

Master Universitario en Computación Paralela y Distribuida
Algoritmos Paralelos Matriciales en Ingeniería

Tema 7

Resolución de sistemas de ecuaciones no lineales

The image is a presentation slide designed to look like a spiral-bound notebook. On the left side, there is a dark brown vertical strip representing the notebook's cover, with a silver-colored metal spiral binding. To the right of the cover is a large, light beige rectangular area representing the notebook's pages. This area has a subtle, repeating pattern of faint, overlapping squares in shades of tan and cream, giving it a textured, paper-like appearance. The text is centered on this page area.

APMI: problemas no lineales.

Introducción

Libros básicos

Iterative Methods for Linear and Nonlinear Equations

C.T. Kelley
North Carolina State University

Ed. SIAM 1995

Numerical Methods for Nonlinear Equations
and Unconstrained Optimization
J.E.Dennis & R.B.Schnabel
Prentice-Hall, Englewood Cliffs, NJ, 1983

Sistemas de ecuaciones lineales: Recordatorio(1).

$$3x_1 + 4x_2 + 1.1x_3 = 4$$

$$2x_1 + 7x_2 - 3x_3 = -1$$

$$4x_1 + 4x_2 + 9x_3 = 2$$

Se puede replantear como:

$$f_1(x_1, x_2, x_3) = 0$$

$$f_2(x_1, x_2, x_3) = 0$$

$$f_3(x_1, x_2, x_3) = 0$$

Donde f_1 , f_2 , f_3 son funciones lineales de sus argumentos.

Sistemas de ecuaciones lineales: Recordatorio(2).

Habitualmente se escriben en forma matricial:

$$\mathbf{Ax} = \mathbf{b}, \mathbf{A} \in \mathbb{R}^{n \times n}, \mathbf{b} \in \mathbb{R}^n$$

Para que exista solución única, la matriz debe ser cuadrada (igual número de incógnitas y de ecuaciones) y la matriz debe ser no singular (debe tener inversa).

Sistemas de ecuaciones lineales: Recordatorio(3).

Métodos de resolución de ecuaciones lineales:

- Métodos directos: Gauss, factorización LU, QR, etc.
- Métodos Iterativos:
 - A) Métodos Estacionarios (Jacobi, SOR, Gauss-Seidel)
 - B) Métodos No Estacionarios (GMRES, CG,...)

Sistemas de ecuaciones lineales: Recordatorio(4).

Métodos Iterativos Estacionarios de resolución de ecuaciones lineales:

Dado el sistema:

$$\mathbf{Ax} = \mathbf{b}, \mathbf{A} \in \mathbb{R}^{n \times n}, \mathbf{b} \in \mathbb{R}^n$$

Lo reformulamos o reordenamos, para obtener otro de la forma

$$\mathbf{x} = \mathbf{Mx} + \mathbf{v}, \mathbf{M} \in \mathbb{R}^{n \times n}, \mathbf{v} \in \mathbb{R}^n$$

Sistemas de ecuaciones lineales: Recordatorio(5).

$$3x_1 + 4x_2 + 1.1x_3 = 4$$

$$2x_1 + 7x_2 - 3x_3 = -1$$

$$4x_1 + 4x_2 + 9x_3 = 2$$

Se transforma en:

$$x_1 = (4 - 4x_2 - 1.1x_3) / 3$$

$$x_2 = (-1 - 2x_1 + 3x_3) / 7$$

$$x_3 = (2 - 4x_1 - 4x_2) / 9$$

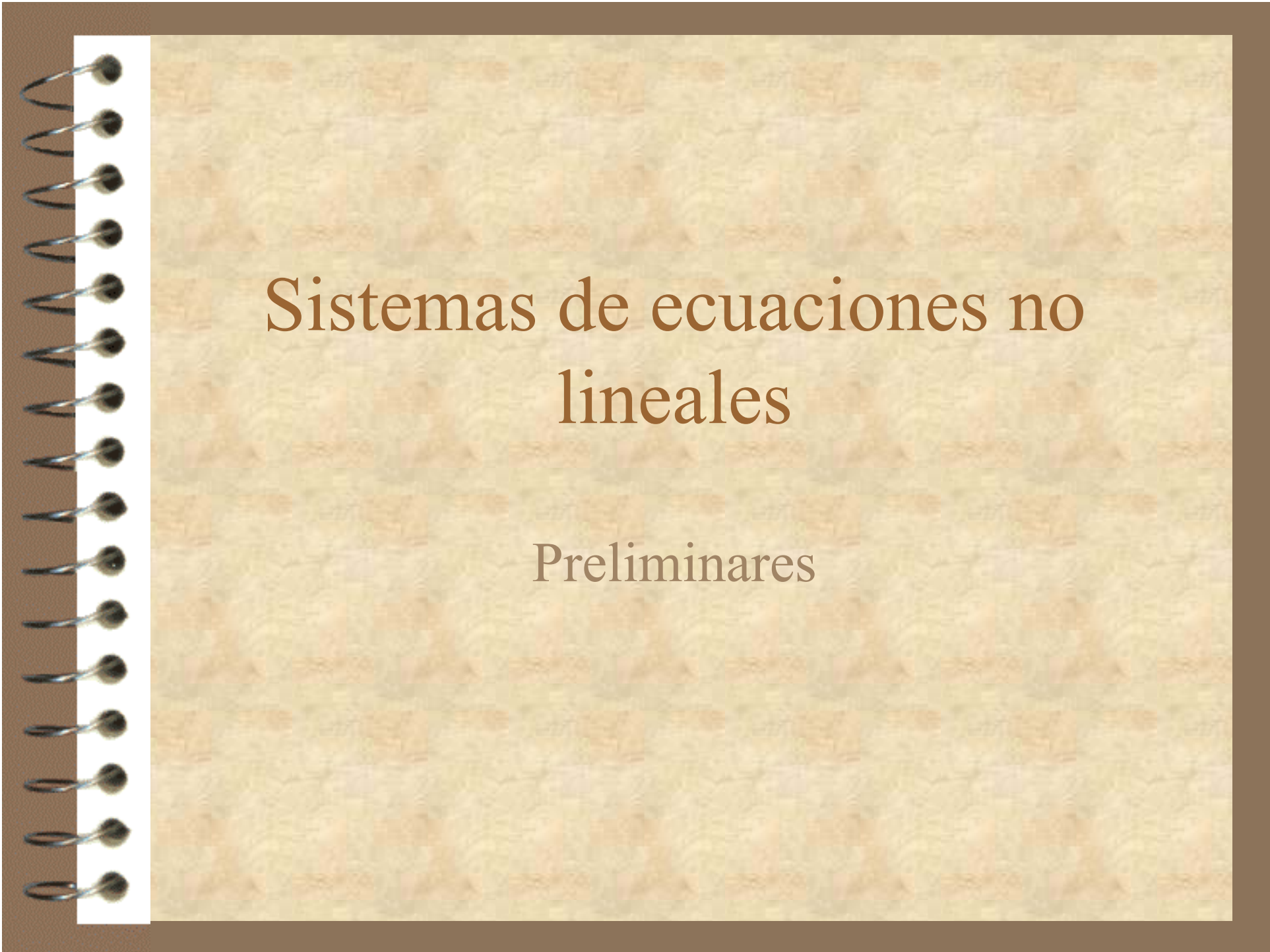
Sistemas de ecuaciones lineales: Recordatorio(6).

Dado un vector inicial x^0 y la fórmula anterior, podemos generar una sucesión de vectores que (quizás) convergen a la solución del sistema:

$$x_1^{n+1} = (4 - 4x_2^n - 1.1x_3^n) / 3$$

$$x_2^{n+1} = (-1 - 2x_1^n + 3x_3^n) / 7$$

$$x_3^{n+1} = (2 - 4x_1^n - 4x_2^n) / 9$$

The image shows a spiral-bound notebook with a brown cover. The left side of the notebook is visible, showing the silver spiral binding. The main part of the notebook is a light beige or cream-colored page with a subtle, repeating pattern of small, faint, stylized figures or shapes. The text is centered on this page.

Sistemas de ecuaciones no lineales

Preliminares

Sistemas de ecuaciones no lineales: Preliminares

Tenemos que resolver el problema:

$$\left. \begin{array}{l} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) = 0 \end{array} \right\} \Rightarrow F(X) = 0, X \in \mathbb{R}^n, F : \mathbb{R}^n \rightarrow \mathbb{R}^n$$

Donde las componentes de la función $F : f_1, \dots, f_n$ son funciones no lineales.

Sistemas de ecuaciones no lineales: Preliminares

Ejemplo 1:

$$\begin{cases} x_1^2 + x_2^2 - 4 = 0 \\ x_1^2 - x_2^2 + 1 = 0 \end{cases}$$

Ejemplo 2:

$$\begin{cases} x_1 x_2 + \cos(x_3) = 0 \\ \text{sen}(x_1 - x_2) + x_3 = 0 \\ x_1 x_3 + 2x_1^2 = 0 \end{cases}$$

Sistemas de ecuaciones no lineales: Preliminares

CONSIDERACIONES GENERALES:

- 1) El número de ecuaciones y de incógnitas ha de ser el mismo.
 - 2) Los métodos para este problema son siempre iterativos. No hay métodos directos para sistemas no lineales.
 - 3) Puede no haber solución: $\cos(x)-3=0$.
 - 4) Es muy posible que no podamos hallar la solución.
- Numerical recipes in Fortran: *“There are NO good, general methods for solving systems of more than one nonlinear equation”*

Sistemas de ecuaciones no lineales: Preliminares

Métodos Iterativos:

Necesitamos un **vector inicial** X^0 a partir del cual empezar a iterar

Necesitamos la **función de iteración**, g que dado el vector de la iteración n nos da el de la $n+1$:

$$g(X^n) = X^{n+1}$$

$$X^0, X^1(=g(X^0)), X^2(=g(X^1)), \dots, X^n(=g(X^{n-1})), X^{n+1}, \dots$$

¿Cuándo parar?

Sistemas de ecuaciones no lineales: Preliminares

Utilizamos normas vectoriales:

- Norma euclidea.
- Norma 1.
- Norma Infinito.

