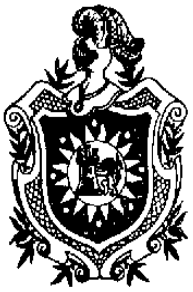
UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA-LEÓN

FACULTAD DE CIENCIAS Y TECNOLOGÍA

DEPARTAMENTO DE MATEMÁTICA Y ESTADÍSTICA



SOLUCIÓN NUMÉRICA DE SISTEMAS NO LINEALES

MONOGRAFÍA

PARA OPTAR AL TÍTULO DE LICENCIADA EN MATEMÁTICA

PRESENTADA POR

BRA. MARÍA MILAGROS PÉREZ OLIVAS

TUTORA

MSC. ANGELA ALTAMIRANO SLINGER

LEÓN, ABRIL 2009

DEDICATORIA

*A **Dios** todopoderoso, por ser la luz que guio mi camino para lograr mi meta.*

*A **mis padres** por ser la pieza fundamental en mis triunfos, en especial a mi madre **Milagros Olivas** por su apoyo incondicional.*

*A **mis hermanos** por soñar en un futuro mejor, y juntos vencer los obstáculos que se nos han presentado a lo largo de la vida.*

AGRADECIMIENTO

*A **Dios** sobre todas las cosas, por guiarme y protegerme siempre.*

*A mi tutora **MSC. Ángela Altamirano Slinger** por su paciencia y dedicación. Por brindarme su valioso tiempo y ser un ejemplo a seguir.*

*A todos **los profesores** que me brindaron sus conocimientos a lo largo de la carrera, sobre todo aquellos que fueron pieza fundamental para la culminación de mi meta.*

*A **mi familia** por ser tan especial y apoyarme en todo momento para la culminación de este trabajo investigativo.*

ÍNDICE

INTRODUCCIÓN

Los sistemas de ecuaciones no lineales se aplican en muchos contextos. Para resolverlos podemos hacerlo por la vía directa o por la vía indirecta, esta última no siempre es muy satisfactoria.

En la forma directa se facilitan las cosas si el método se basa en aspectos ya conocidos para el caso de una ecuación no lineal con una variable, adaptándolos o generalizándolos para incorporar todas las variables del sistema no lineal, utilizando para ello un enfoque vectorial.

Dada la importancia que tienen los sistemas de ecuaciones no lineales, consideré de mucho interés el desarrollar una monografía que proporcione los métodos más usuales para la solución numérica de sistemas no lineales, presentando su descripción, algoritmo e ilustración.

Este trabajo monográfico puede ser utilizado por estudiantes o profesionales como un material de consulta ante una situación en la que se requiera elegir un método particular de solución numérica de sistemas no lineales.

El presente trabajo consta de tres capítulos. En el primer capítulo se introducen los elementos básicos para estudiar los métodos numéricos que se desarrollan en el capítulo dos. En el segundo capítulo se analizan seis métodos iterativos para aproximar la solución de sistemas de ecuaciones no lineales.

En el capítulo tercero se presentan aplicaciones, con cuatro casos donde se utilizan los métodos estudiados. En los anexos se incluyen los programas elaborados para los diversos métodos abordados, así como su corrida correspondiente. Los programas fueron implementados en MATLAB, un software apropiado para desarrollar tareas de Análisis Numérico.

OBJETIVOS

OBJETIVO GENERAL

- Mostrar algunos métodos iterativos de aproximación numérica a la solución de un sistema de ecuaciones no lineales.

OBJETIVOS ESPECÍFICOS

- Caracterizar algunos métodos iterativos para la solución numérica de sistemas no lineales, mediante la descripción de sus componentes y la elaboración del algoritmo respectivo.

- Implementar los algoritmos asociados a los métodos estudiados utilizando el paquete MATLAB.

- Ilustrar la aplicación de los métodos abordados, mediante el estudio de casos.

CAPÍTULO 1

ELEMENTOS BÁSICOS

1.1 Introducción

En este capítulo presentaremos los conceptos y resultados que sirven de base para el estudio de los métodos iterativos que aproximan numéricamente la solución de sistemas de ecuaciones no lineales.

Para la solución numérica de ecuaciones de una variable y de problema de valor inicial en ecuaciones diferenciales ordinarias, que se muestran la final del capítulo, se presentan únicamente las técnicas numéricas asociadas a los métodos estudiados en el capítulo 2.

1.2 Sucesiones convergentes en $\mathbb{R}$

Una sucesión de números reales $a_1, a_2, a_3, \dots, a_n, \dots$ que denotaremos $\{a_n\}_{n=1}^{\infty}$ tiene limite ℓ si para cada $\varepsilon > 0$ existe un numero $N \in \mathbb{N}$ tal que $|a_n - \ell| < \varepsilon$ cuando $n > N$.

Definición 1.1

Si una sucesión $\{a_n\}_{n=1}^{\infty}$ tiene un límite, se dice que la sucesión es convergente y decimos que a_n converge a ese límite. Si la sucesión no es convergente, se dice que es divergente.

Teorema 1.2

Si $\{a_n\}_{n=1}^{\infty}$ y $\{b_n\}_{n=1}^{\infty}$ son sucesiones convergentes y c es una constante, entonces

i) la sucesión constante $\{c\}$ tiene a c como su límite;

$ii)$ $\lim_{n\to\infty} ca_n = c \lim_{n\to\infty} a_n;$

$iii)$ $\lim_{n\to\infty} (a_n \pm b_n) = \lim_{n\to\infty} a_n \pm \lim_{n\to\infty} b_n;$

$iv)$ $\lim_{n\to\infty} a_n b_n = \left(\lim_{n\to\infty} a_n\right)\left(\lim_{n\to\infty} b_n\right);$

$v)$ $\lim_{n\to\infty} \frac{a_n}{b_n} = \frac{\lim_{n\to\infty} a_n}{\lim_{n\to\infty} b_n}$ si $\lim_{n\to\infty} b_n \neq 0.$

Definición 1.3

Supongamos que $\{p_n\}_{n=0}^{\infty}$ es una sucesión que converge a p y que $e_n = p_n - p$ para cada $n \geq 0$. Si existen constantes positivas λ y α tales que

$$\lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^{\alpha}} = \lambda,$$

entonces se dice que $\{p_n\}_{n=0}^{\infty}$ converge a p de orden α , con una constante de error asintótico λ .

A una técnica iterativa para resolver un problema de la forma $x = g(x)$ se le denomina de orden α si, siempre que el método produce una convergencia para una sucesión $\{p_n\}_{n=0}^{\infty}$ donde $p_n = g(p_{n-1})$ para $n \geq 1$, la sucesión converge a la solución con orden α.

En general, una sucesión con un orden de convergencia grande convergerá más rápidamente que una sucesión con un orden más bajo. La constante asintótica afectara la rapidez de convergencia,

pero no es tan importante como el orden. Se dará atención especial a dos casos de orden:

i) Si $\alpha = 1$, entonces el método se denomina lineal.

ii) Si $\alpha = 2$, entonces el método se denomina cuadrático.

Definición 1.4

Se dice que una sucesión $\{p_n\}_{n=0}^{\infty}$ es superlinealmente convergente a p si $\lim_{n\to\infty} \frac{p_{n+1} - p}{p_n - p} = 0.$

1.3 Vectores en $\mathbb{R}^n$

Sea n un entero positivo, entonces una n-ada ordenada es una sucesión de n números reales $(x_1, x_2, \ldots, x_n)$. El conjunto de todas las n-adas ordenadas se conoce como espacio n dimensional y se denota por $\mathbb{R}^n$.

Definición 1.5

Las normas $\|.\|_2$ y $\|.\|_\infty$ para el vector $x = (x_1, x_2, \ldots, x_n)$ de $\mathbb{R}^n$ se definen como

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

y

$$\|x\|_\infty = \max_{1 \le i \le n} |x_i|$$

La norma $\|.\|_2$ se denomina frecuentemente norma Euclidiana del vector x ya que representa la noción usual de distancia al origen en el caso en que x esté en $\mathbb{R}^1 = \mathbb{R}, \mathbb{R}^2$ o $\mathbb{R}^3$.

Definición 1.6

Se dice que una sucesión $\{x^{(k)}\}_{k=1}^{\infty}$ de vectores en $\mathbb{R}^n$ converge a x con respecto a la norma $\|\cdot\|$ si, dado cualquier $\varepsilon > 0$ existe un entero $N(\varepsilon) \in \mathbb{N}$ tal que $\|x^{(k)} - x\| < \varepsilon$ para toda $k \geq N(\varepsilon)$.

Teorema 1.7

La sucesión de vectores $\{x^{(k)}\}_{k=1}^{\infty}$ converge a x en $\mathbb{R}^n$ con respecto a $\|\cdot\|_\infty$ si y sólo si $\lim_{n\to\infty} x_i^{(k)} = x_i$ para cada $i = 1, 2, \ldots, n$.

Definición 1.8

Se dice que una sucesión $\{x^{(k)}\}_{k=1}^{\infty}$ de vectores en $\mathbb{R}^n$ es superlinealmente convergente a p si $\lim_{i\to\infty} \frac{\|x^{(i+1)} - p\|}{\|x^{(i)} - p\|} = 0.$

1.4 Funciones de varias variables

Sean $x = (x_1, x_2, \ldots, x_n)$ un vector del espacio $\mathbb{R}^n$ y $D \subset \mathbb{R}^n$, entonces una función $f: D \to \mathbb{R}$ es una regla que asocia cada vector $x \in D$ con un único número real $f(x) = f(x_1, x_2, \ldots, x_n)$.

Definición 1.9

Sea $D \subset \mathbb{R}^n$ y sea $f: D \to \mathbb{R}$ una función. Se dice que $\lim_{x \to x_0} f(x) = L$ si dado cualquier número $\varepsilon > 0$, existe un numero $\delta > 0$ con la propiedad de que $|f(x) - L| < \varepsilon$ siempre que $x \in D$ y $0 < \|x - x_0\| < \delta$.

Debe hacerse notar que la existencia de un límite es independiente de la norma vectorial particular usada debido a la equivalencia de las normas vectoriales en $\mathbb{R}^n$.

Definición 1.10

Sea $D \subset \mathbb{R}^n$ y sea $f: D \to \mathbb{R}$ una función. Se dice f que es continua en $x_0 \in D$ siempre y cuando el $\lim_{x \to x_0} f(x)$ exista y $\lim_{x \to x_0} f(x) = f(x_0)$.

Si $f: D \to \mathbb{R}$, entonces $\frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \ldots, \frac{\partial f(x)}{\partial x_n}$ denota las derivadas parciales de f con respecto a x_i, con $i = 1, 2, \ldots, n$.

