



UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE CIENCIAS UNAM
DIVISION DE ESTUDIOS DE POSGRADO

UN SISTEMA PARA LA GENERACION
DE MALLAS ESTRUCTURADAS
ARMONICAS

T E S I S

QUE PARA OBTENER EL GRADO ACADEMICO DE
MAESTRO EN CIENCIAS
M A T E M A T I C A S
P R E S E N T A:
MÓNICA ZAIMA VÍQUEZ CANO

MEXICO, D.F.

2008



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Índice general

1. Conceptos Preliminares	1
1.1. Solución de ecuaciones diferenciales parciales	1
1.2. Discretización del dominio	1
1.3. Generación de mallas	3
1.4. Mallas estructuradas	5
1.4.1. Métodos para generar mallas estructuradas	5
2. Métodos Variacionales Continuos	11
2.1. Métodos Variacionales	11
2.1.1. Problema	11
2.1.2. Ejemplos	13
2.2. Funcional de Longitud	14
2.3. Funcional de Área	16
2.4. Funcional de Suavidad	19
2.5. Funcional de Ortogonalidad	24
2.6. Ejemplos	26
3. Métodos Variacionales Discretos	27
3.1. Formulación del Problema	27
3.2. El Mapeo Bilineal	31
3.3. Discretización General	44
3.3.1. Discretización de la Funcional de Longitud	54
3.3.2. Discretización de la Funcional de Área	56
3.3.3. Discretización de la Funcional de Ortogonalidad	58
3.3.4. Discretización de la Funcional de Suavidad	60
3.4. Ejemplos De Funcionales Clásicas Discretas	64

4. Funcionales Cuasi-Armónicas y Épsilon-Convexidad	67
4.1. Generación numérica de mallas	67
4.2. ϵ -Convexidad	68
4.2.1. Funcionales Cuasi-Armónicas	72
4.3. Nuevo Funcional de Área	76
5. Herramientas esenciales para generar mallas Armónicas y Cuasi-Armónicas de manera eficiente	79
5.1. El Método de Interpolación Transfinita	79
5.2. Combinaciones interesantes entre funcionales Nuevos y Clásicos .	81
5.2.1. Normalización de las funcionales	83
5.3. Método de Newton punto a punto	84
5.3.1. Optimización local	88
6. Diseño e implementación de un sistema para la generación de mallas estructuradas armónicas	91
6.1. Ciclo de vida de un sistema de información	91
6.2. Etapa A : Investigación Preliminar	94
6.3. Objetivos de la investigación	96
6.3.1. Objetivo General	96
6.3.2. Objetivos Específicos	96
6.4. Etapa B : Análisis de las actividades	96
6.4.1. Documentación descriptiva de los procesos	96
6.4.2. Clasificación de las actividades	97
6.4.3. Mapa de procesos	99
6.5. Etapa C : Identificación de los Requisitos	100
6.5.1. Rango de alcance del sistema	100
6.5.2. Requisitos Específicos	102
6.5.3. Requisitos de interfase externa	105
6.5.4. Requisitos no funcionales	106
6.6. Etapa D : Evaluación de los sistemas existentes	106
6.7. Etapa E : Identificación de las estrategias para cumplir los requisitos	109
6.8. Etapa F : Diseño del Sistema UNAMALLA para PC 3.0	109
6.9. Etapa G : Implementación del Sistema	113
6.10. Etapa H : Revisión posterior a la Implementación	120
7. Conclusiones	133

Prefacio

El interés fundamental de este trabajo, es actualizar un sistema para generación de mallas estructuradas sobre regiones planas irregulares, que requiera un mínimo de recursos del sistema de cómputo, en este caso del sistema MS-DOS¹. En este sistema, la actualización consiste en implantar algunos de las nuevas funcionales de área y Cuasi-Armónicas que son muy eficientes en regiones irregulares.

En el primer capítulo hacemos una introducción y una historia del problema de generación de mallas estructuradas sobre una región plana.

En el segundo capítulo, presentamos los métodos variacionales continuos que son el punto de partida para muchos de los métodos más recientes. La idea es elegir de todos los mapeos del cuadrado unitario en la región dada, aquel que optimice alguna propiedad geométrica y resolver la ecuación diferencial que caracteriza el mapeo óptimo.

En el tercer capítulo, desarrollamos la teoría básica de los métodos variacionales discretos, que se deducen siguiendo la teoría que Ivanenko aplicó a la funcional de suavidad o armónica, como la llamaremos de ahora en adelante. La idea fundamental consiste en discretizar la funcional en lugar de las ecuaciones de Euler-Lagrange, así en vez de resolver un sistema de ecuaciones diferenciales parciales, lo que haremos será resolver un problema de optimización de gran escala.

En el capítulo cuatro consideramos el problema de generar mallas armónicas minimizando ciertas funcionales que hacen sencilla la implementación del algoritmo de minimización y que además producen resultados adecuados. Además se introduce el concepto de ϵ -convexidad, que permite eliminar la inestabilidad numérica que se presenta al generar mallas convexas con la condición de que el área de los triángulos sea positiva para todos los triángulos de la malla.

¹MS-DOS son las siglas de MicroSoft Disk Operating System, Sistema operativo de disco de Microsoft. Es un sistema operativo comercializado por Microsoft perteneciente a la familia DOS. Fue un sistema operativo para el IBM PC que alcanzó gran difusión

En el capítulo cinco explicamos las herramientas que se usan para obtener mallas estructuradas armónicas de manera eficiente.

Primero damos las fórmulas de la interpolación transfinita, que se usan para construir una malla inicial, luego se explica como para combinar funcionales es necesario normalizarlas y como se lleva a cabo esta normalización, al final se explica como se utiliza el método de Newton punto a punto para realizar la optimización de un problema de gran escala.

En el capítulo seis, se presenta la descripción del sistema.

Agradecimientos

A Dios por darme una segunda oportunidad de vida a través de las manos del Dr. Miguel Ángel López Escutia y el Dr. Esteban Maldonado Torres con su trabajo tan acertado y por permitirles formar un gran equipo y que sin ello no hubiera concluido este trabajo.

A la UNAM por ser mi alma Matter. En especial a la Facultad de Ciencias.

A la Universidad Anáhuac del Sur, a los Legionarios de Cristo, al Movimiento Regnum Christi y a Familia Misionera por darme una formación humanística, pedagógica e integral a lo largo de estos 7 años y la cual me esfuerzo por proyectar día a día en mi labor cotidiana y que he tratado de reflejar en este trabajo.

A mis hermanos Salvador y Alejandro por ser mi ejemplo a seguir con sus logros y por cuidarme cuando más lo he necesitado.

Al Dr. Pablo Barrera Sánchez por compartir cada conocimiento, por que al cabo de estos años ha conducido de manera acertada mi educación en el campo del Cómputo Científico, por enseñarme a mirar por esa ventana del conocimiento y por motivar a cada uno de sus estudiantes en la superación cotidiana, para ser líderes en ciencia y como alguna vez me dijo pocos tienen la satisfacción de alcanzar.

Al Dr. Fernando Brambila Paz por haberme apoyado a ingresar a la facultad de Ciencias al aceptar ser mi tutor y ayudarme a conseguir la beca para mis estudios de posgrado, por ser un gran ser humano que ayuda sin mirar a quien.

A cada uno de los sinodales por haber aceptado revisar este trabajo, por que con sus sugerencias se enriqueció el mismo.

Dr. Humberto Madrid de la Vega.

Dr. Pedro González-Casanova Henríquez.

Dr. José Gerardo Tinoco Ruiz.

Dr. Francisco Javier Domínguez Mota.

M en C. María Lourdes Velasco ArreguÍ.

M en C. Guilmer Ferdinand González Flores.

Dedicatorias

Yo he oído de ti...que en ti se halló luz, entendimiento y mayor sabiduría..., Yo, pues, he oído de ti que puedes dar interpretaciones y resolver problemas...serás vestido de púrpura y un collar de oro llevarás en el cuello... -Daniel 5:15

A la memoria de mis padres Tere y Poli por su amor infinito y por que este era un sueño que a ellos les hubiera gustado mucho compartir conmigo y que al final he podido alcanzar. Por que a lo largo de mi vida la caja de cristal permanece intacta a pesar de las caídas que me sirvieron para fortalecer el espíritu y resultaron ser excelentes enseñanzas de vida.

A los mas pequeños de la familia con mucho cariño a Omar, la Tite, Anne y a los miembros que vendrán de la familia Víquez Cano.

A Paty Cano, Argelia Castillo por los años de amistad.

A mis amigos que han compartido la idea de amar a los seres humanos y nos hemos preocupado por hacer un mundo mejor al ser honestos en cada proyecto que realizamos, por dar una mano de ayuda a quienes mas lo necesitan a través de una sonrisa y un buen consejo, de los cuales aprendí mucho:

Esperanza Méndez, Concepción González Enríquez, Joaquín Rodrigo Careaga Perkins, Aldo López Meléndez, Virgilio Girón Guzmán, Sandra Aviña Plata, Beatriz Carriedo, Mónica Pacheco, Sandra Hidalgo, Tere Arreola, Katia Toxtli, Edith Castellanos Ramírez, Bernardo Ramírez Martínez, Sara Camacho Cancino, Tere Carrillo, Mine, Manuel Valadez Rodríguez y por supuesto a Mike Escutia por que los seres humanos no se conocen por casualidad!

A los Compañeros del grupo UNAMALLA y con los cuales tuve la oportunidad de compartir el laboratorio de cómputo científico:

A los de la vieja generación Irma, Guilmer, Francisco Domínguez, Luis, Mari, Claudia Alquicira, Mariano, Vicky Mederos y los que se me olvidan por la ayuda mutua durante nuestros estudios y por haber hecho un ambiente de trabajo agradable y fantástico.

A los de la nueva generación Cesar (el loco), Francisco (el chamaco), Javier de Jesús, Luis Carlos (el sub comandante Lu), José Luis (el colado), Gustavo y los que solo van de vez en vez, por sus ocurrencias y sus sonrisas; pero sobre todo por compartir la inmadurez de los 20's.

Se me olvidaba a Javier Aurrecochea por ser el encargado del laboratorio de cómputo y esforzarse por cuidar los arbolitos con el proyecto de reciclaje de papel.

Al personal del departamento de matemáticas de la facultad de ciencias investigadores, secretarias y alumnos que he tenido la fortuna de conocer a lo largo de estos años, a los que aun están, en especial a:

Mirna, Hermila, Lucy Gasca, Rafael Rojas Barbachano, Luis Briseño Aguirre, Juan Rico Arvizú, Marcos Zepeda y a los que se han marchado.

A los Doctores Longina Castellanos Noda, Ángel Albeo Pérez Domínguez, Hugo Alberto Altamirano Corzo y a la Profesora María de los Ángeles Barrera Sánchez por sus platicas en estos últimos meses siempre tan acertadas, pero sobre todo por su amistad.

A cada uno de mis alumnos de la FES - Acatlán, de la Universidad Anáhuac del Sur y de la Facultad de Ciencias por que con cada una de sus preguntas me motivaron a seguir aprendiendo, por ayudarme a encontrar esta gran misión en mi vida, la de la enseñanza! Y por permitirme transmitirles la luz del conocimiento.

Lista de figuras

- Figura 1.1 Discretización de una región Trivial.
- Figura 1.2 Discretización de una región Complicada.
- Figura 1.3 Mallas generadas utilizando cuadriláteros.
- Figura 1.4 Mallas generadas utilizando triángulos.
- Figura 1.5 Mallas generadas con TFI.
- Figura 1.6 Mallas generadas con TFI.
- Figura 1.7 Malla generada utilizando la ecuación de Laplace.
- Figura 1.8 Malla generada con el funcional de suavidad, con el generador TTM.
- Figura 1.9 Malla generada con el funcional de área+longitud +ortogonalidad utilizando el generador TTM.
- Figura 1.10 Instituciones que participan en el grupo UNAMALLA.
- Figura 2.1 Mapeo de la frontera B_2 a la frontera Ω .
- Figura 2.2 Ejemplos de Mallas sobre una misma región.
- Figura 2.3 Correspondencia entre las fronteras del cuadrado y la región.
- Figura 2.4 Mapeo inverso de Ω a B_2 .
- Figura 2.5 Funcional combinado Área con longitud.
- Figura 2.6 Funcional de longitud.
- Figura 2.7 Funcional de suavidad.
- Figura 3.1 Mapeo directo entre la región B_2 y Ω .
- Figura 3.2 Construcción del homeomorfismo ϕ por medio de una correspondencia de las fronteras del cuadrado y Ω .
- Figura 3.3 Celda B_{ij} .
- Figura 3.4 Representación del mapeo bilineal.
- Figura 3.5 Lado PS del cuadrilátero PQRS.
- Figura 3.6 Lado QR del cuadrilátero PQRS.
- Figura 3.7 Segmento $\bar{x}_l(t), \bar{x}_r(t)$.
- Figura 3.8 Lado SR del cuadrilátero PQRS.
- Figura 3.9 Lado PQ del cuadrilátero PQRS.
- Figura 3.10 Segmento $\bar{x}_b(t), \bar{x}_\tau(t)$.
- Figura 3.11 Mapeo Bilineal.
- Figura 3.12 Orientación negativa de un triángulo.

- Figura 3.14 Área Positiva del triángulo que tiene los vértices A,B,C.
Figura 3.15 Área negativa del triángulo que tiene los vértices A,B,C.
Figura 3.16 Triángulo A,B,C que muestra la definición área orientada.
Figura 3.17 Triángulos definidos sobre un cuadrilátero a partir de sus vértices.
Figura 3.18 Ejemplo de triángulos orientados.
Figura 3.19 Triángulo SPQ.
Figura 3.20 Triángulo PQR.
Figura 3.21 Triángulo RSP.
Figura 3.22 Triángulo QRS.
Figura 3.23 Mapeo de la celda B_{ij} a la celda C_{ij} .
Figura 3.24 Puntos de la frontera de la región.
Figura 3.25 Segmentos de la frontera de la región Ω .
Figura 3.26 Mapeo de la celda B_{ij} a B_2 y aplicación del mapeo $\bar{r}(s, t)$.
Figura 3.27 Mapeo a_{ij} .
Figura 3.28 Triángulo Δ_q .
Figura 3.29 Mallas óptimas generadas con la funcional de área discreta.
Figura 3.30 Mallas óptimas generadas con la funcional de Suavidad discreta.
Figura 4.1 Gráfica de la función $\varphi_w(\alpha)$.
Figura 5.1 Segmentos de frontera de $\partial\Omega$.
Figura 5.2 Funcionales con orden de magnitud distinta.
Figura 5.3 Funcionales con orden de magnitud igual.
Figura 5.4 Malla de 3×3 .
Figura 5.5 Malla de 3×3 con nodos vecinos del punto P_{ij} .
Figura 5.6 Extracción de una submalla de 3×3 .
Figura 5.7 Inserción de una submalla de 3×3 .
Figura 5.8 Triángulos asociados al punto P_{ij} .
Figura 6.1 Etapas para el diseño e implementación del sistema.
Figura 6.2 Modelo conceptual para generar mallas estructuradas armónicas.
Figura 6.3 Mapa de procesos para el diseño del sistema UNAMALLA 3.0.
Figura 6.4 Alcances del sistema UNAMALLA 3.0.
Figura 6.5 Alcances y perspectivas del sistema UNAMALLA 3.0.
Figura 6.6 Mallas generadas con las funcionales k-Suavidad y área Clásica.
Figura 6.7 Mallas generadas con las funcionales k-area y Longitud.
Figura 6.8 Diagrama de bloques del módulo menú principal.
Figura 6.9 Diagrama de bloques del módulo de contornos.
Figura 6.10 Diagrama de bloques del módulo mallas.
Figura 6.11 Diagrama de bloques del módulo optimización.
Figura 6.12 Continuación del diagrama de bloques del módulo optimización.
Figura 6.13 Diagrama de bloques del sistema UNAMALLA 3.0.
Figura 6.14 Menú principal sistema UNAMALLA 3.0.
Figura 6.15 Submenú segmentos de frontera.
Figura 6.16 Opción salvar contorno en el sistema UNAMALLA 3.0.
Figura 6.17 Definición de las dimensiones il de la malla.
Figura 6.18 Definición de las dimensiones jl de la malla.
Figura 6.19 Menú principal sistema UNAMALLA 3.0.
Figura 6.20 Menú malla.

- Figura 6.21 Opción leer malla.
- Figura 6.22 Opción guardar malla.
- Figura 6.23 Opción optimizar malla.
- Figura 6.24 Opción asignar valor de alfa.
- Figura 6.25 Submenú elige funcional.
- Figura 6.26 Submenú elige método de optimización.
- Figura 6.27 Ejemplo funcional S3-Long. Sobre región Sudamérica, 30x30 por lado.
- Figura 6.28 Ejemplo funcional S3-Long. Sobre región Habana, 40x40 por lado.
- Figura 6.29 Ejemplo funcional S3-Long. Sobre región Cisne, 40x40 por lado.
- Figura 6.30 Ejemplo funcional S3-Ort. Sobre región Sudamérica, 30x30 por lado.
- Figura 6.31 Ejemplo funcional S3-Ort. Sobre región Habana, 40x40 por lado.
- Figura 6.32 Ejemplo funcional S3-Ort . Sobre región Cisne, 40x40 por lado.
- Figura 6.33 Ejemplo funcional S3-area-Ort. Sobre región Sudamérica,
30x30 por lado.
- Figura 6.34 Ejemplo funcional S3-area-Ort. Sobre región Habana, 40x40 por lado.
- Figura 6.35 Ejemplo funcional S3-area- Ort . Sobre región Cisne, 40x40 por lado.
- Figura 6.36 Ejemplo funcional Hws-area. Sobre región Sudamérica, 30x30 por lado.
- Figura 6.37 Ejemplo funcional Hws-area. Sobre región Habana, 40x40 por lado.
- Figura 6.38 Ejemplo funcional . Hws-area. Sobre región Cisne, 40x40 por lado.
- Figura 6.39 Ejemplo funcional Hws -Ort. Sobre región Sudamérica, 30x30 por lado.
- Figura 6.40 Ejemplo funcional Hws -Ort. Sobre región Habana, 40x40 por lado.
- Figura 6.41 Ejemplo funcional Hws - Ort . Sobre región Cisne, 40x40 por lado.
- Figura 6.42 Ejemplo funcional Hws -AO. Sobre región Sudamérica, 30x30 por lado.
- Figura 6.43 Ejemplo funcional Hws -AO. Sobre región Habana, 40x40 por lado.
- Figura 6.44 Ejemplo funcional Hws - AO. Sobre región Cisne, 40x40 por lado.

Capítulo 1

Conceptos Preliminares

”αυτη δε εστιν η αιωνιος ζωη ινα γνωσκωσιν σε του μονου αληθινου θεου και ου απεστειλας ιησουν χριστου”

KATA ΙΩΑΝΝΗΝ 17,3

1.1. Solución de ecuaciones diferenciales parciales

La modelación matemática de problemas de la física y la ingeniería da lugar a ecuaciones diferenciales parciales sobre una región en el plano o en el espacio (región física) donde el modelo está definido.

El problema de resolver numéricamente ecuaciones diferenciales parciales surge del hecho de que las ecuaciones diferenciales parciales sólo se pueden resolver de manera analítica para casos muy simples, por lo que es necesario desarrollar métodos numéricos que proporcionen buenas aproximaciones y sean lo más eficientes posible.

La solución de sistemas de ecuaciones diferenciales parciales por lo general requiere de la discretización del dominio en un conjunto de celdas en las cuales se realizan aproximaciones que transforman la ecuación diferencial en un sistema algebraico de ecuaciones el cual se resuelve numéricamente[30, 1, 39, 47].

1.2. Discretización del dominio

La discretización de la región de interés consiste en dividir la región en pequeñas regiones elementales, llamadas celdas, estas celdas son triángulos, cuadriláteros, hexágonos, etc. En el caso de regiones en el espacio son tetraedros, hexaedros, prismas, etc. En este trabajo solo estamos interesados en regiones planas, esto es dividir regiones planas en celdas, es decir, si Ω es la región de interés entonces

$$\Omega = \bigcup_{i=1}^N \Omega_i$$

donde N es el número de elementos en que se divide Ω .

Este problema puede ser trivial, como sería el caso de dividir el cuadrado unitario en rectángulos o muy difícil si la región es muy irregular como la superficie de la república mexicana.

Figura1.1 Discretización de una región Trivial.

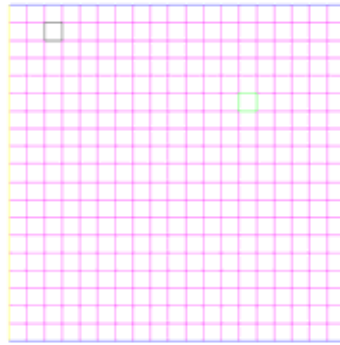
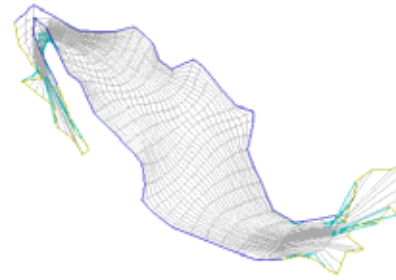


Figura1.2 Discretización de una región Complicada.



En los últimos 30 años se ha empezado a desarrollar software que construye estas divisiones y el área que aborda esta temática se conoce como generación de mallas, ya que a la región con subdivisiones en celdas se le llama una malla de la región[30, 47].

1.3. Generación de mallas

Una región se puede mallar de varias formas: en triángulos ó rectángulos ó cuadrados.

Ejemplos

Cuadriláteros

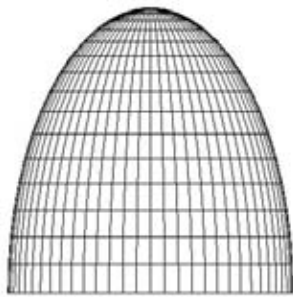


Fig. tomada de

[http : //www.scielo.cl/fbpe/img/in.fotec/v17n3/fig16 - 1.jpg](http://www.scielo.cl/fbpe/img/in.fotec/v17n3/fig16-1.jpg)

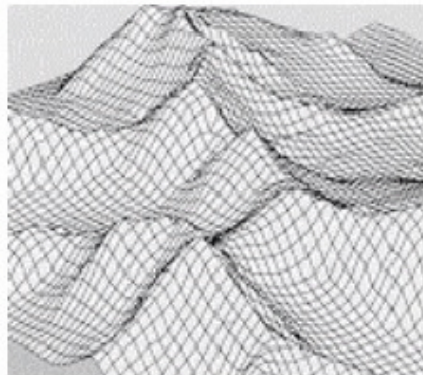


Fig. tomada de

[http : //mapserver.inegi.gob.mx/geografia/espanol/normatividad/mde/mde01.jpg](http://mapserver.inegi.gob.mx/geografia/espanol/normatividad/mde/mde01.jpg)

Figura 1.3 Mallas generadas utilizando cuadriláteros.

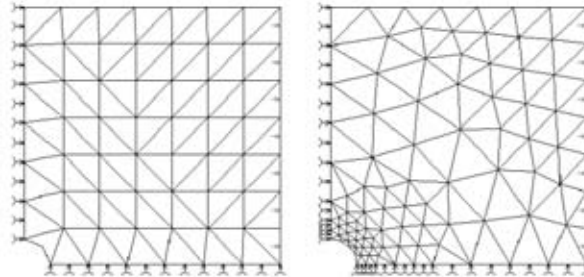
Triángulos:

Fig. tomada de

www.research.ibm.com/.../srf_fit/sf_prb.gifejemplos

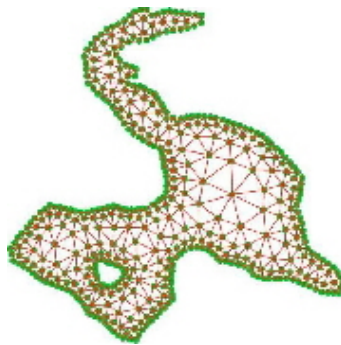


Fig. tomada de

www.uoguelph.ca/~mgarvie/mesh_temp.jpg

Figura 1.4 Mallas generadas utilizando triángulos.

Dos métodos numéricos fundamentales en la solución de ecuaciones diferenciales parciales son los de diferencias finitas y el método de elemento finito. Algunas personas prefieren el método de diferencias finitas porque es muy fácil numerar los nodos y las celdas y los sistemas algebraicos resultantes tienen matrices con estructuras de tipo banda por bloques y otras propiedades que resultan muy útiles en la solución numérica.

El problema que presenta la solución usando diferencias finitas es que no se puede aplicar fácilmente en regiones irregulares. Sin embargo desde hace aproximadamente medio siglo se propuso usar cambio de variables y transformar un problema sobre una región arbitraria a un problema sobre el cuadrado unitario. Para poder aplicar este método, es necesario construir una transformación suave y biyectiva entre la región de interés y el cuadrado unitario. Este es el problema que abordaremos en los capítulos siguientes y se conoce con el nombre de generación numérica de mallas estructuradas sobre regiones planas irregulares.

Otras personas prefieren el Método del elemento finito porque se puede aplicar en mallas que no tienen que ser estructuradas.

1.4. Mallas estructuradas

Llamamos mallas estructuradas en una región a una transformación del cuadrado unitario sobre la región. Las regiones estandar son usualmente bolas unitarias bajo alguna norma. Nosotros consideraremos como región estandar al conjunto $B_2 = [0, 1] \times [0, 1]$. Pero de aquí en adelante usaremos el término bola unitaria.

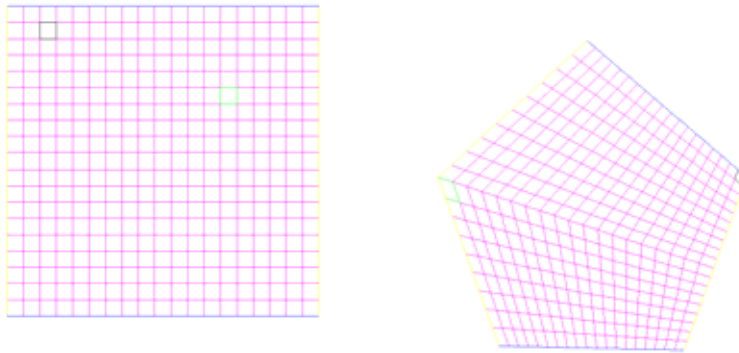
1.4.1. Métodos para generar mallas estructuradas

Métodos Algebraicos

Se basan en la transformación de coordenadas de un dominio físico; dada su naturaleza algebraica se han aprovechado sus características de cálculo rápido con el que se generan en comparación con los métodos que utilizan ecuaciones diferenciales parciales; sin embargo este tipo de métodos no garantizan suavidad de la malla. Entre los métodos más utilizados en la generación de mallas algebraicas se puede mencionar al método de interpolación Lagrange, la interpolación de Hermite, Splines o bien la interpolación Transfinita (TFI por sus siglas del inglés)[39, 47].

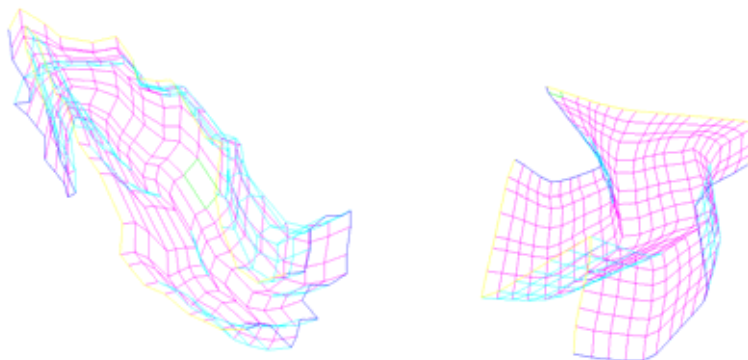
A continuación se presentan algunas mallas generadas por medio de interpolación transfinita (TFI).

Figura 1.5 Mallas generadas con TFI.



Regiones simples.

Figura 1.6 Mallas generadas con TFI.



Regiones Complicadas.

Métodos que utilizan Ecuaciones Diferenciales.

Métodos Elípticos Los métodos elípticos se basan en la solución de ecuaciones diferenciales parciales elípticas con algunas condiciones sobre ellas que son los llamados términos fuerza (forcing terms). Este problema es formulado por medio de un conjunto de ecuaciones de Poisson (ver Thompson 1977). La solución del sistema puede ser obtenida por medio de un método iterativo, por ejemplo el método de sobrerelajaciones (SOR) o bien algún otro método iterativo. Los sistemas elípticos producen mallas muy suaves, además pueden ser utilizadas para suavizar mallas con picos.

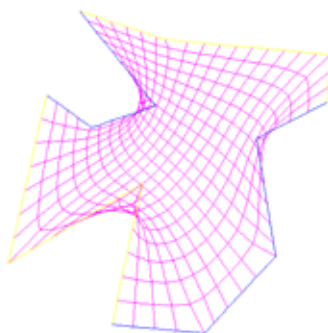


Figura 1.7 Malla generada utilizando la ecuación de Laplace.

Métodos Hiperbólicos Los métodos hiperbólicos se basan en la solución de ecuaciones diferenciales del tipo hiperbólico, estas son resueltas hacia afuera

de la frontera del dominio.

La idea de utilizar ecuaciones diferenciales parciales hiperbólicas es muy efectiva para flujos externos donde se tiene paredes en las fronteras.

La formulación básica de mallas hiperbólicas sienta su base en condiciones sobre la ortogonalidad y el área de las celdas. Estos métodos operan suavizando líneas en la malla y chequeando la ortogonalidad de las celdas.

Métodos Ortogonales Tinoco[46] define a las mallas ortogonales como:

Las mallas que tienen la propiedad de que sus líneas coordenadas son ortogonales (ó casi ortogonales) son preferidas en la solución numérica de ecuaciones diferenciales parciales, debido a que con tales mallas el error de truncamiento se ve reducido. También tienen la ventaja de que al transformar la ecuación a resolver, aparecen menos términos que para una malla no ortogonal y que las condiciones de frontera se pueden representar más fácilmente. Una desventaja es que requieren de una parametrización muy particular en la frontera. En la práctica, el encontrar una malla ortogonal para una región dada puede ser extremadamente difícil. Los sistemas ortogonales no necesitan ser conformes, y pueden ser generados tanto por sistemas hiperbólicos como elípticos.

Métodos Variacionales De entre los métodos que se han desarrollado en esta área estamos interesados en los que se obtienen por medio de una formulación variacional de las propiedades geométricas de la malla deseada. A esta rama se le conoce como Generación Variacional de Mallas. [13, 23, 24, 50]

Dentro de esta formulación variacional se puede hacer una gran subdivisión entre los métodos continuos y los discretos[15, 16, 2, 4, 5].

Métodos Variacionales Continuos Los métodos variacionales continuos para la generación de mallas fueron introducidos por Winslow en 1967 [50]. Brackbill y Saltzman 1982 [13] retoman el trabajo de Winslow y proponen un método adaptivo de zona, con el propósito de tratar de obtener una malla más suave. Más tarde Roche y Steinberg¹ hacen una formulación novedosa[42].

Métodos Variacionales Discretos Castillo² [16] propone construir mallas optimizando funciones que midan las propiedades geométricas de las celdas, a estas funciones se les llama las funcionales y solo dependen de los vértices de las celdas de la malla, así si uno desea una malla de $m \times n$ entonces la función solo depende de los puntos internos de la malla $(m-2)(n-2)$ y entonces el problema que se tiene es el de encontrar el mínimo de una función $2(m-2)(n-2)$ variables, para lo cual se usa un método de optimización de gran escala, como el gradiente conjugado no lineal o memoria limitada de Nocedal.[38]

¹<http://www.math.unm.edu/~stanly/>

²<http://www-rohan.sdsu.edu/~castillo/>

Castillo [16] introduce las funcionales³ de longitud área y de ortogonalidad y usando combinaciones lineales de ellas puede generar mallas sobre regiones no muy complicadas, sobre regiones irregulares es muy difícil encontrar una combinación que produzca mallas no dobladas⁴.

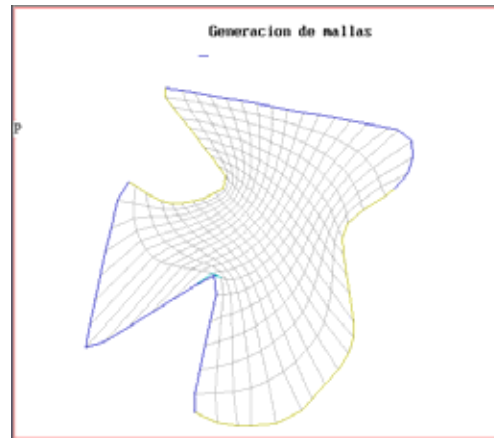


Figura 1.8 Malla generada con el funcional de suavidad, con el generador TTM.

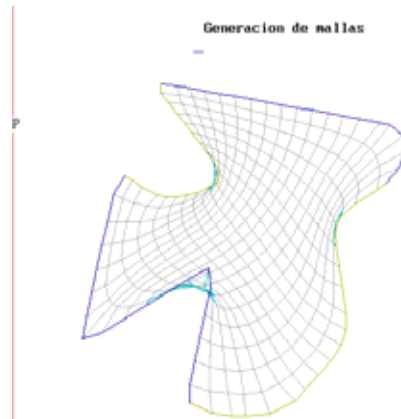


Figura 1.9 Malla generada con el funcional de área+longitud+ortogonalidad utilizando el generador TTM.

³Una funcional $I(x)$ es una función definida sobre los mapeos x que pertenecen a un subconjunto de un espacio de funciones.

