ICIMAF
MINISTERIO DE LA CIENCIA, TECNOLOGIA Y MEDIO AMBIENTE
GRUPO DE METODOS NUMERICOS

GENERACION NUMERICA DE REDES USANDO NEWTON TRUNCADO

TESIS PRESENTADA EN OPCION AL GRADO CIENTIFICO
DE DOCTOR EN CIENCIAS MATEMATICAS

AUTOR: LIC. J. LONGINA CASTELLANOS NODA

TUTOR: DR. PABLO BARRERA SANCHEZ
ASESOR: DR. ANGEL A. PEREZ DOMINGUEZ

LA HABANA
1995

Para mis hijos

Por mis hijos

AGRADECIMIENTOS

A mis padres

A mi esposo

A mi familia

A mis tutores

A mis amigos

A mis compañeros del Grupo de Métodos Numéricos y Estadística

Al ICIMAF

A CONACYT de México

A ICTP de Trieste, por el RESEARCH GRANT No.93-069

R E S U M E N

El trabajo que se presenta abarca dos aspectos fundamentales:
1) La obtención de funcionales discretos que generen mallas en regiones planas irregulares con propiedades adecuadas.
2) Investigación e implementación de métodos eficientes para optimización de gran escala adaptados al objetivo 1.
Se estudian los distintos funcionales variacionales que aparecen en la literatura, así como se formaliza un procedimiento general para la discretización de un funcional. Se establecen propiedades de los funcionales discretos, que ayudan a saber que se puede esperar del proceso de minimización y se realiza la regularización del funcional de suavidad. Como se minimiza una función de muchas variables, pues esta depende de los nodos interiores de la red, se estudian varios tipos de métodos de optimización y se reportan las experiencias numéricas con los mismos en la generación de redes con los distintos funcionales.
De los métodos de optimización estudiados, se hace una implementación original de los Newton Truncado, y se logra que sea de las que mejor responden al problema, dándose resultados numéricos y aspectos de su implementación y uso práctico.

I N D I C E

I N T R O D U C C I O N

La generación numérica de redes o mallas curvilíneas sobre regiones bidimensionales arbitrarias, es una de las herramientas más útiles en la solución de ecuaciones en derivadas parciales.
En las técnicas de generación de redes está implícita una transformación de un cierto dominio canónico como por ejemplo, un cuadrado o un rectángulo, a un dominio físico o real. La imagen de una malla en el dominio canónico será otra sobre el dominio físico. En general el problema puede plantearse de la siguiente manera:

"Dada una región $\Omega \subset R^2$, se desea encontrar una transformación suave $\bar{c}$ de Ω sobre el cuadrado unitario B y viceversa, una transformación suave $\bar{d}$ del cuadrado unitario B a la región Ω tal que

$$\bar{d}(\partial B) = \partial\Omega \quad ; \quad \bar{c}(\partial\Omega) = \partial B \qquad (I)$$

donde ∂B y $\partial\Omega$ son los contornos de B y Ω, respectivamente.
En resumen, se deben resolver los dos problemas siguientes:

(a) Hallar $\bar{d}$: $B \longrightarrow \Omega$, $\bar{d}(\xi,\eta) = (x(\xi,\eta), y(\xi,\eta))$
tal que $\bar{d}(\partial B) = \partial\Omega$ y $\bar{d}(B) \subset \Omega$.

(b) Hallar $\bar{c}$: $\Omega \longrightarrow B$, $\bar{c}(x,y) = (\xi(x,y), \eta(x,y))$
tal que $\bar{c}(\partial\Omega) = \partial B$ y $\bar{c}(\Omega) \subset B$.
(ver fig.1)

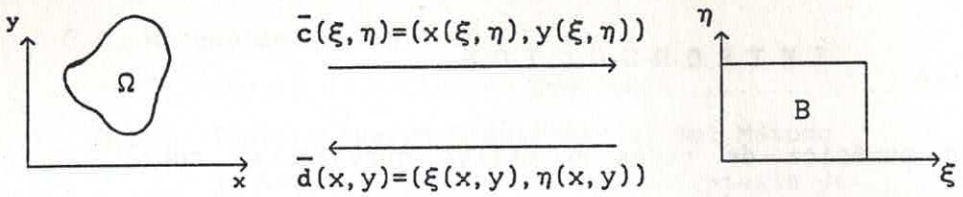


fig. 1. Transformación de Ω a B y viceversa

Para que la transformación esté bien definida debe ser biunívoca, de forma que un punto del espacio lógico se transforma en un único punto del espacio físico y cada punto del espacio físico Ω es la imagen de un punto del espacio canónico o lógico. Además, el requerimiento de suavidad se refiere a las derivadas de la transformación, es decir cada una de las funciones coordenadas $x(\xi,\eta)$, $y(\xi,\eta)$, debe ser continua y tener derivadas continuas como funciones de ξ y η, y si B está definido como un objeto cerrado, pues incluye su frontera, esto implica que la transformación también tiene que ser suave en la frontera y en particular, esto trae como consecuencia que las esquinas del objeto lógico se transforman a las esquinas en el objeto físico (ver fig. 2).

fig. 2. Topología de la frontera

En regiones simples es fácil encontrar dichas transformaciones, por ejemplo, si Ω es la región comprendida entre dos círculos concéntricos y B es un cuadrado unitario:

$$\Omega = \{(x,y) \;/\; r_1 \leq x^2 + y^2 \leq r_2\} \quad ; \quad r_1 < r_2$$

$$B = \{(\xi,\eta) \;/\; 0 \leq \xi,\eta \leq 1\}$$

haciendo

$$\varphi(\xi)=2\pi\xi \quad ; \quad r(\eta)=r_1+(r_2-r_1)\eta$$

se tiene

$$x(\xi,\eta)=r(\eta)\cos(\varphi(\xi)) \quad ; \quad y(\xi,\eta)=r(\eta)\sin(\varphi(\xi))$$

$$\eta(x,y)=\frac{(x^2+y^2-r_1)}{r_2-r_1} \quad ; \quad \xi(x,y)=\frac{1}{2\pi}\arctan(\frac{y}{2})$$

es decir, se tienen analíticamente las expresiones de $\bar{d}$ y $\bar{c}$.
Sin embargo, lo que se necesita es obtener automáticamente para una región arbitraria, los nodos de la red sobre la región física o real tal que dicha red tenga determinadas propiedades. Como no es posible tener una transformación explícita para cualquiera sea la región, lo que se trata es de obtener valores discretos de la transformación que resultan de la minimización de algún funcional cuyo valor óptimo satisface la propiedad que se desea obtener de la malla, es decir, en la práctica lo que se tiene por resultado no es la transformación propiamente dicha, sino los nodos sobre la región física.
Los métodos de generación de redes pueden agruparse convenientemente como sigue:

1) Métodos de Interpolación o Algebraicos, que usan los polinomios de Lagrange para calcular los nodos de la red.
2) Métodos Variacionales: Se construyen funcionales que miden determinadas propiedades geométricas deseadas de la red, la cual se obtiene:

i) Resolviendo las ecuaciones diferenciales parciales de Euler-Lagrange.

ii) Minimizando el funcional discretizado, es decir, se minimiza el análogo discreto de la formulación

variacional.

Los métodos que se usan en este trabajo para la generación de redes son del tipo ii) y la red se obtiene usando algoritmos de optimización de gran escala sobre funcionales discretos, por lo que los objetivos fundamentales que se proponen tienen que ver con la obtención de funcionales discretos que generen mallas en regiones planas irregulares, y la implementación de métodos eficientes para optimización de gran escala que calculen la malla óptima.

La exposición se hace a través de cuatro capítulos que tratan los siguientes tópicos:

En el capítulo I se hace la presentación de las formulaciones variacionales continuas usadas por distintos autores y la solución del problema a través de las ecuaciones de Euler-Lagrange

El capítulo II presenta los funcionales discretos y su relación con las formulaciones continuas, además se construye un procedimiento general para obtener una formulación discreta a partir de una continua; se plantean y demuestran propiedades de los funcionales discretos; se expone la forma en que se logró una regularización del funcional de Suavidad discreto y las propiedades de las redes que generan los funcionales que se proponen.

El capítulo III aborda el problema de optimización de gran escala, dando la descripción de los métodos de optimización que se usan en la experimentación numérica con el problema de mallas. En particular, se hace énfasis en los de Newton Truncado en el aspecto teórico y la implementación que se propone para el caso general de optimización.

El capítulo IV describe la implementación computacional de los métodos de optimización para el caso de generación de redes y da los resultados de la experimentación numérica con los diferentes métodos y funcionales.

C A P I T U L O I

METODOS VARIACIONALES

En la práctica la generación de redes se realiza minimizando simultáneamente varias propiedades medibles como son: la suavidad de las líneas verticales y horizontales, la variación del área de las celdas y la ortogonalidad de las líneas. Este capítulo se dedicará a presentar los diferentes funcionales usados con mayor frecuencia en la literatura, y la forma en que se utilizan por distintos autores para el problema que se quiere resolver.

1.1. Planteamiento variacional del problema de generación de redes.

El problema (I) puede plantearse en forma de integrales que midan las propiedades deseadas de la transformación. Así se tiene que para medir la suavidad global de la transformación, es decir, la variación de la longitud en los segmentos de las curvas de nivel de ξ y η, se utiliza la integral (Winslow [65])

$$I_s(\bar{c}(x,y)) = \int_\Omega \left[\|\xi\|^2 + \|\eta\|^2 \right] dxdy \quad , \qquad (1.1)$$

la ortogonalidad de los vectores tangentes a las líneas de la malla se mide por

$$I_o(\bar{c}(x,y)) = \int_\Omega (\xi . \eta)^2 / \bar{J}^3 \; dxdy \quad , \qquad (1.2)$$

y la variación ponderada del área se tiene mediante

$$I_a(\bar{c}(x,y)) = \int_\Omega w \,/\, \bar{J}\, dxdy \qquad , \qquad (1.3)$$

donde $w=w(x,y)$ es una función dada y $\bar{J}$ es el determinante de la matriz Jacobiana de la transformación $\bar{c}(x,y)$, es decir,

$$\bar{J}(x,y) = \begin{vmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{vmatrix}$$

$$= \xi_x \eta_y - \xi_y \eta_x$$

el cual es distinto de cero dado que se supone que la transformación $\bar{c}$ es inyectiva (ver Apostol [2]).

Resulta más cómodo trabajar sobre el espacio lógico o de cálculo B del conjunto de índices que en el espacio físico o real de la región Ω sobre la que se desea obtener la malla, por lo que se debe construir la transformación inversa $\bar{d}$: B $\rightarrow$ Ω; $\bar{d}=(x(\xi,\eta),y(\xi,\eta))$, lo que se consigue haciendo un cambio de variables. Para ello se necesitan las expresiones de las derivadas parciales ξ_x, ξ_y, η_x y η_y (ver Brackbill y Saltzman [11]):

Se tiene que

$$\frac{\partial x}{\partial x} = 1 \qquad , \text{ pero también}$$

$$\frac{\partial x}{\partial x} = x_\xi \xi_x + x_\eta \eta_x \quad ,$$

de donde se obtiene $\xi_x = \dfrac{1 - x_\eta \eta_x}{x_\xi}$ (1.4)

Además,

$$\frac{\partial y}{\partial x} = 0 \quad , \text{ pero también}$$

$$\frac{\partial y}{\partial x} = y_\xi \xi_x + y_\eta \eta_x \quad ,$$

y entonces $\eta_x = - \dfrac{y_\xi \xi_x}{y_\eta}$, (1.5)

y $$\frac{\partial x}{\partial y} = 0 \quad , \text{ pero también}$$

$$\frac{\partial x}{\partial y} = x_\xi \xi_y + x_\eta \eta_y \quad ,$$

por lo tanto $\xi_y = - \dfrac{x_\eta \eta_y}{x_\xi}$ (1.6)

y $$\frac{\partial y}{\partial y} = 1 \quad , \quad \text{pero también}$$

$$\frac{\partial y}{\partial y} = y_\xi \xi_y + y_\eta \eta_y \quad , \tag{1.7}$$

luego $\eta_y = \dfrac{1 - y_\xi \xi_y}{y_\eta}$

A partir de (1.4), (1.5), (1.6) y (1.7) se tiene que

$$\xi_x = + \frac{y_\eta}{J} \qquad \eta_x = - \frac{y_\xi}{J} \quad ,$$

$$\xi_y = - \frac{x_\eta}{J} \qquad \eta_y = + \frac{x_\xi}{J} \quad , \tag{1.8}$$

donde $J(\xi,\eta) = x_\xi y_\eta - x_\eta y_\xi$ es el Jacobiano de $\bar{\mathbf{d}}$, y como se supone una transformación biunívoca, debe ser distinto de cero.

Con (1.8) se puede efectuar el cambio de variables de los funcionales I_s , I_a y I_o , de forma que se tome la integral sobre el cuadrado B, es decir, el espacio lógico. Recuérdese que si

(ver Apostol [2]):

$$\xi = \xi(x,y) \quad , \quad x = x(\xi,\eta) \quad ,$$

$$\eta = \eta(x,y) \quad , \quad y = y(\xi,\eta) \quad ,$$

se conoce que:

$$\iint_{\Omega} f(\xi(x,y),\eta(x,y))\, dxdy = \iint_{B} f(x(\xi,\eta),y(\xi,\eta))J(\xi,\eta)\, d\xi d\eta$$

A partir de esto se pueden realizar las transformaciones al espacio lógico de las integrales propuestas:

1) Para I_s se tiene que

$$f(\bar{c}(x,y)) = \|\xi\|^2 + \|\eta\|^2 \quad ,$$

$$I_s = \iint_{\Omega} f(\bar{c}(x,y))dxdy = \iint_{B} \| \xi(x,y)\|^2 + \| \eta(x,y)\|^2 \, dxdy$$

$$= \iint_{B} f(\bar{d}(\xi,\eta))\, J(\xi,\eta)\, d\xi d\eta$$

$$= \iint_{B} \left(\frac{y_\eta^2}{J^2} + \frac{x_\xi^2}{J^2} + \frac{y_\xi^2}{J^2} + \frac{x_\eta^2}{J^2} \right) J \; d\xi d\eta \quad ,$$

$$\text{por tanto, } I_s(\bar{d}(\xi,\eta)) = \iint_{B} \left(\frac{x_\xi^2 + y_\eta^2 + x_\xi^2 + y_\eta^2}{J} \right) d\xi d\eta$$

$$\Leftrightarrow \qquad I_s[\bar{d}] = \int_0^1 \int_0^1 \frac{(\|\nabla x\|^2 + \|\nabla y\|^2)}{J} \, d\xi d\eta \quad , \qquad (1.9)$$

En (1.9) se tiene al Jacobiano como una función de peso $w(x,y)=J(x,y)$, este actúa dando mayor suavidad a las líneas (los gradientes son menores) allí donde el Jacobiano es mayor. Podría entonces desearse que el Jacobiano no controlara la suavidad, es decir, poner $w(x,y)=cte$, por ejemplo 1, y asi se tiene la expresión del funcional conocido como de longitud (Steinberg y Roache [60]), ya que da por resultado líneas de longitudes

proporcionales:

$$I_{\ell}[\bar{d}] = \int_0^1\int_0^1 (\|\nabla x\|^2 + \|\nabla y\|^2)\, d\xi d\eta \ . \qquad (1.10)$$

2) Para I_a se tiene que

$$f = \frac{w(x,y)}{\bar{J}(x,y)}$$

$$I_a = \iint_{\Omega} f(\bar{c}(x,y))\, dxdy = \iint_{\Omega} \frac{w(x,y)}{\bar{J}(x,y)}\, dxdy \ .$$

Como $J\bar{J}=1$ entonces se puede poner $f=w(x,y)J(\xi,\eta)$,
y se puede escribir

$$I_a = \iint_{\Omega} w(x,y)\ J(\xi,\eta)\, dxdy = \iint_{B} f(\bar{d}(\xi,\eta))\ J(\xi,\eta)\, d\xi d\eta \ .$$

Para $w(x,y)=1$ se tiene que

$$I_a = \iint_{B} J(\xi,\eta)J(\xi,\eta)\, d\xi d\eta \ .$$

Por lo tanto, $I_a[\bar{d}] = \iint_{B} \bar{J}^2(\xi,\eta)\, d\xi d\eta$. $\qquad (1.11)$

3) Para I_o se tiene que

$$f = \frac{(\xi.\eta)^2}{\bar{J}^3\ (x,y)} = (\xi.\eta)^2\ J^3(\xi,\eta) = (\xi_x\eta_x + \xi_y\eta_y^2)\ J^3(\xi,\eta) \ ,$$

$$I_o = \iint_{\Omega} f(\bar{c}(x,y))dxdy = \iint_{\Omega} (\xi_x\eta_x + \xi_y\eta_y)^2 J^3(\xi,\eta)\, dxdy \ ,$$

$$I_o = \iint_{B} \left(\frac{x_\eta x_\xi + y_\eta y_\xi}{J^2(\xi,\eta)} \right)^2 J^3(\xi,\eta)J(\xi,\eta)\, d\xi d\eta \ .$$

Por lo tanto, $I_o[\bar{d}] = \iint_{B} (x_\eta x_\xi + y_\eta y_\xi)^2\, d\xi d\eta$.

De esta forma se tienen funcionales definidos sobre una transformación $\bar{d}$, en la que sólo se imponen condiciones que debe

satisfacer (ver Introducción), pero que será la solución continua de la minimización de funcionales que miden una propiedad de la malla y por lo tanto, su óptimo (es decir, $\bar{\mathbf{d}}$) tiene esa propiedad. El problema de hallar ese mínimo en la práctica se resuelve de diferentes formas, como se verá a continuación.

1.2. Ecuaciones de Euler-Lagrange

Varios autores han usado los funcionales anteriores para generar redes a través de la solución, por algún método (ver epígrafe 1.3), de las ecuaciones correspondientes de Euler Lagrange. Es decir, sea la función $f(x(\xi,\eta),y(\xi,\eta))$, y se desea hallar un punto crítico de

$$\iint_B f(x(\xi,\eta),y(\xi,\eta))d\xi d\eta \quad , \tag{1.13}$$

entonces hay que hallar la solución del sistema de ecuaciones diferenciales parciales:

$$\frac{\partial}{\partial\xi}\left(\frac{\partial f}{\partial x_\xi}\right) + \frac{\partial}{\partial\eta}\left(\frac{\partial f}{\partial x_\eta}\right) - \frac{\partial f}{\partial x} = 0 \quad , \tag{1.14}$$

$$\frac{\partial}{\partial\xi}\left(\frac{\partial f}{\partial y_\xi}\right) + \frac{\partial}{\partial\eta}\left(\frac{\partial f}{\partial y_\eta}\right) - \frac{\partial f}{\partial y} = 0 \quad , \tag{1.15}$$

que no son más que las ecuaciones de Euler-Lagrange para la integral (1.13) (ver Apostol [2]).

Una forma de generar las redes, es decir, de obtener los nodos sobre Ω que son la solución numérica del problema de minimización de algunos de los funcionales presentados, es resolviendo las ecuaciones de Euler-Lagrange en forma iterativa usando aproximaciones finitas de las derivadas. Como (ξ,η) son variables continuas que toman valores enteros en los nodos de la red de cálculo, ellas forman una red rectilínea espaciada uniformemente. Las derivadas con respecto a las variables independientes se calculan fácilmente sobre esta malla, donde cada nodo de la red

$(\xi,\eta)=(i,j)$. El valor de las derivadas en los nodos se aproxima entonces mediante las ecuaciones en diferencias (ver Brackbill y Saltzman [11]):

$$x_\xi \simeq (x_{i+1,j} - x_{i-1,j})$$

$$x_\eta \simeq (x_{i,j+1} - x_{i,j-1})$$

y las segundas derivadas por

$$x_{\xi\xi} \simeq x_{i+1,j} - 2x_{i,j} + x_{i-1,j}$$

$$x_{\xi\eta} \simeq (x_{i+1,j+1} + x_{i-1,j-1} - x_{i+1,j-1} - x_{i-1,j+1})$$

$$x_{\eta\eta} \simeq x_{i,j+1} - 2x_{i,j} + x_{i,j-1}$$

y de manera similar para y. Las ecuaciones algebraicas en cada nodo resultan de la sustitución de las derivadas por las diferencias, con lo que todo se reduce a resolver un sistema algebraico de ecuaciones no lineales.

1.3. Breve reseña del uso de los funcionales por distintos autores

Winslow [65] usa el funcional de suavidad (1.1) para obtener redes y para ello resuelve las ecuaciones de Euler usando un método de sobrerrelajación. Posteriormente, Brackbill y Saltzman [11] revisando las ideas de Winslow, proponen usar simultáneamente varios funcionales, de forma que su generador de redes optimice distintas propiedades a la vez, dejando a elección del usuario el peso a dar a cada una en su solución. Concretamente, proponen minimizar

$$I_{comb} = I_s + \lambda_a I_a + \lambda_o I_o \quad ,$$

donde λ_a, λ_o son valores positivos, para lo cual resuelven las ecuaciones de Euler, cuyas derivadas sustituyen por diferencias

según se vió en el acápite 1.2 y luego aplican las iteraciones de Jacobi [19].

Steinberg y Roache [60] proponen minimizar

$$I_{comb} = \lambda_\ell I_\ell[\bar{d}] + \lambda_a I_a[\bar{d}] + \lambda_o I_o[\bar{d}]$$

donde λ_ℓ, λ_a y λ_o son valores normalizados,

$$\lambda_\ell + \lambda_a + \lambda_o = 1 \quad .$$

Knupp [41] en su trabajo señala, que a partir de su experiencia práctica con la combinación de funcionales, resulta que la dada por $\lambda_\ell=0$ y $\lambda_a=\lambda_o=1/2$, da lugar a un generador, que aunque no es elíptico, sorprendentemente da en la práctica redes suaves, y como el peso dado a la uniformidad de las celdas (λ_a) es igual al peso para la ortogonalidad de las líneas (λ_o), se crea además un compromiso entre ambas propiedades, obteniéndose así un funcional que presenta las tres propiedades geométricas que se necesitan para una red.

De manera explícita el funcional de área-ortogonalidad es:

$$\begin{aligned} I_{ao}[\bar{d}] &= 1/2\ I_a[\bar{d}] + 1/2\ I_{ort}[\bar{d}] \\ &= 1/4 \int_0^1\int_0^1 (x_\xi^2 + y_\xi^2) \cdot (x_\eta^2 + y_\eta^2)\, d\xi d\eta \quad , \end{aligned} \tag{1.16}$$

Hasta aquí se han resumido las expresiones continuas de los funcionales que se usarán para generar mallas y la solución de generación de mallas a través de las ecuaciones de Euler-Lagrange. En los siguientes capítulos se abordarán los aspectos que conciernen más específicamente a este trabajo.

C A P I T U L O II

METODOS DISCRETOS DIRECTOS

En el capítulo anterior se vió la forma de obtener redes bidimensionales a partir de la minimización de distintos funcionales, surgidos de la formulación variacional de algunas propiedades de la transformación del espacio lógico al físico, la que se realiza por medio de la discretización de las ecuaciones de Euler-Lagrange correspondientes. En este trabajo la forma de obtener las redes es usando métodos de optimización para encontrar los mínimos de los funcionales, que previamente deben ser discretizados para que sea factible la aplicación de un método numérico, de ahí que en este capítulo se formalice un procedimiento para la discretización de un funcional continuo definido sobre el cuadrado unitario, haciéndose la discretización de los funcionales continuos vistos en el capítulo I. Se presenta además el funcional de Castillo que él obtiene directamente midiendo propiedades de la malla sobre los nodos de la región física, y luego la extensión que del mismo hacen Barrera y Pérez para el caso del funcional de área, siguiendo algunas ideas de Ivanienko [36], se demuestran resultados que caracterizan las soluciones óptimas de los funcionales discretizados.
En lo sucesivo se usará la siguiente notación:

P^t ⟶ Para P de orden mxn, indica su transpuesta, es decir, la matriz de nxm

$\frac{\partial f}{\partial P}$ ⟶ Vector de orden 2x1 de la primera derivada parcial de f respecto de las coordenadas del punto P, es decir,

$$\frac{\partial f}{\partial P} = \begin{pmatrix} \frac{\partial f}{\partial x_P} \\ \frac{\partial f}{\partial y_P} \end{pmatrix} \quad \text{para} \quad P^t = (x_p, y_p)$$

$\frac{\partial^2 f}{\partial P^2}$ ⟶ Matriz de orden 2x2 de la segunda derivada parcial de f respecto a las coordenadas del punto P, es decir,

$$\frac{\partial^2 f}{\partial P^2} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_P^2} & \frac{\partial^2 f}{\partial y_P \partial x_P} \\ \frac{\partial^2 f}{\partial x_P \partial y_P} & \frac{\partial^2 f}{\partial y_P^2} \end{bmatrix}$$

$\frac{\partial^2 f}{\partial A \partial P}$ ⟶ Matriz de orden 2x2 de la segunda derivada parcial de f primero con respecto a las coordenadas de P y luego con respecto a las de A, es decir,

$$\frac{\partial^2 f}{\partial A \partial P} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_A \partial x_P} & \frac{\partial^2 f}{\partial y_A \partial x_P} \\ \frac{\partial^2 f}{\partial x_A \partial y_P} & \frac{\partial^2 f}{\partial y_A \partial y_P} \end{bmatrix} \quad \text{para } A^t = (x_A, y_A)$$

I_k ⟶ Matriz Identidad de orden k

$\otimes$ ⟶ Producto Kronecker de matrices [19]

T_{m-2} ⟶ Es la matriz tridiagonal de orden m-2 siguiente

$$T_{m-2} = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & & \\ -1 & 2 & -1 & 0 & \cdots & & \\ 0 & -1 & 2 & -1 & \cdots & & \\ & & \ddots & & & & \\ & & & \ddots & & & \\ & & & & \ddots & & \\ & & & & & & 0 \\ & & & & & 0 & -1 \\ & & & & & 0 & -1 \quad 2 \end{bmatrix}$$

J_2 ⟶ Es la matriz de orden 2x2

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Se van a considerar en lo sucesivo sólo regiones poligonales, por lo que se omitirá el término cuando se hace referencia a ellas. Además en este trabajo sólo interesan los aspectos de optimización sobre los nodos interiores de la red, ya que la frontera de la región está fija, por lo que no se va a tratar ni la forma de dar las coordenadas de la frontera ni la generación de una red inicial. Para esos aspectos puede consultarse Ojeda [49]. Se supone que se tiene una red inicial dada sobre la región Ω.

Definición.- Dada una región $\Omega \subset \mathbb{R}^2$, se dice que es una región poligonal, si su frontera ($\partial\Omega$) es una poligonal cerrada.

Definición.- Una red en una región poligonal Ω, es una subdivisión de ésta en cuadriláteros. Los vértices de los cuadriláteros son llamados los puntos de la red y los cuadriláteros sus celdas (ver fig. 2.1).