Teorema 1.11

Sea f una función de $D \subset \mathbb{R}^n$ en $\mathbb{R}$ y $x_0 \in D$. Si existen constantes $\delta > 0$ y $K > 0$ con $\left|\frac{\partial f(x)}{\partial x_j}\right| \leq K$ para cada $j = 1, 2, \ldots, n$, siempre que $\|x - x_0\| < \delta$ y $x \in D$, entonces f es continua en x_0.

Definición 1.12

Sean $f_1(x), f_2(x), \ldots f_n(x)$ funciones de n variables independientes, entonces su matriz jacobina es la matriz definida por

$$J(x) = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \cdots & \frac{\partial f_2(x)}{\partial x_n} \\ \vdots & \vdots & & \\ \frac{\partial f_n(x)}{\partial x_1} & \frac{\partial f_n(x)}{\partial x_2} & \cdots & \frac{\partial f_n(x)}{\partial x_n} \end{bmatrix}$$

Cuando tenemos una función de varias variables, la diferencial total es el instrumento que se usa para mostrar el efecto de los cambios de la variable independiente con respecto de la variable dependiente.

Definición 1.13

Sea $w = f(x_1, x_2, \cdots, x_n)$ una función de $D \subset \mathbb{R}^n$ en $\mathbb{R}$, entonces la diferencial total de la función es $dw = \frac{\partial f(x)}{\partial x_1} dx_1 + \frac{\partial f(x)}{\partial x_2} dx_2 + \cdots + \frac{\partial f(x)}{\partial x_n} dx_n$ donde $dx_i = \Delta x_i$.

Definición 1.14

Sea una función $G: \mathbb{R}^n \to \mathbb{R}$, el gradiente en $x = (x_1, x_2, \ldots, x_n)$ se denota por $\nabla G(x)$ y se define $\nabla G(x) = \left(\frac{\partial G(x)}{\partial x_1}, \frac{\partial G(x)}{\partial x_2}, \cdots, \frac{\partial G(x)}{\partial x_n}\right)$.

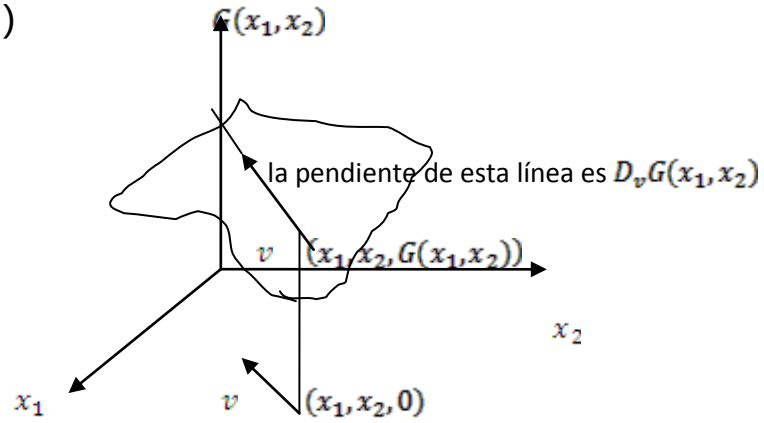
El gradiente de una función multivariada es análogo a la derivada de una función de una variable en el sentido de que una función multivariada diferenciable puede tener un mínimo en x solo cuando el gradiente es cero.

Definición 1.15

Sea $v = (v_1, v_2, \dots, v_n)$ un vector de $\mathbb{R}^n$ tal que $\|v\|_2^2 = \sum_{i=1}^{n} v_i^2 = 1$. La derivada direccional de G en x en la dirección de v se define por $D_v G(x) = \lim_{h \to 0} \frac{1}{h} [G(x + hv) - G(x)]$.

La derivada direccional de G en x en la dirección de v mide el cambio en el valor de la función G relativo al cambio de la variable en la dirección de v. (Ver en la fig. 1.1 una ilustración cuando G es una función de dos variables.)

figura 1.1



Un resultado estándar del cálculo de funciones multivariadas establece que la dirección que produce el valor máximo de la derivada direccional se presenta cuando se escoge v de manera que sea paralelo a $\nabla G(x)$ siempre y cuando $\nabla G(x) \neq 0$.Como consecuencia, la dirección en la que el valor de G en x decrece más es aquella dada por $-\nabla G(x)$.

Sean $x = (x_1, x_2, \dots, x_n)$ un vector del espacio $\mathbb{R}^n$ y $D \subset \mathbb{R}^n$, entonces una función $F: D \to \mathbb{R}^n$ es una regla que asocia cada vector $x \in D$ con

un único elemento de $\mathbb{R}^n$ que representamos por $F(x) = \big(f_1(x), f_2(x), \dots, f_n(x)\big)$.

Definición 1.16

Sea una función $F: D \to \mathbb{R}^n$. Definimos $\lim_{x \to x_0} F(x) = L = (L_1, L_2, \cdots, L_n)$ si y sólo si $\lim_{x \to x_0} f_i(x) = L_i$ para cada $i = 1, 2, \dots, n$.

Definición 1.17

Sea una función $F: D \to \mathbb{R}^n$. Se dice que F es continua en $x_0 \in D$ si $\lim_{x \to x_0} F(x)$ existe y $\lim_{x \to x_0} F(x) = F(x_0)$. Se dice que F es continua en el conjunto D si F es continua en cada x de D.

Sea una función $F: D \to \mathbb{R}^n$ tal que $F(x) = \big(f_1(x), f_2(x), \dots, f_n(x)\big)$, la diferencial de F es $dF = (df_1, df_2, \dots, df_n)$ donde

$$df_i = \sum_{i=1}^{n} \frac{\partial f_i(x)}{\partial x_i} dx_i$$

1.5 Solución numérica de ecuaciones de una variable

Se abordan únicamente dos métodos para encontrar una raíz o solución de una ecuación de la forma $f(x) = 0$ para una función dada f.

1.5.1 Método de Newton-Raphson

El método de Newton-Raphson es uno de los métodos numéricos más poderoso para la resolución del problema de búsqueda de raíces de $f(x) = 0$.

Teorema 1.18

Sea f una función tal que f' y f'' son continuas en $[a,b]$. Si $p \in [a,b]$ es tal que $f(p) = 0$ y $f'(p) \neq 0$, entonces existe $\delta > 0$ tal que el método de Newton genera una sucesión $\{p_n\}_{n=1}^{\infty}$ donde $p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}$, $n \geq 1$ que converge a p para cualquier aproximación inicial $p_0 \in [p-\delta, p+\delta]$.

Algoritmo de Newton-Raphson

Para encontrar una solución de la ecuación $f(x) = 0$ dada una aproximación inicial p_0:

ENTRADA; aproximación inicial p_0; tolerancia TOL; número máximo de iteraciones N_0.

SALIDA: solución aproximada p o mensaje de fracaso.

Paso 1 Tomar $i = 1$.

Paso 2 Mientras que $i \leq N_0$ seguir Pasos 3-6.

Paso 3 Tomar $p = p_0 - f(p_0)/f'(p_0)$. (Calcular p_i).

Paso 4 Si $|p - p_0| < TOL$ entonces

SALIDA (p); (Procedimiento completado

satisfactoriamente). PARAR.

Paso 5 Tomar $i = i + 1$.

Paso 6 Tomar $p_0 = p$. (Redefinir p_0).

Paso 7 SALIDA (′El método fracasó después de N_0 iteraciones,

N_0=′, N_0); (procedimiento completado sin éxito). PARAR.

1.5.2 Método de la Secante

Para evitar el problema de la evaluación de la derivada en el método de Newton-Raphson se emplea una variante. Se usa $f'(p_{n-1})$ por su valor aproximado $\frac{f(p_{n-1})-f(p_{n-2})}{p_{n-1}-p_{n-2}}$.

Sustituyendo esta aproximación para $f'(p_{n-1})$ en la formula de Newton en el teorema 1.17, se obtiene

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}$$

Esta técnica se llama método de la Secante.

Algoritmo de la Secante

Para encontrar una solución de la ecuación $f(x) = 0$, dadas las aproximaciones iniciales p_0 y p_1:

ENTRADA: aproximaciones iniciales p_0 y p_1; tolerancia TOL; número máximo de iteraciones N_0.

SALIDA; solución aproximada p o mensaje de fracaso.

Paso 1 Tomar $i = 2$;

$q_0 = f(p_0)$;

$q_1 = f(p_1)$.

Paso 2 Mientras que $i \leq N_0$ seguir los pasos 3-6.

Paso 3 Tomar $p = p_1 - q_1(p_1 - p_0) / (q_1 - q_0)$.(Calcular p_i).

Paso 4 Si $|p - p_1| < TOL$ entonces

SALIDA (p); (Procedimiento completado

satisfactoriamente). PARAR.

Paso 5 Tomar $i = i + 1$.

Paso 6 Tomar $p_0 = p_1$; (Redefinir p_0,q_1,p_1,q_1).

$q_0 = q_1$;

$p_1 = p$;

$q_1 = f(p)$.

Paso 7 SALIDA (′El método fracasó después de N_0 iteraciones,

N_0 =′, N_0); (procedimiento completado sin éxito). PARAR.

1.6 Solución numérica de problemas de valor inicial para ecuaciones diferenciales ordinarias

Resolver un problema de valor inicial es resolver una ecuación diferencial de la forma $y' = f(t,y), a \leq t \leq b$ sujeta a la condición inicial $y(a) = \alpha$.

Definición 1.19

Se dice que una función $f(t,y)$ satisface una condición de Lipschitz para la variable y en un conjunto $D \subset \mathbb{R}^2$, si existe una constante $L > 0$ con la propiedad de que $|f(t,y_1) - f(t,y_2)| \leq L|y_1 - y_2|$ siempre y cuando $(t,y_1),(t,y_2) \in D$.

La constante L se llama una constante de Lipschitz para f.

Definición 1.20

Se dice que un conjunto $D \subset \mathbb{R}^2$ es convexo si, siempre que (t_1,y_1) y (t_2,y_2) pertenecen a D, $\left((1-\lambda)t_1 + \lambda t_2\,,\ (1-\lambda)y_1 + \lambda y_2\right)$ también pertenece a D para cada λ, $0 \leq \lambda \leq 1$.

En términos geométricos, la definición dice que un conjunto es convexo si para cualquier pareja de puntos pertenecientes al conjunto, el segmento entero de recta que une a los dos puntos también pertenece al conjunto.

Teorema 1.21

Suponga que $f(t,y)$ está definida en un conjunto convexo $D \subset \mathbb{R}^2$. Si existe una constante $L > 0$ tal que

$$\left|\frac{\partial f(t,y)}{\partial y}\right| \leq L \ para\, toda \ (t,y) \in D$$

entonces f satisface una condición de Lipschitz en D para la variable y con constante de Lipschitz L.