⁴Celdas no convexas

El grupo UNAMALLA



Figura 1.10 Instituciones que participan en el grupo UNAMALLA.

En la facultad de ciencias de la UNAM⁵, el grupo de trabajo UNAMALLA ha desarrollado una línea de investigación sobre generación de mallas estructuradas desde finales de los años 80's, con la colaboración del Instituto de Cibernética, Matemática y Física (ICIMAF) de Cuba⁶, la Universidad Michoacana de San Nicolas de Hidalgo (U.M.S.N.H)⁷ y La Universidad Autónoma de Coahuila (UAdeC)⁸. Uno de los principales frutos obtenido de estas colaboraciones es el sistema de cómputo UNAMALLA.

Esta área de trabajo de generación de mallas ha resultado muy fértil. Se han obtenido buenos resultados en la creación de nuevos funcionales (ver Barrera⁹ [2][10][9], Tinoco¹⁰[46], Domínguez-Mota[22]¹¹) para la generación de mallas

⁵<http://www.fciencias.unam.mx/>

⁶<http://www.icmf.inf.cu/>

⁷<http://www.fismat.umich.mx/web/>

⁸<http://www.mate.uadec.mx/>

⁹<http://www.matematicas.unam.mx/pablo/>

¹⁰<http://garota.fismat.umich.mx/~jtinoco/>

¹¹<http://fismat.umich.mx/~dmota/>

y así resolver el problema de celdas dobladas que se presentan en regiones irregulares.

También se le ha dado seguimiento a la implementación de métodos eficientes de optimización a gran escala. Para resolver el problema de generación de mallas, se utiliza un método de optimización (ver García ¹²[23], Castellanos[14]¹³), donde se inicia el proceso con una malla inicial. Entre los métodos que se han implementado para este desarrollo en las diferentes versiones del sistema, están los métodos llamados de gran escala¹⁴ como son el método de Gradiente conjugado no lineal, L-BFGS, Newton -Truncado, Newton punto a punto (ver Nocedal[38]), TRON¹⁵ (Ver More[40]).

Por último la parte fundamental es el ensamblaje de todos estos elementos en el sistema de cómputo UNAMALLA en sus diferentes plataformas (ver González¹⁶ [24, 25], De la Cruz [21], Ojeda[41]¹⁷, García [23], Barrera[2, 4, 5, 6], Castellanos[14]). Una de las principales bondades del sistema es que es capaz de generar mallas en regiones muy irregulares con excelentes resultados.

El sistema puede ser obtenido de la página del grupo¹⁸, además podemos obtener información adicional del grupo de trabajo y publicaciones relacionadas con el sistema.

El sistema ha sido utilizado para obtener mallas óptimas sobre las cuales se resuelven algunos problemas modelo de ecuaciones diferenciales parciales (Ver Vázquez [49], De la Cruz[21], Chávez).

¹²<http://www.cima.uadec.mx/index.php?seccion=igarcia>

¹³http://www.icmf.inf.cu/grupos3_matematica.htm

¹⁴Estos métodos son excelentes en problemas que tienen un gran número de variables y cuando se tienen pocos recursos además de aprovechar la estructura generada por la función que se desea resolver.

¹⁵Este es un método de región de confianza que se utiliza en problemas de gran escala, utiliza un método de proyección, para generar un paso de Cauchy, además de un método de gradiente conjugado preconditionado, con una factorización incompleta de Cholesky para generar la dirección, esto hace que sea un método eficaz que requiere de pocas iteraciones.

¹⁶<http://www.matematicas.unam.mx/gfgf/>

¹⁷http://www.cima.uadec.mx/index.php?seccion=rina_ojeda

¹⁸<http://www.mathmoo.unam.mx/unamalla/downloads.html>

Capítulo 2

Métodos Variacionales Continuos

El problema de generación de mallas estructuradas se puede plantear como el problema de construir transformaciones suaves y biyectivas del cuadrado unitario en una región dada. En este capítulo presentaremos los métodos variacionales continuos que son el punto de partida para muchos de los métodos más recientes. La idea de estos métodos es elegir, de todos los mapeos del cuadrado unitario en la región dada, aquel que optimice alguna propiedad geométrica y resolver la ecuación diferencial que caracteriza el mapeo óptimo. La función a optimizar se le llama funcional debido a que su dominio de definición es un espacio de funciones y así presentaremos las funcionales de longitud, área y suavidad además de algunos ejemplos.

2.1. Métodos Variacionales

Dada una región Ω en el plano estamos interesados en encontrar soluciones al siguiente problema.

2.1.1. Problema

Encontrar un difeomorfismo de la bola unitaria B_2 sobre Ω que mapee la frontera de B_2 a la frontera de Ω ; es decir:

Encontrar

$$\bar{x} : B_2 \rightarrow \Omega$$

tal que

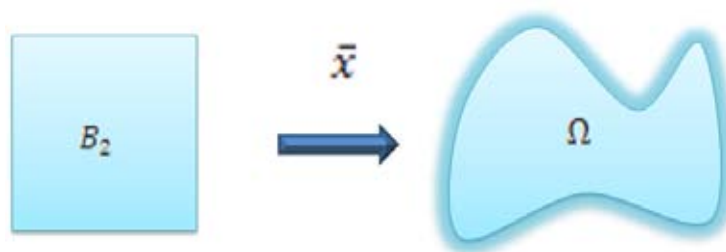


Figura 2.1 Mapeo de la frontera B_2 a la frontera Ω .

1. $\bar{x}(\partial B_2) = \partial \Omega$.
2. \bar{x} es biyectiva.
3. \bar{x} es C^1 .

En general este problema puede tener muchas soluciones.

2.1.2. Ejemplos

A continuación mostramos diferentes mallas sobre una misma región.

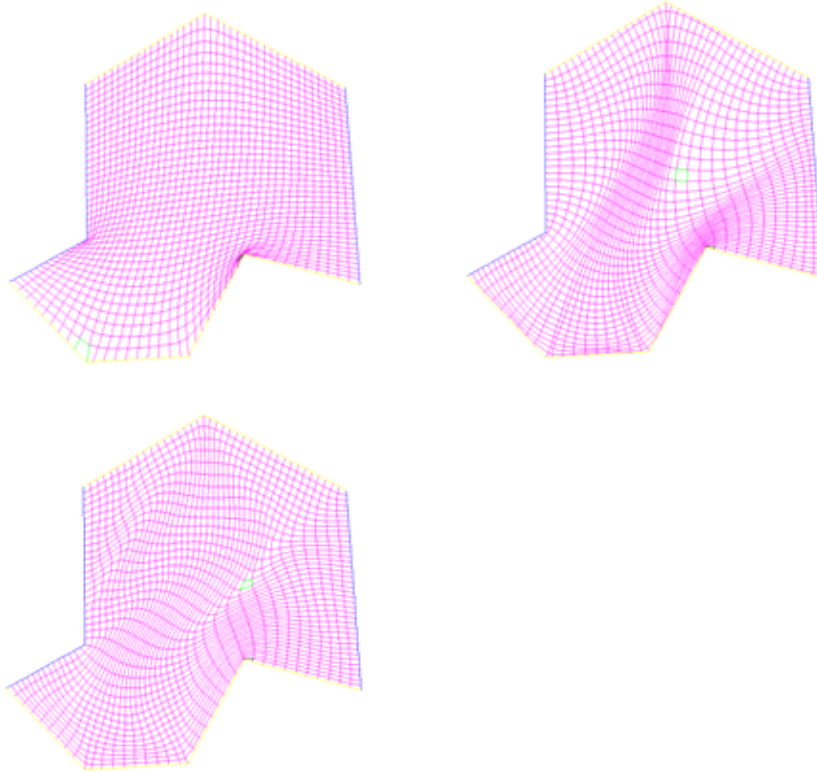


Figura 2.2 Ejemplos de Mallas sobre una misma región.

Por lo general necesitamos algún criterio para elegir una solución. Lo más usual en este caso es construir una medida de alguna propiedad geométrica de la malla que nos interese y obtener una solución que optimice esta medida sobre la región. Así que usualmente las medidas son de la forma:

$$I[\bar{x}] = \iint_{B_2} f(\bar{x}_\xi, \bar{x}_\eta) d\xi d\eta$$

donde

$$\bar{x}_\xi = \begin{pmatrix} x_\xi(\xi, \eta) \\ y_\xi(\xi, \eta) \end{pmatrix}, \quad \bar{x}_\eta = \begin{pmatrix} x_\eta(\xi, \eta) \\ y_\eta(\xi, \eta) \end{pmatrix}$$

y f en C^1 .

De lo anterior concluimos la siguiente definición

Definición 1 *Un método variacional continuo para la generación de mallas estructuradas sobre Ω es uno que busca una solución del problema variacional siguiente:*

$$\min_{\bar{x}} \iint_{B_2} f(\bar{x}_\xi, \bar{x}_\eta) d\xi d\eta$$

donde

- $\bar{x} : B_2 \rightarrow \Omega$.

tal que

- $\bar{x}(\partial B_2) = \partial\Omega$

f es una función dada de clase C^1 .

Las ecuaciones de Euler-Lagrange del problema anterior son:

$$\begin{aligned} \frac{\partial f}{\partial x} - \frac{d}{d\xi} \frac{\partial f}{\partial x_\xi} - \frac{d}{d\eta} \frac{\partial f}{\partial x_\eta} &= 0 \\ \frac{\partial f}{\partial y} - \frac{d}{d\xi} \frac{\partial f}{\partial y_\xi} - \frac{d}{d\eta} \frac{\partial f}{\partial y_\eta} &= 0 \end{aligned}$$

2.2. Funcional de Longitud

Uno de los primeros intereses es que la malla sea suave y para esto se propuso inicialmente controlar la longitud de los segmentos verticales y horizontales de la malla por lo que el funcional de longitud se define como

$$\begin{aligned} I_l(\bar{x}) &= \iint_{B_2} (\|\bar{x}_\xi\|^2 + \|x_\eta\|^2) d\xi d\eta \\ &= \iint_{B_2} (x_\xi^2 + y_\xi^2 + x_\eta^2 + y_\eta^2) d\xi d\eta \end{aligned}$$

entonces si deseamos encontrar \bar{x} tal que:

$$\min_{\bar{x}} I_l(\bar{x})$$

sujeto a

$$\bar{x}(\partial B_2) = \partial\Omega$$

Usualmente esta última condición se elige también y lo más común es indicando como se van a corresponder las fronteras.

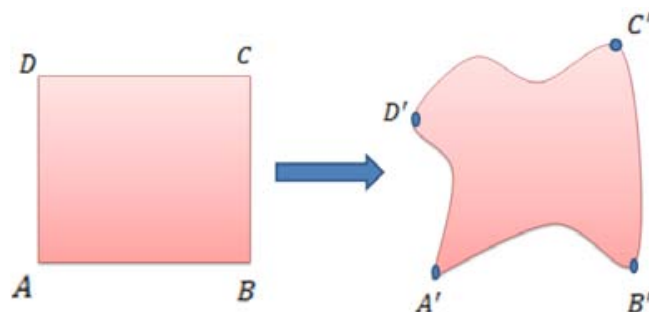


Figura 2.3 Correspondencia entre las fronteras del cuadrado y la región.

Las ecuaciones de Euler-Lagrange de este problema son:

$$x_{\xi\xi}^2 + x_{\eta\eta}^2 = 0$$

$$y_{\xi\xi}^2 + y_{\eta\eta}^2 = 0$$

$$\bar{x}(\partial B_2) = \partial\Omega$$

Como podemos observar las ecuaciones de Euler Lagrange son las ecuaciones de Laplace para las componentes $\bar{x} = (x, y)$

$$\nabla x^2 = 0 \quad y \quad \nabla y^2 = 0$$

$$x(\partial B_2) = x(\partial\Omega); \quad y \quad y(\partial B_2) = y(\partial\Omega)$$

Por lo que el problema se puede escribir como

$$\begin{aligned}\nabla \bar{x}^2 &= 0 \\ \bar{x}(\partial B_2) &= \partial \Omega\end{aligned}$$

Por lo que se obtienen problemas que son muy fáciles de resolver numéricamente. Sin embargo las soluciones de este problema no son uno a uno en regiones no convexas, por lo que para usar este funcional en la práctica, es necesario combinarlo con algún otro funcional como el de área.

La solución de \bar{x} nos proporciona un mapeo suave, por tal motivo en la literatura se considera como una funcional de suavidad.

2.3. Funcional de Área

Otro de los objetivos en la generación de mallas es el de tratar de obtener celdas que tengan aproximadamente la misma área, por lo cual se busca minimizar:

$$I_a(\bar{x}) = \frac{1}{2} \int_{B_2} \int_{B_2} J^2(\xi, \eta) d\xi d\eta = \frac{1}{2} \int_{B_2} \int_{B_2} (x_\xi y_\eta - x_\eta y_\xi)^2 d\xi d\eta$$

Las ecuaciones de Euler Lagrange de esta funcional son:

$$\begin{aligned}\frac{d}{d\xi} \left(\frac{\partial J^2}{\partial x_\xi} \right) + \frac{d}{d\eta} \left(\frac{\partial J^2}{\partial x_\eta} \right) &= 0 \\ \frac{d}{d\xi} \left(\frac{\partial J^2}{\partial y_\xi} \right) + \frac{d}{d\eta} \left(\frac{\partial J^2}{\partial y_\eta} \right) &= 0\end{aligned}$$

Como

$$\begin{aligned}\frac{\partial J^2}{\partial x_\xi} &= 2Jy_\eta & \frac{\partial J^2}{\partial x_\eta} &= -2Jy_\xi \\ \frac{\partial J^2}{\partial y_\xi} &= -2Jx_\eta & \frac{\partial J^2}{\partial y_\eta} &= 2Jx_\xi\end{aligned}$$

sustituyendo obtendremos:

$$\begin{aligned}\frac{d}{d\xi} \frac{\partial J^2}{\partial x_\xi} &= \frac{d(2Jy_\eta)}{d\xi} \\ &= 2Jy_{\xi\eta} + 2y_\eta \frac{dJ}{d\xi}\end{aligned}$$

$$\begin{aligned}\frac{d}{d\eta} \frac{\partial J^2}{\partial x_\eta} &= \frac{d(-2Jy_\xi)}{d\eta} \\ &= -2Jy_{\xi\eta} - 2y_\xi \frac{dJ}{d\eta}\end{aligned}$$

$$\begin{aligned}\frac{d}{d\xi} \frac{\partial J^2}{\partial y_\xi} &= \frac{d(-2Jx_\eta)}{d\xi} \\ &= -2Jx_{\xi\eta} - 2x_\eta \frac{dJ}{d\xi}\end{aligned}$$

$$\begin{aligned}\frac{d}{d\eta} \frac{\partial J^2}{\partial y_\eta} &= \frac{d(-2Jx_\xi)}{d\eta} \\ &= 2Jx_{\xi\eta} + 2x_\xi \frac{dJ}{d\eta}\end{aligned}$$

Entonces las ecuaciones de Euler Lagrange pueden ser reescritas como

$$\frac{d}{d\xi} (2Jy_\eta) + \frac{d}{d\eta} (-2Jy_\xi) = 2 \left(y_\eta \frac{dJ}{d\xi} - y_\xi \frac{dJ}{d\eta} \right) = 0$$

$$\frac{d}{d\xi} (-2Jx_\eta) + \frac{d}{d\eta} (2Jx_\xi) = 2 \left(-x_\eta \frac{dJ}{d\xi} + x_\xi \frac{dJ}{d\eta} \right) = 0$$

o bien, en forma matricial por

$$\begin{pmatrix} -x_\eta & x_\xi \\ y_\eta & -y_\xi \end{pmatrix} \begin{pmatrix} J_\xi \\ J_\eta \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Si

$$M = \begin{pmatrix} -x_\eta & x_\xi \\ y_\eta & -y_\xi \end{pmatrix}$$

entonces

$$M(\nabla J) = 0$$

donde J es el Jacobiano es decir

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

es fácil ver que

$$J_\xi = x_{\xi\xi} y_\eta^2 + x_\xi y_{\eta\xi} - x_{\eta\xi} y_\xi - x_\eta y_{\xi\xi}$$

$$J_\eta = y_\eta x_{\eta\xi} + x_\xi y_{\eta\eta} - x_{\eta\eta} y_\xi - x_\eta y_{\eta\xi}$$

de donde

$$\nabla J = \begin{pmatrix} J_\xi \\ J_\eta \end{pmatrix}$$

$$M \begin{pmatrix} x_{\xi\xi} y_\eta^2 + x_\xi y_{\eta\xi} - x_{\eta\xi} y_\xi - x_\eta y_{\xi\xi} \\ y_\eta x_{\eta\xi} + x_\xi y_{\eta\eta} - x_{\eta\eta} y_\xi - x_\eta y_{\eta\xi} \end{pmatrix} = 0$$

$$\begin{aligned} J_\xi y_\eta - J_\eta y_\xi &= (x_{\xi\xi} y_\eta + x_\xi y_{\eta\xi} - x_{\eta\xi} y_\xi - x_\eta y_{\xi\xi}) y_\eta \\ &\quad - (x_{\xi\eta} y_\eta + x_\xi y_{\eta\eta} - x_{\eta\eta} y_\xi - x_\eta y_{\xi\eta}) y_\xi \\ &= (y_\eta^2, -x_\eta y_\eta) \begin{pmatrix} x_{\xi\xi} \\ y_{\xi\xi} \end{pmatrix} + (-2x_\xi y_\eta, x_\xi y_\eta + x_\eta y_\xi) \begin{pmatrix} x_{\xi\eta} \\ y_{\xi\eta} \end{pmatrix} \\ &\quad + (y_\xi^2, -x_\xi y_\xi) \begin{pmatrix} x_{\eta\eta} \\ y_{\eta\eta} \end{pmatrix} \end{aligned}$$

$$\begin{aligned} -J_\xi x_\eta + J_\eta x_\xi &= -(x_{\xi\xi} y_\eta + x_\xi y_{\eta\xi} - x_{\eta\xi} y_\xi - x_\eta y_{\xi\xi}) x_\eta \\ &\quad - (x_{\xi\eta} y_\eta + x_\xi y_{\eta\eta} - x_{\eta\eta} y_\xi - x_\eta y_{\xi\eta}) x_\xi \\ &= (-x_\eta y_\eta, x_\eta^2) \begin{pmatrix} x_{\xi\xi} \\ y_{\xi\xi} \end{pmatrix} + (x_\xi y_\eta + x_\eta y_\xi, -2x_\xi y_\eta) \begin{pmatrix} x_{\xi\eta} \\ y_{\xi\eta} \end{pmatrix} \\ &\quad + (-x_\xi y_\xi, x_\xi^2) \begin{pmatrix} x_{\eta\eta} \\ y_{\eta\eta} \end{pmatrix} \end{aligned}$$

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

ahora que esto se puede escribir en la forma

$$Px_{\xi\xi} + Qx_{\xi\eta} + Rx_{\eta\eta} = 0$$

Donde

$$\begin{aligned} P &= \begin{pmatrix} y_\eta^2 & -x_\eta y_\eta \\ -y_\eta x_\eta & x_\eta^2 \end{pmatrix} \\ Q &= \begin{pmatrix} -2y_\eta y_\xi & x_\xi y_\eta + x_\eta y_\xi \\ x_\xi y_\eta + x_\eta y_\xi & -2x_\eta x_\xi \end{pmatrix} \\ R &= \begin{pmatrix} y_\xi^2 & -x_\xi y_\xi \\ -y_\xi x_\xi & x_\xi^2 \end{pmatrix} \end{aligned}$$

El problema que presentan éstas ecuaciones es que no se puede garantizar la existencia de soluciones.

2.4. Funcional de Suavidad

La idea es considerar el mapeo de la region dada Ω , en el cuadrado unitario y pedir que este mapeo sea armónico es decir, pedir que minimice

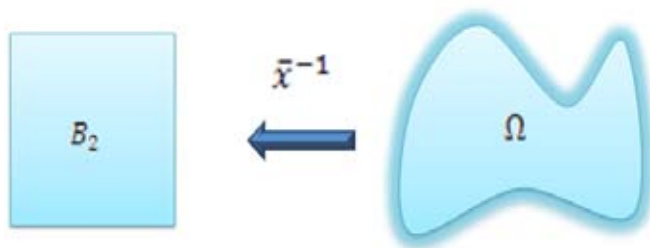


Figura 2.4 Mapeo inverso de Ω a B_2 .

$$I_S(\bar{x}) = \int \int_{\Omega} (\xi_x^2 + \xi_y^2 + \eta_x^2 + \eta_y^2) d\xi d\eta$$

sobre la región. Obsérvese el parecido con la funcional de longitud, en ese caso el mapeo que es armónico es el del cuadrado unitario en la región Ω , la desventaja de esta formulación es que nosotros lo que necesitamos es el mapeo del cuadrado unitario en la región, sin embargo la teoría no nos garantiza que el mapeo sea 1 – 1 para la funcional de longitud y sí lo garantiza para la funcional de suavidad I_S , que llamaremos la funcional armónica. Hay toda una teoría sobre los mapeos armónicos, pero que aquí no nos detendremos en revisarla. Más adelante indicaremos como podemos lograr mapeos 1 – 1 y suaves usando la versión discreta de esta funcional en los capítulos siguientes.

Para obtener el mapeo de interés lo que haremos es hacer un cambio de variable para que la integral transformada esté definida sobre el cuadrado unitario. Para ello nos será útil el siguiente lema:

Lema 2 : Si $\tilde{x} : \Omega \rightarrow B_2$ es 1-1 y C^2 , si $\tilde{x} = (\xi(x, y), \eta(x, y))$ y el mapeo inverso es

$$\tilde{x} = (\bar{x})^{-1} : B_2 \rightarrow \Omega$$

y si además.

$$f : \Omega \rightarrow R$$

en C^2 , entonces :

1.

$$f_x = (y_\eta f_\xi - y_\xi f_\eta) J^{-1}$$

$$f_y = (x_\xi f_\eta - x_\eta f_\xi) J^{-1}$$

donde

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

2.

$$f_{xx} = (y_n^2 f_{\xi\xi} - 2y_\xi y_\eta f_{\xi\eta} + y_\xi^2 f_{\eta\eta}) J^{-2} + [(y_n^2 y_{\xi\xi} - 2y_\xi y_\eta y_{\xi\eta} + y_\xi^2 y_{\eta\eta}) (x_\eta f_\xi - x_\xi f_\eta) + (y_n^2 x_{\xi\xi} - 2y_\xi y_\eta x_{\xi\eta} + y_\xi^2 x_{\eta\eta}) (y_\xi f_\eta - y_\eta f_\xi)] J^{-3}$$

$$f_{yy} = (x_n^2 f_{\xi\xi} - 2x_\xi x_\eta f_{\xi\eta} + x_\xi^2 f_{\eta\eta}) J^{-2} + [(x_n^2 y_{\xi\xi} - 2x_\xi x_\eta y_{\xi\eta} + x_\xi^2 y_{\eta\eta}) (x_\eta f_\xi - x_\xi f_\eta) + (x_n^2 x_{\xi\xi} - 2x_\xi x_\eta x_{\xi\eta} + x_\xi^2 x_{\eta\eta}) (y_\xi f_\eta - y_\eta f_\xi)] J^{-3}$$

La demostración se omite por ser muy larga pero automática directa.

Usaremos ahora el lema anterior para transformar las ecuaciones de Euler-Lagrange:

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

sujeto a las correspondientes condiciones a la frontera, en ecuaciones para el mapeo

$$\bar{x} = (x(\xi, \eta), y(\xi, \eta))$$

que van a ser de la forma

$$\begin{aligned} \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} &= 0 \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} &= 0 \end{aligned}$$

Hagamos $f = \xi$ entonces aplicando el lema tenemos:

$$\xi_x = y_\eta J^{-1}; \quad \xi_y = -x_\eta J^{-1}$$

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

Donde

$$\xi_{xx} = [(y_\eta^2 y_{\xi\xi} - 2y_\xi y_\eta y_{\xi\eta} + y_\xi^2 y_{\eta\eta})x_\eta - y_\eta (y_\eta^2 x_{\xi\xi} - 2y_\xi y_\eta x_{\xi\eta} + y_\xi^2 x_{\eta\eta})] J^{-3}$$

$$\xi_{yy} = [x_\eta (x_\eta^2 y_{\xi\xi} - 2x_\xi x_\eta y_{\xi\eta} + x_\xi^2 y_{\eta\eta}) - y_\eta (x_\eta^2 x_{\xi\xi} - 2x_\xi x_\eta x_{\xi\eta} + x_\xi^2 x_{\eta\eta})] J^{-3}$$

Si ahora tomamos $f = \eta$, nos queda :

$$\eta_x = -y_\xi J^{-1} \qquad \eta_y = x_\xi J^{-1}$$

$$\eta_{xx} = [-x_\xi (y_\eta^2 y_{\xi\xi} - 2y_\xi y_\eta y_{\xi\eta} + y_\xi^2 y_{\eta\eta}) + y_\xi (y_\eta^2 x_{\xi\xi} - 2y_\xi y_\eta x_{\xi\eta} + y_\xi^2 x_{\eta\eta})] J^{-3}$$

$$\eta_{yy} = [-x_\xi (x_\eta^2 y_{\xi\xi} - 2x_\xi x_\eta y_{\xi\eta} + x_\xi^2 y_{\eta\eta}) + y_\xi (x_\eta^2 x_{\xi\xi} - 2x_\xi x_\eta x_{\xi\eta} + x_\xi^2 x_{\eta\eta})] J^{-3}$$

de donde sustituyendo

$$\begin{aligned} J^3 (\xi_{xx} + \xi_{yy}) &= x_\eta [(x_\eta^2 + y_\eta^2) y_{\xi\xi} - 2(x_\xi x_\eta + y_\xi y_\eta) y_{\xi\eta} \\ &\quad + (x_\xi^2 + y_\xi^2) y_{\eta\eta}] - y_\eta [(x_\eta^2 + y_\eta^2) x_{\xi\xi} \\ &\quad - (x_\xi x_\eta + y_\xi y_\eta) x_{\xi\eta} + (x_\xi^2 + y_\xi^2) x_{\eta\eta}] \end{aligned}$$

Si hacemos

$$\alpha = x_\eta^2 + y_\eta^2$$

$$\beta = x_\xi x_\eta + y_\xi y_\eta$$

$$\gamma = x_\xi^2 + y_\xi^2$$

entonces

$$\begin{aligned} J^3 (\xi_{xx} + \xi_{yy}) &= [x_\eta (\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta})] \\ &\quad - [y_\eta (\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta})] \end{aligned}$$

de manera parecida obtenemos:

$$J^3 (\eta_{xx} + \eta_{yy}) = [-x_\xi (\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta})] \\ + [y_\xi (\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta})]$$

como

$$\xi_{xx} + \xi_{yy} = 0 \\ \eta_{xx} + \eta_{yy} = 0$$

sustituyendo en la ecuación anterior y haciendo

$$e_1 = \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} \\ e_2 = \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta}$$

implican

$$x_\eta e_2 - y_\eta e_1 = 0 \\ -x_\xi e_2 - y_\xi e_1 = 0$$

El determinante del sistema es

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

Si $J \neq 0$ sobre B_2 , es decir si el mapeo es 1-1 y sobre, entonces

$$e_1 = e_2 = 0.$$

así que para obtener el mapeo tenemos que resolver el sistema:

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = 0 \\ \alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = 0$$

sujeto a las condiciones a la frontera

$$\bar{x}(\partial B_2) = \partial\Omega$$

2.5. Funcional de Ortogonalidad

Entre los sistemas curvilíneos más usados están aquellos que son ortogonales, ya que son muy útiles en la solución de ecuaciones diferenciales. La condición para que los vectores tangentes $\bar{x}_\xi, \bar{x}_\eta$ en el punto $x(\xi, \eta)$ sean ortogonales es que

$$\bar{x}_\xi(\xi, \eta) \cdot \bar{x}_\eta(\xi, \eta) = 0$$

de aquí que es natural buscar en mapeos $\bar{x}(\xi, \eta)$ que sean lo más ortogonales posibles dadas las condiciones a la frontera, esto nos lleva a considerar la funcional:

$$\begin{aligned} I_o(\bar{x}) &= \frac{1}{2} \int \int_{B_2} (\bar{x}_\xi \cdot \bar{x}_\eta)^2 d\xi d\eta \\ &= \frac{1}{2} \int \int_{B_2} (x_\xi y_\xi + x_\eta y_\eta)^2 d\xi d\eta \end{aligned}$$

a la cual se le conoce como la funcional de ortogonalidad.

El problema de minimizar

$$\min_{\bar{x}(\partial B_2) = \partial \Omega} I_o(\bar{x})$$

da lugar a las ecuaciones de Euler-Lagrange

$$((\bar{x}_\xi \cdot \bar{x}_\eta) \bar{x}_\eta)_\xi + ((\bar{x}_\xi \cdot \bar{x}_\eta) \bar{x}_\xi)_\eta = 0$$

Desarrollando y agrupando obtenemos

$$Px_{\xi\xi} + Qx_{\xi\eta} + Rx_{\eta\eta} + S = 0$$

Donde P, Q, R, S son las matrices

$$P = \begin{pmatrix} x_\eta^2 & x_\eta y_\eta \\ x_\eta y_\eta & y_\eta^2 \end{pmatrix}$$

$$Q = \begin{pmatrix} 4x_\xi x_\eta + 2y_\xi y_\eta & x_\xi y_\eta + x_\eta y_\xi \\ x_\xi y_\eta + x_\eta y_\xi & 4y_\xi y_\eta + 2x_\xi x_\eta \end{pmatrix}$$

$$R = \begin{pmatrix} x_\xi^2 & x_\xi y_\xi \\ x_\xi y_\xi & y_\xi^2 \end{pmatrix}$$

$$S = 0$$

Este sistema de ecuaciones diferenciales, es cuasilineal no elíptico, por lo que no es posible garantizar solución aún en casos sencillos, razón por la cual esta funcional siempre se utiliza combinado con otra funcional, ya sea con la de área, ó la de longitud ó ambas.

2.6. Ejemplos

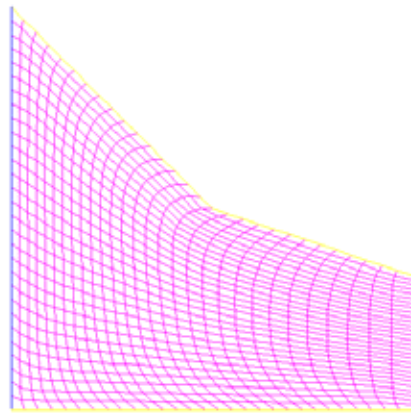


Figura 2.5 Funcional combinado Área con longitud.

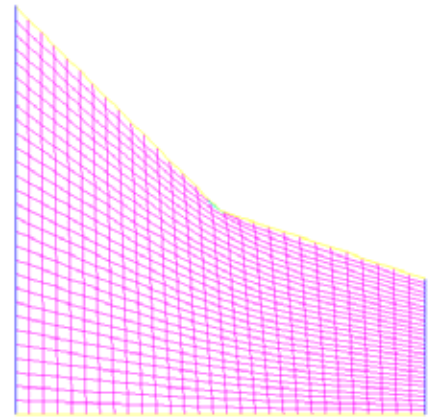


Figura 2.6 Funcional de longitud.

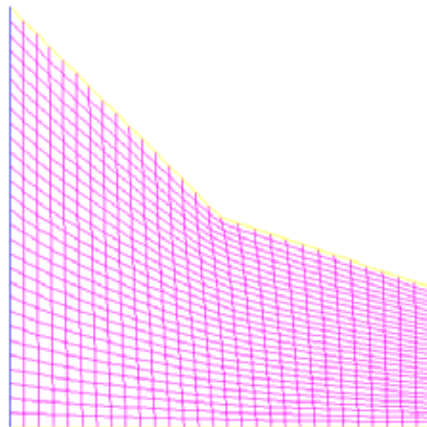


Figura 2.7 Funcional de suavidad.

Capítulo 3

Métodos Variacionales Discretos

En este capítulo desarrollaremos la teoría básica de los métodos variacionales discretos. Estos métodos se deducen siguiendo las ideas que Ivanenko[28, 29, 30] aplicó a la funcional de suavidad o armónica como la llamaremos de ahora en adelante, la idea fundamental consiste en discretizar la funcional en lugar de las ecuaciones de Euler-Lagrange, así en vez de resolver un sistema de ecuaciones diferenciales parciales lo que haremos será resolver un problema de optimización de gran escala. Empezaremos por formular el problema y después veremos las propiedades del mapeo bilineal que jugará un papel fundamental en el procedimiento general y por último lo aplicaremos a las funcionales de longitud, área y suavidad.

3.1. Formulación del Problema

El problema que tenemos de construir una malla estructurada sobre una región Ω , simplemente conexa, se puede plantear, de la manera siguiente:

Problema 3 (A) *Dada una región Ω , encontrar un mapeo suave 1 – 1 del cuadrado unitario sobre Ω ; de tal forma que las fronteras se correspondan, es decir, encontrar*

$$\bar{x} = (x(\xi, \eta), y(\xi, \eta))$$

Tal que

1. \bar{x} es C^1
2. \bar{x} es 1-1
3. $\bar{x}(\partial B_2) = \partial\Omega$.

