual a 0 para todos los valores de x_1 y x_2 , entonces esto quiere decir que el integrando es igual a 0, y con esto obtenemos la ecuación diferencial

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} f(u(x, t)) = 0, \quad (3.3)$$

que escrita de esta manera es llamada una *ley de conservación escalar*, y es una ecuación usada ampliamente en diferentes campos de la física.

Para el problema que estamos tratando, el transporte de una sustancia en un fluido a una velocidad constante, tenemos que $f(u) = au$, donde a es una constante que representa la velocidad del fluido. Utilizando estos valores en la ecuación (3.3) obtenemos

$$\frac{\partial}{\partial t} u(x, t) + a \frac{\partial}{\partial x} u(x, t) = 0, \quad (3.4)$$

que es la ecuación de advección. Esta ecuación requiere condiciones iniciales y, en la mayoría de los casos, condiciones de frontera para que su solución pueda ser determinada. El caso más sencillo es el *problema de Cauchy* en $-\infty < x < \infty$, donde no hay frontera y por ello es llamado también un problema de condición inicial pura. Entonces sólo tenemos que especificar la información de inicio

$$u(x, 0) = \nu(x).$$

En el sentido físico, se espera simplemente que el perfil inicial de ν sea transportado con el fluido a una velocidad a , entonces debería cumplirse que

$$u(x, t) = \nu(x - at).$$

Es fácil verificar que esta ecuación satisface la ecuación de advección (3.4).

Cuando tenemos condiciones de frontera las cosas se complican un poco más, pero hoy en día existen, y pueden ser encontrados fácilmente, muchos métodos para aproximar numéricamente la solución de la ecuación. Este trabajo está enfocado únicamente a los esquemas de diferencias finitas para la resolución de este tipo de problemas, y sólo se enuncian algunos de los principales.

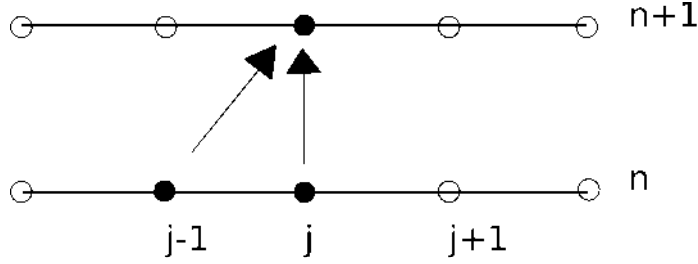


Figura 3.1: Un ejemplo de un stencil tipo Upwind

3.1.1. El método Upwind en $1 + 1D$

Si hacemos la expansión en serie de Taylor alrededor del punto (x_j, t_n) obtenemos

$$u(x_j, t_n + \Delta t) = u(x_j, t_n) + \frac{\partial u}{\partial t}(x_j, t_n) \Delta t + \mathcal{O}(\Delta t^2),$$

o, de manera equivalente, utilizando la notación del capítulo 1

$$u_j^{n+1} = u_j^n + \frac{\partial u}{\partial t}(x_j, t^n) \Delta t + \mathcal{O}(\Delta t^2).$$

De esta expresión podemos despejar $(\partial u / \partial t)(x_j, t_n)$, y así obtenemos una forma de aproximar la derivada respecto al tiempo

$$\frac{\partial u}{\partial t}(x_j, t^n) = \frac{u_j^{n+1} - u_j^n}{\Delta t} + \mathcal{O}(\Delta t). \quad (3.5)$$

Si ya contamos con una expresión para aproximar la derivada temporal de la ecuación (3.4) el paso lógico siguiente es encontrar una manera de aproximar la derivada espacial de dicha ecuación. Debido a que utilizaremos un método tipo “upwind” (ver fig. 3.1), la aproximación que necesitaremos dependerá del valor de a en (3.4), en el caso de éste trabajo nos centraremos en el caso en que a es un número positivo. Entonces siguiendo el mismo procedimiento que seguimos para encontrar (3.5) podemos encontrar la aproximación, basada en diferencias finitas, para la derivada espacial

$$\frac{\partial u}{\partial x}(x_j, t^n) = \frac{u_j^n - u_{j-1}^n}{\Delta x} + \mathcal{O}(\Delta x), \quad (3.6)$$

en este caso utilizaremos una malla uniforme tanto en el tiempo como en el espacio, *i.e.* Δt y Δx son constantes.

Ya que contamos con ambas expresiones para las derivadas parciales, entonces podemos sustituirlas en la expresión de la ecuación diferencial parcial original, para así obtener

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -a \left(\frac{u_j^n - u_{j-1}^n}{\Delta x} \right) + \mathcal{O}(\Delta t, \Delta x),$$

de esta última expresión, podemos despejar el término u_j^{n+1} , para así obtener el esquema tipo upwind para la ecuación de advección

$$u_j^{n+1} = u_j^n - a \frac{\Delta t}{\Delta x} (u_j^n - u_{j-1}^n). \quad (3.7)$$

Podemos analizar la estabilidad de este esquema haciendo el análisis de von Neumann. Para esto, escribimos la solución en la forma

$$u_j^n = \xi^n e^{ijkx} \quad (3.8)$$

donde k es el número de onda espacial. Si consideramos la representación simbólica del esquema como

$$u_j^{n+1} = \mathcal{T}(\Delta t^p, \Delta x^q) u_j^n,$$

con $\mathcal{T}(\Delta t^p, \Delta x^q)$ siendo el operador de evolución de orden p en el tiempo y q en el espacio, entonces podemos ver que el criterio de estabilidad debe ser aquel donde el módulo del factor de amplificación es siempre menor que 1

$$|\xi|^2 = \xi \xi^* \leq 1.$$

Para obtener el factor de amplificación hacemos

$$\xi^{n+1} e^{ikxj} = \xi^n e^{ikxj} - \alpha (\xi^n e^{ikxj} - \xi^n e^{ikxj-1})$$

de donde podemos obtener

$$\begin{aligned} \xi &= 1 - \alpha(1 - e^{-ik\Delta x}) \\ &= (1 - \alpha) + \alpha \cos(k\Delta x) - i\alpha \sin(k\Delta x). \end{aligned}$$

donde $\alpha = a\Delta t/\Delta x$, y su módulo $|\xi|^2 = \xi \xi^*$ es entonces

$$\begin{aligned} |\xi|^2 &= (1 - \alpha)^2 + 2\alpha(1 - \alpha) \cos(k\Delta x) + \alpha^2 \cos^2(k\Delta x) + \alpha^2 \sin^2(k\Delta x) \\ &= 1 - 2\alpha + \alpha^2 + 2\alpha(1 - \alpha) \cos(k\Delta x) + \alpha^2 \\ &= 1 - 2\alpha(1 - \alpha)(1 - \cos(k\Delta x)), \end{aligned}$$

que es siempre menor que uno cuando se satisface la *condición de Courant-Friedrichs-Löwy* (CFL)

$$\alpha \leq 1.$$

3.1.2. El esquema FTCS en 1 + 1D

Consideremos otra vez la ecuación de advección (3.4), pero ahora tomaremos una aproximación con más precisión para la derivada espacial. Para hacer esto, calculamos la expansión en Taylor en $x_j + \Delta x$

$$u(x_j + \Delta x, t^n) = u(x, t^n) + \frac{\partial u}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (x, t^n) \Delta x^2 + \mathcal{O}(\Delta x^3),$$

y también en $x_j - \Delta x$

$$u(x_j - \Delta x, t^n) = u(x, t^n) - \frac{\partial u}{\partial x} \Delta x + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (x, t^n) \Delta x^2 + \mathcal{O}(\Delta x^3).$$

Si restamos estas 2 expresiones y el resultado lo dividimos entre $2\Delta x$ podemos eliminar los términos de primer orden para obtener

$$\frac{\partial u}{\partial x}(x_j, t^n) = \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} + \mathcal{O}(\Delta x^2), \quad (3.9)$$

ahora, podemos utilizar la aproximación a la derivada temporal de la sección anterior,

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -a \left(\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} \right) + \mathcal{O}(\Delta t, \Delta x^2) \quad (3.10)$$

para así obtener un esquema basado en diferencias de segundo orden en el espacio y primero en el tiempo

$$u_j^{n+1} = u_j^n - \frac{a}{2} \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_{j-1}^n) + \mathcal{O}(\Delta t^2, \Delta x^2 \Delta t), \quad (3.11)$$

Infortunadamente, el esquema FTCS es *incondicionalmente inestable*. Utilizando la definición (3.8) en la ecuación (3.9), obtenemos

$$\xi_j^{n+1} e^{ikx_j} = \xi_j^n e^{ikx_j} - \frac{\alpha}{2} (\xi_j^n e^{ikx_{j+1}} - \xi_j^n e^{ikx_{j-1}}),$$

de donde podemos obtener el factor de amplificación

$$\begin{aligned} \xi &= 1 - \frac{\alpha}{2} (e^{ik\Delta x} - e^{-ik\Delta x}) \\ &= 1 - \alpha i \sin(k\Delta x) \end{aligned}$$

que tiene un módulo

$$|\xi|^2 = 1 + (\alpha \sin(k\Delta x))^2 > 1.$$

De aquí resulta que el esquema sea *incondicionalmente inestable*.

3.1.3. El Esquema de Lax-Friedrichs

Una manera de tratar el problema de estabilidad del esquema FTCS fue propuesta por Lax y Friedrichs. La idea básica es bastante simple y está basada en reemplazar, en el esquema FTCS, el término u_j^n por su promedio espacial, *i.e.*, $u_j^n = (u_{j+1}^n + u_{j-1}^n)/2$, y así obtener el esquema de Lax-Friedrichs

$$u_j^{n+1} = \frac{1}{2} (u_{j+1}^n + u_{j-1}^n) - \frac{a}{2} \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_{j-1}^n) + \mathcal{O}(\Delta x^2). \quad (3.12)$$

En este esquema estamos asumiendo que se satisface la condición CFL, es por eso que el tenemos $\mathcal{O}(\Delta^2)$ en lugar de $\mathcal{O}(\Delta t^2, \Delta x^2 \Delta t)$. A partir de este momento, asumiremos que esto siempre se cumple.

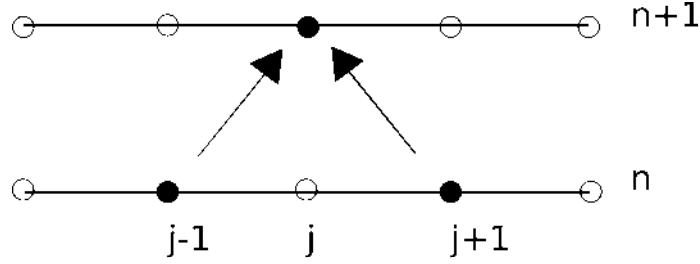


Figura 3.2: Un ejemplo de un stencil para el esquema de Lax-Friedrichs

El stencil utilizado en este esquema puede verse en la figura (3.2). Este nuevo esquema resulta ser *condicionalmente estable*, lo cual puede verse si se hace el análisis de estabilidad de von Neumann. Siguiendo el mismo camino que utilizamos para el esquema FTCS pero usando ahora (3.12) obtenemos

$$\xi^{n+1} e^{ikxj} = \frac{1}{2} (\xi^n e^{ikxj+1} - \xi^n e^{ikxj-1}) - \frac{\alpha}{2} (\xi^n e^{ikxj+1} - \xi^n e^{ikxj-1})$$

de podemos observar que el módulo del factor de amplificación está dado por

$$\begin{aligned} \xi &= \frac{1}{2} (e^{ik\Delta x} + e^{-ik\Delta x}) - \frac{\alpha}{2} (e^{ik\Delta x} - e^{-ik\Delta x}) \\ &= \cos(k\Delta x) + \alpha i \sin(k\Delta x) \end{aligned}$$

cuyo módulo al cuadrado

$$\begin{aligned} |\xi|^2 &= \cos^2(k\Delta x) + \alpha^2 \sin^2(k\Delta x) \\ &= 1 - \sin^2(k\Delta x) + \alpha^2 \sin^2(k\Delta x) \\ &= 1 - (1 - \alpha^2) \sin^2(k\Delta x) \end{aligned}$$

es siempre menor que 1 y cuando se cumpla la condición CFL.

Cuando Lax-Friedrichs realiza esta corrección introducen *disipación numérica* (Viscosidad). Para ver esto, escribimos (3.12) para que se observe claramente la corrección en (3.10)

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -a \left(\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} \right) + \mathcal{O}(\Delta t, \Delta x^2) + \frac{1}{2} \left(\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta t} \right).$$

Esta expresión es la representación en diferencias finitas de la ecuación

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = \frac{1}{2} \left(\frac{\Delta x^2}{\Delta t} \right) \frac{\partial^2 u}{\partial x^2}$$

donde aparece un termino de difusión $\partial^2 u / \partial x^2$.

3.1.4. El Esquema Leapfrog en $1 + 1D$

Tanto el esquema FTCS como el de Lax-Friedrichs son esquemas de “un nivel” con aproximaciones de primer orden para la derivada temporal y de segundo orden para la derivada espacial.

Una aproximación de segundo orden en el tiempo puede obtenerse si insertamos

$$\frac{\partial u(x_j, t_n)}{\partial t} = \frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} + \mathcal{O}(\Delta t^2)$$

en el esquema FTCS, para así llegar al esquema *Leapfrog*

$$u_j^{n+1} = u_j^{n-1} - a \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_{j-1}^n) + \mathcal{O}(\Delta x^2) \quad (3.13)$$

el cual utiliza un stencil como el que se muestra en la figura (3.3).

Existen ciertos aspectos del esquema Leapfrog que debe de tomarse en cuenta

- En un esquema Leapfrog que es estable en el sentido de Courant, no hay disipación de la amplitud (*i.e.*, $|\xi|^2 = 1$). De hecho, si hacemos el análisis de estabilidad de von Neumann, donde tenemos que utilizar

$$\xi^{n+1} e^{ikxj} = \xi^{n-1} e^{ikxj} - \alpha (\xi^n e^{ikxj+1} - \xi^n e^{ikxj-1})$$

para obtener

$$\begin{aligned} \xi^2 &= 1 - \alpha \xi (e^{ik\Delta x} - e^{-ik\Delta x}), \\ &= 1 - 2\alpha \sin(k\Delta x) i \xi, \end{aligned}$$

y entonces, después de resolver el sistema cuadrático

$$\xi^2 + 2\alpha \sin(k\Delta x) i \xi - 1 = 0,$$

obtener el factor de amplificación

$$\xi = -i\alpha \sin(k\Delta x) \pm \sqrt{1 - [\alpha \sin(k\Delta x)]^2}$$

y entonces, si $\alpha < 1$,

$$|\xi|^2 = \alpha^2 \sin^2(k\Delta x) + (1 - [\alpha \sin(k\Delta x)]^2) = 1.$$

lo que dice que el factor de amplificación es siempre 1.

- El esquema Leapfrog es un esquema de “tres niveles”, por lo que se necesitan los valores en el tiempo n y $n - 1$ para poder calcular la aproximación en el tiempo $n + 1$.

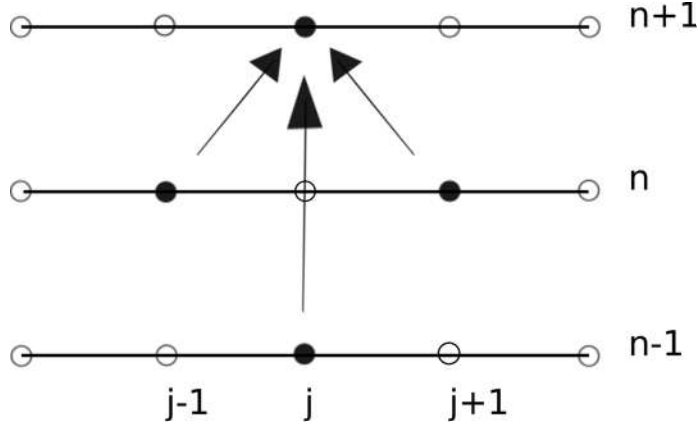


Figura 3.3: Un ejemplo de un stencil para el esquema de Leapfrog.

3.1.5. El Esquema de Lax-Wendroff en $1 + 1D$

El método de Lax-Wendroff es una extensión de segundo orden de precisión del esquema de Lax-Friedrichs. Aquí ocuparemos 2 niveles de tiempo para obtener la aproximación en un nuevo nivel de tiempo.