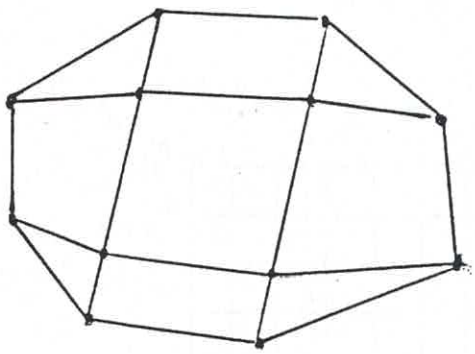


fig. 2.1. Malla sobre Ω

Otra forma de definir una red es la siguiente:

Definición.- Una red de dimensión mxn en una región Ω, está formada por celdas y un subconjunto $\{P_{i,j}\}$ de mxn puntos que tienen la siguiente propiedad:

$$\partial\Omega = \text{Poligonal}(P_{1,1}, P_{1,2}, \ldots, P_{1,n}, P_{2,n}, \ldots, P_{m,n},$$
$$P_{m,n-1}, \ldots, P_{m,1}, P_{m-1,1}, \ldots, P_{1,1})$$

es decir, $\partial\Omega$ es la frontera de la red $\{P_{i,j}\}$ (ver fig.2.2)

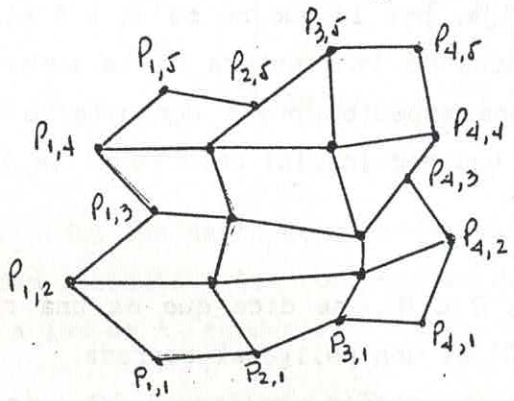
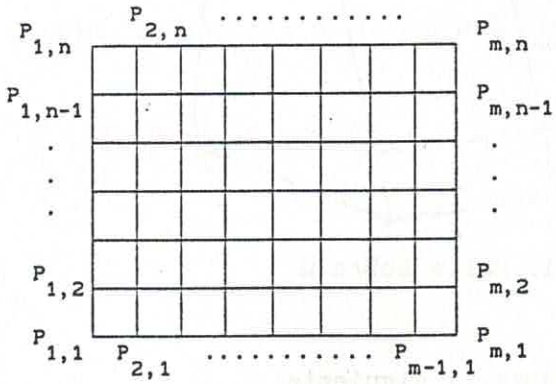


fig. 2.2. Frontera de Ω

Para simplificar, sin perder generalidad, la descripción de los funcionales se hace para el caso en que Ω es un dominio rectangular (fig.2.3)



Red Rectangular

fig. 2.3

Definición.- Dada una red con un conjunto de puntos $\{P_{i,j}\}$ i=1,m,

j=1,n, en Ω , se dice que:

i) Los conjuntos

$$\{P_{1,1}, P_{2,1}, P_{3,1}, \ldots, P_{m-1,1}, P_{m,1}\}$$

y

$$\{P_{1,n}, P_{2,n}, P_{3,n}, \ldots, P_{m-1,n}, P_{m,n}\}$$

son los puntos de la **frontera horizontal.**

ii) Los conjuntos

$$\{P_{1,1}, P_{1,2}, P_{1,3}, \ldots, P_{1,n-1}, P_{1,n}\}$$

y

$$\{P_{m,1}, P_{m,2}, P_{m,3}, \ldots, P_{m,n-1}, P_{m,n}\}$$

son los puntos de la **frontera vertical.**

iii) Los puntos interiores de la red son el conjunto ordenado siguiente:

$$\{P_{2,2}, P_{2,3}, P_{2,4}, \ldots, P_{2,n-1},$$
$$P_{3,2}, P_{3,3}, P_{3,4}, \ldots, P_{3,n-1},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$P_{m-1,2}, P_{m-1,3}, P_{m-1,4}, \ldots, P_{m-1,n-1}\} \quad ,$$

formado por (m-2)x(n-2) puntos.
Sea **z** el vector columna formado por las coordenadas de los puntos interiores de la red en el orden indicado, entonces

si $P^t_{r,s} = (x_{r,s}, y_{r,s})$

$$z^t = (x_{2,2}, y_{2,2}, x_{2,3}, y_{2,3}, \ldots, x_{m-1,n-1}, y_{m-1,n-1})$$

y **z** es de orden 2(m-2)(n-2) x 1.

Considérense los puntos $P_{i,j}$ como vectores de dimensión 2x1, entonces:

$$z^t=\{P^t_{2,2},P^t_{2,3},P^t_{2,4},\ldots,P^t_{2,n-1},$$
$$P^t_{3,2},P^t_{3,3},P^t_{3,4},\ldots,P^t_{3,n-1},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$P^t_{m-1,2},P^t_{m-1,3},P^t_{m-1,4},\ldots,P^t_{m-1,n-1}\}$$

donde

$$z^t_k=P^t_{i,j}=(x_{i,j},y_{i,j})$$

(ver fig 2.4)

z_2	P_{23}	z_4	z_6	z_8
z_1	P_{22}	z_3	z_5	z_7

fig. 2.4. Orden de los nodos sobre una red de 6x4

Es fácil expresar a k como función de i,j, es más, se puede verificar que

$$k=\varphi(i,j)=(n-2)(i-2)+j-1 \qquad (2.1)$$

donde $2\leq i\leq m-1$ y $2\leq j\leq n-1$

De todas las redes de dimensión mxn con la misma frontera, se tiene el interés de determinar aquellas cuyos puntos interiores minimizan un funcional dado. Es decir, si f es un funcional definido sobre las redes de mxn, entonces, si se consideran solamente los puntos interiores como las variables, se tiene que

$f:\mathbb{R}^N \longrightarrow \mathbb{R}$, donde $N=2(m-2)x(n-2)$.
Lo que se quiere es determinar la red óptima $\hat{z}$ tal que minimice a f, pero para ello es necesario contar con funcionales tomados sobre los nodos de la red en la región física Ω.

2.1. PROCEDIMIENTO GENERAL PARA DISCRETIZAR UN FUNCIONAL CONTINUO

Se quiere formalizar un procedimiento que dado un funcional variacional continuo sobre el cuadrado unitario B, facilite la construcción de su análogo discreto tomado sobre los nodos de la región real Ω, es decir, se tiene

$$I[\bar{d}] = \int_0^1\int_0^1 f(x_\xi, x_\eta, y_\xi, y_\eta)\, d\xi d\eta$$

$$= \int_0^1\int_0^1 f(\bar{d}_\xi, \bar{d}_\eta, J(\xi,\eta))\, d\xi d\eta \qquad (2.3)$$

donde f es una función que refleja alguna propiedad de la malla, y se desea obtener una expresión discreta de $I[\bar{d}]$ tomada sobre los nodos de la malla en la región física, es decir, se quiere obtener el funcional discretizado como una función definida sobre las celdas, el que se puede definir como:

$$I_d[\{P_{i,j}\}] = \sum_{i,j} f(c_{i,j})$$

donde $c_{i,j}$ denota a la ij-ésima celda de la malla y

$$f(c_{i,j}) = f(P_{i,j}, P_{i+1,j}, P_{i+1,j+1}, P_{i,j+1}) \qquad (2.2)$$

es la función que mide la propiedad geométrica, pero tomada ahora sobre los cuatro vértices de la celda.
Para describir el proceso, se considera el caso especial cuando Ω es un cuadrilátero con vértices P,Q,R,S (ver fig. 2.5).

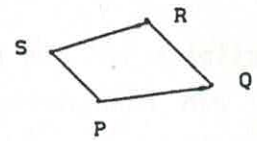


fig. 2.5. Cuadrilátero sobre Ω

La forma más natural de transformar un cuadrado en un cuadrilátero es usando una función bilineal

$$r(\xi,\eta) = A + B\xi + C\eta + D\xi\eta \quad . \tag{2.4}$$

Los vectores A, B, C y D se determinan a partir de los valores de $r(\xi,\eta)$ en los vértices del cuadrado, es decir, deben satisfacer las condiciones:

$$\begin{aligned} r(0,0)&=P \quad , \quad r(1,0)=Q \quad , \\ r(0,1)&=S \quad , \quad r(1,1)=R \quad , \end{aligned} \tag{2.5}$$

Luego

$$r(\xi,\eta)= P+(Q-P)\xi+(S-P)\eta+[(R-Q)+(P-S)]\xi\eta \tag{2.6}$$

y

$$r_\xi = (Q-P)+[(R-Q)+(P-S)]\eta \tag{2.7}$$

$$r_\eta = (S-P)+[(R-Q)+(P-S)]\xi \tag{2.8}$$

de donde

$$\begin{aligned} r_\xi(0,0)&=Q-P & r_\xi(1,0)&=Q-P \\ r_\eta(0,0)&=S-P & r_\eta(1,0)&=R-Q \\ r_\xi(0,1)&=R-S & r_\xi(1,1)&=R-S \\ r_\eta(0,1)&=S-P & r_\eta(1,1)&=R-Q \end{aligned} \tag{2.9}$$

y los Jacobianos en cada vértice quedan como

$$
\begin{aligned}
J(0,0) &= 2\ \text{area}(P,Q,S) \\
J(1,0) &= 2\ \text{area}(Q,R,P) \\
J(0,1) &= 2\ \text{area}(R,S,Q) \\
J(1,1) &= 2\ \text{area}(S,P,R)
\end{aligned}
\tag{2.10}
$$

Si ahora se aproxima $\bar{\mathbf{d}}(\xi,\eta)$ por $\mathbf{r}(\xi,\eta)$ entonces

$$
\begin{aligned}
f(\bar{\mathbf{d}}_\xi, \bar{\mathbf{d}}_\eta) &\approx f(\mathbf{r}_\xi, \mathbf{r}_\eta) \\
f(\bar{\mathbf{d}}_\xi, \bar{\mathbf{d}}_\eta, J(\bar{\mathbf{d}}(\xi,\eta))) &\approx f(\mathbf{r}_\xi, \mathbf{r}_\eta, J(\mathbf{r}(\xi,\eta)))
\end{aligned}
\tag{2.11}
$$

y usando (2.7) y (2.8) se tiene que

$$
\begin{aligned}
J(\mathbf{r}(\xi,\eta)) = \{&(x_Q - x_P) + [(x_R - x_Q) + (x_P - x_S)]\eta \} * \\
\{&(y_S - y_P) + [(y_R - y_Q) + (y_P - y_S)]\xi \} - \\
\{&(y_Q - y_P) + [(y_R - y_Q) + (y_P - y_S)]\eta \} * \\
\{&(x_S - x_P) + [(x_R - x_Q) + (x_P - x_S)]\xi \}
\end{aligned}
\tag{2.12}
$$

Con esto se puede construir una aproximación de $\bar{I}[\mathbf{d}]$ usando una regla de cuadratura:

$$
\begin{aligned}
I[\bar{\mathbf{d}}] \approx 1/4\ [&f(\mathbf{r}_\xi(0,0), \mathbf{r}_\eta(0,0), J(\mathbf{r}(0,0))) + \\
&f(\mathbf{r}_\xi(1,0), \mathbf{r}_\eta(1,0), J(\mathbf{r}(1,0))) + \\
&f(\mathbf{r}_\xi(1,1), \mathbf{r}_\eta(1,1), J(\mathbf{r}(1,1))) + \\
&f(\mathbf{r}_\xi(0,1), \mathbf{r}_\eta(0,1), J(\mathbf{r}(0,1)))\ ]
\end{aligned}
\tag{2.13}
$$

Si se le llama $f_d(P,Q,R,S)$ a esta aproximación, se tiene

$$f_d(P,Q,R,S) = 1/4\ [f(Q-P, S-P, \text{area}(P,Q,S)) + f(Q-P, R-Q, \text{area}(Q,R,P)) + f(R-S, R-Q, \text{area}(R,S,Q)) + f(R-S, S-P, \text{area}(S,P,R))] \quad (2.14)$$

Finalmente se obtiene un funcional discreto sobre una malla $\{P_{i,j}\}$ de dimensión mxn como la suma sobre todas las celdas $c_{i,j}$ de la malla

$$I_d(\{P_{i,j}\}) = \sum_{i,j} f_d(P_{i,j}, P_{i+1,j}, P_{i+1,j+1}, P_{i,j+1}) \quad (2.15)$$

2.1.1. FUNCIONALES DISCRETOS ASOCIADOS A ALGUN FUNCIONAL CONTINUO

A continuación se muestran las expresiones de algunos de los funcionales discretos más conocidos y reportados en la literatura.

FUNCIONAL DISCRETO DE LONGITUD

Este funcional es uno de los propuestos por Steinberg y Roache [60] dado poar (1.10) en el capítulo anterior, sólo que aqui se considera con un parámetro $\tau \geq 0$ que permite tener un mejor control de la tensión de las líneas verticales y horizontales de la malla obteniéndose la expresión

$$I_{\ell,\tau}[\bar{d}] = \int_0^1 \int_0^1 \left[\tau(x_\xi^2 + x_\eta^2) + (y_\xi^2 + y_\eta^2) \right] d\xi d\eta \quad , \quad (2.16)$$

en este caso se tiene que

$$f(\bar{d}(\xi,\eta)) = \tau x_\xi^2 + \tau x_\eta^2 + y_\xi^2 + y_\eta^2 \quad , \quad (2.17)$$

de donde

$$f(\mathbf{r}(0,0)) = \tau\|Q-P\|^2+\|S-P\|^2 \qquad f(\mathbf{r}(1,0)) = \tau\|Q-P\|^2+\|R-Q\|^2$$

$$f(\mathbf{r}(0,1)) = \tau\|R-S\|^2+\|S-P\|^2 \qquad f(\mathbf{r}(1,1)) = \tau\|R-S\|^2+\|R-Q\|^2$$

$$\therefore f_{\ell,\tau}(P,Q,R,S) = 1/2\left[\tau\|Q-P\|^2+\|S-P\|^2+ \tau\|R-S\|^2+\|R-Q\|^2\right]$$

$$= 1/2\ [\ \tau\ f_H(P,Q,R,S) + f_V(P,Q,R,S)\ ] \qquad (2.18)$$

FUNCIONAL DISCRETO DE AREA

La formulación variacional propuesta por Steinberg y Roache [60] para controlar el área de las celdas está dada por (1.11):

$$I_a[\bar{d}] = \int_0^1\int_0^1 J^2\, d\xi d\eta \quad , \qquad (2.19)$$

es decir

$$f(\bar{d}(\xi,\eta)) = (x_\xi y_\eta - x_\eta y_\xi)^2 \quad , \qquad (2.20)$$

y por consiguiente

$$f_a(P,Q,R,S) = 1/4\left[J^2(r(0,0))+J^2(r(0,1))+J^2(r(1,0))+J^2(r(1,1))\right]$$

$$= 1/2\left[\text{area}^2(P,Q,S) + \text{area}^2(Q,R,P) + \text{area}^2(R,S,Q) + \text{area}^2(S,P,R)\right] \qquad (2.21)$$

que sumando sobre los nodos de la red, da el funcional de área discretizado.

FUNCIONAL DISCRETO DE SUAVIDAD

El funcional que da la suavidad de las líneas de la malla (1.9):

$$I_s[\bar{d}] = \iint_B \frac{x_\xi^2 + y_\xi^2 + x_\eta^2 + y_\eta^2}{J}\, d\xi d\eta \quad , \qquad (2.22)$$

y su discretización fue obtenida por Ivanienko [36] y es la

siguiente

$$f_S(P,Q,R,S)=1/4\left\{\frac{\|P-S\|^2+\|Q-P\|^2}{2area(P,Q,S)}+\frac{\|Q-P\|^2+\|R-Q\|^2}{2area(Q,R,P)}+\right.$$
$$\left.\frac{\|S-R\|^2+\|R-Q\|^2}{2area(R,S,Q)}+\frac{\|P-S\|^2+\|S-R\|^2}{2area(S,P,R)}\right\} \qquad (2.23)$$

FUNCIONAL DISCRETO DE ORTOGONALIDAD

El funcional (1.12) busca la ortogonalidad de las líneas de la malla (1.12)

$$I_o[\bar{d}] = \iint_B (\bar{d}_\xi . \bar{d}_\eta)^2 \, d\xi d\eta = \iint_B (x_\xi x_\eta + y_\xi y_\eta)^2 \, d\xi d\eta \qquad (2.24)$$

y su discretización usando el procedimiento descrito es

$$f_{ort}(P,Q,R,S) = ((P-S)^t(Q-P))^2 + ((Q-P)^t(R-Q))^2 + \qquad (2.25)$$
$$((S-R)^t(R-Q))^2 + ((S-R)^t(P-S))^2$$

FUNCIONAL DISCRETO DE AREA-ORTOGONALIDAD

El funcional de Area-Ortogonalidad (1.29) presentado por Knupp [41] visto en el capítulo es

$$I_{ao}[\bar{d}] = \int_0^1\int_0^1 (x_\xi^2 + y_\xi^2) \cdot (x_\eta^2 + y_\eta^2) \, d\xi d\eta \ , \qquad (2.26)$$

y su discretización queda como:

$$f(\bar{d}(\xi,\eta)) = (x_\xi^2 + y_\xi^2) \cdot (x_\eta^2 + y_\eta^2) \ ,$$

$$f(r(0,0)) = \|Q-P\|^2 * \|S-P\|^2 \ , \quad f(r(1,0)) = \|Q-P\|^2 * \|R-Q\|^2 \ ,$$

$$f(r(0,1)) = \|R-S\|^2 * \|S-P\|^2 \ , \quad f(r(1,1)) = \|R-S\|^2 * \|R-Q\|^2 \ ,$$

$$f_{ao}(P,Q,R,S) = 1/4 \ [\ f_H(P,Q,R,S) * f_V(P,Q,R,S)\ ] \qquad (2.27)$$

2.2. FUNCIONALES DISCRETOS DIRECTOS

Se desea formar un funcional discreto tomado directamente sobre los nodos de una malla dada sobre una región, la que se quiere que posea determinada propiedad geométrica, entonces esta propiedad se puede introducir tomando la suma sobre todos los nodos, lo que resulta un enfoque distinto al visto en el acápite anterior, donde los funcionales discretos se obtienen a partir de discretizar sus análogos continuos. Sin embargo, según se comprobará más adelante para el caso de los funcionales de longitud y de área de Barrera y Pérez, el funcional discreto obtenido por ambos enfoques es el mismo.

2.2.1. FUNCIONAL DISCRETO DIRECTO DE LONGITUD

Castillo [13] plantea que el control más simple de la longitud se logra tratando, en un sentido variacional, de hacer los segmentos de la red iguales. Para ello, debe minimizarse la suma de los cuadrados de las longitudes entre los puntos de la red. Supongamos que s_{ij} es la longitud entre el (i,j)-ésimo punto y cualquier punto vecino, entonces hay que hallar

$$\text{Min } f_{\ell} = \sum s_{ij}^2 \qquad (2.28)$$

con la restricción

$$\sum s_{ij} = \text{cte}$$

Esta restricción sobre el espaciamiento de las líneas de la red se satisface automáticamente, ya que la suma de todos los segmentos es una suma telescópica que depende sólo de los valores en la frontera.

En 1986, Castillo [13] introduce un funcional para generar redes en el que uno de los términos se usa para controlar el tamaño de los segmentos "horizontales" y otro para los segmentos

"verticales" de la red. Esto se mostrará a continuación, tomando como referencia a Barrera-Castillo [3].
Si se considera una red de dimensión mxn, entonces su j-ésimo segmento horizontal está dado por el conjunto

$$\{P_{1,j}, P_{2,j}, \ldots, P_{i,j}, \ldots, P_{m,j}\} \quad .$$

Lo que se quiere es controlar la longitud de cada uno de los (m-1) segmentos horizontales, para lo cual se trata de que cada uno de ellos esté formado por segmentos elementales del mismo tamaño (ver fig.2.6), donde

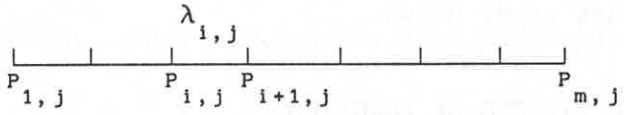


fig. 2.6. Puntos sobre una línea horizontal

$$\lambda_{i,j} = \| P_{i+1,j} - P_{i,j} \|_2$$

Por lo tanto, para lograr que

$$\lambda_{1,j} \approx \lambda_{2,j} \approx \ldots\ldots \approx \lambda_{m-1,j}$$

se construye el funcional

$$\sum_{i=1}^{m-1} \lambda_{i,j}^2 = \sum_{i=1}^{m-1} \| P_{i+1,j} - P_{i,j} \|_2^2$$

y como se quiere lograr la misma propiedad con todos los segmentos de la red, entonces se considera

$$f_H(z)=\sum_{j=2}^{n-1}\sum_{i=1}^{m-1}\lambda_{i,j}^2=\sum_{j=2}^{n-1}\sum_{i=1}^{m-1}\|P_{i+1,j}-P_{i,j}\|_2^2$$

Este funcional depende únicamente de los puntos interiores de la red, pues los puntos de la frontera horizontal quedan fijos, como ya se había mencionado.

Es un resultado conocido que el mínimo de f_H, sujeto a la restricción de que la suma de los segmentos horizontales de cada línea sea la longitud de la misma de un extremo a otro de la frontera, ocurre cuando todos los segmentos de esta son iguales entre sí.

Para construir el funcional de los segmentos verticales, se considerará una forma análoga a lo anterior en una red de dimensión mxn el i-ésimo segmento vertical vendrá dado por el conjunto

$$\{P_{i,1},P_{i,2},\ldots,P_{i,j},\ldots,P_{i,n}\} \quad ,$$

y se desea controlar la longitud de cada uno de los n-1 segmentos verticales, para lo cual se trata de que cada uno de ellos esté formado por segmentos elementales del mismo tamaño (fig.2.7), es decir

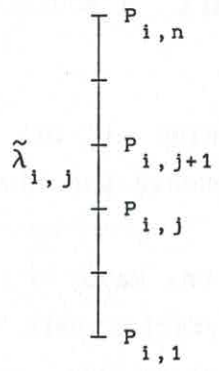


fig. 2.7. Puntos sobre una línea vertical

$$\tilde{\lambda}_{i,1}\approx\tilde{\lambda}_{i,2}\approx\ldots..\approx\tilde{\lambda}_{i,n-1} \quad ,$$

Después se construye la sumatoria

$$\sum_{j=1}^{n-1} \tilde{\lambda}^2_{i,j} = \sum_{j=1}^{n-1} \| P_{i,j+1} - P_{i,j} \|_2^2$$

Como se quiere trabajar con todos los segmentos verticales de la malla, se considera entonces

$$f_V(z) = \sum_{i=2}^{m-1} \sum_{j=1}^{n-1} \tilde{\lambda}^2_{i,j} = \sum_{i=2}^{m-1} \sum_{j=1}^{n-1} \| P_{i,j+1} - P_{i,j} \|_2^2$$

A partir de lo anterior, se tiene que f_ℓ se puede escribir como

$$f_\ell = f_H + f_V$$

y éste es el funcional que usa Castillo [13] para generar redes suaves.

Después de experimentar con f_ℓ, se observó que una forma de poder controlar mejor las propiedades de la red óptima es asignando un peso diferente a f_H y f_V por lo que se utilizó el siguiente funcional:

$$f_{\ell,\tau} = \tau f_H + f_V \quad ; \quad \text{para} \quad \tau \geq 0 \quad ,$$

que coincide con la discretización (2.18) del funcional variacional de longitud (1.10) propuesto por Steinberg y Roache [60].

Este parámetro τ da como resultado una mayor o menor tensión de las líneas, logrando el siguiente efecto: para τ 0, las líneas horizontales quedan "sueltas", mientras que las verticales se "tensan", y por el contrario, para $\tau \infty$, las verticales se "sueltan" y las horizontales se "tensan". En los ejemplos de este funcional que se dan en el anexo I, se ven claramente los efectos

geométricos de los distintos valores de τ.

Como el enfoque sobre generación de redes se va a hacer a través de la optimización de funcionales, se requiere de los gradientes y Hessianas de los mismos para poder demostrar las condiciones que garanticen la existencia de mínimos locales con las propiedades deseadas. Por tal razón, a continuación se presentan algunos lemas y teoremas que permitirán establecer estos resultados.

Por ser $f_{\ell,\tau}$ la suma de dos funcionales, es más sencillo ver los gradientes y Hessianas de f_H y f_V por separado y a partir de ellos obtener los correspondientes para $f_{\ell,\tau}$.

Lema 2.1.- Sean $A^t=(x_A,y_A)$; $C^t=(x_C,y_C)$; $P^t=(x_P,y_P)$ puntos dados de $\mathbb{R}^2$ (ver fig. 2.8) y sea $\bar{f}_H$ la función

$$\bar{f}_H = \|P-A\|_2^2 + \|P-C\|_2^2 \quad ,$$

entonces,

$$\frac{\partial \bar{f}_H}{\partial P} = 2[(P-A) + (P-C)] \quad ,$$

$$\frac{\partial^2 \bar{f}_H}{\partial P^2} = 4I_2 \quad , \qquad \frac{\partial^2 \bar{f}_H}{\partial A \partial P} = \frac{\partial^2 \bar{f}_H}{\partial C \partial P} = -2I_2 \quad ,$$

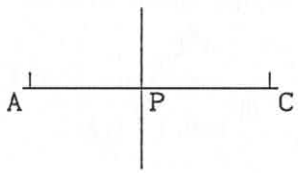


fig. 2.8. Puntos para formar $\bar{f}_H$.

Usando el lema anterior, se extiende su resultado para el

funcional f_H en el siguiente teorema:

Teorema 2.1.-

Sea

$$f_H(z)=\sum_{j=2}^{n-1}\sum_{i=1}^{m-1}\lambda_{i,j}^2=\sum_{j=2}^{n-1}\sum_{i=1}^{m-1}\|P_{i+1,j}-P_{i,j}\|_2^2$$

su gradiente y Hessiana tienen la estructura (ver fig.2.9)

a) $\dfrac{\partial f_H}{\partial P_{i,j}}=2[(P_{i,j}-P_{i-1,j})+(P_{i,j}-P_{i+1,j})]$

para $2\leq i\leq m-1$ y $2\leq j\leq n-1$

b) $\nabla^2 f_H = H = T_{m-2}\otimes I_{n-2}\otimes I_2 = T_{m-2}\otimes I_{2(n-2)}$

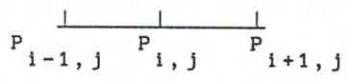


fig. 2.9. Puntos adyacentes a $P_{i,j}$ en la línea horizontal

Demostración.-

a) Se obtiene directamente a partir del lema anterior

b) Por el lema 2.1 se tiene que

$$\frac{\partial^2 f_H}{\partial P_{i,j}^2}=4I_2 \quad y \quad \frac{\partial^2 f_H}{\partial P_{i-1,j}\partial P_{i,j}}=\frac{\partial^2 f_H}{\partial P_{i+1,j}\partial P_{i,j}}=-2I_2$$

La Hessiana en términos de los $P_{i,j}$ se puede escribir como

$$H = \begin{bmatrix} \dfrac{\partial^2 f_H}{\partial P_{2,2}^2} & \dfrac{\partial^2 f_H}{\partial P_{2,3}\partial P_{2,2}} & \cdots & \dfrac{\partial^2 f_H}{\partial P_{m-1,n-1}\partial P_{2,2}} \\ \vdots & & & \vdots \\ \dfrac{\partial^2 f_H}{\partial P_{2,2}\partial P_{m-1,n-1}} & \dfrac{\partial^2 f_H}{\partial P_{2,3}\partial P_{m-1,n-1}} & \cdots & \dfrac{\partial^2 f_H}{\partial P_{m-1,n-1}^2} \end{bmatrix}$$

Nótese que para $P_{i,j}$ dado, sólo tienen afectación en la derivada parcial respecto a él los puntos adyacentes en la horizontal, esto es, $P_{i-1,j}$ y $P_{i+1,j}$, pero debido al convenio sobre el ordenamiento del vector de variables $\mathbf{z}^t$, no están ni a la izquierda ni a la derecha de $P_{i,j}$, sino (n-2) puntos antes y (n-2) después, respectivamente. Entonces resulta que la Hessiana es una matriz "sparse" (con muchos ceros), y es posible escribirla por bloques de orden (n-2)x(n-2) y que en total serán (m-2)x(m-2). Sean estos bloques $H_{l,k}$, de tal forma que son las derivadas parciales respecto al grupo **l** de puntos verticales primero y luego respecto al grupo **k**, esto es, definiendo

1^{er} grupo de puntos $\quad (P_{2,2}, P_{2,3}, P_{2,4}, \ldots, P_{2,n-1})$

k-ésimo grupo de puntos $\quad (P_{k,2}, P_{k,3}, P_{k,4}, \ldots, P_{k,n-1})$

cada bloque va a ser la siguiente matriz

$$H_{l,k}=\begin{bmatrix} \frac{\partial^2 f_H}{\partial P_{k,2}\partial P_{l,2}} & \frac{\partial^2 f_H}{\partial P_{k,3}\partial P_{l,2}} & \cdots & \frac{\partial^2 f_H}{\partial P_{k,n-1}\partial P_{l,2}} \\ \vdots & & & \vdots \\ \frac{\partial^2 f_H}{\partial P_{k,2}\partial P_{l,n-1}} & \frac{\partial^2 f_H}{\partial P_{k,3}\partial P_{l,n-1}} & \cdots & \frac{\partial^2 f_H}{\partial P_{k,n-1}\partial P_{l,n-1}} \end{bmatrix}$$

Si r representa el conjunto de puntos $(P_{r+1,2},\ldots,P_{r+1,n-1})$, entonces sólo tienen derivadas parciales distintas de cero con respecto a este conjunto de puntos, los bloques $H_{r-1,r}$, $H_{r,r}$ y $H_{r,r+1}$ y como H es simétrica, entonces $H^t_{r,r+1} = H_{r+1,r}$. Además, cada uno de estos bloques a su vez es distinto de cero sólo en la diagonal, pues es donde van las derivadas parciales del punto dado con su colindante en la horizontal. Luego se tiene que:

$$H_{r,r}=2I_{n-2}\otimes I_2$$

$$H^t_{r,r+1}=H_{r+1,r}=-I_{n-2}\otimes I_2$$

y el resto de los bloques son de la forma $0I_{n-2}\otimes I_2$.
Entonces,

$$\frac{1}{2}H = \begin{bmatrix} 2I_{n-2} & -I_{n-2} & 0I_{n-2} & \cdots & \\ -I_{n-2} & 2I_{n-2} & -I_{n-2} & \cdots & \\ 0I_{n-2} & -I_{n-2} & 2I_{n-2} & \cdots & \\ \vdots & & & & \vdots \\ & & & & 0I_{n-2} \\ 0I_{n-2} & \cdots & & -I_{n-2} & 2I_{n-2} \end{bmatrix} \otimes I_2$$

$$\Leftrightarrow \qquad H = T_{m-2} \otimes I_{n-2} \otimes I_2 = T_{m-2} \otimes I_{2(n-2)} \qquad \blacksquare$$

De igual forma es posible establecer los resultados para f_v.