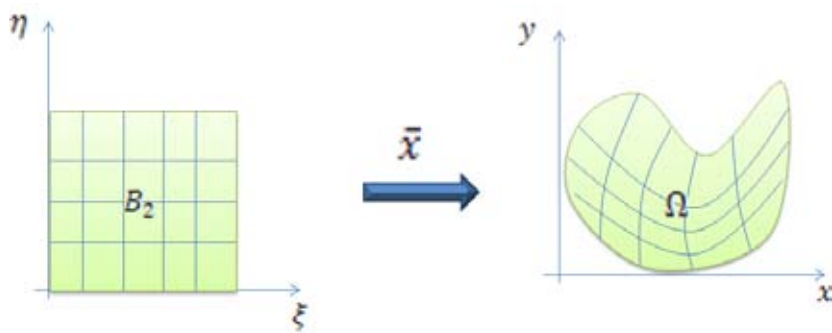


Figura3.1 Mapeo directo entre la región B_2 y Ω .

Otra forma de plantear el problema es la siguiente:

Problema 4 (B) Construir un homeomorfismo ϕ de la frontera del cuadrado unitario en la frontera de Ω .

$$\phi : \partial B_2 \longrightarrow \Omega$$

y además construir una extensión del mapeo ϕ a todo B_2 , es decir, encontrar

$$\bar{x} : B_2 \longrightarrow \Omega$$

1-1, C^1 y tal que

$$\bar{x}(\partial B_2) = \phi(\partial B_2) = \partial\Omega.$$

Usualmente ϕ se construye eligiendo cuatro puntos en la frontera de Ω que

indican como se van a corresponder las fronteras del cuadrado y las de Ω .

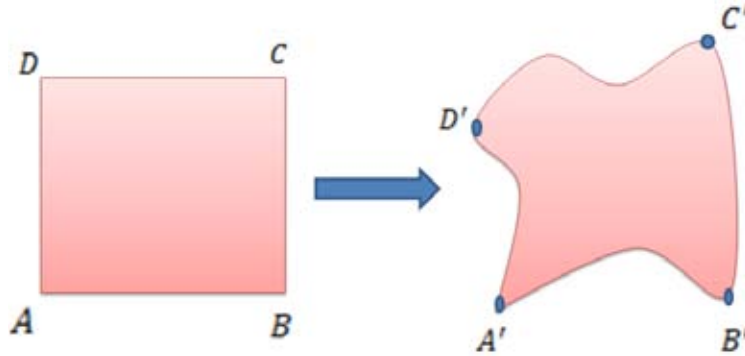


Figura 3.2 Construcción del homeomorfismo ϕ por medio de una correspondencia de las fronteras del cuadrado y Ω .

El problema que tenemos es como garantizamos que el mapeo que obtengamos sea 1 - 1. Para ello necesitaremos que el Jacobiano

$$J(\xi, \eta) = x_\xi y_\eta - x_\eta y_\xi$$

sea positivo en la región y si las orientaciones se preservan, entonces tenemos el siguiente teorema:

Teorema 5 Si $\bar{x} : B_2 \longrightarrow \Omega$, C^1 , satisfice

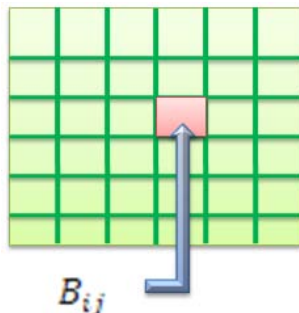
$$\begin{aligned} \bar{x}(\partial B_2) &= \partial \Omega \\ J(\xi, \eta) &> 0 \quad \forall (\xi, \eta) \in B_2 \end{aligned}$$

Entonces $\bar{x}(\xi, \eta)$ es un homeomorfismo de B_2 sobre Ω .

Este resultado es difícil de usar en la práctica ya que no se cuenta con una expresión analítica del Jacobiano por lo que buscaremos dividir la región B_2 en regiones sencillas donde sea fácil evaluar Jacobiano.

Si consideramos una malla $\{B_{ij}\}$ sobre el cuadrado unitario

$$B_2 = \bigcup_{\substack{i=1 \\ j=1}}^{n-1} B_{ij}$$

Figura 3.3 Celda B_{ij} .

Tenemos el siguiente resultado

Teorema 6 Sea

$$\bar{x} : B_2 \longrightarrow \Omega$$

continuo y $\{\bar{x}_{ij}\}$ las restricciones de \bar{x} a las celdas, es decir

$$\bar{x}_{ij} : x \mid B_{ij} \longrightarrow \Omega$$

si $\{\bar{x}_{ij}\}$ son de clase C^1 , para $i = 1, \dots, m-1$; $j = 1, \dots, n-1$ con

$$\bar{x}(\partial B_2) = \partial \Omega.$$

y si los Jacobianos J_{ij} de \bar{x}_{ij} satisfacen

$$J_{ij} > 0, \quad i = 1, \dots, m-1; \quad j = 1, \dots, n-1.$$

entonces, \bar{x} es un homeomorfismo de B_2 sobre Ω .

Este resultado es el que vamos a usar en la práctica, para ello nos restringiremos a regiones Ω con frontera poligonal, simplemente conexas, y elegiremos los mapeos \bar{x}_{ij} como funciones bilineales, así que empezaremos por considerar el mapeo bilineal.

3.2. El Mapeo Bilineal

La región plana más simple para construir una malla es un cuadrilátero y la transformación más simple que transforma el cuadrado unitario en un cuadrilátero arbitrario PQRS, es el mapeo bilineal

$$\bar{r} : B_2 \longrightarrow \square PQRS$$

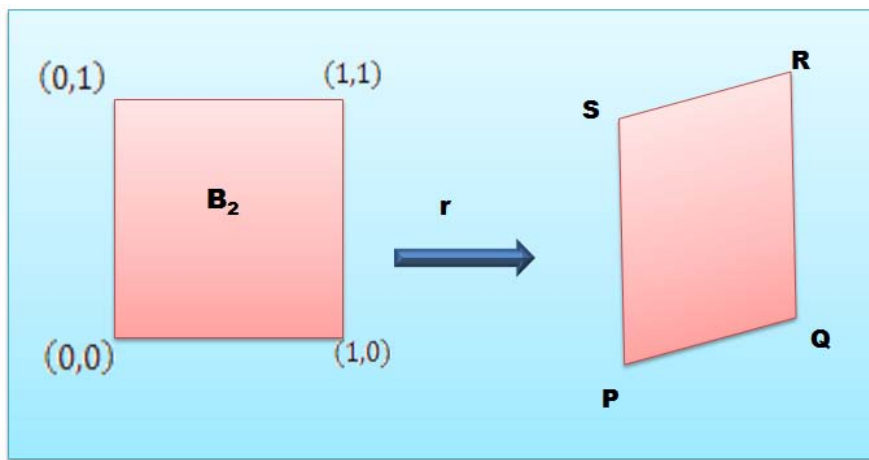


Figura 3.4 Representación del mapeo bilineal.

que podemos reescribir en su forma vectorial como

$$\bar{r}(s, t) = \begin{pmatrix} x(s, t) \\ y(s, t) \end{pmatrix}$$

donde

$$\bar{r} = \bar{a} + \bar{b}s + \bar{c}t + \bar{d}st$$

$$\bar{a} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}, \quad \bar{b} = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}, \quad \bar{c} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix}, \quad \bar{d} = \begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix},$$

Calculemos los coeficientes, imponiendo la condición de que los vértices de B_2 se transformen en los vértices del $\square PQRS$

$$\begin{aligned}
 \bar{r}(0,0) &= \bar{a} = P \\
 \bar{r}(1,0) &= \bar{a} + \bar{b} = Q \quad \text{donde } \bar{b} = Q - P \\
 \bar{r}(1,1) &= \bar{a} + \bar{b} + \bar{c} + \bar{d} = R \\
 \bar{r}(0,1) &= \bar{a} + \bar{c} = S \quad \text{donde } \bar{c} = S - P
 \end{aligned}$$

$$\begin{aligned}
 D &= R - \bar{a} - \bar{b} - \bar{c} \\
 &= R - P - (Q - P) - (S - P) \\
 &= R - Q - S + P
 \end{aligned}$$

sustituyendo se tiene que

$$\bar{r}(s,t) = P + (Q - P)s + (S - P)t + [R + P - (S + Q)]st$$

que también se puede escribir como

$$\begin{aligned}
 &= P(1 - s - t + st) + Q(s - st) + Rst + S(t - st) \\
 &= P(1 - s)(1 - t) + Qs(1 - t) + Rst + St(1 - s)
 \end{aligned}$$

El mapeo bilineal se puede obtener también de la forma siguiente; el lado PS del cuadrilátero $PQRS$ se puede reescribir como

$$\bar{x}_l(t) = (1 - t)P + tS$$

$$0 \leq t \leq 1.$$

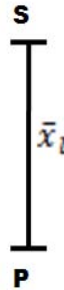


Figura 3.5 Lado PS del cuadrilátero PQRS.

De manera similar el lado QR se puede escribir, para $0 \leq t \leq 1$,

$$\bar{x}_r(t) = (1-t)Q + tR.$$

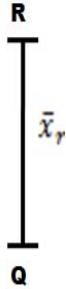


Figura 3.6 Lado QR del cuadrilátero PQRS.

Por tanto, el segmento $\bar{x}_l(t)$, $\bar{x}_r(t)$ se puede ahora escribir como

$$r(s,t) = (1-s)\bar{x}_l(t) + s\bar{x}_r(t) \quad 0 \leq s,t \leq 1$$

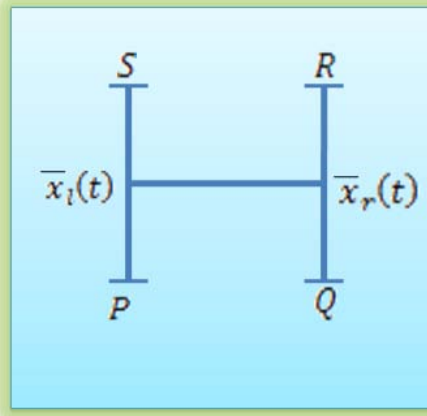


Figura 3.7 Segmento $\bar{x}_l(t)$, $\bar{x}_r(t)$.

Es fácil probar que este mapeo satisface las condiciones de frontera, es decir, que la frontera de B_2 va en la frontera de $\square PQRS$.

$$\bar{r}(s,t) = (1-s)(1-t)P + (1-s)tS + s(1-t)Q + stR$$

Un procedimiento similar puede ser aplicado para unir un punto $\bar{x}_b(s)$ en la subfrontera \bar{x}_b y un punto $\bar{x}_\tau(s)$ en la subfrontera \bar{x}_τ .

$$\bar{x}_\tau(s) = (1-s)S + sR$$

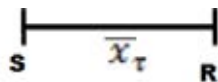


Figura 3.8 Lado SR del cuadrilátero PQRS.

$$\bar{x}_b(s) = (1-s)P + sQ$$

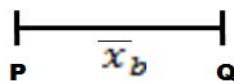


Figura 3.9 Lado PQ del cuadrilátero PQRS.

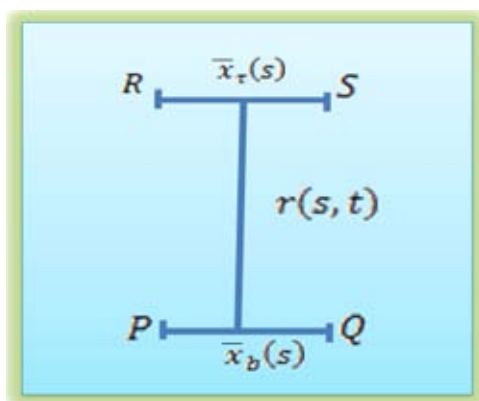


Figura 3.10 Segmento $\bar{x}_b(t)$, $\bar{x}_\tau(t)$.

$$r(s, t) = (1-t)\bar{x}_b(s) + \bar{x}_\tau(s)$$

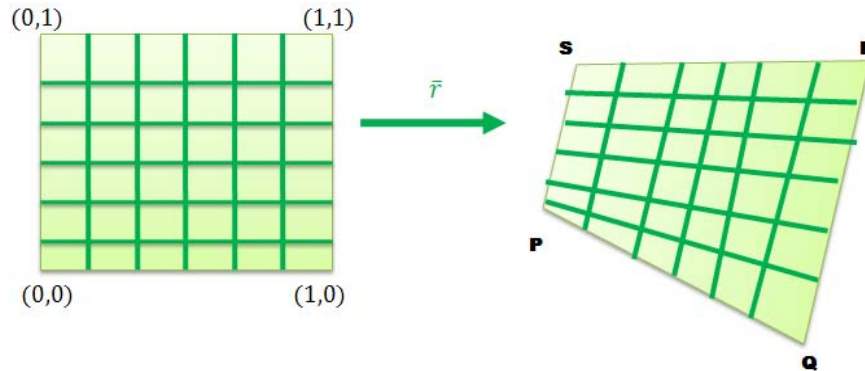


Figura 3.11 Mapeo Bilineal.

Nuestro siguiente punto a tratar será estudiar las propiedades del mapeo bilineal definido anteriormente.

Iniciaremos discutiendo de manera breve la definición de triángulos orientados y sus áreas, que son parte esencial de nuestro estudio.

Diremos que un triángulo ABC está orientado positivamente si cuando nos movemos sobre su frontera en el orden ABC el interior queda a mano izquierda, en caso contrario, diremos que está orientado negativamente.

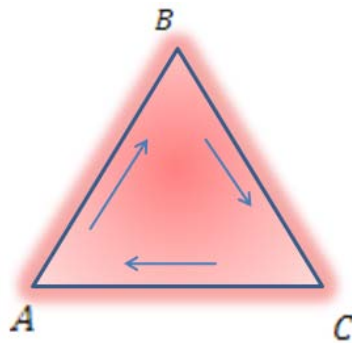


Figura 3.12 Orientación negativa de un triángulo.

Orientación en el sentido del movimiento de las manecillas del reloj (Orientación negativa) ΔABC (ó ΔBCA ó ΔCAB).

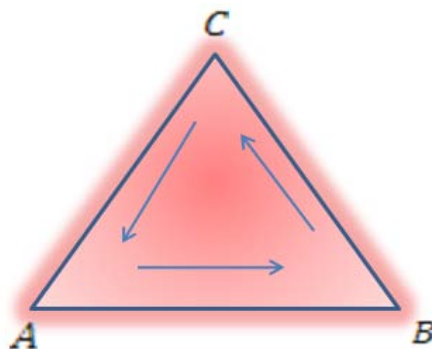


Figura 3.12 Orientación positiva de un triángulo.

Podemos resumir esto diciendo que, un triángulo orientado no se determina simplemente por sus vértices, sino también por los tres vértices dados en cierto orden.

Definición 7 Por *área orientada* de un triángulo orientado ΔABC , debe entenderse aquel número cuyo valor absoluto, es igual al área del triángulo (no orientado) que tiene los vértices A, B, C ; el área orientada del triángulo es positiva, si el triángulo está orientado en sentido contrario al movimiento de las manecillas del reloj y negativa si el triángulo está orientado en el sentido del movimiento de las manecillas del reloj.

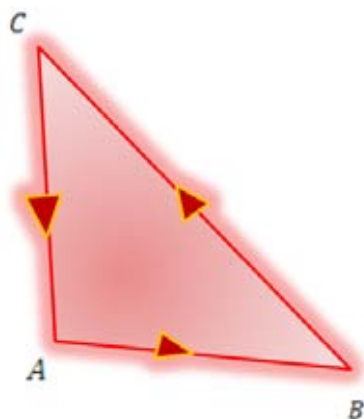


Figura 3.14 Área Positiva del triángulo que tiene los vértices A,B,C.

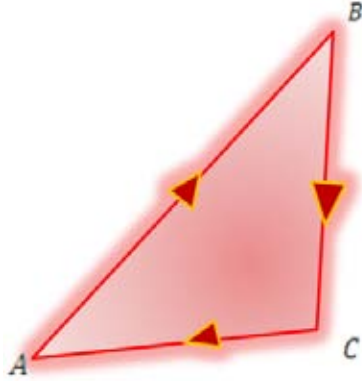


Figura 3.15 Área negativa del triángulo que tiene los vértices A,B,C.

Cuando consideremos el área de un triángulo siempre se referirá al área orientada.

Es fácil demostrar que:

$$\text{area}(A, B, C) = \frac{1}{2} \det(B - A, C - A)$$

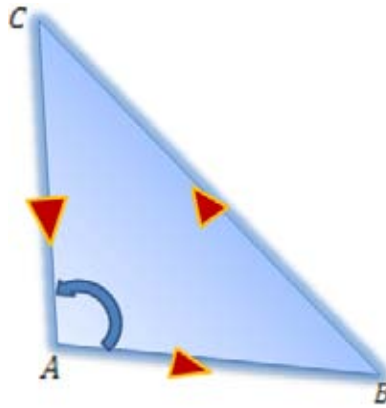


Figura 3.16 Triángulo A,B,C que muestra la definición área orientada.

$$\begin{aligned} &= \frac{1}{2} \begin{vmatrix} x_b - x_a & x_c - x_a \\ y_b - y_a & y_c - y_a \end{vmatrix} \\ &= \frac{1}{2} (B - A)^T J_2 (C - A), \end{aligned}$$

donde

$$J_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

Definición 8

$$\alpha(A, B, C) = 2\text{area}(\Delta(A, B, C))$$

Si consideramos el cuadrilátero $PQRS$ orientado por el orden de sus vértices, entonces este induce la orientación en los 4 triángulos:

$$\begin{aligned} \Delta_1 & : \Delta(S, P, Q) \\ \Delta_2 & : \Delta(P, Q, R) \\ \Delta_3 & : \Delta(Q, R, S) \\ \Delta_4 & : \Delta(R, S, P) \end{aligned}$$

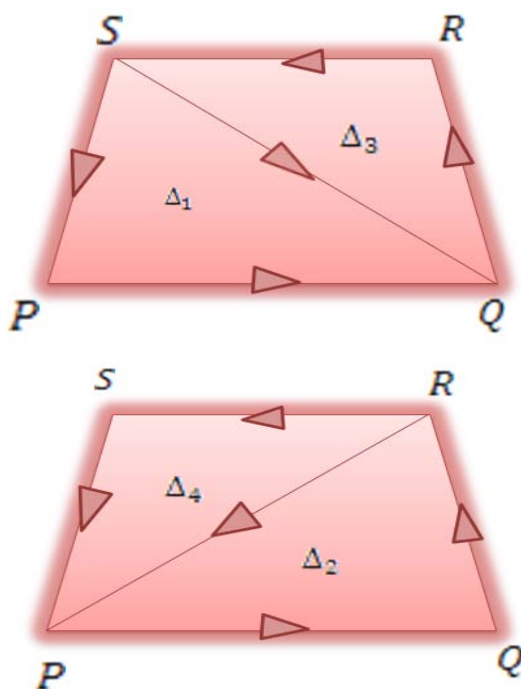


Figura 3.17 Triángulos definidos sobre un cuadrilátero a partir de sus vértices.

Ejemplo

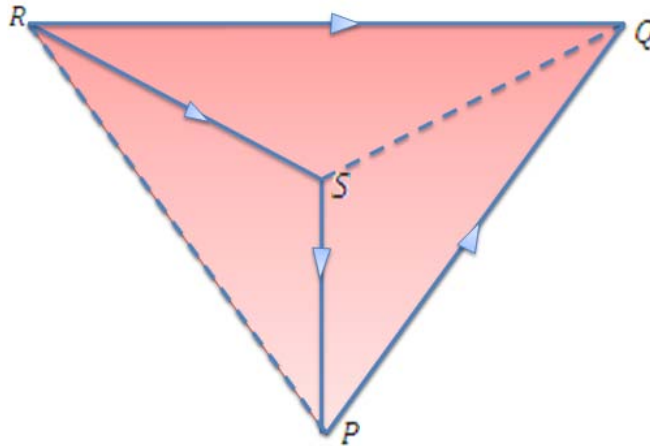


Figura 3.18 Ejemplo de triángulos orientados.

- $\triangle SPQ \longrightarrow$ orientado positivamente.
- $\triangle PQR \longrightarrow$ orientado positivamente.
- $\triangle QRS \longrightarrow$ orientado positivamente.
- $\triangle RSP \longrightarrow$ orientado negativamente.

Nos interesa la relación de las áreas con el Jacobiano por lo que calculamos, para el mapeo,

$$\bar{r}(s, t) = P + (Q - P)s + (S - P)t + [R + P - (S + Q)]st$$

Los vectores tangentes están dados por:

$$\begin{aligned} \frac{\partial \bar{r}}{\partial s} &= \bar{r}_s(s, t) = Q - P + [R + P - (S + Q)]t \\ \frac{\partial \bar{r}}{\partial t} &= \bar{r}_t(s, t) = S - P + [R + P - (S + Q)]s \end{aligned}$$

$$\begin{aligned} r(s, t) &= \begin{pmatrix} x(s, t) \\ y(s, t) \end{pmatrix} \\ r_s(s, t) &= \begin{pmatrix} x_s(s, t) \\ y_s(s, t) \end{pmatrix} \end{aligned}$$

$$\begin{aligned}x_s(s, t) &= x_q - x_p + [x_r + x_p - (x_s + x_q)] t \\y_s(s, t) &= y_q - y_p + [y_r + y_p - (y_s + y_q)] s\end{aligned}$$

De manera similar,

$$\begin{aligned}x_t(s, t) &= x_s - x_p + [x_r + x_p - (x_s + x_q)] t \\y_t(s, t) &= y_s - y_p + [y_r + y_p - (y_s + y_q)] s\end{aligned}$$

Como sabemos el jacobiano está dado por

$$J(s, t) = x_s y_t - x_t y_s,$$

por lo que si sustituimos las parciales en el jacobiano, tendremos:

$$\begin{aligned}J(s, t) &= (x_q - x_p + [x_r + x_p - (x_s + x_q)] t) * (y_s - y_p + [y_r + y_p - (y_s + y_q)] s) \\&\quad - (x_s - x_p + [x_r + x_p - (x_s + x_q)] t) * (y_q - y_p + [y_r + y_p - (y_s + y_q)] s) \\&= A_0 + A_1 s + A_2 t.\end{aligned}$$

Evaluando el Jacobiano en las esquinas del cuadrilátero unitario tenemos

$$\begin{aligned}J(0, 0) &= -(-x_p + x_s)(-y_p + y_q) + (-x_p + x_q)(-y_p + y_s) \\&= (x_q - x_p)(y_s - y_p) - (y_q - y_p)(x_s - x_p) \\&= \begin{vmatrix} x_q - x_p & x_s - x_p \\ y_q - y_p & y_s - y_p \end{vmatrix} \\&= \det(P - Q, S - P)\end{aligned}$$

$$= 2\text{area}(\Delta SPQ)$$

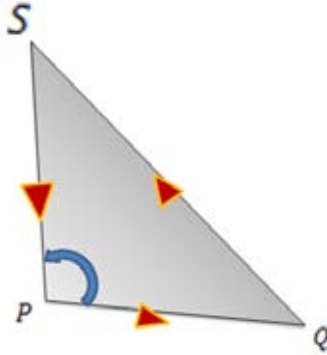


Figura 3.19 Triángulo SPQ.

$$= \alpha(S, P, Q) = J(0, 0) = A_0.$$

Análogamente,

$$\begin{aligned} J(1, 0) &= -(-x_q + x_r)(-y_p + y_q) + (-x_p + x_q)(-y_p + y_r) \\ &= (x_r - x_q)(y_p - y_q) - (y_r - y_q)(x_p - x_q) \\ &= \begin{vmatrix} x_r - x_q & x_p - x_q \\ y_r - y_q & y_p - y_q \end{vmatrix} \\ &= \det(R - Q, P - Q) \end{aligned}$$

$$= 2\text{area}(\Delta PQR)$$

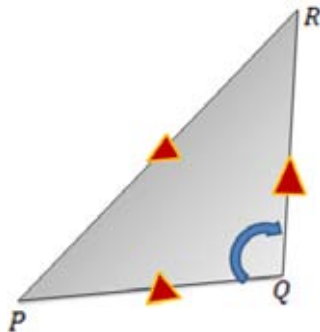


Figura 3.20 Triángulo PQR.

$$= \alpha(P, Q, R) = J(1, 0) = A_0 + A_1$$

y

$$\begin{aligned} J(0, 1) &= -(-x_p + x_s)(y_r - y_s) + (x_r - x_s)(-y_p + y_s) \\ &= (x_p - x_s)(y_r - y_s) - (y_p - y_s)(x_r - x_s) \\ &= \begin{vmatrix} x_p - x_s & x_r - x_s \\ y_p - y_s & y_r - y_s \end{vmatrix} \\ &= \det(P - S, R - S) \end{aligned}$$

$$= 2\text{area}(\Delta RSP)$$

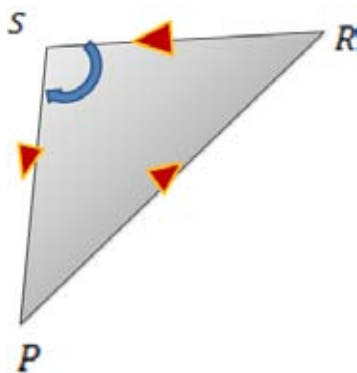


Figura 3.21 Triángulo RSP.

$$= \alpha(R, S, P) = J(0, 1) = A_0 + A_2$$

$$\begin{aligned} J(1, 1) &= (x_r - x_s)(-y_q + y_r) - (-x_q + x_r)(y_r - y_s) \\ &= \begin{vmatrix} x_s - x_r & x_q - x_r \\ y_s - y_r & y_q - y_r \end{vmatrix} \\ &= \det(S - R, Q - R) \end{aligned}$$

$$= 2\text{area}(\Delta QRS)$$

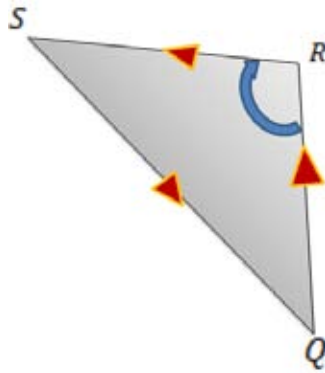


Figura 3.22 Triángulo QRS.

$$= \alpha(Q, R, S) = J(1, 1) = A_0 + A_1 + A_2$$

Estos cálculos se resumen en el siguiente teorema:

Teorema 9 *El Jacobiano del mapeo bilineal es lineal sobre un cuadrilátero y está determinado por su valor sobre tres puntos distintos en B_2 . En particular el Jacobiano de la transformación puede escribirse de la forma:*

$$J(s, t) = (1 - s - t)J(0, 0) + sJ(1, 0) + tJ(0, 1)$$

Demostración. Como ya vimos $J(s, t)$ es lineal,

$$J(s, t) = A_0 + sA_1 + tA_2$$

con A_0 , A_1 y A_2 números reales. Los valores del Jacobiano en las esquinas, bajo esta representación, son:

$$\begin{aligned} J(0, 0) &= A_0 \\ J(1, 0) &= A_0 + A_1 \\ J(1, 1) &= A_0 + A_1 + A_2 \\ J(0, 1) &= A_0 + A_2 \end{aligned}$$

considerando que $J(1,1) = J(1,0) + J(0,1) - J(0,0)$ si se sustituyen las expresiones anteriores, se obtiene que

$$\begin{aligned} A_0 &= J(0,0) \\ A_1 &= J(1,0) - J(0,0) \\ A_2 &= J(0,1) - J(0,0) \end{aligned}$$

Así que si se sustituyen estos valores se tiene que

$$\begin{aligned} J(s,t) &= J(0,0) + s[J(1,0) - J(0,0)] + t[J(0,1) - J(0,0)] \\ &= (1-s-t)J(0,0) + sJ(1,0) + tJ(0,1) \end{aligned}$$

■

La validez del corolario que sigue se basa en el hecho de que la gráfica de J está sobre un plano. De la misma linealidad del Jacobiano se tiene que su valor menor se alcanza en algunas de las esquinas de B_2 .

Corolario 10

$$\min_{s,t} J(s,t) = \min \{J(0,0), J(1,0), J(1,1), J(0,1)\}$$

Por tanto podemos concluir que el Jacobiano es positivo solo si lo es en cada una de sus esquinas, es decir $\square PQRS$ es convexo si y sólo si los cuatro triángulos tienen área positiva.

Este sería un criterio muy útil en la práctica para saber si una malla es convexa.

3.3. Discretización General

Procederemos ahora a describir un procedimiento general para obtener una funcional discreta a partir de una continua.

Consideremos la funcional

$$I(x) = \iint_{B_2} f(\bar{x}_\xi, \bar{x}_\eta) d\xi d\eta$$

y una malla uniforme de $m \times n$ sobre B_2 y sean $\{B_{ij}\}$, $i = 1, 2, \dots, m$ y $j = 1, 2, \dots, n$, las celdas de la malla, entonces como $B_2 = \cup_{i,j} B_{ij}$

$$I(\bar{x}) = \sum_{i=1}^m \sum_{j=1}^n \iint_{B_{ij}} f(\bar{x}_\xi, \bar{x}_\eta) d\xi d\eta.$$

Ahora, hacemos una primera aproximación, el mapeo

$$\bar{x} : B_2 \longrightarrow \Omega,$$

vamos a aproximarlo por un mapeo

$$\bar{r} : B_2 \longrightarrow \Omega$$

donde \bar{r} es bilineal por pedazos, es decir, en cada celda B_{ij} , \bar{r} está dado por un mapeo bilineal \bar{r}_{ij} .

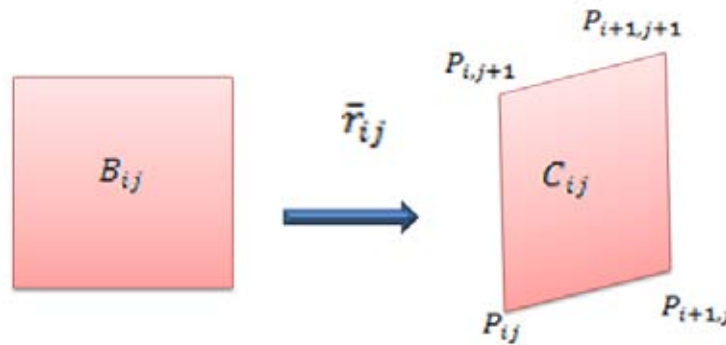


Figura 3.23 Mapeo de la celda B_{ij} a la celda C_{ij} .

entonces

$$I_{ij} = \iint_{B_{ij}} f(\bar{x}_\xi, \bar{x}_\eta) d\xi d\eta \approx \iint_{B_{ij}} f(\bar{r}_{ij\xi}, \bar{r}_{ij\eta}) d\xi d\eta,$$

y esta última integral la aproximamos por la regla de cuadratura de los cuatro puntos, que es el promedio de los valores en los cuatro vértices de B_{ij} .

Obtenemos

$$I_{ij} \approx \frac{1}{4} [f(\bar{r}_{ij\xi}(P), \bar{r}_{ij\eta}(P)) + f(\bar{r}_{ij\xi}(Q), \bar{r}_{ij\eta}(Q)) + f(\bar{r}_{ij\xi}(R), \bar{r}_{ij\eta}(R)) + f(\bar{r}_{ij\xi}(S), \bar{r}_{ij\eta}(S))] = \tilde{I}_{ij},$$

donde

$$P = P_{ij}, \quad Q = P_{i,j+1}, \quad R = P_{i+1,j+1}, \quad S = P_{i,j+1}$$

finalmente, nos queda

$$\begin{aligned} I(\tilde{x}) &\approx \sum_{i=1}^{m-1} \sum_{j=1}^n \tilde{I}_{i,j} = \\ &= \frac{1}{4} \sum_{i=1}^n \sum_{j=1}^n [\bar{f}[\bar{r}_{ij\xi}(P_{i,j}), \bar{r}_{ij\eta}(P_{i,j})]] + [\bar{f}[\bar{r}_{ij\xi}(P_{i+1,j}), \bar{r}_{ij\eta}(P_{i+1,j})]] \\ &\quad + [\bar{f}[\bar{r}_{ij\xi}(P_{i+1,j+1}), \bar{r}_{ij\eta}(P_{i+1,j+1})]] + [\bar{f}[\bar{r}_{ij\xi}(P_{i,j+1}), \bar{r}_{ij\eta}(P_{i,j+1})]] \\ &\triangleq I_d(\{P_{i,j}\}) \end{aligned}$$

El símbolo en la última relación significa que esta expresión va a ser la funcional discreta asociada a la funcional continua discretizada.

$I_d(\{P_{i,j}\})$ es una función de los puntos $P_{i,j}$, y lo que queremos es determinar los puntos $\{P_{i,j}\}$ de la malla de tal forma que optimice la funcional $I_d(\{P_{i,j}\})$, como suponemos de entrada que se tiene un homeomorfismo de la frontera de B_2 en la frontera de Ω , esto significa que los puntos de la frontera de la región se definen como:

$$\begin{aligned} l_1 &= \{P_{1,1}, P_{2,1}, \dots, P_{m,1}\}, \\ l_2 &= \{P_{m,1}, P_{m,2}, \dots, P_{m,n}\}, \\ l_3 &= \{P_{1,n}, P_{2,n}, \dots, P_{m,n}\}, \\ l_4 &= \{P_{1,1}, P_{1,2}, \dots, P_{1,n}\}. \end{aligned}$$

son conocidos (ver figura 3.24).