Teorema 1.22

Supongamos que $D = \{(t,y)/a \leq t \leq b \ \mathbf{y} \ -\infty < y < \infty\}$ y que $f(t,y)$ es continua en D .Si f satisface una condición de Lipschitz en D para la variable y, entonces el problema de valor inicial $y' = f(t,y), a \leq t \leq b, \ y(a) = \alpha,$ tiene una única solución $y(t)$ para $a \leq t \leq b$.

Definición 1.23

Se dice que el problema de valor inicial $\frac{dy}{dt} = f(t,y), a \leq t \leq b, y(a) = \alpha$ es un problema bien planteado si:

i) existe una solución única, $y(t)$, para este problema;

ii) existen constates positiva ε y k, con la propiedad de que existe una única solución $z(t)$, al problema

$$\frac{dz}{dt} = f(t,z) + \delta(t), a \leq t \leq b, z(a) = \alpha + \varepsilon_0, \quad \text{(EC. 1.1)}$$

con $|z(t) - y(t)| < k\varepsilon \ para\, toda\, a \leq t \leq b$ siempre que $|\varepsilon_0| < \varepsilon$ y $|\delta(x)| < \varepsilon$.

El problema especificado por la ecuación (1.1) se acostumbra llamar problema perturbado asociado con el problema original. El teorema

siguiente especifica las condiciones que aseguran que el problema de valor inicial esté bien planteado.

Teorema 1.24

Supongamos que $D = \{(t,y)/a \leq t \leq b\, \mathrm{y} - \infty < y < \infty\}$. El problema de valor inicial $\frac{dy}{dt} = f(t,y), a \leq t \leq b, y(a) = \alpha,$ está bien planteado si f es continua y satisface una condición de Lipschitz para la variable y en el conjunto D.

1.6.1 Método de Runge-Kutta de cuarto orden

Los métodos de Runge-Kutta sirven para obtener una aproximación al problema de valor inicial bien planteado

$$\frac{dy}{dt} = f(t,y), a \leq t \leq b, y(a) = \alpha.$$

No se obtiene una aproximación continua de la solución $y(t)$ sino que se generarán aproximaciones de y en varios puntos, llamados puntos de red, en el intervalo $[a,b]$. Los puntos de red están distribuidos uniformemente sobre el intervalo $[a,b]$. Esta condición se puede garantizar escogiendo un entero positivo N y seleccionando los puntos de red $\{t_0, t_1, t_2, \cdots, t_N\}$ donde $t_i = a + ih$ para cada $i = 0,1,2., \ldots, N$. La distancia común entre los puntos, $h = (b-a)/N$ se llama tamaño de paso.

El método de Runge-Kutta que se usa más a menudo es de orden cuatro y, en forma de diferencia está dado por:

$$w_0 = \alpha,$$

$$k_1 = h(t_i, w_i),$$

$$k_2 = hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1\right),$$

$$k_3 = hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2\right),$$

$$k_4 = hf(t_{i+1}, w_i + k_3),$$

$$w_{i+1} = w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

para cada $i = 0,1,2,\ldots,N-1$. Este método tiene un error de truncamiento de $\mathcal{O}(h^4)$, siempre y cuando la solución $y(t)$ tenga cinco derivadas continuas. Los términos k_1, k_2, k_3, k_4 se introducen en el método para eliminar la necesidad de encajar sucesivamente la evaluación en la segunda variable de la función $f(t,y)$.

Algoritmo de Runge-Kutta *(de cuarto orden)*

Para aproximar la solución del problema de valor inicial

$$y' = f(t,y), a \leq t \leq b, y(a) = \alpha,$$

en $(N+1)$ números igualmente espaciados en el intervalo $[a,b]$:

ENTRADA: puntos extremos a, b ; entero N; condición inicial α.

SALIDA: aproximación w de y en los $(N+1)$ valores de t.

Paso 1 Tomar $h = (b-a)/N$;

$t = a$;

$w = \alpha$;

SALIDA (t, w).

Paso 2 Para $i = 0,1,2,\ldots,N$ seguir los pasos 3-5

Paso 3 Tomar $k_1 = hf(t,w)$;

$k_2 = hf(t + h/2, w + k_1/2)$;

$k_3 = hf(t + h/2, w + k_2/2)$;

$k_4 = hf(t + h, w + k_3)$.

Paso 4 Tomar $w = w + (k_1 + 2k_2 + 2k_3 + k_4)/6$;

(Calcular w_i).

$t = a + ih$ (Calcular t_i).

Paso 5 SALIDA (t, w).

Paso 6 PARAR

1.6.2 Sistemas de ecuaciones diferenciales

Un sistema de orden m de problemas de valor inicial de primer orden se puede expresar en la forma

$$\frac{du_1}{dt} = f_1(t,u_1,u_2,\cdots,u_m),$$

$$\frac{du_2}{dt} = f_2(t,u_1,u_2,\cdots,u_m),$$

$$\vdots \qquad \text{(EC. 1.2)}$$

$$\frac{du_m}{dt} = f_m(t,u_1,u_2,\cdots,u_m)$$

para $a \leq t \leq b$; con las condiciones iniciales

$$u_1(a) = \alpha_1,$$

$$u_2(a) = \alpha_2,$$

$$\vdots \qquad \text{(EC. 1.3)}$$

$$u_m(a) = \alpha_m.$$

Se pretende encontrar m funciones $u_1,u_2,\cdots,u_m$ que satisfagan las ecuaciones diferenciales del sistema y las condiciones iniciales.

Definición 1.25

Se dice que la función $f(t,y_1,y_2,\cdots,y_m)$, definida en el conjunto $D = \{\ (t,u_1,u_2,\cdots,u_m)\ /\ a \leq t \leq b\ ,\quad -\infty < u_i < \infty,\quad para\ cada\quad i = 1,2,\ldots,m\ \}$ satisface una condición de Lipschitz en D en las variables $u_1,u_2,\cdots,u_m$ si existe una constante $L > 0$ con la propiedad de que

$$|f(t,u_1,u_2,\cdots,u_m) - f(t,z_1,z_2,\cdots,z_m)| \leq L\sum_{j=1}^{m}|u_j - z_j|$$

para todas $(t,u_1,u_2,\cdots,u_m)$ y $(t,z_1,z_2,\cdots,z_m)$ en D.

Teorema 1.26

Suponga que si f y sus primeras derivadas parciales son continuas en D y si $\left|\frac{\partial f(t,u_1,u_2,\cdots,u_m)}{\partial u_i}\right| \leq L$ para cada $i = 1,2,\cdots,m$ y toda $(t,u_1,u_2,\cdots,u_m)$ en D, entonces f satisface una condición de Lipschitz en D con constante de Lipschitz L.

Teorema 1.27

Supongamos que $D = \{(t,u_1,u_2,\cdots,u_m) / a \leq t \leq b, -\infty < u_i < \infty,$

$para\, cada\, i = 1,2,\cdots,m\}$ y sea $f_i(t,u_1,u_2,\cdots,u_m)$, para cada $i = 1,2,\ldots,m$ continua en D y tal que satisface una condición de Lipschitz ahí. El sistema de ecuaciones diferenciales de primer orden (1.2) sujetas a las condiciones iniciales (1.3) tiene una solución única $u_1(t),u_2(t),\cdots,u_m(t)$ para $a \leq t \leq b$.

Los métodos para resolver sistemas de ecuaciones diferenciales de primer orden son simplemente generalizaciones de los métodos para una sola ecuación de primer orden presentados anteriormente en este capítulo.

Generalizando el método de Runge-Kutta de orden cuatro para una sola ecuación de primer orden se obtiene el método de Runge-Kutta para sistemas de ecuaciones diferenciales de primer orden.

Para ello escogemos un entero $N > 0$ y tomamos $h = (b-a)/N$ para dividir el intervalo $[a,b]$ en N subintervalos con puntos de red $t_j = a + jh$ para cada $j = 0,1,2.,\ldots,N$. Usaremos la notación $w_{i,j}$ para denotar una aproximación de $u_i(t_j)$ para cada $j = 0,1,2.,\ldots,N$ e

$i = 1,2,\dots,m$; esto es, $w_{i,j}$ aproximará a la i-ésima solución $u_i(t)$ de (1.3) en el j-ésimo punto de red t_j.Para las condiciones iniciales, tomamos

$$w_{1,0} = \alpha_1,$$

$$w_{2,0} = \alpha_2,$$

$$\vdots$$

$$w_{m,0} = \alpha_m.$$

Si suponemos que los valores $w_{1,j}, w_{2,j}, \cdots, w_{m,j}$ han sido calculados, obtenemos $w_{1,j+1}, w_{2,j+1}, \cdots, w_{m,j+1}$ calculando primero

$$k_{1,i} = hf_i\left(t_j, w_{1,j}, w_{2,j}, \dots, w_{m,j}\right) para\ cada\ i = 1,2,\dots,m\ ;$$

$$k_{2,i} = hf_i\left(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{1,1}, w_{2,j} + \frac{1}{2}k_{1,2}, \dots, w_{m,j} + \frac{1}{2}k_{1,m}\right) para\ cada\ i = 1,2,\dots,m;$$

$$k_{3,i} = hf_i\left(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{2,1}, w_{2,j} + \frac{1}{2}k_{2,2}, \dots, w_{m,j} + \frac{1}{2}k_{2,m}\right) para\ cada\ i = 1,2,\dots,m;$$

$$k_{4,i} = hf_i\left(t_j + h, w_{1,j} + k_{3,1}, w_{2,j} + k_{3,2}, \dots, w_{m,j} + k_{3,m}\right) para\ cada\ i = 1,2,\dots,m;$$

y entonces

$$w_{i,j+1} = w_{i,j} + \frac{1}{6}\left[k_{1,i} + 2k_{2,i} + 3k_{3,i} + k_{4,i}\right] para\ cada\ i = 1,2,\dots,m.$$

Debe notarse que $k_{1,1}, k_{1,2}, \cdots, k_{1,m}$ deben ser calculados antes de que $k_{2,1}$ pueda ser determinado. En general, cada $k_{l,1}, k_{l,2}, \cdots, k_{l,m}$ debe ser calculado antes que cualquiera de las expresiones $k_{l+1,i}$.

Algoritmo de Runge-Kutta para sistemas de ecuaciones diferenciales de primer orden

Para aproximar la solución del sistema de orden m de problemas de valor inicial de primer orden

$$u_j' = f_j(t, u_1, u_2, \cdots, u_m), j = 1,2, \cdots, m$$

$$a \leq t \leq b, u_j(a) = \alpha_j, j = 1,2, \cdots, m$$

en $(n+1)$ números igualmente espaciados en el intervalo $[a, b]$:

ENTRADA: puntos extremos a, b; números de ecuaciones m; entero N; condiciones iniciales $\alpha_1, \alpha_2, \cdots, \alpha_m$.