Dada una cierta tolerancia **tol**, pararemos de generar miembros de la sucesión cuando:

- | | |
|---|------------------|
| A) $\ X^{n+1} - X^n\ < \mathbf{tol}$ | ERROR ABSOLUTO |
| B) $(\ X^{n+1} - X^n\ / \ X^{n+1}\) < \mathbf{tol}$ | ERROR RELATIVO |
| C) $\ F(X^{n+1})\ < \mathbf{tol}$ | RESIDUO ABSOLUTO |
| D) $\ F(X^{n+1})\ < \mathbf{tol} \cdot \ F(X^0)\ $ | RESIDUO RELATIVO |

Se suelen combinar varios criterios de parada

Sistemas de ecuaciones no lineales: Preliminares

Velocidad de convergencia

Si la sucesión de vectores converge a X^* , tenemos:

- 1) Convergencia lineal: $\|X^{n+1}-X^*\| \leq q\|X^n-X^*\|$, $q \in (0,1)$
- 2) Convergencia superlineal: $\|X^{n+1}-X^*\| \leq K\|X^n-X^*\|^\alpha$, $K>0$, $\alpha>1$
- 3) Convergencia cuadrática: $\|X^{n+1}-X^*\| \leq K\|X^n-X^*\|^2$, $K>0$

Sistemas de ecuaciones no lineales: Preliminares

JACOBIANO DE LA FUNCIÓN VECTORIAL DE VARIAS VARIABLES $F(X)$

Dada $F(X)$ \longrightarrow
$$\begin{cases} f_1(x_1, x_2, \dots, x_n) \\ f_2(x_1, x_2, \dots, x_n) \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) \end{cases}$$

Definimos el jacobiano de $F(X)$, $J(X)$ como la matriz $\frac{\partial F_i}{\partial X_j}$

Sistemas de ecuaciones no lineales: Preliminares

JACOBIANO DE LA FUNCIÓN VECTORIAL DE VARIAS VARIABLES $F(X)$

$$\frac{\partial F_i}{\partial X_j}(X_1, \dots, X_n) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}_{(x_1, \dots, x_n)}$$

Sistemas de ecuaciones no lineales: Preliminares

JACOBIANO DE LA FUNCIÓN VECTORIAL DE VARIAS VARIABLES $F(X)$

$$F(X) = F(x_1, x_2) = \begin{cases} f_1(x_1, x_2) = x_1^2 + x_2^2 - 4 \\ f_2(x_1, x_2) = x_1^2 - x_2^2 + 1 \end{cases};$$

$$F'(X) = F'(x_1, x_2) = \begin{pmatrix} 2x_1 & 2x_2 \\ 2x_1 & -2x_2 \end{pmatrix};$$

$$F'(1, 1) = \begin{pmatrix} 2 & 2 \\ 2 & -2 \end{pmatrix}$$

The background of the slide is a spiral-bound notebook. The left side shows the silver metal spiral binding on a dark brown cover. The right side is a light beige, textured paper surface. The title text is centered on the paper.

Métodos para Sistemas de Ecuaciones No Lineales

Iteración funcional o de Picard

Métodos para S. Ec. No Lin.

Iteración funcional o de Picard:

Dado el sistema de ecuaciones no lineales

$$F(X) = 0, X \in \mathbb{R}^n, F : \mathbb{R}^n \rightarrow \mathbb{R}^n$$

Si se puede reordenar en la forma

$$X = K(X)$$

Podemos tomar K como nuestra función de iteración, y utilizarla para generar la sucesión:

$$X^{n+1} = K(X^n)$$

Métodos para S. Ec. No Lin.

Iteración funcional o de Picard:

Dado el ejemplo 1

$$\begin{cases} x_1^2 + x_2^2 - 4 = 0 \\ x_1^2 - x_2^2 + 1 = 0 \end{cases}$$

Reordenamos como:

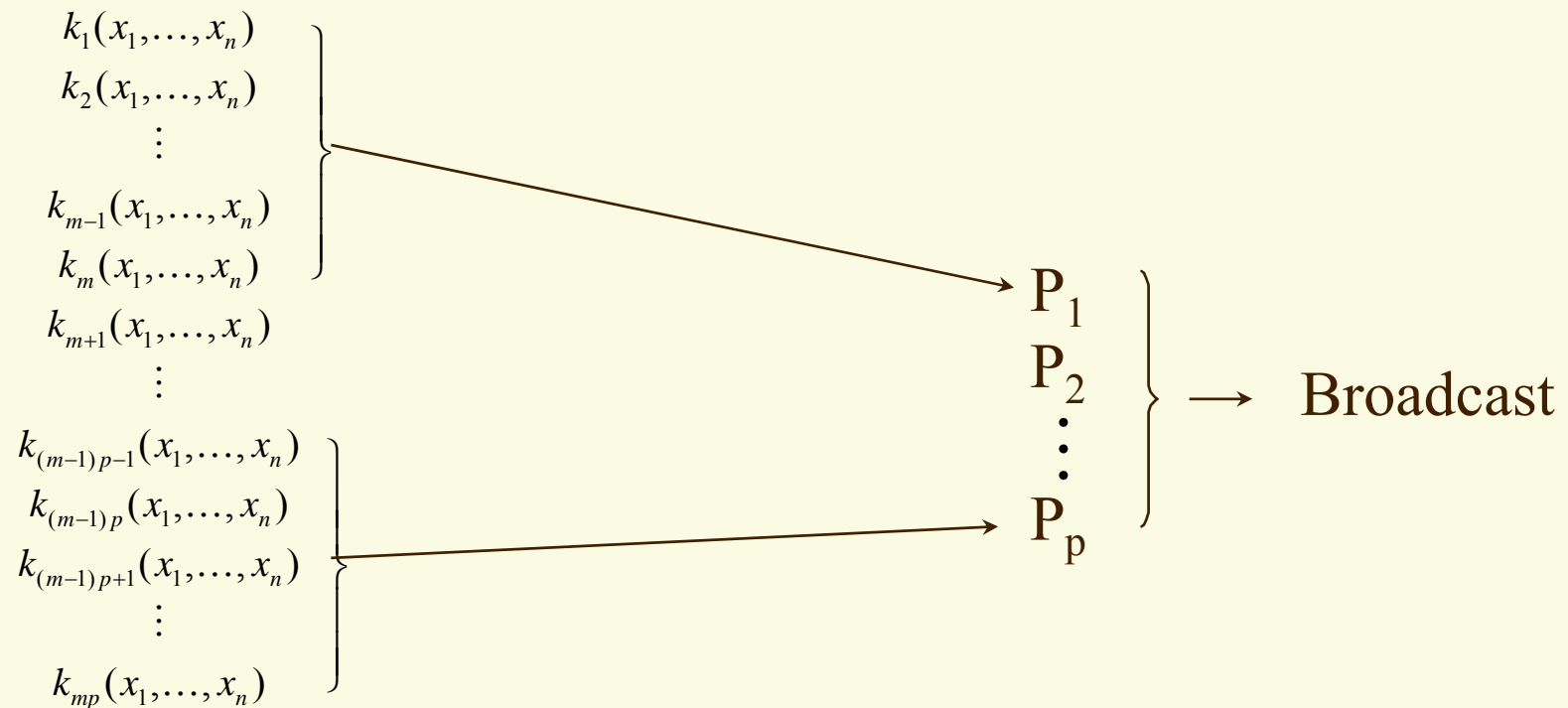
$$\begin{cases} x_1 = \sqrt{-x_2^2 + 4} \\ x_2 = \sqrt{x_1^2 + 1} \end{cases}$$

La iteración de Picard es, en general, muy poco robusta.

Métodos para S. Ec. No Lin.

Iteración funcional o de Picard:

Implementación paralela: $n=mp$



The background of the slide is a spiral-bound notebook. The left side shows the silver metal spiral binding and a white strip of paper. The main area is a light brown, textured surface resembling aged paper or parchment, framed by a dark brown border.

Métodos para Sistemas de Ecuaciones No Lineales

Método de Newton

Métodos para S. Ec. No Lin.

Método de Newton (1)

Método de Newton en una variable para resolver $f(x)=0$:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)}$$

Con esta fórmula y un punto inicial x^0 , generamos una sucesión que (a menudo pero no siempre) converge a la solución.

Método de Newton en n variables para resolver $F(X)=0$:

$$X^{k+1} = X^k - F'(X^k)^{-1} F(X^k)$$

Generamos una sucesión de vectores.

Métodos para S. Ec. No Lin.

Método de Newton (2)

$$X^{k+1} = X^k - F'(X^k)^{-1} F(X^k)$$

$F'(X^k)$ es una matriz (jacobiano)



Para calcular $F'(X^k)^{-1}$ hay que invertir la matriz $F'(X^k)$



No se necesita calcular la inversa, sino $F'(X^k)^{-1} F(X^k)$



Resolver el sistema de ecuaciones lineales $F'(X^k) \cdot \Delta X = -F(X^k)$

Obtener $X^{k+1} = X^k + \Delta X$

Métodos para S. Ec. No Lin.

Método de Newton (3)

Consideraciones generales:

- A) El método de Newton puede converger o no, dependiendo de varias circunstancias (vector inicial); si converge, lo hace **cuadráticamente**.
- B) Es necesario construir el jacobiano y, en cada iteración, resolver un sistema de ecuaciones lineales (factorización LU o otros métodos).

Métodos para S. Ec. No Lin.

Método de Newton (4)

Implementación con LU (Kelley)

newtonLU(x, F, tolR, tolA)

1. $R0 = \|F(x)\|$
2. Do While $\|F(x)\| > tolR * R0 + tolA$
 - 2.1 Calcular Jacobiano $F'(x)$
 - 2.2 Descomponer Jacobiano $F'(x) = L \cdot U$
 - 2.3 Resolver i) $Ly = -F(x)$ ii) $Us = y$;
 - 2.4 $x = x + s$
 - 2.5 evaluar $F(x)$

Métodos para S. Ec. No Lin.

Método de Newton (5)

Costes

A) Descomponer Jacobiano $F'(x)=L \cdot U$;
Coste $2n^3/3$ flops por iteración

B) Calcular Jacobiano:

B.1 Analítico (programación manual)

B.2 Diferencias finitas (más usado)

B.3 Diferenciación automática.

Métodos para S. Ec. No Lin.