Lema 2.2.- Sean $B=(x_B, y_B)$, $D=(x_D, y_D)$ y $P=(x_P, y_P)$ puntos dados en $\mathbb{R}^2$ (ver fig. 2.10), y sea $\bar{f}_v$ la funcional

$$\bar{f}_v = \|P-B\|_2^2 + \|P-D\|_2^2$$

entonces

a) $$\frac{\partial \bar{f}_v}{\partial P} = 2[(P-B)+(P-D)]$$

b) $$\frac{\partial^2 \bar{f}_v}{\partial P^2} = 4I_2 \quad ; \quad \frac{\partial^2 \bar{f}_v}{\partial B \partial P} = \frac{\partial^2 \bar{f}_v}{\partial D \partial P} = -2I_2$$

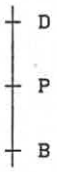


fig. 2.10. Puntos sobre los que se toma $\bar{f}_V$.

Teorema 2.2.-

Sea

$$f_V(z)=\sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\tilde{\lambda}^2_{i,j}=\sum_{i=2}^{m-1}\sum_{j=1}^{n-1}\|P_{i,j+1}-P_{i,j}\|_2^2$$

entonces, su gradiente y Hessiana tienen la siguiente estructura:

a) $\dfrac{\partial f_V}{\partial P_{i,j}} = 2[(P_{i,j}-P_{i,j-1})+(P_{i,j}-P_{i,j+1})]$

para $2\leq i\leq m-1$ y $2\leq j\leq n-1$

b) Si V es la matriz Hessiana de f_V , se tiene

$$-\nabla^2 f_V = -V = I_{m-2}\otimes T_{n-2}\otimes I_2$$

Teniendo ya los resultados para f_H y f_V , es fácil obtener los correspondientes para $f_{\ell,\tau}$.

Teorema 2.3.- Si $g_{\ell,\tau}$ denota el gradiente de $f_{\ell,\tau}$ y $H_{\ell,\tau}$ su matriz Hessiana, entonces:

a) $g_{\ell,\tau}=(g_1^t, g_2^t, \ldots, g_k^t, \ldots, g^t_{(m-2)(n-2)})$

con k dado por (2.1) para $2\leq i\leq m-1$; $2\leq j\leq n-1$ y

$$g_k = \frac{\partial f_{\ell,\tau}}{\partial P_{i,j}} = 2[\tau(P_{i,j}-P_{i-1,j})+(P_{i,j}-P_{i,j-1}) + \tau(P_{i,j}-P_{i+1,j})+(P_{i,j}-P_{i,j+1})]$$

b) Si $H_{\ell,\tau}$ denota la matriz Hessiana de $f_{\ell,\tau}$ entonces

$$H_{\ell,\tau} = L_{(m-2)(n-2)} \otimes I_2$$

donde $L_{(m-2)(n-2)}$ es tridiagonal por bloques dada por :

$$\begin{bmatrix} T_{n-2}+2\tau I_{n-2} & -\tau I_{n-2} & 0\ I_{n-2} & \cdots & & \\ -\tau I_{n-2} & T_{n-2}+2\tau I_{n-2} & -\tau I_{n-2} & \cdots & & \\ 0\ I_{n-2} & \cdot & & & & \\ \cdot & & & & \cdot & \\ \cdot & & & & \cdot & \\ \cdot & & & & -\tau I_{n-2} & \\ 0\ I_{n-2} & \cdots\cdots\cdots\cdots & & & -\tau I_{n-2} & T_{n-2}+2\tau I_{n-2} \end{bmatrix}$$

Ya con las expresiones del gradiente y la matriz Hessiana de $f_{\ell,\tau}$ se puede probar la existencia de un mínimo.

Corolario.-

a) $H_{\ell,\tau}$ es definida positiva.

b) $f_{\ell,\tau}$ tiene un mínimo único y en el mínimo los puntos de la red satisfacen

$$P_{i,j} = \frac{\tau P_{i-1,j} + P_{i,j-1} + \tau P_{i+1,j} + P_{i,j+1}}{2\tau+2}$$

Demostración:

Se sabe que $L_{(m-2)(n-2)}$ es definida positiva (ver Conte [19]) y

por tanto, se puede escribir como

$$L_{(m-2)(n-2)} = U^t U$$

donde U^t es no singular, entonces:

$$\begin{aligned} H_{\ell,\tau} &= U^t U \otimes I_2 \\ &= (U^t \otimes I_2)(U \otimes I_2) \\ &= (U \otimes I_2)^t (U \otimes I_2) \quad . \end{aligned}$$

Esto se debe a las propiedades del producto Kronecker [19], ya que $(U \otimes I_2)$ es no singular, entonces se concluye que $H_{\ell,\tau}$ es definida positiva.

Para la parte b) se tiene:

$f_{\ell,\tau}$ es una función cuadrática con matriz definida positiva, por tanto, tiene un mínimo único. Además

$$\begin{aligned} \frac{\partial f_{\ell,\tau}}{\partial P_{i,j}} = 2[&\tau(P_{i,j} - P_{i-1,j}) + (P_{i,j} - P_{i,j-1}) \\ &+ \tau(P_{i,j} - P_{i+1,j}) + (P_{i,j} - P_{i,j+1})] \end{aligned}$$

y se hace cero cuando

$$P_{i,j} = \frac{\tau P_{i-1,j} + P_{i,j-1} + \tau P_{i+1,j} + P_{i,j+1}}{2\tau+2} \quad ,$$

y como $H_{\ell,\tau}$ es definida positiva, resulta que $P_{i,j}$ es el mínimo. ∎

2.2.2. FUNCIONAL DISCRETO DIRECTO DE AREA

Las propiedades del área de los polígonos orientados juegan un papel importante en los resultados de los funcionales que se proponen para controlar el área de las celdas $c_{i,j}$ de la red, por lo cual se presentan a continuación.

Definición.- Dado el triángulo PQR, se define su área $\alpha(PQR)$ en la

forma usual como

$$\alpha(PQR) = \frac{1}{2}\det(Q-P, R-P)$$

$$= \frac{1}{2}\begin{vmatrix} x_Q - x_P & x_R - x_P \\ y_Q - y_P & y_R - y_P \end{vmatrix}$$

Es fácil demostrar que $\alpha(PQR)$ es positiva si el triángulo se orienta en el sentido contrario a las manecillas del reloj, y negativa si es orientado en el sentido de las manecillas del reloj (ver fig. 2.11)

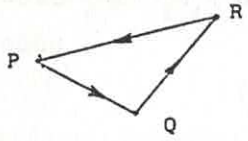


fig. 2.11 Triángulo orientado positivamente

Propiedad.- Dado un n-polígono orientado $P_1P_2\ldots P_n$ (ver fig. 2.12) su área se puede obtener como

$$\alpha(P_1P_2\ldots P_n) = \sum_{i=2}^{n-1} \alpha(P_1, P_i, P_{i+1})$$

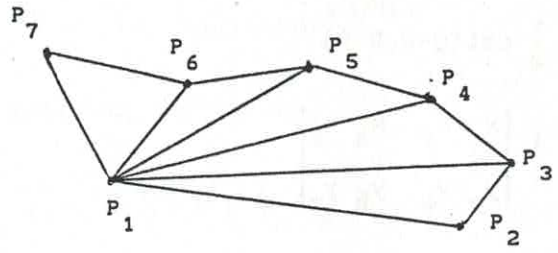


fig. 2.11. Area de la región poligonal tomada como suma de los triángulos que la forman

Propiedad.- Dado un n-polígono orientado $P_1P_2\ldots P_n$ se tiene:

a) $\alpha(P_1P_2\ldots P_n)=\sum_{i=1}^{n-2}\det(P_{i+1}-P_1,P_{i+2}-P_1)$

donde $P_{n+1}=P_1$

b) $\alpha(P_1P_2\ldots P_n)=\alpha(P_k,P_{k+1},\ldots,P_n,P_1,P_2,\ldots,P_{k-1})$

Propiedad.- Dados dos n-polígonos orientados $P_1P_2\ldots P_n$ y $P_nQ_1\ldots Q_mP_1$, que tienen el lado común P_1P_n pero con orientación opuesta, se tiene entonces que la suma de las áreas de los polígonos es igual al área del polígono formado con la unión de los dos (ver fig. 2.12), esto es:

$$\alpha(P_1P_2\ldots P_n)+\alpha(P_nQ_1\ldots Q_mP_1)=\alpha(P_1P_2\ldots P_nQ_1\ldots Q_m)$$

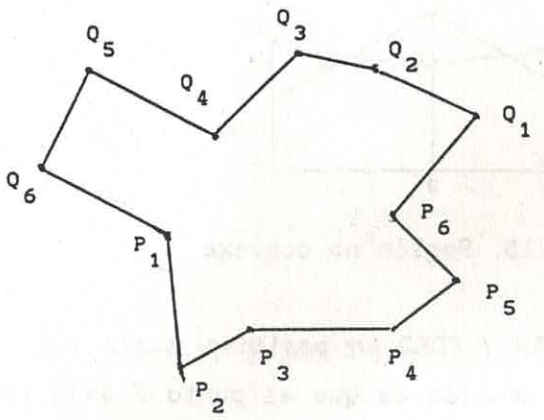


fig. 2.12. Area de una región formada por dos polígonos que tienen un lado común

Propiedad.- Dada una región rectangular Ω y una red sobre ella, se tiene que

$$\sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\alpha_{i,j} = \alpha(\Omega)$$

donde

$$\alpha_{i,j} = \alpha(P_{i,j}, P_{i+1,j}, P_{i+1,j+1}, P_{i,j+1})$$

Hasta el momento se ha hablado de que una de las propiedades que se necesita tener en una malla es que esta sea "convexa". En términos prácticos, una red es convexa si todas sus celdas tienen áreas positivas. Sin embargo, esta definición puede tener problemas, por ejemplo, en la figura siguiente

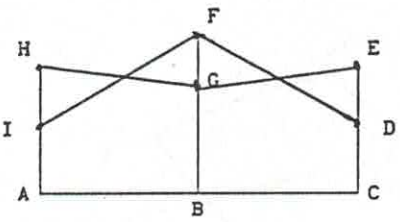


fig. 2.15. Región no convexa

el área de las celdas IFGH Y FDEG es positiva y sin embargo, la red no es convexa, en el sentido de que el punto F sale fuera de la región.

Castillo [13] propone usar para controlar las áreas de las celdas de la malla un funcional que sea la suma de todas estas áreas al cuadrado. Sin embargo, este tiene dificultades en su realización práctica (ver Barrera y otros [5]), ya que para algunas regiones sucede lo de la región en la fig 2.15, donde a pesar de que sus áreas son iguales, la línea sale fuera de la región. Para evitar las deficiencias del funcional de Castillo, Barrera y Pérez en 1988 propusieron una variante que se describe a continuación.

Considérense en la celda $c_{i,j}$ los triángulos (ver fig. 2.16)

$$\Delta^{(1)}_{i,j} = \Delta(P_{i,j}, P_{i+1,j}, P_{i,j+1})$$

$$\Delta^{(2)}_{i,j} = \Delta(P_{i+1,j}, P_{i+1,j+1}, P_{i,j})$$

$$\Delta^{(3)}_{i,j} = \Delta(P_{i+1,j+1}, P_{i,j+1}, P_{i+1,j})$$

$$\Delta^{(4)}_{i,j} = \Delta(P_{i,j+1}, P_{i,j}, P_{i+1,j+1})$$

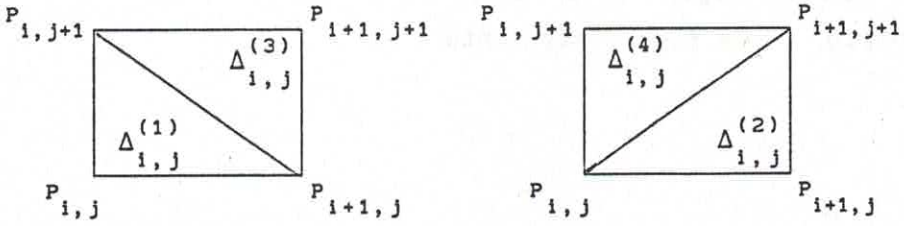


fig. 2.16. Cuatro triángulos en una celda

y sea

$$\alpha_{i,j}^{(k)} = \text{área}(\Delta_{i,j}^{(k)})$$

$$f_t^t = (f_1, f_2, \ldots, f_l, \ldots f_N)$$

donde

$$f_l = \alpha_{i,j}^{(k)} \quad \text{con } 1 \le i \le m-1,\ 1 \le j \le n-1 \text{ y } k=1,2,3,4$$

luego

$$l = \Phi(i,j,k) = 4[(n-1)(i-1) + (j-1)] + k \qquad (2.32)$$

y $l = 1,2,\ldots,N$, donde N es el número total de triángulos: $N = 4(n-1)(m-1)$.
Entonces, el funcional que se propone para controlar la no convexidad de algunas celdas es

$$f_a = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \sum_{k=1}^{4} (\alpha_{i,j}^{(k)})^2 \qquad (2.33)$$

que coincide con la discretización que se hizo del funcional variacional (1.11) y cuya expresión se puede ver en (2.21).
Las propiedades de f_a se obtendrán a partir de las expresiones de su gradiente y matriz Hessiana.

Lema 2.4.- Sea $\alpha(PQR)$ el área del triángulo orientado PQR (ver fig. 2.11), entonces:

a) $$\alpha(PQR) = \frac{1}{2} \det(Q-P, R-P) = \frac{1}{2} \begin{vmatrix} x_Q - x_P & x_R - x_P \\ y_Q - y_P & y_R - y_P \end{vmatrix}$$

b) $2\frac{\partial\alpha}{\partial P} = J_2(Q-R)$; $2\frac{\partial\alpha}{\partial R} = J_2(P-Q)$

$$2\frac{\partial\alpha}{\partial Q} = J_2(R-P)$$

c) $$\frac{\partial^2\alpha}{\partial P^2} = \frac{\partial^2\alpha}{\partial Q^2} = \frac{\partial^2\alpha}{\partial R^2} = 0$$

$2\frac{\partial^2\alpha}{\partial Q\partial P} = J_2$; $2\frac{\partial^2\alpha}{\partial R\partial P} = -J_2$

$2\frac{\partial^2\alpha}{\partial P\partial Q} = -J_2$; $2\frac{\partial^2\alpha}{\partial R\partial Q} = J_2$

$2\frac{\partial^2\alpha}{\partial P\partial R} = J_2$; $2\frac{\partial^2\alpha}{\partial Q\partial R} = -J_2$

$$\nabla^2\alpha = \begin{pmatrix} 0 & J_2 & -J_2 \\ -J_2 & 0 & J_2 \\ J_2 & -J_2 & 0 \end{pmatrix}$$

Teorema 2.8.- Sea g_a el gradiente de f_a y H_a su Hessiana, entonces:

a) $g_a = 2B_t^t f_t$

donde $B_t^t = [g_1 | g_2 | \ldots g_1 | \ldots | g_N]$

y g_1= gradiente $\alpha_{i,j}^{(k)}$ y la función que relaciona los índices i,j,k con el 1 es (2.32), es decir, g_1 es el gradiente de f_1 y entonces B_t^t es una matriz de 2(m-2)(n-2)x4(m-1)(n-1), cuyas columnas son los gradientes de las áreas de los triángulos que forman las celdas de la red.

b) $H_a = B_t^t B_t + \sum_{l=1}^{N} f_l H_l$

donde H_l es la Hessiana de $f_l = \alpha_{i,j}^{(k)}$

Para establecer y demostrar el siguiente teorema, considérese el punto interior P de la red y sus puntos vecinos A, B, C, D, E, F, G, H. En la fig. 2.18 se muestran todos los cuadrángulos que tienen a P como uno de sus vértices y en la fig 2.19a) y b) se muestran por separado los triángulos de acuerdo a la diagonal de cada cuadrilátero.

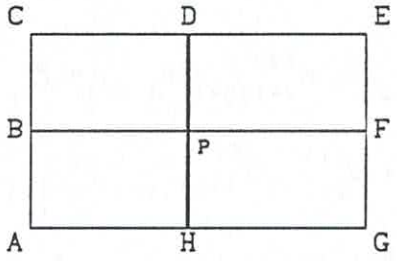
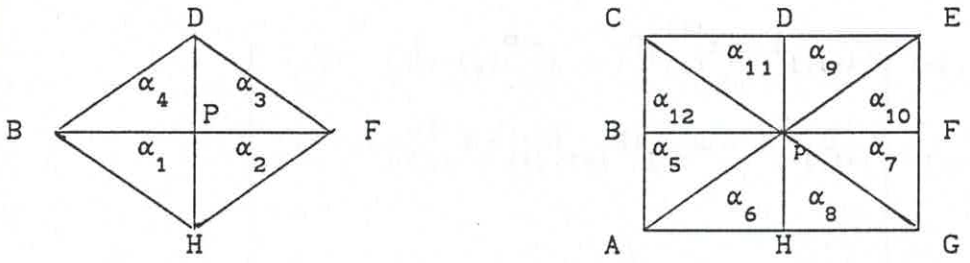


fig. 2.18 Celdas que tienen a P como vértice



a) b)

fig. 2.19. Triángulos que tienen al punto P como vértice

Usando el lema 2.4, se puede probar que

$$2\frac{\partial f_a}{\partial P} = J_2[\alpha_1(B-H)+\alpha_2(H-F)+\alpha_3(F-D)+\alpha_4(D-B)$$
$$\alpha_5(B-A)+\alpha_6(A-H)+\alpha_7(G-F)+\alpha_8(H-G)$$
$$\alpha_9(E-D)+\alpha_{10}(F-E)+\alpha_{11}(D-C)+\alpha_{12}(C-B)$$

donde α_i es el i-ésimo triángulo que tiene a P como vértice, ya que únicamente los triángulos desde i=1 hasta 12 cumplen esta condición.

Teorema 2.9.- La primera derivada respecto a un punto P, del funcional de área f_a es

$$2\frac{\partial f_a}{\partial P} = J_2[\alpha^{(2)}_{i-1,j-1}(P_{i-1,j-1} - P_{i,j-1}) + \alpha^{(3)}_{i-1,j-1}(P_{i-1,j} - P_{i,j-1})$$
$$+ \alpha^{(4)}_{i-1,j-1}(P_{i-1,j} - P_{i-1,j-1}) + \alpha^{(1)}_{i-1,j}(P_{i-1,j+1} - P_{i-1,j})$$
$$+ \alpha^{(2)}_{i-1,j}(P_{i,j+1} - P_{i-1,j}) + \alpha^{(3)}_{i-1,j}(P_{i,j+1} - P_{i-1,j+1})$$
$$+ \alpha^{(1)}_{i,j-1}(P_{i,j-1} - P_{i+1,j-1}) + \alpha^{(3)}_{ij-1}(P_{i+1,j-1} - P_{i+1,j})$$
$$+ \alpha^{(4)}_{i,j-1}(P_{i,j-1} - P_{i+1,j}) + \alpha^{(1)}_{i,j}(P_{i+1,j} - P_{i,j+1})$$
$$+ \alpha^{(2)}_{i,j}(P_{i+1,j} - P_{i+1,j+1}) + \alpha^{(4)}_{i,j}(P_{i+1,j+1} - P_{i,j+1})$$

El siguiente teorema da la forma explícita del gradiente de f_a.

Teorema 2.10.- El gradiente del funcional f_a tiene la estructura

$$g_a = (I_{(m-2)(n-2)} \otimes J_2) \bar{B}_t f_t$$

donde $\bar{B}_t$ es una matriz bidiagonal por bloques y cada bloque es también bidiagonal por bloques, donde los últimos son vectores de

dimensión 1x4, como se muestra

$$\bar{B}_t = \begin{bmatrix} D_{1,1} & D_{1,2} & 0 & \cdots & 0 \\ 0 & D_{2,2} & D_{2,3} & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & & D_{m-2,m-2} & D_{m-2,m-1} \end{bmatrix}$$

con

$$D_{r,r} = \begin{bmatrix} a_1^{(r)} & b_2^{(r)} & 0 & \cdots & 0 \\ 0 & a_2^{(r)} & b_2^{(r)} & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & & a_{n-2}^{(r)} & b_{n-2}^{(r)} \end{bmatrix}$$

$$D_{r,r+1} = \begin{bmatrix} c_1^{(r)} & d_1^{(r)} & 0 & \cdots & 0 \\ 0 & c_2^{(r)} & d_2^{(r)} & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & & c_{n-2}^{(r)} & d_{n-2}^{(r)} \end{bmatrix}$$

y

$$a_q^{(r)} = [(P_{r,q} - P_{r,q-1}), (P_{r+1,q} - P_{r,q+1}), (P_{r+1,q} - P_{r,q}),\ 0\ ]$$

$$b_q^{(r)} = [0\ , (P_{r,q+2} - P_{r+1,q+2}), (P_{r,q+1} - P_{r+1,q+2}), (P_{r,q+1} - P_{r,q+2})]$$

$$c_q^{(r)} = [(P_{r+2,q+1} - P_{r+1,q}), (P_{r+2,q+1} - P_{r+2,q}),\ 0, (P_{r+2,q} - P_{r+1,q})]$$

$$d_q^{(r)} = [(P_{r+1,q+2} - P_{r+2,q+2}),\ 0\ , (P_{r+2,q+2} - P_{r+1,q+1}),$$

$$(P_{r+1,q+2} - P_{r+2,q+1})]$$

para $1 \le r \le m-2$ y $1 \le q \le n-2$

Demostración.-

Se tiene que

$$f_t^t = (f_1, \ldots, f_l, \ldots, f_{4(m-1)(n-1)})$$

con $1 \le l \le 4(m-1)(n-1)$

y $f_l = \alpha_{u,v}^{(k)}$ el área de cada triángulo de las celdas y l dada por la relación (2.32), entonces

$$g_l = \frac{\partial f_l}{\partial P_{i,j}} \quad \text{para} \quad 2 \le i \le m-1 \quad \text{y} \quad 2 \le j \le n-1$$

y $$g_a = 2B_t^t f_t = \frac{\partial f_a}{\partial P_{i,j}}$$

donde

$$B_t^t = [g_1 \mid g_2 \mid \ldots g_l \mid \ldots \mid g_{4(m-1)(n-1)}]$$

y cada g_l es de orden $2(m-2)(n-2) \times 1$. La matriz B_t^t tiene una estructura de bloques. Sean $D_{r,s}$ sus bloques que representan las derivadas del conjunto **s** de áreas verticales con respecto al grupo **r** de puntos verticales $(P_{r+1,2}, \ldots, P_{r+1,n-2})$ para $1 \le r \le m-2$

(ver fig. 2.20)

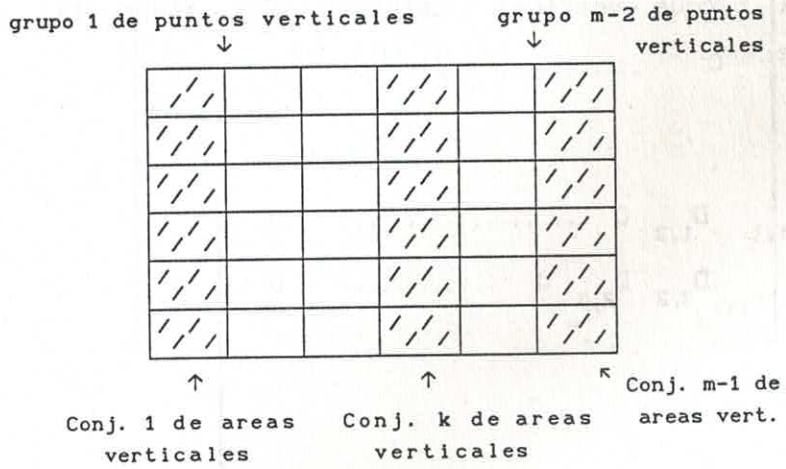


fig. 2.20. Bloques de celdas

Ya que $g_1=\frac{\partial f_1}{\partial P_{i,j}}$ es distinto de cero sólo para 3 de los cuatro triángulos de cada una de las cuatro celdas que lo tienen como vértice, y que son las que se muestran en la fig. 2.21.