SALIDA: aproximaciones w_j a $u_j(t)$ en los $(N+1)$ valores de t.

Paso 1 Tomar $h = (b-a)/N$;

$$t = a.$$

Paso 2 Para $j = 1,2, \ldots, m$ tomar $w_j = \alpha_j$.

Paso 3 SALIDA $(t, w_1, w_2, \cdots, w_m)$.

Paso 4 Para $i = 1,2, \ldots, N$ seguir los pasos 5-11.

Paso 5 Para $j = 1,2, \ldots, m$ tomar

$$k_{1,j} = hf_j(t, w_1, w_2, \dots, w_m).$$

Paso 6 Para $j = 1,2,\dots,m$ tomar

$$k_{2,j} = hf_j\left(t + \frac{h}{2}, w_1 + \frac{1}{2}k_{1,1}, w_2 + \frac{1}{2}k_{1,2}, \dots, w_m + \frac{1}{2}k_{1,m}\right).$$

Paso 7 Para $j = 1,2,\dots,m$ tomar

$$k_{3,j} = hf_j\left(t + \frac{h}{2}, w_1 + \frac{1}{2}k_{2,1}, w_2 + \frac{1}{2}k_{2,2}, \dots, w_m + \frac{1}{2}k_{2,m}\right).$$

Paso 8 Para $j = 1,2,\dots,m$ tomar

$$k_{4,j} = hf_j\left(t + h, w_1 + k_{3,1}, w_2 + k_{3,2}, \dots, w_m + k_{3,m}\right)$$

Paso 9 Para $j = 1,2,\dots,m$ tomar

$$w_j = w_j + \left[k_{1,j} + 2k_{2,j} + 3k_{3,j} + k_{4,j}\right]/6.$$

Paso 10 Tomar $t = a + ih$.

Paso 11 SALIDA $(t, w_1, w_2, \cdots, w_m)$.

Paso 12 PARAR.

1.7 Homotopía

A continuación presentaremos algunos conceptos topológicos que se utilizarán para el estudio de uno de los métodos a estudiar en el capítulo 2.Iniciaremos analizando la definición de topología y espacio topológico.

Definición 1.28

a) Una colección τ de subconjuntos de un conjunto X se dice es una topología en X si τ cumple las propiedades:

$i)$ $\phi \in \tau$ y $X \in \tau$.

$ii)$ Si $V_i \in \tau$ para $i = 1,2,\dots,n$, entonces $V_1 \cap V_2 \cap \cdots \cap V_n \in \tau$.

$iii)$ Si $\{V_\alpha\}$ es una colección arbitraria de de elementos de τ

(finito, contable, o incontable), entonces $\bigcup_\alpha V_\alpha \in \tau$.

b) Si τ es una topología en X, entonces X es llamado un espacio topológico, y los elementos de τ son llamados los conjuntos abiertos en X .

Definición 1.29

Si X es un espacio topológico, una curva X en es un mapeo continuo γ de un intervalo compacto $[\alpha, \beta] \subset \mathbb{R}^1$ en X (dentro de); aquí $\alpha < \beta$. Llamaremos $[\alpha, \beta]$ el intervalo paramétrico de γ y denotaremos el rango de γ por γ^*. Entonces γ es un mapeo, y γ^* es el conjunto de todos los puntos $\gamma(t)$, para $\alpha \leq t \leq \beta$.

Si el punto inicial $\gamma(\alpha)$ de γ coincide con el punto final $\gamma(\beta)$, llamaremos a γ una curva cerrada.

Definición 1.30

Supongamos que γ_0 y γ_1 son curvas cerradas en un espacio topológico X, con intervalo $I = [0,1]$. Decimos que γ_0 y γ_1 son homotópicas si es un continuo mapeo H de la unidad cuadrada $I^2 = I \times I$ en X esto es

$$H(s,0) = \gamma_0(s), \qquad H(s,1) = \gamma_1(s), \qquad H(s,t) = H(1,t)$$

CAPÍTULO 2

MÉTODOS ITERATIVOS PARA SISTEMAS NO LINEALES

2.1 Introducción

En este capítulo se estudian seis métodos de aproximación numérica de la solución de sistemas de ecuaciones no lineales.

Representaremos un sistema de ecuaciones no lineales en la forma general

$$f_1(x_1, x_2, \cdots, x_n) = 0$$

$$f_2(x_1, x_2, \cdots, x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2, \cdots, x_n) = 0$$

donde f_i para $i = 1,2, \ldots, n$ es una función de $D \subset \mathbb{R}^n$ en $\mathbb{R}$. Alternativamente el sistema puede representarse como $F(x) = 0$ donde F una función de $D \subset \mathbb{R}^n$ en $\mathbb{R}^n$.

2.2 Iteración de Punto Fijo

Iniciaremos definiendo el concepto de punto fijo para una función de varias variables.

Definición 2.1

Sea $D \subset \mathbb{R}^n$. Se dice que una función $G: D \to \mathbb{R}^n$ tiene un único punto fijo $p \in D$ si $G(p) = p$.

Teorema 2.2

Sea $D = \{(x_1, x_2, \cdots, x_n) \,/\, a_i \leq x_i \leq b_i \;\; para\, cada \;\; i = 1,2,\ldots,n\}$ para alguna colección de constantes $a_1, a_2, \cdots, a_n$ y $b_1, b_2, \cdots, b_n$.Supongamos que $G: D \to \mathbb{R}^n$ es una función continua con primeras derivadas parciales continuas con la propiedad de que $G(x) \in D$ para $x \in D$. Entonces G tiene un punto fijo en D. Además, supóngase que existe una constante $K < 1$ con $\left|\frac{\partial g_i}{\partial x_j}(x)\right| \leq \frac{K}{n}$ $siempre\, que\, x \in D, para\, cada$ $j = 1,2,\ldots,n$ y cada función g_i. Entonces la sucesión $\{x^{(k)}\}_{k=0}^{\infty}$ definida por una $x^{(0)}$ en D seleccionada arbitrariamente y generada por $x^{(k)} = G\left(x^{(k-1)}\right)$ para cada $k \geq 1$ converge al punto fijo único $p \in D$ y $\left\|x^{(k)} - p\right\|_\infty \leq \frac{K^k}{1-K}\left\|x^{(1)} - x^{(0)}\right\|_\infty$.

Ejemplo

Sea $D = \{(x_1, x_2, x_3) \,/\, 0 \leq x_i \leq 1.5 \;\; para\, cada \;\; i = 1,2,3\}$ y la función $G: D \to \mathbb{R}^3$ definida por $G(x_1, x_2, x_3) = (g_1(x_1, x_2, x_3), g_2(x_1, x_2, x_3),$ $g_3(x_1, x_2, x_3))$ donde

$$g_1(x_1, x_2, x_3) = \frac{13 - x_2^2 + 4x_3}{15} \;\; , \;\; g_2(x_1, x_2, x_3) = \frac{11 + x_3 - x_1^2}{10}$$

$$g_3(x_1, x_2, x_3) = \frac{22 + x_2^3}{25}$$

Demuestre que G tiene un punto fijo único en D y aproxime el punto fijo p, tomando $x^{(0)} = (1,1,1)$ con cota de error de 10^{-4}.

Solución:

Usando el teorema 2.2 para demostrar que G tiene un único punto fijo en D, se tiene que

$$|g_1(x_1,x_2,x_3)| = \left|\frac{13 - x_2^2 + 4x_3}{15}\right| \leq 1.1167$$

$$|g_2(x_1,x_2,x_3)| = \left|\frac{11 + x_3 - x_1^2}{10}\right| \leq 1.025$$

$$|g_3(x_1,x_2,x_3)| = \left|\frac{22 + x_2^3}{25}\right| \leq 1.015$$

como $0 \leq g_i(x_1,x_2,x_3) \leq 1.5\ para\ cada\ \ i = 1,2,3$. Entonces $G(x) \in D$ siempre que $x \in D$.

Para las cotas de las derivadas parciales en D se obtiene:

$$\left|\frac{\partial g_1}{\partial x_1}\right| = 0, \left|\frac{\partial g_2}{\partial x_2}\right| = 0, \left|\frac{\partial g_3}{\partial x_3}\right| = 0, \left|\frac{\partial g_3}{\partial x_1}\right| = 0,$$

mientras que

$$\left|\frac{\partial g_1}{\partial x_2}\right| = \left|\frac{-2x_2}{15}\right| \leq 0.2, \left|\frac{\partial g_1}{\partial x_3}\right| = \frac{4}{15} \leq 0.267, \left|\frac{\partial g_2}{\partial x_1}\right| = \left|\frac{-2x_1}{10}\right| \leq 0.3,$$

$$\left|\frac{\partial g_2}{\partial x_3}\right| = \frac{1}{10} \leq 0.1, \left|\frac{\partial g_3}{\partial x_2}\right| = \left|\frac{3x_2^2}{25}\right| \leq 0.27,$$

como las derivadas parciales de g_1, g_2 y g_3 están acotadas en D, implica por el teorema 1.11 que estas funciones son continuas en D. Luego G es continua en D. Por lo tanto G tiene un punto fijo en D.

Además, para toda $x \in D \left|\frac{\partial g_i}{\partial x_j}(x)\right| \leq 0.3 \; para\, cada \;\; i = 1,2,3$ y $j = 1,2,3$.

Por tanto el valor que satisface a K es 0.9.

Para aproximar el punto fijo $p = (p_1, p_2, p_3)$ escogemos $x^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, x_3^{(0)}\right)$ generamos la sucesión $\left\{x^{(k)}\right\}_{k=1}^{\infty}$ mediante $x^{(k)} = G\left(x^{(k-1)}\right)$ que converja a p.

La sucesión generada por

$$x_1^{(k)} = \frac{13 - \left(x_2^{(k-1)}\right)^2 + 4x_3^{(k-1)}}{15}$$

$$x_2^{(k)} = \frac{11 + x_3^{(k-1)} - \left(x_1^{(k-1)}\right)^2}{10}$$

$$x_3^{(k)} = \frac{22 + \left(x_2^{(k-1)}\right)^3}{25}$$

da los siguientes resultados

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\left\|x^{(k)} - x^{(k-1)}\right\|_\infty$
0	1	1	1	_
1	1.066667	1.1	0.92	0.1
2	1.031333	1.078222	0.93324	0.035333
3	1.038026	1.086960	0.930140	8.7×10^{-3}
4	1.035939	1.085264	0.931369	2.08×10^{-3}
5	1.036512	1.085820	0.931129	5.7×10^{-4}

El punto que se aproxima a la solución del sistema no lineal es $x=(1.036512, 1.085820, 0.931129)$.