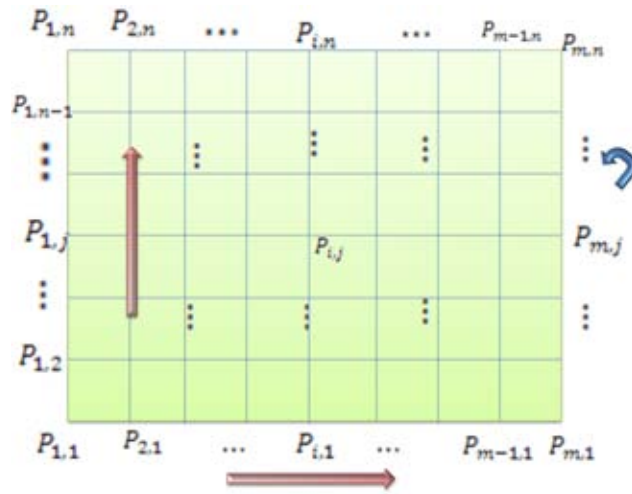


Figura 3.24 Puntos de la frontera de la región.

Definición 11 Una malla estructurada G de Ω , de $m \times n$ es una colección $\{P_{i,j}\}$ de puntos de R^2 tales que los puntos

$$\begin{aligned} l_1(G) &= \{P_{1,1}, P_{2,1}, \dots, P_{m,1}\}, \\ l_2(G) &= \{P_{m,1}, P_{m,2}, \dots, P_{m,n}\}, \\ l_3(G) &= \{P_{1,n}, P_{2,n}, \dots, P_{m,n}\}, \\ l_4(G) &= \{P_{1,1}, P_{1,2}, \dots, P_{1,n}\}. \end{aligned}$$

están sobre la frontera de G .

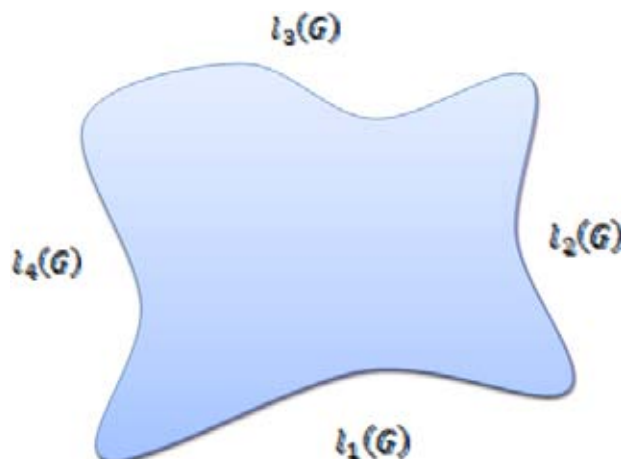


Figura 3.25 Segmentos de la frontera de la región Ω .

Así, el problema que tenemos es determinar los puntos interiores de G , de tal forma que la malla G tenga propiedades geométricas adecuadas.

Es decir, tenemos que resolver el problema

$$\min_G I_d(G)$$

sobre todas las mallas G de $m \times n$ con los mismos puntos en la frontera de Ω .

Pasaremos ahora a aplicar el procedimiento anterior para obtener las funcionales discretas asociadas a las funcionales continuas del capítulo anterior.

Como necesitamos formulas precisas y el manejo de los indices debe ser muy cuidadoso, usaremos un esquema general de tal forma que las funcionales queden en la forma más clara y compacta que sea posible.

Es decir, necesitamos unas fórmulas sencillas para los mapeos bilineales,

$$\bar{r}_{ij} : B_{ij} \longrightarrow C_{ij},$$

donde B_{ij} es la celda ij de B_2 y C_{ij} es la celda ij de Ω .

Como ya tenemos el mapeo $\bar{r}(s, t)$ bilineal de B_2 en el cuadrilátero $PQRS$, entonces, podemos obtener el mapeo \bar{r}_{ij} primero *escalando* la celda B_{ij} a B_2 ,

$$a_{ij} : B_{ij} \longrightarrow B_2$$

y luego aplicar el mapeo $\bar{r}(s, t)$; donde el cuadrilátero $PQRS$ es la celda C_{ij} , es decir, (vease la figura siguiente).

$$P = P_{ij}; \quad Q = P_{i+1,j}, \quad R = P_{i+1,j+1}; \quad S = P_{i,j+1}$$

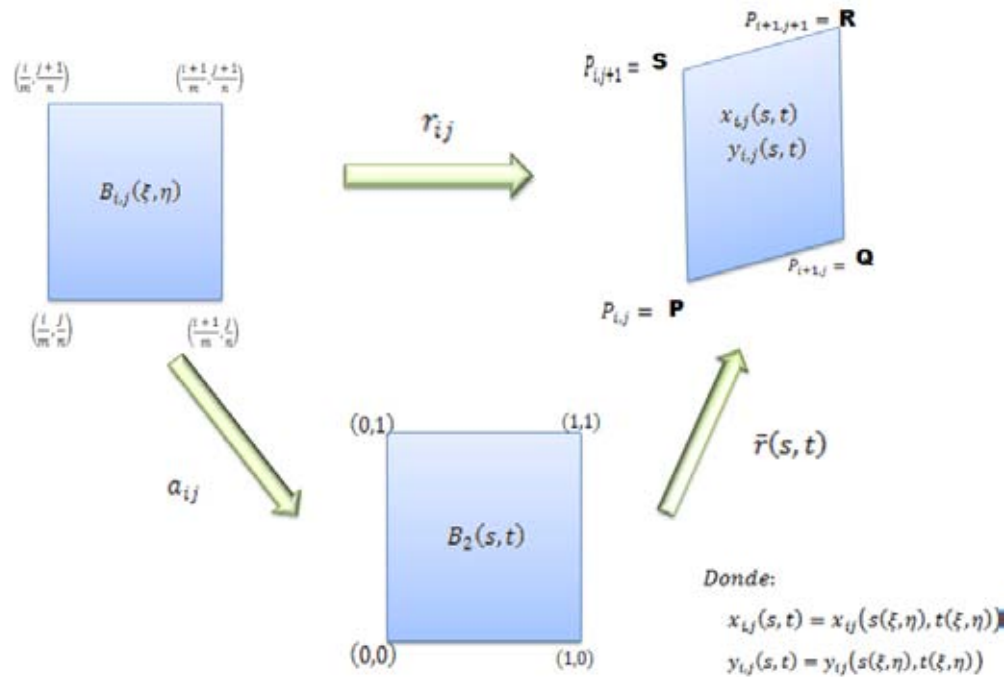


Figura 3.26 Mapeo de la celda B_{ij} a B_2 y aplicación del mapeo $\bar{r}(s, t)$.

esto es,

$$\bar{r}(s, t) = P(1-s)(1-t) + St(1-s) + Qs(1-t) + Rst.$$

Entonces tenemos:

$$\begin{aligned} \bar{r}(0, 0) &= P = P_{ij} \\ \bar{r}(1, 0) &= Q = P_{i+1,j} \\ \bar{r}(1, 1) &= R = P_{i+1,j+1} \\ \bar{r}(0, 1) &= S = P_{i,j+1} \end{aligned}$$

$$\bar{r}_s(s, t) = Rt - P(1 - t) - St + Q(1 - t)$$

$$\bar{r}_t(s, t) = Rs - P(1 - s) + S(1 - s) - Qs$$

$$\begin{aligned}\bar{r}_s(0, 0) &= Q - P = P_{i+1, j} - P_{ij} \\ \bar{r}_s(1, 0) &= Q - P = P_{i+1, j} - P_{ij} \\ \bar{r}_s(1, 1) &= R - S = P_{i+1, j+1} - P_{i, j+1} \\ \bar{r}_s(0, 1) &= R - S = P_{i+1, j+1} - P_{i, j+1}\end{aligned}$$

$$\begin{aligned}\bar{r}_t(0, 0) &= S - P = P_{i, j+1} - P_{ij} \\ \bar{r}_t(1, 0) &= S - P = P_{i, j+1} - P_{ij} \\ \bar{r}_t(1, 1) &= R - Q = P_{i+1, j+1} - P_{i+1, j} \\ \bar{r}_t(0, 1) &= R - Q = P_{i+1, j+1} - P_{i+1, j}\end{aligned}$$

La matriz Jacobiana esta dada por

$$\bar{r}'(s, t) = (\bar{r}_s(s, t) \mid \bar{r}_t(s, t)),$$

y entonces

$$\bar{r}'(0, 0) = [Q - P \mid S - P]$$

$$J(0, 0) = \det \bar{r}'(0, 0) = \det [Q - P \mid S - P] = 2\text{area}\Delta(P, Q, S)$$

$$\bar{r}'(1, 0) = [Q - P \mid R - Q]$$

$$J(1, 0) = \det \bar{r}'(1, 0) = \det [R - Q \mid P - Q] = 2\text{area}\Delta(Q, R, P)$$

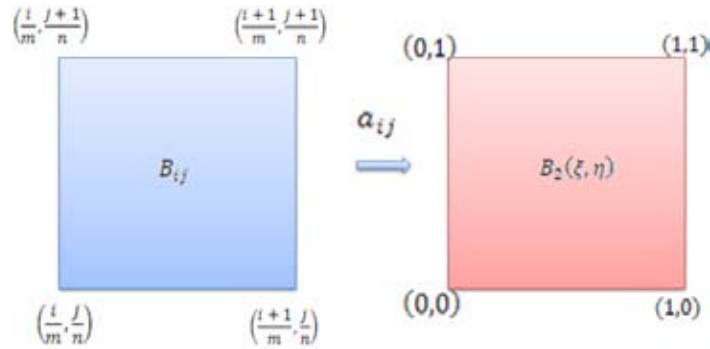
$$\bar{r}'(1, 1) = [R - S \mid R - Q]$$

$$J(1, 1) = \det \bar{r}'(1, 1) = \det [R - Q \mid P - Q] = 2\text{area}\Delta(R, S, Q)$$

$$\bar{r}'(0, 1) = [R - S \mid S - P]$$

$$J(0, 1) = \det \bar{r}'(0, 1) = \det [P - S \mid R - S] = 2\text{area}\Delta(S, P, R).$$

Pasemos ahora al mapeo a_{ij} .

Figura 3.27 Mapeo a_{ij} .

$$a_{ij}(\xi, \eta) = \begin{pmatrix} m \left(\xi - \frac{i}{m} \right) \\ n \left(\eta - \frac{j}{n} \right) \end{pmatrix}$$

La matriz Jacobiana es

$$a'_{ij}(\xi, \eta) = \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix} = a'_{ij}$$

La primera columna es:

$$\frac{\partial a_{ij}}{\partial \xi} = \overline{a'_{ijs}} = \begin{pmatrix} m \\ 0 \end{pmatrix}$$

La segunda columna es:

$$\frac{\partial a_{ij}}{\partial \eta} = \overline{a'_{ijt}} = \begin{pmatrix} 0 \\ n \end{pmatrix}$$

Entonces el mapeo \bar{r}_{ij} que queremos está dado por

$$\bar{r}_{ij} = \bar{r}(a_{ij}(\xi, \eta))$$

y entonces

$$J_{ij} = \det \left(a'_{ij} \right) = mn$$

$$\begin{aligned} r'_{ij} &= r' (a_{ij} (\xi, \eta)) a'_{ij} \\ &= r' (a_{ij}) \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix} \\ &= r' (a_{ij} (\xi, \eta)) \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix} \end{aligned}$$

$$r'_{ij} (\xi, \eta) = [\bar{r}_{ij\xi} \mid \bar{r}_{ij\eta}] = [\bar{r}_{ij\xi} (\xi, \eta) \mid \bar{r}_{ij\eta} (\xi, \eta)]$$

$$\frac{\partial r_{ij}}{\partial \xi} (\xi, \eta) = \frac{\partial}{\partial \xi} r (a_{ij} (\xi, \eta))$$

$$\frac{\partial r_{ij}}{\partial \xi} (\xi, \eta) = r' (a_{ij} (\xi, \eta)) \cdot a'_{ij} (\xi, \eta) \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Recordando que

$$\begin{aligned} a_{ij} \left(\frac{i}{m}, \frac{j}{n} \right) &= (0, 0) \\ a_{ij} \left(\frac{i-1}{m}, \frac{j}{n} \right) &= (1, 0) \\ a_{ij} \left(\frac{i+1}{m}, \frac{j+1}{n} \right) &= (1, 1) \\ a_{ij} \left(\frac{i}{m}, \frac{j+1}{n} \right) &= (0, 1) \end{aligned}$$

entonces

$$\begin{aligned} \frac{\partial r_{ij}}{\partial \xi} \left(\frac{i}{m}, \frac{j}{n} \right) &= r' (0, 0) \cdot a'_{ij} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= \left[\frac{\partial r}{\partial s} (0, 0) \mid \frac{\partial r}{\partial t} (0, 0) \right] \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= m \frac{\partial r}{\partial s} (0, 0) \\ &= (Q - P) m = (P_{i+1, j} - P_{i, j}) m \end{aligned}$$

Análogamente

$$\begin{aligned}\frac{\partial r_{ij}}{\partial \xi} \left(\frac{i+1}{m}, \frac{j}{n} \right) &= (Q - P)m = (P_{i+1,j} - P_{i,j})m \\ \frac{\partial r_{ij}}{\partial \xi} \left(\frac{i+1}{m}, \frac{j+1}{n} \right) &= (R - S)m = (P_{i+1,j+1} - P_{i,j+1})m \\ \frac{\partial r_{ij}}{\partial \xi} \left(\frac{i}{m}, \frac{j+1}{n} \right) &= (R - S)m = (P_{i+1,j+1} - P_{i,j+1})m\end{aligned}$$

$$\begin{aligned}\frac{\partial r_{ij}}{\partial \eta} (\xi, \eta) &= \frac{\partial}{\partial \eta} r(a_{ij}(\xi, \eta)) \\ \frac{\partial r_{ij}}{\partial \eta} (\xi, \eta) &= r'(a_{ij}(\xi, \eta)) \cdot a'_{ij}(\xi, \eta) \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix}.\end{aligned}$$

Entonces tenemos

$$\begin{aligned}\frac{\partial r_{ij}}{\partial \eta} \left(\frac{i}{m}, \frac{j}{n} \right) &= n \frac{\partial r}{\partial t} (0, 0) \\ &= n(S - P) = n(P_{i+1,j} - P_{i,j})\end{aligned}$$

de manera semejante,

$$\begin{aligned}\frac{\partial r_{ij}}{\partial \eta} \left(\frac{i+1}{m}, \frac{j}{n} \right) &= n(S - P) = n(P_{i+1,j} - P_{i,j}) \\ \frac{\partial r_{ij}}{\partial \xi} \left(\frac{i+1}{m}, \frac{j+1}{n} \right) &= n(R - Q) = n(P_{i+1,j+1} - P_{i+1,j}) \\ \frac{\partial r_{ij}}{\partial \xi} \left(\frac{i}{m}, \frac{j+1}{n} \right) &= n(R - Q) = n(P_{i+1,j+1} - P_{i+1,j}).\end{aligned}$$

Pasemos ahora a la integral

$$\int \int_{B_{ij}} f(r_{ij\xi}, r_{ij\eta}) d\xi d\eta = \frac{1}{mn} \int \int_{B_2} f(r_s, r_t) ds dt$$

3.3.1. Discretización de la Funcional de Longitud

Aplicaremos ahora el proceso anterior a la funcional de longitud

$$\begin{aligned} \mathcal{I}_l(\bar{x}) &= \int \int_{B_2} \left(\|\bar{x}_\xi\|^2 + \|\bar{x}_\eta\|^2 \right) d\xi d\eta \\ &= \sum_{i=1}^{n-1} \sum_{j=1}^{m-1} \int \int_{B_{ij}} \left(\|\bar{x}_\xi\|^2 + \|\bar{x}_\eta\|^2 \right) d\xi d\eta. \end{aligned}$$

Si ahora aplicamos la aproximación sobre B_{ij} donde

$$\bar{x} \approx \bar{r}_{ij}$$

entonces tenemos que

$$\int \int_{B_{ij}} \left(\|\bar{x}_\xi\|^2 + \|\bar{x}_\eta\|^2 \right) d\xi d\eta \approx \int \int_{B_{ij}} \left(\|r_{ij\xi}\|^2 + \|r_{ij\eta}\|^2 \right) d\xi d\eta;$$

si efectuamos ahora el cambio de variables

$$\bar{r}_{ij} = (a_{ij}(s, t))$$

tenemos

$$\int \int_{B_{ij}} \left(\|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2 \right) ds dt = \frac{1}{mn} \int \int_{B_2} \left\| \bar{r}'(a_{ij}) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| \bar{r}'(a_{ij}) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2 ds dt.$$

Aproximando por la regla de cuadratura de los 4 puntos

$$\begin{aligned} \approx & \frac{1}{4mn} \left[\left\| \bar{r}'(0,0) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| \bar{r}'(0,0) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2 + \right. \\ & \left\| \bar{r}'(1,0) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| \bar{r}'(1,0) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2 + \\ & \left\| \bar{r}'(1,1) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| \bar{r}'(1,1) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2 + \\ & \left. \left\| \bar{r}'(0,1) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| \bar{r}'(0,1) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2 \right], \end{aligned}$$

donde

$$\begin{aligned}
&= \frac{1}{4mn} [\|Q - P\|^2 m^2 + \|S - P\|^2 n^2 + \\
&\quad \|Q - P\|^2 m^2 + \|R - Q\|^2 n^2 + \\
&\quad \|R - S\|^2 m^2 + \|R - Q\|^2 n^2 + \\
&\quad \|R - S\|^2 m^2 + \|S - P\|^2 n^2]
\end{aligned}$$

$$\begin{aligned}
\int \int_{B_{ij}} (\|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2) d\xi d\eta &= \frac{1}{4mn} [2m^2 [\|Q - P\|^2 + \|R - S\|^2] + \\
&\quad 2n^2 [\|S - P\|^2 + \|R - Q\|^2]] \\
&= \frac{1}{2mn} [m^2 [\|Q - P\|^2 + \|R - S\|^2] + \\
&\quad n^2 [\|S - P\|^2 + \|R - Q\|^2]]
\end{aligned}$$

donde

$$\begin{aligned}
P &= P_{i,j} & R &= P_{i+1,j+1} \\
Q &= P_{i+1,j} & S &= P_{i,j+1}.
\end{aligned}$$

Sustituyendo P, Q, R, S por sus valores

$$\begin{aligned}
\sum_{i,j=1}^{m-1,n-1} \int \int_{B_{ij}} (\|x_\xi\|^2 + \|x_\eta\|^2) d\xi d\eta &\approx \frac{1}{2mn} \sum_{i,j=1}^{m-1,n-1} [m^2 [\|P_{i+1,j} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i,j+1}\|^2] + \\
&\quad n^2 [\|P_{i,j+1} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i+1,j}\|^2]]
\end{aligned}$$

definimos la funcional discreta de longitud como

$$\begin{aligned}
I_L(G) &= \frac{1}{2mn} \sum_{i,j=1}^{m-1,n-1} [m^2 [\|P_{i+1,j} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i,j+1}\|^2] + \\
&\quad n^2 [\|P_{i,j+1} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i+1,j}\|^2]]
\end{aligned}$$

Fijemos ahora el primer término, así obtenemos que este es la suma de todos los segmentos horizontales de la malla por lo que es natural definir

$$S_H(G) = \sum_{i,j=1}^{m-1,n-1} [\|P_{i+1,j} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i,j+1}\|^2]$$

efectuamos un proceso análogo para el segundo término que representa los segmentos verticales.

$$S_v(G) = \sum_{i,j=1}^{m-1,n-1} [\|P_{i,j+1} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i,j+1}\|^2]$$

Así que podemos escribir la funcional como

$$I_L(G) = \frac{1}{2mn} (m^2 S_V(G) + n^2 S_H(G)).$$

3.3.2. Discretización de la Funcional de Área

$$\begin{aligned} I_a(\bar{x}) &= \int \int_{B_2} J^2(\xi, \eta) d\xi d\eta \\ &= \int \int_{B_2} (x_\xi y_\eta - x_\eta y_\xi)^2 d\xi d\eta \\ &= \int \int_{B_2} (\bar{x}_\xi^t J_2 \bar{x}_\eta)^2 d\xi d\eta, \end{aligned}$$

donde

$$\bar{x}_\xi^t = (x_\xi, y_\xi), \quad J_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad \bar{x}_\eta = \begin{pmatrix} x_\eta \\ y_\eta \end{pmatrix},$$

entonces

$$I_a(x) = \sum_{i,j} \int \int_{B_{ij}} (\bar{x}_\xi^t J_2 \bar{x}_\eta)^2 d\xi d\eta;$$

aproximando \bar{x} sobre B_{ij} por \bar{r}_{ij}

$$\int \int_{B_{ij}} (\bar{x}_\xi^t J_2 \bar{x}_\eta)^2 d\xi d\eta \approx \int \int_{B_{ij}} (\bar{r}_{ij\xi}^t J_2 \bar{r}_{ij\eta})^2 d\xi d\eta$$

y utilizando el cambio de variables

$$r_{ij} = \bar{r}(a_{ij}(s, t))$$

entonces obtenemos

$$\int \int_{B_{ij}} (\bar{r}_{ij\xi}^t J_2 \bar{r}_{ij\eta})^2 d\xi d\eta = \frac{1}{mn} \int \int_{B_2} \left[\left(r' (a_{ij}) \begin{pmatrix} m \\ 0 \end{pmatrix} \right)^t J_2 \left(r' (a_{ij}) \begin{pmatrix} 0 \\ n \end{pmatrix} \right) \right]^2 ds dt.$$

Si ahora aproximamos por la regla de cuadratura de los cuatro puntos tendremos que

$$\begin{aligned} \int \int_{B_{ij}} \|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2 d\xi d\eta &\approx \frac{1}{4mn} \left[\left(\bar{r}' (0, 0) m e_1 \right)^t J_2 \left(\bar{r}' (0, 0) n e_2 \right) \right]^2 + \\ &\quad \left[\left(\bar{r}' (1, 0) m e_1 \right)^t J_2 \left(\bar{r}' (1, 0) n e_2 \right) \right]^2 + \\ &\quad \left[\left(\bar{r}' (1, 1) m e_1 \right)^t J_2 \left(\bar{r}' (1, 1) n e_2 \right) \right]^2 + \\ &\quad \left[\left(\bar{r}' (0, 1) m e_1 \right)^t J_2 \left(\bar{r}' (0, 1) n e_2 \right) \right]^2] \\ &= \frac{1}{4mn} [((Q - P) J_2 (S - P))^2 + \\ &\quad ((Q - P)^2 J_2 (R - Q))^2 + \\ &\quad ((R - S)^2 J_2 (R - Q))^2 + \\ &\quad ((R - S)^2 J_2 (S - P))^2] \cdot m^2 n^2 \end{aligned}$$

donde

$$\begin{aligned} P &= P_{i,j} & R &= P_{i+1,j+1} \\ Q &= P_{i+1,j} & S &= P_{i,j+1} \end{aligned}$$

$$\begin{aligned} &= \frac{m^2 n^2}{4mn} [((Q - P) J_2 (S - P))^2 + \\ &\quad ((Q - P)^2 J_2 (R - Q))^2 + \\ &\quad ((R - S)^2 J_2 (R - Q))^2 + \\ &\quad ((R - S)^2 J_2 (S - P))^2] \end{aligned}$$

$$\int \int_{B_{ij}} (\bar{r}_{ij\xi}^t J_2 \bar{r}_{ij\eta})^2 d\xi d\eta \approx \frac{mn}{4} [\text{area}(P_{i,j}, P_{i+1,j}, P_{i,j+1})^2 + \text{area}(P_{i+1,j}, P_{i+1,j+1}, P_{i,j})^2 + \text{area}(P_{i+1,j+1}, P_{i,j+1}, P_{i+1,j})^2 + \text{area}(P_{i,j+1}, P_{i,j}, P_{i+1,j+1})^2]$$

Si nombramos los triángulos como

$$\begin{aligned} T_{ij}^1 &= \Delta P_{i,j}, P_{i+1,j}, P_{i,j+1} \\ T_{ij}^2 &= \Delta P_{i+1,j}, P_{i+1,j+1}, P_{i,j} \\ T_{ij}^3 &= \Delta P_{i+1,j+1}, P_{i,j+1}, P_{i+1,j} \\ T_{ij}^4 &= \Delta P_{i,j+1}, P_{i,j}, P_{i+1,j+1}, \end{aligned}$$

Podemos definir la funcional de área discreta

$$I_A(G) = \sum_{i,j=1}^{m-1,n-1} \sum_{k=1}^4 \text{area}(T_{ij}^k)^2;$$

en la práctica usaremos $\alpha(T_{ij}^k) = 2\text{area}(T_{ij}^k)$ y entonces

$$I_A(G) = \sum_{q=1}^N \alpha(T_q)^2$$

donde N es el número total de triángulos de G .

3.3.3. Discretización de la Funcional de Ortogonalidad

$$\begin{aligned} I_O &= \int \int_{B_2} (\bar{x}_\xi \cdot \bar{x}_\eta) d\xi d\eta \\ &= \sum_{i,j=1}^{m-1,n-1} \int \int_{B_{ij}} (\bar{x}_\xi \cdot \bar{x}_\eta) d\xi d\eta \end{aligned}$$

Si aproximamos $\bar{x}(\xi, \eta)$ por $\bar{r}_{ij}(s, t)$ en B_{ij} tenemos

$$I_O \approx \sum_{i,j=1}^{m-1,n-1} \int \int_{B_{ij}} (\bar{r}_{ij\xi} \cdot \bar{r}_{ij\eta})^2 d\xi d\eta$$

Utilizando el cambio de variable

$$\int_{B_{ij}} \int (\bar{r}_{ij\xi} \cdot \bar{r}_{ij\eta})^2 d\xi d\eta = \frac{1}{mn} \int_{B_2} \int [(\bar{r}(a_{ij}) me_1) \cdot (\bar{r}(a_{ij}) ne_2)]^2 ds dt,$$

aproximando por la regla de cuadratura

$$\begin{aligned} \frac{1}{mn} \int_{B_2} \int [(\bar{r}(a_{ij}) me_1) \cdot (\bar{r}(a_{ij}) ne_2)]^2 ds dt &\approx \frac{1}{4mn} \left[\left(\bar{r}'(0,0) me_1 \right) \cdot \left(\bar{r}'(0,0) ne_2 \right) \right]^2 \\ &\quad \left[\left(\bar{r}'(1,0) me_1 \right) \cdot \left(\bar{r}'(1,0) ne_2 \right) \right]^2 + \\ &\quad \left[\left(\bar{r}'(1,1) me_1 \right) \cdot \left(\bar{r}'(1,1) ne_2 \right) \right]^2 + \\ &\quad \left[\left(\bar{r}'(0,1) me_1 \right) \cdot \left(\bar{r}'(0,1) ne_2 \right) \right]^2 \Big] \\ &= \frac{mn}{4} \left[((Q - P)^t \cdot (S - P))^2 + \right. \\ &\quad ((P - Q)^t \cdot (R - Q))^2 + \\ &\quad ((S - R)^t \cdot (Q - R))^2 + \\ &\quad \left. ((R - S)^t \cdot (P - S))^2 \right]. \end{aligned}$$

Si ahora utilizamos

$$\begin{aligned} P &= P_{i,j} & R &= P_{i+1,j+1} \\ Q &= P_{i+1,j} & S &= P_{i,j+1}, \end{aligned}$$

para aproximar la funcional de ortogonalidad por los nodos interiores de la malla tendremos

$$\begin{aligned} &= \frac{mn}{4} \left[((P_{i+1,j} - P_{i,j})^t \cdot (P_{i,j+1} - P_{i,j}))^2 + \right. \\ &\quad ((P_{i,j} - P_{i+1,j})^t \cdot (P_{i+1,j+1} - P_{i+1,j}))^2 + \\ &\quad ((P_{i,j+1} - P_{i+1,j+1})^t \cdot (P_{i+1,j} - P_{i+1,j+1}))^2 + \\ &\quad \left. ((P_{i+1,j+1} - P_{i,j+1})^t \cdot (P_{i,j} - P_{i,j+1}))^2 \right] = \frac{mn}{4} \theta(c_{ij}) \end{aligned}$$

y entonces definimos la funcional discreta de ortogonalidad como

$$I_O(G) = \sum_{i,j=1}^{m-1,n-1} \theta(c_{ij})$$

donde

$$\begin{aligned} \theta(c_{ij}) = & [((P_{i+1,j} - P_{i,j})^t \cdot (P_{i,j+1} - P_{i,j}))^2 + \\ & ((P_{i,j} - P_{i+1,j})^t \cdot (P_{i+1,j+1} - P_{i+1,j}))^2 + \\ & ((P_{i,j+1} - P_{i+1,j+1})^t \cdot (P_{i+1,j} - P_{i+1,j+1}))^2 + \\ & ((P_{i+1,j+1} - P_{i,j+1})^t \cdot (P_{i,j} - P_{i,j+1}))^2] . \end{aligned}$$

3.3.4. Discretización de la Funcional de Suavidad

$$I_S(\bar{x}) = \int \int_{B_2} \frac{(\bar{x}_\xi^2 + \bar{x}_\eta^2 + \bar{y}_\xi^2 + \bar{y}_\eta^2)}{J} d\xi d\eta = \sum_{i,j} \int \int_{B_{ij}} \frac{\|\bar{x}_\xi\|^2 + \|\bar{x}_\eta\|^2}{\bar{x}_\xi^t J_2 \bar{x}_\eta} d\xi d\eta$$

Si ahora aproximamos $\bar{x} \approx \bar{r}_{ij}$ en B_{ij}

$$\int \int_{B_{ij}} \frac{\|\bar{x}_\xi\|^2 + \|\bar{x}_\eta\|^2}{\bar{x}_\xi^t J_2 \bar{x}_\eta} d\xi d\eta \approx \int \int_{B_{ij}} \frac{\|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2}{\bar{r}_{ij\xi}^t J_2 \bar{r}_{ij\eta}} d\xi d\eta,$$

donde

$$J_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

$$I_S \approx \sum_{i,j=1}^{m-1,n-1} \int \int_{B_{ij}} \frac{\|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2}{\bar{r}_{ij\xi}^t J_2 \bar{r}_{ij\eta}} d\xi d\eta$$

haciendo el cambio de variable

$$\frac{1}{mn} \int \int_{B_2} \frac{\left\| r'(a_{ij}) \begin{pmatrix} m \\ 0 \end{pmatrix} \right\|^2 + \left\| r'(a_{ij}) \begin{pmatrix} 0 \\ n \end{pmatrix} \right\|^2}{\left(r'(a_{ij}) \begin{pmatrix} m \\ 0 \end{pmatrix} \right)^t J_2 \left(r'(a_{ij}) \begin{pmatrix} 0 \\ n \end{pmatrix} \right)} ds dt$$

$$\begin{aligned}
&\approx \frac{1}{4mn} \left[\frac{m^2 \|r'(0,0)me_1\|^2 + n^2 \|r'(0,0)ne_2\|^2}{mn \left((r'(0,0)me_1)^t J_2(r'(0,0)ne_2) \right)} \right. \\
&\quad + \frac{m^2 \|r'(1,0)me_1\|^2 + n^2 \|r'(1,0)ne_2\|^2}{mn \left((r'(1,0)me_1)^t J_2(r'(1,0)ne_2) \right)} \\
&\quad + \frac{m^2 \|r'(1,1)me_1\|^2 + n^2 \|r'(1,1)ne_2\|^2}{mn \left((r'(1,1)me_1)^t J_2(r'(1,1)ne_2) \right)} \\
&\quad \left. + \frac{m^2 \|r'(0,1)me_1\|^2 + n^2 \|r'(0,1)ne_2\|^2}{mn \left((r'(0,1)me_1)^t J_2(r'(0,1)ne_2) \right)} \right].
\end{aligned}$$

Sustituyendo

$$\begin{aligned}
P &= P_{i,j} & R &= P_{i+1,j+1} \\
Q &= P_{i+1,j} & S &= P_{i,j+1}
\end{aligned}$$

tenemos

$$\begin{aligned}
\int \int_{B_{ij}} \frac{\|\bar{r}_{ij\xi}\|^2 + \|\bar{r}_{ij\eta}\|^2}{r_{ij\xi} J_2 r_{ij\eta}} d\xi d\eta &\approx \frac{1}{4mn} \left[\frac{m^2 \|Q - P\|^2 + n^2 \|S - P\|^2}{mn (Q - P)^t J_2 (S - P)} \right. \\
&\quad + \frac{m^2 \|Q - P\|^2 + n^2 \|R - Q\|^2}{mn (P - Q)^t J_2 (R - Q)} \\
&\quad + \frac{m^2 \|R - S\|^2 + n^2 \|R - Q\|^2}{mn (S - R)^t J_2 (Q - R)} \\
&\quad \left. + \frac{m^2 \|R - S\|^2 + n^2 \|S - P\|^2}{mn (R - S)^t J_2 (P - S)} \right] \\
&\approx \frac{1}{4mn} \left[\frac{\frac{m}{n} \|Q - P\|^2 + \frac{n}{m} \|S - P\|^2}{(Q - P)^t J_2 (S - P)} + \frac{\frac{m}{n} \|Q - P\|^2 + \frac{n}{m} \|R - Q\|^2}{(P - Q)^t J_2 (R - Q)} \right. \\
&\quad \left. + \frac{\frac{m}{n} \|R - S\|^2 + \frac{n}{m} \|R - Q\|^2}{(S - R)^t J_2 (Q - R)} + \frac{\frac{m}{n} \|R - S\|^2 + \frac{n}{m} \|S - P\|^2}{(R - S)^t J_2 (P - S)} \right],
\end{aligned}$$

sustituyendo P, Q, R, S por sus valores y si tomamos el caso particular en el que $m = n$ tendremos que

$$\begin{aligned}
\int \int_{B_{ij}} \frac{\|r_{ij\xi}\|^2 + \|r_{ij\eta}\|^2}{r_{ij\xi} J_2 r_{ij\eta}} d\xi d\eta &\approx \sum \frac{1}{4mn} \left[\frac{\|P_{i+1,j} - P_{i,j}\|^2 + \|P_{i,j+1} - P_{i,j}\|^2}{(P_{i+1,j} - P_{i,j})^t J_2 (P_{i,j+1} - P_{i,j})} \right. \\
&+ \frac{\|P_{i+1,j} - P_{i,j}\|^2 + \|P_{i+1,j+1} - P_{i+1,j}\|^2}{(P_{i,j} - P_{i+1,j})^t J_2 (P_{i+1,j+1} - P_{i+1,j})} \\
&+ \frac{\|P_{i+1,j+1} - P_{i,j+1}\|^2 + \|P_{i+1,j+1} - P_{i+1,j}\|^2}{(P_{i,j+1} - P_{i+1,j+1})^t J_2 (P_{i+1,j} - P_{i+1,j+1})} \\
&\left. + \frac{\|P_{i+1,j+1} - P_{i,j+1}\|^2 + \|P_{i,j+1} - P_{i,j}\|^2}{(P_{i+1,j+1} - P_{i,j+1})^t J_2 (P_{i,j} - P_{i,j+1})} \right].
\end{aligned}$$

Como podemos observar la funcional de suavidad se aproxima por los nodos interiores de la malla.