Existen varias formas de obtener el esquema de Lax-Wendroff, pero probablemente una de las maneras más convenientes para obtenerlo es verlo como una combinación del esquema de Lax-Friedrichs y el esquema Leapfrog. En particular el esquema de Lax-Wendroff puede ser obtenido como

1. Un esquema de Lax-Friedrichs con “medio paso”:

$$\begin{aligned} u_{j+\frac{1}{2}}^{n+\frac{1}{2}} &= \frac{1}{2}(u_{j+1}^n + u_j^n) - \frac{a}{2} \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_j^n) + \mathcal{O}(\Delta x^2) \\ u_{j-\frac{1}{2}}^{n+\frac{1}{2}} &= \frac{1}{2}(u_j^n + u_{j-1}^n) - \frac{a}{2} \frac{\Delta t}{\Delta x} (u_j^n - u_{j-1}^n) + \mathcal{O}(\Delta x^2) \end{aligned}$$

2. Un esquema Leapfrog con medio paso:

$$u_j^{n+1} = u_j^n - a \frac{\Delta t}{\Delta x} (u_{j+\frac{1}{2}}^{n+\frac{1}{2}} - u_{j-\frac{1}{2}}^{n+\frac{1}{2}}) + \mathcal{O}(\Delta x^2)$$

De una manera más específica, los “medios pasos” pueden ser calculados como

$$u_{j\pm\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}(u_j^n + u_{j\pm 1}^n) \mp \frac{a}{2} \frac{\Delta t}{\Delta x} (u_{j\pm 1}^n - u_j^n) + \mathcal{O}(\Delta x^2) \quad (3.14)$$

y, entonces, si usamos Leapfrog de medio paso, la solución en un nuevo nivel de tiempo es

$$u_j^{n+1} = u_j^n - \frac{a\Delta t}{\Delta x} (u_{j+\frac{1}{2}}^{n+\frac{1}{2}} - u_{j-\frac{1}{2}}^{n+\frac{1}{2}}) + \mathcal{O}(\Delta x^2), \quad (3.15)$$

y sustituyendo (3.14), entonces obtenemos el esquema de Lax-Wendroff

$$u_j^{n+1} = u_j^n - \frac{a}{2} \frac{\Delta t}{\Delta x} (u_{j+1}^n - u_{j-1}^n) + \frac{a^2}{2} \frac{\Delta t^2}{\Delta x^2} (u_{j+1}^n - 2u_j^n + u_{j-1}^n) + \mathcal{O}(\Delta x^2). \quad (3.16)$$

Los aspectos importantes que debemos tomar en cuenta de este esquema son

- En el esquema de Lax-Wendroff puede existir dispersión del factor de amplificación. De hecho, si hacemos el análisis de von Neumann, donde utilizamos

$$\begin{aligned} \xi^{n+1} e^{ikx_j} &= \xi^n e^{ikx_j} - \frac{\alpha}{2} (\xi^n e^{ikx_{j+1}} - \xi^n e^{ikx_{j-1}}) \\ &+ \frac{\alpha^2}{2} (\xi^n e^{ikx_{j+1}} - 2\xi^n e^{ikx_j} + \xi^n e^{ikx_{j-1}}), \end{aligned}$$

para obtener el factor de amplificación

$$\begin{aligned} \xi &= 1 - \frac{\alpha}{2} (e^{ik\Delta x} - e^{-ik\Delta x}) + \frac{\alpha^2}{2} (e^{ik\Delta x} - 2 + e^{-ik\Delta x}), \\ &= 1 - \alpha \sin(k\Delta x) - \alpha^2(1 - \cos(k\Delta x)). \end{aligned}$$

Y entonces el módulo del cuadrado del factor de amplificación es

$$\begin{aligned} |\xi|^2 &= [1 - \alpha^2(1 - \cos(k\Delta x))]^2 + \alpha^2 \sin^2(k\Delta x), \\ &= 1 - \alpha^2[2 - 2\cos(k\Delta x) - \alpha^2(1 - \cos(k\Delta x))^2 - \sin^2(k\Delta x)], \\ &= 1 - \alpha^2[(1 - \cos(k\Delta x)) - \alpha^2(1 - \cos(k\Delta x))^2], \\ &= 1 - \alpha^2(1 - \alpha^2)(1 - \cos^2(k\Delta x)), \end{aligned}$$

esto quiere decir que, siempre y cuando se cumpla que $\alpha^2 \leq 1$, tendremos que $0 \leq |\xi|^2 \leq 1$, esto es lo mismo a pedir que se cumpla la condición CFL. Algo que se debe de notar es que a menos que $\alpha^2 = 1$, entonces $|\xi|^2 < 1$ y se presentará algo de disipación. En este sentido las propiedades de dispersión del método de Lax-Friedrichs no se pierden por completo, pero en el esquema de Lax-Wendroff son mucho menos severas.

- El esquema de Lax-Wendroff es un esquema de 2 niveles, pero puede ser escrito en forma de un nivel por medio de manipulación algebraica. Estos se puede ver claramente en la expresión (3.16), donde sólo aparecen cantidades en los niveles de tiempo n y $n + 1$.

3.2. Advección en $2 + 1D$

En la sección anterior se explicó la importancia de la ecuación de advección, así como algunos esquemas utilizados para aproximar su solución en $1 + 1D$. En

esta sección nos enfocaremos a la aproximación en $2 + 1D$. Es necesario precisar que, actualmente, existen muy pocos trabajos sobre este tema, y es por eso que se presentará únicamente el esquema de Lax-Wendroff para la aproximación de la ecuación de advección.

En el caso de $2 + 1D$ la ecuación de advección tiene la forma

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} = 0, \quad (3.17)$$

pero, para nuestra conveniencia, la escribiremos de la forma

$$\frac{\partial u}{\partial t} = -a \frac{\partial u}{\partial x} - b \frac{\partial u}{\partial y}. \quad (3.18)$$

3.2.1. El Caso Rectangular

Si nos encontramos en el caso en que tenemos una malla rectangular, en la cual se pueden tomar pasos espaciales constantes Δx y Δy , el esquema de Lax-Wendroff para la ecuación de advección en $2 + 1D$ puede ser obtenido fácilmente de los desarrollos en serie de Taylor de u

$$\begin{aligned} u(x, y, t + \Delta t) &= u(x, y, t) + \frac{\partial u(x, y, t)}{\partial t} (\Delta t) + \mathcal{O}(\Delta t), \\ u(x + \Delta x, y, t) &= u(x, y, t) + \frac{\partial u(x, y, t)}{\partial x} (\Delta x) + \mathcal{O}(\Delta x), \\ u(x, y + \Delta y, t) &= u(x, y, t) + \frac{\partial u(x, y, t)}{\partial y} (\Delta y) + \mathcal{O}(\Delta y), \end{aligned}$$

de donde, despejando los términos $du(x, y, t)/dt$, $du(x, y, t)/dx$, y $du(x, y, t)/dy$, se obtiene una manera de aproximar las derivadas, tanto espaciales como la temporal

$$\begin{aligned} \frac{\partial u(x, y, t)}{\partial t} &= \frac{u(x, y, t + \Delta t) - u(x, y, t)}{\Delta t} + \mathcal{O}(\Delta t), \\ \frac{\partial u(x, y, t)}{\partial x} &= \frac{u(x + \Delta x, y, t) - u(x, y, t)}{\Delta x} + \mathcal{O}(\Delta x), \\ \frac{\partial u(x, y, t)}{\partial y} &= \frac{u(x, y + \Delta y, t) - u(x, y, t)}{\Delta y} + \mathcal{O}(\Delta y), \end{aligned}$$

las cuales, por cuestiones de simplicidad y siguiendo la misma notación que en la sección anterior, pueden ser escritas como

$$\frac{\partial u_{i,j}^n}{\partial t} = \frac{u_{i,j}^{n+1} - u_{i,j}^n}{\Delta t} + \mathcal{O}(\Delta t), \quad (3.19)$$

$$\frac{\partial u_{i,j}^n}{\partial x} = \frac{u_{i+1,j}^n - u_{i,j}^n}{\Delta x} + \mathcal{O}(\Delta x), \quad (3.20)$$

$$\frac{\partial u_{i,j}^n}{\partial y} = \frac{u_{i,j+1}^n - u_{i,j}^n}{\Delta y} + \mathcal{O}(\Delta y). \quad (3.21)$$

$$(3.22)$$

Si sustituimos (3.19), (3.20) y (3.21) en (3.18) podemos obtener

$$\frac{u_{i,j}^{n+1} - u_{i,j}^n}{\Delta t} = -a \frac{u_{i+1,j}^n - u_{i,j}^n}{\Delta x} - b \frac{u_{i,j+1}^n - u_{i,j}^n}{\Delta y},$$

que puede ser manipulada por medio de un poco de algebra para conseguir

$$u_{i,j}^{n+1} = u_{i,j}^n - a \frac{\Delta t}{\Delta x} (u_{i+1,j}^n - u_{i,j}^n) - b \frac{\Delta t}{\Delta y} (u_{i,j+1}^n - u_{i,j}^n), \quad (3.23)$$

que resulta ser el esquema de Lax-Wendroff para la ecuación de advección en $2 + 1D$.

3.2.2. El Caso en Regiones no Rectangulares

Cuando nos encontramos con problemas en regiones irregulares, para los cuales no se puede generar una malla con espaciado uniforme, el esquema de la sección anterior no puede utilizarse. Es por esto que surgió la necesidad de tener un esquema que pudiera adaptarse a este tipo de regiones.

Recordemos primero que, en general, los esquemas de diferencias finitas se obtienen de considerar un punto central p_0 , y un conjunto finito de nodos vecinos $p_1, p_2, \dots, p_k$ en una cierta región de interés, para los cuales es necesario encontrar los coeficientes $\Gamma_1, \Gamma_2, \dots, \Gamma_q$ que cumplan con

$$\sum_{i=0}^q \Gamma_i \varphi(p_i) \approx \frac{\partial^k \varphi}{\partial x^l \partial y^{k-l}} \bigg|_{x=p_0}; \quad k = 0, 1, 2, \dots; \quad 0 \leq l \leq k.$$

Si recordamos que la forma general que tienes las ecuaciones lineales de segundo orden es

$$L\varphi = A\varphi_{xx} + B\varphi_{xy} + C\varphi_{yy} + D\varphi_x + E\varphi_y + F,$$

entonces, para aproximar dicho operador en el punto p_0 utilizando aproximaciones a los valores de φ en algunos de los nodos vecinos $(p_1, p_2, \dots, p_q)$ de p_0 (ver figura 3.4), utilizando un esquema de diferencias finitas, el cual se puede escribir como la combinación lineal

$$L_0 = \Gamma_0 \varphi(p_0) + \Gamma_1 \varphi(p_1) + \dots + \Gamma_q \varphi(p_q). \quad (3.24)$$

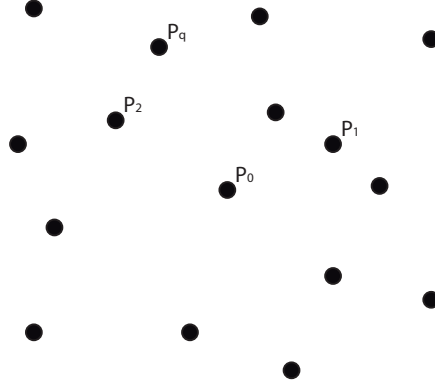


Figura 3.4: Distribución arbitraria de p_0 y sus vecinos.

con $\Gamma_0, \Gamma_1, \dots, \Gamma_k$ constantes.

Para poder decir que tenemos un esquema consistente, esto es, que cumple con que la aproximación se acerque a la solución a medida que los nodos vecinos se aproximan al nodo central p_0 ; necesitamos que se cumpla

$$\sum_{i=0}^q \Gamma_i \varphi(p_i) - [A\varphi_{xx} + B\varphi_{xy} + C\varphi_{yy} + D\varphi_x + E\varphi_y]_{p_0} \rightarrow 0. \quad (3.25)$$

Esto puede escribirse usando la expansión en serie de Taylor hasta segundo orden y reagrupando algunos términos como

$$\begin{aligned} [A\varphi_{xx} + B\varphi_{xy} + C\varphi_{yy} + D\varphi_x + E\varphi_y + F\varphi]_{p_0} - \sum_{i=0}^q \Gamma_i \varphi(p_i) = \\ \left(F(p_0) - \sum_{i=0}^q \Gamma_i \right) \varphi(p_0) + \left(D(p_0) - \sum_{i=1}^q \Gamma_i \Delta x_i \right) \varphi_x(p_0) + \\ \left(E(p_0) - \sum_{i=1}^q \Gamma_i \Delta y_i \right) \varphi_y(p_0) + \left(A(p_0) - \sum_{i=1}^q \frac{\Gamma_i (\Delta x_i)^2}{2} \right) \varphi_{xx}(p_0) + \\ \left(B(p_0) - \sum_{i=1}^q \Gamma_i \Delta x_i \Delta y_{i,0} \right) \varphi_{xy}(p_0) + \left(C(p_0) - \sum_{i=1}^q \frac{\Gamma_i (\Delta y_i)^2}{2} \right) \varphi_{yy}(p_0) + \\ \mathcal{O}(\max\{\Delta x_i, \Delta y_i\})^4 \end{aligned} \quad (3.26)$$

donde Δx y Δy son las distancias horizontal y vertical respectivamente del nodo central p_0 a cada uno de los nodos vecinos p_i .

Lo que queremos es que cada uno de los coeficientes entre paréntesis en (3.26) sea cero, si logramos esto, la diferencia entre nuestro esquema y el operador quedará acotada por una función $\mathcal{O}$, que involucra a las derivadas de orden mayor,

y disminuirá conforme los nodos vecinos se acerquen a p_0 [10].

Entonces necesitamos encontrar los valores de las Γ_i que cumplan con esto, esto es, necesitamos resolver el sistema

$$\begin{pmatrix} 1 & 1 & \dots & 1 \\ 0 & \Delta x_1 & \dots & \Delta x_q \\ 0 & \Delta y_1 & \dots & \Delta y_q \\ 0 & (\Delta x_1)^2 & \dots & (\Delta x_q)^2 \\ 0 & \Delta x_1 \Delta y_1 & \dots & \Delta x_q \Delta y_q \\ 0 & (\Delta y_1)^2 & \dots & (\Delta y_q)^2 \end{pmatrix} \begin{pmatrix} \Gamma_0 \\ \Gamma_1 \\ \Gamma_2 \\ \cdot \\ \cdot \\ \Gamma_q \end{pmatrix} = \begin{pmatrix} F(p_0) \\ D(p_0) \\ E(p_0) \\ 2A(p_0) \\ B(p_0) \\ 2C(p_0) \end{pmatrix} \quad (3.27)$$

Desafortunadamente este sistema cuenta con 6 ecuaciones y $q + 1$ incógnitas; esto quiere decir que en la mayoría de los casos no estará bien determinado. Una manera de sobrellevar esta indeterminación es resolviendo el sistema en un sentido de cuadrados mínimos. Para hacer este procedimiento en una manera no iterativa consideramos únicamente las últimas 5 ecuaciones de (3.27)

$$\begin{pmatrix} \Delta x_1 & \dots & \Delta x_q \\ \Delta y_1 & \dots & \Delta y_q \\ (\Delta x_1)^2 & \dots & (\Delta x_q)^2 \\ \Delta x_1 \Delta y_1 & \dots & \Delta x_q \Delta y_q \\ (\Delta y_1)^2 & \dots & (\Delta y_q)^2 \end{pmatrix} \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \\ \cdot \\ \cdot \\ \Gamma_q \end{pmatrix} = \begin{pmatrix} D(p_0) \\ E(p_0) \\ 2A(p_0) \\ B(p_0) \\ 2C(p_0) \end{pmatrix}, \quad (3.28)$$

cuyas ecuaciones normales pueden ser resueltas utilizando cualquier algoritmo de solución de sistemas de ecuaciones lineales, por ejemplo el algoritmo de factorización de Cholesky, para obtener los valores de $\Gamma_1, \dots, \Gamma_q$.

Después de hacer esto, aún nos falta calcular el valor de Γ_0 , pero esto puede hacerse de una manera sencilla tomando en cuenta la primera ecuación de (3.27), con lo cual obtenemos

$$\Gamma_0 = -\Gamma_1 \dots - \Gamma_q, \quad (3.29)$$

y con esto obtenemos el conjunto de coeficientes Γ necesarios para definir el esquema (3.24) [11].

Ahora que ya contamos con un esquema para aproximar ecuaciones diferenciales parciales, buscamos la forma de aplicarlo a la ecuación de advección. Si tomamos en cuenta la ecuación de advección en $2 + 1D$

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0$$

la cual podemos escribir, por conveniencia, como

$$\frac{\partial u}{\partial t} = -a \frac{\partial u}{\partial x} - b \frac{\partial u}{\partial y} \quad (3.30)$$

y le podemos aplicar la regla de la cadena para así obtener

$$\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2} + 2ab \frac{\partial^2 u}{\partial x \partial y} + b^2 \frac{\partial^2 u}{\partial y^2}. \quad (3.31)$$

Ahora, tomando en cuenta el desarrollo en serie de Taylor de u alrededor de $(x, y, t + \Delta t)$

$$u(x, y, t + \Delta t) = u(x, y, t) + \frac{\partial u(x, y, t)}{\partial t} \Delta t + \frac{1}{2} \frac{\partial^2 u(x, y, t)}{\partial t^2} (\Delta t)^2 + \dots,$$

y sustituyendo (3.30) y (3.31) en ella obtenemos [20]

$$\begin{aligned} u(x, y, t + \Delta t) &= u(x, y, t) + \\ &\Delta t \left(-a \frac{\partial u}{\partial x} - b \frac{\partial u}{\partial y} \right) + \\ &\frac{(\Delta t)^2}{2} \left(a^2 \frac{\partial^2 u}{\partial x^2} + 2ab \frac{\partial^2 u}{\partial x \partial y} + b^2 \frac{\partial^2 u}{\partial y^2} \right) + \dots \end{aligned}$$

El esquema que definimos en (3.28) y (3.29) se puede aplicar al operador

$$-\Delta t \left(a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} \right) + \frac{(\Delta t)^2}{2} \left(a^2 \frac{\partial^2 u}{\partial x^2} + 2ab \frac{\partial^2 u}{\partial x \partial y} + b^2 \frac{\partial^2 u}{\partial y^2} \right),$$

y los coeficientes $\Gamma_0, \Gamma_1, \dots, \Gamma_q$ que se obtienen definen el esquema modificado de Lax-Wendroff

$$u_0^{k+1} = u_0^k + \sum_{i=0}^q \Gamma_i u_i^k, \quad (3.32)$$

siendo $p_0, \dots, p_q$ los nodos involucrados en el stencil.

3.3. Advección en $2 + 1D$: Esquema de 4 Puntos

El esquema definido en la sección anterior puede ser usado tanto en mallas estructuradas como en mallas no estructuradas. En nuestro caso el uso de mallas lógicamente rectangulares no permite aprovechar la estructura dada por los índices en la malla $G = \{p_{i,j} \mid 1 \leq i \leq m, 1 \leq j \leq n\}$.

Como primer esquema utilizamos un esquema que involucra cada nodo de la malla y 3 nodos vecinos, utilizando para ello un stencil de tipo “upwind” como se muestra en la figura (3.5).

Utilizando este stencil, entonces el esquema (3.32)

$$u_0^{k+1} = u_0^k + \sum_{i=0}^q \Gamma_i u_i^k,$$

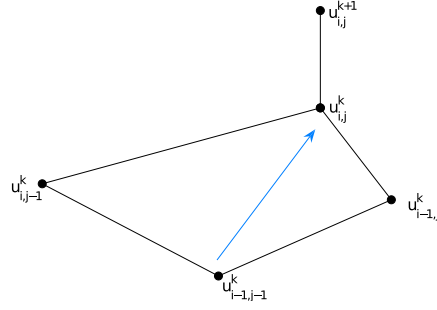


Figura 3.5: Stencil tipo “upwind”.

toma la forma

$$u_{i,j}^{k+1} = u_{i,j}^k + \Gamma_0 u_{i,j}^k + \Gamma_1 u_{i-1,j}^k + \Gamma_2 u_{i,j-1}^k + \Gamma_3 u_{i-1,j-1}^k, \quad (3.33)$$

aquí tomamos $u_{i,j}^k$ como la aproximación a u en el punto i, j de la malla en el nivel de tiempo k .