Dado un sistema no lineal

$$f_1(x_1, x_2, \cdots, x_n) = 0$$

$$f_2(x_1, x_2, \cdots, x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2, \cdots, x_n) = 0$$

si de la i-ésima ecuación se despeja x_i el sistema se transforma a $x_i = g_i(x_1, x_2, \cdots, x_n), i = 1,2, \cdots, n$ y puede resolverse como un problema de punto fijo.

Un punto fijo del sistema $x_i = g_i(x_1, x_2, \cdots, x_n), i = 1,2, \cdots, n$ es un punto $p = (p_1, p_2, \cdots, p_n)$ tal que $p_1 = g_1(p_1, p_2, \cdots, p_n)$,

$p_2 = g_2(p_1, p_2, \cdots, p_n), \cdots, p_n = g_n(p_1, p_2, \cdots, p_n)$.

Para aproximar el punto fijo $p = (p_1, p_2, \cdots, p_n)$ de $G = (g_1, g_2, \cdots, g_n)$, se genera a partir de $x^{(0)}$ una sucesión $\{x^{(k)}\}_{k=1}^{\infty}$ donde $x^{(k)} = g_i\left(x^{(k-1)}\right)$ que converja a p_i , $i = 1,2, \cdots, n$ respectivamente.

Para encontrar las componentes del vector $x_i^{(k)}$. se utiliza el esquema de cálculo

$$x_1^{(k)} = g_1\left(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \dots, x_{i-1}^{(k-1)}\right)$$

$$x_2^{(k)} = g_2\left(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \dots, x_{i-1}^{(k-1)}\right)$$

$$x_3^{(k)} = g_3\left(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \dots, x_{i-1}^{(k-1)}\right)$$

$$\vdots$$

$$x_n^{(k)} = g_n\left(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \dots, x_{i-1}^{(k-1)}\right)$$

Ejemplo

Sea el sistema no lineal

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0.$$

Transforme el sistema para resolverlo como un problema de punto fijo.

Solución:

El sistema dado puede transformarse al problema de punto fijo:

$$x_1 = g_1(x_1, x_2) = \frac{x_1^2 + x_2^2 + 8}{10}$$

$$x_2 = g_2(x_1, x_2) = \frac{x_1 x_2^2 + x_1 + 8}{10}$$

Aplicamos el teorema 2.2 para demostrar que $G = (g_1, g_2): D \subset \mathbb{R}^2 \to \mathbb{R}^2$ tiene un único punto fijo en $D = \{(x_1, x_2)/0 \leq x_1, x_2 \leq 1.5\}$.

Note que

$$g_1(x_1, x_2) = \left|\frac{x_1^2 + x_2^2 + 8}{10}\right| \leq 1.25$$

$$g_2(x_1, x_2) = \left|\frac{x_1 x_2^2 + x_1 + 8}{10}\right| \leq 1.2875$$

como $0 \leq g_i(x_1, x_2) \leq 1.5, para\ cada\ i = 1,2.$ Entonces $G(x) \in D$ siempre que $x \in D$.

Para las cotas de las derivadas parciales en D se obtiene:

$$\left|\frac{\partial g_1}{\partial x_1}\right| = \left|\frac{2x_1}{10}\right| \leq 0.3, \left|\frac{\partial g_1}{\partial x_2}\right| = \left|\frac{2x_2}{10}\right| \leq 0.3, \left|\frac{\partial g_2}{\partial x_1}\right| = \left|\frac{x_2^2+1}{10}\right| \leq 0.325,$$

$$\left|\frac{\partial g_2}{\partial x_2}\right| = |2x_1 x_2| \leq 0.45.$$

Como las derivadas parciales de g_1 y g_2 están acotadas en D, implica por el teorema 1.11 que estas funciones son continuas en D . Luego G es continua en D . Por lo tanto G tiene un punto fijo en D.

Además, para toda $x \in D$ $\left|\frac{\partial g_i}{\partial x_j}(x)\right| \leq 0.45\ para\ cada\ i = 1,2$ y $j = 1,2.$

Por tanto el valor que satisface a K es 0.9.

Aplicando iteración funcional, con $x^{(0)} = (0,0)$ y cota de error de 10^{-4} se genera la sucesión con las fórmulas

$$x_1^{(k)} = \frac{\left(x_1^{(k-1)}\right)^2 + \left(x_2^{(k-1)}\right)^2 + 8}{10}$$

$$x_2^{(k)} = \frac{\left(x_1^{(k-1)}\right)\left(x_1^{(k-1)}\right)^2 + \left(x_1^{(k-1)}\right) + 8}{10}$$

Los resultados se presentan en la tabla siguiente:

k	$x_1^{(k)}$	$x_2^{(k)}$	$\|x^{(k)} - x^{(k-1)}\|_\infty$
0	0	0	_
1	0.8	0.8	0.8
2	0.928	0.9312	0.1312
3	0.972832	0.973270	0.044832
4	0.989366	0.989435	0.208243
5	0.995783	0.995794	6.4×10^{-3}
6	0.998319	0.998321	2.5×10^{-3}
7	0.999329	0.999329	1.009×10^{-3}
8	0.999731	0.999732	4.03×10^{-4}

El punto que se aproxima a la solución del sistema no lineal dado es $x = (0.999731, 0.999732)$.

Algoritmo de Iteración de Punto Fijo

Para encontrar un punto fijo del sistema $x = G(x)$ dada una aproximación inicial $x^{(0)}$.

ENTRADA Numero n de ecuaciones y variables, aproximación inicial $x^{(0)} = x0 = (x0_1, x0_2, \cdots, x0_n)$, tolerancia tol, número máximo de iteraciones max.

SALIDA Solución aproximada $x = (x_1, x_2, \cdots, x_n)$ o mensaje que el número de iteraciones fue excedido.

Paso 1 Tomar $k = 1$

Paso 2 Mientras que $(k \leq max)$ seguir los pasos 3-6.

Paso 3 Tomar para $i = 1,2,\cdots,n \quad x = g_i(x0)$

Paso 4 Si $\|x - x0\| < tol$ entonces SALIDA $(x_1, x_2, \cdots, x_n)$

PARAR

Paso 5 Tomar $k = k + 1$

Paso 6 $x0 = x$

Paso 7 SALIDA (Número máximo de iteraciones excedido) PARAR.

En el anexo 1, se presenta el programa en MATLAB correspondiente al algoritmo del método de Iteración de Punto Fijo para sistemas no lineales.

En el método de Punto Fijo debemos generar una sucesión de tal forma que converja a una solución del sistema no lineal, el cumplimiento del teorema 2.2 garantiza que la iteración de punto fijo sea convergente, y también el punto inicial debe estar cerca de la solución, en otro caso se necesitaría una cantidad muy grande de iteraciones para dar solución al sistema no lineal. Además, debemos usar fórmulas de iteración distintas para encontrar otra solución.

2.3 Método Iterativo de Seidel

Podemos llevar a cabo una mejora del método de iteración de punto fijo para sistemas no lineales. Esta mejora lo que hace es acelerar la convergencia de la iteración de punto fijo y consiste

en usar las últimas estimaciones de $x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \ldots, x_{i-1}^{(k)}$ en vez de $x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \ldots, x_{i-1}^{(k-1)}$ para calcular $x_i^{(k)}$.

Para encontrar las componentes del vector $x_i^{(k)}$. se utiliza el esquema de cálculo

$$x_1^{(k)} = g_1\left(x_1^{(k-1)}, x_2^{(k-1)}, x_3^{(k-1)}, \ldots, x_{i-1}^{(k-1)}\right)$$
$$x_2^{(k)} = g_2\left(x_1^{(k)}, x_2^{(k-1)}, x_3^{(k-1)}, \ldots, x_{i-1}^{(k-1)}\right)$$
$$x_3^{(k)} = g_3\left(x_1^{(k)}, x_2^{(k)}, x_3^{(k-1)}, \ldots, x_{i-1}^{(k-1)}\right)$$
$$\vdots$$
$$x_n^{(k)} = g_n\left(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \ldots, x_{i-1}^{(k-1)}\right)$$

Ejemplo

Considere el sistema no lineal

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Aplique el método Iterativo de Seidel para resolverlo. ¿Acelera la convergencia el método Iterativo de Seidel?

Solución:

Utilizando las fórmulas

$$x_1^{(k)} = \frac{\left(x_1^{(k-1)}\right)^2 + \left(x_2^{(k-1)}\right)^2 + 8}{10}$$

$$x_2^{(k)} = \frac{\left(x_1^{(k)}\right)\left(x_2^{(k-1)}\right)^2 + \left(x_1^{k}\right) + 8}{10}$$

con $x^{(0)} = (0,0)$ y cota de error de 10^{-4} obtenemos los siguientes resultados:

k	$x_1^{(k)}$	$x_2^{(k)}$	$\|x^{(k)} - x^{(k-1)}\|_\infty$
0	0	0	_
1	0.8	0.88	0.88
2	0.94144	0.967049	0.14144
3	0.982149	0.990064	0.040709
4	0.994484	0.996930	0.012335
5	0.998287	0.999045	0.003803
6	0.999467	0.999703	0.001180
7	0.999834	0.999907	0.000367

El punto que se aproxima a la solución del sistema no lineal es $x = (0.999834,\ 0.999907)$. El método Iterativo de Seidel acelera la convergencia del método de Iteración de Punto Fijo.

Algoritmo del método Iterativo de Seidel

Para encontrar un punto fijo del sistema $x = G(x)$ dada una aproximación inicial $x^{(0)}$.

ENTRADA Numero n de ecuaciones y variables, aproximación inicial $x^{(0)} = x0 = (x0_1, x0_2, \cdots, x0_n)$, tolerancia tol, número máximo de iteraciones max .

SALIDA Solución aproximada $x = (x_1, x_2, \cdots, x_n)$ o mensaje que el número de iteraciones fue excedido.

Paso 1 Tomar $k = 1$

Paso 2 Mientras que $(k \leq max)$ seguir los pasos 3-6.

Paso 3 Tomar para $i = 1,2,\cdots,n$ $x = g_i(x0)$

$$x0_i = x_i$$

Paso 4 Si $\|x - x0\| < tol$ entonces SALIDA $(x_1, x_2, \cdots, x_n)$

PARAR.

Paso 5 Tomar $k = k + 1$

Paso 6 $x0 = x$

Paso 7 SALIDA (Número máximo de iteraciones excedido) PARAR.

En el anexo 2, se presenta el programa en MATLAB correspondiente al algoritmo del Método Iterativo de Seidel para sistemas no lineales.

Una desventaja del método iterativo de Seidel es que no siempre acelera la convergencia del método de Iteración de Punto Fijo.