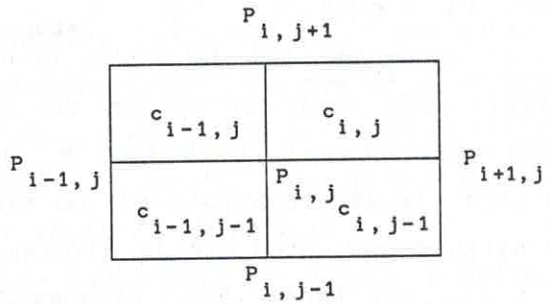


fig. 2.21. Celdas que tienen a $P_{i,j}$ como vértice.

resulta entonces que para un conjunto de puntos $(P_{r+1,2},\ldots,P_{r+1,n-2})$, sólo tienen derivadas diferentes de cero respecto a ellos, el bloque que representa las derivadas del conjunto de áreas verticales a su izquierda $D_{r,r}$ y el de su

derecha $D_{r,r+1}$, para $1 \leq r \leq m-2$, siendo estos bloques de tamaño $(n-2)x4(n-1)$, puesto que son n-2 puntos verticales y 4(n-1) triángulos en cada bloque vertical. Entonces B_t^t tiene la estructura siguiente:

$$B_t^t = \begin{bmatrix} D_{1,1} & D_{1,2} & 0 & \cdots & 0 \\ 0 & D_{2,2} & D_{2,3} & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \cdots & & D_{m-2,m-2} & D_{m-2,m-1} \end{bmatrix}$$

Además, en cada bloque $D_{r,r}$ y $D_{r,r+1}$, se tiene que sólo hay dos celdas cuyos triángulos tienen derivadas diferentes de cero para el conjunto r de puntos verticales $\{P_{r+1,q+1}\}$ con r fija en algún valor entre 1 y m-2 y **q** tomando valores desde 1 hasta n-2 y que son $c_{r,q}$ y $c_{r,q+1}$ para el bloque $D_{r,r}$ y $c_{r+1,q}$ y $c_{r+1,q+1}$ para el bloque $D_{r,r+1}$. Agrupando las derivadas de los cuatro triángulos de una celda dada como un vector de 1x4, resulta que en las celdas que contienen al punto dado $P_{r+1,q+1}$ habrá una componente igual a cero (la del triángulo que no tiene al punto como vértice) y las otras componentes son la diferencia de los puntos opuestos al $P_{r+1,q+1}$. Por tanto, los bloques $D_{r,r}$ y $D_{r,r+1}$ son a su vez bidiagonales, ya que en cada fila habrá sólo dos celdas con parciales diferentes de cero y que como se vió son las de la diagonal y supradiagonal, lo que se muestra a continuación:

$$D_{r,r} = \begin{bmatrix} a_1^{(r)} & b_2^{(r)} & 0 & \cdots & 0 \\ 0 & a_2^{(r)} & b_2^{(r)} & 0 & \cdots & 0 \\ \cdot & & \cdot & \cdot & & \\ \cdot & & & \cdot & \cdot & \cdot \\ \cdot & & & & \cdot & \cdot \\ \cdot & & & & & \\ 0 & \cdots & & & a_{n-2}^{(r)} & b_{n-2}^{(r)} \end{bmatrix}$$

$$D_{r,r+1} = \begin{bmatrix} c_1^{(r)} & d_1^{(r)} & 0 & \cdots & 0 \\ 0 & c_2^{(r)} & d_2^{(r)} & 0 & \cdots & 0 \\ \cdot & & \cdot & \cdot & & \\ \cdot & & & \cdot & \cdot & \cdot \\ \cdot & & & & \cdot & \cdot \\ \cdot & & & & & \\ 0 & \cdots & & & c_{n-2}^{(r)} & d_{n-2}^{(r)} \end{bmatrix}$$

y para poder escribir la forma de los elementos distintos de cero en los bloques anteriores, se necesita primero relacionar los subíndices i,j de los puntos interiores de la red, que van desde $2 \leq i \leq m-1$ y $2 \leq j \leq n-1$, con el subíndice r del bloque y su q-ésima fila, tales que $1 \leq r \leq m-2$ y $1 \leq q \leq n-2$

Resulta que en la matriz B_t^t el renglón k-ésimo se relaciona con el punto $P_{i,j}$ por la función (2.1) y a su vez k se relaciona con el q renglón del bloque r por:

$$k=(n-2)(r-1)+q$$

con lo que se obtiene

$$(r-1)=(i-2) \Leftrightarrow r=i-1$$

y $q=j-1$

Como para $P_{i,j}$ dado los puntos que tienen afectación en su parcial son $(i-1,j-1)$, $(i,j-1)$, $(i+1,j-1)$, $(i+1,j)$, $(i+1,j+1)$, $(i,j+1)$, $(i-1,j+1)$ y $(i-1,j)$ (ver fig. 2.22)

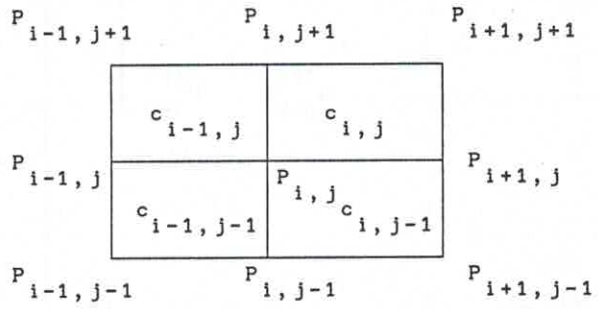


fig. 2.22. Puntos con parciales respecto a $P_{i,j}$ que son diferentes de cero

sustituyendo por sus valores en función de r y q , teniendo en cuenta que $a_q^{(r)}$ es el vector de las derivadas de los cuatro triángulos de la celda $c_{i-1,j-1}$ respecto a $P_{i,j}$ ó equivalentemente $c_{r,q}$, $b_q^{(r)}$ las de la celda $c_{i-1,j}$ ó $c_{r,q+1}$ $c_q^{(r)}$ las de la celda $c_{i,j-1}$ ó $c_{r+1,q}$ y $d_q^{(r)}$ las de la celda $c_{i,j}$ ó $c_{r+1,q+1}$, es fácil comprobar que sus expresiones explícitas son las siguientes:

$$a_q^{(r)}=J_2[0,(P_{r,q}-P_{r+1,q}),(P_{r,q+1}-P_{r+1,q+1}),(P_{r,q+1}-P_{r,q})]$$

$$b_q^{(r)}=J_2[(P_{r,q+2}-P_{r,q+1}),(P_{r+1,q+2}-P_{r,q+1}),(P_{r+1,q+2}-P_{r,q+2}),0]$$

$$c_q^{(r)}=J_2[(P_{r+1,q}-P_{r+2,q}),0,(P_{r+2,q}-P_{r+2,q+1}),(P_{r+1,q}-P_{r+2,q+1})]$$

$$d_q^{(r)}=J_2[(P_{r+2,q+1}-P_{r+1,q+2}),(P_{r+2,q+1}-P_{r+2,q+2}),0,(P_{r+2,q+2}-P_{r+1,q+2})]$$

Como la matriz J_2 aparece de factor común en todos los elementos de los bloques, es posible poner

$$B_t^t = (I_{(m-2)(n-2)} \otimes J_2)\bar{B}_t$$

donde $\bar{B}_t$ es la propia matriz B_t^t pero quitando el factor común J_2 de todos sus elementos, con lo que queda que

$$g_a = (I_{(m-2)(n-2)} \otimes J_2)\bar{B}_t f_t \qquad \blacksquare$$

Ahora se verá la forma explícita del segundo término de la Hessiana de f_a para poder obtener después su expresión completa.

Teorema 2.11.-

$$G=2\sum_{l=1}^{N} f_l H_l = T \otimes J_2$$

donde T es una matriz de (m-2)(n-2)x(m-2)(n-2) tridiagonal por bloques y antisimétrica, dada por

$$T = \begin{bmatrix} T_{1,1} & T_{1,2} & 0 & \cdots & 0 \\ -T_{1,2} & T_{2,2} & T_{2,3} & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & T_{m-3,m-2} \\ & & & -T_{m-3,m-2} & T_{m-2,m-2} \end{bmatrix}$$

con $T_{r,r}$ tridiagonal antisimétrica y $T_{r,r+1}$ diagonal como se muestra:

$$T_{r,r} = \begin{bmatrix} 0 & \xi_1^{(r)} & 0 & \cdots & 0 \\ -\xi_1^{(r)} & 0 & \xi_2^{(r)} & \cdots & 0 \\ 0 & -\xi_2^{(r)} & 0 & \xi_3^{(r)} & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \xi_{n-3} \\ & & & -\xi_{n-3} & 0 \end{bmatrix}$$

y

$$T_{r,r+1} = \text{diag}\ (\tilde{\xi}_q^{(r)})$$

con

$$\xi_q^{(r)} = \alpha_{r,q+1}^{(1)} + \alpha_{r,q+1}^{(3)} - \alpha_{r+1,q+1}^{(1)} - \alpha_{r+1,q+1}^{(4)}$$

$$\tilde{\xi}_q^{(r)} = \alpha_{r+1,q+1}^{(2)} + \alpha_{r+1,q+1}^{(1)} - \alpha_{r+1,q}^{(3)} - \alpha_{r+1,q}^{(4)}$$

para $1 \leq r \leq m-2$ y $1 \leq q \leq n-2$

Demostración.-

Sea

$$G=2\sum_{l=1}^{N} f_l H_l=(\Gamma_{r,s})$$

donde los $\Gamma_{r,s}$ son bloques de 2x2

$$f_l=\alpha_{i,j}^{(k)}$$

con l dado por (2.32), entonces

$$\Gamma_{r,s}=2\sum f_l \frac{\partial^2 f_l}{\partial P_{u,v}\partial P_{i,j}}$$

Se tiene que para $P_{i,j}$ dado , sólo tienen parciales distintas de cero los triángulos que contienen a $P_{i,j}$ y $P_{i+1,j}$ al mismo tiempo (ver fig. 2.23 y los que tienen a $P_{i,j}$ y $P_{i,j+1}$ también a la vez (ver fig. 2.24)

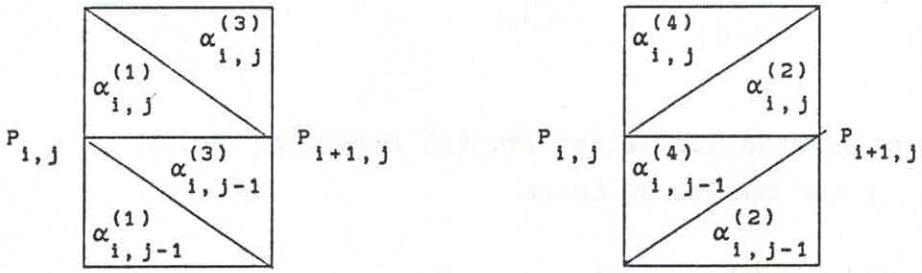


fig. 2.23. Los cuatro triángulos que tienen segundas parciales respecto a $P_{i,j}$ primero y luego respecto a $P_{i+1,j}$

Para $P_{i,j}$ y $P_{i+1,j}$, las parciales que tienen que ver con ambos,

son las de los cuatro triángulos que tienen a ambos puntos como vértices, y que según se ve en la fig. 2.23 son los de áreas $\alpha^{(1)}_{i,j}$, $\alpha^{(2)}_{i,j}$, $\alpha^{(3)}_{i,j-1}$, $\alpha^{(4)}_{i,j-1}$.

Por el lema 2.4 se sabe que

$$\frac{\partial^2 \alpha^{(3)}_{i,j-1}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(4)}_{i,j-1}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(1)}_{i,j}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(2)}_{i,j}}{\partial P^2_{i,j}} = 0$$

$$2 \frac{\partial^2 \alpha^{(4)}_{i,j-1}}{\partial P_{i+1,j} \partial P_{i,j}} = -J_2 \alpha^{(4)}_{i,j-1}$$

$$2 \frac{\partial^2 \alpha^{(3)}_{i,j-1}}{\partial P_{i+1,j} \partial P_{i,j}} = -J_2 \alpha^{(3)}_{i,j-1}$$

$$2 \frac{\partial^2 \alpha^{(2)}_{i,j}}{\partial P_{i+1,j} \partial P_{i,j}} = J_2 \alpha^{(2)}_{i,j}$$

$$2 \frac{\partial^2 \alpha^{(1)}_{i,j}}{\partial P_{i+1,j} \partial P_{i,j}} = J_2 \alpha^{(1)}_{i,j}$$

Por otro lado, considerando los triángulos que contienen a los puntos $P_{i,j}$ y $P_{i,j+1}$ y que son los de áreas

$$\alpha^{(1)}_{i,j}, \quad \alpha^{(4)}_{i,j}, \quad \alpha^{(2)}_{i-1,j}, \quad \alpha^{(3)}_{i-1,j}$$

según se ve en la fig. 2.24, se tiene que :

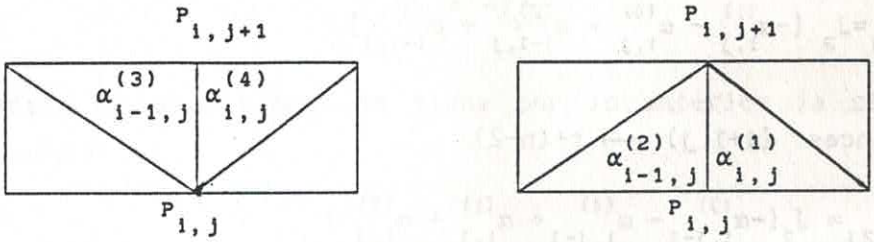


fig. 2.24. Los cuatro triángulos con segundas parciales respecto a $P_{i,j}$ primero y luego respecto a $P_{i,j+1}$

$$\frac{\partial^2 \alpha^{(3)}_{i-1,j}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(2)}_{i-1,j}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(1)}_{i,j}}{\partial P^2_{i,j}} = \frac{\partial^2 \alpha^{(4)}_{i,j}}{\partial P^2_{i,j}} = 0$$

$$2\,\frac{\partial^2 \alpha^{(3)}_{i-1,j}}{\partial P_{i,j+1}\partial P_{i,j}} = J_2 \alpha^{(3)}_{i-1,j}$$

$$2\,\frac{\partial^2 \alpha^{(2)}_{i-1,j}}{\partial P_{i,j+1}\partial P_{i,j}} = J_2 \alpha^{(2)}_{i-1,j}$$

$$2\,\frac{\partial^2 \alpha^{(1)}_{i,j}}{\partial P_{i,j+1}\partial P_{i,j}} = -J_2 \alpha^{(1)}_{i,j}$$

$$2\,\frac{\partial^2 \alpha^{(4)}_{i,j}}{\partial P_{i,j+1}\partial P_{i,j}} = -J_2 \alpha^{(4)}_{i,j}$$

Con la descripción anterior ya se pueden dar explícitamente los elementos $\Gamma_{r,s}$ de la matriz G que son distintos de cero:

Si $(i,j) \longrightarrow r$ entonces $(i,j+1) \longrightarrow r+1$ y

$$\Gamma_{r,r+1} = J_2 \left[-\alpha^{(1)}_{i,j} - \alpha^{(4)}_{i,j} + \alpha^{(2)}_{i-1,j} + \alpha^{(3)}_{i-1,j}\right]$$

Si $(i,j) \longrightarrow r$ entonces $(i+1,j) \longrightarrow r+(n-2)$

$$\Gamma_{r,r+(n-2)} = J_2 \left[-\alpha^{(3)}_{i,j-1} - \alpha^{(4)}_{i,j-1} + \alpha^{(1)}_{i,j} + \alpha^{(2)}_{i,j}\right]$$

Como $P_{k,n-1}$ y $P_{k+1,2}$ son consecutivos pero no adyacentes, se tiene que

$$\Gamma_{n-2,n-1} = \Gamma_{2(n-2),2(n-2)+1} = \ldots = \Gamma_{(n-2),3(n-2)+1} = 0$$

Además, si $(i-1,j) \longrightarrow r$ entonces $(i,j) \longrightarrow r+(n-2)$ y se tiene que:

$$\Gamma_{r+(n-2),r} = \Gamma^t_{r,r+(n-2)}$$

y como $J_2^t = -J_2$ queda que

$$\Gamma_{r+(n-2),r} = -\Gamma_{r,r+(n+2)}$$

También si $(i,j-1) \longrightarrow r$ entonces $(i,j) \longrightarrow r+1$ y se tiene

$$\Gamma_{r+1,r} = \Gamma^t_{r,r+1}$$

$$\Gamma_{r+1,r} = -\Gamma_{r,r+1}$$

Resulta que G es una matriz por bloques, donde cada bloque $T_{l,k}$ está formado por la suma de las segundas parciales de todos los triángulos de las celdas primero respecto del conjunto de puntos verticales l y después con respecto al conjunto de puntos verticales k, multiplicadas por la correspondiente área del triángulo sobre el que se toma la parcial. Asi, para el conjunto r de puntos verticales, sólo son distintos de cero los bloques $T_{r,r-1}$, $T_{r,r}$ y $T_{r,r+1}$ y tales que

$$T_{r,r+1} = -T_{r+1,r}$$

y dentro de cada bloque se tiene por lo anterior la siguiente estructura:

$$T_{r,r} = \begin{bmatrix} 0 & \xi_1^{(r)} & 0 & \cdots & 0 \\ -\xi_1^{(r)} & 0 & \xi_2^{(r)} & \cdots & 0 \\ 0 & -\xi_2^{(r)} & 0 & \xi_3^{(r)} & \\ & \ddots & \ddots & \ddots & \vdots \\ & & \ddots & \ddots & \xi_{n-3} \\ & & & -\xi_{n-3} & 0 \end{bmatrix}$$

$$T_{r,r+1} = \text{diag}\ (\tilde{\xi}_q^{(r)})$$

$$G = \begin{bmatrix} T_{1,1} & T_{1,2} & 0 & \cdots & 0 \\ -T_{1,2} & T_{2,2} & T_{2,3} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ & \ddots & \ddots & \ddots & T_{m-3,m-2} \\ & & & -T_{m-3,m-2} & T_{m-2,m-2} \end{bmatrix} \otimes J_2$$

Siguiendo la misma deducción que en el teorema anterior para obtener la relación entre los subíndices i, j y los r, q de los bloques y a partir de las parciales obtenidas para los puntos colindantes con $P_{i,j}$, se tiene que

$$\xi_q^{(r)} = \alpha^{(2)}_{r,q+1} + \alpha^{(3)}_{r,q+1} - \alpha^{(1)}_{r+1,q+1} - \alpha^{(4)}_{r+1,q+1}$$

$$\tilde{\xi}_q^{(r)} = \alpha^{(1)}_{r+1,q+1} + \alpha^{(2)}_{r+1,q+1} - \alpha^{(3)}_{r+1,q} - \alpha^{(4)}_{r+1,q}$$

para $1 \leq r \leq m-2$ y $1 \leq q \leq n-2$

∎

Corolario.-

$$H_a = 2B_t^t B_t + T \otimes J_2$$

$$H_a = (I_{(m-2)(n-2)} \otimes J_2)\bar{B}_t \bar{B}_t^t (I_{(m-2)(n-2)} \otimes J_2)^t + T \otimes J_2$$

donde $\bar{B}_t$ es la matriz del teorema 2.

Corolario.- Si se tiene una red tal que todas los $\alpha^{(k)}_{i,j}$ son iguales, entonces la red es óptima.

Demostración.-

Teniendo en cuenta la fig. 2.19

$$\frac{\partial f_a}{\partial P} = J_2[\alpha_1(B - H) + \alpha_2(H - F) + \alpha_3(F - D) + \alpha_4(D - B)$$
$$\alpha_5(B - A) + \alpha_6(A - H) + \alpha_7(G - F) + \alpha_8(H - G)$$
$$\alpha_9(E - D) + \alpha_{10}(F - E) + \alpha_{11}(D - C) + \alpha_{12}(C - B)$$

por lo que si $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4$, el primer término se anula y de manera semejante los otros dos si resultan iguales todas las áreas.
Para la matriz Hessiana se tiene

$$2 H_t = B_t^t B_t + G_t$$

como G_t sólo depende de la diferencia de las áreas de los

triángulos, entonces claramente $G_t = 0$ y $H_t \geq 0$. ∎

De este Corolario garantiza que en caso de exitir una red con todos los triángulos de las celdas de la misma área, entonces la minimización de este funcional conduce a dicha red, y en caso de que no exista, se acerca lo más posible a una red con esta característica. El problema de la existencia de una red convexa es muy dependiente de la frontera de la región y de la distribución de puntos sobre la misma.
Resulta que la definición de red convexa esbozada cuando se mencionó el funcional de Castillo podía clasificar como convexa a una red y darse el caso de que esta en realidad no lo fuera (ver fig. 2.15), con la nueva forma del funcional de área de Barrera y Pérez esta situación se puede solucionar, dando la siguiente definición:
Definición.- Se dice que una red es convexa si para todos los triángulos de todas las celdas se satisface que

$$\text{área(triángulo)} > 0$$

A partir del último corolario y teniendo en cuenta la definición de red convexa, resulta que para el caso presentado en la fig. 2.15 si se usa el funcional de área de Barrera y Pérez, esta situación no da una solución, ya que las áreas de los triángulos FGI, GHF, GFD y DEF son negativas y por consiguiente no clasificaría esta red como convexa, lo que no sucede con el funcional de área de Castillo, según se explicó anteriormente.

2.2.3. FUNCIONALES COMPUESTOS

En 1987 Castillo [13] propuso generar redes usando el funcional

$$C_\sigma = \sigma f_\ell + (1-\sigma) f_c$$

donde $0 \leq \sigma \leq 1$ y f_ℓ es el funcional discreto de longitud y f_c es el

funcional discreto de la suma de las áreas de las celdas (ver Castillo [13]), que luego fue modificado por Barrera y Pérez al cambiar f_c por f_a, esto es, el funcional de la suma del área de los cuatro triángulos de cada celda, y f_ℓ por $f_{\ell,\tau}$, quedando el funcional compuesto de Barrera y Pérez como

$$f_{\sigma,\tau} = \sigma f_{\ell,\tau} + (1-\sigma) f_a \qquad (2.34)$$

que es el que se usa en la experimentación numérica del capítulo IV.

2.3. PROPIEDADES DEL FUNCIONAL DISCRETO DE SUAVIDAD

Ivanienko [36] resuelve el problema de obtener un óptimo del funcional (2.23) buscando un cero del gradiente, o sea, resuelve el siguiente sistema de ecuaciones no lineales:

$$\frac{\partial f_s}{\partial P_{i,j}} = 0$$

para lo que emplea un método especial de tipo quasi-Newton.
El problema fundamental que tiene su método es que necesita una red inicial convexa, pues todas las áreas de los triángulos deben ser positivas para que la aplicación esté bien definida. En un trabajo posterior [37], propone corregir la restricción de que la red inicial sea convexa, usando una tolerancia para la negatividad de las áreas de los triángulos, asignando:

$$\text{area}(P,Q,S) = \begin{cases} \text{area}(P,Q,S) & \text{si area}(P,Q,S) > \varepsilon \\ \varepsilon & \text{si area}(P,Q,S) \leq \varepsilon \end{cases}$$

donde P,Q,S son los vértices de los triángulos de las celdas y

$$\varepsilon = \max\left(\alpha \frac{|Z|}{N + 0.01}, \varepsilon_1\right)$$

y Z es el doble del área total de los triángulos con área negativa, N es el número de estos triángulos, ε_1 es una cota inferior y α lo elige experimentalmente.

Con el propósito de poder implementar de manera segura el funcional de suavidad, usando métodos de optimización sin restricciones que usan el gradiente de las funciones, en 1990 Barrera y otros [6] reinterpretan el funcional en el contexto geométrico que a continuación se describe, y hacen ver que el funcional discreto de Suavidad presentado por Ivanienko [36] y [37], trata de obtener mallas con celdas rectangulares, además de proponer una regularización del funcional, para que pueda ser usado de forma segura en los métodos que aparecen en el capítulo III.

Se descompone el funcional como la suma de funcionales sobre triángulos:

Considérese el triángulo PQS y el funcional sobre él:

$$\varphi(P,Q,S)= \frac{\|Q-P\|^2+ \|S-P\|^2}{\det(Q-P,S-P)} \qquad (2.35)$$

(ver fig. 2.25)

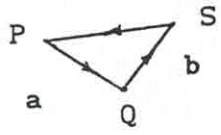


fig.2.25.- Triángulo orientado positivo con a=Q-P y b=S-P

Sean $\mathbf{a} = Q-P$ y $\mathbf{b} = S-P$. Sea entonces

$$\varphi(P,Q,S) = f(\mathbf{a},\mathbf{b}) = \frac{\|\mathbf{a}\|^2+ \|\mathbf{b}\|^2}{\mathbf{a}^t J_2 \mathbf{b}} \qquad (2.36)$$

Propiedades:

i) $f(a,b) = -f(b,a)$ y $f(-a,b) = -f(a,b)$ (2.37)

ii) $f(\alpha a, \alpha b) = f(a,b)$ para $\alpha \in \mathbb{R}$ (2.38)

Lema 2.5.- Sea $\Delta = a^t J_2 b$, entonces

$$\frac{\partial \Delta}{\partial a} = +J_2 b \quad ; \quad \frac{\partial \Delta}{\partial b} = -J_2 a \tag{2.40}$$

Lema 2.2.- Sea

$$f(a,b) = \frac{\|a\|^2 + \|b\|^2}{a^t J_2 b} \tag{2.41}$$

entonces,

$$\frac{\partial f}{\partial a} = \frac{2a + J_2 b f}{a^t J_2 b} \tag{2.42}$$

$$\frac{\partial f}{\partial b} = \frac{2b - J_2 a f}{a^t J_2 b} \tag{2.43}$$

Lema 2.6.- Si (a,b) es un punto crítico de f, entonces $f(a,b)=\pm 2$

Demostración.-

$$(a,b) \text{ mínimo} \Rightarrow \frac{\partial f}{\partial a} = \frac{\partial f}{\partial b} = 0$$

$$\Rightarrow 2a = -J_2 b f \tag{2.44}$$

$$2b = J_2 a f$$

$$\Rightarrow 2J_2 b = J_2^2 a f = -af$$

$$\Rightarrow 2.2a = 2(-J_2 b)f = (-f)(-af) = f^2 a$$

$$\Rightarrow f^2 = 4 \Rightarrow f = \pm 2$$

∎

Teorema 2.3.- En los puntos críticos se tiene que $\|a\|=\|b\|$, es decir, el triángulo PQS es isósceles.

Demostración.-

Como $J_2^t J_2 = I_2$ y de (2.44) se tiene que

$$2\,\mathbf{a}^t\mathbf{a}_2 = \frac{f^2}{2}\,\mathbf{b}^t\mathbf{b} \tag{2.45}$$

Ya que $(\mathbf{a},\mathbf{b})$ es un punto crítico, entonces

$$\frac{f^2}{2} = 2$$

con lo que se tiene que (2.45) $\Rightarrow \|\mathbf{a}\| = \|\mathbf{b}\|$

Además,

$$\mathbf{a} = -\frac{f}{2}\,J_2\mathbf{b}$$

luego

$$\mathbf{b}^t\mathbf{a} = -\frac{f}{2}\,\mathbf{b}^t J_2\mathbf{b} = 0 \tag{2.46}$$

lo que quiere decir que f alcanza su punto crítico sobre triángulos rectángulos isósceles y que para una orientación son mínimos y para otra son máximos. ∎

Lema 2.7.- Sea $\varphi(P,Q,S)=f(Q-P,S-P)=f(\mathbf{a},\mathbf{b})$, entonces

$$\frac{\partial\varphi}{\partial P} = -\frac{\partial f}{\partial \mathbf{a}} - \frac{\partial f}{\partial \mathbf{b}}$$

$$\frac{\partial\varphi}{\partial Q} = \frac{\partial f}{\partial \mathbf{a}}$$

$$\frac{\partial\varphi}{\partial S} = \frac{\partial f}{\partial \mathbf{b}}$$

Lema 2.8.- Sea $\varphi(P,Q,S)=f(Q-P,S-P)=f(\mathbf{a},\mathbf{b})$, entonces

$$\frac{\partial^2\varphi}{\partial P^2} = \frac{\partial^2 f}{\partial a^2} + \frac{\partial^2 f}{\partial b^2} + 2\frac{\partial^2 f}{\partial b \partial a}$$

$$\frac{\partial^2\varphi}{\partial Q \partial P} = -\frac{\partial^2 f}{\partial a^2} - \frac{\partial^2 f}{\partial a \partial b}$$

$$\frac{\partial^2\varphi}{\partial S \partial P} = -\frac{\partial^2 f}{\partial b^2} - \frac{\partial^2 f}{\partial b \partial a}$$

$$\frac{\partial^2\varphi}{\partial Q^2} = \frac{\partial^2 f}{\partial a^2}$$

$$\frac{\partial^2\varphi}{\partial S \partial Q} = \frac{\partial^2 f}{\partial b \partial a}$$

$$\frac{\partial^2\varphi}{\partial S^2} = \frac{\partial^2 f}{\partial b^2}$$

y por tanto,

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial a^2} + \frac{\partial^2 f}{\partial b^2} + 2\frac{\partial^2 f}{\partial b \partial a} & -\frac{\partial^2 f}{\partial a^2} - \frac{\partial^2 f}{\partial a \partial b} & -\frac{\partial^2 f}{\partial b^2} - \frac{\partial^2 f}{\partial b \partial a} \\ -\frac{\partial^2 f}{\partial a^2} - \frac{\partial^2 f}{\partial a \partial b} & \frac{\partial^2 f}{\partial a^2} & \frac{\partial^2 f}{\partial b \partial a} \\ -\frac{\partial^2 f}{\partial b^2} - \frac{\partial^2 f}{\partial b \partial a} & \frac{\partial^2 f}{\partial b \partial a} & \frac{\partial^2 f}{\partial b^2} \end{bmatrix}$$

Demostración.- Cálculo directo.

Lema 2.9.- Sea f(a,b) la definida, entonces su Hessiana (H) es singular.

Demostración.-

A partir de la expresión de H obtenida en el lema 2.5, puede

verse que la primera fila es la suma de la segunda y la tercera con el signo cambiado. ∎

Regularización del Funcional Discreto de Suavidad

Como el denominador del funcional f_s se puede hacer muy pequeño o cero aun para redes convexas, es conveniente controlar esta situación permitiendo que el funcional acepte valores del área muy pequeños en valor absoluto. Ya se describió la implementación hecha por Ivanienko, ahora se describe la que en este trabajo se denomina funcional regularizado de suavidad.

Es natural tratar de resolver el problema antes mencionado aproximando la función

$$f(t) = \frac{1}{t}$$

por

$$\tilde{f}(t) = \begin{cases} \dfrac{1}{t} & \text{para } t \geq \varepsilon_c \\ \dfrac{a}{(t+\varepsilon)^2} & \text{para } -\varepsilon_c < t < \varepsilon_c \end{cases} \qquad (2.47)$$

(ver fig. 2.26)

donde ε_c es un valor determinado experimentalmente para cada malla.