Definimos

$$\begin{aligned}
I_s(G) &= \frac{1}{4} \sum_{i,j} \left[\frac{\frac{m}{n} \|P_{i+1,j} - P_{i,j}\|^2 + \frac{n}{m} \|P_{i,j+1} - P_{i,j}\|^2}{(P_{i+1,j} - P_{i,j})^t J_2 (P_{i,j+1} - P_{i,j})} \right. \\
&+ \frac{\frac{m}{n} \|P_{i+1,j} - P_{i,j}\|^2 + \frac{n}{m} \|P_{i+1,j+1} - P_{i+1,j}\|^2}{(P_{i,j} - P_{i+1,j})^t J_2 (P_{i+1,j+1} - P_{i+1,j})} \\
&+ \frac{\frac{m}{n} \|P_{i+1,j+1} - P_{i,j+1}\|^2 + \frac{n}{m} \|P_{i+1,j+1} - P_{i+1,j}\|^2}{(P_{i,j+1} - P_{i+1,j+1})^t J_2 (P_{i+1,j} - P_{i+1,j+1})} \\
&\left. + \frac{\frac{m}{n} \|P_{i+1,j+1} - P_{i,j+1}\|^2 + \frac{n}{m} \|P_{i,j+1} - P_{i,j}\|^2}{(P_{i+1,j+1} - P_{i,j+1})^t J_2 (P_{i,j} - P_{i,j+1})} \right].
\end{aligned}$$

Si Δ_q es un triángulo genérico de G , entonces definimos

$$\Psi(\Delta_q) = \frac{\frac{m}{n} \|\bar{a}_q\|^2 + \frac{n}{m} \|\bar{b}_q\|^2}{a_q^t J_2 b_q} = I_s(G) = \sum_{q=1}^N \Psi(\Delta_q).$$

de otra manera:

Dado un triángulo Δ_q , consideramos las funciones

$$\begin{aligned}
l(\Delta_q) &= \|\bar{a}_q\|^2 + \|\bar{b}_q\|^2 \\
\alpha(\Delta_q) &= 2\text{area}(\Delta_q) = a_q^t J_2 b_q
\end{aligned}$$

donde

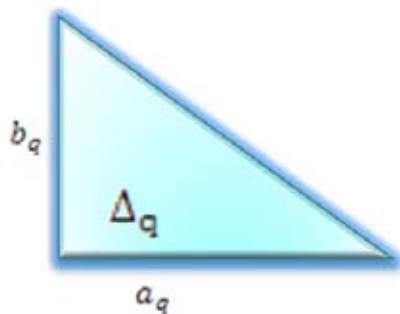


Figura 3.28 Triángulo Δ_q .

a_q , b_q son los lados del triángulo que forman parte de la malla y $\alpha(\Delta_q)$ es el doble de la área orientada. Escribiremos la funcional armónica como

$$H(G) = \sum_{q=1}^N \frac{l(\Delta_q)}{\alpha(\Delta_q)}.$$

3.4. Ejemplos De Funcionales Clásicas Discretas

En esta sección se presentan algunos ejemplos de los funcionales clásicos como son suavidad, área y ortogonalidad¹.

Ejemplos De la funcional de área

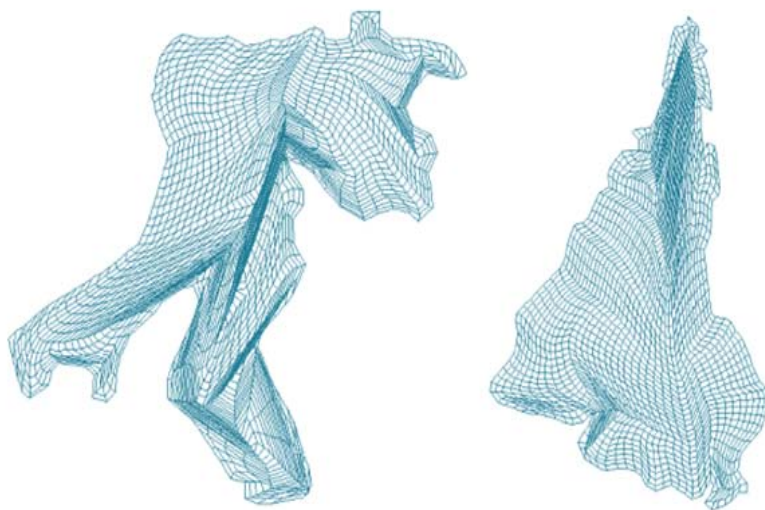


Figura 3.29 Mallas óptimas generadas con la funcional de área discreta.

¹Para la generación de estos ejemplos se utilizó Software creado por el grupo UNAMALLA.

Ejemplos De la funcional de suavidad.

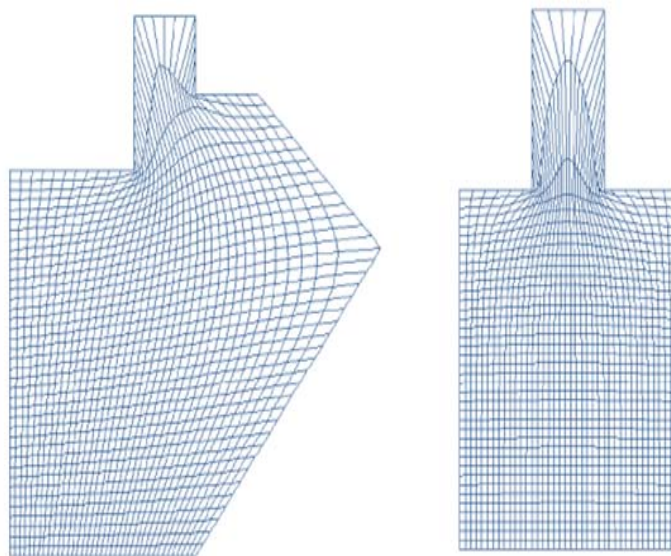


Figura 3.30 Mallas óptimas generadas con la funcional de Suavidad discreta.

Capítulo 4

Funcionales Cuasi-Armónicas y Épsilon-Convexidad

En este capítulo consideramos el problema de generar mallas armónicas usando funcionales que sean muy sencillas de implementar y que produzcan resultados confiables. Además, se introducirá el concepto de ϵ -convexidad que permitirá eliminar la inestabilidad numérica que se presenta al generar mallas convexas con la condición de que el área de los triángulos sea positiva para todos los triángulos de la malla, es decir, que el problema no esté bien planteado.

4.1. Generación numérica de mallas

El procedimiento usual para obtener una malla será:

- Construir una malla inicial G_0 , por algún procedimiento algebraico.
- Seleccionar una funcional discreta, que generalmente será una combinación de la forma

$$F = \alpha_1 F_1 + \alpha_2 F_2$$

donde

$$\alpha_1 + \alpha_2 = 1$$

y F_1, F_2 son dos funcionales que reflejan las propiedades geométricas deseadas y α_1 y α_2 son los pesos que les asignaremos en base a las propiedades anteriores y a la dificultad del problema.

- Posteriormente se resuelve el problema de optimización

$$\min_G F(G) = F(G^*)$$

donde G^* es la malla generada con las propiedades deseadas.

Con frecuencia la suavidad y ortogonalidad de la malla óptima son las propiedades más requeridas tomando en cuenta que la convexidad de la malla es esencial.

4.2. ϵ -Convexidad

Un indicador de que la condición para la convexidad de la malla era inadecuada, es que sobre algunas regiones muy irregulares se obtenían mallas convexas, que al guardarlas en un archivo se modificaban un poco debido al redondeo y al leerlas después no resultaban convexas.

Una primera idea para corregir este problema sería elegir un número pequeño $\epsilon > 0$ y cambiar la condición

$$\alpha(\Delta_q) > 0 \quad \forall \Delta_q \in G^*.$$

por

$$\alpha(\Delta_q) > \epsilon \quad \forall \Delta_q \in G^*.$$

sin embargo, el problema es cómo elegir la ϵ , ya que se ve que la escala juega un papel central y necesitamos que la selección de la ϵ sea robusta, es decir independiente de la escala.

Una clave para la elección de una ϵ adecuada, es la siguiente propiedad de las mallas.

Propiedad

Si

$$area(G) = \sum_{i,j=1}^{n-1,m-1} area(c_{ij})$$

y si G es una malla orientada, de Ω , es decir si las celdas c_{ij} de G tienen orientación inducida por la frontera de Ω entonces

$$area(G) = area(\Omega)$$

y si $\bar{\alpha}(G)$ es el promedio del área de los triángulos de las celdas de la malla, entonces

$$\bar{\alpha}(G) = \frac{area(\Omega)}{(m-1)(n-1)}.$$

$$area(c_{ij}) = area(T_{ij}^1) + area(T_{ij}^3) + area(T_{ij}^2) + area(T_{ij}^4)$$

$$2area(c_{ij}) = \sum_{k=1}^4 area(T_{ij}^k)$$

$$\Rightarrow area(G) = \frac{1}{2} \sum_{i,j=1}^{n-1,m-1} \sum_{k=1}^4 \alpha(T_{ij}^k)$$

$$= \frac{1}{2} \sum_{q=1}^N \alpha(T_q)$$

$$\sum_{q=1}^N \alpha(T_q) = 2area(\Omega)$$

$$\Rightarrow \bar{\alpha}(G) = \frac{1}{N} \sum_{q=1}^N \alpha(T_q)$$

$$= \frac{2area(\Omega)}{N}$$

$$\bar{\alpha}(G) = \frac{4}{4} \frac{area(\Omega)}{(m-1)(n-1)} = \frac{area(\Omega)}{(m-1)(n-1)}.$$

Corolario 12 $\bar{\alpha}(G)$ es independiente de G , sólo depende de m, n y por lo tanto escribimos

$$\bar{\alpha}(G) = \bar{\alpha}(\Omega, m, n) = \bar{\alpha}(\Omega)$$

porque m, n estarán fijas.

Propiedad

Si

$$A : R^2 \longrightarrow R^2$$

es lineal y no-singular, entonces si $\tilde{G} = A(G)$

$$\frac{\alpha(\tilde{\Delta}_i)}{\bar{\alpha}(\tilde{G})} = \frac{\alpha(\Delta_i)}{\bar{\alpha}(G)}$$

Demostración. Si $\tilde{\Delta}_i = A(\Delta_i)$ donde Δ_i es un triángulo de G y $\tilde{\Delta}_i$ es su imagen, entonces es fácil verificar que

$$\alpha(\tilde{\Delta}_i) = \det(A) \alpha(\Delta_i),$$

lo que implica

$$\bar{\alpha}(\tilde{G}) = \det(A) \bar{\alpha}(G).$$

Dividiendo las igualdades anteriores

$$\frac{\alpha(\tilde{\Delta}_i)}{\bar{\alpha}(\tilde{G})} = \frac{\alpha(\Delta_i)}{\bar{\alpha}(G)}$$

es decir, el cociente

$$\frac{\alpha(\Delta_i)}{\bar{\alpha}(G)} = \frac{\alpha(\Delta_i)}{\bar{\alpha}(\Omega)}$$

es independiente de la escala. ■

Definición 13 Una malla G de Ω es ϵ -convexa si

$$\frac{\alpha(\Delta_q)}{\bar{\alpha}(\Omega)} > \epsilon \quad \forall \Delta_q \in G.$$

La expresión anterior se puede escribir en forma compacta si introducimos

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

$$\alpha_+(G) = \max_{\Delta_q \in G} \alpha(\Delta_q)$$

Definición 14 Una malla G de Ω es ϵ -convexa si

$$\frac{\alpha_-(G)}{\bar{\alpha}(\Omega)} > \epsilon.$$

Como

$$\alpha_-(G) \leq \bar{\alpha}(\Omega) \leq \alpha_+(G)$$

entonces

$$\frac{\alpha_-(G)}{\bar{\alpha}(\Omega)} \leq 1.$$

lo que significa que para que existan mallas ϵ -convexas es necesario que $\epsilon < 1$.

Así que de ahora en adelante estaremos interesados en generar mallas ϵ -convexas, lo cual puede ser un problema muy difícil numéricamente si el valor

$$\text{máx} \left\{ \frac{\alpha_-(G)}{\bar{\alpha}(G)} \mid G; \text{ malla de } \Omega \right\}$$

es muy pequeño.

Definición 15

$$k(G) = \text{máx} \left\{ \frac{\alpha_-(G)}{\bar{\alpha}(G)} \mid G; \text{ malla de } \Omega \right\}$$

es el número de condición de la malla.

Si $k(G) \ll 1$, el problema de generar mallas ϵ -convexa es muy sensible a errores.

4.2.1. Funcionales Cuasi-Armónicas

Una malla \hat{G} de Ω se dice que es armónica si es un mínimo de la funcional armónica, es decir si

$$H(\hat{G}) = \min_G H(G)$$

donde

$$H(G) = \sum_{q=1}^N \frac{l(\Delta_q)}{\alpha(\Delta_q)}$$

dado que las mallas armónicas, son usualmente suaves, estamos interesados en calcular \hat{G} de manera económica.

El problema práctico que se presenta, al tratar de calcular el mínimo de la funcional armónica es que la funcional sólo tiene minimizadores sobre el conjunto de mallas G que sean convexas.

Sea

$$M_0(\Omega) = \{G \mid G \text{ es una malla de } m \times n \text{ de } \Omega \text{ convexa}\}.$$

Como vamos a calcular el mínimo de la funcional armónica por medio de un método iterativo, necesitamos un método que dado una malla inicial G_0 convexa, $G_0 \in M_0(\Omega)$, genere una sucesión G_1, G_2, \dots, G_n de mallas convexas que converjan a \tilde{G} . Un método de optimización que realizara el problema anterior no sería muy económico ya que sería un método de optimización con restricciones, $\alpha(\Delta_q) > 0 \forall q \in G$.

Además de que, el costo de generar G_0 inicial convexa puede ser muy alto si la región Ω es muy irregular.

En 1996, Tinoco construyó una familia de funcionales que dependen de un parámetro k ,

$$H_k(G) = \sum_{q=1}^N \frac{l(\Delta_q)}{k + \alpha(\Delta_q)},$$

y demostró que es posible obtener económicamente una malla cuasi armónica \tilde{G} a partir de cualquier malla inicial es decir

$$\tilde{G} = \min \{H_k(G) \mid \alpha_-(G) > 0 \text{ y } |k| \ll 1\}.$$

La idea es la siguiente dada una malla inicial G_0 elegir k_0 tal que

$$k_0 + \alpha(\Delta_q) > 0 \quad \forall \Delta_q \in G_0.$$

y calcular el mínimo G_1 de la funcional

$$H_{k_0}(G_1) = \min_{\alpha_-(G) + k_0 > 0} H_{k_0}(G)$$

Tinoco demostró que

$$\alpha_-(G_1) > \alpha_-(G_0),$$

así que elige $k_1 < k_0$, tal que

$$k_1 + \alpha_-(\tilde{G}_1) > 0.$$

empleando G_1 como malla inicial y se calcula el mínimo de la funcional

$$H_{k_1}(G_2) = \min_{\alpha_-(G)+k_1} H_{k_1}(G)$$

y demostró que este proceso, en un número finito de pasos, obtiene una G tal que

$$\alpha_-(G_n) > 0$$

y $|k_n| \ll 1$.

A continuación, veremos que es posible construir otra familia de funcionales que dependan de un parámetro de tal forma que al cabo de un número finito de optimizaciones sencillas, se obtenga una malla armónica. La idea básica de la familia de funcionales es cambiar el denominador $\frac{1}{\alpha(\Delta_q)}$ por una función parecida pero que penalice las áreas negativas y gradualmente penalice más cuando el parámetro se acerque a cero.

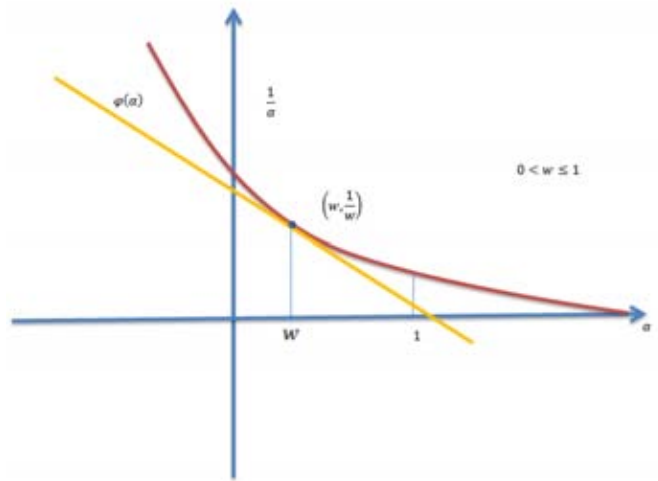


Figura 4.1 Gráfica de la función $\varphi_w(\alpha)$.

Sea $\varphi_0(\alpha)$ la función dada por

$$\varphi_w(\alpha) = \begin{cases} \frac{1}{\alpha} & \alpha \geq w \\ l_w(\alpha) & \alpha \leq w \end{cases}$$

donde $l_w(\alpha)$ es la recta que pasa por $\left(w, \frac{1}{w}\right)$ y tiene la misma tangente que ese punto es decir

$$\begin{aligned} l_w(\alpha) &= \frac{1}{w} + \left(-\frac{1}{w^2}\right)(\alpha - w) \\ &= \frac{1}{w} + \left(-\frac{\alpha}{w^2} + \frac{1}{w}\right) \\ &= \frac{2}{w} - \frac{\alpha}{w^2} = \frac{2w - \alpha}{w^2}, \end{aligned}$$

con esta familia $\varphi_w(\alpha)$ definiremos la familia de funcionales

$$F_w(G) = \sum_{q=1}^n l(\Delta_q) \varphi_w(\Delta_q).$$

A estas funcionales las llamaremos cuasi-ármonicas debido a la siguiente propiedad.

Propiedad

Si \widehat{G} es una malla armónica convexa, entonces es posible elegir $w > 0$ tal que la funcional $F_w(G)$ y $H(G)$ coincidan en \widehat{G} , es decir que \widehat{G} es una malla óptima de las dos

$$F_w(\widehat{G}) = H(\widehat{G})$$

Demostración.

$$F_w(G) = H(G) \quad \text{si} \quad \alpha_-(\widehat{G}) \geq w.$$

■

Algoritmo para generar mallas usando la funcional Cuasi-Armónica H_w

Dada una malla inicial G_0 elija
 $w_0 \leq 1$
 calcule $G_1 = \arg \min_G H_{w_0}(G)$
 con G_0 como condición inicial
 Si $\alpha_-(G_1) < 0$
 $G_0 \leftarrow G_1$
 $w_0 \leftarrow \frac{w_0}{2}$
 repita el proceso
 si $\alpha_-(G_1) > 0$, G_1 es una malla armónica.

4.3. Nuevo Funcional de Área

Como el problema de generación de mallas convexas es central en este trabajo, a continuación se presentan las nuevas funcionales que son más simples y más económicas.

Las funcionales discretas de área se originan en los trabajos de Castillo e Ivanenko. Sin embargo el enfoque de este último ha sido el más efectivo por lo que lo consideraremos a continuación. Siguiendo el procedimiento estructural para la construcción de funcionales discretas, la funcional de área queda como:

$$I_A(G) = \sum_{q=1}^n \alpha(\Delta_q)^2$$

pero esta funcional no genera celdas convexas sobre regiones irregulares, por lo que fue necesario hacer una modificación. Tinoco en su tesis doctoral y otros trabajos encontró que una funcional efectiva para la generación de mallas estructuradas convexas era la k - Área funcional que es totalmente similar a la de k - Suavidad, y es la siguiente

$$A_k(G) = \sum_{q=1}^n \frac{1}{k + \alpha(\Delta_q)}$$

donde $k > 0$ es un parámetro que se elige para penalizar las celdas de área negativa, la idea es la siguiente:

Dada G_0 una malla inicial elegir k_0 de tal forma que $k_0 + \alpha(\Delta_q) > 0$, para toda Δ_q de la malla, encontrar el mínimo de la funcional \widehat{G}_0 , $A_{k_0}(G)$ y usar este como nueva condición inicial, eligiendo $k_1 < k_0$ y repitiendo el proceso hasta que al cabo de un número finito de pasos se obtiene $k_n \leq 0$, esto quiere decir que hemos obtenido una malla convexa .

Recientemente Domínguez-Mota, en su trabajo de doctorado buscaba generar mallas convexas lo más ortogonales posibles, para ello observó que tenía que combinar la funcional de ortogonalidad con una funcional que garantice celdas convexas. La única conocida hasta entonces era la funcional de $k - Area$ de Tinoco, por lo que se planteó el problema de si esta era la única y qué propiedad tenía que le permitía generar mallas convexas. Domínguez-Mota encontró que una de las propiedades fundamentales es que la función

$$\frac{1}{k + \alpha}$$

para $\alpha \geq -k$ es convexa y monótona decreciente, por lo que propuso sustituirla por una función convexa y monótona decreciente para toda α , como por ejemplo $e^{-\alpha}$, eso es

$$A_e(G) = \sum_{q=1}^n e^{-\alpha(\Delta_q)}$$

y encontró que esta, en efecto, cuando se optimizaba a partir de una malla inicial G_0 , obtenía una malla óptima \widehat{G}_0 , con un número menor de celdas no convexas, pero que aún no era convexa completamente dado que esta función no tiene ningún parámetro, la idea que propuso entonces era recalculer el problema para que las celdas no convexas "sean más no convexas", si aplicamos ahora la funcional siguiente:

$$A_{2e}(G) = \sum_{q=1}^n e^{-2\alpha(\Delta_q)}$$

aplicamos ahora el proceso de optimización con malla inicial $G_1 = \widehat{G}_0$. Vemos que en efecto la nueva malla óptima \widehat{G}_1 es mucho mejor, Domínguez-Mota demuestra que si el problema tiene solución, entonces el proceso anterior, en un número finito de repeticiones, produce una malla óptima \widehat{G} convexa.

Finalmente Domínguez-Mota hace notar que es posible usar una función más económica de evaluar que la exponencial y propone la función

$$s_3(\alpha) = \begin{cases} \frac{1}{\alpha} & \alpha \geq 1 \\ q(\alpha) & \alpha \leq 1 \end{cases}$$

$$q(x) = 1 - (\alpha - 1) + (\alpha - 1)^2$$

obteniendo la funcional As_3

$$As_3(G) = \sum_{q=1}^N s_3(\alpha_q).$$

Dominguez-Mota demuestra que si existe una malla convexa G_0 de Ω , entonces para un valor suficientemente grande de t , la funcional

$$As_3(tG) = \sum_{q=1}^N s_3(t\alpha(\Delta_q)),$$

alcanza sus mínimos en mallas convexas.

**Algoritmo para generar mallas ε -convexas
usando la funcional $As_3(G)$**

Dada una malla inicial G_0 , $\varepsilon > 0$ y $t = 1$

calcule $G_1 = \arg \min_G As_3(tG)$

con G_0 como malla inicial

Si $\frac{\alpha_-(G_1)}{\bar{\alpha}(G)} < \varepsilon$

Hacer

$G_0 \leftarrow G_1$

$t \leftarrow 2t$

y repita el proceso

Si $\frac{\alpha_-(G_1)}{\bar{\alpha}(G)} > \varepsilon$, Terminamos.

Capítulo 5

Herramientas esenciales para generar mallas Armónicas y Cuasi-Armónicas de manera eficiente

En este capítulo explicaremos las herramientas que usamos para obtener mallas estructuradas armónicas de manera eficiente. Primero daremos las fórmulas de la interpolación transfinita, que se usan para construir una malla inicial, luego, se explica, como al combinar funcionales es necesario normalizarlos y como se lleva a cabo esta normalización y al final, se explica como se utiliza el método de Newton punto a punto para realizar la optimización de un problema de gran escala.

En nuestro caso elegimos esta opción por que el objetivo de este trabajo fue el diseño de un sistema de software que sea sumamente económico y no requiera de grandes recursos de cómputo. Nuestro sistema corre en cualquier computadora con sistema operativo MS-DOS.

5.1. El Método de Interpolación Transfinita

Los métodos más simples usados en la generación de mallas son los basados en interpolación. A continuación presentaremos la definición de interpolación transfinita.

El método de interpolación transfinita tiene la ventaja de ser de fácil implementación y muy rápido en comparación con los métodos variacionales continuos

y discretos que veremos más adelante. Tienen sin embargo la desventaja de que transmiten la falta de suavidad de la frontera hacia el interior y aún más, para regiones que no sean convexas, a menudo dan lugar a mallas dobladas; esto es a mapeos que no son 1 – 1. Sin embargo debido a que estos métodos son usados extensivamente en Diseño Asistido por Computadora (CAD) y a que las mallas que producen son usadas como punto de partida por los métodos que se estudian más adelante, es conveniente revisar al menos la versión más usada de ellos: el llamado método de interpolación transfinita (TFI). El lector interesado en obtener una mayor familiaridad con estos métodos algebraicos puede consultar [16, 32, 34, 46].

Para generar una malla sobre una región Ω es necesario conocer las imágenes de los lados del cuadrado unitario; esto es, se requiere del conocimiento de las cuatro partes en las cuales la frontera de Ω es dividida y que son imágenes de los correspondientes cuatros segmentos de frontera de B_2 . Supóngase que la descripción de la frontera de Ω está dada por cuatro ecuaciones paramétricas:

$$x_b(\xi), \quad x_t(\xi), \quad 0 \leq \xi \leq 1$$

$$x_l(\eta), \quad x_r(\eta), \quad 0 \leq \eta \leq 1$$

Aquí los subíndices b, r, t, l son abreviaturas en inglés de bottom, right, top y left. Por supuesto que se debe conservar la orientación y, para que x sea continua, se debe cumplir que en las esquinas los correspondientes mapeos coincidan; es decir,

$$x_b(0) = x_l(0) = x_{00}$$

$$x_b(1) = x_r(0) = x_{10}$$

$$x_r(1) = x_t(1) = x_{11}$$

$$x_t(0) = x_l(1) = x_{01}$$

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

$$x(\xi, \eta) = (1 - \eta)x_b(\xi) + \eta x_t(\xi) + (1 - \xi)x_l(\eta) + \xi x_r(\eta) - [\xi \eta x_t(1) + \xi(1 - \eta)x_b(1) + \eta(1 - \xi)x_t(0) + (1 - \xi)(1 - \eta)x_b(0)]$$

y se obtiene de combinar la interpolación lineal vertical

$$\bar{x}_v(\xi, \eta) = (1 - \eta)\bar{x}_b(\xi) + \eta\bar{x}_t(\xi)$$

y la interpolación lineal horizontal

$$\bar{x}_\eta(\xi, \eta) = (1 - \xi)\bar{x}_l(\eta) + \xi\bar{x}_r(\eta)$$

y corregir con la interpolación bilineal sobre el cuadrilatero $\bar{x}_{00}, \bar{x}_{10}, \bar{x}_{11}, \bar{x}_{01}$

$$x_{bil}(\xi, \eta) = (1 - \xi)(1 - \eta)\bar{x}_{00} + \xi(1 - \eta)\bar{x}_{10} + \eta x_t(\xi) + \xi\eta\bar{x}_{11} + (1 - \xi)\eta\bar{x}_{01}.$$

es fácil checar que coinciden con $x_b(\xi), x_t(\xi), x_l(\eta), x_r(\eta)$ sobre la frontera.

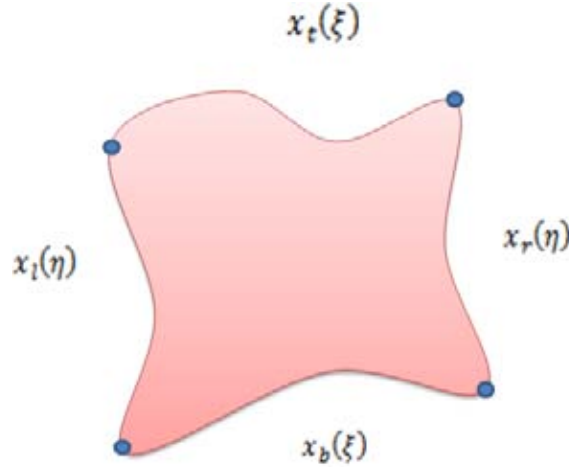


Figura 5.1 Segmentos de frontera de $\partial\Omega$.

El único uso que le daremos a este método es el de generar una malla inicial para el proceso de optimización, aunque, para regiones sencillas, generalmente es la mejor opción.

5.2. Combinaciones interesantes entre funcionales Nuevos y Clásicos

Cuando describimos los funcionales clásicos hicimos hincapié en la necesidad de realizar combinaciones que nos conduzcan a resultados adecuados a las propiedades geométricas deseadas. En el transcurso del capítulo anterior describimos funcionales que de manera automática nos conducen a mallas convexas, tal es el caso de las funcionales S_3 y H_{ws} ; ahora combinaremos sus propiedades con la de las funcionales clásicos.

La combinación de funcionales logra generar mallas con características deseables acordes a la ponderación de ellos. Para contar con una buena combinación, es imprescindible lograr una buena normalización de las funcionales, con

el fin de que los elementos involucrados se encuentren en un mismo plano de valores.

Castillo y Barrera observaron la necesidad de combinaciones convexas entre funcionales de valores semejantes

$$F_c = \sigma F_1 + (1 - \sigma) F_2.$$

Cada una de las funcionales, aún contando con el mismo dominio de definición miden propiedades geométricas de magnitudes reales muy distintas; si deseamos combinar propiedades para trabajar con diversas funcionales, las magnitudes de los valores deben ser semejantes, ver [24, 25].

Utilizando una adecuada normalización \tilde{F}_1, \tilde{F}_2 de las funcionales involucradas estamos en condiciones de llevar a cabo una combinación objetiva de las propiedades geométricas a observar. Para contar con una combinación en porcentajes, usaremos la combinación convexa descrita anteriormente

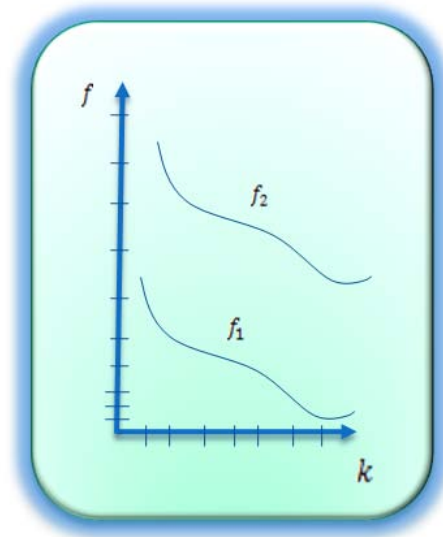


Figura 5.2 Funcionales con orden de magnitud distinta.

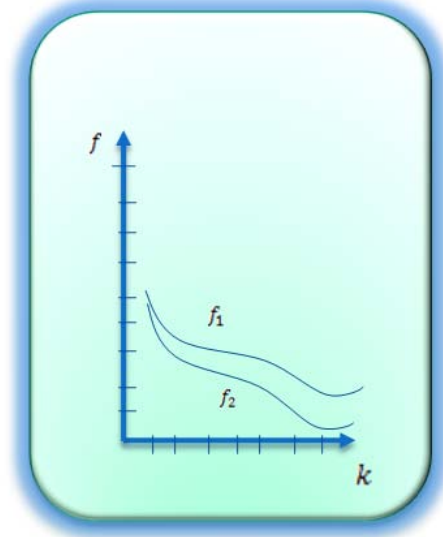


Figura 5.3 Funcionales con orden de magnitud igual .

$$\tilde{F}_c = \sigma \tilde{F}_1 + (1 - \sigma) \tilde{F}_2 \quad 0 \leq \sigma \leq 1$$

y de esta forma ponderar las funcionales \tilde{F}_1 y \tilde{F}_2 normalizadas obteniendo la combinación promedio deseada. Un problema interesante es cómo encontrar una normalización lo más adecuada posible.