2.3 Método de Newton

Teniendo en cuenta la deseabilidad de convergencia cuadrática para la sucesión de soluciones aproximadas de un sistema no lineal $F(x) = 0$, generada a partir de una aproximación inicial $x^{(0)}$ determinaremos a continuación un esquema de iteración funcional cuadrática conocido como método de Newton para sistemas no lineales.

Teorema 2.3

Supóngase que p es una solución de $G(x) = x$ para alguna función $G = (g_1, g_2, \cdots, g_n)$, de $\mathbb{R}^n$ en $\mathbb{R}^n$. Si existe un numero $\delta > 0$ con la propiedad de que

$i)$ $\partial g_i / \partial x_j$ es continua en $N_\delta = \{x \,/\, \|x - p\| < \delta\}$ para $i = 1,2,\cdots,n$.

$ii)$ $\partial^2 g_i(x) \,/\, (\partial x_j \, \partial x_k)$ es continua y $\left|\partial^2 g_i(x) \,/\, (\partial x_j \, \partial x_k)\right| \leq M$ para alguna constante M, siempre que $x \in N_\delta$ para cada $i = 1,2,\cdots,n$, $j = 1,2,\cdots,n$ y $k = 1,2,\cdots,n$.

$iii)$ $\partial g_i(p) / \partial x_j = 0$ para cada $i = 1,2,\cdots,n$ y $j = 1,2,\cdots,n$ entonces existe un numero $\hat{\delta} \leq \delta$ tal que la sucesión generada por $x^{(k)} = G(x^{(k-1)})$ converge cuadráticamente a p para cualquier $x^{(0)}$ siempre que $\|x^{(0)} - p\| < \hat{\delta}$. Además $\|x^{(k)} - p\|_\infty \leq \frac{n^2 M}{2} \|x^{(k-1)} - p\|_\infty^2$ para cada $k \geq 1$.

Supongamos que el sistema no lineal $F(x) = 0$ tiene una solución p tal que $\partial f_{ij}(p) \,/\, \partial x_j \neq 0$ con $i = 1,2,\cdots,n$ y $j = 1,2,\cdots,n$.

Consideremos el esquema de punto fijo $x^{(k)} = G\left(x^{(k-1)}\right)$ con $k \geq 1$ con G de la forma $G(x) = x - \Phi(x)F(x)$ donde $\Phi(x)$ es una matriz de funciones cuyos elementos $b_{ij}(x)$ especificaremos más adelante.

Como

$$G(x) = x - \Phi(x)F(x), \qquad g_i = x_i - \sum_{j=1}^{n} b_{ij}(x) f_j(x);$$

así que

$$\frac{\partial g_i(x)}{\partial x_k} = \begin{cases} 1 - \sum_{j=1}^{n} \left(b_{ij}(x)\partial f_j + \frac{\partial b_{ij}}{\partial x_k}(x) f_j(x) \right), & si \ \ i = k \\ - \sum_{j=1}^{n} \left(b_{ij}(x)\partial f_j + \frac{\partial b_{ij}}{\partial x_k}(x) f_j(x) \right), & si \ i \neq k \end{cases}$$

El teorema 2.3 implica que $\frac{\partial g_i}{\partial x_k}(p) = 0$ para cada $i = 1,2,\cdots,n$ y $k = 1,2,\cdots,n$. Esto significa que para $i = k$,

$$0 = 1 - \sum_{j=1}^{n} b_{ij}(p) \frac{\partial f_j}{\partial x_i}(p),$$

así que

$$\sum_{j=1}^{n} b_{ij}(p) \frac{\partial f_j}{\partial x_i}(p) = 1,$$

y cuando $k \neq i$,

$$0 = - \sum_{j=1}^{n} b_{ij}(p) \frac{\partial f_j}{\partial x_k}(p),$$

así que

$$\sum_{j=1}^{n} b_{ij}(p) \frac{\partial f_j}{\partial x_k}(p) = 0$$

Las dos condiciones anteriores requieren que $\Phi(p)\ J(p) = I$, de donde $\Phi(p) = J(p)^{-1}$.

Una elección apropiada para $\Phi(x)$ es $\Phi(x) = J(x)^{-1}$ ya que la condición (iii) del teorema 2.3 se satisface con esta condición.

La función G se define como $G(x) = x - J(x)^{-1}F(x)$, y el procedimiento de iteración funcional surge de seleccionar $x^{(0)}$ y de generar, para $k \geq 1$

$$x^{(k)} = G\left(x^{(k-1)}\right) = x^{(k-1)} - J\left(x^{(k-1)}\right)^{-1} F\left(x^{(k-1)}\right) \qquad (2.1)$$

En la práctica usamos un esquema de cálculo de dos pasos. Si en (2.1) hacemos $Y = -J(x)^{-1}$ se tiene $G(x) = x + Y$.

Calculando el vector Y como aquel que satisface el sistema $J\left(x^{(k)}\right)Y = -F\left(x^{(k)}\right)$, la nueva aproximación se obtiene mediante la expresión $x^{(k+1)} = x^{(k)} + Y$.

Ejemplo

Encuentre una solución al siguiente sistema no lineal usando el método de Newton, y calcule la cota de error dada en el teorema 2.3. Itere hasta que $\left\|x^{(i)} - x^{(i-1)}\right\|_\infty < 10^{-4}$

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Solución:

Tomando $x^{(0)} = (0,0)$ como aproximación inicial.

La matriz jacobiana $J(x)$ para este sistema está dada por

$$J(x) = \begin{bmatrix} 2x_1 - 10 & 2x_2 \\ x_2^2 + 1 & 2x_1x_2 - 10 \end{bmatrix}$$

$$\begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \end{bmatrix} = \begin{bmatrix} x_1^{(k-1)} \\ x_2^{(k-1)} \end{bmatrix} + \begin{bmatrix} Y_1^{(k-1)} \\ Y_1^{(k-1)} \end{bmatrix}$$

Por lo tanto en el k-ésimo paso, se debe resolver el sistema lineal

$$\begin{bmatrix} 2x_1^{(k-1)} - 10 & 2x_2^{(k-1)} \\ \left(x_2^{(k-1)}\right)^2 + 1 & 2\left(x_1^{(k-1)}\right)(x)_2^{(k-1)} - 10 \end{bmatrix} \begin{bmatrix} x_1^{(k)} - x_1^{(k-1)} \\ x_2^{(k)} - x_2^{(k-1)} \end{bmatrix}$$

$$= -\begin{bmatrix} \left(x_1^{(k-1)}\right)^2 - 10\left(x_1^{(k-1)}\right) + \left(x_2^{(k-1)}\right)^2 + 8 \\ \left(x_1^{(k-1)}\right)\left(x_2^{(k-1)}\right)^2 + x_1^{(k-1)} - 10\left(x_2^{(k-1)}\right) + 8 \end{bmatrix}$$

Los resultados se muestran en la siguiente tabla.

k	$x_1^{(k)}$	$x_2^{(k)}$	$\|x^{(k)} - x^{(k-1)}\|_\infty$
0	0	0	_
1	0.8	0.88	0.88
2	0.991787	0.991712	0.111712
3	0.999975	0.999969	0.008257
4	1.000000	1.000000	0.000031

El punto que se aproxima a la solución del sistema no lineal es $x = (1.000000, 1.000000)$.

La convergencia del método de Newton se hace muy rápida una vez que estamos cerca de p. Esto ilustra la convergencia cuadrática del método cerca de la solución.

Algoritmo del método de Newton para sistemas no lineales

Para encontrar un punto fijo del sistema $x = G(x)$ dada una aproximación inicial $x^{(0)}$.

ENTRADA Numero n de ecuaciones y variables, aproximación inicial $x^{(0)} = x0 = (x0_1, x0_2, \cdots, x0_n)$, tolerancia tol, número máximo de iteraciones max.

SALIDA Solución aproximada $x = (x_1, x_2, \cdots, x_n)$ o mensaje que el número de iteraciones fue excedido.

Paso 1 Tomar $k = 1$

Paso 2 Mientras que $(k \leq max)$ seguir los pasos 3-8.

Paso 3 Calcular $F(x0)$ y $J(x0)$, $J(x)_{i,j} = \left(\partial f_i(x0)/\partial x0_j\right)$ para

$$1 \leq i, j \leq n$$

Paso 4 Resolver el sistema lineal de $J(x0)Y = -F(x0)$

Paso 5 Tomar $x = x0 + Y$

Paso 6 Si $\|Y\| < tol$ entonces SALIDA (x);

(Procedimiento completado satisfactoriamente)

PARAR

Paso 7 Tomar $k = k + 1$

Paso 8 $x0 = x$

Paso 9 SALIDA (Número máximo de iteraciones excedido);

PARAR.

En el anexo 3, se presenta el programa en MATLAB correspondiente al algoritmo del método de newton para sistemas no lineales.

Una limitación importante del Método de Newton para resolver sistemas de ecuaciones no lineales es la necesidad de calcular la matriz jacobiana en cada iteración y resolver un sistema de nxn asociado a esta matriz. En la mayoría de los casos la evaluación exacta de las derivadas parciales es complicada y en muchas ocasiones imposible.

2.5 Métodos Cuasi-Newton

Los métodos Cuasi-Newton reemplazan a la matriz jacobiana en el método de Newton por una matriz de aproximación que se renueva en cada iteración. La desventaja de estos métodos consiste en que se pierde la convergencia cuadrática del método de Newton y se reemplaza por una convergencia superlineal.

Estudiaremos uno de los métodos Cuasi-Newton llamado el método de Broyden que es una generalización del método de la Secante a sistemas no lineales que busca sustituir la matriz jacobiana $J^{(k-1)}$ del método de Newton dado por $x^{(k)} = x^{(k-1)} - J\left(x^{(k-1)}\right)^{-1} F\left(x^{(k-1)}\right)$, $k \geq 1$ por una matriz A_k.

Supongamos que se da una aproximación inicial $x^{(0)}$ a la solución p del sistema no lineal $F(x) = 0$. Calculamos la siguiente aproximación $x^{(1)}$ de la misma manera que en el método de Newton. Sin embargo para calcular $x^{(2)}$ se reemplaza la matriz $J(x^{(1)})$ por una matriz A_1 con la propiedad de que

$$A_1(x^{(1)} - x^{(0)}) = F(x^{(1)}) - F(x^{(0)}) \qquad (2.2)$$

Además se pide que

$A_1 z = J(x^{(0)})z$ siempre que $(x^{(1)} - x^{(0)})^t z = 0$. (2.3)

Esta condición especifica que cualquier vector ortogonal a $x^{(1)} - x^{(0)}$ no será afectado por la renovación de $J(x^{(0)})$, que se utilizó para calcular $x^{(1)}$ a A_1, y que se usará en la determinación de $x^{(2)}$.