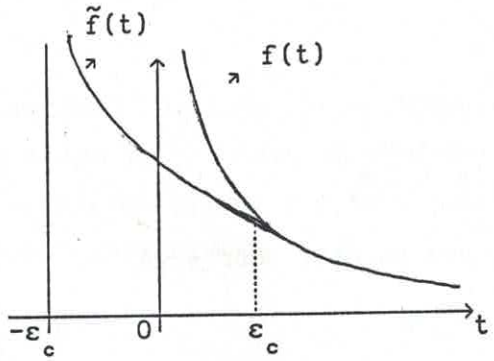


fig. 2.26. Aproximación de f(t) por $\tilde{f}(t)$

La idea está en determinar los valores de a y ε que hagan que el

paso de una expresión a la otra se realice de forma continuamente diferenciable. Para ello, en el punto problemático ε_c debe satisfacerse que:

$$\tilde{f}(\varepsilon_c) = f(\varepsilon_c) \tag{2.48}$$

$$\tilde{f}'(\varepsilon_c) = f'(\varepsilon_c) \tag{2.49}$$

De (2.48) se tiene

$$\left.\frac{a}{(t+\varepsilon)^2}\right]_{\varepsilon=\varepsilon_c} = \frac{1}{\varepsilon_c}$$

$$\Rightarrow \quad a\varepsilon_c = (\varepsilon_c + \varepsilon)^2$$

Por otro lado

$$f'(t) = \frac{-2a}{(t+\varepsilon)^2} \qquad \Rightarrow \qquad f'(\varepsilon_c) = \frac{-2a}{(\varepsilon_c + \varepsilon)^3}$$

luego, $\tilde{f}'(\varepsilon_c) = f'(\varepsilon_c)$ si

$$\frac{-2a}{(\varepsilon_c + \varepsilon)^3} = -\frac{1}{\varepsilon_c^2}$$

$$2a\varepsilon_c^2 = (\varepsilon_c + \varepsilon)^3$$

$$\Rightarrow \quad \varepsilon = \varepsilon_c$$

por lo tanto, resulta

$$a = 4\,\varepsilon_c \tag{2.50}$$

y la función de aproximación que se debe usar será

$$\tilde{f}(t) = \begin{cases} \dfrac{1}{t} & , \text{ para } t \geq \varepsilon_c \\ \dfrac{4\,\varepsilon_c}{(t+\varepsilon_c)^2} & , \text{ para } -\varepsilon_c < t < \varepsilon_c \end{cases} \qquad (2.51)$$

Esta función es la que se propone usar para sustituir al funcional de suavidad discretizado de Ivanienko, con el fin de evitar que se indefina para valores muy pequeños del área de los triángulos. Como conclusión, el funcional discreto de Suavidad Regularizado que se usa en la implementación computacional para la generación de mallas tiene la siguiente forma:

$$f_S(P,Q,R,S) = \frac{\|P-S\|^2+\|Q-P\|^2}{J(P,Q,S)} + \frac{\|Q-P\|^2+\|R-Q\|^2}{J(Q,R,P)} + \frac{\|S-R\|^2+\|R-Q\|^2}{J(R,S,Q)} + \frac{\|P-S\|^2+\|S-R\|^2}{J(S,P,R)} \qquad (2.52)$$

donde

$$\frac{1}{J(P,Q,S)} = \begin{cases} \dfrac{1}{2\ \text{area}(P,Q,S)} & \text{para } 2\ \text{area}(P,Q,S) \geq \varepsilon_c \\ \dfrac{4\,\varepsilon_c}{(2\ \text{area}(P,Q,S)+\varepsilon_c)^2} & \text{para } 2\ \text{area}(P,Q,S) < \varepsilon_c \\ & \text{y } 2\ \text{area}(P,Q,S) > -\varepsilon_c \end{cases} \qquad (2.53)$$

Para el caso del funcional discreto de ortogonalidad (2.19) aun no se tienen establecidas propiedades equivalentes a las de los otros funcionales, ya que es de muy reciente aparición y está en proceso de investigación actualmente.

C A P I T U L O III

METODOS DE OPTIMIZACION DE GRAN ESCALA PARA EL PROBLEMA DE GENERACION DE REDES OPTIMAS

La obtención de una red óptima sobre una región dada, requiere la minimización de algún funcional. Como en el caso que se trata en este trabajo la frontera de la región se mantiene fija, este funcional depende únicamente de los nodos interiores, esto es, tendrá 2(m-2)(n-2) variables, siendo m la cantidad de puntos en la horizontal y n la cantidad de puntos en la vertical, como ya se había señalado. El problema de optimización es entonces de mayor dimensión en la medida que resulta más fina la red, esto es, según sean mayores n o m. Por lo tanto, esta minimización deriva en un problema de minimización de gran escala, por lo que en este capítulo se describen varios métodos que se han aplicado al problema de generación de redes.

3.1. METODOS DE DESCENSO

El problema que se necesita resolver es el siguiente:

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{Minimizar}}\ f(\mathbf{x}) \tag{3.1}$$

donde $f(\mathbf{x})$ denota una función de $\mathbb{R}^n \rightarrow \mathbb{R}$, acotada inferiormente y que tiene al menos derivadas de segundo orden continuas, y se supone que (3.1) tiene una solución local $\mathbf{x}^*$ en la cual $\nabla f(\mathbf{x}^*)=\mathbf{0}$ y $\nabla^2 f(\mathbf{x}^*)\geq 0$ (matriz Hessiana semi-definida positiva).
Se dice que un vector $\mathbf{d}\in\mathbb{R}^n$ es una dirección de descenso en $\mathbf{x}$ de $\mathbb{R}^n$ si

$$\nabla f(\mathbf{x})^t \mathbf{d} < 0 \tag{3.2}$$

Un método de **descenso** para calcular $\mathbf{x}^*$, es aquel que a partir de un $\mathbf{x}_0$ inicial genera una sucesión de iteraciones de la forma

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad , \tag{3.3}$$

donde $\mathbf{d}_k$ es una dirección de descenso en $\mathbf{x}_k$ y $\alpha_k \in \mathbb{R}$ debe ser positivo, tal que la sucesión satisfaga que

a) $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$, (3.4a)

b) $\lim_k \mathbf{x}_k = \mathbf{x}^*$ (3.4b)

La diferencia entre los distintos métodos de descenso radica en la forma como se elige la dirección $\mathbf{d}_k$, la que depende de la información que se tenga en el punto $\mathbf{x}_k$ de la iteración. Algunas de las posibilidades de elección son:

a) $\mathbf{d}_k = -\mathbf{g}_k (\mathbf{g}_k = \nabla f(\mathbf{x}_k))$

b) $\mathbf{d}_k = -H_k \mathbf{g}_k$, con H_k una matriz de nxn definida positiva

c) $\mathbf{d}_k = [\nabla^2 f(\mathbf{x}_k)]^{-1} \mathbf{g}_k$

El método que corresponde a la elección a) es conocido como Método de Descenso más rápido o Máximo Descenso (Steepest Descent) debido a que la dirección opuesta al gradiente es la mejor dirección de descenso de la función en una vecindad del punto (ver [33]). La elección c) es el conocido Método de Newton y es la más usada siempre que sea posible, ya que tiene una rápida convergencia local. Por último, la elección b) es la que corresponde a los métodos de tipo Newton, pues la matriz H_k aproxima, de alguna manera, a la inversa de la matriz Hessiana en cada punto.

Luego de seleccionar la dirección de descenso que se va a usar, es importante elegir α_k que es el tamaño del vector $\mathbf{d}_k$ o lo que es lo mismo, el tamaño del paso en la dirección de descenso. Por lo general, esto se hace de forma tal que se satisfagan las condiciones suficientes para garantizar la convergencia global del algoritmo. Por ejemplo, se puede elegir α_k tal que se obtenga el mayor decrecimiento de f a lo largo de $\mathbf{d}_k$, esto es,

$$\text{Hallar } \alpha_k \text{ que satisfaga}$$
$$\alpha_k = \arg\min_{\alpha \geq 0} f(\mathbf{x}_k + \alpha \mathbf{d}_k) \tag{3.5}$$

Sin embargo, en la práctica se sustituye por un valor aproximado, pues no tiene sentido buscar el mínimo exacto (ver Barrera-Castellanos [9]), de tal forma que el α_k debe satisfacer las condiciones:

$$\text{i) } f(\mathbf{x}_k + \alpha \mathbf{d}_k) \leq f(\mathbf{x}_k) + c_1 \alpha \, \mathbf{g}_k^t \mathbf{d}_k \tag{3.6a}$$

$$\text{ii) } |\mathbf{g}(\mathbf{x}_k + \alpha \mathbf{d}_k)^t \mathbf{d}_k| \leq |c_2 \, \mathbf{g}_k^t \mathbf{d}_k| \tag{3.6b}$$

con $c_1, \; c_2 \geq 0$

La primera condición garantiza el decrecimiento de la función, pero permite valores de α muy pequeños que podrían hacer que la sucesión $\{\mathbf{x}_k\}$ no converja a $\mathbf{x}^*$. La segunda condición evita lo anterior. Estas reglas se conocen como **condiciones de Wolfe** y son las que se usan en los algoritmos que buscan mínimos en una dirección en el problema de optimización no lineal multidimensional.

En general los métodos que buscan un óptimo aproximado en la línea siguen el esquema siguiente:

1) A partir de una aproximación inicial hallar un intervalo que contenga un mínimo.

2) Los puntos de la iteración se calculan aproximando $\varphi(\alpha)=f(\mathbf{x}_k+\alpha\mathbf{d}_k)$ por una función cuadrática o cúbica usando los valores de la función o el gradiente.

En los métodos de optimización que se describen en lo que sigue, el algoritmo de búsqueda en la línea que se usa es el dado por Moré y Thuente [45], cuya iteración básica es

Dado $\alpha_0 \in [\alpha_{min}, \alpha_{max}]$, $I_0 = [0, \infty]$

Para $k = 0, 1, \ldots$

Seleccionar α_k "protegido" $\in I_k \cap [\alpha_{min}, \alpha_{max}]$

Probar convergencia

Actualizar el intervalo I_k

El término "protegido" se refiere a las reglas que fuerzan la convergencia del algoritmo. Los valores de prueba α_k se seleccionan mediante interpolación cúbica o cuadrática usando los valores de la función o el gradiente en los extremos del intervalo y el último punto probado. Los parámetros α_{min} y α_{max} son la cota inferior y superior sobre el argumento de la función, con el fin de que no se prueben valores que no tengan sentido. En caso de que el método que usa este algoritmo no tenga restricciones, se fijan a un valor mínimo y otro máximo, respectivamente.

Lo más importante de este algoritmo, es que sus autores prueban su convergencia en un número finito de iteraciones. En la práctica, este algoritmo ha resultado muy efectivo y usualmente se prueban a lo sumo dos o tres valores para obtener el α^* satisfactorio.

3.2. METODO DE DIRECCIONES CONJUGADAS

El método de Direcciones Conjugadas [34] fue propuesto inicialmente para resolver sistemas de ecuaciones lineales y extendido posteriormente por Fletcher y Reeves [27] para la solución de mínimos de funciones no lineales. Es muy económico en cuanto a requerimientos de memoria interna, pues las direcciones de descenso dependen sólo del gradiente en el punto actual y la dirección de descenso anterior.

Este método tuvo su origen en la solución de sistemas de ecuaciones lineales con matrices definidas positivas y para este

caso se tiene que ya que el mínimo en la línea es exacto, entonces $g_{k+1}^t d_k = 0$, de donde $g_{k+1}^t g_k = 0$ (ver [35]), lo que implica que β_k queda como

$$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \tag{3.7}$$

El supraíndice FR usado en la expresión anterior, hace referencia a Fletcher y Reeves [26] que fueron los que propusieron esta formulación para el caso lineal. Ellos también propusieron la extensión del método de gradientes conjugados a funciones no cuadráticas, observando que la matriz Hessiana que aparece en la aproximación de la función f no lineal mediante una cuadrática, sólo hace falta para calcular el tamaño del paso que da el mínimo α_k en la dirección de descenso, que podría sustituirse por un proceso que lo calculara usando los valores de la función y el gradiente y también habría que tener en cuenta que como con esto se pierde la propiedad de convergencia en n iteraciones, sería necesario recomenzar después de n ciclos con la dirección de descenso máximo, para garantizar que las direcciones que se generan sean linealmente independientes.

A partir de estas observaciones, propusieron el siguiente algoritmo:

Método de Gradientes Conjugados de Fletcher-Reeves para funciones no lineales

Dado $x_0 \in \mathbb{R}^n$, $d_0 = -g_0$, $k = 0$

Para $k = 0, 1, \ldots,$ hasta converger hacer

Paso 1.- Calcular

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x_k + \alpha d_k)$$

usando un algoritmo de búsqueda en la línea

Paso 2.- $x_{k+1} = x_k + \alpha_k d_k$

Paso 3.- Si $k<n$ **entonces**

$$\beta_k = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$$

Si $k \equiv 0 \bmod n$ entonces

$$\beta_k = 0$$

Paso 4.- $d_{k+1} = -g_{k+1} + \beta_k d_k$

Polak y Ribiere [51] consideran que como la función no es cuadrática, no tiene por que satisfacerse la condición (3.18) de ortogonalidad de los gradientes, y por ello proponen tomar β_k como

$$\beta_k^{PR} = \frac{g_{k+1}^t (g_{k+1} - g_k)}{g_k^t g_k} \qquad (3.8)$$

La formulación de Hestness-Stiefel [35] omite la suposición de búsqueda exacta, con lo que β_k es de la forma

$$\beta_k^{HS} = \frac{g_{k+1}^t (g_{k+1} - g_k)}{d_k^t (g_{k+1} - g_k)} \qquad (3.9)$$

dando origen a las tres variantes correspondientes para el método

de gradientes conjugados no lineal.

En la práctica sucede que la elección de β_k tiene repercusión en el comportamiento del método de gradientes conjugados. De esta forma se ha observado que el método de Fletcher-Reeves (FR) es algo errático: unas veces es tan eficiente como el de Polak-Ribiere (PR) y Hestness-Stiefel (HS), pero con frecuencia es mucho más lento. Los métodos de HS y PR se comportan de manera similar y se prefieren al de FR. Gilbert y Nocedal proponen la siguiente modificación del método de PR, que es globalmente convergente (nótese que aquí se permiten valores negativos para β_k):

$$\beta_k = \begin{cases} -\beta_k^{FR} & \text{si } \beta_k^{PR} < -\beta_k^{FR} \\ \beta_k^{PR} & \text{si } |\beta_k^{PR}| < \beta_k^{FR} \\ \beta_k^{FR} & \text{si } \beta_k^{PR} > \beta_k^{FR} \end{cases}, \quad \text{para } k \geq 0 \qquad (3.10)$$

y es la que se usa en la implementación que se hace para el problema que aborda este trabajo y que más adelante se describe.

La parte más costosa del método de gradientes conjugados es la búsqueda del óptimo en la dirección de descenso. Para hacerlo más económico, es necesario que la búsqueda sea imperfecta, esto es, que el cálculo del mínimo a lo largo de la línea o dirección de descenso se sustituya por el de una aproximación que satisfaga condiciones generales de descenso como las condiciones de Wolfe. Sin embargo, esto puede ocasionar que la dirección $\mathbf{d}_{k+1}$ dada por

$$\mathbf{d}_{k+1} = -\mathbf{g}_{k+1} + \beta_k \mathbf{d}_k$$

no sea de descenso, ya que

$$\mathbf{d}_{k+1}^t \mathbf{g}_{k+1} = -\|\mathbf{g}_{k+1}\|^2 + \beta_k \mathbf{d}_k^t \mathbf{g}_{k+1}$$

y si $\mathbf{x}_{k+1}$ es tal que

$$|\beta_k d_k^t g_{k+1}| > \|g_{k+1}\|_2^2$$

significaría que d_{k+1} no fuera una dirección de descenso. Esto se puede resolver usando un recomienzo, es decir, haciendo $d_{k+1} = -g_{k+1}$, aunque frecuentes recomienzos pueden afectar la eficiencia del método, por lo cual a pesar de que la búsqueda en la línea sea imperfecta debe ser, no obstante, una buena aproximación al mínimo real.

La búsqueda en la línea requiere además, un valor inicial bueno para α_k, lo que es crucial si se quiere obtener un algoritmo eficiente. Fletcher obtuvo experimentalmente que en el gradiente conjugado el decrecimiento de la función de x_{k-1} a x_k era del mismo orden de magnitud que el de x_k a x_{k+1}. Suponiendo que $f(x)$ es cuadrática, se tiene que el tamaño del paso en la iteración anterior α_{k-1} y la actual α_k están relacionados por la expresión

$$\alpha_k = \alpha_{k-1} \frac{d_{k-1}^t g_{k-1}}{d_k^t g_k} \qquad (3.11)$$

y este es el valor inicial a tomar para comenzar la búsqueda en la línea, para el caso del método de gradientes conjugados.

El método de gradientes conjugados mejora mucho su efectividad si se recomienza cada n iteraciones, pero debido a que la búsqueda en la línea no es perfecta, es necesario evitar que los gradientes consecutivos no se alejen mucho de la ortogonalidad. Powell [54] encontró que en la práctica cuando

$$|g_{k+1}^t g_k| > \gamma \|g_k\|^2 \qquad (3.12)$$

donde γ es un número positivo menor que uno, es conveniente recomenzar. El valor $\gamma = 0.2$ es ampliamente recomendado.

Otro criterio para hacer un recomienzo, también recomendado por Powell, es cuando la dirección d_{k+1} obtenida no es de "suficiente" descenso, esto es, si

$$\frac{g_{k+1}^{t} d_{k+1}}{\|g_{k+1}\|^{2}} \notin [-1.2, -0.8] \qquad (3.13)$$

La implementación del algoritmo de Gradientes Conjugados que se hizo para usarla en las pruebas de distintos métodos de optimización en el problema que se trata, tiene el siguiente proceso iterativo:

Algoritmo de Gradientes Conjugados

Paso 1.- Escoger $x_0 \in \mathbb{R}^n$, $\varepsilon > 0$

Paso 2.- $\alpha_0 = \frac{1}{\|g_0\|}$; $d_0 = -g_0$; $k = 0$

Paso 3.- Si $\|g_k\| < \varepsilon$ terminar

Si no, ir al paso 4

Paso 4.- Búsqueda en la línea. Hallar

$\alpha_k \approx \underset{\alpha \geq 0}{\text{argmin}}\ f(x_k + \alpha d_k)$ usando el algoritmo de Moré y Thuente [45] y las condiciones de Wolfe con $c_1 = 0.0001$ y $c_2 = 0.1$

Paso 5.- Poner $x_{k+1} = x_k + \alpha_k d_k$

Paso 6.- Calcular la dirección d_{k+1}

Si $k=0 \bmod n$ o $\frac{g_{k+1}^t d_{k+1}}{\|g_{k+1}\|^2} \notin [-1.2, -0.8]$ hacer

$$d_{k+1} = -g_{k+1} \quad ; \quad \alpha_{k+1} = 1$$

Si no

$$d_{k+1} = -g_k + \beta_k d_k$$

$$\beta_k = \begin{cases} -\beta_k^{FR} & \text{si } \beta_k^{PR} < -\beta_k^{FR} \\ \beta_k^{PR} & \text{si } |\beta_k^{PR}| < \beta_k^{FR} \\ \beta_k^{FR} & \text{si } \beta_k^{PR} > \beta_k^{FR} \end{cases}$$

$$\alpha_{k+1} = \alpha_k \frac{d_k^t g_k}{d_{k+1}^t g_{k+1}}$$

En ambos casos hacer $k = k+1$, ir al paso 3.

3.3. METODO DE GRADIENTES CONJUGADOS DE SHANNO

Una de las mayores dificultades que se presenta en el método de gradientes conjugados, es que la búsqueda imperfecta en la línea

ocasiona que las direcciones generadas no resulten conjugadas para funciones cuadráticas, lo que no sucede con los métodos quasi-Newton, ya que si

$$\mathbf{d}_{k+1} = -H_{k+1}\mathbf{g}_{k+1}$$

entonces $\mathbf{d}_{k+1}^{t}\mathbf{g}_{k+1} < 0$, si $H_{k+1} > 0$.

Shanno [57] utiliza la relación existente entre el método de gradientes conjugados y el método de quasi-Newton que usa la actualización de BFGS y que es la siguiente:

i) $\mathbf{d}_{k+1}^{GC} = -R^{*}\mathbf{g}_{k+1}$ si la búsqueda en la línea es perfecta, donde $\mathbf{d}^{GC}$ se refiere a la dirección de descenso del método de gradientes conjugados que usa la expresión de β_k de Hestness-Stiefel (3.9).

ii) $R^{*} = U_{BFGS}(I, \mathbf{s}_k, \mathbf{y}_k)$, es decir, la matriz R^{*} resulta ser la actualización mediante la fórmula de BFGS de la matriz identidad, usando los vectores $\mathbf{s}_k$ y $\mathbf{y}_k$ de la iteración:

$$R^{*} = I - \frac{\mathbf{s}_k\mathbf{y}_k^{t} + \mathbf{y}_k\mathbf{s}_k^{t}}{\mathbf{s}_k^{t}\mathbf{y}_k} + \left(1 + \frac{\mathbf{y}_k^{t}\mathbf{y}_k}{\mathbf{s}_k^{t}\mathbf{y}_k}\right)\frac{\mathbf{s}_k\mathbf{s}_k^{t}}{\mathbf{s}_k^{t}\mathbf{y}_k} \qquad (3.14)$$

Luego es posible ver al método de Gradientes Conjugados como un método quasi-Newton de BFGS en el que en cada iteración se aproxima el inverso de la matriz Hessiana por $R^{*} = U_{BFGS}(I, \mathbf{s}_k, \mathbf{y}_k)$. Esto introduce un cambio significativo en la formulación del método de Gradientes Conjugados, ya que aunque la búsqueda en la línea no sea perfecta, lo único que se requiere es garantizar que $\mathbf{s}_k^{t}\mathbf{y}_k > 0$, lo que se obtiene con facilidad si el algoritmo de búsqueda imperfecta en la línea cumple las condiciones de Wolfe (3.6a) y (3.7).

Shanno [57] usa estas ideas para el método de Gradientes Conjugados de Beale, con lo que obtiene un método de gradientes conjugados, que calcula la dirección de descenso como dos actualizaciones de BFGS: en la primera iteración o en cada

iteración de recomienzo, actualiza la matriz identidad y en las siguientes iteraciones, actualiza la matriz resultante de esta primera actualización. El usa un algoritmo de interpolación cúbica para la búsqueda en la línea, y el criterio de Powell (3.12) para el recomienzo. Una experimentación computacional extensa indica que en cada recomienzo el método pierde un poco de su efectividad, por lo que propone usar la actualización de la Identidad escalada:

$$\tilde{H}_t = U_{BFGS}(\gamma_t I, s_t, y_t) \quad , \tag{3.15}$$

donde

$$\gamma_t = \frac{s_t^t y_t}{y_t^t y_t} \quad . \tag{3.16}$$

La conveniencia de este escalamiento está todavía abierta a discusión y se usa en base a la experiencia computacional.
El algoritmo de Gradientes Conjugados implementaco por Shanno y que se reporta en [58], es el siguiente:

Algoritmo de Gradientes Conjugados de Shanno [58]

Paso 1.- Escoger $x_0 \in \mathbb{R}^n$, $\varepsilon > 0$

Paso 2.- $\alpha_0 = \dfrac{1}{\|g_0\|}$; $d_0 = -g_0$; $k = 0$

Paso 3.- Si $\|g_k\| < \varepsilon$ terminar

Si no, hacer $k = k+1$, ir al paso 4

Paso 4.- Búsqueda en la línea. Hallar

$\alpha_k \approx \underset{\alpha \geq 0}{\operatorname{argmin}} f(x_k + \alpha d_k)$ usando interpolación

cúbica protegida, probando siempre dos puntos

como mínimo y las condiciones de Wolfe con

$c_1 = 0.0001$ y $c_2 = 0.9$

Paso 5.- Poner $x_{k+1} = x_k + \alpha_k d_k$

Paso 6.- Calcular la dirección d_{k+1}

Si $k=0 \bmod n$ o $|g_{k+1}^t g_k| \geq 0.2\|g_k\|^2$ hacer

$k \longrightarrow t$; $d_{t+1} = -\tilde{H}_t g_{t+1}$; $\alpha_{t+1} = 1$

con $\tilde{H}_t = U_{BFGS}(\gamma_t I, s_t, y_t)$; $k = k+1$

ir al paso 3.

Si no

$d_{k+1} = -U_{BFGS}(\tilde{H}_t, s_k, y_k) g_{k+1} = -H^*_{k+1} g_{k+1}$

$\alpha_{k+1} = \alpha_k \dfrac{d_k^t g_k}{d_{k+1}^t g_{k+1}}$; $k = k+1$

ir al paso 3.

A partir de las fórmulas que aparecen se puede verificar que no es necesario almacenar ninguna de las dos matrices anteriormente indicadas y que sólo se requiere almacenar los siete vectores y_k, s_k, y_t, s_t, g_{k+1}, x_{k+1} y x_k.

3.4. METODO DE MEMORIA LIMITADA DE NOCEDAL O L-BFGS

Las matrices obtenidas por el método de BFGS son vistas por

Nocedal [47] como la matriz en el paso anterior más una actualización, esto es,

$$H_{k+1} = H_k - \frac{H_k y_k s_k^t + s_k y_k^t H_k}{s_k^t y_k} + \left(1 + \frac{y_k^t H_k y_k}{s_k^t y_k}\right)\frac{s_k s_k^t}{s_k^t y_k}$$

$$H_{k+1} = H_k + U(s_k, y_k, H_k) \qquad (3.17)$$

En la forma tradicional de los métodos de quasi-Newton H_{k+1} se escribe sobre H_k, lo que requiere n(n+1)/2 localizaciones (pues H_k es simétrica y sólo es necesario almacenar una de las dos mitades). En esta nueva forma, como no se dispone de la memoria para almacenar la matriz, sólo se guardan los vectores que la componen y esto equivale a guardar en cada iteración la corrección U, es decir,

$$H_1 = H_0(s_0, y_0, H_0)$$

$$H_2 = H_1 + U(s_1, y_1, H_1) = H_0 + U(s_0, y_0, H_0) + U(s_1, y_1, H_1)$$

$$\vdots$$

Sea m el múmero máximo de matrices de corrección U que se pueden almacenar. Como H_0 es diagonal, entonces basta con un vector de dimensión n para su almacenamiento, y para cada matriz de corrección U, sólo es necesario guardar los vectores s_k, y_k, luego el número total de vectores de dimensión n a guardar es igual a 2m+1.

Ya que cada matriz se obtiene usando los vectores anteriores, esto es, para calcular H_k se necesita tener almacenados s_0, $y_0, \ldots, s_{k-1}, y_{k-1}$, una vez que ya se tienen 2m+1 vectores (los 2m de los anteriores y uno correspondiente a la matriz diagonal H_0), sería bueno poder eliminar la información más vieja y hacer uso de la más reciente para construir la siguiente matriz Hessiana para el cálculo de la dirección en la nueva iteración; por ejemplo, se eliminarían primero s_0 y y_0 para almacenar los primeros s_k y y_k para los que ya no hay espacio. Si se pone a

H_{k+1} en forma factorizada:

$$H_{k+1} = (I-\rho_k s_k y_k)^t H_k (I-\rho_k y_k s_k)^t + \rho_k s_k s_k^t$$

$$= V_k^t H_k V_k + \rho_k s_k s_k^t \qquad (3.18)$$

donde

$$\rho_k = \frac{1}{y_k^t s_k} \qquad V_k = I - \rho_k y_k s_k^t \qquad (3.19)$$

resulta que eliminar una corrección (lo que equivale a quitar una pareja de vectores) se logra al hacer V = I y $\rho ss^t = 0$.
Estas matrices que se obtienen eliminando los términos más viejos en cada iteración después que se agota la capacidad de memoria, Nocedal las denomina como matrices de BFGS "especiales" y son las que él usa para un método de tipo quasi-Newton que denomina como L-BFGS (la L viene del inglés por "limited", o sea, BFGS limitado o equivalentemente, BFGS con memoria limitada) como sigue:

Algoritmo L-BFGS [48]

Paso 1.- Seleccionar

$\mathbf{x}_0$, m , $H_0>0$, poner k = 0

Paso 2.- Calcular

$\mathbf{d}_k = -H_k \mathbf{g}_k$

$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$, donde α_k satisface las condiciones de Wolfe y el paso inicial α_k^0 es 1 usando el algoritmo de Moré y Thuente y las condiciones de Wolfe $c_1 = 0.0001$ y $c_2 = 0.9$

Paso 3.- Asignar

$\bar{m} = \min\{k, m-1\}$

actualizar H_0 $\bar{m}+1$ veces usando las parejas de vectores $\{\mathbf{s}_j, \mathbf{y}_j\}$ para $j = k-\bar{m}, \ldots, k$, de la siguiente forma:

$$H_{k+1} = (V_k^t \ldots V_{k-\bar{m}}^t) H_0 (V_{k-\bar{m}} \ldots V_k)$$
$$+ \rho_{k-\bar{m}} (V_k^t \ldots V_{k-\bar{m}+1}^t) \mathbf{s}_{k-\bar{m}} \mathbf{s}_{k-\bar{m}}^t (V_{k-\bar{m}+1} \ldots V_k)$$
$$+ \rho_{k-\bar{m}+1} (V_k^t \ldots V_{k-\bar{m}+2}^t) \mathbf{s}_{k-\bar{m}+1} \mathbf{s}_{k-\bar{m}+1}^t (V_{k-\bar{m}+2} \ldots V_k)$$
$$\vdots$$
$$+ \rho_k \mathbf{s}_k \mathbf{s}_k^t$$

Paso 4.- Asignar k = k+1 , ir al paso 2

La versión de este algoritmo que se usó para las pruebas con los funcionales discretos del capítulo II, fue codificada por Liu y Nocedal quienes nos la facilitaron para su uso.