5.2.1. Normalización de las funcionales

La forma más sencilla de normalizar valores de las funcionales es hacerlo con respecto al valor esperado o deseado que se alcance en el mínimo. El caso de la funcional de área es clave para entender esta idea. En [3] se mostró que el óptimo de esta funcional, tiende a construir configuraciones donde los triángulos cuentan con área uniforme. De ésta manera, en regiones regulares y en configuraciones donde es posible conseguirlo. El valor de la funcional $I_A(G)$ óptima es

$$I_A^* = N\bar{\alpha}^2$$

por lo que la funcional normalizada de área vendría dado por

$$\begin{aligned} \tilde{I}_A(G) &= \frac{1}{N\bar{\alpha}^2} I_A(G) \\ &= \frac{1}{N\bar{\alpha}^2} F_A. \end{aligned}$$

En combinación con otras funcionales, como por ejemplo Longitud ó k -Suavidad, debemos ser muy cuidadosos al medir el rango de valores donde se espera encontrar cada funcional.

Barrera y Tinoco sugieren observar a la funcional de Longitud de la siguiente forma

$$\begin{aligned} I_L(G) &= \sum_{q=1}^N l(\Delta_q); \quad l(\Delta_q) = l_q; \\ &= \sum_{q=1}^N l_q - 2 \sum_{q=1}^N \alpha_q + 2 \sum_{q=1}^N \alpha_q \\ &= \sum_{q=1}^N (l_q - 2\alpha_q) + 8 \text{area}(\Omega) \end{aligned}$$

donde

$$l(\Delta A_1 A_2 A_3) = \|A_1 - A_2\|^2 + \|A_3 - A_2\|^2$$

y

$$\alpha(\Delta_q) = \alpha_q.$$

para de esta forma relacionarlo con el área de la región. Por una parte,

$$l_q - 2\alpha_q \geq 0$$

luego el valor mínimo posible para $I_L(G)$ vendría cuando $l_q - 2\alpha_q = 0$ y en consecuencia el valor óptimo esperado sería $8 \cdot \text{area}(\Omega)$, por lo que la normalización de la funcional queda

$$\begin{aligned} \tilde{I}_L(G) &= \frac{1}{8 \cdot \text{area}(\Omega)} \sum_{q=1}^N l_q \\ &= \frac{1}{2N\bar{\alpha}} \sum_{q=1}^N l_q \end{aligned}$$

Para el caso de la funcional de Suavidad, observemos que su valor sobre un triángulo está dado por

$$f(\Delta) = \frac{l(\Delta)}{\alpha(\Delta)},$$

se sabe que $f(\Delta)$ alcanza un valor mínimo de 2 sobre los triángulos rectángulos isósceles[46], por lo que la funcional cuasi-armónica $H_w(G)$ queda normalizada con

$$\tilde{H}_w(G) = \frac{1}{2N} H_w(G).$$

Como la funcional A_{s_3} es una funcional de área usamos la normalización.

$$\tilde{A}_{s_3}(G) = \frac{\bar{\alpha}}{N} A_{s_3}(G).$$

En lo sucesivo cuando hablemos de estas funcionales tendremos en mente a las ya normalizadas, a menos que explícitamente se comente lo contrario.

5.3. Método de Newton punto a punto

Cuando se trabajan funciones con un gran número de variables, debemos tratar de explotar cualquier estructura que presente la función para tener algoritmos mas económicos. Muchas de las funciones de gran escala que aparecen en la práctica poseen una propiedad conocida como separabilidad parcial y en los últimos años se han desarrollado métodos que explotan esta estructura en la función objetivo.

En un problema de optimización separable la función objetivo puede descomponerse en una suma de funciones simples que pueden optimizarse de forma independiente, si tenemos

$$f(x) = f_1(x_1) + f_2(x_2) + \dots + f_n(x_n)$$

entonces podemos encontrar el óptimo de $f(x)$ minimizando cada función f_i , $i = 1, \dots, n$, de manera independiente. En muchos casos, realizar n minimizaciones unidimensionales es más económico que el realizar una minimización n -dimensional.

En muchos problemas de gran escala, la función objetivo

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

no es separable, pero puede escribirse como suma de funciones simples conocidas como *funciones elementales*. Para cada función elemental es posible definir una base ortogonal en \mathbb{R}^n de tal forma que el valor de la función no varíe al movernos sobre estas direcciones base. Si esta propiedad se satisface para todas las funciones elementales decimos que f es *parcialmente separable*. Todas las funciones cuyas Hessianas son *sparse* son parcialmente separables, pero también existe la separabilidad parcial en funciones cuyas Hessianas no son *sparse*.

La forma más simple de separabilidad parcial aparece cuando la función objetivo puede escribirse como

$$f(x) = \sum_{i=1}^{ne} f_i(x)$$

donde ne es el número de funciones elementales y cada $f_i(x)$ depende solo de unos cuantas componentes de la variable x , entonces los gradientes ∇f_i y las matrices Hessianas $\nabla^2 f_i$ de cada función elemental típicamente contendrán solo unos cuantos elementos distintos de cero. El gradiente y la matriz Hessiana de la función f están dados por

$$\begin{aligned} \nabla f &= \sum_{i=1}^{ne} \nabla f_i(x) \\ \nabla^2 f &= \sum_{i=1}^{ne} \nabla^2 f_i(x) \end{aligned}$$

y si se aplica un método tipo Quasi-Newton para resolver el problema de optimización es más económico actualizar por el método Quasi-Newton cada matriz Hessiana elemental $\nabla^2 f_i(x)$ en lugar de actualizar toda la matriz Hessiana total.

En el problema que tenemos de generación de mallas, podemos observar que la funcional total a optimizar se describe como la suma de una funcional aplicada a cada celda o a cada triángulo de la malla; esto es, la funcional total a optimizar trabajando por celdas. Es de la forma

$$F(\bar{x}) = \sum_{i=1}^{N_c} f_i(\bar{x})$$

con

$$f_i(\bar{x}) = f(c_i)$$

donde c_i es una celda representativa de la malla y N_c es el número total de celdas, esto es, la funcional es parcialmente separable.

Las funcionales a optimizar para generar las mallas óptimas, como ya hemos mencionado, son funciones parcialmente separables. Tienen un subespacio invariante grande. Entonces se pensó en realizar una optimización considerando la función como una función separable, donde cada función elemental dependiera solo de dos variables: las coordenadas de un nodo interior; esto es, se propone una optimización nodo a nodo.

La idea es la siguiente:

Al optimizar una malla los nodos a la frontera permanecen fijos y solo nos interesa realizar la optimización para los nodos interiores de la malla. Si consideramos una malla de 3×3 como en la figura 5.4, entonces al realizar la optimización resulta un problema de dos variables:

$$z = (z_1, z_2) \in \mathbb{R}^2.$$

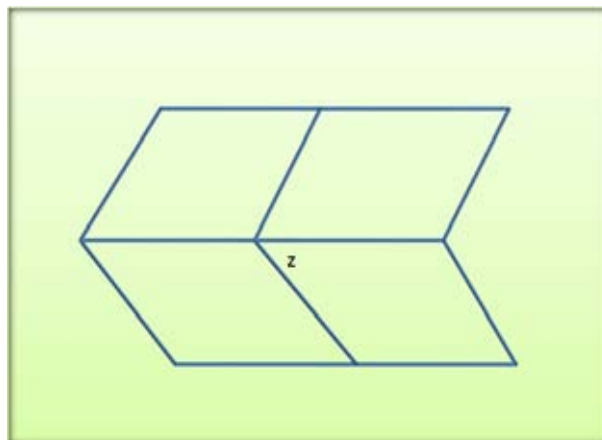


Figura 5.4 Malla de 3×3 .

Si tenemos una malla de $m \times n$ el valor de una funcional sobre una submalla de 3×3 no varía si movemos un nodo interior de la malla ajeno a esta submalla de 3×3 . Entonces resulta razonable realizar la optimización de la funcional solo de manera local, esto es, para cada nodo interior P_{ij} extraemos la submalla de 3×3 formada por sus nodos vecinos, como se muestra en la figura 5.5. Y optimizamos el funcional restringido a esta submalla de 3×3 . Como ya mencionamos se realiza una optimización de solo dos variables.

Una vez realizada la optimización sobre este nodo, se reinserta la submalla de 3×3 ya optimizada a su posición en la malla original. De esta forma solo se altera el nodo P_{ij} en toda la malla.

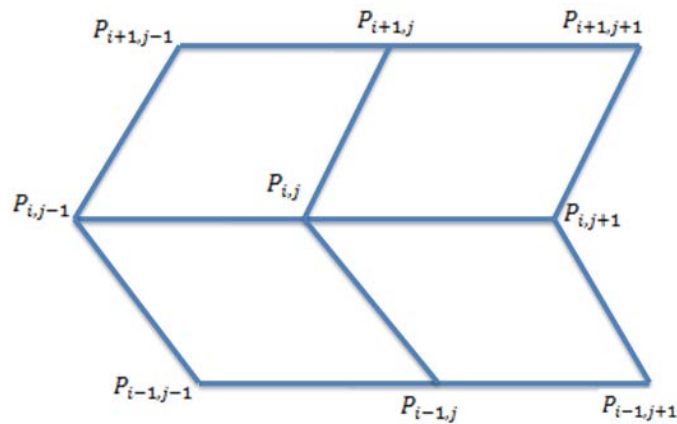


Figura 5.5 Malla de 3×3 con nodos vecinos del punto P_{ij} .

Este proceso se hace para cada nodo interior de la malla, con lo que los requerimientos de memoria son muy bajos. No necesitamos almacenar matrices de grandes dimensiones, solo vectores y matrices de 2×2 que se utilizan en la optimización local.

5.3.1. Optimización local

Veamos ahora como es el proceso de optimización local. Tenemos una malla de 3×3 y una funcional definida sobre los triángulos de sus celdas.

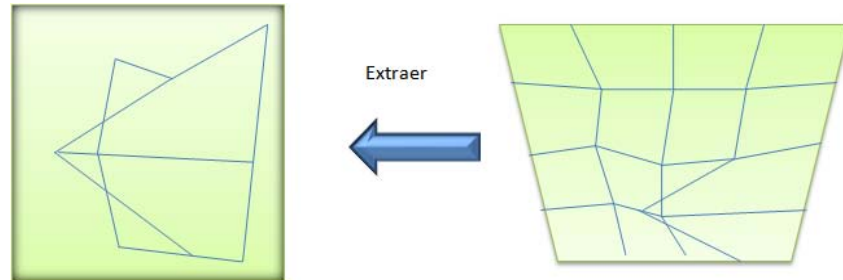


Figura 5.6 Extracción de una submalla de 3×3 .

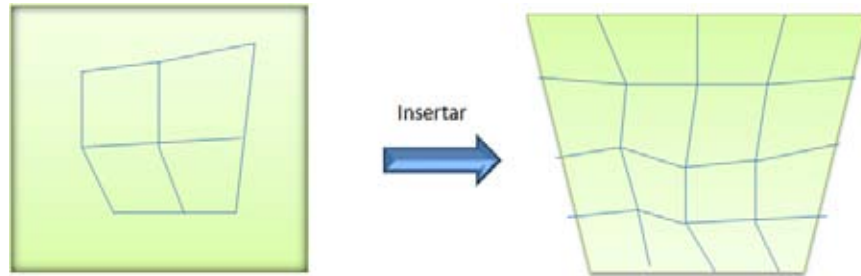


Figura 5.7 Inserción de una submalla de 3×3 .

Hay 12 triángulos en la malla de 3×3 en los que interviene el nodo interior P_{ij} , éstos se ilustran en la figura.

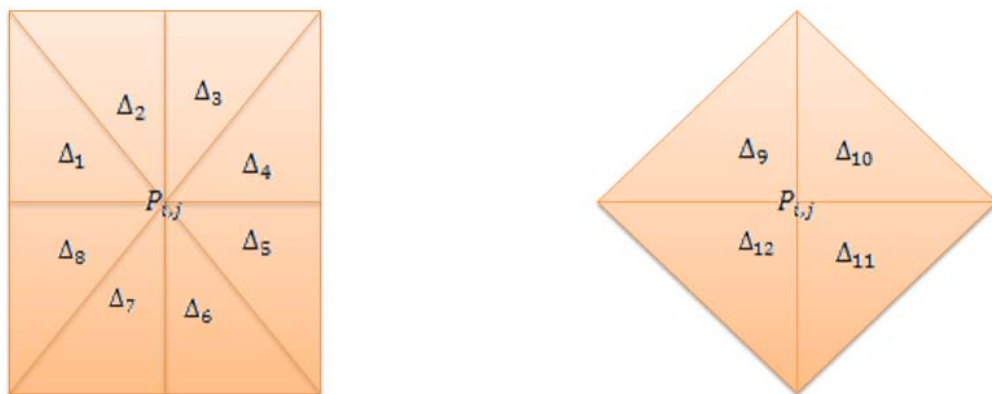


Figura 5.8 Triángulos asociados al punto P_{ij} .

Entonces la funcional local por optimizar es de la forma

$$f = \sum_{i=1}^{12} f(\Delta_i)$$

con $f(\Delta_i)$ la funcional sobre el triángulo Δ_i . Calculando el gradiente y la matriz Hessiana locales de 2×2 analíticamente, el sistema hace la optimización local utilizando algunas iteraciones del método de Newton en dos dimensiones. Como ya mencionamos, no necesitamos almacenar más que un vector de grandes dimensiones el vector de variables totales. Localmente solo se trabajan vectores y matrices de 2×2 , por lo que el sistema puede operar con pocos recursos.

Los resultados obtenidos han sido muy satisfactorios, hemos generado mallas óptimas de grandes dimensiones con este proceso de optimización punto a punto.

Algunas veces cuando la función a minimizar es más difícil, conviene modificar el método de Newton usando una regla de selección del tamaño del paso, véase [20].

En nuestro caso utilizamos un proceso de selección del tamaño del paso, que se conoce como “regla de Armijo” que es muy útil y sencillo de implantar.

Si en x_k la corrección de Newton es p_k y $x_{k+i} = x_k + p_k$ no decrece el valor de la función, entonces se aplica el procedimiento:

Algoritmo para implementar la regla de Armijo

Sea $\alpha = 1$ y $c \in (0, 1)$ (usualmente $c = 10^{-4}$)

Repita hasta que

$$f(x_k + \alpha p_k) \leq f(x_k) + c\alpha \nabla f_k^t p_k$$

$$\alpha \leftarrow \frac{1}{2}\alpha$$

fin

Termina con $\alpha_k = \alpha$,

y entonces la nueva corrección está dada por

$$x_{k+1} = x_k + \alpha_k p_k.$$

Capítulo 6

Diseño e implementación de un sistema para la generación de mallas estructuradas armónicas

6.1. Ciclo de vida de un sistema de información

El ciclo de vida de un sistema de información es un enfoque por fases del análisis y diseño que sostiene que los sistemas son desarrollados de mejor manera, mediante el uso de un ciclo específico de actividades del analista y del usuario.

Según James Senn (Ver [45]), existen tres estrategias para el desarrollo de sistemas: el método clásico del ciclo de vida de desarrollo de sistemas, el método de desarrollo por análisis estructurado y el método de construcción de prototipos de sistemas.

El método de ciclo de vida para el desarrollo de sistemas es el conjunto de actividades que los analistas, diseñadores y usuarios realizan para desarrollar e implementar un sistema de información.

Al ciclo de vida de los sistemas de información también se le denomina ciclo de vida de desarrollo del software. Las etapas típicas del ciclo de vida de desarrollo del software son: la investigación preliminar, la definición de objetivos, el análisis de actividades, identificación de los requisitos, identificación de las estrategias para cumplir los requisitos, planificación, diseño (incluyendo el diseño de la base de datos), creación de prototipos, implementación, revisión y mantenimiento.

Este ciclo de vida hace énfasis en la identificación de las funciones que realiza la institución que desea implementar el sistema, ya sea educativa (o bien una empresa) y en el desarrollo de las aplicaciones que lleven a cabo estas funciones.

Podemos decir que el ciclo de vida de desarrollo del software sigue un enfoque orientado a funciones. Ya que los sistemas se ven desde el punto de vista de las funciones que se llevan a cabo. Por esta razón, el análisis estructurado hace énfasis en los diagramas de flujo de datos; siguiendo el movimiento de los datos a través de una secuencia de transformaciones, y refinando éstas a través de una serie de niveles.

Lo mismo ocurre en el diseño estructurado, que ve a un sistema como una función que se descompone sucesivamente en niveles o subfunciones; concentrándose en las funciones donde la estructura de los datos son manipulados por las funciones. Sin embargo estos sistemas tienen valor durante poco tiempo en relación con las necesidades de los usuarios a largo plazo. Esto sucede debido a que al poco tiempo de haber instalado un sistema, las funciones implementadas son en realidad un subconjunto de las funciones que los usuarios realmente desean. Casi inmediatamente, los usuarios descubren una gran variedad de servicios adicionales que quisieran incorporar al sistema. Estas necesidades causan problemas a los sistemas obtenidos con un diseño orientado a funciones; puesto que este diseño puede requerir una revisión importante para acomodar las funciones adicionales.

En contraste, el enfoque orientado a datos centra el foco de atención en el análisis de los datos utilizados por las funciones. Esto tiene dos ventajas: La primera que los datos son una parte considerablemente más estable que las funciones. La segunda ventaja que la propia estructura de un esquema de base de datos (archivo) requiere de un análisis sofisticado, de los datos y de sus relaciones. Una vez construido un esquema para el archivo de datos, se diseñarán las funciones necesarias para sacar provecho del mismo. Sin embargo, sin un esquema, el archivo de datos sólo es útil para una aplicación. Por tanto, el enfoque orientado a funciones puede ser bueno para el desarrollo a corto plazo, pero pierde su valor real a largo plazo. Usaremos un enfoque orientado a datos; los datos pasan a ser los cimientos sobre los cuales se puede construir una gran variedad de funciones diferentes. En este capítulo se van a estudiar cada una de las etapas del ciclo de vida de desarrollo del software, desde la perspectiva del desarrollo de una aplicación de bases de datos; siguiendo un enfoque orientado a datos. Las etapas que seguiremos para el diseño y la implementación del sistema se muestran a continuación.

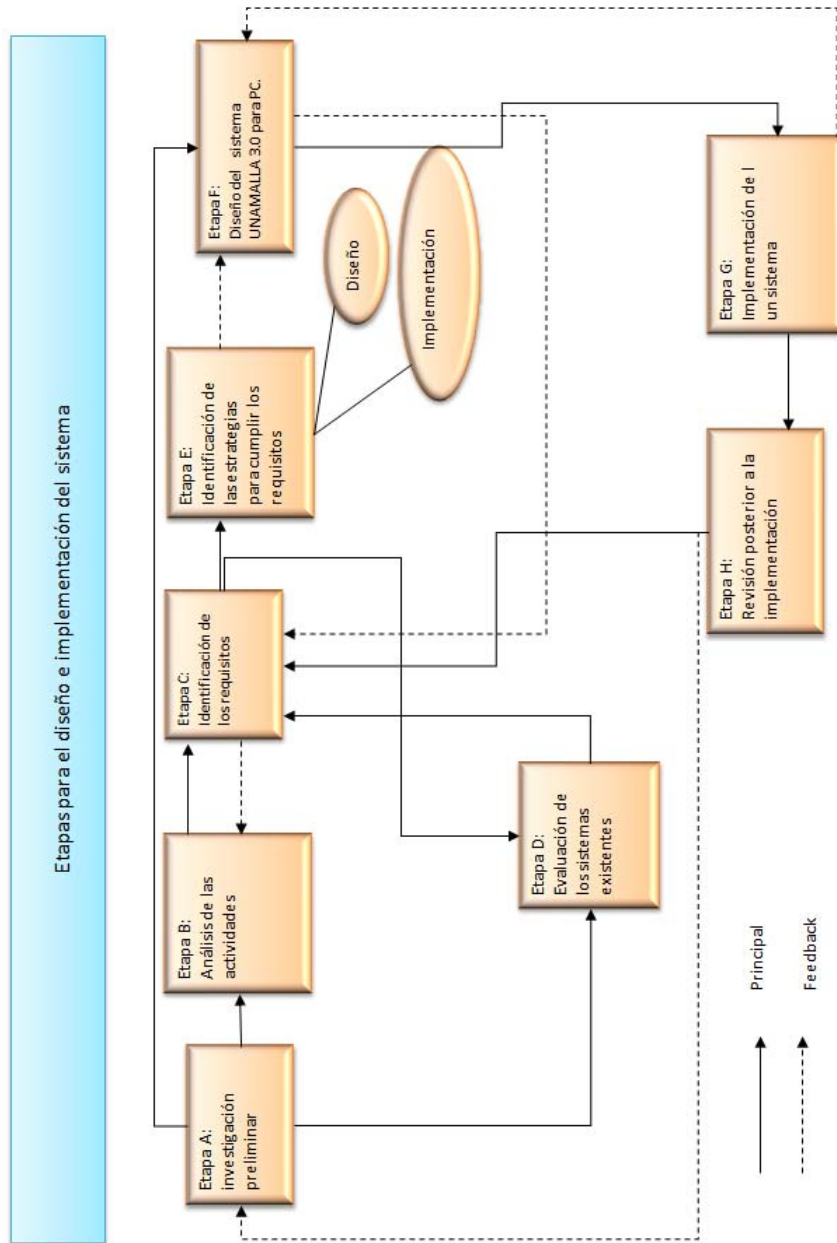


Figura 6.1 Etapas para el diseño e implementación del sistema.

6.2. Etapa A : Investigación Preliminar

El grupo de generación de mallas del departamento de matemáticas de la facultad de ciencias de la UNAM, se crea a finales de los años 80's y con ello el desarrollo de software, debido a la necesidad de automatizar los procesos de generación de mallas. En el año de 1989 se desarrolla el primer programa GRID, en el cual se plantea la necesidad de diseñar e implementar un sistema que tuviera la capacidad de resolver el problema sobre la bahía de La Habana; este sistema primario considera la generación de una malla, a partir de una colección de triángulos orientados, que permita resolver un problema de minimización de áreas hasta obtener una malla convexa; otra aportación fundamental de este sistema fue la implementación del funcional de área Barrera-Pérez, el cual fue comparado con el funcional de área de Castillo. Este sistema fue escrito en lenguaje FORTRAN 77 las rutinas fundamentales y en lenguaje C, el programa principal, se utilizó un graficador independiente elaborado en QUICKBASIC, esto implementado en sistema operativo MS-DOS, uno de los principales logros alcanzados por este sistema fue poder implementar rutinas de interpolación, optimización a gran escala con tan escasos recursos logrando excelentes resultados.

Después en el año 90 se crea el sistema MALLA 1.0, el cual fue elaborado en PASCAL, el graficador de mallas en Lenguaje C haciendo uso de la biblioteca gráfica de Microsoft. Las aportaciones de este sistema radicaron en definir los archivos de entrada y salida del contorno de una región poligonal, así como aceptar regiones que contengan agujeros; los funcionales que se consideraron fueron el funcional de área de Barrera-Pérez y el funcional de longitud. Se utiliza la idea de usar nuevamente triángulos orientados además de utilizar trapecios, se integraron los módulos de generación inicial de la malla y de optimización. Paralelamente se desarrolla el sistema SEGRED, que permite suavizar los datos de la región por medio de un proceso de interpolación lineal y splines cúbicos paramétricos. Esto permite suavizar los datos de una región, poligonal antes de construir una malla sobre ella. Este programa fue escrito en FORTRAN 77, la interacción del sistema se efectúa en PASCAL. Paralelamente se desarrollaron módulos de Optimización, los cuales fueron utilizados para resolver el problema de generación de mallas. Se utiliza el compilador de Fortran Lahey que permite trabajar en una PC y en una estación de trabajo, se desarrollan las bibliotecas gráficas con lo que se permite visualizar el proceso de optimización de mallas.

Hacia el año de 1994 se desarrolla la versión del sistema UNAMALLA 1.0 en esta versión se incorporan todas las ideas hasta el momento estudiadas, se incorporan las utilerías para la creación y edición gráfica de los contornos, se incorpora la generación de malla inicial mediante el método de interpolación transfinita (TFI); es posible visualizar gráficamente el proceso de optimización; se incorpora el proceso de optimización de Memoria Limitada de Nocedal. Más tarde se desarrollaron sistemas experimentales para el suavizamiento de contornos utilizando splines racionales, lo que dio como resultado la generación de mallas suaves y convexas sobre algunas regiones usando la funcional de Area-Ortogonalidad y de Suavidad Regularizado.

A mediados de la década de los 90's se desarrolló y se validó, el sistema TRIANG, el cual utiliza triangulaciones de Delaunay con el propósito de comparar los resultados con los funcionales de Barrera-Pérez y suavidad formulados sobre triángulos orientados. Este sistema fue desarrollado en lenguaje C ANSI haciendo uso de las bibliotecas gráficas de Microsoft 6.0; se consideró un formato sencillo para el intercambio de archivos, el cual pudo ser utilizado para la resolución de algunas ecuaciones diferenciales parciales utilizando el toolbox PDETOOLS de MATLAB.

UMALLA es el resultado de los estudios realizados por Gerardo Tinoco, en este programa se desarrollan funcionales adaptivas de área y de suavidad, se considera una modalidad de optimización puntual que permite manejar mallas de grandes dimensiones este sistema fue escrito en lenguaje FORTRAN de Lahey, haciendo uso de las bibliotecas graficas desarrolladas hasta entonces.

En 1998 Irma García desarrolla un sistema UNAMALLA v.2.0 para PC. que incorpora las nuevas funcionales de área y de suavidad en combinación con las funcionales clásicas, en el cual se incorpora la optimización puntual de Newton, se incorpora una modalidad de homotopía para lograr la combinación convexa de las funcionales adaptivas de área con longitud y la adaptiva de suavidad con área clásica. Se implantó la interfase con el usuario de forma que fuera la más manejable posible, se incorporó al sistema una modalidad de suavizamiento que consiste en aplicarle a la malla algunos pasos del optimizador utilizando la funcional de longitud para eliminar muchas celdas no convexas en mallas dobladas.

Más tarde en 2003 Guilmer Gonzalez Flores desarrolla el sistema UNAMALLA v.2.0 para X-Windows, donde el sistema puede ser utilizado en UNIX y permite utilizar bibliotecas gráficas de libre distribución y portables; se implementaron las funcionales desarrolladas por Tinoco. Hacia Finales de los 90's se inicia con la programación del sistema UNAMALLA utilizando MATLAB aprovechando que varias de las rutinas creadas tanto en Fortran como en Lenguaje C pueden ser compiladas en MATLAB obteniendo resultados gráficos de excelente calidad, esto permite elaborar pruebas rápidas para la implementación de nuevas funcionales así como distintos métodos de optimización.

En 2005 Adriana de la Cruz elabora un sistema en MATLAB para la generación numérica de mallas Armónicas-Adaptivas.

En el mismo año Francisco Domínguez Mota, desarrolla en su trabajo Doctoral métodos efectivos para generar mallas estructuradas, convexas y cuasi ortogonales en regiones irregulares y propone los nuevos funcionales de área que generan y mantienen la convexidad de las mallas, que son C^2 en todo el dominio.

Recientemente en 2008 César Carreón Otañez adicionó un módulo para el tratamiento de contornos en el sistema UNAMALLA 3.0 para MATLAB.

Actualmente se trabaja en el desarrollo del sistema UNAMALLA 3.0 para varias plataformas como parte del Macroproyecto: Tecnologías para la Universidad de la Información y la Computación de la UNAM en colaboración con El

Instituto de Cibernética, Matemática y Física (ICIMAF) de Cuba. Este trabajo es el que ha motivado a seguir desarrollando la implementación del sistema UNAMALLA, el propósito de construir un sistema de información para PC titulado UNAMALLA 3.0, es aprovechar las rutinas elaboradas dada su modularidad y portabilidad y así poder observar, como un sistema que requiere de escasos recursos de equipo de cómputo puede utilizar métodos de optimización, métodos de interpolación y nuevas funcionales sofisticadas y modernas que permiten resolver el problema de generación de mallas en regiones irregulares de una manera eficiente, y que puede ser utilizado por instituciones educativas para resolver problemas de manera sencilla. Las versiones de los sistemas elaborados por el grupo UNAMALLA de la facultad de Ciencias de la UNAM han sido utilizados en diferentes instituciones del mundo en países como: Alemania, Estados Unidos en donde generalmente los avances tecnológicos y científicos están a la vanguardia en esta área; es por esto que el desarrollo de este sistema resulta atractivo. Sin embargo un paso fundamental para el éxito del desarrollo del sistema es la validación de la calidad del software que se implementa mediante un buen planteamiento de proyecto del sistema.

6.3. Objetivos de la investigación

6.3.1. Objetivo General

- Diseñar e implementar el sistema UNAMALLA 3.0 para PC para la generación de mallas estructuradas armónicas.

6.3.2. Objetivos Específicos

- Establecer la sustentación teórica del sistema, para la generación de mallas estructuradas armónicas.
- Diseñar un sistema para la generación de mallas estructuradas armónicas lo más funcional posible.
- Implementar el sistema en lenguajes de programación Fortran y C.
- Evaluar la eficiencia y la eficacia del sistema mediante pruebas de laboratorio elaborando reportes escritos.

6.4. Etapa B : Análisis de las actividades

6.4.1. Documentación descriptiva de los procesos

La programación del sistema UNAMALLA 3.0 para PC, tiene como objetivo la implementación de las nuevas funcionales desarrolladas; en particular la funcional Cuasi-Armónica (Barrera 2007) y la funcional S_3 (Domingez-Mota y Barrera 2001). Dichas funcionales actúan como métodos de barrera cuando se

resuelve el problema de generación de mallas. Una de las principales características de estas funcionales es que al combinarse con algunas funcionales clásicas permiten obtener mallas convexas en regiones muy irregulares; entre las combinaciones de funcionales que se consideraran en la implementación serán:

S₃ – Longitud
S₃ – Ortogonalidad
S₃ – area – Ortogonalidad
Hws – area
Hws – Ortogonalidad
Hws – area – Ortogonalidad

Para alcanzar este objetivo, se desarrollaran dos módulos el primero para la generación de contornos y el segundo para la generación de mallas.

- Módulo de generación de contornos: Debe considerar la opción de crear el contorno y decidir cuando el contorno es convexo.
- Módulo para la generación de mallas: Se debe considerar crear un menú que sea capaz de leer contornos previamente definidos, así como un sub-módulo para la optimización donde se elegirá el tipo de funcional, que puede ser Cuasi-Armónica y funcional de área tipo S_3 .

6.4.2. Clasificación de las actividades

El sistema de generación de mallas estructuradas armónicas se basará en el modelo conceptual que parte de un modelo simbólico que contiene el modelo matemático y las reglas que se seguirán para implementar el sistema. A su vez el modelo matemático será estático, este es el que se ha desarrollado para modelar el problema de generación de mallas y el modelo dinámico, en el cual se considera el modelo numérico que será utilizado para resolver de manera eficiente el problema de generación de mallas numéricas.

Nuestras dos principales áreas, como mencionamos, serán la construcción de dos módulos principales: el módulo de generación de mallas y el módulo de generación de contornos.

- Módulo de generación de contornos:

Debemos hacer la lectura de los datos de un contorno desde un archivo ó bien debemos poder definirlo. Los contornos deben ser cerrados, el usuario podrá definirlos con la ayuda del mouse y el teclado, marcando los puntos en los lugares que desee definir su región, la cual puede ser lo más irregular posible. Una vez definido el contorno deberá ser suavizado a través de un proceso de interpolación.

- Módulo de generación de mallas.

Debemos permitir leer archivos de mallas existentes para poderlas generar de manera óptima o bien se debe permitir reutilizar mallas ya generadas previamente y volverlas a utilizar. Debemos generar códigos que contengan la implementación de las nuevas funcionales; es decir, desarrollaremos códigos óptimos. Debemos generar archivos con la información de las mallas óptimas con el objetivo de ser reutilizadas.

Consideraremos las necesidades de todos los tipos de usuarios que utilizarán el sistema UNAMALLA 3.0, debemos considerar a 2 tipos de usuarios los expertos que modificarán con facilidad cada uno de los procesos y los inexpertos que generalmente serán aquellos que tendrán que aprender a modificar los parámetros a través de la experiencia, realizando varias pruebas con el sistema y el manual de uso del sistema, este tipo de usuarios generalmente serán estudiantes entusiastas que desean aprender el proceso de la generación de mallas o bien usuarios de la rama de ingeniería que necesitan generar una malla inicial para la solución de algún problema específico. En la siguiente subsección se presenta el mapa de procesos que seguiremos en el diseño e implementación del sistema UNAMALLA 3.0 para PC.

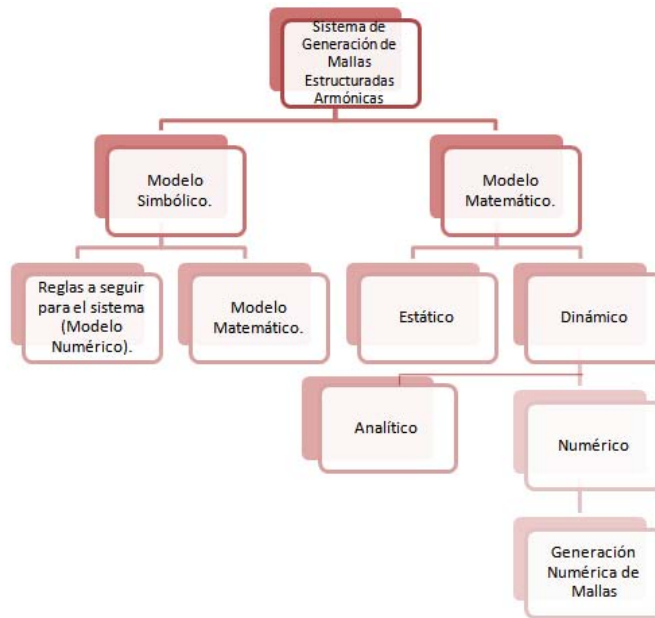


Figura 6.2 Modelo conceptual para generar mallas estructuradas armónicas.