El esquema definido en (3.33) es un esquema modificado de Lax-Wendroff explícito, que utilizaremos para aproximar la solución del problema de advección. También se puede definir un esquema implícito del mismo tipo. Para hacer esto tomamos el esquema (3.32) y le agregamos un parámetro λ para utilizar 2 niveles de tiempo del lado derecho, con esto obtenemos el esquema

$$u_0^{k+1} = u_0^k + \lambda \left[\sum_{i=0}^q \Gamma_i u_i^k \right] + (1 - \lambda) \left[\sum_{i=0}^q \Gamma_i u_i^{k+1} \right].$$

Utilizando el mismo tipo de stencil que en el caso explícito, donde tomamos 3 vecinos de p_0 , el esquema anterior llega a la forma

$$\begin{aligned} u_{i,j}^{k+1} &= u_{i,j}^k + \lambda [\Gamma_0 u_{i,j}^k + \Gamma_1 u_{i-1,j}^k + \Gamma_2 u_{i,j-1}^k + \Gamma_3 u_{i-1,j-1}^k] + \\ &\quad (1 - \lambda) [\Gamma_0 u_{i,j}^{k+1} + \Gamma_1 u_{i-1,j}^{k+1} + \Gamma_2 u_{i,j-1}^{k+1} + \Gamma_3 u_{i-1,j-1}^{k+1}]. \end{aligned}$$

que tras algunos pasos de álgebra, y en aras de la simplicidad, puede escribirse como

$$u_{i,j}^{k+1} = \frac{E + B u_{i-1,j}^{k+1} + C u_{i-1,j-1}^{k+1} + D u_{i,j-1}^{k+1}}{A}$$

donde

$$E = \lambda [\Gamma_1 u_{i-1,j}^k + \Gamma_2 u_{i-1,j-1}^k + \Gamma_3 u_{i,j-1}^k] + u_{i,j}^k [1 + \lambda \Gamma_0]$$

y además

$$\begin{aligned} A &= 1 + (\lambda - 1)\Gamma_0 \\ B &= (1 - \lambda)\Gamma_1 \\ C &= (1 - \lambda)\Gamma_2 \\ D &= (1 - \lambda)\Gamma_3 \end{aligned}$$

Este esquema es un esquema implícito para aproximar la solución de la ecuación de advección. Este tipo de esquemas suelen llegar a ser *incondicionalmente estables*, y es por ello que se recurrió a obtenerlo.

3.4. Advección en $2 + 1D$: Esquema de 6 Puntos

Otro esquema diseñado para aproximar esta ecuación se logra si, al utilizar nuevamente el esquema (3.32), utilizamos cada nodo de la malla y 5 de sus vecinos para así calcular el valor de u en cada nodo de la malla. En este esquema el stencil utilizado varía dependiendo de la localización lógica del nodo correspondiente a lo largo de la malla (véase figura 3.6).

En cada caso obtenemos

$$u_{i,j}^{k+1} = u_{i,j}^k + \Gamma_0 u_{i,j}^k + \Gamma_1 u_1^k + \Gamma_2 u_2^k + \Gamma_3 u_3^k + \Gamma_4 u_4^k + \Gamma_5 u_5^k, \quad (3.34)$$

los u_i^k son las aproximaciones a u en el tiempo k en los nodos correspondientes a cada stencil.

Este esquema también es un esquema explícito para la aproximación de la ecuación de advección, en este caso, al usar 5 stenciles diferentes que involucran 6 nodos de la malla, el esquema logra mayor precisión que esquemas anteriores, lo cual se podrá notar en la sección siguiente.

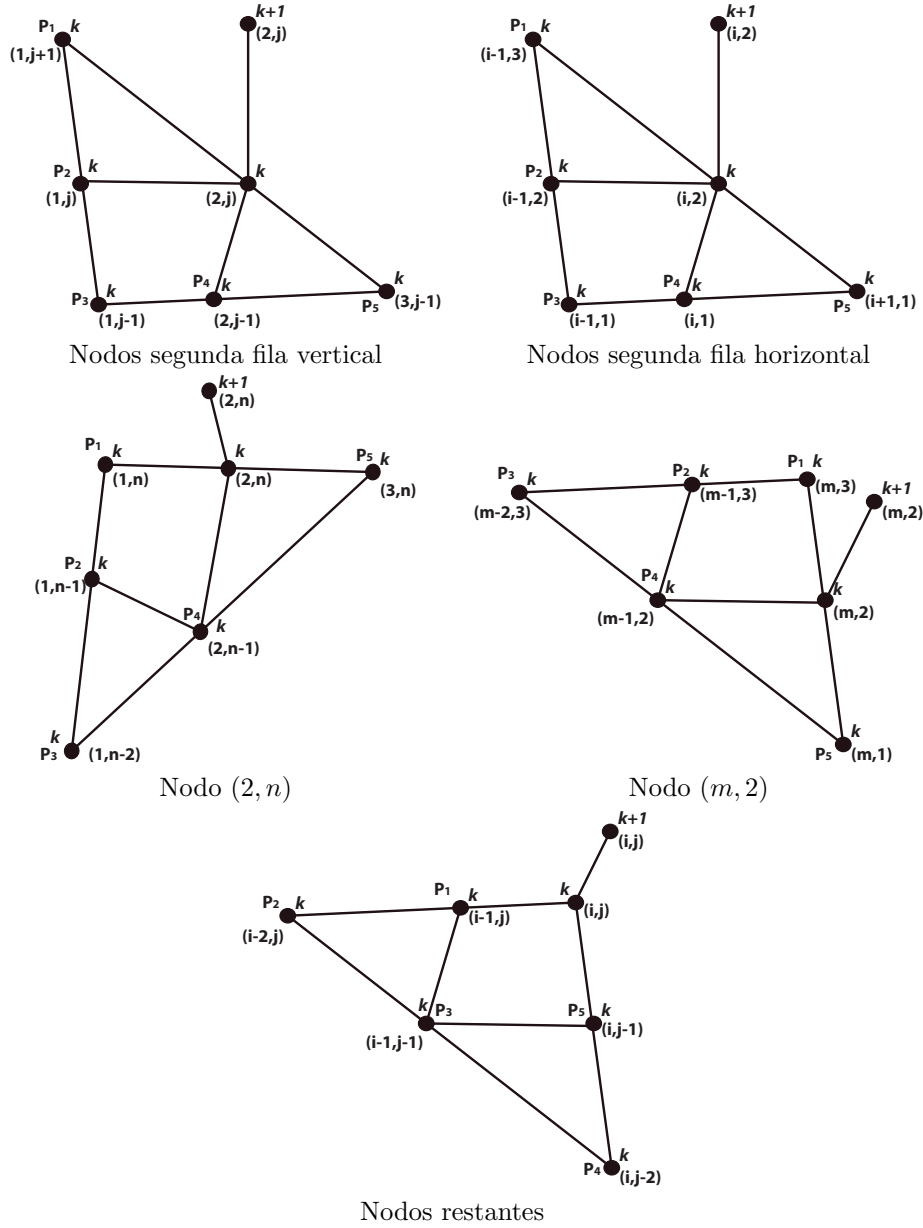


Figura 3.6: Diferentes estenciles del esquema de 6 puntos.

Capítulo 4

Resultados y conclusiones

4.1. Resultados

En este capítulo presentamos los resultados numéricos de las pruebas realizadas a los diferentes esquemas que se trataron a lo largo del presente trabajo, primeramente se hace un análisis, de manera comparativa, de los resultados obtenidos con los esquemas en $1 + 1D$. Posteriormente se trabajan con los esquemas de $2 + 1D$, donde se hablará con más detalle de los resultados.

Los códigos de cada uno de estos esquemas se pueden encontrar en el Anexo A.

4.1.1. Esquemas en $1 + 1D$

Primero se hará una revisión rápida de los esquemas en $1 + 1D$, para esto utilizaremos una malla uniforme en el intervalo espacial $[0, 2\pi]$, para que se cumpla la condición CFD en cada uno de los esquemas, en los esquemas Upwind, FTCS, Leapfrog y Lax-Wendroff se dividirá el intervalo $[0, 2\pi]$ en 21, 41 y 81 subintervalos; mientras que para el esquema Lax-Friedrichs se dividirá en 1001, 2001 y 4001. La discretización del tiempo se hace dividiendo el intervalo $[0, 5]$ en 1000 subintervalos uniformemente espaciados.

En todos los casos se utilizó la función

$$u = \sin\left(x - \frac{\pi}{10}t\right),$$

como la condición inicial y las condiciones de frontera en $x = 0$ y $x = 1$.

Podemos probar fácilmente que esta función cumple con la ecuación de advección, para ello calculamos primero sus derivadas parciales

$$\begin{aligned}\frac{\partial u}{\partial t} &= -\frac{\pi}{10} \cos\left(\frac{\pi}{10}t - x\right) \\ \frac{\partial u}{\partial x} &= \cos\left(\frac{\pi}{10}t - x\right).\end{aligned}$$

Si sustituimos estas expresiones en la ecuación (3.4), que es la ecuación de advección, obtenemos

$$\frac{\partial u}{\partial t} + \frac{\pi}{10} \frac{\partial u}{\partial x} = -\frac{\pi}{10} \cos\left(\frac{\pi}{10}t - x\right) + \frac{\pi}{10} \cos\left(\frac{\pi}{10}t - x\right) = 0.$$

De esta manera hemos comprobado que la función utilizada cumple con las características que deseamos.

En cada caso se calculó la norma $\|\cdot\|_2$ para el error en cada nivel de tiempo como

$$\|u - v\|_2 = \sqrt{\sum_i^M (u_i - v_i)^2 \Delta x}.$$

donde u y v son las soluciones aproximadas y exacta, respectivamente, calculadas en el i -ésimo elemento; y M es el número de nodos.

Upwind

En la tabla (4.1) se muestran los resultados para el esquema upwind en $1 + 1D$, los datos mostrados en dicha tabla son:

- NODOS: Número de nodos en que se divide la región.
- TIEMPO: Valor de tiempo en que se mide el error.
- ERROR: Error cuadrático respecto a la solución real del problema.

En la figura (4.1) se muestra una comparación de los resultados en el tiempo $t = 5$, con fines comparativos se agrega también la gráfica de la solución exacta.

Nótese que el error crece conforme avanza el tiempo, esto se debe a dos factores:

1. La calidad de aproximación del esquema: Hay que señalar que Upwind es un esquema de orden dos.
2. La discretización de la región. Esto se puede mostrar señalando que los errores decrecen conforme se aumenta el número de nodos en la discretización.

FTCS

En la tabla (4.2) se muestran los resultados para el esquema FTCS en $1 + 1D$, los datos que se muestran se ordenan de la misma manera en que fueron ordenados para el método upwind. De igual manera en la figura (4.2) se muestra una comparación de los resultados en el tiempo $t = 5$.

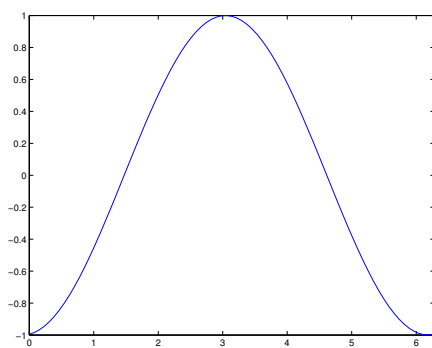
Al igual que en el esquema Upwind, con FTCS el error crece conforme el tiempo. Pero debe notarse que la aproximación es mejor para este caso, eso se debe al orden de aproximación de este esquema.

NODOS	TIEMPO	ERROR
21	1.0	3.357238E-02
	2.0	6.416559E-02
	3.0	9.049054E-02
	4.0	1.119140E-01
	5.0	1.290456E-01
41	1.0	1.700147E-02
	2.0	3.307577E-02
	3.0	4.751899E-02
	4.0	5.978906E-02
	5.0	6.986262E-02
81	1.0	8.482462E-03
	2.0	1.665244E-02
	3.0	2.416086E-02
	4.0	3.069171E-02
	5.0	3.614946E-02

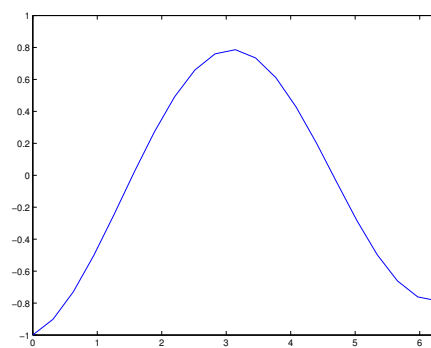
Tabla 4.1: Errores para el método upwind en $1 + 1D$.

NODOS	TIEMPO	ERROR
21	1.0	3.302758E-03
	2.0	6.606363E-03
	3.0	9.917568E-03
	4.0	1.325458E-02
	5.0	1.661552E-02
41	1.0	8.637862E-04
	2.0	1.720079E-03
	3.0	2.571373E-03
	4.0	3.427478E-03
	5.0	4.289234E-03
81	1.0	2.676115E-04
	2.0	5.339900E-04
	3.0	8.012385E-04
	4.0	1.069820E-03
	5.0	1.336072E-03

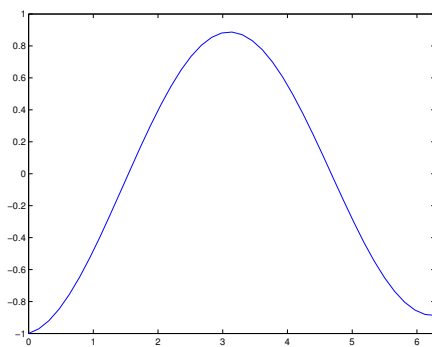
Tabla 4.2: Errores para el método FTCS en $1 + 1D$.



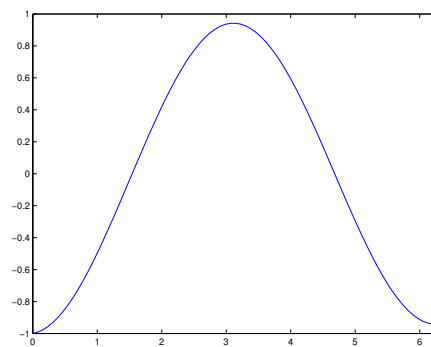
Solución exacta.



Upwind con 21 nodos.

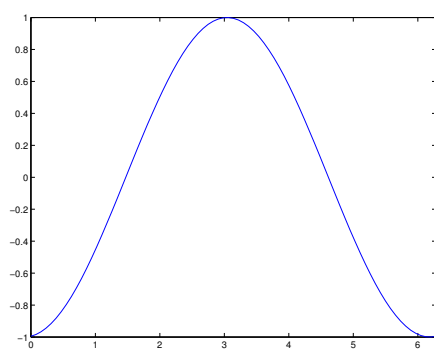


Upwind con 41 nodos.

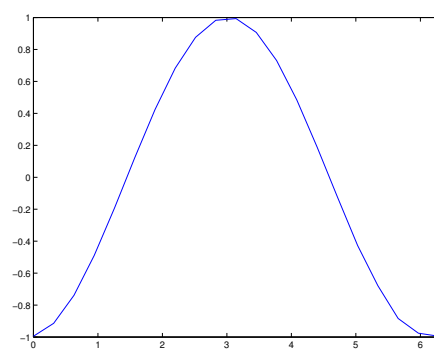


Upwind con 81 nodos.

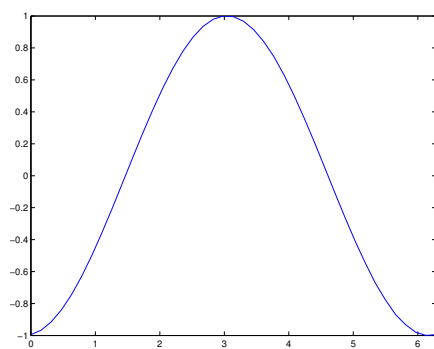
Figura 4.1: Comparación de resultados método upwind en $t = 5$.



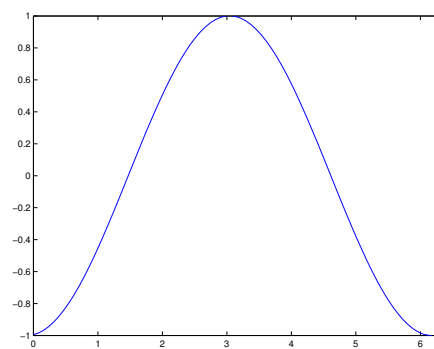
Solución exacta.



FTCS con 21 nodos.



FTCS con 41 nodos.



FTCS con 81 nodos.

Figura 4.2: Comparación de resultados método FTCS en $t = 5$.

NODOS	TIEMPO	ERROR
1001	1.0	2.607317E-03
	2.0	5.150868E-03
	3.0	7.525431E-03
	4.0	9.625300E-03
	5.0	1.140303E-02
2001	1.0	5.220786E-04
	2.0	1.033631E-03
	3.0	1.513851E-03
	4.0	1.941147E-03
	5.0	2.304932E-03
4001	1.0	3.489592E-07
	2.0	6.912618E-07
	3.0	1.013045E-06
	4.0	1.299807E-06
	5.0	1.544288E-06

Tabla 4.3: Errores para el método Lax-Friedrichs en $1 + 1D$.

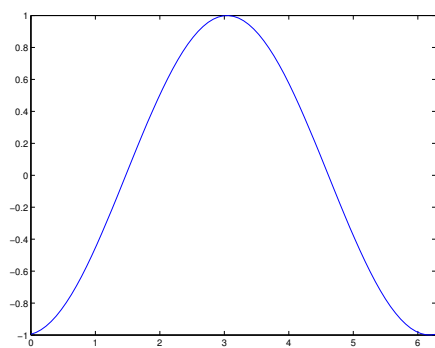
Lax-Friedrichs

Debido a las condiciones para la estabilidad de este método, explicadas en el capítulo anterior, la discretización en el espacio se hizo diferente para éste ejemplo; el intervalo espacial $[0, 2\pi]$ se divide entonces en 1001, 2001 y 4001 puntos para cumplir con la condición CFD. En la tabla (4.3) se muestran los resultados de utilizar este esquema con esta discretización; de igual manera en la figura (4.3) se muestran las gráficas al tiempo $t = 5$.

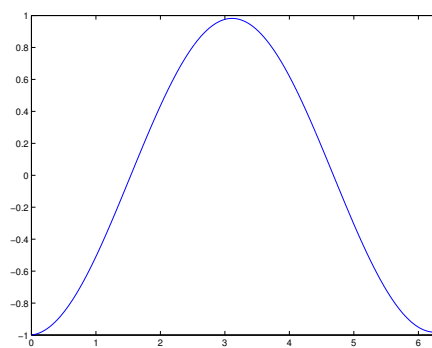
Con fines comparativos, en la tabla (4.4) se muestran los errores obtenidos con este esquema utilizando la misma discretización que se usó en los esquemas anteriores. Las gráficas que se muestran en la figura (4.4) corresponden a la aproximación con estos datos.