Jacobiano por derivadas finitas

Definición de derivada en una variable:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Derivada parcial con respecto a la k-ésima variable:

$$\frac{\partial f(x_1, x_2, \dots, x_k, \dots, x_n)}{\partial x_k} = \lim_{h \rightarrow 0} \frac{f(x_1, x_2, \dots, x_k + h, \dots, x_n) - f(x_1, x_2, \dots, x_k, \dots, x_n)}{h}$$

Tomando h pequeño, podemos aproximar la derivada parcial como:

$$\frac{\partial f(x_1, x_2, \dots, x_k, \dots, x_n)}{\partial x_k} \approx \frac{f(x_1, x_2, \dots, x_k + h, \dots, x_n) - f(x_1, x_2, \dots, x_k, \dots, x_n)}{h}$$

Métodos para S. Ec. No Lin.

Jacobiano por diferencias finitas

Dado un punto (x_1, x_2, \dots, x_n) y un valor h apropiado (si la máquina tiene precisión ε , se suele tomar $\sqrt{\varepsilon}$)

Evaluar $F_p = F(x_1, x_2, \dots, x_n)$

Para $j=1:n$

Evaluar $F_j = F(x_1, x_2, \dots, x_j+h, \dots, x_n)$

Para $i=1:n$

$J(i,j) = (F_j(i) - F_p(i))/h$

Fin Para

Fin Para

Coste: $N+1$ evaluaciones de la función

Problema de error de redondeo: sustracciones de cantidades muy parecidas. (h debe ser un “número de máquina”).

The background of the slide is a spiral-bound notebook. The left side shows the silver metal spiral binding on a dark brown cover. The right side is a light beige, textured paper surface. The title is centered on the paper in a dark brown serif font.

Métodos para Sistemas de Ecuaciones No Lineales

Modificaciones del método de
Newton

Métodos para S. Ec. No Lin.

Modif. Newton (1)

Método de la cuerda (Chord method)

ChordLU(x, F, tol)

1. $R0 = \|F(x)\|$
2. Calcular Jacobiano $F'(x)$
3. Descomponer Jacobiano $F'(x) = L \cdot U$
4. Do While $\|F(x)\| > tol_r * R0 + tol_A$
 - 2.1 Resolver i) $Ly = -F(x)$ ii) $Us = y$;
 - 2.2 $x = x + s$
 - 2.3 evaluar $F(x)$

Métodos para S. Ec. No Lin.

Modif. Newton (2)

Método de la cuerda (Chord method): Comparación

- A) Coste por iteración mucho menor
- B) Velocidad de convergencia lineal

Métodos para S. Ec. No Lin.

Modif. Newton (3)

Método de Shamanskii

Newton: Cálculo y factorización del jacobiano en cada iteración.

Chord: Cálculo y factorización del jacobiano sólo en la primera iteración.

Shamanskii: Cálculo y factorización del jacobiano cada **m** iteraciones, $1 < m < \infty$.

Opción: Calcular y factorizar el jacobiano cuando la tasa de convergencia disminuya por debajo de un cierto umbral.

Métodos para S. Ec. No Lin.

Modif. Newton (4)

Método de Shamanskii

Newton_sham(x, F, tol)

1. $R0 = \|F(x)\|$
2. Do While $\|F(x)\| > \text{tolr} * R0 + \text{tolA}$
 - 2.1 Calcular Jacobiano $F'(x)$
 - 2.2 Descomponer Jacobiano $F'(x) = L \cdot U$
 - 2.3 para $i=1,..m$
 - 2.3.1 Resolver i) $Ly = -F(x)$ ii) $Us = y$;
 - 2.3.2 $x = x + s$
 - 2.3.3 evaluar $F(x)$

Orden de convergencia superlineal (dependiendo de m)

Métodos para S. Ec. No Lin.