Las condiciones (2.2) y (2.3) definen únicamente a A_1, como

$$A_1 = J(x^{(0)}) + \frac{[F(x^{(1)}) - F(x^{(0)}) - J(x^{(0)})(x^{(1)} - x^{(0)})](x^{(1)} - x^{(0)})^t}{\|x^{(1)} - x^{(0)}\|_2^2}.$$

Se calcula $x^{(2)}$ mediante $x^{(2)} = x^{(1)} - A_1^{-1}F(x^{(1)})$.

Haciendo $Y_i = F(x^{(i)}) - F(x^{(i-1)})$ y $s_i = x^{(i)} - x^{(i-1)}$ podemos escribir la expresión que define A_i mediante

$A_i = A_{i-1} + \frac{(Y_i - A_{i-1}s_i)s_i^t}{\|s_i\|_2^2}$ y la expresión que define $x^{(i+1)}$ como $x^{(i+1)} = x^{(i)} - A_1^{-1}F(x^{(i)})$.

Para eliminar la necesidad de invertir una matriz en cada iteración usaremos una fórmula que permite calcular A_1^{-1} directamente de A_{i-1}^{-1}, esta es

$$A_i^{-1} = A_{i-1}^{-1} + \frac{(s_i - A_{i-1}^{-1}Y_i)s_i^t A_{i-1}^{-1}}{s_i^t A_{i-1}^{-1} Y_i}.$$

Ejemplo

Encuentre una solución al siguiente sistema no lineal usando el método de Broyden. Iterar hasta que $\|x^{(i)} - x^{(i-1)}\|_\infty < 10^{-4}$

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Solución:

Tomando $x^{(0)} = (0,0)^t$ como aproximación inicial.

La matriz jacobiana $J(x)$ para este sistema está dada por

$$J(x) = \begin{bmatrix} 2x_1 - 10 & 2x_2 \\ x_2^2 + 1 & 2x_1x_2 - 10 \end{bmatrix}$$

$$F(x^{(0)}) = \begin{bmatrix} 8 \\ 8 \end{bmatrix}$$

$$A_0 = J\left(x_1^{(0)}, x_2^{(0)}\right) = \begin{bmatrix} -10 & 0 \\ 1 & -10 \end{bmatrix}$$

$$A_0^{-1} = J\left(x_1^{(0)}, x_2^{(0)}\right)^{-1} = \begin{bmatrix} -0.1 & 0 \\ -0.01 & -0.1 \end{bmatrix}$$

$$x^{(1)} = x^{(0)} - A_0^{-1}F\left(x^{(0)}\right) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} -0.1 & 0 \\ -0.01 & -0.1 \end{bmatrix}\begin{bmatrix} 8 \\ 8 \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix}$$

$$F\left(x^{(1)}\right) = \begin{bmatrix} 1.4144 \\ 0.61952 \end{bmatrix}$$

$$y_1 = F\left(x^{(1)}\right) - F\left(x^{(0)}\right) = \begin{bmatrix} -6.5856 \\ -7.38048 \end{bmatrix}$$

$$s_1 = x^{(1)} - x^{(0)} = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix}$$

$$s_1^t A_0^{-1} y_1 = \begin{bmatrix} 0.8 & 0.88 \end{bmatrix}\begin{bmatrix} -0.1 & 0 \\ -0.01 & -0.1 \end{bmatrix}\begin{bmatrix} -6.5856 \\ -7.38048 \end{bmatrix} = 1.2331552$$

$$F\left(x^{(1)}\right) = \begin{bmatrix} 1.4144 \\ 0.61952 \end{bmatrix}$$

$$y_1 = F\left(x^{(1)}\right) - F\left(x^{(0)}\right) = \begin{bmatrix} -6.5856 \\ -7.38048 \end{bmatrix}$$

$$s_1 = x^{(1)} - x^{(0)} = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix}$$

$$s_1 - A_0^{-1} y_1 = \begin{bmatrix} 0.8 \\ 0.88 \end{bmatrix} - \begin{bmatrix} -0.1 & 0 \\ -0.01 & -0.1 \end{bmatrix}\begin{bmatrix} -6.5856 \\ -7.38048 \end{bmatrix} = \begin{bmatrix} 0.14144 \\ 0.076096 \end{bmatrix}$$

$$s_1^t A_0^{-1} y_1 = \begin{bmatrix} 0.8 & 0.88 \end{bmatrix}\begin{bmatrix} -0.1 & 0 \\ -0.01 & -0.1 \end{bmatrix}\begin{bmatrix} -6.5856 \\ -7.38048 \end{bmatrix} = 1.2331552$$

$$A_1^{-1} = A_0^{-1} + (1/s_1^t A_0^{-1} y_1)[(s_1 - A_0^{-1} y_1)s_1^t A_0^{-1}] = \begin{bmatrix} -0.1102 & -0.0101 \\ -0.0155 & -0.1054 \end{bmatrix}$$

$$x^{(2)} = x^{(1)} - A_1^{-1}F\left(x^{(1)}\right) = \begin{bmatrix} 0.962080 \\ 0.967201 \end{bmatrix}$$

Los resultados se muestran en la siguiente tabla

k	$x_1^{(k)}$	$x_2^{(k)}$	$\|x^{(k)} - x^{(k-1)}\|_\infty$
0	0	0	_
1	0.962080	0.967201	0.967201
2	0.997434	0.996786	0.029585
3	0.999904	0.999845	0.003059
4	0.999998	0.999997	0.000152

Un punto que se aproxima a la solución del sistema no lineal es $x = (0.999998, 0.999997)$.

Algoritmo de Broyden

Para encontrar un punto fijo del sistema $x = G(x)$ dada una aproximación inicial $x^{(0)}$.

ENTRADA Numero n de ecuaciones y variables, aproximación inicial $x^{(0)} = x0 = (x0_1, x0_2, \cdots, x0_n)$, tolerancia tol, número máximo de iteraciones max.

SALIDA Solución aproximada $x = (x_1, x_2, \cdots, x_n)$ o mensaje que el número de iteraciones fue excedido.

Paso 1 Tomar $A_0 = J(x0)$ donde $J(x)_{i,j} = \left(\partial f_i(x0)/\partial x0_j\right)$ para

$$1 \leq i, j \leq n;$$

$v = F(x0)$ (Nota: $v = F\left(x^{(0)}\right)$)

Paso 2 Tomar $A = A_0^{-1}$

Paso 3 Tomar $k = 1$;

$s = -Av$ (Nota: $s = s_1$)

$x = x0 + s$ (Nota: $x = x^{(1)}$)

Paso 4 Mientras que $(k \leq max)$ seguir los pasos 5-13.

Paso 5 Tomar $w = v$; (Guardar v)

$v = F(x)$; (Nota: $v = F$)

$Y = v - w$; (Nota: $Y = Y_k$)

Paso 6 Tomar $z = -AY$ (Nota: $z = -A_{k-1}^{-1} Y_k$)

Paso 7 Tomar $p = -s^t z$ (Nota: $p = s_k^t A_{k-1}^{-1} Y_k$)

Paso 8 Tomar $c = pI + (s + z)s^t$

(Nota: $c = s_k^t A_{k-1}^{-1} Y_k I + (s_k + A_{k-1}^{-1} Y_k) s_k^t$)

Paso 9 Tomar $A = (1/p)cA$ (Nota: $A = A_k^{-1}$)

Paso 10 Tomar $s = -Av$ (Nota: $s = -A_k^{-1} F\left(x^{(k)}\right)$)

Paso 11 Tomar $x = x + s$ (Nota: $x = x^{(k+1)}$)

Paso 12 Si $\|s\| < tol$ entonces SALIDA (x);

(Procedimiento completado

Satisfactoriamente)

PARAR.

Paso 13 Tomar $k = k + 1$

Paso 14 Tomar $x0 = x$

Paso14 SALIDA (Número máximo de iteraciones excedido); PARAR

En el anexo 4, se presenta el programa en MATLAB correspondiente al algoritmo del método Broyden para sistemas no lineales.

2.6 Método del Descenso más rápido.

Notemos que un sistema de ecuaciones no lineales

$$f_1(x_1, x_2, \cdots, x_n) = 0$$

$$f_2(x_1, x_2, \cdots, x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2, \cdots, x_n) = 0$$

tiene una solución en $x = (x_1, x_2, \cdots, x_n)$ precisamente cuando la función G definida como $G(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} [f_i(x_1, x_2, \ldots, x_n)]^2$ tiene el valor mínimo cero.

El método del Descenso más rápido permite encontrar un mínimo local de una función G de $\mathbb{R}^n$ en $\mathbb{R}$ y podemos emplear este método en la aproximación de la solución del sistema

$$f_1(x_1, x_2, \cdots, x_n) = 0$$

$$f_2(x_1, x_2, \cdots, x_n) = 0$$

$$\vdots$$

$$f_n(x_1, x_2, \cdots, x_n) = 0$$

con solo reemplazar la función G por $\sum_{i=1}^{n} f_i^2$.

Para encontrar un mínimo local de G, evaluamos G en una aproximación inicial $x^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}\right)$. A partir de $x^{(0)}$ se encuentra una dirección que resulte en un descenso en el valor de G. Por lo estudiado en la sección (1.4), la dirección en la que el valor de G en x decrece más es aquella dada por $-\nabla G(x)$. Una elección apropiada para x es $x^{(1)} = x^{(0)} - \alpha \nabla G\left(x^{(0)}\right)$ para alguna constante $\alpha > 0$.

El problema se reduce a escoger α tal que $G\left(x^{(1)}\right)$ sea significativamente menor que $G\left(x^{(0)}\right)$. Para ello consideramos la función de una variable $h(\alpha) = G\left(x^{(0)} - \alpha \nabla G\left(x^{(0)}\right)\right)$.

El valor de α que minimiza a h es el valor de α a escoger. Se halla α de manera indirecta mediante el polinomio cuadrático que interpola a h en $\alpha_1, \alpha_2, \alpha_3$ (aproximaciones no negativas de α), luego se encuentra el número que minimiza este polinomio cuadrático y se define como α.

Se usa este valor de α para determinar la nueva iteración para aproximar el valor mínimo de G. La fórmula deducida es

$$x^{(k)} = x^{(k-1)} - \nabla G\left(x^{(k-1)}\right).$$

Algoritmo del Descenso más rápido

Para aproximar una solución p al problema de minimización

$$G(p) = \min_{x \in \mathbb{R}^n} G(x)$$

dada una aproximación inicial $x0$:

ENTRADA Numero n de ecuaciones y variables, aproximación inicial $x0 = (x0_1, x0_2, \cdots, x0_n)$, tolerancia tol, número máximo de iteraciones max.

SALIDA Solución aproximada $x = (x_1, x_2, \cdots, x_n)$ o mensaje de fracaso.