Leapfrog

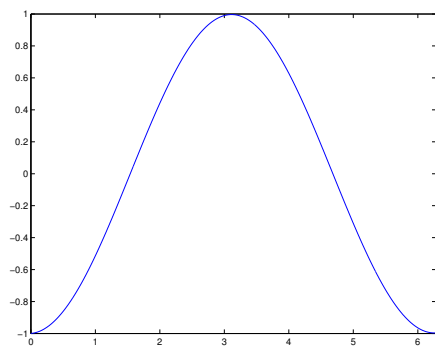
El esquema Leapfrog tiene grandes problemas con la mayor parte de las funciones, ya que, por la manera como está diseñado, los nodos pares y los impares se encuentran desacoplados. Para pequeños intervalos de tiempo esto no es un problema grave, pero mientras más tiempo pase, esto generará una gran dispersión numérica, como se puede ver en la tabla (4.5) y la figura (4.5), donde se encuentran los errores y las gráficas obtenidas al usar este esquema.



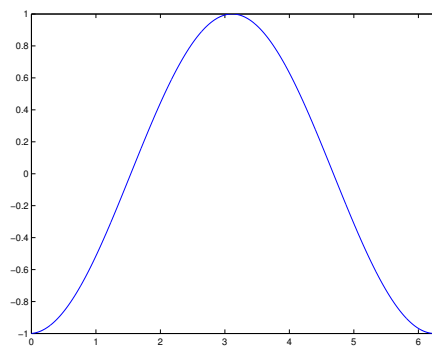
Solución exacta.



Lax-Friedrichs con 1001 nodos.



Lax-Friedrichs con 2001 nodos.



Lax-Friedrichs con 4001 nodos.

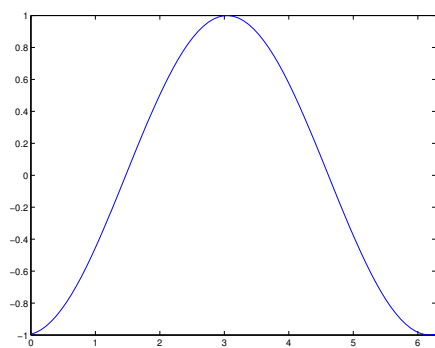
Figura 4.3: Comparación de resultados método Lax-Friedrichs en $t = 5$.

NODOS	TIEMPO	ERROR
21	1.0	7.293105E-01
	2.0	8.476843E-01
	3.0	1.006064E-00
	4.0	1.139332E-00
	5.0	1.210734E-00
41	1.0	6.428185E-01
	2.0	7.197975E-01
	3.0	8.142575E-01
	4.0	9.432134E-01
	5.0	1.060593E-00
81	1.0	3.222416E-01
	2.0	4.796974E-01
	3.0	5.509072E-01
	4.0	5.968804E-01
	5.0	6.528482E-01

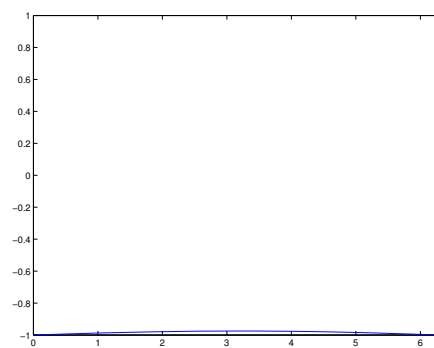
Tabla 4.4: Errores para el método Lax-Friedrichs en $1 + 1D$.

NODOS	TIEMPO	ERROR
21	1.0	2.022438E-01
	2.0	3.951509E-01
	3.0	5.763236E-01
	4.0	7.482981E-01
	5.0	9.104062E-01
41	1.0	2.113279E-01
	2.0	4.078606E-01
	3.0	5.907091E-01
	4.0	7.636833E-01
	5.0	9.261214E-01
81	1.0	2.136509E-01
	2.0	4.109577E-01
	3.0	5.942639E-01
	4.0	7.674672E-01
	5.0	9.299573E-01

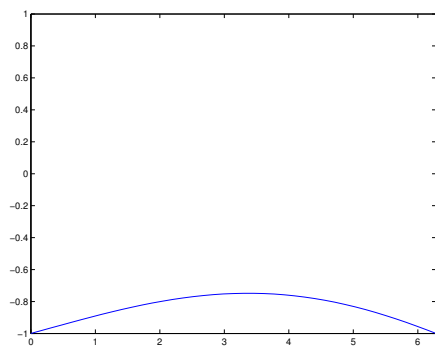
Tabla 4.5: Errores para el método Leapfrog en $1 + 1D$.



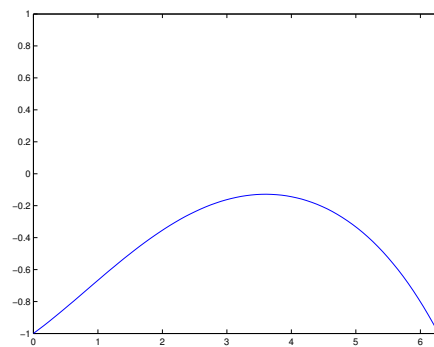
Solución exacta.



Lax-Friedrichs con 21 nodos.

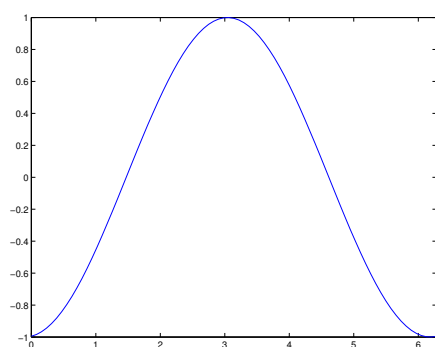


Lax-Friedrichs con 41 nodos.

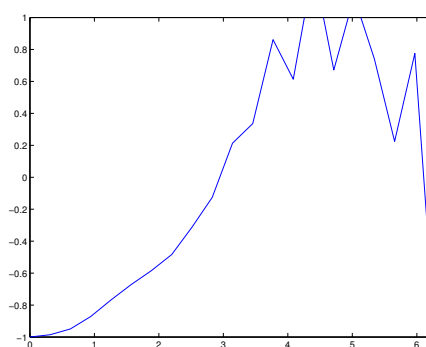


Lax-Friedrichs con 81 nodos.

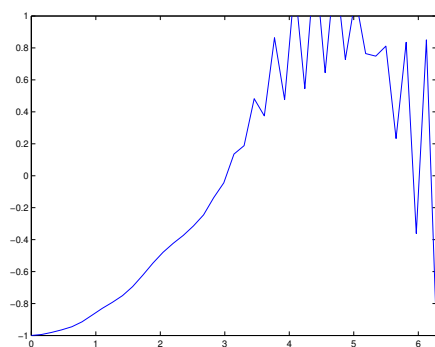
Figura 4.4: Comparación de resultados método Lax-Friedrichs en $t = 5$.



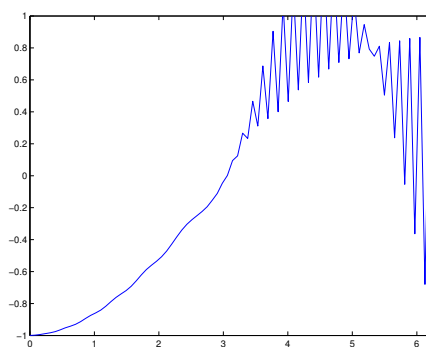
Solución exacta.



Leapfrog con 21 nodos.



Leapfrog con 41 nodos.



Leapfrog con 81 nodos.

Figura 4.5: Comparación de resultados método Leapfrog en $t = 5$.

NODOS	TIEMPO	ERROR
21	1.0	1.105326E-02
	2.0	2.150859E-02
	3.0	3.105454E-02
	4.0	3.950757E-02
	5.0	4.695498E-02
41	1.0	5.330685E-03
	2.0	1.047718E-02
	3.0	1.524278E-02
	4.0	1.945477E-02
	5.0	2.307323E-02
81	1.0	2.565827E-03
	2.0	5.064620E-03
	3.0	7.395704E-03
	4.0	9.462203E-03
	5.0	1.122675E-02

Tabla 4.6: Errores para el método de Lax-Wendroff en $1 + 1D$.**Lax-Wendroff**

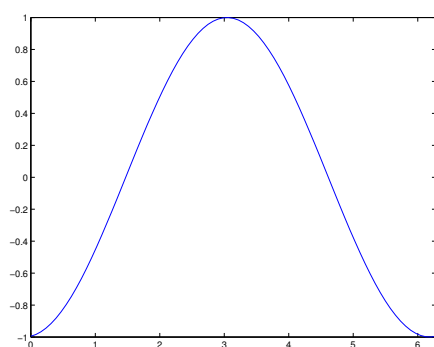
En el método de Lax-Wendroff se corrigen la mayor parte de los problemas de dispersión numérica, desgraciadamente nos encontramos con un nuevo problema, el problema de las fronteras. Es bien sabido que el método de Lax-Wendroff tiene serios problemas cuando se trata de las fronteras, estos problemas se pueden “suavizar” mientras que el término de dispersión sea pequeño, esto es, hacer que α se haga bastante pequeño. Utilizando los mismos datos que en los esquemas anteriores, se calculó el error obtenido al usar este esquema. Los resultados pueden consultarse en la tabla (4.6) y en la figura (4.6).

Los resultados del esquema de Lax-Wendroff nos muestran que la calidad de aproximación de este esquema depende mucho de la condición CFL. Mientras que el término α de dicha condición se haga más pequeño, mayor estabilidad tendrá el método en las fronteras de la región. Un estudio más a fondo de esta condición puede ser encontrado en [15].

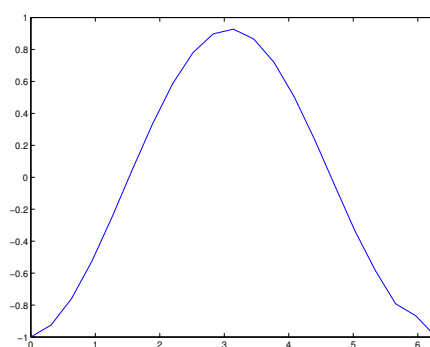
4.1.2. Esquemas en $2 + 1D$: Lax-Wendroff

Una vez revisados los esquemas en $1 + 1D$, podemos pasar a realizar pruebas numéricas para los esquemas en $2 + 1D$ que se describen en el capítulo anterior. Hay que resaltar que el esquema en $2 + 1D$ presentado únicamente puede ser aplicado en mallas regulares y uniformes, y esto es una gran limitación para el método.

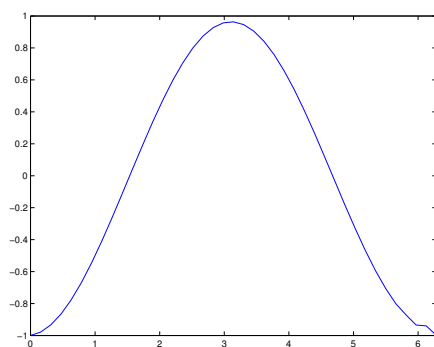
En el caso que nos concierne utilizamos una región Ω rectangular en intervalo



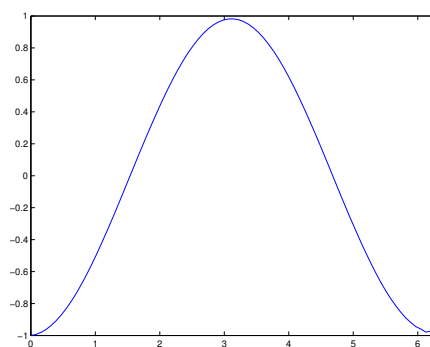
Solución exacta.



Lax-Wendroff con 21 nodos.



Lax-Wendroff con 41 nodos.



Lax-Wendroff con 81 nodos.

Figura 4.6: Comparación de resultados método de Lax-Wendroff en $t = 5$.

$[0, 1] \times [0, 1]$, haciendo diferentes mallas de la misma con 21, 41 y 81 puntos por lado. La región Ω es un dominio plano simplemente conexo, el cual cumple con que su frontera $\partial\Omega$ es un polígono de Jordan con orientación positiva y $\partial\Omega = S_1 \cup S_2$, con S_1 y S_2 conectados (fig. 4.7). El intervalo de tiempo $[0, 1]$ se discretizó uniformemente partiéndolo en 1000 subintervalos. Utilizando estos datos, el esquema de Lax-Wendroff en $2 + 1D$ fue utilizado para aproximar la función

$$\sin(x - 0.1t)e^{y-0.1t}$$

que fue utilizada como condición inicial y de frontera en S_1 .

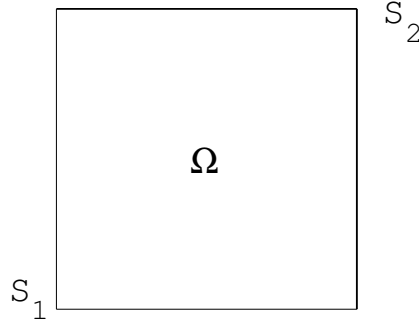


Figura 4.7: Dominio Ω rectangular con $\partial\Omega = S_1 \cup S_2$.

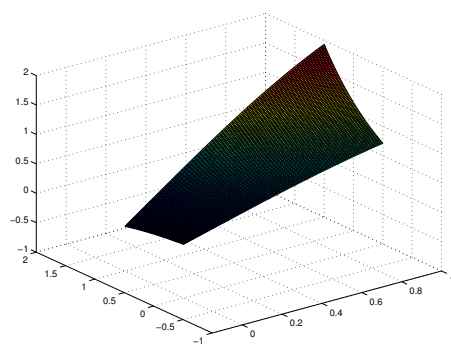
En las gráficas de la figura (4.8) se muestra la solución exacta y las aproximaciones con 21×21 , 41×41 y 81×81 nodos. El error calculado en cada nivel de tiempo como

$$\|e^k\|_2 = \sqrt{\sum_{i,j} (u_{i,j}^k - U_{i,j}^k)^2 \mathcal{A}_{i,j}},$$

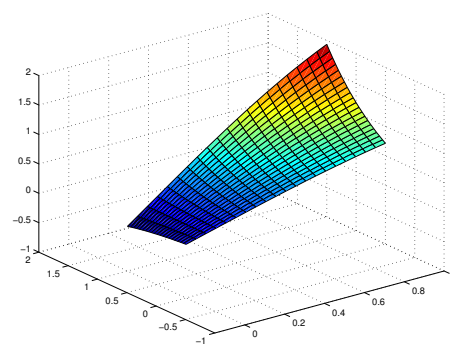
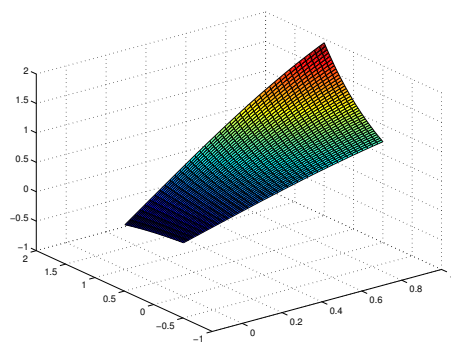
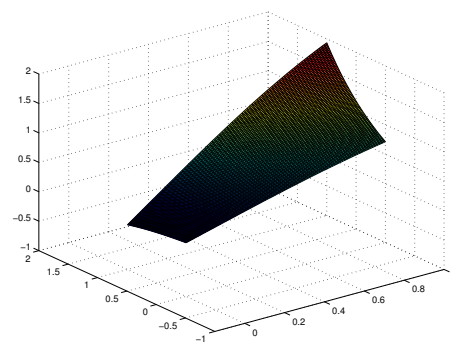
donde $u_{i,j}^k$ y $U_{i,j}^k$ son el valor exacto y aproximado de la solución, respectivamente, calculado en el elemento i, j , y $\mathcal{A}_{i,j}$ es el área del polígono definido por $\{P_{i+1,j}, P_{i,j+1}, P_{i-1,j}, P_{i,j-1}\}$ y el paso de tiempo k . Estos errores son reportados en la tabla (4.7).

Al igual que en el caso en $1 + 1D$. los datos mostrados en estas tablas son:

- NODOS: Número de nodos en que se divide la región.
- TIEMPO: Valor de tiempo en que se mide el error.
- ERROR: Error cuadrático respecto a la solución real del problema.



Solución exacta.

Lax-Wendroff con 21×21 nodos.Lax-Wendroff con 41×41 nodos.Lax-Wendroff con 81×81 nodos.Figura 4.8: Comparación de resultados método de Lax-Wendroff en $2 + 1D$.

NODOS	TIEMPO	ERROR
21×21	0.2	5.628927E-06
	0.4	1.095435E-05
	0.6	1.604631E-05
	0.8	2.094988E-05
	1.0	2.567461E-05
41×41	0.2	1.787963E-06
	0.4	3.399628E-06
	0.6	4.844548E-06
	0.8	6.131544E-06
	1.0	7.263796E-06
81×81	0.2	2.601839E-06
	0.4	5.066541E-06
	0.6	7.398977E-06
	0.8	9.602191E-06
	1.0	1.166894E-05

Tabla 4.7: Errores para el método de Lax-Wendroff en $2 + 1D$.

Con una simple inspección de la tabla (4.7) se puede comprobar que el esquema tiene un comportamiento como es el esperado, el error crece conforme al tiempo debido a la calidad de la aproximación y a que los errores se van acumulando en cada nuevo nivel de tiempo; y la aproximación mejora haciendo un refinamiento en la malla de la región. Más adelante se hará una comparación de éste método con los esquemas que se proponen, resolviendo el mismo problema.

4.1.3. Esquemas en $2 + 1D$: Esquema Explícito de 4 Puntos

Los esquemas propuestos de 4 y 6 puntos pueden ser usados en mallas tanto regulares como irregulares, *i.e. en mallas no uniformes también*. Al igual que en el caso anterior, utilizaremos regiones planas Ω simplemente conexas, cuya frontera $\partial\Omega$ sea un polígono de Jordan orientado positivamente y que cumpla con que $\partial\Omega = S_1 \cup S_2$, donde, una vez más, S_1 y S_2 estén conectados (fig. 4.9).