3.5. METODO DE NEWTON TRUNCADO CON BUSQUEDA EN LA LINEA

Un método clásico para resolver el sistema de ecuaciones no lineales que surge al plantear la condición necesaria para un mínimo, $\mathbf{g}(\mathbf{x})=0$, es el método de Newton, el que dada una estimación inicial $\mathbf{x}_0$, calcula una sucesión de direcciones $\{\mathbf{p}_k\}$ e

itera según:

Para k = 0 hasta converger hacer

Resolver

$$B(x_k)p_k = -g(x_k) \qquad (3.20)$$

$$x_{k+1} = x_k + p_k$$

El método de Newton es importante porque da un estándar para comparar métodos de convergencia rápida para resolver el problema de minimización de una función. Una forma de caracterizar la convergencia superlineal es que el vector de descenso calculado debe aproximarse al de Newton en magnitud y dirección. Los aspectos positivos y negativos del método de Newton se pueden resumir como sigue. En el lado positivo, el algoritmo es localmente y cuadráticamente convergente para f suficientemente suave, esto es, para x_0 suficientemente cercana a un mínimo local x^*, existe una constante C tal que

$$\|x_{k+1} - x^*\| \leq C \|x_k - x^*\|^2$$

Sus desventajas son, sin embargo:

i) No es globalmente convergente, es decir, no converge a partir de cualquier punto inicial, independientemente de cuán lejos esté la solución.

ii) No está definido en puntos donde la matriz Hessiana es singular.

iii) Para problemas no convexos no genera necesariamente una sucesión de direcciones de descenso.

iv) Debe resolverse en cada iteración un sistema lineal n dimensional

v) Debe darse la expresión analítica de la matriz Hessiana.

En general, el método de Newton se puede modificar para evitar sus inconv e nientes , excepto el iv), según la iteración siguiente:

Iteración Principal

Calcular $f(\mathbf{x}_k)$, $g(\mathbf{x}_k)$ y $\bar{B}(\mathbf{x}_k)$

probar convergencia

Iteración Menor

Calcular $\mathbf{p}_k$ **tal que**

$\bar{B}_k \mathbf{p}_k = -\mathbf{g}_k$

donde $\bar{B}_k$ es alguna aproximación definida positiva a la Hessiana en $\mathbf{x}_k$

Calcular algún α_k que satisfaga las condiciones de Wolfe con $\alpha_k^0 = 1$

Asignar $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$

La iteración principal controla el comportamiento global del método, por otro lado, asume un papel importante cuando se está cerca de una solución $\mathbf{x}^*$, ya que en una vecindad de él la matriz Hessiana es definida positiva y la dirección $\mathbf{p}_k$ de búsqueda es la de Newton. Más aún, un tamaño de paso α_k=1 resulta aceptable en esta región, y el algoritmo posee la misma velocidad asintótica que el método de Newton.

Ya que las ventajas del método de Newton son principalmente locales, parece no haber justificación para emplear tanto esfuerzo en obtener una solución exacta de las ecuaciones de Newton modificadas, cuando se está lejos de la solución del problema de minimización. Sería entonces más conveniente resolver dichas ecuaciones usando un método iterativo que calcule una aproximación de su solución y que tenga poco requerimiento de memoria, dado que se desea resolver un problema de gran escala. Por tanto, existe una relación estrecha entre la cantidad de trabajo necesario para calcular la dirección de búsqueda y la precisión con que se resuelven las ecuaciones de Newton.

Una medida natural e independiente de la escala, es el residual

relativo

$$\frac{\|r_k\|}{\|g_k\|} \qquad (3.21)$$

donde si p_k es la dirección de descenso a calcular, entonces r_k está dado por

$$r_k = \bar{B}_k p_k + g_k$$

El método de Newton Truncado se basa entonces en "truncar" la iteración de Gradientes Conjugados Precondicionado cuando se aplica a las ecuaciones de Newton (3.20), cuya solución aproximada se obtiene cuando (3.21) es suficientemente pequeño. El algoritmo es el siguiente, donde M es una matriz precondicionadora y el parámetro *macheps* hace referencia a la precisión de la máquina (ver Gill y otros [33]):

Algoritmo de Gradientes Conjugados Lineal Precondicionado

Paso 1.- $p_0 = 0$, $r_0 = -g$, $i = 0$

Paso 2.- $Mz_i = r_i$, $i = i+1$

Si $i = 1$ entonces $d_i = z_0$, determinar η

Si no $\beta_i = r_{i-1}^t z_{i-1} / (r_{i-2}^t z_{i-2})$

$d_i = z_{i-1} + \beta_i d_{i-1}$

Paso 3.- **Si** $d_i^t B d_i \leq (\text{macheps})^{1/2} * (d_i^t d_i)$

$\Rightarrow$ Curvatura negativa. Hacer

$p = d_1$ si $i = 1$ **y salida (1)**

$p = p_i$ en otro caso y **salida (2)**

Si no, ir al paso 3

Paso 4.- Calcular paso al mínimo:

$\alpha_i = (r_{i-1}^t z_{i-1}) / (d_i^t B d_i)$

Paso 5.- Hacer

$p_i = p_{i-1} + \alpha_i d_i$

$r_i = r_{i-1} - \alpha_i B d_i$

Paso 6.- Criterio de parada

Si $\dfrac{\|r_i\|}{\|g\|} \leq \eta$ $\Rightarrow$ Se satisface el criterio de truncamiento

Hacer $p = p_i$ y **salida (3)**

Si no, ir al paso 2

Como se puede apreciar, hay tres salidas diferentes en el algoritmo:

Salida (1).- El gradiente apunta en la dirección de curvatura negativa, es decir, $(M^{-1}g)^t B(M^{-1}g) < (\text{macheps})^{1/2} * (M^{-1}g)^t (M^{-1}g)$. En este caso la dirección que se toma es la de descenso máximo escalada por el precondicionador M^{-1}, esto es, $p=d_1=-M^{-1}g$.

Salida (2).- La iteración de gradiente conjugado ha encontrado una dirección de curvatura negativa, o sea, tal que $d_i^t B d_i < (\text{macheps})^{1/2} * d_i^t d_i$, se termina entonces tomando como

solución el último $\mathbf{p}_i$ obtenido.

Salida (3).- El algoritmo termina porque se satisface el criterio de truncamiento para la dirección de Newton. En este caso se usa una sucesión $\{\eta_k\}$ llamada "forcing sequence", que se explicará más adelante.

En la práctica es necesario restringir además la cantidad de iteraciones de gradientes conjugados lineal permisibles, ya que no es costeable hacer demasiadas.

El siguiente teorema muestra que cerca del mínimo, el algoritmo termina usando la dirección de Newton Truncado de la salida (3), la que satisface las condiciones de Wolfe para $\alpha_k=1$, por lo que no es necesario hacer búsqueda en la línea.

Teorema 3.1 [22].- Sea $\mathbf{x}_k \rightarrow \mathbf{x}^*$ donde $B(\mathbf{x}^*)>0$, entonces, para $\varepsilon>0$ pequeño, existe k_0 tal que el criterio de salida (3) se satisface y $\alpha_k=1$ es aceptable para $k\geq k_0$. Por tanto, para $k\geq k_0$ el método de Newton Truncado se reduce a lo siguiente

Para $k = k_0$ hasta converger

  Encontrar $\mathbf{p}_k$ que satisfaga

$$B_k \mathbf{p}_k = -\mathbf{g}_k + \mathbf{r}_k \qquad (3.22)$$

  Asignar $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k$

El siguiente teorema da un método constructivo para el algoritmo de Newton Truncado, de forma que poseerá un orden de convergencia 1 ó 2 según se haya prefijado.

Teorema 3.2 [22].- Sea $\mathbf{x}_k \longrightarrow \mathbf{x}^*$ donde $B(\mathbf{x}^*)$ es definida positiva y suponiendo que B cumple la condición de Lipschitz [10] en $\mathbf{x}^*$, entonces:

1) $\mathbf{x}_k \longrightarrow \mathbf{x}^*$ superlinealmente si y sólo si

$$\frac{\|\mathbf{r}_k\|}{\|\mathbf{g}_k\|} \longrightarrow 0 \qquad \text{cuando} \qquad k \longrightarrow \infty \qquad (3.23)$$

2) $x_k \longrightarrow x^*$ con orden (1+t) si y sólo si

$$\limsup_{k \to \infty} \frac{\|r_k\|}{\|g_k\|^{t+1}} < \infty \qquad (3.24)$$

Luego, si se selecciona

$$\eta_k = \min_{0 \le c \le 1} \{c/k, \|g_k\|^t\} \qquad (3.25)$$

para algún $0 \le t \le 1$, se tiene un método de Newton Truncado de orden de convergencia 1+t. La "forcing sequence" da por resultado un algoritmo adaptivo para resolver problemas de optimización sin restricciones. Cuando se está lejos de la solución $\|g_k\|$ es grande y por tanto, se requerirá poco trabajo para obtener una solución que satisfaga la condición de terminación de Newton Truncado (salida (3)); mientras que en la medida que se acerca a la solución ocurre que $\|g_k\| \longrightarrow 0$ y por consiguiente $\eta_k \longrightarrow 0$, obligando al algoritmo de gradientes conjugados a calcular una dirección p_k más cercana a la de Newton, tanto en dirección como en magnitud. Ya que este método se desea usar en problemas de gran escala, esto no sería posible si hubiese que almacenar la matriz Hessiana en memoria. Sin embargo, la iteración de gradientes conjugados lineal sólo necesita el producto $B_k d$, el que se puede obtener usando la aproximación por diferencias

$$B_k d \approx \frac{1}{\sigma} [g(x_k + \sigma d) - g(x_k)] \qquad (3.26)$$

donde σ se escoge según

$$\sigma = \frac{(\text{macheps})^{1/2}}{\|d\|} \qquad (3.27)$$

De esta forma se evita el almacenamiento de la matriz Hessiana, aunque se necesita una evaluación más del gradiente en cada iteración de gradientes conjugados.

3.5.1. IMPLEMENTACION COMPUTACIONAL DEL METODO DE NEWTON TRUNCADO CON BUSQUEDA EN LA LINEA

La descripción del método de Newton Truncado con búsqueda en la línea hecha en el epígrafe anterior, fue implementada de la siguiente forma

Iteración Principal de Newton Truncado que usa Búsqueda en la línea

Paso 1.- Dados $f(x_0)$, $g(x_0)$, poner $k = 0$

Paso 2.- Calcular p_k tal que $B_k p_k \approx -g_k$
usando un método de gradientes conjugados lineal precondicionado.

Paso 3.- Poner
$$\cos(\varphi) = \frac{p^t g}{\|g\|\|p\|}$$

Si $\cos(\varphi) > -10^{-4}$ poner $p = -g$

Si $\|p\| \leq (\text{macheps})^{1/2} \Rightarrow$ **fin (1)**

Paso 4.- Asignar $\alpha_k = 1$

Hallar $\alpha_k \approx \underset{\alpha \geq 0}{\text{argmin}}\ f(x_k + \alpha p_k)$ usando el algoritmo de Moré y Thuente [45], que satisface las condiciones de Wolfe con $c_1 = 0.0001$ y $c_2 = 0.9$

Paso 5.- Poner $x_{k+1} = x_k + \alpha_k p_k$

evaluar $f(x_{k+1})$, $g(x_{k+1})$

Paso 6.- **Si satisface criterio de parada, fin (2)**
Si no, ir al paso 2.

El fin (1) de ejecución puede ser debido a que ya se llegó a la solución y no se ha detectado por el algoritmo, porque las condiciones de parada sean demasiado estrictas.
El paso 2 del algoritmo se hace usando la siguiente versión de un método de gradientes conjugados precondicionado, donde M es una matriz definida positiva, por ejemplo, la diagonal de la matriz

Hessiana en el punto inicial o en cada iteración, siempre que ésta sea definida positiva

Algoritmo de Gradientes Conjugados Lineal Precondicionado para calcular una solución aproximada de las ecuaciones de Newton

Paso 1.- $p_0 = 0$, $r_0 = -g$, $i = 0$

Paso 2.- $Mz_i = r_i$, $i = i+1$

Si $i=1$ entonces $d_i = z_0$, $\eta = \min\{\|g\|, c/k\}$

Si no

Si $i>mxlin \Rightarrow$ **salida (4)**

Si no

$\beta_i = r_{i-1}^t z_{i-1} / (r_{i-2}^t z_{i-2})$

$d_i = z_{i-1} + \beta_i d_{i-1}$, ir al paso 3

Paso 3.- $\sigma = \max (macheps^{1/2}, macheps^{1/2} / \|d_i\|)$

$Bd_i \approx (g(x_k + \sigma d_i) - g(x_k)) / \sigma$

Paso 4.- Si $d_i^t B d_i \leq (macheps)^{1/2} * (d_i^t d_i)$

$\Rightarrow$ Curvatura negativa. Hacer

$p = d_1$ si $i = 1$ **y salida (1)**

$p = p_i$ en otro caso y **salida (2)**

Si no, ir al paso 5

Paso 5.- Calcular paso al mínimo:

$\alpha_i = (r_{i-1}^t z_{i-1}) / (d_i^t B d_i)$

Paso 6.- Hacer

$p_i = p_{i-1} + \alpha_i d_i$

$r_i = r_{i-1} - \alpha_i Bd_i$

Paso 7.- Criterio de parada

Si $\dfrac{\|r_i\|}{\|g\|} \leq \eta \Rightarrow$ Se satisface el criterio de truncamiento

Hacer $p = p_i$ **y salida (3)**

Si no, ir al paso 2

Los valores del parámetro c de la "forcing sequence" y MXLIN del máximo de iteraciones de gradientes conjudos que se permite hacer por cada iteración mayor, son valores que se ajustan haciendo las pruebas numéricas con los funcionales discretos del capítulo II, sobre diferentes regiones y que se reportan en el capítulo IV.

3.6. METODO DE NEWTON TRUNCADO CON ESTRATEGIA DE REGION DE CONFIANZA

La idea de "truncar" las ecuaciones de Newton usando una solución aproximada obtenida de aplicar un método de gradientes conjugados lineal, puede aplicarse también a una iteración principal en la que en vez de hacer una búsqueda en la línea, se trabaje con una estrategia de Región de Confianza [59].
La extensión del método de Newton al caso no lineal se basa en la idea de aproximar la función por su desarrollo de Taylor hasta el segundo orden, es decir, se aproxima por un modelo cuadrático que sólo es válido en un entorno del punto en cuestión, esto es,

$$f(\mathbf{x}+\mathbf{p}) - f(\mathbf{x}) \approx \phi(\mathbf{p}) = \mathbf{g}^t\mathbf{p} + 1/2\ \mathbf{p}^t B\mathbf{p} \qquad (3.28)$$

Los métodos de región de confianza logran obtener una dirección de búsqueda tal que sea la solución de las ecuaciones de Newton, sólo cuando dicho vector esté contenido en una bola de radio Δ donde se "confía" que la cuadrática modela bien a la función no lineal, y si este no es el caso, toman una dirección que es una combinación lineal de la de Newton y la de descenso máximo, tal que su longitud sea igual al radio Δ de confianza. De esta manera, el problema que se debe resolver es

$$\begin{aligned} &\min\ \phi(\mathbf{p}) \\ &\text{s.a. } \|\mathbf{p}\| \leq \Delta \end{aligned} \qquad (3.29)$$

El problema (3.29) siempre tiene solución aunque no necesariamente es única. Por otro lado, la idea de los métodos de

Newton Truncado es, como ya se vió, resolver las ecuaciones de Newton con más exactitud en la medida en que se está más cerca del óptimo, o lo que es lo mismo, en la medida en que la función se parece más a una cuadrática.

El aspecto más interesante es que (3.29) se puede hallar usando un método de gradientes conjugados lineal modificado que controle el tamaño de los vectores de sus iteraciones para adaptarse a la restricción, como sigue:

Algoritmo de Gradientes Conjugados Lineal Precondicionado con Estrategia de Región de Confianza.

Paso 1.- $p = 0$, $r = -g$, $i = 0$

Paso 2.- $Mz_i = r_i$, $i = i+1$

Si $i = 1$ entonces $d_i = z_0$, determinar η

Si no $\beta_i = r_{i-1}^t z_{i-1} / (r_{i-2}^t z_{i-2})$

$d_i = z_{i-1} + \beta_i d_{i-1}$

Paso 3.- Si $d_i^t B d_i \leq (\text{macheps})^{1/2} * (d_i^t d_i)$

Calcular $\tau > 0$ tal que $\|p_i + \tau d_i\| = \Delta$

$p = p_i + \tau d_i$

salida (1)

Si no, ir al paso 4.

Paso 4.- Hacer $p_{i+1} = p_i + \alpha_i d_i$

Si $\|p_{i+1}\| > \Delta$

Calcular $\tau > 0$ tal que $\|p_i + \tau d_i\| = \Delta$

poner $p = p_i + \tau d_i$, salida (2)

Si no, ir al paso 5.

Paso 5.- Calcular paso al mínimo:

$$\alpha_i = (r_{i-1}^t z_{i-1}) \; / \; (d_i^t B d_i)$$

Paso 6.- Hacer

$$p_i = p_{i-1} + \alpha_i d_i$$
$$r_i = r_{i-1} - \alpha_i B d_i$$

Paso 7.- Criterio de parada

Si $\dfrac{\|r_i\|}{\|g\|} \leq \eta$ $\Rightarrow$ Se satisface el criterio de truncamiento

Hacer $p = p_{i+1}$ y **salida (3)**

Si no, ir al paso 2

El seleccionar el paso a la frontera en las salidas (1) y (2) se justifica por el siguiente teorema.

Teorema 3.3 [59].- Sea p_j, $j=0,\ldots,i$, el conjunto de iteraciones generadas por el algoritmo anterior. Entonces, $\phi(p_j)$ es estrictamente decreciente y

$$\phi(p) \leq \phi(p_i) \tag{3.30}$$

Más aún, $\|p_j\|$ es estrictamente creciente para $j = 0,\ldots,i$ y

$$\|p\| > \|p_i\| \tag{3.31}$$

La iteración principal del Método de Newton Truncado con Región de Confianza es

Iteración Principal de Newton Truncado que usa Estrategia de Región de Confianza.

Paso 1.- Dado $x_0 \in \mathbb{R}^n$ y $\Delta_0 > 0$, calcular $f(x_0)$, $g(x_0)$, $k = 0$

Paso 2.- Calcular p_k tal que $B_k p_k \approx -g_k$
usando un método de gradientes conjugados lineal precondicionado con Región de Confianza.

Paso 3.- Poner $x_{k+1} = x_k + p_k$
evaluar $f(x_{k+1})$, $g(x_{k+1})$

Paso 4.- Si satisface criterio de parada, fin
Si no, ir al paso 5.

Paso 5.- Actualizar la Región de Confianza Δ_{k+1}
Poner $k = k + 1$, ir al paso 2

3.6.1. IMPLEMENTACION COMPUTACIONAL DEL METODO DE NEWTON TRUNCADO CON REGION DE CONFIANZA

De la misma forma que se hizo con el método de Newton Truncado con búsqueda en la línea, a continuación se detalla la implementación que se hizo de la variante con Región de Confianza, aunque hay que señalar que en este caso no se ha hecho una implementación muy sofisticada del mismo, pues en experimentos previos el de búsqueda en la línea se comportaba mejor, además de haber sido sugerido por sus autores (comunicación personal en evento de Análisis Numérico, Mérida, México) que este último debe trabajar mejor. No obstante, se usa esta implementación en los experimentos numéricos con las mallas y sus resultados son buenos, como se verá en el próximo capítulo

Iteración Principal de Newton Truncado que usa Estrategia de Región de Confianza.

Paso 1.- Dado $x_0 \in \mathbb{R}^n$, calcular $f(x_0)$, $g(x_0)$, k=0, factor=100

Si $g_0^t B_0 g_0 > 0$ entonces $\Delta_0 = g_0^t g_0 / g_0^t B_0 g_0$

Si $\Delta_0 < 0.1$ entonces $\Delta_0 = \text{factor} * \|x_0\|$

Si $\Delta_0 = 0$ entonces $\Delta_0 = \text{factor}$

Paso 2.- Calcular p_k tal que $B_k p_k \approx -g_k$

usando un método de gradiente conjugado lineal precondicionado

Paso 3.- Poner $x_{k+1} = x_k + p_k$

evaluar $f(x_{k+1})$, $g(x_{k+1})$

Paso 4.- Si $f_{k+1} - f_k > 10^{-4} g_k^t p_k$ (la función no decreció lo suficiente)

Hacer

$\Delta_{k+1} \leftarrow \frac{1}{2} \Delta_k$

$x_{k+1} \leftarrow x_k$

$f_{k+1} \leftarrow f_k$

$g_{k+1} \leftarrow g_k$

ir al paso 7.

Si no, ir al paso 5.

Paso 5.- Si satisface criterio de parada, fin (1)

si no ir al paso 6

Paso 6.- Actualizar la región de confianza

Si $|\phi(p) - (f_{k+1} - f_k)| \leq 0.01 |f_{k+1} - f_k|$ (la cuadrática $\approx$ "suficientemente" a la función)

poner $\Delta_{k+1} = 2\Delta_k$

Si $|\phi(p) - (f_{k+1} - f_k)| > 0.75 |f_{k+1} - f_k|$ (la cuadrática no $\approx$ "suficientemente a la función)

poner $\Delta_{k+1} = \frac{1}{2} \Delta_k$

En otro caso poner $\Delta_{k+1} = \Delta_k$

ir al paso 7

Paso 7.- **Si** $\Delta_{k+1} \leq (\text{macheps})^{1/2}$, **fin (2)**

Si no, k=k+1, ir al paso 2.

El fin (2) en este algoritmo tiene la misma argumentación que para el caso de búsqueda en la línea.

El algoritmo de Gradientes Conjugados Precondicionado que se implementó para esta variante es el siguiente

Algoritmo de Gradientes Conjugados Lineal Precondicionado con Estrategia de Región de Confianza.

Paso 1.- $p = 0$, $r = -g$, $i = 0$

Paso 2.- $Mz_i = r_i$, $i = i+1$

Si $i=1$ entonces $d_i = z_0$, $\eta=\min\{\|g\|, c/k\}$

Si no $\beta_i = r_{i-1}^t z_{i-1} / (r_{i-2}^t z_{i-2})$

$d_i = z_{i-1} + \beta_i d_{i-1}$

Paso 3.- $\sigma = \max (\text{macheps}^{1/2}, \text{macheps}^{1/2} / \|d_i\|)$

$Bd_i \approx (g(x_k + \sigma d_i) - g(x_k)) / \sigma$

Paso 4.- **Si** $d_i^t B d_i \leq (\text{macheps})^{1/2} * (d_i^t d_i)$

Calcular $\tau>0$ tal que $\|p_i + \tau d_i\| = \Delta$

$p = p_i + \tau d_i$

salida (1)

Si no, ir al paso 5.

Paso 5.- **Hacer** $p_{i+1} = p_i + \alpha_i d_i$

Si $\|p_{i+1}\| > \Delta$

Calcular $\tau > 0$ tal que $\|p_i + \tau d_i\| = \Delta$

poner $p = p_i + \tau d_i$, **salida (2)**

Si no, ir al paso 6.

Paso 6.- Calcular paso al mínimo:

$$\alpha_i = (r_{i-1}^t z_{i-1}) \; / \; (d_i^t B d_i)$$

Paso 7.- Hacer

$$p_i = p_{i-1} + \alpha_i d_i$$
$$r_i = r_{i-1} - \alpha_i B d_i$$

Paso 8.- Criterio de parada

$$\text{Si} \quad \frac{\|r_i\|}{\|g\|} \leq \eta \Rightarrow$$

Se satisface criterio de truncamiento

Hacer $p = p_{i+1}$

y **salida (3)**

Al igual que en el caso de Newton Truncado con Búsqueda en la línea, los valores del parámetro c de la "forcing sequence" y MXLIN del máximo de iteraciones de gradientes conjudos que se permite hacer por cada iteración mayor, son valores que se ajustan haciendo las pruebas numéricas con los funcionales discretos del capítulo II, sobre diferentes regiones y que se reportan en el capítulo IV.

CAPITULO IV

IMPLEMENTACION DE LOS METODOS DE OPTIMIZACION PARA GENERACION DE REDES. EXPERIMENTACION NUMERICA

La implementación computacional de los métodos de optimización considerados en el capítulo III para el problema de generación de redes, es el aspecto que se aborda en este capítulo. Se trata acerca del almacenamiento de los nodos de la red y la forma de evaluar los funcionales y sus gradientes. Además, para el caso de los métodos de Newton Truncado se describe la forma de almacenamiento de los elementos no nulos de las matrices Hessianas de los funcionales de longitud, área, suavidad y área ortogonalidad. Al final se reportan los resultados de la experimentación numérica con los distintos funcionales y métodos de optimización.