6.4.3. Mapa de procesos

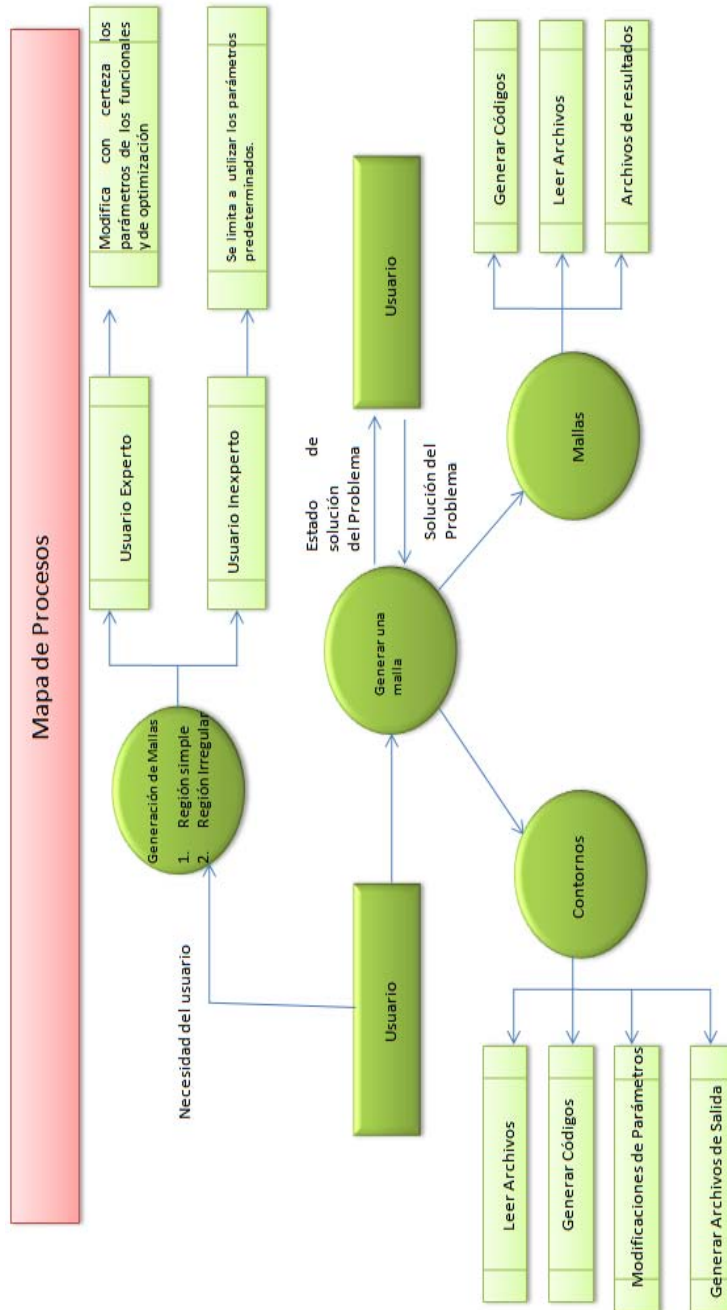


Figura 6.3 Mapa de procesos para el diseño del sistema UNAMALLA 3.0

6.5. Etapa C : Identificación de los Requisitos

6.5.1. Rango de alcance del sistema

El uso de tecnologías informáticas y de automatización dentro de la investigación científica, abarca actualmente espacios muy importantes por la amplia posibilidad que se aporta en la generación y transmisión de los conocimientos, debido a que este tipo de tecnologías requieren que sean dirigidas hacia la eficiencia, esto implica que debemos prever este aspecto, el tipo de tecnología a utilizar y el alcance dentro del proyecto que estamos planteando.

Por tanto nosotros estamos interesados en que el sistema UNAMALLA 3.0 para PC, se diseñe e implemente con los más importantes avances tecnológicos tanto en la construcción de nuevas funcionales, como en los métodos utilizados para el proceso de optimización, de tal manera que el sistema en su conjunto sea lo más económico y eficiente posible, esto deberá ser respaldado por las decisiones de los modeladores matemáticos y los desarrolladores de Software, para que la resolución de generación de mallas sea lo más amigable posible para los usuarios potenciales. Como usuarios potenciales se consideran a estudiantes universitarios y a científicos de las áreas relacionadas con problemas que requieran de una generación de mallas.

En el área educativa sentaremos nuestra atención en fomentar en los estudiantes una actitud y capacidad intelectual amplia y diversa sobre tópicos de la generación de mallas estructuradas armónicas, así como en los conceptos básicos de optimización para adquirir experiencia en el manejo de los parámetros necesarios para los procesos que se desarrollan en el sistema.

En el área científica se pretende que especialistas en el tema se interesen en los métodos utilizados para obtener mallas eficientes a través del sistema, además de interesarse por los aspectos considerados en la calidad de la malla (convexidad, área, ortogonalidad, Épsilon-Convexidad); así como en los procesos de optimización.

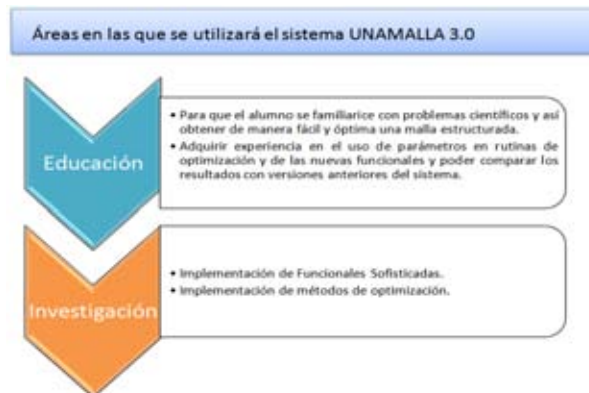


Figura 6.4 Alcances del sistema UNAMALLA 3.0

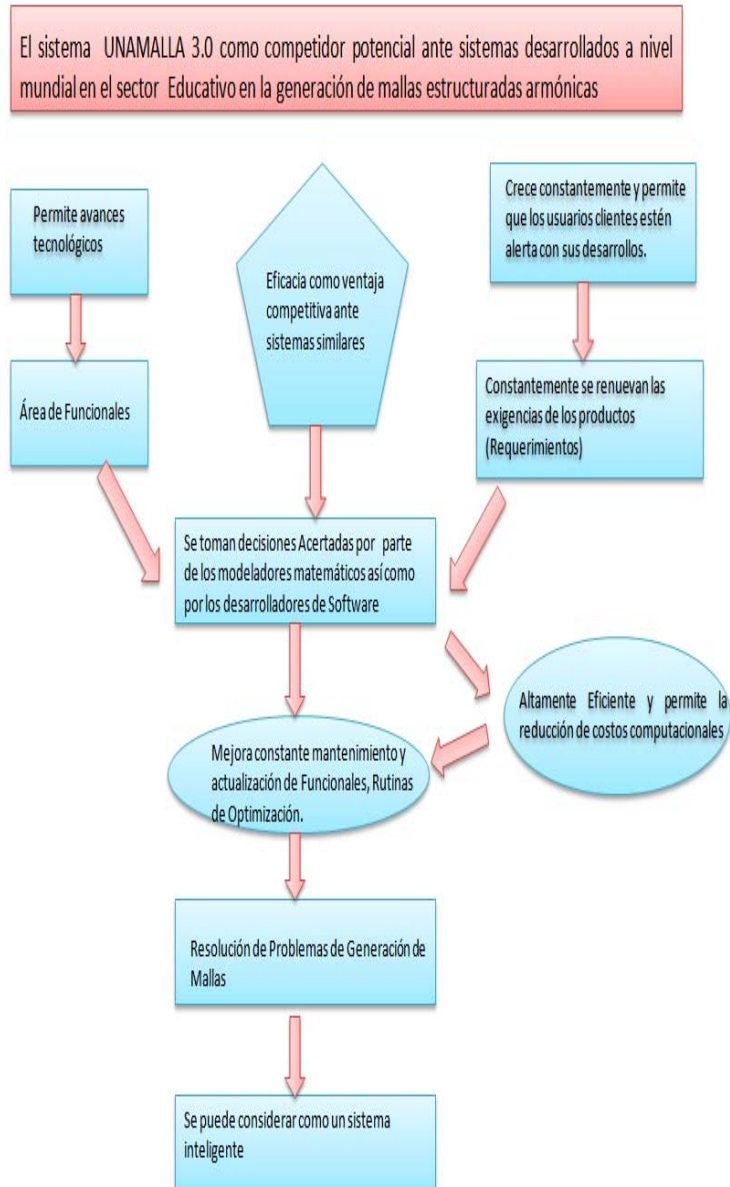


Figura 6.5 Alcances y perspectivas del sistema UNAMALLA 3.0

6.5.2. Requisitos Específicos

Requisito funcional 1: Ménu Principal.

- **Resumen:** Requeriremos de un entorno de trabajo donde se visualicen los submenús de contornos y mallas y la salida del sistema.
- **Entrada:** Para ejecutar el sistema el usuario no requiere de ningún parámetro de entrada.
- **Procedimiento:** Controla y distribuye las tareas principales del sistema.
- **Salida:** Todo el trabajo se realiza en curso y sobre el sistema.

Requisito funcional 2: Construcción de contornos.

- **Resumen:** Requeriremos de una herramienta que nos permita la construcción de un contorno.
- **Entrada:** No hay elementos de entrada, puesto que se lleva a cabo la construcción de la región de trabajo.
- **Procedimiento:** Se definen los segmentos de contornos introduciendo los puntos, aquí se hará uso del mouse. Conforme se vayan introduciendo los datos para el contorno estos se añadirán a una lista en memoria para desplegar posteriormente estos datos en la pantalla ya enumerados. No se debe permitir que haya puntos repetidos.
- **Salida:** Al finalizar la captura, de la información del contorno esta permanecerá en memoria y el usuario podrá trabajar con ella durante su ejecución.

Requisito funcional 3: Lectura de contornos.

- **Resumen:** El sistema debe poder efectuar la llamada de un archivo existente de datos de una región determinada.
- **Entrada:** Nombre del archivo.
- **Procedimiento:** Mediante un browser el usuario debe elegir el nombre del archivo y señalar la ubicación.
- **Salida:** El contorno en memoria.

Requisito funcional 4: Definición de fronteras.

- **Resumen:** Se deberá asignar los segmentos de frontera de la región que será mapeada a las correspondientes fronteras del cuadrado unitario.

- **Entrada:** Un contorno que ha sido definido con anterioridad, ya sea en pantalla o bien en un archivo. Si las fronteras ya están asignadas se tendrá la libertad de modificarlas.
- **Procedimiento:** Se debe de tomar en cuenta una modalidad para elegir automáticamente una distribución de segmentos de frontera, buscando uniformidad a 4 partes del contorno. Utilizando opciones de control y campos de entrada para modificar la elección permitiendo elegir la opción para que sea aceptable.
- **Salida:** Se contará con un contorno donde no haya un pico hacia adentro y en principio se pueda obtener una malla convexa; es decir que no se doble.

Requisito funcional 5: Guardar información del contorno.

- **Resumen:** Una vez creado el contorno es útil contar con la información en disco en formato [nombre del archivo].con.
- **Entrada:** La información del contorno en memoria y el nombre del archivo que se desea guardar.
- **Procedimiento:** Se desplegará en pantalla la opción de guardar contorno.
- **Salida:** El archivo del contorno en disco.

Requisito funcional 6: Generación de la malla inicial.

- **Resumen:** Una vez que se cuenta con un contorno aceptable, podemos generar una malla por interpolación que nos permita usarla como punto de partida o inicial para el procedimiento de optimización involucrado en la generación de la malla por métodos discretos. El método TFI de interpolación permite conservar la frontera de la región poligonal y propaga las singularidades al interior de la región.
- **Entrada:** Un contorno aceptable en la asignación de la frontera y la dimensión de la malla.
- **Procedimiento:** Por medio del campo de entrada se introducirá la dimensión de la malla y se generará de manera automática la malla inicial.
- **Salida:** Si la elección de dimensión de la malla es adecuada para obtener una malla que conserve la forma poligonal sin que el contorno deje de ser aceptable, tendremos una malla inicial. En caso contrario no se podrá trabajar con el resultado.

Requisito funcional 7: Lectura de mallas.

- **Resumen:** Se requiere trabajar con mallas que se han generado previamente o bien continuar con el trabajo que se realiza sobre ellas. Para llevar a cabo este proceso se requerirá llevar a la memoria una malla en disco.
- **Entrada:** Un archivo de mallas en formato [nombre del archivo].red.
- **Procedimiento:** Por medio del campo de entrada se introducirá la dimensión de la malla y se generará de manera automática la malla inicial.
- **Salida:** Si la elección de dimensión de la malla es adecuada para obtener una malla que conserve la forma poligonal sin que el contorno deje de ser aceptable, tendremos una malla inicial. En caso contrario no se podrá trabajar con el resultado.

Requisito funcional 8: Herramientas de despliegue gráfico.

- **Resumen:** Una herramienta que ha abierto mucho el estudio posterior de los métodos discretos es implementar la visualización de la malla conforme se va obteniendo en la pantalla de la computadora, esto permitirá observar los puntos discretos y observar como se va obteniendo hasta que se obtenga la convergencia, así como para el análisis de los resultados.
- **Entrada:** Una malla en memoria.
- **Procedimiento:** Con la malla en memoria se tendrá una ventana subdividida en dos donde se desplegará la evolución de la misma, en la subdivisión continua se desplegará la información de los parámetros más importantes.
- **Salida:** Despliegue en pantalla.

Requisito funcional 9: Optimización de la malla.

- **Resumen:** El proceso de optimización de una malla tiene como papel fundamental lograr propiedades óptimas de acuerdo a las funcionales discretas, esto es que la malla que se obtenga esté cerca de las propiedades de la funcional. Aquí se deberán utilizar las nuevas funcionales discretas para la obtención de mallas suaves y convexas.
- **Entrada:** Una malla en memoria.
- **Procedimiento:** A través de una serie de opciones y menús el usuario podrá elegir la opción que desee ó bien usar las predeterminadas. Al ejecutar la opción de realizar el procedimiento deberá aparecer en pantalla una ventana donde se muestre el desarrollo de las iteraciones así como en pantalla podrá verse gráficamente como el proceso de optimización va evolucionando hasta encontrar la malla óptima.
- **Salida:** Una malla optimizada.

Requisito funcional 10: Guardar información de la malla a disco.

- **Resumen:** Una vez obtenida la malla será necesario guardar la información en disco para su uso posterior.
- **Entrada:** Una malla en memoria y el nombre del archivo donde se guardarán los datos.
- **Procedimiento:** Se desplegará en pantalla la opción de guardar archivo.
- **Salida:** El archivo de la malla en disco.

Requisito funcional 11: Elección del funcional a utilizar en la optimización de la malla.

- **Resumen:** Antes de iniciar la optimización se debe elegir, una funcional adecuada que será utilizada como función objetivo en el proceso de la optimización.
- **Entrada:** Una malla en memoria.
- **Procedimiento:** Se desplegará un submenú donde se debe elegir la funcional a elegir.
- **Salida:** La funcional que se elige como función objetivo en el proceso de optimización.

6.5.3. Requisitos de interfase externa**Interfase con el usuario.**

1. La interfase con el usuario debe de ser amigable, se debe de utilizar recursos como mouse y teclado.
2. Deseamos tener un sistema totalmente gráfico por lo que la interfase debe de fácil manejable para el usuario.
3. Se debe tener interacción entre las tareas de tal manera que los procesos de contornos y mallas se encuentren totalmente identificados.

Interfase con el Hardware.

- Se debe contar con sistema operativo DOS.

Interfase con el Software.

- Para el despliegue grafico se utilizarán librerías creadas en lenguaje de programación C.

6.5.4. Requisitos no funcionales

El sistema debe de contar con los siguientes atributos.

- **Disponibilidad:** El usuario debe de poder ejecutar el sistema en cualquier momento, siempre y cuando se encuentre trabajando en el sistema de ventanas de la computadora.
- **Mantenimiento:** El mantenimiento del sistema es un factor fundamental en futuros desarrollos. Si se prevee que el sistema cuente con módulos que puedan ser actualizados mediante la sustitución de sus componentes esto permitirá su crecimiento con un costo mínimo. Además si se cuenta con una documentación adecuada permitiremos que un mayor número de usuarios puedan participar en desarrollos futuros.

6.6. Etapa D : Evaluación de los sistemas existentes

El sistema que se evaluó fue el sistema UNAMALLA para PC. 2.0. Este sistema resuelve el problema de generar mallas óptimas en regiones planas irregulares la manera de generar es por medio de un proceso de optimización. El problema de generación de mallas se resuelve siguiendo el orden siguiente:

Primero se genera una malla inicial a través de TFI (Interpolación transfinita), después se optimiza una funcional que refleja alguna propiedad geométrica buscada en las mallas; después se genera una sucesión de mallas que convergen a una malla óptima.

La programación del sistema UNAMALLA para PC. 2.0. es modular lo que permite que sea fácil de modificar, este sistema tiene implementado el método de Newton punto a punto para obtener mallas óptimas.

Este sistema considera 2 modos de interacción el modo manual y el modo automático; el modo automático contiene parámetros determinados por tanto no se requiere ser un usuario experto que domine los conceptos de generación de mallas, el modo manual permite que el usuario modifique parámetros de acuerdo a su experiencia en la generación de mallas.

Este programa permite crear contornos y manejar contornos ya existentes, además se pueden leer mallas óptimas y guardar mallas obtenidas; entre las funcionales que se tienen implementados en esta versión del sistema se encuentran disponibles cuatro combinaciones entre las funcionales básicas de *área*, *k-Suavidad*, *Longitud*, *k-área* y *Ortogonalidad*.

$$\begin{aligned}
 &w(k - Suavidad) + (1 - w)area \\
 &w(area) + (1 - w)Longitud \\
 &w(k - area) + (1 - w)Longitud \\
 &area - Ortogonalidad
 \end{aligned}$$

El usuario debe de dar un peso asociado a la primer funcional en la combinación convexa a trabajar la cual se escoge en el intervalo

$$0 \leq w \leq 1$$

Este sistema utiliza el método de Newton que requiere de utilizar la matriz Hessiana y del gradiente de la funcional calculadas analíticamente esto permite utilizar 2 métodos de solución para resolver el sistema lineal, el método componente a componente y el método simultaneo.

Ahora se presenta una serie de ejemplos de mallas generadas por el sistema, para llevar estos se utilizaron las combinaciones entre la funcional de $k - Suavidad$ y $área$ Clásica con $w = 0,25$ y la combinación entre $k - area$ y $Longitud$ con $w = 0,75$.

Mallas obtenidas con las funcionales de $k - Suavidad$ y $área$ Clásica

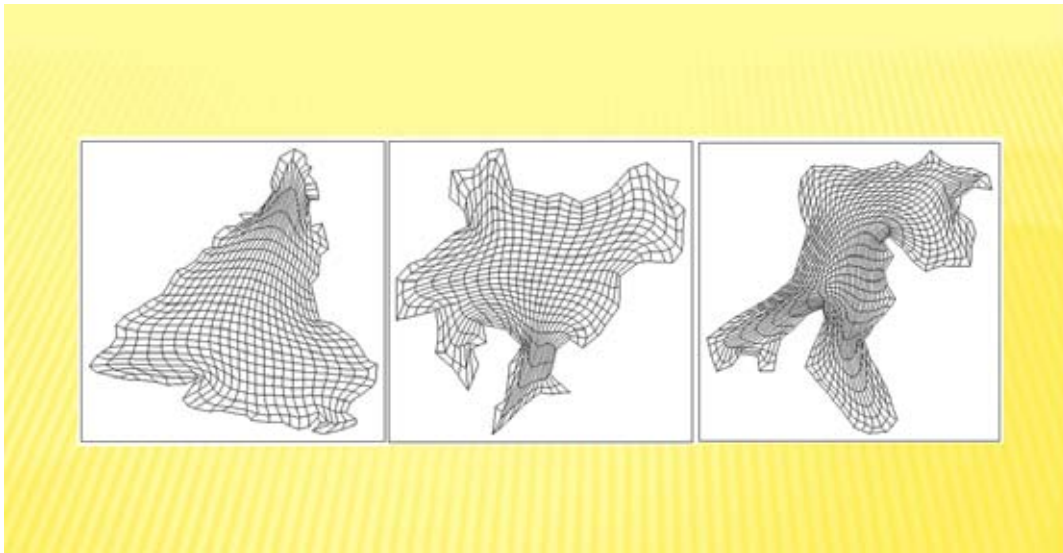


Figura 6.6 Mallas generadas con las funcionales K-Suavidad y $área$ Clásica.

Mallas obtenidas con las funcionales de $k - area$ y $Longitud$.

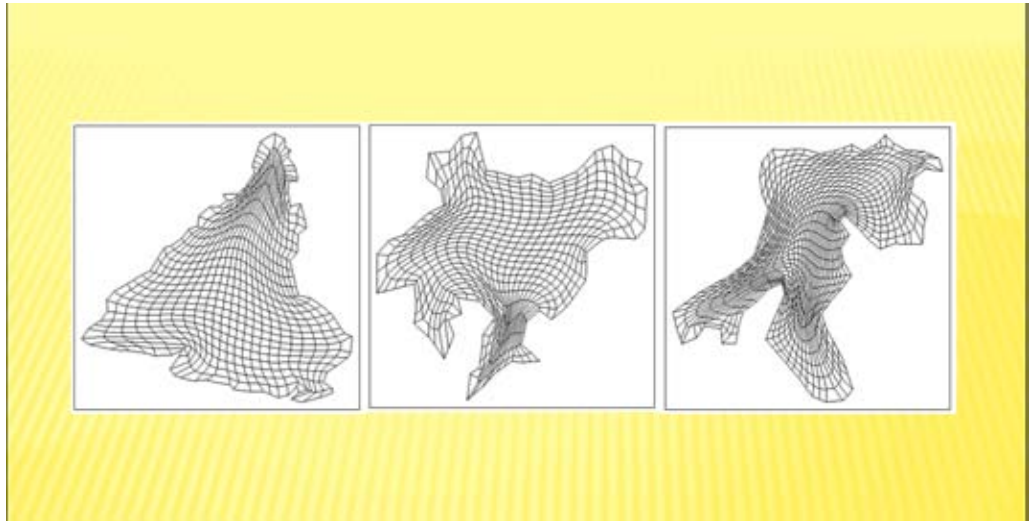


Figura 6.7 Mallas generadas con las funcionales k-area y Longitud

6.7. Etapa E : Identificación de las estrategias para cumplir los requisitos

En esta sección definiremos las estrategias necesarias para llevar a cabo cada uno de los requisitos.

1. Hacer una detallada revisión del sistema UNAMALLA 3.0 para PC.
2. Identificación de cada uno de los módulos del sistema UNAMALLA 3.0 para PC. (Generación de la malla Inicial, proceso de optimización y la elección de la funcional)
3. Analizar con detalle el módulo del proceso de optimización sobre todo dedicar tiempo en el análisis del método de Newton y observar su evolución durante el proceso de generación de mallas.
4. Implementación de las funcionales Armónicas y Cuasi-armónicas.
5. Validación de la implementación de las funcionales haciendo una serie de pruebas con diferentes contornos sencillos (cuadrado, pentágono etc); realizar el mismo proceso con regiones complicadas (la Habana, Sudamerica, Rusia etc).
6. Si se detecta algún problema en la solución del problema de optimización del tipo numérico solucionarlo de la manera más eficiente.
7. Modificar salidas de resultados de tal manera que se obtengan ambientes amigables que permitan que el usuario comprenda de manera sencilla cada proceso que se involucra en el sistema.

6.8. Etapa F : Diseño del Sistema UNAMALLA para PC 3.0

Para el diseño del sistema se debe de preservar la modularidad del sistema en base a rutinas existentes de sistemas anteriores esto permite seguir agregando modulos fáciles de adaptar en sistemas posteriores. La base que se tomará será el sistema UNAMALLA para PC 2.0; además se adicionaran las funcionales Cuasi-Armónicas y la Épsilon-Convexidad, para lograr este objetivo se procederá de la siguiente manera:

1. Tendremos un menú principal el cual contará con las opciones de contorno,

malla y salida del sistema.

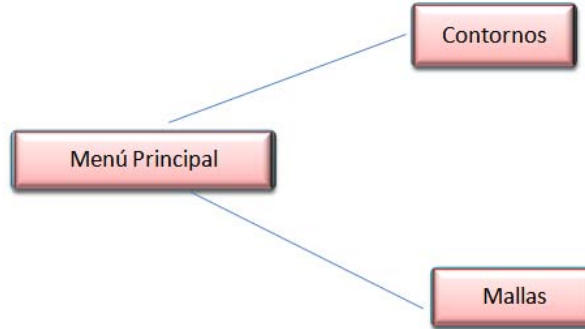


Figura 6.8 Diagrama de bloques del módulo menú principal.

2. El menú contorno debe permitirnos contar con un submenú donde se tendrán las opciones de generar contorno, lectura de datos, guardar datos, generar una malla inicial. Si nosotros deseamos generar un contorno se desplegará la opción definir frontera, esta opción deberá permitirnos definir los segmentos de la frontera y hacer la definición de las fronteras, la cual podrá ser manual o bien deberá contar con la elección automática de la definición de fronteras. Se tendrá la opción de continuar y después se desplegará un submenú para guardar la información, esta nos permitirá salvar el contorno que se ha definido por el usuario. Dentro de esta opción también podremos proporcionar las dimensiones de la malla que será utilizada como malla inicial y a la cual se le aplicará una interpolación transfinita.

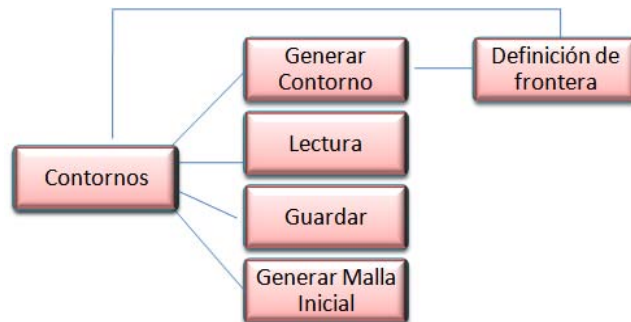


Figura 6.9 Diagrama de bloques del módulo de contornos.

3. El menú malla nos deberá desplegar un submenú donde se desplegarán las opciones de leer una malla existente, guardar una malla, optimizar, definir el alfa límite y la opción de salida. Si elegimos la opción de leer se desplegará el submenú malla en este se tendrán las opciones de dar nombre, buscar un archivo existente y la opción de salir. Ya que elegimos

el archivo se desplegará información de la malla en caso de leer un archivo de una malla que fue creada con anterioridad. Si elegimos la opción de optimizar se desplegará el tipo de la funcional que se desea utilizar esta puede ser Cuasi-Armónica o bien del tipo S_3 , se deberá desplegar otro submenú donde se preguntará si se desean definir los parámetros para efectuar la optimización, esta podrá ser hecha manualmente o bien de forma automática con parámetros preestablecidos. Si se elige de forma manual se tendrán 2 submenús uno para las funcionales Cuasi-Armónicas y otro para las funcionales del tipo S_3 . Se elegirá el tipo de optimización que se desea hacer Newton componente-componente o simultáneo, a su vez se tendrá un submenú donde se desplegará si se desea hacer la optimización utilizando las condiciones de Armijo o bien sin las condiciones de Armijo.



Figura 6.10 Diagrama de bloques del módulo mallas.

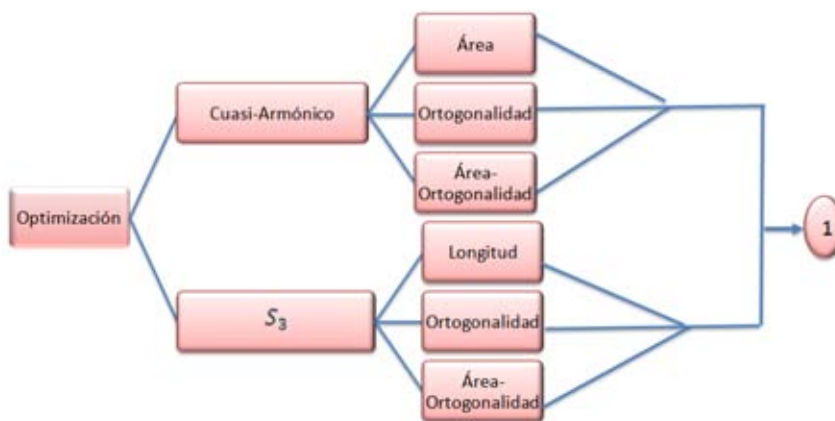


Figura 6.11 Diagrama de bloques del módulo optimización.

6. DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA LA GENERACIÓN DE MALLAS ESTRUCTURADAS ARMÓNICAS
112

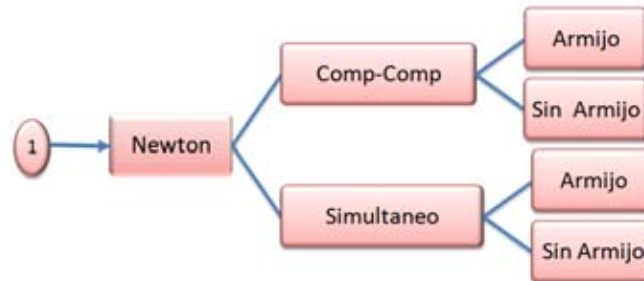


Figura 6.12 Continuación del diagrama de bloques del módulo optimización.

- Después de haber definidos los módulos del sistema estos serán ensamblados en un programa principal donde se seguirá la estructura que se muestra a continuación.

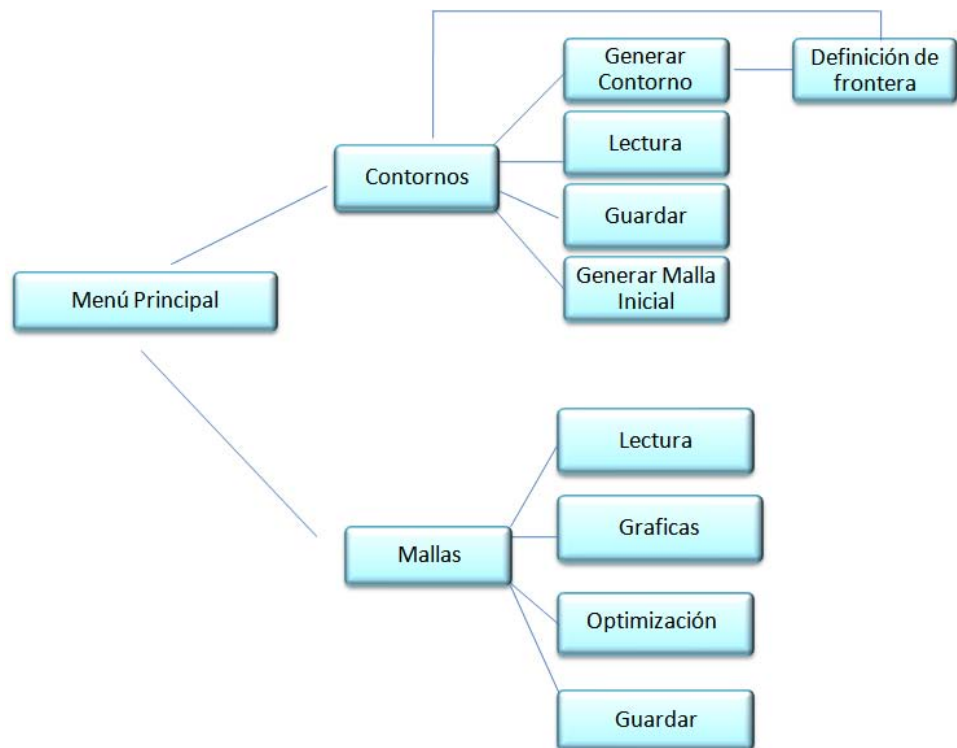


Figura 6.13 Diagrama de bloques del sistema UNAMALLA 3.0

6.9. Etapa G : Implementación del Sistema

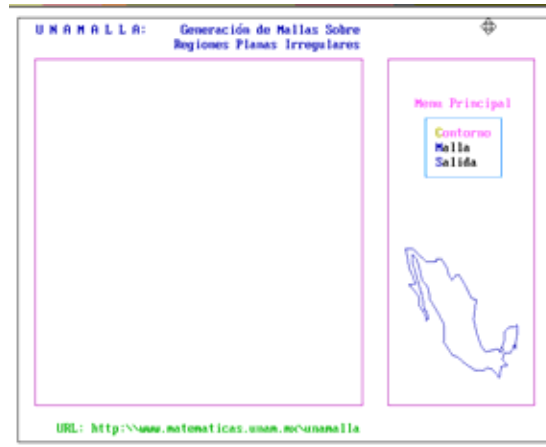


Figura 6.14 Menú principal sistema UNAMALLA3.0.

- El menú principal nos muestra las opciones de contorno, malla y salida del sistema.

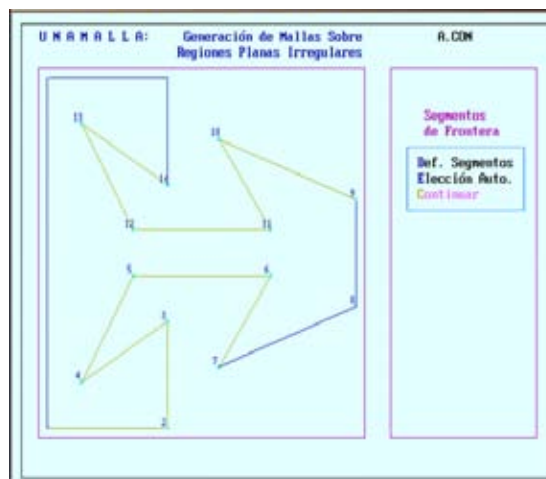


Figura 6.15 Submenú segmentos de frontera.