Comparación de resultados

88

ITERATIVE METHODS FOR LINEAR AND NONLINEAR EQUATIONS

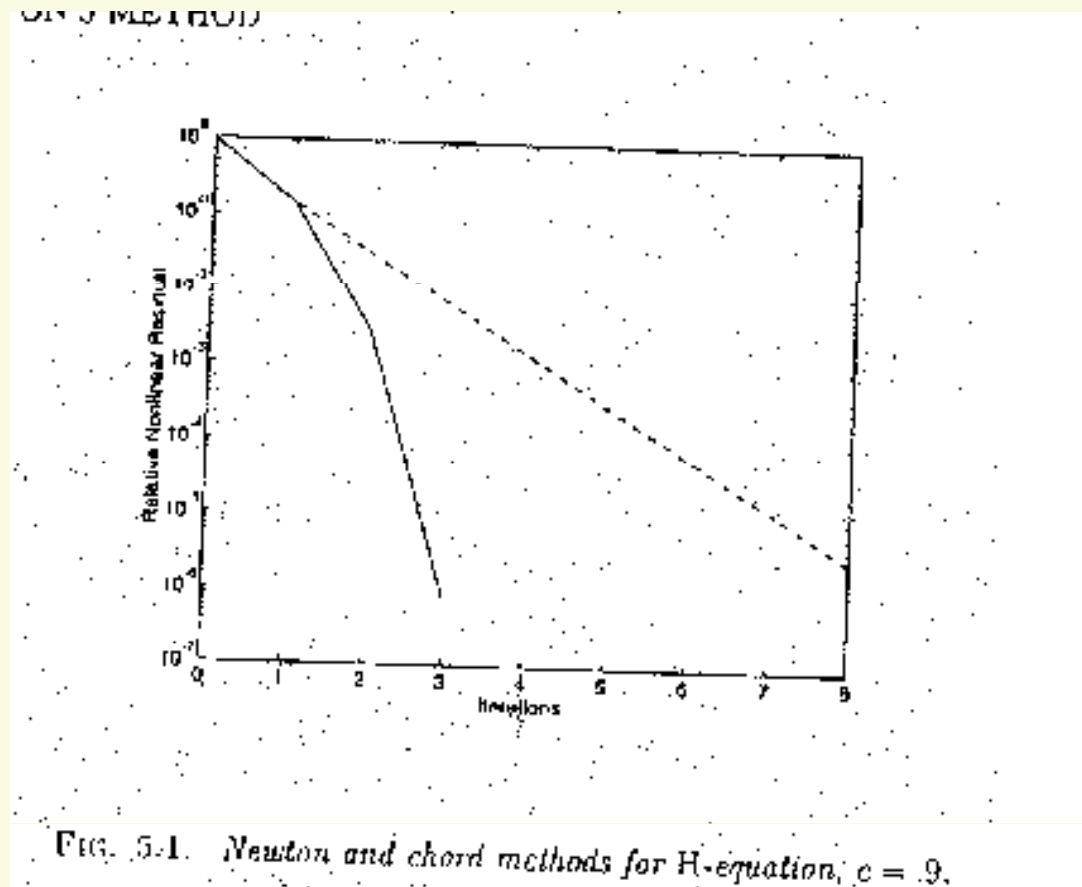
TABLE 5.1

Comparison of Newton and chord iteration.

n	$\ F(x_n)\ _\infty / \ F(x_0)\ _\infty$	R_n	$\ F(x_n)\ _\infty / \ F(x_0)\ _\infty$	R_n
0	1.000e+00		1.000e+00	
1	1.480e-01	1.480e-01	1.480e-01	1.480e-01
2	2.698e-03	1.823e-02	3.074e-02	2.077e-01
3	7.729e-07	2.865e-04	6.511e-03	2.118e-01
4			1.388e-03	2.132e-01
5			2.965e-04	2.136e-01
6			6.334e-05	2.136e-01
7			1.353e-05	2.136e-01
8			2.891e-06	2.136e-01

Métodos para S. Ec. No Lin.

Comparación de resultados



Métodos para S. Ec. No Lin.

Comparación de resultados

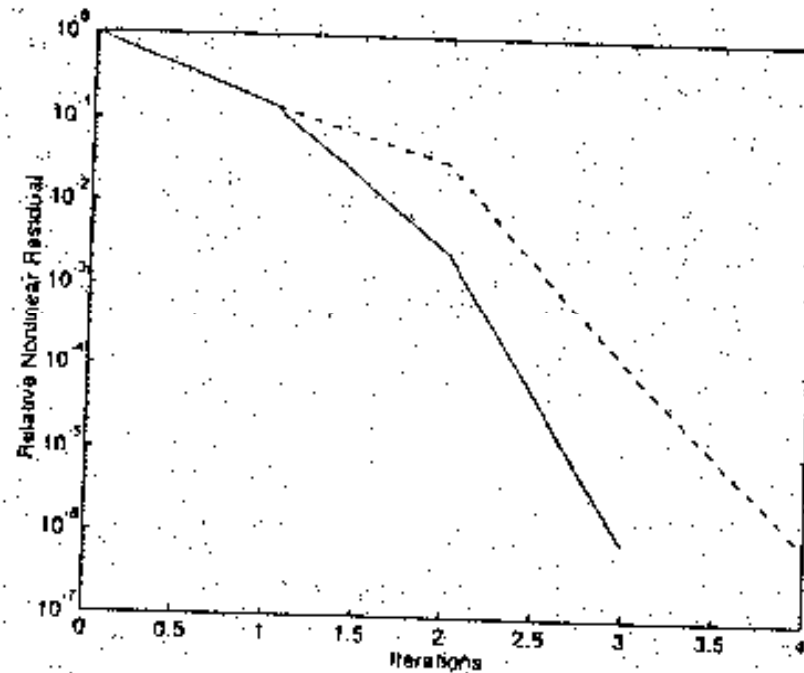


FIG. 5.2. Newton and Shamanskii method for H-equation, $c = .9$.

Métodos para S. Ec. No Lin.

Paralelización de métodos

Factorización del jacobiano

- Uso de LU paralelizada
- Uso de QR paralelizada
- Matrices estructuradas (bloques)
- En general, LU paralela se implementa usando paralelismo de nivel fino; recomendable en ordenadores con memoria compartida, no en redes.

Métodos para S. Ec. No Lin.

Métodos Inexactos de Newton.(1)

Resolución del sistema de ecuaciones lineales mediante un método iterativo.

Métodos válidos: Jacobi, Gauss-Seidel, SOR, GMRES, CG,
...

Diferencias importantes:

- Paralelización y optimización para jacobianos dispersos.
- Precondicionado
- Uso apropiado de las tolerancias.

Métodos para S. Ec. No Lin.

Métodos Inexactos de Newton(2).

Newton_jacobi(x, F, tol)

1. $R0 = \|F(x)\|$
2. Do While $\|F(x)\| > \text{tolr} * R0 + \text{tolA}$
 - 2.1 Calcular Jacobiano $F'(x)$
 - 2.2 $s = \text{Jacobi}(F', F(x), x0, 100, \text{tol_lin})$
 - 2.4 $x = x + s$
 - 2.5 evaluar $F(x)$

Métodos para S. Ec. No Lin.