Para estos métodos se seleccionaron cuatro regiones de prueba, que son denotadas como CAB, HAB, MIC y UC. Se generaron mallas con 21, 41 y 81 puntos por lado para cada una de estas regiones utilizando el funcional

$$\sigma H_\omega + (1 - \sigma) F_{A^2}$$

con $\sigma = 1/2$ que se encuentra implementado en UNAMALLA [31]. Con fines comparativos estos esquemas también fueron aplicados a la región rectangular $[0, 1] \times [0, 1]$ con 21, 41 y 81 puntos por lado, donde se aproximó nuevamente la función

$$\sin(x - 0.1t)e^{y-0.1t}$$

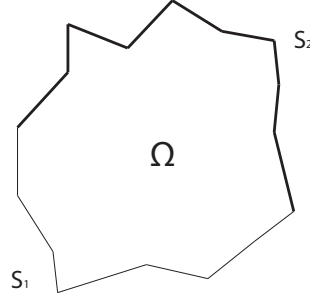


Figura 4.9: Dominio Ω irregular con $\partial\Omega = S_1 \cup S_2$.

utilizada como condición inicial y de frontera en S_1 . El intervalo de tiempo $[0, 1]$ fue dividido de manera uniforme en 1000 subintervalos.

Para las otras regiones fueron utilizadas diferentes funciones como condición inicial y de frontera en S_1 , dependiendo de la geometría de cada región, intentado así que el pulso que describe cada función iniciara en el centroide de la región. Las funciones seleccionadas para esto fueron

$$\text{CAB: } u(x, y, t) = 0.2e^{(-(x-0.5-0.3t)^2 - (y-0.3-0.3t)^2)/.01},$$

$$\text{HA: } u(x, y, t) = 0.2e^{(-(x-0.45-0.3t)^2 - (y-0.5-0.3t)^2)/.01},$$

$$\text{MIC: } u(x, y, t) = 0.2e^{(-(x-0.4-0.3t)^2 - (y-0.25-0.3t)^2)/.01},$$

$$\text{UC: } u(x, y, t) = 0.2e^{(-(x-0.45-0.3t)^2 - (y-0.5-0.3t)^2)/.01}.$$

La tabla (4.8) muestra los errores obtenidos al usar el esquema explícito de cuatro puntos en el rectángulo $[0, 1] \times [0, 1]$ aproximando la función $\sin(x - 0.1t)e^{y-0.1t}$. Las gráficas obtenidas con este esquema para este problema pueden verse en la figura (4.10).

La tabla (4.9) muestra los errores obtenidos para las regiones CAB, HA, MIC y UC, utilizando en cada una de ellas la función que le corresponde. Las gráficas mostradas en las figuras (4.11), (4.12), (4.13) y (4.14) corresponden a las aproximaciones en estas regiones en el tiempo $t = 1$. Se puede ver que, a pesar de la irregularidad de las regiones, las aproximaciones logradas son aceptables y que el comportamiento buscado continua, mientras más se refine la malla, mayor calidad de la aproximación tendremos.

4.1.4. Esquemas en 2+1D: Esquema Implícito de 4 Puntos

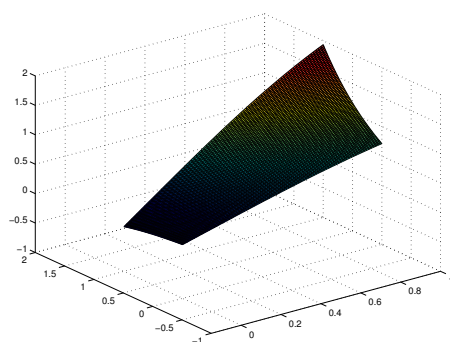
En el esquema implícito de 4 puntos hacemos las mismas pruebas que con el esquema explícito de 4 puntos. Primero, con fines de comparación, resolvemos

NODOS	TIEMPO	ERROR
21×21	0.2	1.242898E-05
	0.4	2.410326E-05
	0.6	3.506288E-05
	0.8	4.533529E-05
	1.0	5.489662E-05
41×41	0.2	3.286637E-06
	0.4	6.385320E-06
	0.6	9.303855E-06
	0.8	1.204727E-05
	1.0	1.460778E-05
81×81	0.2	1.046689E-06
	0.4	2.057142E-06
	0.6	3.025426E-06
	0.8	3.949854E-06
	1.0	4.825828E-06

Tabla 4.8: Errores para el esquema de 4 puntos en el rectángulo $[0, 1] \times [0, 1]$.

NODOS	TIEMPO	E-CAB	E-HA	E-MIC	E-UC
21×21	0.2	5.069154E-03	5.660705E-03	5.954066E-03	3.969107E-03
	0.4	7.946386E-03	9.178673E-03	9.335736E-03	6.408144E-03
	0.6	9.335199E-03	1.166374E-02	1.152109E-02	8.096681E-03
	0.8	9.493458E-03	1.342250E-02	1.312104E-02	9.247494E-03
	1.0	5.368555E-03	1.440957E-02	1.428179E-02	9.478101E-03
41×41	0.2	2.890902E-03	3.272535E-03	3.377792E-03	2.111557E-03
	0.4	4.935039E-03	5.800321E-03	5.724077E-03	3.636245E-03
	0.6	6.086828E-03	7.865132E-03	7.475342E-03	4.803203E-03
	0.8	6.313563E-03	9.534951E-03	8.899328E-03	5.621319E-03
	1.0	3.558740E-03	1.055537E-02	1.004930E-02	6.058625E-03
81×81	0.2	1.533863E-03	1.762528E-03	1.797068E-03	1.091610E-03
	0.4	2.785911E-03	3.332335E-03	3.216870E-03	1.967069E-03
	0.6	3.589348E-03	4.763404E-03	4.388266E-03	2.680803E-03
	0.8	3.822691E-03	6.052521E-03	5.418092E-03	3.197199E-03
	1.0	2.190477E-03	6.947383E-03	6.321243E-03	3.525116E-03

Tabla 4.9: Errores para el esquema de 4 puntos.



Solución exacta.

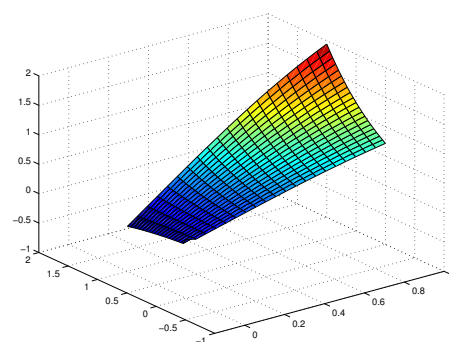
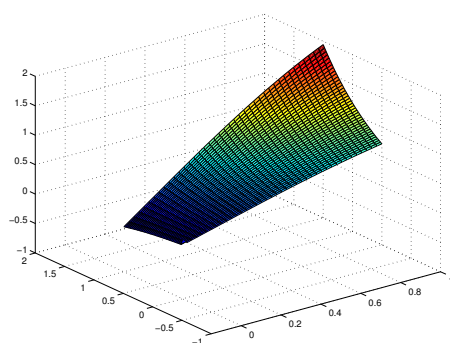
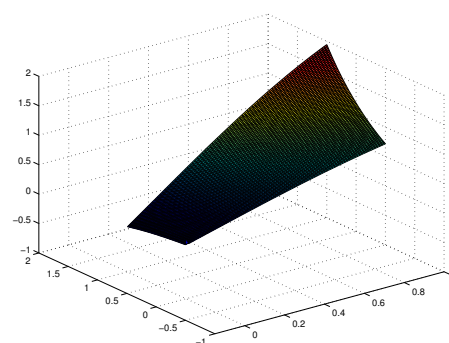
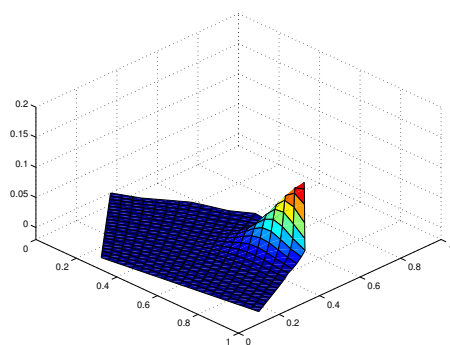
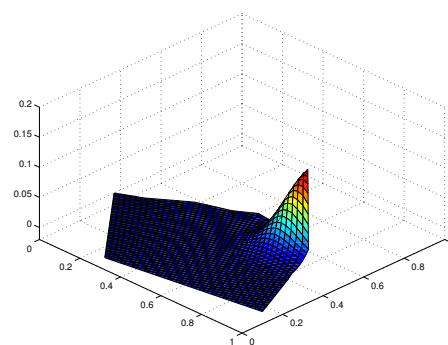
4 puntos explícito con 21×21 nodos.4 puntos explícito con 41×41 nodos.4 puntos explícito con 81×81 nodos.

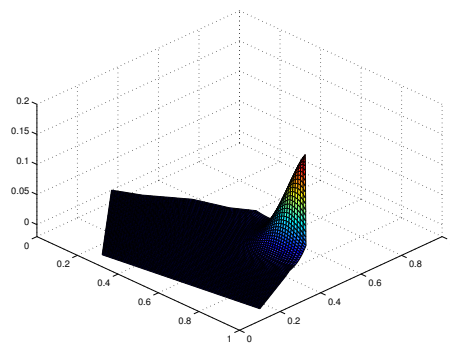
Figura 4.10: Comparación de resultados del esquema de 4 puntos en el rectángulo $[0, 1] \times [0, 1]$.



4 puntos explícito con 21×21 nodos.

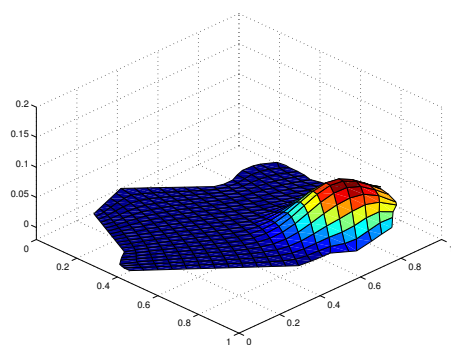


4 puntos explícito con 41×41 nodos.

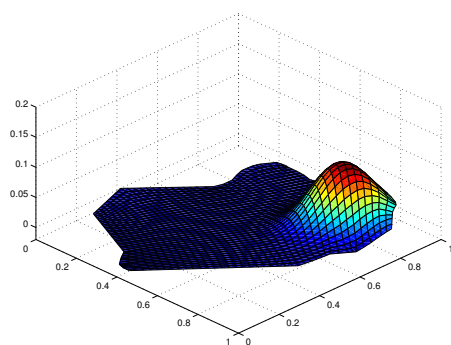


4 puntos explícito con 81×81 nodos.

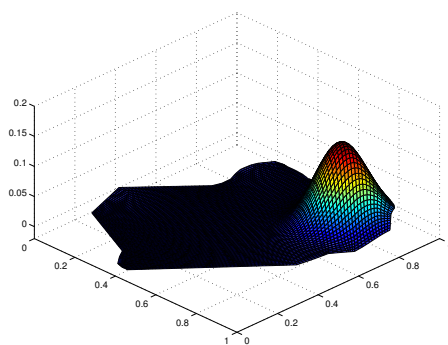
Figura 4.11: Comparación de resultados del esquema de 4 puntos en CAB.



4 puntos explícito con 21×21 nodos.



4 puntos explícito con 41×41 nodos.



4 puntos explícito con 81×81 nodos.

Figura 4.12: Comparación de resultados del esquema de 4 puntos en HA.

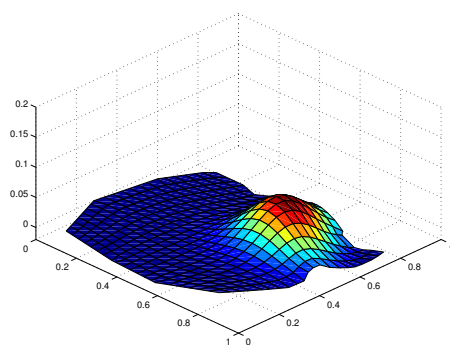
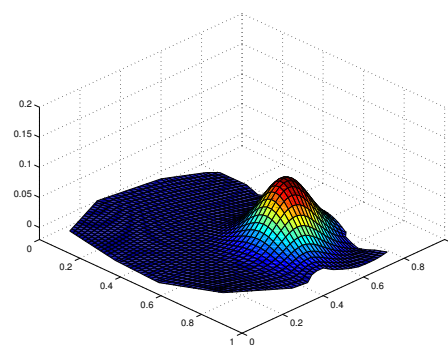
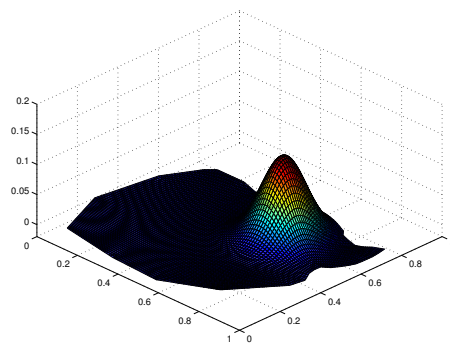
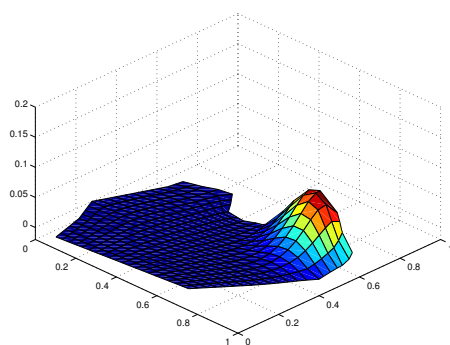
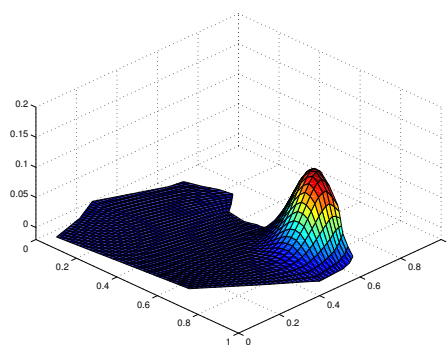
4 puntos explícito con 21×21 nodos.4 puntos explícito con 41×41 nodos.4 puntos explícito con 81×81 nodos.

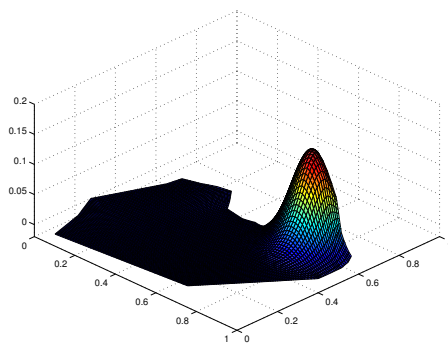
Figura 4.13: Comparación de resultados del esquema de 4 puntos en MIC.



4 puntos explícito con 21×21 nodos.



4 puntos explícito con 41×41 nodos.



4 puntos explícito con 81×81 nodos.

Figura 4.14: Comparación de resultados del esquema de 4 puntos en UC.

NODOS	TIEMPO	ERROR
21×21	0.2	3.660232E-05
	0.4	6.676841E-05
	0.6	9.110240E-05
	0.8	1.101099E-04
	1.0	1.242161E-04
41×41	0.2	1.090654E-05
	0.4	1.998815E-05
	0.6	2.739095E-05
	0.8	3.324663E-05
	1.0	3.766604E-05
81×81	0.2	4.604946E-06
	0.4	8.507292E-06
	0.6	1.175082E-05
	0.8	1.437762E-05
	1.0	1.642166E-05

Tabla 4.10: Errores para el esquema de 4 puntos implícito en el rectángulo $[0, 1] \times [0, 1]$.

el problema en el rectángulo $[0, 1] \times [0, 1]$, utilizando la función adecuada. Los resultados de la aplicación del método implícito a este problema se muestran en la tabla (4.10) y, al igual que en el caso anterior, en la figura (4.15) se muestran las gráficas de las soluciones en el tiempo $t = 1$.

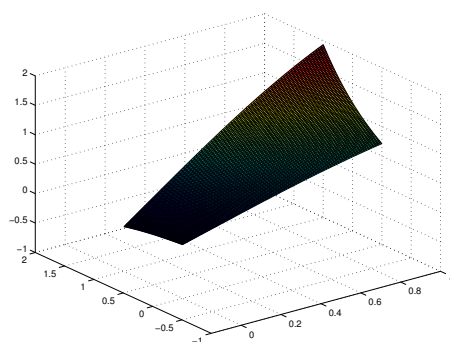
Hay que hacer notar que, para todos los resultados de este trabajo, en el esquema implícito se utilizó el valor de $\lambda = 1/2$.

Los resultados para las regiones de prueba se muestran en la tabla (4.11). También, en las figuras (4.16), (4.17), (4.18) y (4.19) se muestran las gráficas de la aproximación en el tiempo $t = 1$ en cada una de las regiones.

4.1.5. Esquemas en $2 + 1D$: Esquema de 6 Puntos

El esquema de 6 puntos es el que mayor precisión aporta. Es un esquema de bajo costo computacional y aún así logra aproximaciones muy buenas a la solución de la ecuación de advección. Con este esquema se corrigen muchos de los pequeños problemas de estabilidad que se encontraban con otros esquemas en $2 + 1D$. Para ejemplificarlo, en la tabla (4.12) podemos ver los resultados obtenidos de resolver el problema inicial en el rectángulo $[0, 1] \times [0, 1]$. Las gráficas que se obtuvieron para este problema en el tiempo $t = 1$ se muestran en la figura (4.20).

También podemos notar una mejoría en cuanto a las aproximaciones logradas para las otras regiones de prueba. Todos los resultados se muestran en la tabla



Solución exacta.

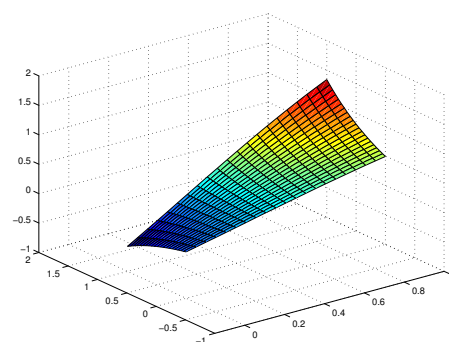
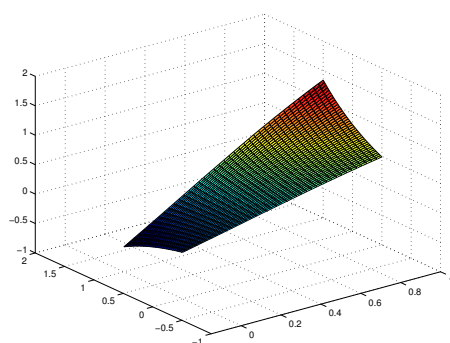
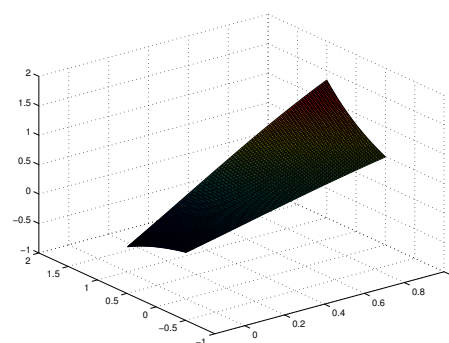
4 puntos implícito con 21×21 nodos.4 puntos implícito con 41×41 nodos.4 puntos implícito con 81×81 nodos.