4.1. ALMACENAMIENTO DE LA MALLA

Para almacenar la red es conveniente considerarla en un arreglo unidimensional, ya que para usar las rutinas de los métodos de optimización debe darse el vector sobre el que se minimiza en un arreglo de dimensión N. En este caso particular, el vector está formado por los nodos interiores de la malla, es decir, para las iteraciones de los métodos no es necesaria la información de los nodos de la frontera de la región. Sin embargo, para la evaluación de los funcionales y sus gradientes sí, pero como todas las rutinas se prepararon usando "reverse communication", esto es, cada vez que se requiere de una evaluación de la función o el gradiente se hace un retorno al programa principal que la llama y que devuelve sólo los valores de dicha o dichas evaluaciones, esto ayuda a poder usar los programas sin tener que hacer modificaciones de los mismos debidas al caso particular al cual se están aplicando. De esta forma la malla completa sólo se tiene en el programa principal que llama a las rutinas que

evalúan la función y el gradiente y a las de optimización sólo se les trasmiten los nodos interiores. Por tales razones, la malla se almacena de la siguiente forma:

En el arreglo RED de dimensión 2xmxn (para una red de mxn, pues cada nodo es un vector de orden 2x1), se ordenan primero los nodos de la frontera que son en total 2(2m*2n-4) y luego los interiores que son 2(m-1)(n-1), según el esquema siguiente

Nodos de la frontera de la región

$(x_{1,1}, y_{1,1}, x_{2,1}, y_{2,1}, \ldots, x_{m,1}, y_{m,1}, x_{m,2}, y_{m,2}, \ldots, x_{m,n}, y_{m,n},$

$x_{m-1,n}, y_{m-1,n}, \ldots, x_{1,n}, y_{1,n}, x_{1,n-1}, y_{1,n-1}, \ldots, x_{1,2}, y_{1,2})$

Nodos del interior de la región

$(x_{2,2}, y_{2,2}, \ldots, x_{2,n-1}, y_{2,n-1}, x_{3,2}, y_{3,2}, \ldots, x_{3,n-1}, y_{3,n-1}, \ldots,$

$x_{m-1,2}, y_{m-1,2}, \ldots, x_{m-1,n-1}, y_{m-1,n-1})$

La frontera de la región está formada a su vez por cuatro fronteras, constituídas cada una por los siguientes puntos

Frontera 1: $x_{1,1}, y_{1,1}, x_{2,1}, y_{2,1}, \ldots, x_{m,1}, y_{m,1}$

Frontera 2: $x_{m,2}, y_{m,2}, x_{m,3}, y_{m,3}, \ldots, x_{m,n}, y_{m,n}$

Frontera 3: $x_{m-1,n}, y_{m-1,n}, x_{m-2,n}, y_{m-2,n}, \ldots, x_{1,n}, y_{1,n}$

Frontera 4: $x_{1,n-1}\ y_{1,n-1}, x_{1,n-2}, y_{1,n-2}, \ldots, x_{1,2}, y_{1,2}$

según se aprecia en la fig. 4.1

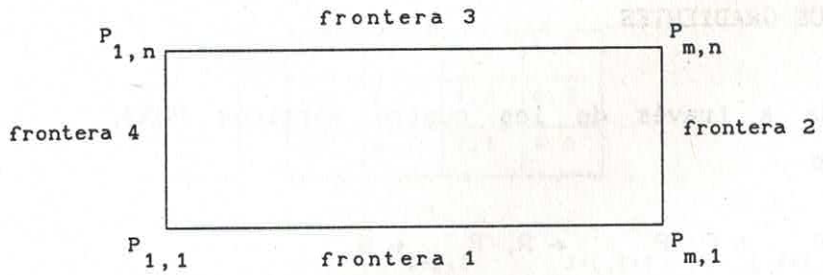


fig. 4.1. Fronteras de la región Ω definidas por los cuatro vértices de la malla

Para las evaluaciones de los funcionales y sus gradientes se necesitan los nodos de la frontera, a pesar de que la minimización se toma sólo sobre los nodos interiores. Para ahorrar cálculos en estas rutinas se forman los funcionales y gradientes sobre cada celda y en el caso del funcional de área (2.33) y de suavidad regularizado (2.52), sobre cada uno de los triángulos de cada celda, pues se calcula una sola vez el área o la longitud de un triángulo o segmento, respectivamente. Por tanto, es conveniente tener dos ciclos que recorrarn los nodos por sus índices i, j, yendo desde i=1 hasta m-1 y desde j=1 hasta n-1, ya que el total de celdas que se evaluan son (m-1)(n-1). Para cada i, j, se necesita tener los nodos $P_{i,j}$, $P_{i+1,j}$, $P_{i+1,j+1}$, $P_{i,j+1}$, pero como esa información se da en el arreglo unidimensional RED, se necesita tener el apuntador a cada uno de ellos y estos están dados por las siguientes expresiones:
Considérese el punto $P_{i,j}$, si está sobre la frontera 1 (j=1), el apuntador a su posición en el arreglo RED es APUN = 2*(m-1). Si está en la frontera 2 (i=m), entonces APUN = 2*(m+j-2). Si pertenece a la frontera 3 (j=n), APUN = 2*(2*m+n-i-2) y si es la frontera 4 (i=1), entonces APUN = 2*(2*m+2*n-3-j). En caso de no estar en ninguna de estas fronteras, es un punto interior y su localización está en APUN = IBO + 2*(n-2)*(i-2)+j-2, donde IBO es la cantidad de nodos de la frontera, cuya expresión se mostró anteriormente. Las coordenadas de $P_{i,j}$ están entonces en RED(APUN+1) y RED(APUN+2) para la x y la y, respectivamente.

4.2. FUNCIONALES Y SUS GRADIENTES

Se define una celda a través de los cuatro vértices PQRS, asignados a los nodos

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

y los triángulos se nombran PQS, suponiendo que su orientación es positiva.

Como ya se mencionó en el acápite anterior, los funcionales y sus gradientes se forman tomando la suma sobre todas las celdas y sobre todos los triángulos de cada una, en caso de que se requiera esta última. Es muy importante tener en cuenta la orientación de los triángulos. En este caso se supone que la malla inicial tiene orientación positiva y así todos los triángulos y celdas se toman con orientación positiva, por lo que hay que ubicar correctamente los vértices a la hora de llamar a las rutinas que calculan las áreas de los triángulos, pues pudieran quedar con un signo que no es el correcto y cambiar completamente los resultados.

El recorrido sobre las celdas de la malla se hace comenzando por la celda (1,1) hacia arriba y luego hacia la derecha. Para el caso de la red de la figura 4.2, las celdas se tomarían en el siguiente orden

$$(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),$$
$$(3,1),(3,2),(3,3),(4,1),(4,2),(4,3)$$

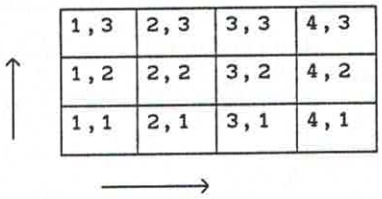


Fig. 4.2. Orden en que se toman las celdas para los cálculos en una red de 5x4

Con esta forma de cálculo se obtienen ventajas en dos aspectos:

i) Dada una celda PQRS, se determinan los apuntadores de sus índices en el vector RED y luego la siguiente celda, por ejemplo, HJKL, tiene el lado común HJ a SR (ver fig. 4.3), no teniendo que volver a calcular los apuntadores para los cuatro nodos, sino unicamente para K y L

ii) Ya que los funcionales son sumas sobre las celdas o los triánmgulos que las conforman, sus gradientes se pueden formar también acumulando los valores sobre cada uno, de esa manera, en cada celda se hacen todos los cálculos necesarios para todos sus nodos, esto es, inclusive las derivadas parciales de primer orden con respecto a todos ellos, que se van acumulando en el vector de gradientes de dimensión N=2(m-2)(n-2), por medio de la función (2.1) que da la localización de la primera derivada respecto a $P_{i,j}$ (que es un vector de 2x1), y asi para los demás nodos de la celda. En resumen, si se tiene la celda $P_{i,j}$, $P_{i+1,j}$, $P_{i+1,j+1}$, $P_{i,j+1}$, se calculan los funcionales para la celda o sus triángulos, según corresponda y luego la derivada con respecto a cada nodo y estas se colocan en el vector del gradiente G en las posiciones k+1 y k+2 con k=2(n-2)(i-2)+j-1.

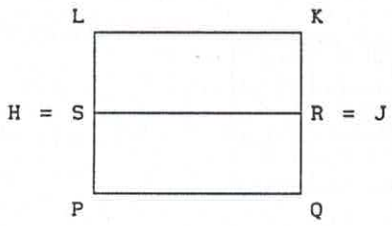


fig. 4.3.- Los lados SR y HJ de dos celdas consecutivas coinciden

En el caso particular del funcional de longitud (2.18), su expresión analítica sobre una celda PQRS y la de su gradiente, son las siguientes:

$$f(Q,R,S) = \|R-Q\|^2 + \tau\|R-S\|^2$$

$$\frac{\partial f}{\partial R} = 2(R-Q) + 2\tau(R-S) \; ; \quad \frac{\partial f}{\partial Q} = -2(R-Q) \quad ; \quad \frac{\partial f}{\partial S} = -2\tau(R-S) \quad (4.1)$$

Nótese que el funcional no depende del punto P de la celda, ya que debido al orden que se sigue para los cálculos sobre las celdas provoca que no haya necesidad de tomar en consideración el punto P, aunque sea un punto interior, como se demuestra a continuación.

En primer lugar, los segmentos de líneas sobre la frontera de la región no se adicionan al funcional, por tanto, comenzando en la celda (1,1), resulta que el punto P no se necesita para evaluar el funcional y como no es un punto interior, no se incluye como variable del problema de minimización y por tanto, no se toma la derivada respecto de él. Siguiendo por las celdas hasta la (1,n-1), sucede lo mismo para todos los P. Para los siguientes grupos de celdas, evidentemente el punto P no se necesita para evaluar el funcional, ya que los segmentos RP y PQ ya fueron adicionados en la celda de la izquierda y de abajo, respectivamente, o simplemente son segmentos de la frontera que no se cuentan. Por consiguiente, los gradientes tampoco dependen del punto P.

Los funcionales que se forman sobre los triángulos de las celdas son el de área de Barrera-Pérez (2.33) y el de suavidad regularizada (2.52), cuyas expresiones y las de sus gradientes sobre el triángulo PQS orientado positivamente, son las siguientes:

Funcional de Area de Barrera-Pérez

$f(P,Q,S) = (1/2\ \alpha)^2$

con $\alpha = \det(Q-P, S-P) = (P-S)^t J_2 (Q-P)$ y $J_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$

$$\frac{\partial f}{\partial P} = \alpha J_2 (Q-S) \qquad \frac{\partial f}{\partial Q} = \alpha J_2 (S-P) \qquad \frac{\partial f}{\partial S} = \alpha J_2 (P-Q) \tag{4.2}$$

Funcional de Suavidad Regularizado

$$f(P,Q,S) = \frac{\|Q-P\|^2 + \|P-S\|^2}{\alpha} \quad \text{con } \alpha \text{ el dado anteriormente}$$

$$\frac{\partial f}{\partial Q} = \frac{2(Q-P) - f\,\frac{\partial \alpha}{\partial Q}}{\alpha} \qquad \frac{\partial f}{\partial S} = \frac{-2(P-S) - f\,\frac{\partial \alpha}{\partial Q}}{\alpha} \tag{4.3}$$

$$\frac{\partial f}{\partial P} = -\frac{\partial f}{\partial Q} - \frac{\partial f}{\partial S}$$

Para ambos casos se tiene una rutina que evalúa el gradiente sobre un triángulo genérico PQS, la cual es llamada para los cuatro triángulos que forman una celda: PQS, QRP, RSQ y SPR, adicionándose al vector de gradiente en la posición correspondiente a los índices sobre la malla, como se ha explicado.

Para el funcional de área ortogonalidad se toma la celda PQRS y su expresión sobre ella, así como la de su gradiente serán

Sea $f(P,Q,R,S) = f_H * f_V$ (ver fig. 4.4)

$a_1 = Q - P \qquad a_2 = R - Q \qquad a_3 = S - R \qquad a_4 = P - S$

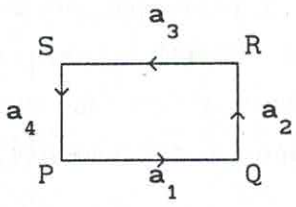


fig. 4.4. Definición de los vectores a_1, a_2, a_3 y a_4 para el funcional de área ortogonalidad

$$f_H(P,Q,R,S) = \|Q - P\|^2 + \|S - R\|^2$$

$$f_V(P,Q,R,S) = \|R - Q\|^2 + \|P - S\|^2$$

$$\Leftrightarrow \qquad f_H(a_1,a_3) = \|a_1\|^2 + \|a_3\|^2$$

$$f_V(a_2,a_4) = \|a_2\|^2 + \|a_4\|^2$$

$$\frac{\partial f_H}{\partial P} = -2(Q - P) = -2a_1 \qquad \frac{\partial f_H}{\partial Q} = 2(Q - P) = 2a_1 \qquad (4.4)$$

$$\frac{\partial f_H}{\partial R} = -2(S - R) = -2a_3 \qquad \frac{\partial f_H}{\partial S} = 2(S - R) = 2a_3 \qquad (4.5)$$

$$\frac{\partial f_V}{\partial P} = 2(P - S) = 2a_4 \qquad \frac{\partial f_V}{\partial Q_2} = -2(R - Q) = -2a_2 \qquad (4.6)$$

$$\frac{\partial f_V}{\partial R} = 2(R - Q) = 2a_2 \qquad \frac{\partial f_V}{\partial S} = -2(P - S) = -2a_4 \qquad (4.7)$$

Las evaluaciones de los funcionales y sus gradientes se realizan en una misma rutina para aprovechar los cálculos comunes a ambas

expresiones.

4.3. NORMALIZACION DE LOS FUNCIONALES

Cualquiera que sea el método de minimización que se pretenda usar, un problema que se debe resolver es decidir cuándo el método convergió. Típicamente se usan criterios sobre las evaluaciones del gradiente y la función, pero que para diferentes escalas de valores pueden resultar o muy estrictos o demasiado relajados. Por esta razón es siempre aconsejable normalizar o reescalar los valores de la función a minimizar para que éstos sólo se muevan en un rango conocido a priori. Una forma de hacerlo es usando el valor exacto o aproximado de la función en el óptimo, siempre que éste se pueda calcular o se conozca de antemano. En el caso de los funcionales que aquí se estudian, este valor se puede estimar teniendo en cuenta las propiedades dadas anteriormente y entonces usar el recíproco del mismo como factor de escala.

FUNCIONAL DE LONGITUD.- Supóngase que la región física es el cuadrado unitario y sobre él se quiere generar una red usando el funcional $f_{\ell,\tau}$ (2.18), entonces las longitudes de los segmentos de la horizontal en el óptimo son las mismas e iguales a 1/m, y equivalentemente para los verticales, es decir, su longitud será de 1/n. Por supuesto, esto es una aproximación grosera del valor en el óptimo para el caso general, pero logra mantener la evaluación del funcional entre 1 y 10.
De esta forma, y teniendo en cuenta que las cuatro fronteras de la región no se adicionan a la evaluación del funcional, su valor óptimo será:

$$f^{*}_{\ell,\tau} = \tau \frac{1}{m^2}(n-2)(m-1) + \frac{1}{n^2}(n-1)(m-2)$$

FUNCIONAL DE AREA DE BARRERA-PEREZ.- En el capítulo II se describió una forma de calcular el área de un polígono a través

de sus vértices. Esta deducción se puede usar para calcular el área de la región si se tiene en cuenta que una vez que se tienen los puntos de la red sobre la frontera, la región se puede considerar como un polígono cuyos vértices son dichos puntos. Como lo que se espera del óptimo del funcional f_a (2.27) es que los cuatro triángulos de todas las celdas sean de la misma área, esto sugiere que

$$\text{area(celda)} = \frac{\text{área de la región}}{\text{cantidad de celdas}} = 2\ (\text{área triángulo}) ,$$

$$\text{area(triángulo)} = \frac{\text{área de la región}}{2(m-1)(n-1)} .$$

Como el funcional es la suma de los cuadrados de todos los triángulos, entonces se tiene que

$$f_a^* = \left[\frac{\text{área de la región}}{2(m-1)(n-1)}\right]^2 4(m-1)(n-1) ,$$

$$f_a^* = \frac{[\text{área de la región}]^2}{(m-1)(n-1)} .$$

FUNCIONAL DE SUAVIDAD REGULARIZADO.- De las propiedades del funcional de Suavidad discreto (2.52) se tiene que en el óptimo el valor del mismo sobre un triángulo es 2, luego la suma sobre todas las celdas es

$$f_S^* = 8(m-1)(n-1)$$

FUNCIONAL DE AREA-ORTOGONALIDAD.- De la misma forma que se planteó para el funcional de longitud, se debe considerar el caso cuando la región es un cuadrado unitario y sobre ella se genera una malla óptima usando el funcional discreto de Area-Ortogonalidad (2.27). Una solución para este caso es la malla uniforme, por lo que de nuevo se pueden considerar los segmentos en el óptimo con longitudes horizontales y verticales de magnitudes 1/m y 1/n respectivamente, por lo que el óptimo

para este caso sería

$$f^{*}_{ao} = 4(m-1)(n-1)/(mn)^2$$

4.4. MATRICES HESSIANAS PARA LOS METODOS DE NEWTON TRUNCADO

En el capítulo II se describen las matrices Hessianas de los funcionales de área de Barrera Pérez y el de Longitud, observándose que son matrices "sparse". Igualmente sucede para el funcional de Suavidad Regularizado y el de Area-Ortogonalidad.
Con el objetivo de ver si los métodos de Newton Truncado mejoran su eficiencia al usar las matrices Hessianas analíticas con respecto al uso de las diferencias finitas para su evaluación, se codificaron estas matrices considerando la ventaja de su estructura llena de ceros y su simetría. Para ello se usa un arreglo unidimensional w que sólo contiene los elementos que no son cero del triángulo superior de la matriz Hessiana.
Dado el ordenamiento que se tiene en el vector RED donde se guarda la información de los nodos de la malla, resulta que aunque son 9 los puntos con segundas derivadas parciales distintas de cero para un $P_{i,j}$ dado, sólo sitúan en el triángulo superior las correspondientes a: $P_{i,j}$, $P_{i,j+1}$, $P_{i+1,j-1}$, $P_{i+1,j}$, $P_{i+1,j+1}$, por lo que son las que se almacenan en w (ver fig. 4.5)

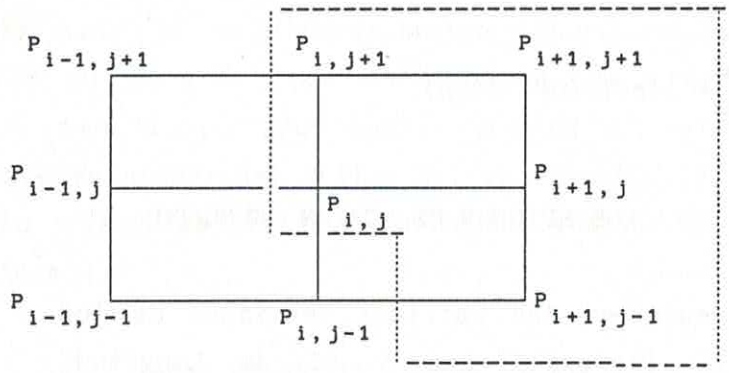


Fig. 4.5.- Para $P_{i,j}$ dado estos son los puntos con segundas parciales distintas de cero. Encerrados en la línea discontinua están los correspondientes al triángulo superior de la matriz Hessiana y que son las que se almacenan.

Cada derivada parcial de segundo orden es una submatriz de orden 2x2 (cada $P_{i,j}$ es un vector de orden 2x1), luego conviene considerar a w como un arreglo formado por un bloque de filas de 10 elementos y un total de N filas (N igual a la cantidad de nodos interiores). Así, para $P_{i,j}$ dado, deben guardarse en w los elementos

$$\frac{\partial^2 f}{\partial P_{i,j}^2} \quad , \quad \frac{\partial^2 f}{\partial P_{i,j+1} \partial P_{i,j}} \quad , \quad \frac{\partial^2 f}{\partial P_{i+1,j-1} \partial P_{i,j}} \qquad (4.8a)$$

$$\frac{\partial^2 f}{\partial P_{i+1,j} \partial P_{i,j}} \quad , \quad \frac{\partial^2 f}{\partial P_{i+1,j+1} \partial P_{i,j}} \qquad (4.8b)$$

que serán las dos filas de 10 elementos correspondientes a las derrivadas parciales respecto de $P_{i,j}$ que se almacenen en w, la primera y la segunda correspondientes a las primeras y segundas filas de cada submatriz. Las derivadas parciales que restan se ubican en el triángulo inferior, pero según el ordenamiento ya fueron calculadas para puntos que preceden en el vector RED a

$P_{i,j}$ y por lo tanto, para tener sus valores basta buscarlos en la posición adecuada del vector w, teniendo en cuenta que

$$\frac{\partial^2 f}{\partial P_{l,h}\partial P_{i,j}} = \left(\frac{\partial^2 f}{\partial P_{i,j}\partial P_{l,h}}\right)^t \qquad (4.9)$$

En los códigos de los programas que calculan las matrices Hessianas, se almacenan sólo estos valores, y en el mismo orden señalado, teniendo en cuenta que cada uno de ellos es una matriz de orden 2x2 y que la cantidad de variables es N=2(m-1)(n-1), para una red de mxn, se tiene que la dimensión necesaria para su almacenamiento es de 10N. También se debe tener presente la orientación de los triángulos en el caso de las funcionales de área y de suavidad regularizado, ya que se codifican las derivadas parciales sobre un triángulo genérico, y como la matriz Hessiana general está formada por las matrices de orden 2x2 de las derivadas parciales de los puntos, hay que considerar en los cálculos la relación (4.9).

El apuntador a las dos filas del vector w donde deben colocarse (o extraerse) las submatrices de las derivadas parciales se calcula mediante la función

$$\phi(u,v) = 2[(n-2)(u-2) + v - 1] - 1$$

para un punto $P_{u,v}$ de una red de dimensión mxn, y luego $k=10(\phi-1)$ y $k=10\phi$ dan respectivamente los apuntadores a la primera y segunda filas cuyos elementos se colocan en el orden dado en (4.8a y 4.8b)

$$\left.\begin{matrix} w(k+1) \\ w(k+2) \end{matrix}\right\} \leftarrow \frac{\partial^2 f}{\partial P_{i,j}^2}$$

$$\left.\begin{matrix} w(k+3) \\ w(k+4) \end{matrix}\right\} \leftarrow \frac{\partial^2 f}{\partial P_{i,j+1}\partial P_{i,j}}$$

$$\left.\begin{matrix} w(k+5) \\ w(k+6) \end{matrix}\right\} \leftarrow \frac{\partial^2 f}{\partial P_{i+1,j-1}\partial P_{i,j}}$$

$$\left.\begin{matrix} w(k+7) \\ w(k+8) \end{matrix}\right\} \leftarrow \frac{\partial^2 f}{\partial P_{i+1,j}\partial P_{i,j}}$$

$$\left.\begin{matrix} w(k+9) \\ w(k+10) \end{matrix}\right\} \leftarrow \frac{\partial^2 f}{\partial P_{i+1,j+1}\partial P_{i,j}}$$

para los dos valores de k.

Usando la misma función ϕ se obtiene la posición de la componente del vector al que se va a aplicar la matriz. Para ello se evalúa ϕ en los índices del punto sobre el cual se toma la segunda derivada parcial, esto es, para

$\frac{\partial}{\partial P_{i,j}\partial P_{u,v}}$, $\phi(i,j)$ da la componente correspondiente en el vector.

A continuación se dan las expresiones de las segundas derivadas parciales de los funcionales discretos presentados en este trabajo, para cuya deducción es conveniente recordar el siguiente lema para el cálculo de derivadas del producto de una función vectorial por una escalar:

Lema.- Sea $F:\mathbb{R}^{2n} \rightarrow \mathbb{R}$, $F(p) = v(p)\varphi(p)$ con $\varphi:\mathbb{R}^{2n} \rightarrow \mathbb{R}$ y $v:\mathbb{R}^{2n} \rightarrow \mathbb{R}^{2n}$, entonces,

$$\frac{\partial F}{\partial p} = \varphi(p)\,\frac{\partial v}{\partial p} + v(p)\left[\,\frac{\partial \varphi}{\partial p}\,\right]^t$$

Funcional de Longitud

$$\frac{\partial^2 f}{\partial R^2} = (2+2\tau)I_2 \quad ; \quad \frac{\partial^2 f}{\partial Q^2} = 2I_2 \quad ; \quad \frac{\partial^2 f}{\partial S^2} = 2\tau I_2$$

$$\frac{\partial^2 f}{\partial R \partial S} = -2\tau I_2 \quad ; \qquad \frac{\partial^2 f}{\partial R \partial Q} = -2I_2$$

Funcional de Area de Barrera-Pérez

$$\frac{\partial^2 f}{\partial P^2} = 1/2 \left[J_2 (Q-S)(Q-S)^t J_2^t \right]$$

$$\frac{\partial^2 f}{\partial Q^2} = 1/2 \left[J_2 (S-P)(S-P)^t J_2^t \right]$$

$$\frac{\partial^2 f}{\partial S^2} = 1/2 \left[J_2 (P-Q)(P-Q)^t J_2^t \right]$$

$$\frac{\partial^2 f}{\partial Q \partial P} = 1/2 \left[J_2 (Q-S)(S-P)^t J_2^t + \alpha J_2 \right]$$

$$\frac{\partial^2 f}{\partial S \partial P} = 1/2 \left[J_2 (Q-S)(P-Q)^t J_2^t + \alpha J_2 \right]$$

$$\frac{\partial^2 f}{\partial S \partial Q} = 1/2 \left[J_2 (S-P)(P-Q)^t J_2^t + \alpha J_2 \right]$$

Funcional de Suavidad Regularizado

$$\frac{\partial^2 f}{\partial P^2} = - \frac{\partial^2 f}{\partial Q^2} - \frac{\partial^2 f}{\partial S^2} \quad ; \qquad \frac{\partial^2 f}{\partial Q^2} = \frac{2I_2 - \left(\frac{\partial \alpha}{\partial Q}\right)\left(\frac{\partial f}{\partial Q}\right)^t - \left(\frac{\partial f}{\partial Q}\right)\left(\frac{\partial \alpha}{\partial Q}\right)^t}{\alpha}$$

$$\frac{\partial^2 f}{\partial S^2} = \frac{2I_2 - \left(\frac{\partial \alpha}{\partial S}\right)\left(\frac{\partial f}{\partial S}\right)^t - \left(\frac{\partial f}{\partial S}\right)\left(\frac{\partial \alpha}{\partial S}\right)^t}{\alpha}$$

$$\frac{\partial^2 f}{\partial P \partial Q} = \frac{-2I_2 + fJ_2 - \left(\frac{\partial \alpha}{\partial Q}\right)\left(\frac{\partial f}{\partial P}\right)^t - \left(\frac{\partial f}{\partial Q}\right)\left(\frac{\partial \alpha}{\partial P}\right)^t}{\alpha}$$

$$\frac{\partial^2 f}{\partial Q \partial S} = \frac{fJ_2 + \left(\frac{\partial \alpha}{\partial S}\right)\left(\frac{\partial f}{\partial Q}\right)^t - \left(\frac{\partial f}{\partial S}\right)\left(\frac{\partial \alpha}{\partial Q}\right)^t}{\alpha}$$

Funcional de Area Ortogonalidad

$$\frac{\partial^2 f}{\partial P^2} = 2(f_H + f_V)\, I_2 + 4(a_1 a_4^t - a_4 a_1^t)$$

$$\frac{\partial^2 f}{\partial Q^2} = 2(f_H + f_V)\, I_2 + 4(a_1 a_2^t + a_2 a_1^t)$$

$$\frac{\partial^2 f}{\partial R^2} = 2(f_H + f_V)\, I_2 - 4(a_3 a_2^t + a_2 a_3^t)$$

$$\frac{\partial^2 f}{\partial S^2} = 2(f_H + f_V)\, I_2 + 4(a_3 a_4^t - a_4 a_3^t)$$

$$\frac{\partial^2 f}{\partial Q\ \partial P} = -2f_V I_2 + 4(a_1 a_2^t + a_4 a_1^t)$$

$$\frac{\partial^2 f}{\partial R\ \partial P} = 0$$

$$\frac{\partial^2 f}{\partial S\, \partial Q} = -\, 4(a_1 a_4^t + a_2 a_3^t)$$

$$\frac{\partial^2 f}{\partial R\, \partial Q} = -2f_H I_2 + 4(a_1 a_2^t + a_2 a_3^t)$$

$$\frac{\partial^2 f}{\partial S\, \partial R} = -2f_V I_2 + 4(a_3 a_4^t + a_2 a_3^t)$$

Es posible usar la diagonal de la matriz Hessiana como precondicionador del método de Gradientes Conjugados que es la iteración menor en los métodos de Newton Truncado descritos en el capítulo anterior, lo que acelera su convergencia. Esto es posible, ya que los elementos de la diagonal de todos los funcionales son estrictamente positivos, lo cual se puede deducir a partir de las expresiones dadas anteriormente, teniendo en cuenta que la diagonal de la matriz total está dada por los elementos de la diagonal de la submatriz de orden 2x2 de la segundas derivadas parciales de un punto respecto a si mismo dos veces.

Se han implementado para su experimentación y uso eficiente en el problema que se quiere resolver, cuatro variantes para cada uno de los Newton Truncado presentados anteriormente, ellas son:

1) Newton Truncado con Búsqueda en la Línea (Región de Confianza) usando diferencias finitas para el producto de matriz Hessiana-vector.

2) Newton Truncado con Búsqueda en la Línea (Región de Confianza) usando la matriz Hessiana "sparse" de las funcionales, lo que conlleva a 10*N localizaciones más de almacenamiento.

3) Newton Truncado con Búsqueda en la Línea (Región de Confianza) usando diferencias finitas para el producto de matriz Hessiana-vector y como precondicionador la diagonal de la matriz Hessiana, para lo que se necesita un N-vector adicional.

4) Newton Truncado con Búsqueda en la Línea (Región de Confianza)

usando la matriz Hessiana "sparse" y como precondicionador la diagonal de la Hessiana, para lo que se necesitan 11 N-vectores adicionales.

La matriz precondicionadora M es un arreglo n-dimensional que contiene la diagonal de la matriz Hessiana para cada funcional, y se calcula iterativamente.

El código diseñado da la posibilidad de experimentar con los parámetros siguientes:

1) Cantidad de iteraciones de Gradientes Conjugados Lineal (MXLIN).