Métodos Inexactos de Newton(3).

Costes

A) Los métodos iterativos estacionarios Jacobi, G-S, SOR tienen un coste de $2n^2$ flops por iteración. (si la matriz es “llena”). Si la matriz (jacobiano en este caso) es dispersa, se puede optimizar mucho más.

B) la tolerancia con que se debe resolver el sistema de ecuaciones lineales no tiene porque ser muy grande, ahorrando así iteraciones.

Métodos para S. Ec. No Lin.

Métodos Inexactos de Newton(4).

C) Los métodos no estacionarios (GMRES, CG, Bicgstab, ...) tienen en general mejor rendimiento que los estacionarios

D) Para lograr la convergencia en un número razonable de iteraciones, suele ser necesario usar preconditionadores.

E) La operación clave en métodos iterativos es el producto Matriz por Vector; también hay productos escalares.

Métodos para S. Ec. No Lin.

“Convergencia global”(1)

Inconveniente del Método de Newton:
radio de convergencia relativamente pequeño.

Ej. $f(x)=\arctan(x)$, $x_0=10$.

Longitud del paso de Newton, $-f(x_{n-1})/f'(x_{n-1})$ apunta en la dirección correcta pero es demasiado largo.

Solución: limitar la longitud del paso

Métodos para S. Ec. No Lin.

“Convergencia global”(2)

line-search o backtracking

Método de Newton (una variable)

- 1 $r0 = |f(x)|$
- 2 do while $|f(x)| > tol_r * r0 + tol_A$
 - 2.a if $f'(x) = 0$ error
 - 2.b $s = -f(x)/f'(x)$
 - 2.c $x_t = x + s$
 - 2.d if $|f(x_t)| < |f(x)|$ then $x = x_t$
else
 - 2.e $s = s/2$; goto 2(c)

The background of the slide is a spiral-bound notebook. The left side shows the silver metal spiral binding on a dark brown cover. The right side is a light beige, textured paper surface. The title text is centered on the paper.

Métodos para Sistemas de Ecuaciones No Lineales

Método de Broyden
(Métodos quasi-Newton)

Métodos para S. Ec. No Lin.

Método de la secante

Método de la secante para raíces en 1 dimensión

Aproximamos la derivada $f'(x)$ como:

$$\frac{f(x^n) - f(x^{n-1})}{x^n - x^{n-1}}$$

$$\Rightarrow x^{n+1} = x^n - \frac{f(x^n)}{\frac{f(x^n) - f(x^{n-1})}{x^n - x^{n-1}}}$$

Métodos para S. Ec. No Lin.

Método de Broyden(1).

Iteración de Newton:

$$X^{k+1} = X^k - F'(X^k)^{-1} F(X^k)$$

Métodos quasi-Newton:

$$X^{k+1} = X^k - B_k^{-1} F(X^k)$$

Donde B_k es una aproximación al jacobiano

No es necesario calcular el jacobiano, ni factorizarlo:
Para obtener las aproximaciones, utilizamos secante
Multidimensional.

Métodos para S. Ec. No Lin.

Método de Broyden(2).

Ecuación multidimensional de la secante

$$B(X^{n+1} - X^n) = F(X^{n+1}) - F(X^n)$$

Dada B_n y X_n , y la función $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$

BROYDEN “UPDATE”

$$B_{k+1} = B_k + \frac{F(X_{k+1})s^T}{s^T s}$$

Donde $s = X_{k+1} - X_k$

Métodos para S. Ec. No Lin.

Método de Broyden(3).

Broyden(x, F, B₀ , tol)

1. R0= $\|F(x)\|$

2. Do While $\|F(x)\| > \text{tolr} * R0 + \text{tolA}$

2.1 $x_n = x - B^{-1}F(x)$

2.2 $B = B + (F(x_n)(x_n - x)^T / (x_n - x)^T (x_n - x))$

2.4 $x = x_n$

Métodos para S. Ec. No Lin.

Método de Broyden(4).

Al mismo tiempo se calcula la aproximación al punto y del jacobiano .

Elección inicial de B_0

1. Tomar la identidad.
2. Gastar $N+1$ evaluaciones en obtener un jacobiano por diferencias finitas

Métodos para S. Ec. No Lin.

Método de Broyden(5).

Actualización o “update” de Broyden .

- En cada iteración, hay que resolver el sistema de ecuaciones lineales $B_k^{-1}F(X_k)$
- Si el coste de formular y/o resolver este sistema es alto, no merecerá la pena en comparación con Newton.

$$B_{k+1} = B_k + \frac{F(X_{k+1})s^T}{s^T s}$$

Veamos técnicas para aprovechar el update de Broyden

Métodos para S. Ec. No Lin.

Método de Broyden(6).

Si tomamos

$$u_k = \frac{F(X_{k+1})}{\|s_k\|}, \quad v_k = \frac{s_k}{\|s_k\|}$$

La fórmula de Broyden se convierte en:

$$B_{k+1} = B_k + u_k v_k^T$$

UPDATE DE RANGO 1

Métodos para S. Ec. No Lin.