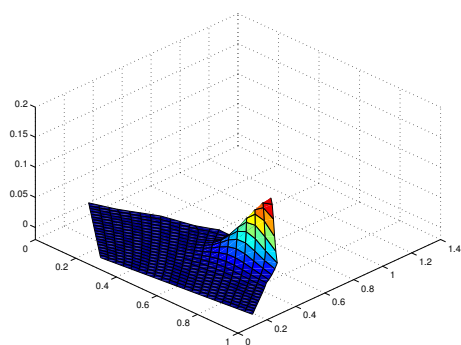
Figura 4.15: Comparación de resultados del esquema de 4 puntos implícito en el rectángulo $[0, 1] \times [0, 1]$.

NODOS	TIEMPO	E-CAB	E-HA	E-MIC	E-UC
21×21	0.2	5.109512E-03	5.690985E-03	5.982824E-03	3.999472E-03
	0.4	8.002868E-03	9.218056E-03	9.375041E-03	6.453467E-03
	0.6	9.404829E-03	1.170434E-02	1.156482E-02	8.155374E-03
	0.8	9.563273E-03	1.346102E-02	1.316638E-02	9.306919E-03
	1.0	5.396373E-03	1.444778E-02	1.432672E-02	9.528745E-03
41×41	0.2	2.946142E-03	3.316067E-03	3.418260E-03	2.146587E-03
	0.4	5.021366E-03	5.865485E-03	5.785877E-03	3.693848E-03
	0.6	6.194423E-03	7.940619E-03	7.550244E-03	4.883971E-03
	0.8	6.421953E-03	9.613585E-03	8.983027E-03	5.711171E-03
	1.0	3.600548E-03	1.063479E-02	1.013755E-02	6.142772E-03
81×81	0.2	1.599522E-03	1.815720E-03	1.845411E-03	1.129765E-03
	0.4	2.899483E-03	3.422036E-03	3.297799E-03	2.033616E-03
	0.6	3.736679E-03	4.878174E-03	4.493575E-03	2.777132E-03
	0.8	3.976744E-03	6.183562E-03	5.543240E-03	3.308591E-03
	1.0	2.253060E-03	7.085660E-03	6.461569E-03	3.638419E-03

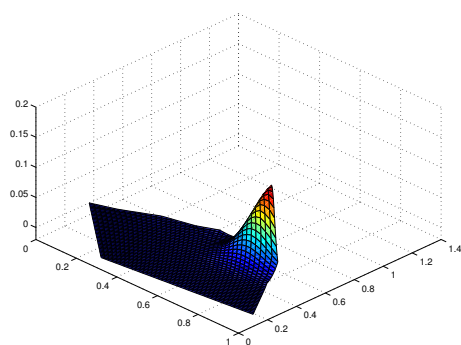
Tabla 4.11: Errores para el esquema de 4 puntos implícito.

NODOS	TIEMPO	ERROR
21×21	0.2	5435518E-06
	0.4	1000364E-05
	0.6	1343282E-05
	0.8	1598972E-05
	1.0	1796252E-05
41×41	0.2	1240188E-06
	0.4	2187412E-06
	0.6	2964444E-06
	0.8	3659482E-06
	1.0	4286517E-06
81×81	0.2	2860577E-07
	0.4	5232744E-07
	0.6	7420528E-07
	0.8	9453293E-07
	1.0	1131465E-06

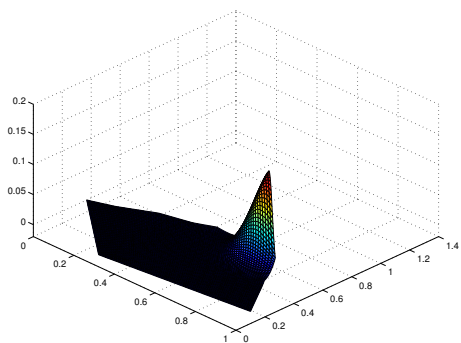
Tabla 4.12: Errores para el esquema de 6 puntos en el rectángulo $[0, 1] \times [0, 1]$.



4 puntos implícito con 21×21 nodos.

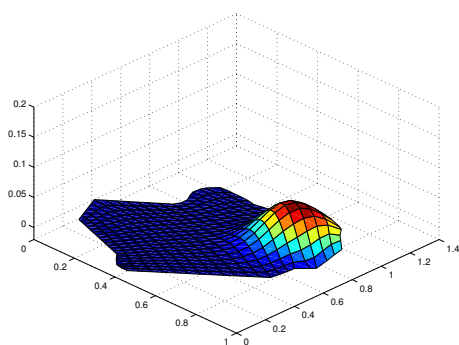


4 puntos implícito con 41×41 nodos.

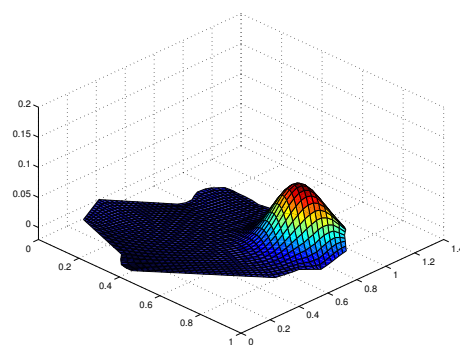


4 puntos implícito con 81×81 nodos.

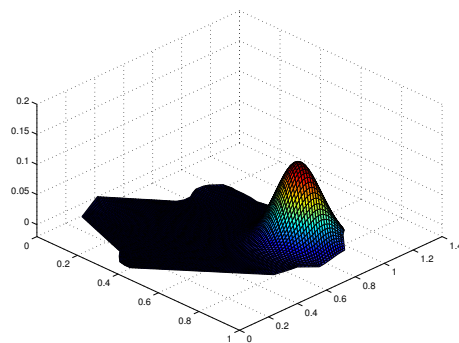
Figura 4.16: Comparación de resultados del esquema de 4 puntos en CAB.



4 puntos implícito con 21×21 nodos.

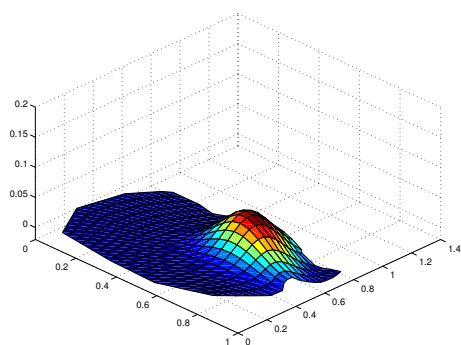


4 puntos implícito con 41×41 nodos.

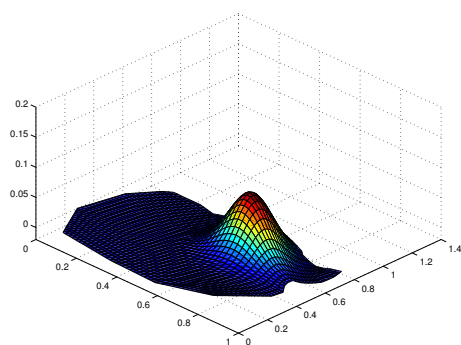


4 puntos implícito con 81×81 nodos.

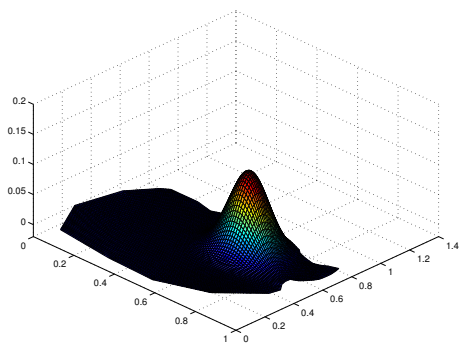
Figura 4.17: Comparación de resultados del esquema de 4 puntos en HA.



4 puntos implícito con 21×21 nodos.



4 puntos implícito con 41×41 nodos.



4 puntos implícito con 81×81 nodos.

Figura 4.18: Comparación de resultados del esquema de 4 puntos en MIC.

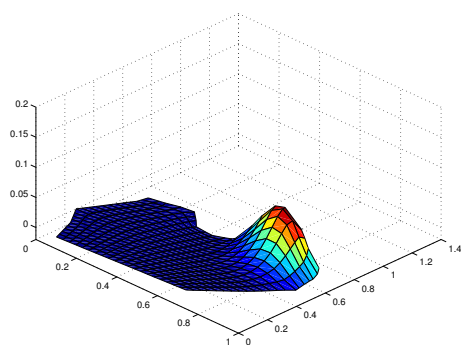
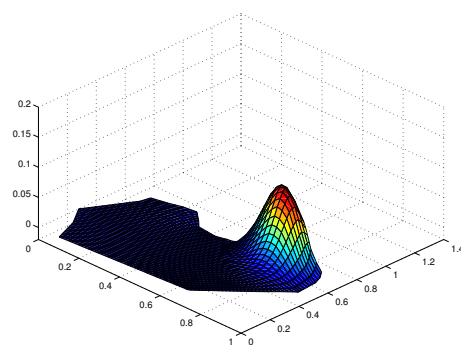
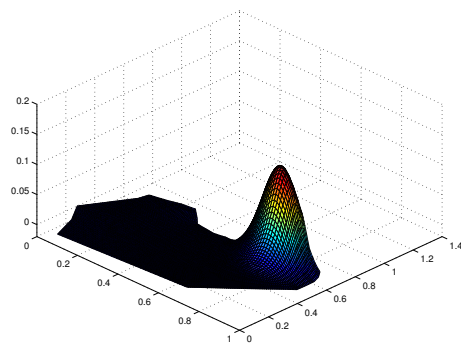
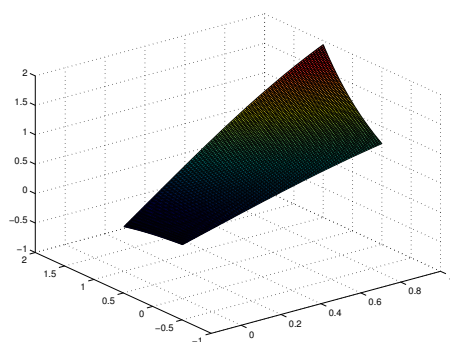
4 puntos implícito con 21×21 nodos.4 puntos implícito con 41×41 nodos.4 puntos implícito con 81×81 nodos.

Figura 4.19: Comparación de resultados del esquema de 4 puntos en UC.



Solución exacta.

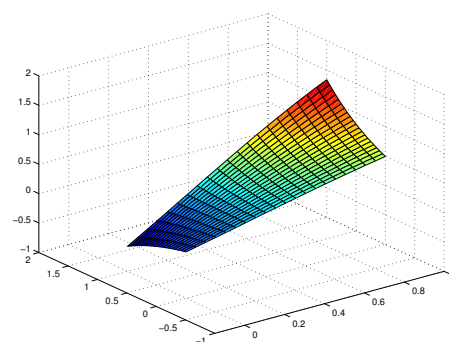
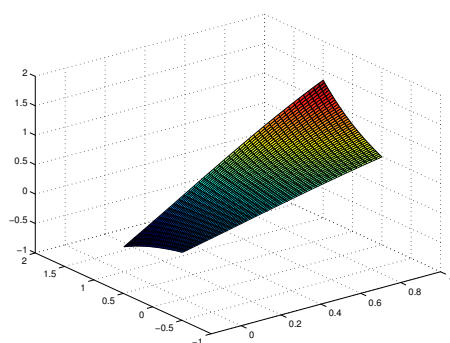
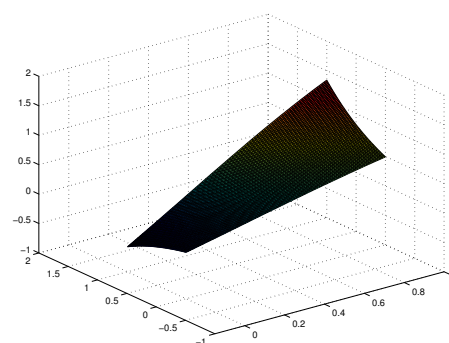
6 puntos con 21×21 nodos.6 puntos con 41×41 nodos.6 puntos con 81×81 nodos.

Figura 4.20: Comparación de resultados del esquema de 6 puntos en el rectángulo $[0, 1] \times [0, 1]$.

NODOS	TIEMPO	E-CAB	E-HA	E-MIC	E-UC
21×21	0.2	2.948721E-03	3.614275E-03	3.331088E-03	2.432902E-03
	0.4	5.184152E-03	6.142163E-03	5.737464E-03	4.468752E-03
	0.6	6.614910E-03	3.322508E-02	7.621632E-03	6.138760E-03
	0.8	6.577109E-03	2.170297E01	9.069541E-03	7.293793E-03
	1.0	4.289068E-03	1.414776E04	1.014699E-02	6.433778E-03
41×41	0.2	8.738800E-04	1.170273E-03	9.940816E-04	7.176966E-04
	0.4	1.746007E-03	2.268737E-03	1.904652E-03	1.456777E-03
	0.6	2.452958E-03	3.332595E-03	2.747314E-03	2.216038E-03
	0.8	2.528250E-03	2.299754E-02	3.523979E-03	2.868712E-03
	1.0	1.841478E-03	6.995041E-00	4.181122E-03	2.689677E-03
81×81	0.2	2.202936E-04	3.036863E-04	2.513862E-04	1.820511E-04
	0.4	4.551862E-04	6.158075E-04	4.951184E-04	3.795539E-04
	0.6	6.708897E-04	9.450673E-04	7.323069E-04	5.927880E-04
	0.8	7.180325E-04	1.300409E-03	9.617715E-04	7.884763E-04
	1.0	5.772294E-04	1.641537E-03	1.175040E-03	7.896830E-04

Tabla 4.13: Errores para el esquema de 6 puntos.

(4.13), en la cual podemos ver que el error llega a ser hasta un orden menor que en los esquemas de 4 puntos. Las gráficas mostradas en las figuras (4.21), (4.22), (4.23) y (4.24) muestran la aproximación obtenida en el nivel de tiempo $t = 1$, como en los casos anteriores. Debido a cuestiones de estabilidad, en el caso para la región HAB (4.22) diverge en la frontera S_2 cuando la malla es de pocos nodos, pero se puede notar que se comporta bien al momento de refinar la misma malla.

En la tabla (4.14) se muestra el tiempo de ejecución, en segundos, para cada una de las regiones de muestra.

ESQUEMA	Nodos	CAB	HA	MIC	UC
4 puntos explícito	21	0.373407s	0.668759s	0.505415s	0.528172s
	41	1.203611s	1.505135s	1.529723s	1.543996s
	81	4.823690s	5.172947s	5.205134s	5.168015s
4 puntos implícito	21	1.158553s	1.357246s	1.342591s	1.356649s
	41	4.697035s	4.710679s	4.834741s	4.901759s
	81	17.95600s	18.50425s	18.24829s	18.58922s
6 puntos	21	0.459241s	0.542580s	0.524271s	0.543279s
	41	1.381222s	1.621193s	1.653820s	1.684437s
	81	5.182945s	5.780940s	5.955649s	5.889886s

Tabla 4.14: Tiempos de ejecución para cada una de las regiones.

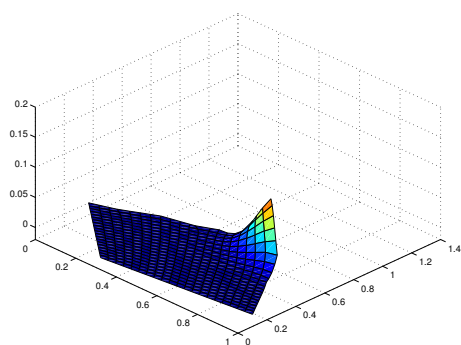
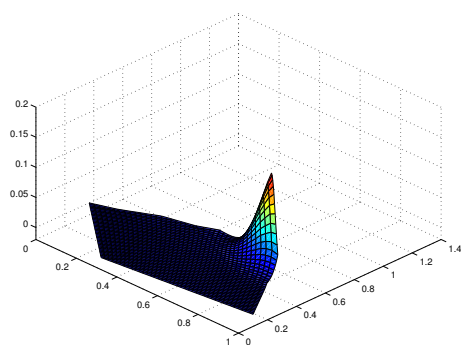
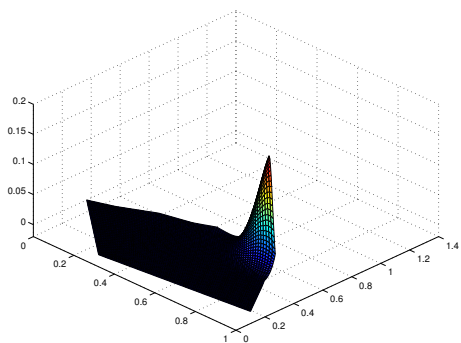
6 puntos con 21×21 nodos.6 puntos con 41×41 nodos.6 puntos con 81×81 nodos.

Figura 4.21: Comparación de resultados del esquema de 4 puntos en CAB.