2) Cada qué cantidad de iteraciones se quiere actualizar la matriz Hessiana o el precondicionador (NUPD).

3) Qué valor de c es conveniente en la "forcing sequence" (ver los algoritmos en acápites 3.5 y 3.6).

4.5. RESULTADOS DE LA EXPERIMENTACION NUMERICA CON LOS METODOS DE NEWTON TRUNCADO EN LA GENERACION DE REDES

El problema de obtención de una red óptima fue probado usando el método de Moré-Thuente [45] para la búsqueda en la línea y los algoritmos siguientes (ya descritos en el capítulo III):

1.- Gradientes Conjugados que usa la variante propuesta por Gilbert y Nocedal para la elección del β_k (ver 3.2) y requiere 5N vectores de memoria.

2.- Gradientes Conjugados de Shanno (ver 3.3) que necesita 7N vectores para su realización.

3.- L-BFGS de Nocedal (ver 3.4) que permite 2 actualizaciones de la matriz Hessiana y requiere de 8N vectores en total.

4.- Newton Truncado con búsqueda en la línea (ver 3.5) y cálculo del producto matriz Hessiana-vector por diferencias finitas, precondicionando el Gradiente Conjugado Lineal en la iteración menor con la diagonal de la matriz Hessiana, el cual necesita 8N vectores de memoria; y la variante con la matriz Hessiana analítica "sparse" que requiere 10N vectores adicionales para su almacenamiento, es decir, un total de 18N vectores.

5.- Newton Truncado con región de confianza (ver 3.6) y las

mismas variantes y requerimientos de memoria que el que usa búsqueda en la línea.

Luego de hacer una extensa experimentación numérica sobre distintas regiones con diferentes grados de complejidad, se pudieron ajustar las variables referidas al máximo de iteraciones de gradientes conjugados lineal (MXLIN) permisibles, asi como determinar el valor del parámetro c en la "forcing sequence". En todos los ejemplos se usó la opción de actualizar la diagonal de la matriz Hessiana que se usa como precondicionador, en todas las iteraciones, es decir, NUPD=1.
A continuación se muestran los resultados de las pruebas sobre 20 regiones (Galeria de Rogue, Knupp [41] y otras) con mallas de dimensiones 15x10, 18x20 y 20x25 sobre cada una, es decir, un total de 60 mallas iniciales, usando los funcionales:
1.- Funcional de Longitud (2.18) y Funcional de Area de Barrera-Pérez (2.33) con la combinación convexa

$$f_{comb} = \sigma f_{\ell,\tau} + (1-\sigma) f_a$$

para los valores

a) $\sigma = 1$, $\tau = 1$
b) $\sigma = 0.01$, $\tau = 1$
c) $\sigma = 0$

2.- Funcional de Suavidad Regularizado (2.52) (sólo en aquellas redes iniciales que son convexas).
3.- Funcional de Area-Ortogonalidad (2.27).

En los experimentos numéricos se compararon los siguientes aspectos:
1.- Para el Método de Newton Truncado con Búsqueda en la línea y calculo del producto matriz Hessiana-vector usando diferencias, se prueban distintos valores para el máximo de iteraciones de Gradientes conjugados permisibles (MXLIN). Para esta prueba el parámetro c de la "forcing sequence" se toma como uno y la matriz

precondicionadora se actualiza en todas las iteraciones (NUPD=1) (ver Tabla I, II, III y IV).

2.- Para el método anterior, pero con MXLIN fijo a 10, se prueban dos valores del parámetro c de la "forcing sequence" (ver Tablas V, VI, VII y VIII).

3.- Se comparan las cuatro variantes de los métodos de Newton Truncado y su efectividad para cada funcional (ver tablas IX a la XVIII)

3.- Se comparan los siete métodos de optimización descritos anteriormente sobre los funcionales (ver tablas IX a la XVIII).

Para todas las pruebas hay dos tablas similares excepto por la última columna, pues las comparaciones referidas al lugar que ocupa el indicador de la primera columna (primero, segundo o tercer lugar), se hace por tiempo en un caso y en el otro por cantidad total de evaluaciones de la función y el gradiente. Además, las tablas que comparan los aspectos 1 y 2 se dividieron en dos ya que para el funcional de Suavidad Regularizado no se pudieron hacer las pruebas sobre todas las regiones, y por esa razón se trata aparte.

La lectura común a todas las tablas es la siguiente

1ro,2do,3ro Total de veces que el indicador de la columna 1 fue el 1ro (2do o 3ro) con respecto al resto de de los indicadores de esa columna, para el indicador de la columna final

IF+IG Total de evaluaciones de la función y el gradiente.

#fallos Total de veces que no se pudo decrecer más el valor de la función en la dirección de descenso (puede que se haya alcanzado la solución y la convergencia no se detecte pues se está siendo demasiado exigente, aunque esta no es una salida deseable).

TIEMPO Tiempo total de ejecución: horas: minutos: segundos: centésimas de segundos.

Para cada tabla específica se tienen además las siguientes lecturas:

MXLIN Máximo de iteraciones de Gradientes Conjugados Lineal

permisibles.

FORC Valor del parámetro c de la "forcing sequence".

MET Método de optimización:

POL Gradientes Conjugados en la variante de Gilbert y Nocedal para la evaluación de β.

M=2 L-BFGS de Nocedal con dos actualizaciones de la matriz Hessiana.

SHA Gradientes Conjugados de Shanno.

TRL Newton Truncado con Búsqueda en la línea y producto matriz-vector por diferencias, usando de precondicionador la diagonal de la matriz Hessiana y los parámetros MXLIN=10, c=1 y NUPD=1

TLH Newton Truncado con Búsqueda en la Línea y matriz Hessiana analítica, usando de precondicionador la diagonal de la matriz Hessiana y los parámetros MXLIN=10, c=1 y NUPD=1

TRC Newton Truncado con Estrategia de Región de Confianza y producto matriz-vector por diferencias, usando de precondicionador la diagonal de la matriz Hessiana y los parámetros MXLIN=10, c=1 y NUPD=1.

TLH Newton Truncado con Estrategia de Región de Confianza y matriz Hessiana analítica, usando de precondicionador la diagonal de la matriz Hessiama y los parámetros MXLIN=10, c=1 y NUPD=1.

Como se observa en las tabla I, II, III y IV, el aumentar la cantidad de iteraciones de Gradientes Conjugados conlleva a un mejor comportamiento del método, tanto en cuanto a tiempo total de ejecución como a cantidad de evaluaciones de la función y el gradiente. En particular parece ser que un valor alrededor de 10 para el máximo de iteraciones de Gradientes Conjugados resulta aceptable.En las tablas V, VI, VII y VIII se hace la comparación entre el valor de c=1 y c=0.5 para el parámetro de la "forcing sequence", observándose que el ser más estrictos en la parada de las iteraciones de gradientes conjugados (c=0.5), no mejora los resultados, por lo que se toma el valor c=1 como satisfactorio.

En las tablas IX a la XVIII se pueden ver las distintas realizaciones de los métodos de Newton Truncado y concluirse que aun cuando se disminuye la cantidad de evaluaciones de la función y el gradiente, y que para el caso de los funcionales de área-longitud (σ=0.01, τ=1), área y suavidad regularizado el tiempo de ejecución de los que usan la matriz Hessiana analítica, con respecto a los que calculan el producto matriz-vector por diferencias finitas, es menor, es de notar que no es tan grande la mejoría de uno respecto al otro, lo cual comprueba la eficiencia de la fórmula de diferencias para aproximar este valor, como también tiene a su favor el uso de las diferencias, el que requiere menor cantidad de vectores de memoria al no tener que almacenar la matriz Hessiana, que aunque "sparse", necesita de 10N vectores adicionales.

En las tablas anteriormente mencionadas, también aparecen los resultados de los 3 métodos de optimización restantes y como se observa, los de Newton Truncado con búsqueda en la línea tienen una realización comparable a los de L-BFGS de Nocedal y Gradientes Conjugados de Shanno, siendo los que hacen menor cantidad de evaluaciones de la función y el gradiente, son de los que menor número de fallos tiene y su tiempo de ejecución es del orden del de los restantes, con excepción de los funcionales de longitud y area ortogonalidad, que tienen la característica de ser los más sencillos.

La comparación entre el funcional propuesto por Ivanienko (ver acápite 2.3) y el de Suavidad Regularizado (2.52) que se propone en este trabajo, sólo es posible hacerla sobre regiones convexas o con muy pocas celdas no convexas, ya que este último así lo requiere. Como la forma de obtener la malla es por un enfoque distinto al que se propone aquí, lo único que se puede enjuiciar es la calidad de la malla y el tiempo en que se obtuvo. Ya que las celdas son todas convexas, la calidad de la malla sólo se puede medir gráficamente, por lo que se remite al lector a los gráficos en el ANEXO II, donde además se reportan los tiempos de ejecución y también redes obtenidas con el resto de los funcionales.

De la misma forma, los resultados en cuanto a si efectivamente los funcionales logran las propiedades esperadas cuando se aplican en una región dada, se reflejan en los gráficos del ANEXO II en los que resaltan los siguientes aspectos:

1.- El funcional de longitud con parámetro τ afloja las líneas horizontales cuando τ 0 (ver gráficos de las regiones ANULUS y CHEVRON).

2.- Con el funcional de área de Barrera y Pérez se obtienen por lo general redes convexas (ver figura BACKSTEP), con celdas de áreas proporcionales.

3.- El funcional de Area Ortogonalidad da redes con celdas proporcionales y segmentos tangentes ortogonales (ver ANULUS, BACKSTEP, CHEVRON).

Todas las pruebas se hicieron en una computadora personal ACERMATE 433s, usando la versión 5.1 de MICROSOFT FORTRAN sobre sistema operativo MS-DOS.

4.6. APLICACION DE LAS MALLAS

Las mallas que se obtienen usando los códigos implementados se están usando actualmente para la explotación del paquete UNAFEM para la solución de problemas en derivadas parciales, mediante el método de los elementos finitos, para lo que se diseñó una interfase que prepara la malla producida, de acuerdo al formato de los datos de entrada dicho paquete de programas.

En el ANEXO II se muestran también varias mallas generadas con el funcional de Area-Ortogonalidad y el de Area de Barrera y Pérez para la región de la Bahía de la Habana.

C O N C L U S I O N E S

El proceso de solución del problema de generación de redes bidimensionales en regiones planas irregulares ha requerido del estudio e investigación de diversos temas de la matemática, obteniendo como resultados y conclusiones del trabajo las siguientes:

1.- Los funcionales variacionales de longitud y suavidad dan la propiedad de suavidad a la red, mientras que en el caso en los de ortogonalidad y área esta propiedad no se tiene, además de que es posible que no se halle una solución en una variedad de regiones, aunque esto último se puede mejorar haciendo uso de un criterio de optimalidad que dé una malla que esté cercana a la propiedad geométrica que se quiere lograr en su aplicación.

2.- La construcción de funcionales discretos a partir de su formulación variacional continua sobre la región lógica, se puede lograr de manera rápida y segura, a través de un procedimiento general que obtiene la formulación variacional discreta de cualquier funcional, que refleje la acumulación de una propiedad sobre la región lógica o de cálculo.

3.- Se obtuvo la regularización novedosa del funcional discreto de Suavidad, lo que permitió su uso de manera segura en los algoritmos de optimización, que resulta muy útil cuando se tiene una red que ya es convexa y se quiere suavizar, o cuando la red inicial tiene muy pocas celdas no convexas.

4.- El funcional de Area-Ortogonalidad da muy buenos resultados en casi todos los casos en cuanto a lograr celdas de áreas proporcionales y líneas con vectores tangentes casi ortogonales.

5.- El funcional de Longitud con parámetro τ permite obtener redes convexas en regiones con determinada geometría, usando el parámetro convenientemente ($\tau \to 0$ afloja las líneas horizontales y tensa las verticales, y lo contrario cuando $\tau \infty$).

6.- El funcional de área de Barrera-Pérez genera redes convexas en la mayoría de los casos, pero que posiblemente necesitan un posterior suavizamiento usando el funcional discreto de Suavidad regularizado.

7.- La implementación computacional que se hizo de dos Métodos de Newton Truncado (con búsqueda en la línea y con región de confianza), resulta eficiente para la solución del problema de aplicación que se quiere resolver, en particular cuando se hacen 10 iteraciones de gradientes conjugados lineal por cada iteración mayor y se toma el parámetro c=1 para la "forcing sequence". Igualmente ocurre con el método que en lugar de usar una evaluación numérica del producto matriz Hessiana-vector, usa explícitamente la Hessiana analítica, siendo esta variante más eficiente en cuanto a tiempo de ejecución y ahorro de evaluaciones de la función y el gradiente, que la correspondiente usando diferencias finitas, como era de esperar, aunque es notable la buena realización del caso que usa diferencias que además tiene menor requerimiento en cuanto a memoria interna.

8.- De los métodos de optimización sin restricciones de gran escala que fueron adaptados al problema en cuestión, luego de la experimentación numérica sobre varias regiones, se tiene que los que mejor se comportan en el problema que se trata son el L-BFGS de Nocedal y el de Newton Truncado con búsqueda en la línea.

9.- Se diseñó una forma original de almacenamiento y localización de los elementos no nulos de la matriz Hessiana, dado que esta es "sparse" y de gran dimensión (ver códigos de los programas que se presentan adjuntos) y la forma de obtener el vector resultante de aplicar dicha matriz a un vector dado.

10.- Se tiene una versión en lenguaje FORTRAN que permite la experimentación con los métodos de optimización estudiados en este trabajo aplicados a la generación de mallas, y la correspondiente experimentción con los de Newton Truncado diseñados y codificados por la autora.

11.- A partir del paquete UNAMALLA (ver [8]) se implementó una versión para usuarios menos experimentados, donde aparte de las facilidades gráficas, de entrada salida, etc, que ya ofrecía el

paquete mencionado, se usa el método de Newton Truncado con Búsqueda en la Línea y matriz Hessiana analítica, con los resultados de la experimentación numérica obtenidos por la autora. Este paquete está codificado en lenguaje C para micro computadoras IBM y compatibles, y se usó para su puesta a punto el compilador VISUAL C++ de la MICROSOFT versión 1.00.

R E C O M E N D A C I O N E S

A partir de la investigación realizada en el tema de generación de redes y de los métodos de optimización para resolverlo, se recomienda continuar trabajando en los siguientes aspectos:

1.- En los métodos de Newton Truncado que usan la matriz Hessiana analítica, sería conveniente probar a hacer la actualización de la matriz cada cierta cantidad de iteraciones, de forma tal que se disminuya el costo de dichas evaluaciones, si es que esto no afecta la velocidad de convergencia del algoritmo.

2.- Analizar la posibilidad de controlar las iteraciones de los métodos de optimización para que una vez que se tenga una red convexa, no se acepten puntos donde esta condición no se satisfaga.

3.- Usar la memoria expandida de las micro computadoras para almacenar los arreglos de gran dimensión que se tienen en el problema de generación de redes.

BIBLIOGRAFIA

[1] Al-Baali M., (1985), *Descent Properties and Global Convergence of the Fletcher-Reeves Method with Inexact Line Search*, IMA J. Numer. Anal., 5, 121-129

[2] Apostol T.M., (1972), *Calculus,* Tomos I y II, Editorial Reverté.

[3] Barrera P., Castillo J.E. (1987), *A Large Scale Optimization Problem Arising from Numerical Grid Generation,* Tech. Report No. 1, Dept. of Math. and Statistics, Univ. of New Mexico.

[4] Barrera P., Castellanos J.L., Ojeda R.B., Pérez A., (1989), *A New Discrete Functional for Grid Generation* , Proceedings of the Fifth Mexico-United States Workshop on Advances in Numerical Partial Differential Equations and Optimization, Mérida, Yucatán, México.

[5] Barrera P., Pérez A., Castellanos J.L., (1992a), *Curvilinear Coordinate System Generation over Plane Irregular Regions*, Vínculos Matemáticos No. 133, Facultad de Ciencias, UNAM, México.

[6] Barrera P., Pérez A., Castellanos J.L., (1992b), *Manual de Usuarios del Paquete MALLA v.1.0*, Vínculos Matemáticos No. 186, Facultad de Ciencias, UNAM, México.

[7] Barrera P., Castellanos J.L., Pérez A.,(1994a), *Métodos Variacionales Discretos para la Generación de Mallas*, Reporte Técnico del Dpto. de Matemáticas de la Facultad de Ciencias de la UNAM.

[8] Barrera P., Pérez A., Castellanos J.L., (1994b), *Manual de Usuarios del Sistema UNAMALLA versión 1.0,* Reporte Técnico del Dpto. de Matemáticas de la Facultad de Ciencias de la UNAM.

[9] Barrera P., Castellanos J.L., (1994c), *Métodos de Optimización de Gran Escala para el Problema de Generación de Redes Optimas*, Reporte Técnico del Dpto. de Matemáticas de la Facultad de Ciencias de la UNAM.

[10] Bartle R., (1976), *The Elements of Real Analysis*, Wiley International Edition, 2nd Edition.

[11] Brackbill J.U., Saltzman J.S., (1982), *Adaptive Zoning for Singular Problems in Two Dimensions*, J.Comp. Physics, 46, 342-368

[12] Broyden C.G., (1967), *Quasi-Newton Methods and their Application to Function Minimization*, Mathematics of Computation 21, 368-381

[13] Castillo J.E. (1987), *Ph. D. Dissertation* , University of New Mexico.

[14] Castillo J. E., Steinberg S., Roache P.J., (1987), *Mathematical Aspects of Variational Grid Generation II*, J. Comp. and Appl. Math. 20, 127-135

[15] ————, (1988), *Parameter Estimation in Variational Grid Generation*, Appl. Math. and Comp., 28, No.2, 1-23

[16] Castillo J.E. ed., (1991), *Mathematical Aspects of Numerical Grid Generation*, SIAM, Phyladelphia.

[17] Castillo J.E., (1991), *A Discrete Variational Grid Generation Method*, SIAM J. Sci. Stat. Comp. 12, No. 2, 454-468

[18] Castillo J.E., (1993), *Grid Generation Methods Consistent with Finite Difference Schemes*, por aparecer.

[19] Conte S.D., De Boor C., (1980), *Elementary Numerical Analysis: An Algorithmic Approach*, 3rd ed., McGraw-Hill,N.Y.

[20] Courant R., (1950), *Dirichlet's Principle, conformal mapping and Minimal Surfaces*, New York, Interscience, 27.

[21] Crowder H.P., Wolfe P., (1972), *Linear Convergence of the Conjugate Gradients Metnod*, IBM J. Res. Develop., 16, 431-433

[22] Dembo R.S., Steihaug T., (1983), *Truncated Newton Algorithm for Large Scale Unconstrained Optimization*, Math. Prog., 26, 190-212

[23] Dennis J.E. Jr., Schnabel R.B. (1983), *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* , Prentice-Hall, Eaglewood Cliffs, N.J.

[24] Dixon L.C.W., (1972a), *Quasi-Newton family generates identical points*, Parts I and II, Math. Prog. 2, 383-387, 2nd Math. Prog. 3, 345-358

[25] Dixon L.C.W., (1972b), *The choice of step length, a crucial factor in the performance of variable metric algorithms*, in Numerical Methods for Non-linear Optimization, F.Lootsma ed., Academic Press, New York, 149-170

[26] Eiseman P.R., (1982), *Orthogonal Grid Generation*, in

Numerical Grid Generation, J.F. Thompson ed., North Holland, New York, 193-226

[27] Fletcher R, Reeves C.M., (1964), *Function Minimization by Conjugate Gradients*, Comput. J. 7, 149-154

[28] Fletcher R. (1988), *Practical Methods of Optimization*, John Wiley and Sons, Second Edition, New York.

[29] Forsythe G.E., Wasow W.R., (1960), *Finite Difference Methods for Partial Differential Equations*, Wiley, New York

[30] Gelfand I.M., Fomin S.V., (1963), *Calculus of Variations*, Prentice Hall, Englewood Cliffs, NJ.

[31] George P.L., (1991), *Automatic Mesh Generation: Applications to Finite Element Methods*, Wiley, New York.

[32] Gilbert J.C., Nocedal J., (1992), *Global Convergence Properties of Conjugate Gradients Methods for Optimization*, SIAM J. in Opt., Vol. 2, No. 1., 21-42.

[33] Gill P.E., Murray W, Wright M., (1981), *Practical Optimization*, Academic Press, London.

[34] Golub G.H., Van Loan C., (1983), *Matrix Computations*, The John Hopkins Univ. Press, Baltimore, Maryland.

[35] Hestenes M.R., Stiefel E., (1952), *Methods of Conjugate Gradients for solving linear systems*, J. Res. Nat. Bur. Standards 49, 409-436

[36] Ivanienko S.A., Charakch'yan A.A., (1987), *An Algorithm for constructing Curvilinear Grids for Convex Quadrilaterals*, Dokl. Acad. Nauk SSSR, 2.

[37] ———, (1988), *Curvilinear Grids of Convex Quadrilaterals*, USSR Comp. Math. and Math. Phys., 28, No. 2, 126-133

[38] Kennon S.R., Dulikravich G.S., (1986), *Generation of Computational Grids using Optimiztion*, AIAA Journal, 24, No. 7, 1069-1073

[39] Knupp P.M., (1990), *On the Invertibility of the Isoparametric Map*, Comp. Meth. Appl. Mech. Eng., 78, 313-329

[40] ———, (1991), *Intrinsec Algebraic Grid Generation*, in Mathematical Aspects of Numerical Grid Generation, J.E. Castillo ed., SIAM, Phyladelphia.

[41] ———, (1992), *A Robust Elliptic Grid Generation*, J. Comp.

Numerical Grid Generation, J.F. Thompson ed., North Holland, New York, 193-226

[27] Fletcher R, Reeves C.M., (1964), *Function Minimization by Conjugate Gradients*, Comput. J. 7, 149-154

[28] Fletcher R. (1988), *Practical Methods of Optimization*, John Wiley and Sons, Second Edition, New York.

[29] Forsythe G.E., Wasow W.R., (1960), *Finite Difference Methods for Partial Differential Equations*, Wiley, New York

[30] Gelfand I.M., Fomin S.V., (1963), *Calculus of Variations*, Prentice Hall, Englewood Cliffs, NJ.

[31] George P.L., (1991), *Automatic Mesh Generation: Applications to Finite Element Methods*, Wiley, New York.

[32] Gilbert J.C., Nocedal J., (1992), *Global Convergence Properties of Conjugate Gradients Methods for Optimization*, SIAM J. in Opt., Vol. 2, No. 1., 21-42.

[33] Gill P.E., Murray W, Wright M., (1981), *Practical Optimization*, Academic Press, London.

[34] Golub G.H., Van Loan C., (1983), *Matrix Computations*, The John Hopkins Univ. Press, Baltimore, Maryland.

[35] Hestenes M.R., Stiefel E., (1952), *Methods of Conjugate Gradients for solving linear systems*, J. Res. Nat. Bur. Standards 49, 409-436

[36] Ivanienko S.A., Charakch'yan A.A., (1987), *An Algorithm for constructing Curvilinear Grids for Convex Quadrilaterals*, Dokl. Acad. Nauk SSSR, 2.

[37] ———, (1988), *Curvilinear Grids of Convex Quadrilaterals*, USSR Comp. Math. and Math. Phys., 28, No. 2, 126-133

[38] Kennon S.R., Dulikravich G.S., (1986), *Generation of Computational Grids using Optimiztion*, AIAA Journal, 24, No. 7, 1069-1073

[39] Knupp P.M., (1990), *On the Invertibility of the Isoparametric Map*, Comp. Meth. Appl. Mech. Eng., 78, 313-329

[40] ———, (1991), *Intrinsec Algebraic Grid Generation*, in Mathematical Aspects of Numerical Grid Generation, J.E. Castillo ed., SIAM, Phyladelphia.

[41] ———, (1992), *A Robust Elliptic Grid Generation*, J. Comp.

Phys., 100, 409-418

[42] Knupp P.M., Steinberg S., (1993), *The Fundamentals of Grid Generation*, por aparecer.

[43] Mastin C.W., Thompson J.F., (1978), *Transformation of Three Dimensional Regions onto Rectangular Region by Elliptic Systems*, Numer. Math., 29, 397-407

[44] Mastin C.W., (1982), *Error induced by Coordinate Systems*, in Numerical Grid Generation, J.F. Thompson ed., North Holland, New York, 31-40

[45] Moré J.J., Thuente D.J., (1990), *On line search algorithms with guaranteed sufficient decrease*, Math. and Computer Science Division Prepint MCS-P153-0590, Argonne National Lab., Argonne, Il.

[46] Nazareth L., (1979), *A Relationship between the BFGS and Conjugate Gradients Algorithms and its implications for new algorithms*, SIAM J. Numer. Anal., 16, 294-300

[47] Nocedal J., (1980), *Updating Quasi-Newton matrices with Limited Storage*, Math. of Comp. 35, 773-782

[48] Nocedal J., Liu D.C. (1989), *On the Limited Memory BFGS Method for Large Scale Optimization*, Math. Programming, Vol. 45, No. 3, 503-528.

[49] Ojeda R.B., (1991), *Métodos Directos para la Generación de Redes en Regiones Planas*, Tesis de Maestría, UNAM, México.

[50] Perry A, (1977), *A Class of Conjugate Gradients algorithn with a two-step Variable Metric Memory*, Discussion paper 269, Center for Math. Studies in Economics and Management Science, Northwestern University.

[51] Polak E., Ribière G., (1969), *Note sur la convergence de méthodes de directions conjuguées*, Française d'Informatique et de Recherche Opérationalle, 16, 35-43

[52] Powell M.J.D., (1976), *Some convergence properties of the Conjugate Gradients Method*, Math. Prog. 11, 42-49

[53] ———, (1976), *Some global convergence properties of a variable metric algorithm without line searches*, in SIAM-AMS Proceedings, Vol.IX, Math. Prog., 53-72

[54] ———, (1977), *Restart procedures for the Conjugate*

Gradient Methods, Math. Prog. 12, 241-254

[55] ———, (1984), *Nonconvex minimization calculations and the Conjugate Gradients Method*, Lecture Notes in Math, 1066, Springer-Verlag, Berlin, 122-141

[56] Saltzman J.S., Brackbill J.U., (1982), *Applications and Generalizations of Variational Methods for Generating Adaptive Meshes*, in Numerical Grid Generation, J.F. Thompson ed., North Holland, New York, 865-884.

[57] Shanno D.F., (1978), *Conjugate Gradients Methods with Inexact Searches*, Math. Ope. Res., 3, 244-256

[58] Shanno D.F., Phua K.H. (1978), *A Variable Method Subroutine for Unconstrained Nonlinear Minimization*, M.I.S. Technical Report No. 28, Univ. of Arizona, Tucson.

[59] Steihaug T. (1983) , *The Conjugate Gradient Method and Trust Region in Large Scale Optimization*, SIAM Journal of Numerical Analysis, Vol. 20, 626-637.

[60] Steinberg S., Roache P.J., (1986), *Variational Grid Generation*, Num. Meth. for PDE, 2, 71-96

[61] ———, (1992), Variational Curve and Surface Grid Generation, J. Comp. Phys. 100, 163-178

[62] Thompson J.F. (Ed.) (1982), *Numerical Grid Generation*, North Holland.

[63] Thompson J.F., Warsi Z.U.A., Mastin C.W., (1985), *Numerical Grid Generation: Foundations and Applications*, North Holland, Elsevier, New York

[64] Touati-Ahmed D., Storey C., (1990), *Efficient Hybrid Conjugate Gradients Techniques*, J. Optim. Theory Appl. 64, 379-397

[65] Winslow A., (1967), *Numerical Solution of the Quasilinear Poisson Equations in a Nonuniform Triangle Mesh*, J. Comp. Physics, 2, 149-172

[66] Wolfe P., (1969), *Convergence Conditions for Ascent Methods*, SIAM Rev. 11, 226-235

[67] ———, (1971), *Convergence Conditions for Ascent Methods II: Some corrections*, SIAM Rev. 13, 185-188

[68] Zou X, Navan I.M., Berger M., Phua K.H., Schilck T., Le

Dimit F.X., (1993), *Numerical Experience with Limited Memory Quasi Newton and Truncated Newton Methods*, SIAM J. on Opt., Vol. 3, No. 3, 582-608

[69] Zoutendijk G., (1970), *Nonlinear Programming, Computational Methods*, in Integer and Nonlinear Programming, J. Abadie ed., North Holland, Amsterdam, 37-86.