Método de Broyden(7).

Dados los vectores

$$u^T = (u_1, \dots, u_n); \quad v^T = (v_1, \dots, v_n)$$

se tiene que :

$$uv^T = \begin{pmatrix} u_1 v_1 & u_1 v_2 & \cdots & u_1 v_n \\ u_2 v_1 & u_2 v_2 & \cdots & u_2 v_n \\ \vdots & \vdots & \ddots & \vdots \\ u_n v_1 & u_n v_2 & \cdots & u_n v_n \end{pmatrix}$$

Matriz de rango 1

Métodos para S. Ec. No Lin.

Método de Broyden(8).

Si el segundo vector es e_j

$$u^T = (u_1, \dots, u_n); \quad v^T = (0, 0, \dots, \overset{j}{1}, 0, \dots, 0)$$

se tiene que :

$$uv^T = \begin{pmatrix} 0 & \dots & u_1 & \dots & 0 \\ 0 & \dots & u_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & & u_{n-1} & & 0 \\ 0 & \dots & u_n & \dots & 0 \end{pmatrix}$$

$B + uv^T$ es sumarle a la j -ésima columna de B el vector u

Métodos para S. Ec. No Lin.

Método de Broyden(9).

Si el primer vector es e_j

$$u^T = (0, 0, \dots, 1, 0, \dots, 0); \quad v^T = (v_1, \dots, v_n)$$

se tiene que :

$$uv^T = \begin{pmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & \dots & 0 \\ v_1 & v_2 & \dots & v_{n-1} & v_n \\ \vdots & \vdots & & & 0 \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

$B + uv^T$ es sumarle a la j -ésima fila de B el vector v

Métodos para S. Ec. No Lin.

Método de Broyden(9).

Si sabemos resolver rápidamente sistemas con matriz de coeficientes B_k , queremos resolver rápidamente sistemas con matriz de coeficientes $B_{k+1} = B_k + uv^T$

Técnica 1: Una vez calculada una descomposición de B_k , intentar actualizarla con coste “pequeño” ($O(n^2)$).

Técnica 2: Si conocemos la inversa de B_k , obtener a partir de ella la inversa de B_{k+1} con coste $O(n^2)$.

Métodos para S. Ec. No Lin.

Updating de la desc. QR.

Planteamiento del problema

- Sea B matriz con m filas y n columnas cuya descomposición QR es conocida: $B=QR$.
- Dada la matriz $A=B+uv^t$, obtener Q_1, R_1 de forma que $A= Q_1 \cdot R_1$ con coste $O(n^2)$

$$A= QR+uv^t = Q(R+uv^t), \text{ donde } w=Q^tu$$

Métodos para S. Ec. No Lin.

Updating de la desc. QR.

Paso 1

- Obtener H ortogonal de forma que $Hw = \alpha e_1$
- Se obtiene como una sucesión de rotaciones de Givens, anulando en primer lugar el elemento m de w , después el $m-1$, etc.

$$H = G_1 \cdot G_2 \cdot \dots \cdot G_{m-1}$$

H transforma la matriz R en Hessenberg superior.

$$\alpha e_1 v^t = \begin{pmatrix} \alpha v_1 & \alpha v_2 & \dots & \alpha v_n \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

Métodos para S. Ec. No Lin.

Updating de la desc. QR.

$HR + \alpha e_1 v^t$ es Hessenberg superior

$$A = QR + uv^t = Q(R + wv^t) = QH^t H(R + wv^t) =$$

↖ H ortogonal

$$= QH^t(HR + Hwv^t) = QH^t(HR + \alpha e_1 v^t)$$

Métodos para S. Ec. No Lin.

Updating de la desc. QR.

Paso 2

Buscar una matriz P ortogonal que triangularice la matriz de Hessenberg

Construimos P como una secuencia de rotaciones de Givens

$$QH^t(HR + \alpha e_1 v^t) = QH^t P P^t (HR + \alpha e_1 v^t)$$

Métodos para S. Ec. No Lin.

Broyden-QR

Implementación de Broyden con QR (Numerical Recipes in Fortran)

Broyden(x, F, B₀, tol)

1. $R0 = \|F(x)\|$
2. Cálculo de B₀ (jacobiano con diferencias finitas).
3. [Q,R]=desc. QR de B
2. Do While $\|F(x)\| > tol_r * R0 + tol_A$
 - 2.1 Resolver $\delta x = B^{-1}F(x)$ usando QR
 - 2.2 Actualizar descomposición QR
 - 2.4 $x = x + \delta x$

Métodos para S. Ec. No Lin.

Updating de la inversa

Planteamiento del problema

- Sea B matriz cuadrada cuya inversa B^{-1} es conocida.
- Dada la matriz $A = B + uv^t$, obtener A^{-1} con coste $O(n^2)$

Proposición: Sea B matriz cuadrada no singular $\in \mathbb{R}^{n \times n}$ y sean $u, v \in \mathbb{R}^n$. Entonces $B + uv^t$ es invertible si y solo si $1 + v^t B^{-1} u \neq 0$. En este caso,

$$(B + uv^t)^{-1} = \left(I - \frac{(B^{-1}u)v^t}{1 + v^t B^{-1}u} \right) B^{-1}$$

Fórmula de Sherman-Morrison