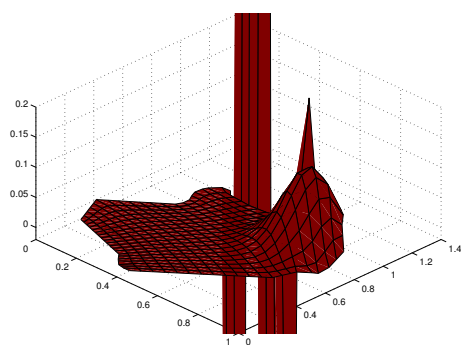
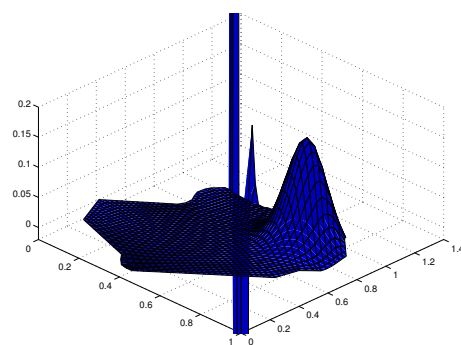
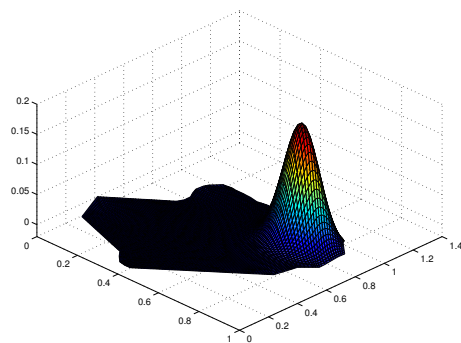
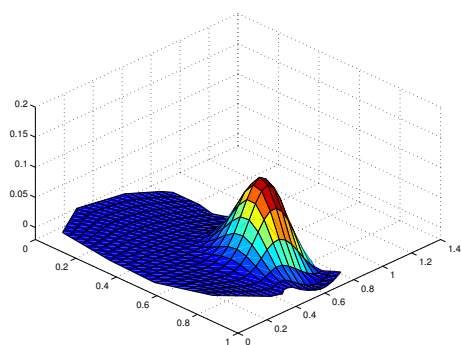
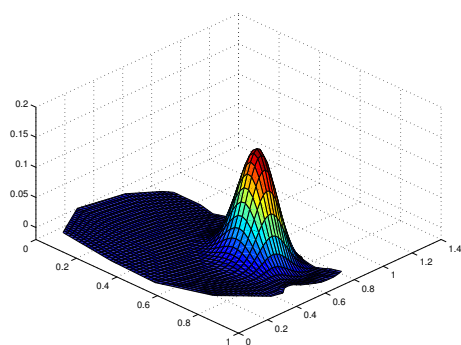
6 puntos con 21×21 nodos.6 puntos con 41×41 nodos.6 puntos con 81×81 nodos.

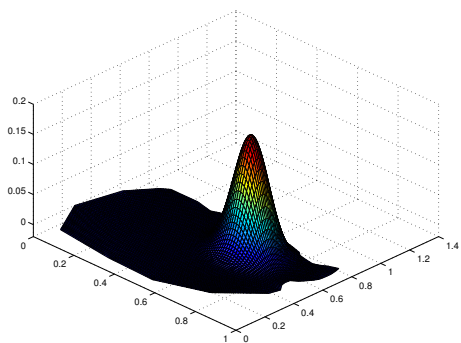
Figura 4.22: Comparación de resultados del esquema de 4 puntos en HA.



6 puntos con 21×21 nodos.



6 puntos con 41×41 nodos.



6 puntos con 81×81 nodos.

Figura 4.23: Comparación de resultados del esquema de 4 puntos en MIC.

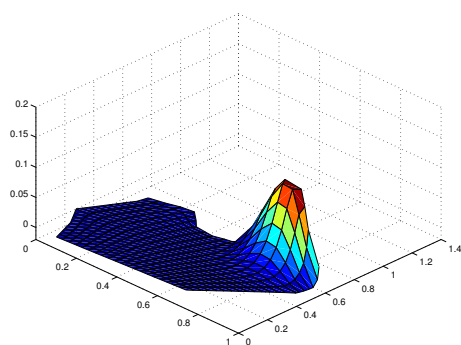
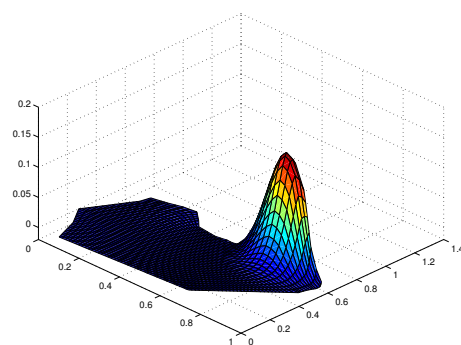
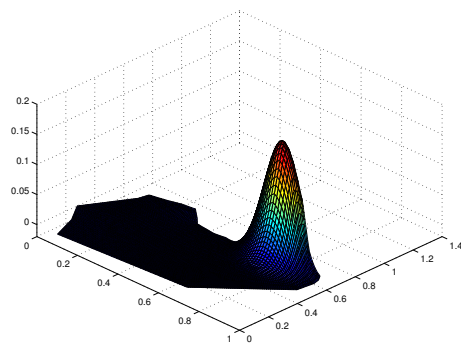
6 puntos con 21×21 nodos.6 puntos con 41×41 nodos.6 puntos con 81×81 nodos.

Figura 4.24: Comparación de resultados del esquema de 6 puntos en UC.

4.2. Conclusiones

La estabilidad es un asunto esencial cuando se estudia la convergencia de esquemas numéricos, y es muy difícil, comúnmente, obtener los límites de estabilidad para estos esquemas. Un estudio de estas dificultades puede encontrarse en [15].

Aún cuando la ecuación de advección es una EDP muy difícil de tratar numéricamente, en este trabajo se presentan 3 esquemas que logran aproximaciones buenas a la solución. Un asunto muy importante sigue siendo la geometría de la región, ya que las pruebas realizadas para el presente trabajo muestran que, aún con nuestros esquemas que logran una buena precisión, cuando la geometría de la región es muy irregular se encuentran algunas oscilaciones espurias en las fronteras. Algunos autores han tratado este problema de la frontera, utilizando un parche donde se supone, *a priori*, que se conoce la solución exacta de la ecuación en la frontera.

La comparación que se hace de los esquemas propuestos muestra que las mejores aproximaciones se logran con el esquema de 6 puntos. En buena parte esto se debe a que, al utilizar más puntos para aproximar la solución, se cuenta con más datos con los que trabajar.

En este trabajo se presentan 3 esquemas que, como se puede ver en la tabla (4.14), son de bajo costo computacional, y que además son relativamente fáciles de programar (ver Anexo A). Todos los programas y las ejecuciones se hicieron utilizando MATLAB[®] R2013a [32], usando una computadora MacBook Pro Mid 2012 con procesador Inter(R) Core(TM) i5 a 2.5 GHz y 4.00 GB de memoria RAM a 1600 MHz corriendo bajo MacOSx 10.9.2 como sistema operativo.

Bibliografía

- [1] Barrera-Sánchez,P., Castellanos-Noda, L., Pérez-Domínguez, A. *Métodos Variacionales Discretos para la Generación de Mallas*. DGAPA-UNAM, México, D.F., 1994.
- [2] Barrera-Sánchez, P. and, Castellanos-Noda, L.A. *Métodos de Optimización de Gran Escala para el Problema de la Generación de Redes Optimas*. DGAPA-UNAM, México, D.F., 1994.
- [3] Barrera-Sánchez,P., González-Flores, G., Pérez-Domínguez, A., Castellanos-Noda, L. *Manual de Usuarios del Sistema UNAMALLA*. DGAPA-UNAM, México, D.F., 1994.
- [4] Barrera-Sánchez,P., Pérez-Domínguez, A., Castellanos-Noda, L. *Curvilinear Coordinate System Generation Over Plane Irregular Regions*. Vínculos Matemáticos No. 133, Facultad de Ciencias, UNAM, Méx. 1992.
- [5] Castillo-Moreno, José E. *On Variational Grid Generation*. Ph. D. Thesis, The University of New Mexico, Albuquerque, New Mexico. 1987.
- [6] Castillo, J.E., *Mathematical Aspects of Numerical Grid Generation I*. en *Numerical Grid Generation in Computational Fluid Dynamics*. J. Hauser and C. Taylor eds., pp 35-43. Pineridge Press, Swansea, UK. 1986.
- [7] Castillo, J. E., Steinberg, S. and Roache, P.J. *Mathematical Aspects of Variational Grid Generation II*. J. Comp. and Appl. Math. 20, 127-135. 1987.
- [8] Castillo, J. E., Steinberg, S. and Roache, P.J. *Parameter Stimulation in Variational Grid Generation*. Appl. Math. and Comp. 28(2), 1-23. 1988.
- [9] Castillo, J. E., Steinberg, S. and Roache, P.J. *On the Folding of Numerically Generated Grids. Use of a Reference Grid*. Comm. Applied Num. Methods 4, 471-481. 1988.
- [10] M. Celia and W. Gray, *Numerical Methods for Differential Equations*, Prentice-Hall, (1992).
- [11] F. Domínguez-M., S. Mendoza and J. Tinoco, *Finite Difference Schemes Satisfying an Optimality Condition*, MASCOT10 Proceedings, (2011).

- [12] P. Barrera-S., L. Castellanos-N., F. Domínguez-M., G. González-F. and A. Pérez-D., *Adaptive discrete harmonic grid generation*, *Mathematics and Computer Simulations*, 79, (2009), pp. 1892-1809.
- [13] Eiseman, P. R. *Coordinate Generation with Precise Controls over Mesh Properties*. J. Comp. Physics 47, 331-351. 1982.
- [14] Elsgoltz, L. *Ecuaciones Diferenciales y Cálculo Variacional*. Segunda Edición. Editorial MIR. Moscú. 1977.
- [15] Ercília S. *The controversial stability analysis*. Appl. Math. Comput. Vol 145, (2003), pp. 777-794.
- [16] G'elfand, I.M., and Fomin, S.V. *Calculus of Variations*. Prentice-Hall. Englewood Cliffs, N.J. 1963.
- [17] Gordon, W. J., and Hall, C. A. *Construction of Curvilinear Coordinate Systems and Applications to Mesh Generations*. International Journal for Numerical Methods in Engineering. 7, 461-477. 1973.
- [18] Gordon, W. J., and Thiel, L. C. *Transfinite Mappings and their Applications to Grid Generation*. En *Numerical Grid Generation*, J. F. Thompson, ed., pp 171-192. North Holland, New York. 1982.
- [19] González-Flores, G.F. *Generación de Mallas en Regiones Planas Irregulares*. Tesis de Licenciatura. Universidad Autónoma de Yucatán. Mérida, Méx. 1994.
- [20] A. Iserles, "A First Course in the Numerical Analysis of Differential Equations", Cambridge Texts in Applied Mathematics, Cambridge University Press, (2008).
- [21] Ivanenko, S.A., and Charakhch'yan, A.A. *Curvilinear Grids of Convex Quadrilaterals*. U.S.S.R. Comput. Maths. Math. Phys., Vol 28, No. 2. pp. 126-133. 1988.
- [22] Ivanenko, S.A. *Generation of Non-Degenerate Meshes*. U.S.S.R. Comput. Maths. Math. Phys., Vol 28, No. 5. pp. 141-146. 1988.
- [23] Ivanenko, S.A. *Adaptive grids and grids on surfaces*, Cmput. Maths. Math. Phys. 33 (9) (1993), pp. 8.1-8.41.
- [24] Knupp, Patrick M., and Steinberg, Stanley. *Fundamentals of Grid Generation*. CRC Press. Boca Raton, Ann Arbor, London, Tokyo. 1993.
- [25] Randall J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations, Steady-State and Time-Dependent Problems*. 2007.
- [26] Saunders, B. V. *Algebraic Grid Generation using Tensor Product B-Splines*. NASA Contractor Report 177968. NASA Langley Research Center, Hampton VA. 1985.

- [27] Smith, R. E. *Algebraic Grid Generation*. En *Numerical Grid Generation*, J. F. Thompson, ed., pp 137-170. North Holland, New York. 1982.
- [28] Tinoco-Ruiz, J.G., *Funcionales Discretos para la Generación de Mallas Suaves y Convexas sobre Regiones Planas Irregulares*, Tesis Doctoral. CI-MAT., Guanajuato, Gto., México. 1997.
- [29] Walter Rudin. *Principles of Mathematical Analysis*, Second Edition. 1976
- [30] William E. Boyce and Richard C. DiPrima. *Elementary Differential Equations and Boundary Value Problems*.
- [31] UNAMALLA Versión 4.0. *An automatic package for numerical grid generation*. Website: http://www.mathmoo.unam.mx/unamalla/home_e.html. Última versión publicada en Junio, 2011
- [32] MATLAB[®] R2013a. *The language of technical computing*. Website: <http://www.mathworks.com/products/matlab/>

Programas

Descripción de los programas

wendroffg.m

El siguiente código aproxima, por medio del esquema explícito de 4 puntos que se presenta en el capítulo 4 la ecuación de advección en una región descrita por las variables de entrada x, y . Se encuentra programado en el lenguaje de programación MATLAB[®] [32].

```
% Esta es la version con mas precision hasta el momento del metodo de
% Lax-Wendroff programado de manera explicita con el esquema de 4 puntos.
%
% Es importante revisar que la matriz coincida con los limites de la malla.
%
% Cuenta con ambas versiones para calcular las gammas, la version donde se
% resuelve el sistema de ecuaciones y la version donde se tienen las
% formulas explicitas.
%
% La llamada al metodo se hace:
%
%       U = wendroffg(x,y,f,a,b,t,gr,graf);
%
% Argumentos de salida
%
%       U       :   Matriz U con la aproximacion final de la funcion.
%
% Argumentos de entrada
%
%       x, y     :   Coordenadas de los puntos de la malla
%       f        :   Funcion para la condicion inicial y de frontera.
%       a, b     :   Velocidades de movimiento del pulso de la funcion f.
%                   Los valores por defecto de a y b son .1
%       t        :   Vector con los diferentes niveles de tiempo.
%                   El valor por defecto de t es 0:1/1023:2;
```

```

%      gr      :   Se utiliza para elegir la manera del calculo de Gammas.
%                0   :   Calculo clasico de Gammas (valor por defecto).
%                1   :   Calculo con formulas explicitas.
%      graf     :   Se utiliza para elegir si se grafica o no.
%                0   :   No se grafica (valor por defecto).
%                1   :   Si se grafica.
%
% La documentacion de este codigo se escribio sin acentos.
%

function U = wendroffg(x,y,f,a,b,t,gr,graf)
tic
close all

switch nargin
    case 3
        a = .1;
        b = .1;
        t = 0:1/1023:2;
        gr = 0;
        graf = 0;
    case 4
        b = .1;
        t = 0:1/1024:1;
        gr = 0;
        graf = 0;
    case 5
        t = 0:1/1024:1;
        gr = 0;
        graf = 0;
    case 6
        gr = 0;
        graf = 0;
    case 7
        graf = 0;
end

[m,n] = size(x);
nt = length(t);
dt = t(2) - t(1);
U1 = zeros(m,n,nt);
Gamma = zeros(m,n,4);

%Primero llenamos k=1 con condiciones iniciales
for i = 1:m
    for j = 1:n

```

```

        U1(i,j,1) = f([x(i,j),y(i,j)],t(1),a,b) ;
    end
end

%Ahora llenamos con condiciones de frontera
for k = 2:nt
    for i = 2:m
        U1(i,1,k) = f([x(i,1),y(i,1)],t(k),a,b);
    end
    for j = 2:n
        U1(1,j,k) = f([x(1,j),y(1,j)],t(k),a,b);
    end
end

% Calculamos las Gammas
if gr == 0
    for i = 2:m
        for j = 2:n
            dx = [x(i-1,j)-x(i,j),x(i-1,j-1)-x(i,j),x(i,j-1)-x(i,j)];
            dy = [y(i-1,j)-y(i,j),y(i-1,j-1)-y(i,j),y(i,j-1)-y(i,j)];
            M = [dx ; dy ; dx.*dx ; dx.*dy ; dy.*dy];
            YY = pinv(M)*[-a*dt -b*dt (a*dt)^2 a*b*dt^2 (b*dt)^2]';
            Gamma(i,j,:) = [-sum(YY);YY];
        end
    end
else
    % Nuevas maneras de calcular los Gammas
    for i = 2:m
        for j = 2:n
            h1 = x(i-1,j)-x(i,j);
            h2 = x(i-1,j-1)-x(i,j);
            h3 = x(i,j-1)-x(i,j);
            k1 = y(i-1,j)-y(i,j);
            k2 = y(i-1,j-1)-y(i,j);
            k3 = y(i,j-1)-y(i,j);
            s1 = (h2*k3-h3*k2)/(h1*k2-h2*k1);
            s2 = (h3*k1-h1*k3)/(h1*k2-h2*k1);
            al = -a*dt;
            be = -b*dt;
            ga = (a*dt)^2;
            de = a*b*dt^2;
            et = (b*dt)^2;
            t1 = (al*k2-be*h2)/(h1*k2-h2*k1);
            t2 = (-al*k1+be*h1)/(h1*k2-h2*k1);
            Gamma(i,j,4) = ((h1^2*s1 + h2^2*s2 + h3^2)*(ga - h1^2*t1 - h2^2*t2) + ...