- Si el usuario elige la función contorno se desplegará el submenú segmentos de frontera aquí podemos elegir las opciones: Definición de segmentos, elección automática, una vez que hemos elegido la opción deseada el usuario podrá elegir continuar

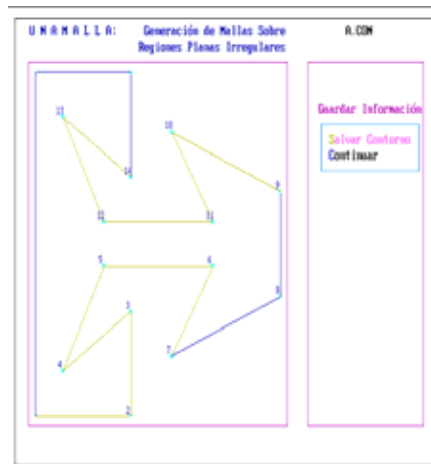


Figura 6.16 Opción salvar contorno en el sistema UNAMALLA 3.0.

- Después el usuario podrá guardar la información generada, con la opción salvar información, por último se seleccionará la opción continuar.

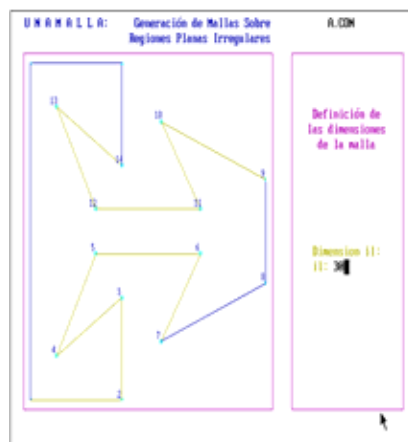


Figura 6.17 Definición de las dimensiones 11 de la malla.



Figura 6.18 Definición de las dimensiones $j1$ de la malla.

- Debemos definir las dimensiones de la malla esto lo podremos hacer con $i1, j1$.

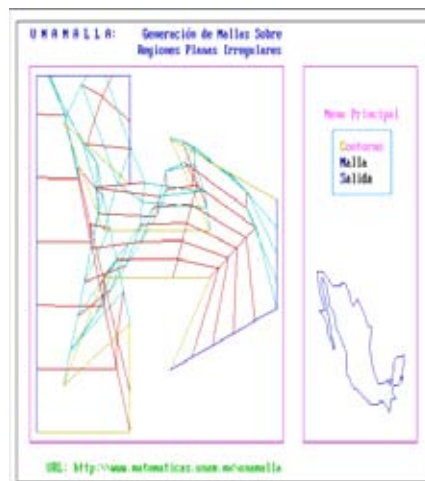


Figura 6.19 Menú principal sistema UNAMALLA 3.0.

- Después de haber definido el contorno regresamos al menú principal donde podremos elegir la opción del menú malla.

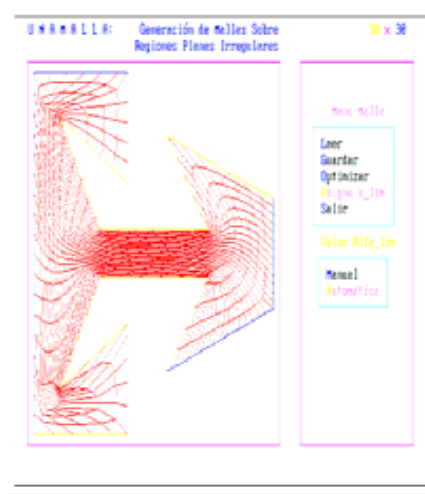


Figura 6.20 Menú malla.

Ahora se desplegara un submenú de malla donde se tienen las opciones

- Leer : en esta opción se permite elegir un contorno generado previamente o bien una malla generada previamente.

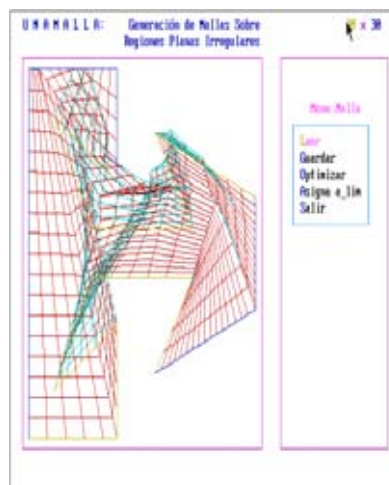


Figura 6.21 Opción leer malla.

- Guardar : nos permite guardar la malla generada.

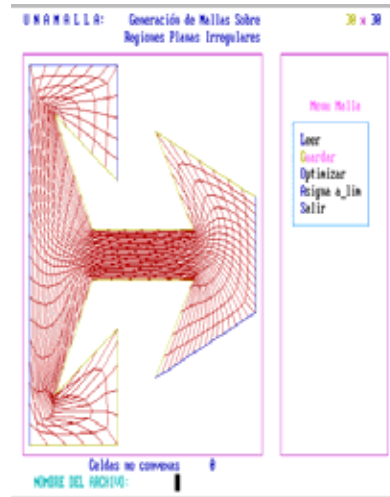


Figura 6.22 Opción guardar malla.

- Optimizar : nos permite elegir el método para optimizar.
- Asigna a_lim : se define el valor de alfa mínima, se despliega un submenú donde se tienen las opciones para asignar el valor de alfa esta puede ser de forma manual o bien de forma automática.

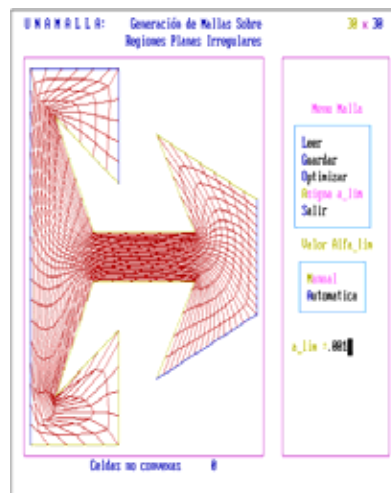


Figura 6.23 Opción optimizar malla.

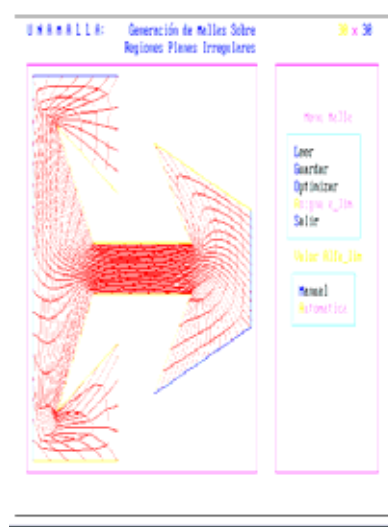


Figura 6.24 Opción asignar valor de alfa.

- Si el usuario elige la opción de optimizar se desplegará un submenú donde se podrá elegir el tipo de la funcional la cual puede ser del tipo armónica o bien del tipo S_3 . Si es del tipo cuasi armónica, se desplegará el submenú donde se presentan las diferentes combinaciones.

$S_3 - Longitud$
 $S_3 - Ortogonalidad$
 $S_3 - area - Ortogonalidad$

de la misma manera se despliega un menu similar para las funcionales del tipo cuasi armonica.

$Hws - area$
 $Hws - Ortogonalidad$
 $Hws - area - Ortogonalidad$

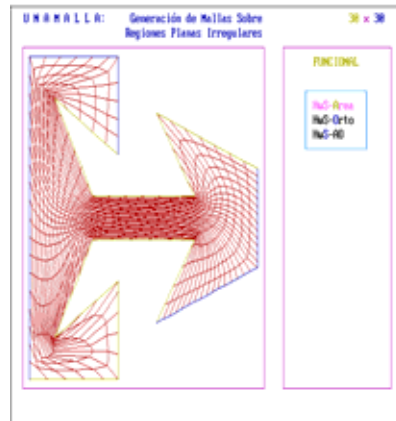


Figura 6.25 Submenú elige funcional.

El siguiente menú que se desplegara será el de la optimización la cual puede efectuarse de manera automática donde se utilizarán los parámetros determinados, o bien de manera manual se desplegara un submenú con las opciones:

- Método de Newton Componente a Componente
- Método de Newton Simultaneo.

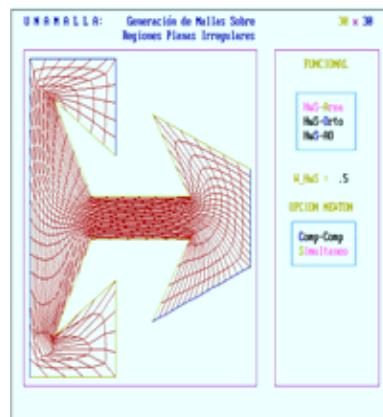


Figura 6.26 Submenú elige método de optimización.

6.10. Etapa H : Revisión posterior a la Implementación

En esta etapa se efectuaron una serie de ejemplos; se eligieron 3 regiones irregulares para nuestros ejemplos que son la región de Sudamérica, Rusia y el Cisne.

Primero se generaron mallas iniciales utilizando TFI, en ellas se observó que las mallas no son convexas, después se procedió a optimizar las mallas utilizando el método de Newton, se aplicaron 2 técnicas diferentes para resolver el sistema la componente a componente y el método simultaneo, el objetivo de utilizar estos métodos es observar la eficiencia de ellos en la solución de un problema de gran escala como es la generación de mallas.

Otro punto de interés tratado durante esta etapa fue observar la eficiencia de utilizar tanto la funcional S_3 como la funcional H_w .

Ejemplos $S_3 - Longitud$

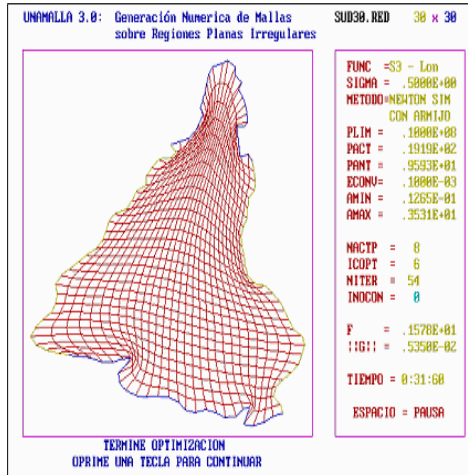


Figura 6.27 Ejemplo funcional S3-Long. Sobre región Sudamérica 30x30 por lado.

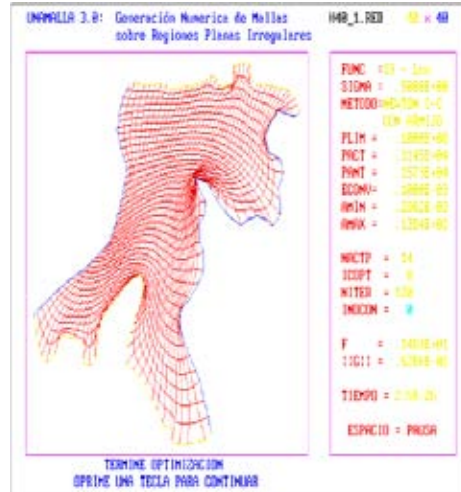


Figura 6.28 Ejemplo funcional S3-Long. Sobre región Habana, 40x40 por lado.

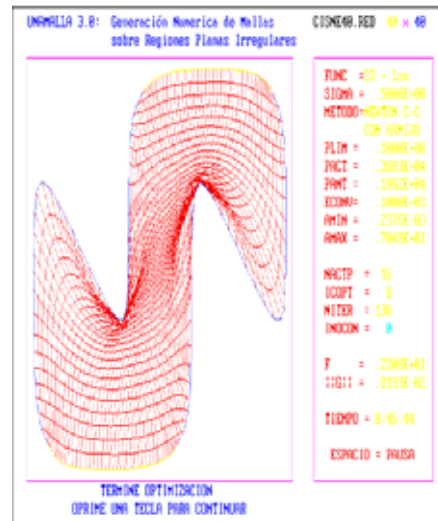


Figura 6.29 Ejemplo funcional S3-Long. Sobre región Cisne, 40x40 por lado.

Ejemplos S_3 – *Ortogonalidad*

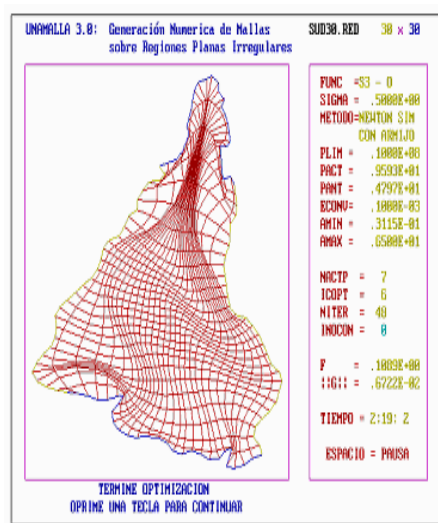


Figura 6.30 Ejemplo funcional S3-Ort. Sobre región Sudamérica, 30x30 por lado.



Figura 6.31 Ejemplo funcional S3-Ort. Sobre región Habana, 40x40 por lado.

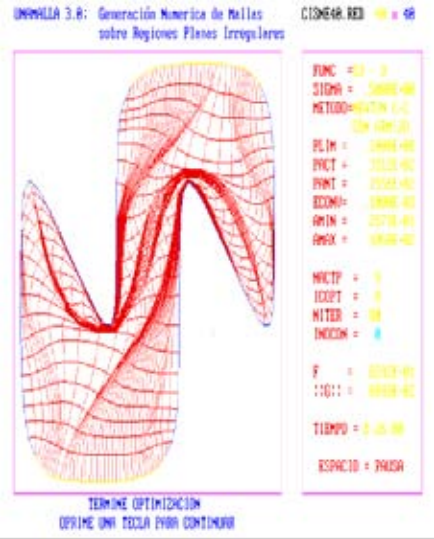


Figura 6.32 Ejemplo funcional S3-Ort. Sobre región Cisne, 40x40 por lado.

Ejemplos S_3 – area – Ortogonalidad

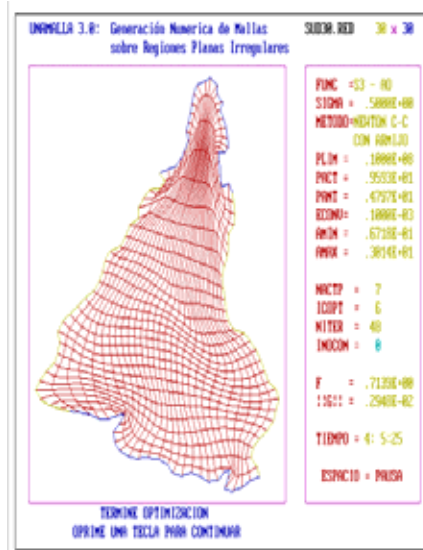


Figura 6.33 Ejemplo funcional S3-area-Ort. Sobre región Sudamérica, 30x30 por lado.

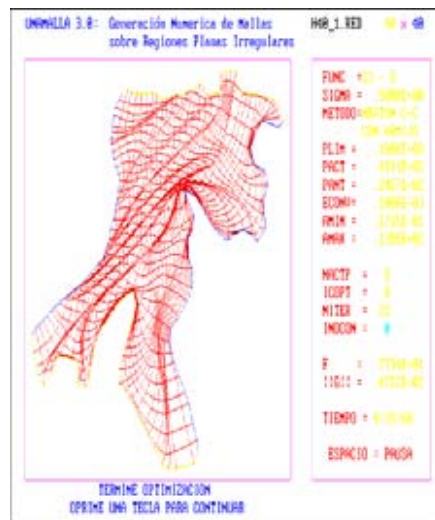


Figura 6.34 Ejemplo funcional S3-area-Ort. Sobre región Habana, 40x40 por lado.

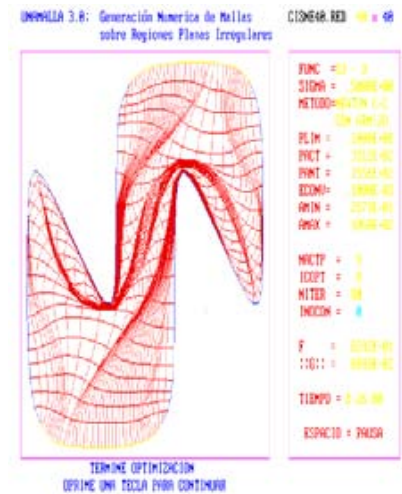


Figura 6.35 Ejemplo funcional S3-area-Ort. Sobre región Cisne, 40x40 po rlado.

Ejemplos $Hws - area$

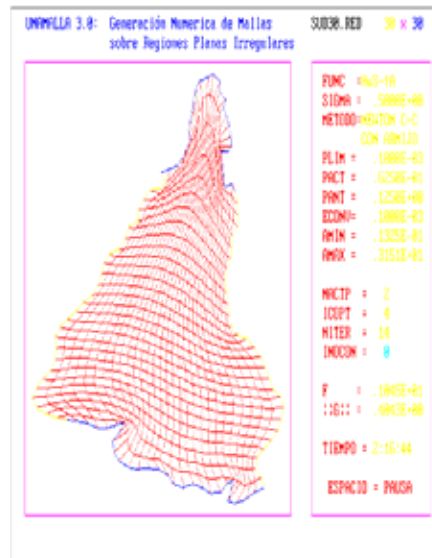


Figura 6.36 Ejemplo funcional Hws-area. Sobre región Sudamérica, 30x30 por lado.

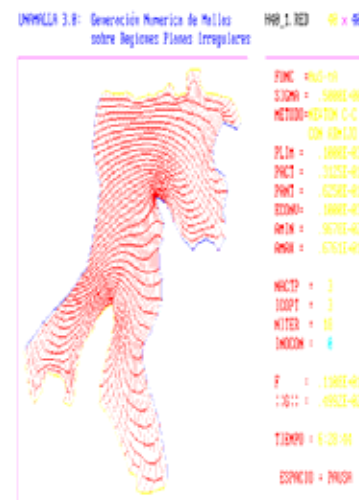


Figura 6.37 Ejemplo funcional Hws-area. Sobre región Habana, 40x40 por lado.

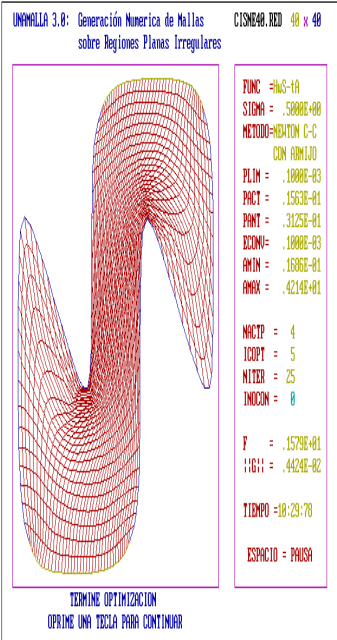


Figura 6.38 Ejemplo funcional. Hws-area. Sobre región Cisne, 40x40 por lado.

6. DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA LA GENERACIÓN DE MALLAS ESTRUCTURADAS ARMÓNICAS
128

Ejemplos *Hws – Ortogonalidad*

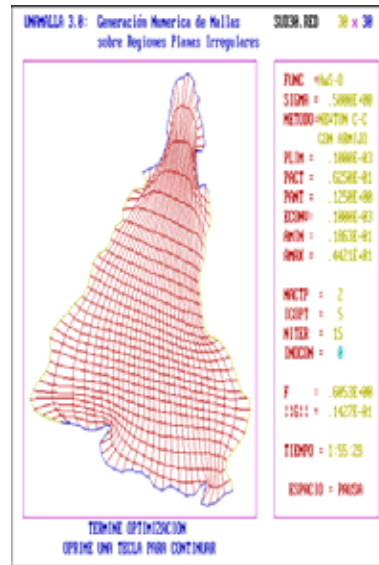


Figura 6.39 Ejemplo funcional Hws-Ort. Sobre región Sudamérica, 30x30 por lado.

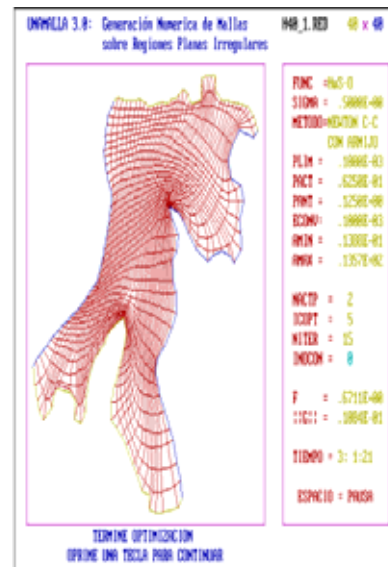


Figura 6.40 Ejemplo funcion Hws-Ort. Sobre región Habana, 40x40 por lado.

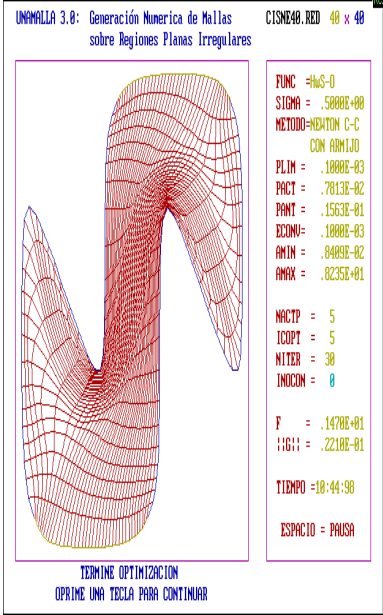


Figura 6.41 Ejemplo funcional Hws-Ort. Sobre región Cisne, 40x40 por lado.

Ejemplos $Hws - area - Ortogonalidad$

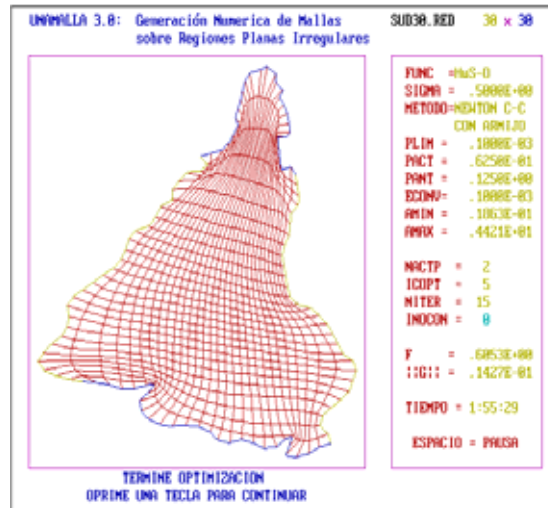


Figura 6.42 Ejemplo funcional Hws-AO. Sobre región Sudamérica, 30x30 por lado.

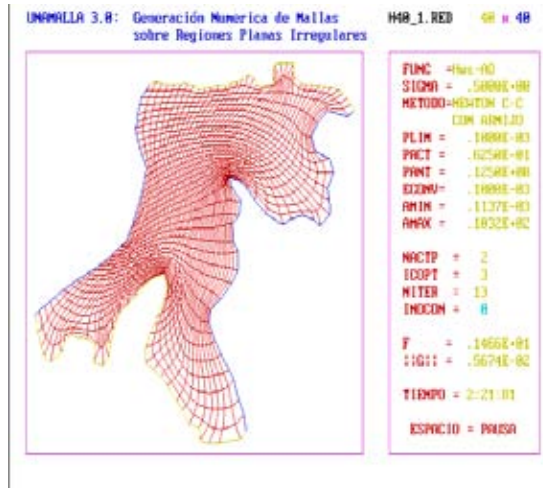


Figura 6.43 Ejemplo funcional Hws-AO. Sobre región Habana, 40x40 por lado.

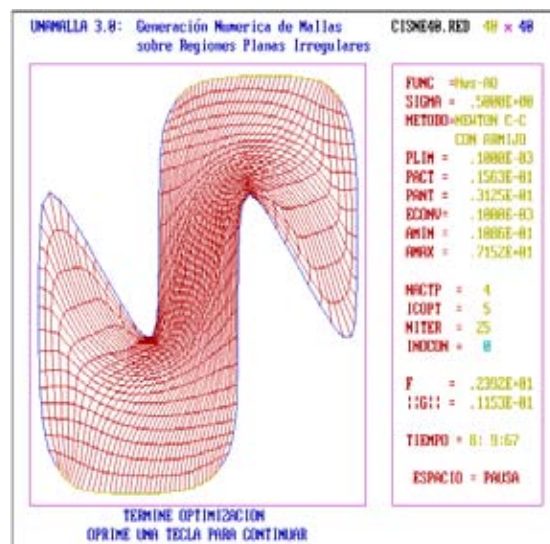


Figura 6.44 Ejemplo funcional Hws-AO. Sobre región Cisne, 40x40 por lado.

Bibliografía

- [1] Arieh Iserles, **A First Course in Numerical Analysis of Differential Equations**, Cambridge Text in Applied Mathematics and Theoretical Physics, University of Cambridge.
- [2] Barrera, P., Castellanos, L. y Pérez, A. **Curvilinear coordinate system generation over plane irregular regions**. Reportes técnicos de la Facultad de Ciencias, U.N.A.M., 1992.
- [3] Barrera, P., Castillo J.E., **A Large Scale Nolinear Optimization Problem Arising from Grid Generation**, Teach.Report. No 1, Dept of Math. and Stat., University of New Mexico, 1987.
- [4] Barrera, P., Domínguez-Mota, F.J. y González, F.G. **Some experiences on orthogonal grid generation**. Proceedings of the 9th International Meshing Roundtable, 2000.
- [5] Barrera, P., Domínguez-Mota, F.J. y González, F.G. **Variational grid generation using exponentials of Área**. Proceedings of the 10th International Meshing Roundtable, 2001.
- [6] Barrera, P., Domínguez-Mota, F.J. y González, F.G. **Robust discrete grid generation on plane irregular regions**. Proceedings of the Computing Centre RAS and R&D Company 2002 Workshop, Moscow, Russia, 2002.
- [7] Barrera, P., Domínguez-Mota, F.J. y González, F.G. **Some experiences on orthogonal grid generation**. Applied Numerical Mathematics 40, 2002.
- [8] Barrera, P. y Tinoco, J.G. **Smooth and convex grid generation over general plane regions**. Mathematics and Computers in Simulation, 1998.
- [9] Barrera, P. y Tinoco, J.G. **Area functionals in plane grid generation**. 6th Conference in Numerical Grid Generation in Computational Field Simulation Conference, 1998.

- [10] Barrera, P. y Tinoco, J.G. **Area control in generating smooth and convex grids over general plane regions**. Journal of Computational and Applied Mathematics, 1999.
- [11] Barrera, P. Castellanos N., Domínguez-Mota, González F., Pérez D. **Adaptive discrete harmonic grid generation**. Science Direct. 2007.
- [12] Barrera, P. Domínguez-Mota, González F., Tinoco, J.G., **Generating - Quality Structured Convex Grid On Irregular Regions**.
- [13] Brackbill, J.U. y Saltzman, J.S. **Coordinate system control: Adaptive meshes**. Numerical Grid Generation, 1982.
- [14] Castellanos J.L., **Generación numérica de redes usando Newton Truncado**, Tesis Doctoral, I.C.I.M.A.F., Ministerio de la Ciencia y Tecnología y el medio ambiente, La Habana, Cuba, 1994.
- [15] Castillo, J.E. **Variational Grid Generation**. PhD thesis, University of New Mexico, 1987.
- [16] Castillo, J.E. **Mathematical Aspects of Numerical Grid Generation**. SIAM, 1991.
- [17] Carreón, O.C. **Un Módulo de para el tratamiento de contornos en el sistema UNAMALLA**, Tesis de licenciatura, UNAM, 2008.
- [18] Cortes, A. J.J. **El método de Newton Truncado para la generación de mallas estructuradas**. Tesina de maestría, UNAM, 2007.
- [19] Dembo R., Eissenstat S. y Steihaug T., **Truncated Newton Algorithms for large-scale optimization**, Math Programing vol 26. 1983.
- [20] Dennis, J.E. y Schnabel R.B. **Numerical Methods for Unconstrained Optimization and Nonlinear Equations**. SIAM, Philadelphia, 1996.
- [21] De la Cruz, U.A., **Mallas Adaptivas Armónicas**, Tesis de Maestría, UJAT, 2005.
- [22] Domínguez-Mota, F.J. **Sobre la generación variacional discreta de mallas casiortogonales en el plano**, Tesis de Doctorado, Facultad de Ciencias, UNAM, Mexico, 146 p 2005 .
- [23] García, I.D. **Métodos de optimización de gran escala y algunas aplicaciones a funciones parcialmente separables**. Tesis de Maestría, Facultad de Ciencias U.N.A.M., 1999.
- [24] González, F.G. **Generación en Mallas en Regiones Planas Irregulares**. Tesis de Licenciatura, Universidad Autónoma de Yucatán, 1994.
- [25] González, F.G. **Un sistema automático para la generación numérica de mallas basado en los nuevos funcionales de área y suavidad**. Tesis de Maestría, Universidad Nacional Autónoma de México, 2003.

- [26] Hernández M.V., **Curvas NURBS: Una aplicación del suavizamiento de los vértices de un polígono**, Vínculos Matemáticos No. 209. Facultad de Ciencias, UNAM, 1994.
- [27] Hernández M.V., Estrada S. J. **Sampling points on regular parametrics curves with control of the distribución** in Computer Aided Geometric Design, Vol XX, No 6, 2003 pp 363-382.
- [28] Ivanenko S.A. y CharakhchÁ.A., **Curvilinear Grids of Convex Quadrilaterals**. U.S.S.R. Comp. Maths. Phys., Vol 28, No 2, 1998.
- [29] Ivanenko S.A. y CharakhchÁ.A., **A Variational Form of Winslow Grid Generator**. Journal of computational Physics, 136, pp 385-398. 1997.
- [30] Ivanenko S.A. **Harmonic maps in Handbook of grid Generation**. CRC Press, INC.
- [31] Kelley C., **Iterative Methods for Optimization**. SIAM Publications, 1999.
- [32] Knupp, P. **Intrinsic algebraic grid generation**. Mathematical aspects of numerical grid generation, J.E. Castillo Ed., 1991.
- [33] Knupp, P. **A robust elliptic grid generator**. J. Comp- Phys. 100, 409-418, 1992.
- [34] Knupp, P. y Steinberg, S. **Fundamentals of grid generation**. CRC Press, 1992.
- [35] Knupp, P. **Mesh Generations using vector.fields**, J. Comp- Phys. 119, 142-148, 1995.
- [36] Knupp, P. **Jacobian-weighted Elliptic Grid Generation**, SIAM J.Sci. Comput., Vol 17, No 6, pp 1475-1490, 1996.
- [37] Leal, L.G. y Ryskin, G. **2d orthogonal grid generation**. Journal of Computational Physics 50, 1983.
- [38] Nocedal J., y Wright J., **Numerical Optimization**, Springer Series in Operations Research, 1999.
- [39] M. Farrashkhalvat and J.P Miles. **Basic structured grid generations with and introduction to unstructured grid generation**, Oxford, 231p. 2003.
- [40] More J. J. and Chih-jen Lin, **Newton's method for large bound-constrained optimization problems**, SIAM Journal on Optimization, Volume 9, Number 4, pp. 1100-1127, 1999.
- [41] Ojeda C, R. B. **Métodos directos para la generación de redes en regiones planas**. Tesis de Maestría, Universidad Nacional Autónoma de México, 1991.

-
- [42] Roache, P.J. y Steinberg, S. **Variational grid generation**. Num. Meth. for P.D.E.s 2, 71-96, 1986.
- [43] Rivera H. A. **Fundamentos de curvas de Bezier racionales y cálculo numérico de la reparametrización de curvas** . Tesis de Maestría, Universidad Nacional Autónoma de México, 2005.
- [44] Rivera H. A. **Un punto de vista sobre las cónicas y su uso en el suavizamiento de polígonos**. Tesis de Licenciatura, Universidad Nacional Autónoma de México, 1999.
- [45] Senn, James A, **Análisis y diseño de sistemas de información**, McGraw-Hill, México, 942 pp., 1992.
- [46] Tinoco, J.G. **Funcionales Discretos para la Generación de Mallas Suaves y convexas sobre Regiones Planas Irregulares**. Tesis de Doctorado, CIMAT, 1997.
- [47] Thompson J.F, Warsi Z.U.A., y Mastin C.W., **Numerical Grid Generation: Foundations and Applications**, North-Holland, Elsevier, New York, 1985.
- [48] Toledo, F. **Métodos Numéricos para Mapeo Conforme con Aplicación a la Generación de Mallas**. Tesis de Maestría, Facultad de Ciencias, U.N.A.M., 1995.
- [49] Vázquez M.L. A. **Generación de mallas por métodos variacionales continuos**. Tesis de Licenciatura, Universidad Nacional Autónoma de México, 1997.
- [50] Winslow, A.M. **Numerical solution of quasilinear poisson equation in nonuniform triangle mesh**. J. Comput. Phys. 1(2), p. 149-172, 1966.
- [51] Whitten, Jeffrey L. **Análisis y diseño de sistemas de información**, México, D.F. McGraw-Hill Interamericana, 897 pp. 2003.