```

```

        (k1*h1*s1 + k2*h2*s2 + k3*h3)*(de - k1*h1*t1 - k2*h2*t2)+ ...
        (k1^2*s1 + k2^2*s2 + k3^2)*(et - k1^2*t1 - k2^2*t2))/ ...
        ((h1^2*s1 + h2^2*s2 + h3^2)^2 + (k1*h1*s1 + k2*h2*s2 + k3*h3)^2 + ...
        (k1^2*s1 + k2^2*s2 + k3^2)^2);
    Gamma(i,j,2) = s1*Gamma(i,j,4) + t1;
    Gamma(i,j,3) = s2*Gamma(i,j,4) + t2;
    Gamma(i,j,1) = - Gamma(i,j,2) - Gamma(i,j,3) - Gamma(i,j,4);
end
end
end

U = U1;

% Llenamos ahora los nodos interiores
for k = 2:nt
    for i = 2:m
        for j = 2:n
            U(i,j,k) = U(i,j,k-1) + Gamma(i,j,1)*U(i,j,k-1) + Gamma(i,j,2)*U(i-1,j,k-1) +
                Gamma(i,j,3)*U(i-1,j-1,k-1)+Gamma(i,j,4)*U(i,j-1,k-1);
        end
    end
end
end
toc

% Hacemos la grafica
if graf == 1
    for i=1:50:nt
        surf(x,y,U(:,:,i));
        axis([0 1 0 1 -.02 .2]);
        view([45 45]);
        pause(0.01)
    end
end
end
end

```

El resultado de este programa es una matriz que representa la aproximación a la solución en cada uno de los puntos del dominio.

implicit5.m

El siguiente código aproxima, por medio del esquema implícito de 4 puntos que se presenta en el capítulo 4 la ecuación de advección en una región descrita por las variables de entrada x, y .

```

% Esta es la version con mas precision hasta el momento del metodo de
% Lax-Wendroff programado de manera implicita con el esquema de 4 puntos.
%
% Es importante revisar que la matriz coincida con los limites de la malla.
%
% Esta version funciona calculando el metodo explicito primeramente para
% utilizar como semilla del metodo implicito.
% Cuenta con ambas versiones para calcular las gammas, la version donde se
% resuelve el sistema de ecuaciones y la version donde se tienen las
% formulas explicitas.
%
% La llamada al metodo se hace:
%
%       U = implicit5(x,y,f,a,b,t,gr);
%
% Argumentos de salida
%
%       U       :   Matriz U con la aproximacion final de la funcion.
%
% Argumentos de entrada
%
%       x, y     :   Coordenadas de los puntos de la malla
%       f        :   Funcion para la condicion inicial y de frontera.
%       a, b     :   Velocidades de movimiento del pulso de la funcion f.
%                   Los valores por defecto de a y b son .1
%       t        :   Vector con los diferentes niveles de tiempo.
%                   El valor por defecto de t es 0:1/1023:2;
%       gr       :   Se utiliza para elegir la manera del calculo de Gammas.
%                   0   :   Calculo clasico de Gammas (valor por defecto).
%                   1   :   Calculo con formulas explicitas.
%       graf     :   Se utiliza para elegir si se grafica o no.
%                   0   :   No se grafica (valor por defecto).
%                   1   :   Si se grafica.
%
%
% La documentacion de este codigo se escribio sin acentos.
%

function [U, iter] = implicit5(x,y,f,a,b,t,gr,graf)
tic
close all

switch nargin
    case 3
        a = .1;
        b = .1;

```

```

        t = 0:1/1023:2;
        gr = 0;
        graf = 0;
    case 4
        b = .1;
        t = 0:1/1024:1;
        gr = 0;
        graf = 0;
    case 5
        t = 0:1/1024:1;
        gr = 0;
        graf = 0;
    case 6
        gr = 0;
        graf = 0;
    case 7
        graf = 0;
end

% Inicializamos variables
lambda = .5;
[m,n] = size(x);
nt = length(t);
dt = t(2)-t(1);
U1 = zeros(m,n,nt);
Gamma = zeros(m,n,4);
iter = zeros(nt,1);

% Llenamos con condicion inicial.
for i = 1:m
    for j = 1:n
        U1(i,j,1) = f([x(i,j),y(i,j)],t(1),a,b);
    end
end

% Llenamos con condicion de frontera.
for k = 2:nt
    for i = 1:m
        U1(i,1,k) = f([x(i,1),y(i,1)],t(k),a,b);
    end
    for j = 2:n
        U1(1,j,k) = f([x(1,j),y(1,j)],t(k),a,b);
    end
end
end

```

```

% Calculamos las Gammas
if gr == 0
    for i = 2:m
        for j = 2:n
            dx = [x(i-1,j)-x(i,j),x(i-1,j-1)-x(i,j),x(i,j-1)-x(i,j)];
            dy = [y(i-1,j)-y(i,j),y(i-1,j-1)-y(i,j),y(i,j-1)-y(i,j)];
            M = [dx ; dy ; dx.*dx ; dx.*dy ; dy.*dy];
            YY = pinv(M)*[-a*dt -b*dt (a*dt)^2 a*b*dt^2 (b*dt)^2]';
            Gamma(i,j,:) = [-sum(YY);YY];
        end
    end
else
    % Nuevas maneras de calcular los Gammas
    for i = 2:m
        for j = 2:n
            h1 = x(i-1,j)-x(i,j);
            h2 = x(i-1,j-1)-x(i,j);
            h3 = x(i,j-1)-x(i,j);
            k1 = y(i-1,j)-y(i,j);
            k2 = y(i-1,j-1)-y(i,j);
            k3 = y(i,j-1)-y(i,j);
            s1 = (h2*k3-h3*k2)/(h1*k2-h2*k1);
            s2 = (h3*k1-h1*k3)/(h1*k2-h2*k1);
            al = -a*dt;
            be = -b*dt;
            ga = (a*dt)^2;
            de = a*b*dt^2;
            et = (b*dt)^2;
            t1 = (al*k2-be*h2)/(h1*k2-h2*k1);
            t2 = (-al*k1+be*h1)/(h1*k2-h2*k1);
            Gamma(i,j,4) = ((h1^2*s1 + h2^2*s2 + h3^2)*(ga - h1^2*t1 - h2^2*t2) + ...
                (k1*h1*s1 + k2*h2*s2 + k3*h3)*(de - k1*h1*t1 - k2*h2*t2)+ ...
                (k1^2*s1 + k2^2*s2 + k3^2)*(et - k1^2*t1 - k2^2*t2))/ ...
                ((h1^2*s1 + h2^2*s2 + h3^2)^2 + (k1*h1*s1 + k2*h2*s2 + k3*h3)^2 + ...
                (k1^2*s1 + k2^2*s2 + k3^2)^2);
            Gamma(i,j,2) = s1*Gamma(i,j,4) + t1;
            Gamma(i,j,3) = s2*Gamma(i,j,4) + t2;
            Gamma(i,j,1) = - Gamma(i,j,2) - Gamma(i,j,3) - Gamma(i,j,4);
        end
    end
end

U = U1;

for k = 2:nt

```



```

er = 1;
for i = 2:m
    for j = 2:n
        U(i,j,k) = U(i,j,k-1) + ...
            Gamma(i,j,1)*U(i,j,k-1) + ...
            Gamma(i,j,2)*U(i-1,j,k-1) + ...
            Gamma(i,j,3)*U(i-1,j-1,k-1) + ...
            Gamma(i,j,4)*U(i,j-1,k-1);
    end
end
while er >= sqrt(eps)
    Z = U(:, :, k);
    for i = 2:m
        for j = 2:n
            U(i,j,k) = (U(i,j,k-1) + ...
                lambda*( ...
                    Gamma(i,j,1)*U(i,j,k-1) + ...
                    Gamma(i,j,2)*U(i-1,j,k-1) + ...
                    Gamma(i,j,3)*U(i-1,j-1,k-1) + ...
                    Gamma(i,j,4)*U(i,j-1,k-1)) + ...
                (1-lambda)*( ...
                    Gamma(i,j,2)*U(i-1,j,k) + ...
                    Gamma(i,j,3)*U(i-1,j-1,k) + ...
                    Gamma(i,j,4)*U(i,j-1,k))) ...
                /(1+(lambda-1)*Gamma(i,j,1)));
        end
    end
    iter(k) = iter(k) + 1;
    er = max(max(abs(U(:, :, k) - Z)));
end
end
toc

% Hacemos la grafica
if graf == 1
    for i=1:50:nt
        surf(x,y,U(:, :, i));
        %CABS y esas cosas
        axis([0 1 0 1 -.02 .2]);
        view([45 45]);
        %fLW
        %axis([-0.15 1 -1 2 -1 2])
        %view(3);
        pause(0.01)
    end
end
end

```

end

pricipal.m

El código del esquema de 6 puntos utiliza 2 algoritmos más que se agregan más adelante, estos algoritmos son “gammas.m” y “derivada.m”. Esta función aproxima, por medio del esquema de 6 puntos que se presenta en el capítulo 4 la ecuación de advección en una región descrita por las variables de entrada x, y .

```
% Esta es la version con mas precision hasta el momento del metodo de
% Lax-Wendroff programado de manera implicita con el esquema de 6 puntos.
%
% Es importante revisar que la matriz coincida con los limites de la malla.
%
% Esta version funciona calculando el metodo explicito primeramente para
% utilizar como semilla del metodo implicito.
% Cuenta con ambas versiones para calcular las gammas, la version donde se
% resuelve el sistema de ecuaciones y la version donde se tienen las
% formulas explicitas.
%
% La llamada al metodo se hace:
%
%       U = principal(x,y,f,a,b,t,graf);
%
% Argumentos de salida
%
%       U       :   Matriz U con la aproximacion final de la funcion.
%
% Argumentos de entrada
%
%       x, y     :   Coordenadas de los puntos de la malla
%       f        :   Funcion para la condicion inicial y de frontera.
%       a, b     :   Velocidades de movimiento del pulso de la funcion f.
%                   Los valores por defecto de a y b son .1
%       t        :   Vector con los diferentes niveles de tiempo.
%                   El valor por defecto de t es 0:1/1023:2;
%       graf     :   Se utiliza para elegir si se grafica o no.
%                   0   :   No se grafica (valor por defecto).
%                   1   :   Si se grafica.
%
% La documentacion de este codigo se escribio sin acentos.
%

function [U1,Gamma] = principal(x,y,f,a,b,t,graf)
tic
close all
```

```

switch nargin
    case 3
        a = .1;
        b = .1;
        t = 0:1/1023:2;
        graf = 0;
    case 4
        b = .1;
        t = 0:1/1024:1;
        graf = 0;
    case 5
        t = 0:1/1024:1;
        graf = 0;
    case 6
        graf = 0;
end

%Inicializamos variables.
[m,n] = size(x);
nt = length(t);
dt = t(2)-t(1);
U1 = zeros(m,n,nt);

%Llenamos con condicion inicial.
for i = 1:m
    for j = 1:n
        U1(i,j,1) = f([x(i,j),y(i,j)],t(1),a,b);
    end
end

%Llenamos con condicion de frontera.
for k = 2:nt
    for i = 1:m
        U1(i,1,k) = f([x(i,1),y(i,1)],t(k),a,b);
    end
    for j = 2:n
        U1(1,j,k) = f([x(1,j),y(1,j)],t(k),a,b);
    end
end

%Llamamos a la funcion gammas que nos regresa la matriz de Gammas para
%todos los nodos de la malla.
Gamma = gammas(x,y,a,b,dt);

%Llenamos la matriz

```

```

for k = 2:nt
%para todos los nodos interiores
    for i = 3:m
        for j = 3:n
            U1(i,j,k) = U1(i,j,k-1) + ...
                Gamma(i,j,1)*U1(i,j,k-1) + ...
                Gamma(i,j,2)*U1(i-1,j,k-1) + ...
                Gamma(i,j,3)*U1(i-2,j,k-1) + ...
                Gamma(i,j,4)*U1(i-1,j-1,k-1) + ...
                Gamma(i,j,5)*U1(i,j-2,k-1) + ...
                Gamma(i,j,6)*U1(i,j-1,k-1);
        end
    end
end

%Llenamos la matriz para todos los nodos de la segunda linea horizontal.

    for i = 2:m-1
        U1(i,2,k) = U1(i,2,k-1) + ...
            Gamma(i,2,1)*U1(i,2,k-1) + ...
            Gamma(i,2,2)*U1(i-1,3,k-1) + ...
            Gamma(i,2,3)*U1(i-1,2,k-1) + ...
            Gamma(i,2,4)*U1(i-1,1,k-1) + ...
            Gamma(i,2,5)*U1(i,1,k-1) + ...
            Gamma(i,2,6)*U1(i+1,1,k-1);
    end

    %Llenamos la matriz para todos los nodos de la segunda linea vertical.

    for j = 3:n-1
        U1(2,j,k) = U1(2,j,k-1) + ...
            Gamma(2,j,1)*U1(2,j,k-1) + ...
            Gamma(2,j,2)*U1(1,j+1,k-1) + ...
            Gamma(2,j,3)*U1(1,j,k-1) + ...
            Gamma(2,j,4)*U1(1,j-1,k-1) + ...
            Gamma(2,j,5)*U1(2,j-1,k-1) + ...
            Gamma(2,j,6)*U1(3,j-1,k-1);
    end

    %Llenamos la matriz para los nodos (2,n) y (m,2)

    U1(2,n,k) = U1(2,n,k-1) + ...
        Gamma(2,n,1)*U1(2,n,k-1) + ...
        Gamma(2,n,2)*U1(1,n,k-1) + ...
        Gamma(2,n,3)*U1(1,n-1,k-1) + ...
        Gamma(2,n,4)*U1(1,n-2,k-1) + ...

```

```

Gamma(2,n,5)*U1(2,n-1,k-1) + ...
Gamma(2,n,6)*U1(3,n,k-1);

U1(m,2,k) = U1(m,2,k-1) + ...
Gamma(m,2,1)*U1(m,2,k-1) + ...
Gamma(m,2,2)*U1(m,3,k-1) + ...
Gamma(m,2,3)*U1(m-1,3,k-1) + ...
Gamma(m,2,4)*U1(m-2,3,k-1) + ...
Gamma(m,2,5)*U1(m-1,2,k-1) + ...
Gamma(m,2,6)*U1(m,1,k-1);

end
toc

% Hacemos la grafica
if graf == 1
    for i=1:50:nt
        surf(x,y,U(:, :, i));
        %CABS y esas cosas
        axis([0 1 0 1 -.02 .2]);
        view([45 45]);
        %fLW
        %axis([-0.15 1 -1 2 -1 2])
        %view(3);
        pause(0.01)
    end
end
end

```

gammas.m

Esta función calcula los coeficientes gamma para el método de 6 puntos.

```

%
% Los comentarios en este codigo se escribieron sin acentos.
%

function Gamma = gammas(x,y,a,b,dt)
[m,n] = size(x);
Gamma = zeros(m,n,6);
D = -a*dt;
E = -b*dt;
A= (a*dt)^2/2.;
B = a*b*dt^2;
C = (b*dt)^2/2;

%Calculamos las Gammas de los nodos interiores

```

```

for i = 3:m
    for j = 3:n
        %Armamos el P que se le enviara a la rutina derivada
        %Primero armamos P(:,1) con las componentes en x
        p(1,1) = x(i,j);
        p(2,1) = x(i-1,j);
        p(3,1) = x(i-2,j);
        p(4,1) = x(i-1,j-1);
        p(5,1) = x(i,j-2);
        p(6,1) = x(i,j-1);

        %Ahora armamos P(:,2) con las componentes en y
        p(1,2) = y(i,j);
        p(2,2) = y(i-1,j);
        p(3,2) = y(i-2,j);
        p(4,2) = y(i-1,j-1);
        p(5,2) = y(i,j-2);
        p(6,2) = y(i,j-1);

        %Llamamos a la rutina derivada que nos regresara las gamas 1...6
        %para el nodo i,j
        Gamma(i,j,:) = derivada(p,A,B,C,D,E);
    end
end

%Calculamos Gammas para nodos de segunda linea horizontal
for i = 2:m-1
    %Armamos el P que se le enviara a la rutina derivada
    %Primero armamos P(:,1) con las componentes x
    p(1,1) = x(i,2);
    p(2,1) = x(i-1,3);
    p(3,1) = x(i-1,2);
    p(4,1) = x(i-1,1);
    p(5,1) = x(i,1);
    p(6,1) = x(i+1,1);

    %Primero armamos P(:,2) con las componentes y
    p(1,2) = y(i,2);
    p(2,2) = y(i-1,3);
    p(3,2) = y(i-1,2);
    p(4,2) = y(i-1,1);
    p(5,2) = y(i,1);
    p(6,2) = y(i+1,1);

```

```

        Gamma(i,2,:) = derivada(p,A,B,C,D,E);
    end

%Calculamos Gammas para los nodos de segunda fila vertical
for j = 3:n-1
    %Armamos el P que se le enviara a la rutina derivada
    %Primero armamos p(:,1) con las componentes x
    p(1,1) = x(2,j);
    p(2,1) = x(1,j+1);
    p(3,1) = x(1,j);
    p(4,1) = x(1,j-1);
    p(5,1) = x(2,j-1);
    p(6,1) = x(3,j-1);

    %Ahora rmamos p(:,2) con las componentes y
    p(1,2) = y(2,j);
    p(2,2) = y(1,j+1);
    p(3,2) = y(1,j);
    p(4,2) = y(1,j-1);
    p(5,2) = y(2,j-1);
    p(6,2) = y(3,j-1);

    Gamma(2,j,:) = derivada(p,A,B,C,D,E);
end

%Calculamos Gammas para los nodos (2,n) y (m,2)
%Armamos el P que se le enviara a la rutina derivada para (2,n)
p(1,1) = x(2,n);
p(2,1) = x(1,n);
p(3,1) = x(1,n-1);
p(4,1) = x(1,n-2);
p(5,1) = x(2,n-1);
p(6,1) = x(3,n);

p(1,2) = y(2,n);
p(2,2) = y(1,n);
p(3,2) = y(1,n-1);
p(4,2) = y(1,n-2);
p(5,2) = y(2,n-1);
p(6,2) = y(3,n);

Gamma(2,n,:) = derivada(p,A,B,C,D,E);

%Armamos el P que se le enviara a la rutina derivada para (m,2)
p(1,1) = x(m,2);
p(2,1) = x(m,3);

```

```
p(3,1) = x(m-1,3);
p(4,1) = x(m-2,3);
p(5,1) = x(m-1,2);
p(6,1) = x(m,1);

p(1,2) = y(m,2);
p(2,2) = y(m,3);
p(3,2) = y(m-1,3);
p(4,2) = y(m-2,3);
p(5,2) = y(m-1,2);
p(6,2) = y(m,1);

Gamma(m,2,:) = derivada(p,A,B,C,D,E);

end
```

derivada.m

Esta función ayuda a calcular los coeficientes gamma para el método de 6 puntos.

```
function [gammain] = derivada(p,A,B,C,D,E)

q = zeros(5,2);
M = zeros(5,6);
gammain = zeros(6,1);

for j = 2:6
    q(j-1,1) = p(j,1)-p(1,1);
    q(j-1,2) = p(j,2)-p(1,2);
end

for j = 1:5
    M(1,j) = q(j,1);
    M(2,j) = q(j,2);
    M(3,j) = q(j,1)^2;
    M(4,j) = q(j,1)*q(j,2);
    M(5,j) = q(j,2)^2;
end

M(1,6) = D;
M(2,6) = E;
M(3,6) = 2*A;
M(4,6) = B;
M(5,6) = 2*C;

dxmax = abs(q(1,1));
```



```
dymax = abs(q(1,2));

for i = 2:5
    dxmax = max(dxmax, abs(q(i,1)));
    dymax = max(dymax, abs(q(i,2)));
end

for i = 1:6
    M(1,i) = M(1,i)/dxmax;
    M(2,i) = M(2,i)/dymax;
    M(3,i) = M(3,i)/(dxmax*dxmax);
    M(4,i) = M(4,i)/(dxmax*dymax);
    M(5,i) = M(5,i)/(dymax*dymax);
end

gammain(2:6) = M(:,1:5)\M(:,6);

gammain(1) = 0;
for i = 2:6
    gammain(1) = gammain(1) - gammain(i);
end

end
```