Paso 1 Hacer $k = 1$

Paso 2 Mientras que $(k \leq max)$ llevar a cabo los pasos 3-15.

Paso 3 Sea $g_1 = G(x0_1, x0_2, \ldots, x0_n)$; (Nota:$g_1 = G\left(x^{(k)}\right)$)

$z = \nabla G(x0_1, x0_2, \ldots, x0_n)$; (Nota: $z = \nabla G\left(x^{(k)}\right)$

$z_0 = \|z\|_2$

Paso 4 Si $z_0 = 0$ entonces SALIDA (´GRADIENTE CERO´);

SALIDA $(x0_1, x0_2, \ldots, x0_n, g_1)$;

(Procedimiento terminado, puede tener un mínimo. Verificar más aún)

PARAR.

Paso 5 Sea $z = z/z_0$; (hacer a z un vector unitario)

$\alpha_1 = 0$;

$\alpha_3 = 1$;

$g_3 = G(x0 - \alpha_3 z)$

Paso 6 Mientras que $(|g_3| \geq |g_1|)$ realizar pasos 7 y 8.

Paso 7 Hacer $\alpha_3 = \alpha_3/2$;

$$g_3 = G(x0 - \alpha_3 z)$$

Paso 8 Si $\alpha_3 = tol/2$ entonces

SALIDA (´NO HUBO MEJORA´);

SALIDA $(x0_1, x0_2, \ldots, x0_n, g_1)$;

(Procedimiento terminado, puede tener un mínimo. Verificar más aún).

PARAR.

Paso 9 Hacer $\alpha_2 = \alpha_3/2$;

$$g_2 = G(x0 - \alpha_2 z)$$

Paso 10 Sea $h_1 = (g_2 - g_1)/(\alpha_2 - \alpha_1)$;

$$h_2 = (g_3 - g_2)/(\alpha_3 - \alpha_2);$$

$$h_3 = (h_2 - h_1)/(\alpha_3 - \alpha_1).$$

(Nota: el polinomio cuadrático $p(\alpha) = g_1 + h_1\alpha + h_3\alpha(\alpha - \alpha_2)$ interpola a $h(\alpha)$ a $\alpha = 0, \alpha = \alpha_2, \alpha = \alpha_3$)

Paso 11 Sea $\alpha_0 = 0.5(\alpha_1 + \alpha_2 - h_1/h_3)$;(En α_0 hay un punto crítico de p)

$$g_0 = G(x0 - \alpha_0 z).$$

Paso 12 Encontrar α de $\{\alpha_0, \alpha_1, \alpha_2, \alpha_3\}$ tal que

$$g = G(x0 - \alpha z) \ \ min\{g_0, g_1, g_2, g_3\}.$$

Paso 13 Hacer $x = x0 - \alpha z$.

Paso 14 Si $|g - g_1| < tol$ entonces

SALIDA $(x_1, x_2, \cdots, x_n, g_1)$;

(Procedimiento terminado con éxito).PARAR.

Paso 15 Hacer $k = k + 1$

Paso 16 Hacer $x0 = x$

Paso 17 SALIDA (´Número máximo de iteraciones excedido´);

(Procedimiento termina sin éxito). PARAR.

En el anexo 5, se presenta el programa en MATLAB correspondiente al algoritmo del método del descenso más rápido para sistemas no lineales.

Ejemplo

Encuentre una solución al siguiente sistema no lineal usando el método del descenso más rápido. Iterar hasta que $\left\|x^{(i)} - x^{(i-1)}\right\|_\infty < 10^{-4}$

$$f_1(x_1, x_2) = x_1^2 - 10x_1 + x_2^2 + 8 = 0$$
$$f_2(x_1, x_2) = x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Solución:

Sea

$$g(x_1, x_2) = [f_1(x, y)]^2 + [f_2(x, y)]^2$$
$$g(x_1, x_2) = [x_1^2 - 10x_1 + x_2^2 + 8]^2 + [x_1 x_2^2 + x_1 - 10x_2 + 8]^2$$

Entonces

$$\nabla g(x_1, x_2) = \nabla g(X) =$$

$$\left(2f_1(X)\frac{\partial f_1(x)}{\partial X_1} + 2f_2(X)\frac{\partial f_2(x)}{\partial X_1}\ ;\ 2f_1(X)\frac{\partial f_1(x)}{\partial X_2} + 2f_2(X)\frac{\partial f_2(x)}{\partial X_2}\right) =$$

$$\nabla g(x_1, x_2) = 2J(X)^t F(X)$$

Con $x^{(0)} = (0,0)$, tenemos

$$g\left(x^{(0)}\right) = 128 \qquad \text{y} \quad z_0 = \left\|\nabla g\left(x^{(0)}\right)\right\|_2 = 215.258$$

Sea $$z = \frac{1}{z_0}\nabla g\left(x^{(0)}\right) = (-0.66896464, -0.743294094)^t$$

Para $\alpha_1 = 0$, tenemos $g_1 = g\left(x^{(0)} - \alpha_1 z\right) = g\left(x^{(0)}\right) = 128$. De manera arbitraria, hacemos $\alpha_3 = 1$ de modo que

$$g_3 = g\left(x^{(0)} - \alpha_3 z\right) = 7.916$$

Como $g_3 < g_1$, aceptamos α_3 y hacemos $\alpha_2 = 0.5$. Así,

$$g_2 = g\left(x^{(0)} - \alpha_2 z\right) = 45.816.$$

Ahora contruimos el polinomio de interpolación de Newton con diferencias divididas hacia adelante

$$P(\alpha) = g_1 + h_1\alpha + h_3\alpha(\alpha - \alpha_2)$$

que interpola a

$$g = g\left(x^{(0)} - \alpha\nabla g\left(x^{(0)}\right)\right) = g\left(x^{(0)} - \alpha z\right)$$

con $\alpha_1 = 0, \alpha_2 = 0.5, \alpha_3 = 1$ obtenemos:

$$\alpha_1 = 0, \quad g_1 = 128$$

$$\alpha_2 = 0.5, \quad g_2 = 45.816, \quad h_1 = \frac{g_2 - g_1}{\alpha_2 - \alpha_1} = -164.369$$

$\alpha_3 = 1, \quad g_3 = 7.916, \quad h_2 = \frac{g_3 - g_2}{\alpha_3 - \alpha_2} = -75.800, \quad h_3 = \frac{h_2 - h_1}{\alpha_3 - \alpha_1} = 88.569$

Por tanto,

$P(\alpha) = 128 - 164.369\alpha + 88.569\alpha(\alpha - 0.5)$

y el punto crítico de P ocurre en $\alpha_0 = 0.5(\alpha_2 - h_1/h_3)$=1.178,

entonces $g_0 = g\left(x^{(0)} - \alpha_0 z\right) = 2.678$. Como g_0 es menor que g_1 y g_3, hacemos

$x^{(1)} = x^{(0)} - \alpha_0 z = (0.787983, 0.875537)^t$

La tabla siguiente contiene el resto de los resultados

k	$x_1^{(k)}$	$x_2^{(k)}$	g
0	0	0	_
1	0.787983	0.875536	2.678354
2	0.982985	0.923515	0.343749
3	0.969119	0.980500	0.053020
4	0.996770	0.987236	0.008613
5	0.994533	0.996515	0.001644
6	0.999406	0.997690	0.000299
7	0.999002	0.999362	0.000055

Un punto que se aproxima a la solución del sistema no lineal es $x = (0.999002, 0.999362)$

Una solución dada por el método del Descenso más rápido después de pocas iteraciones puede servir como aproximación inicial para los métodos de Newton y Broyden, ya que el método del Descenso más rápido es linealmente convergente y converge independientemente de la aproximación inicial aunque más lentamente que los métodos mencionados.

2.7 Métodos de Homotopía y Continuación

En los métodos de Homotopía y Continuación un problema dado $F(x) = 0$ está integrado en una familia de problemas de un parámetro λ con valores en $[0,1]$. El problema original con solución desconocida $x^{(1)}$ corresponde a $\lambda = 1$ y un problema con una solución conocida $x^{(0)}$ corresponde a $\lambda = 0$.

Sea la función $G: [0,1] \times \mathbb{R}^n \to \mathbb{R}^n$ definida por $G(\lambda, x) = F(x) + (\lambda - 1)F\left(x^{(0)}\right)$

donde $x^{(0)}$ es una aproximación inicial de la solución de $F\left(x^{(1)}\right) = 0$.

Observe que para determinar una solución de $G(\lambda, x) = 0$, cuando $\lambda = 0$ la ecuación asume la forma $G(0, x) = F(x) - F\left(x^{(0)}\right) = 0$ y $x^{(0)}$ es una solución.

Cuando $\lambda = 1$, la ecuación tiene la forma $G(1, x) = F(x) = 0$ y $x^{(1)}$ es la solución.

La función G, con el parámetro λ, nos proporciona una familia de funciones que nos pueden conducir del valor conocido $x^{(0)}$ a la solución $x^{(1)}$. La función G se llama una homotopía entre la función $G(0, x) = F(x) - F\left(x^{(0)}\right)$ y la función $G(1, x) = F(x)$.

Mediante la Continuación se determina la forma de pasar de la solución conocida $x^{(0)}$ de $G(0,x)=0$ a la solución desconocida $x^{(1)}$ de $G(1,x)=0$ que resuelva $F(x)=0$. Para ello supongamos que $x(\lambda)$ es la única solución de la ecuación $G(\lambda,x)=0$ para cada $\lambda \in [0,1]$. Si $x(\lambda)$ y G son diferenciables entonces al derivar la ecuación $G(\lambda,x)=0$ con respecto λ a se obtiene

$$0=\frac{\partial G(\lambda,x(\lambda))}{\partial \lambda}+\frac{\partial G(\lambda,x(\lambda))}{\partial x}x'(\lambda),$$

y al despejar $x'(\lambda)$ se tiene

$$x'(\lambda)=-\left[\frac{\partial G(\lambda,x(\lambda))}{\partial x}\right]^{-1}\frac{\partial G(\lambda,x(\lambda))}{\partial \lambda}.$$

Este es un sistema de ecuaciones con la condición inicial $x^{(0)}$.

Como $G(\lambda,x(\lambda)=F\left(x^{(\lambda)}\right)+(\lambda-1)F\left(x^{(0)}\right)$, podemos determinar

$$\frac{\partial G}{\partial x}(\lambda,x(\lambda))=\begin{bmatrix}\frac{\partial f_1(x(\lambda))}{\partial x_1} & \frac{\partial f_1(x(\lambda))}{\partial x_2} & \cdots & \frac{\partial f_1(x(\lambda))}{\partial x_n}\\ \frac{\partial f_2(x(\lambda))}{\partial x_1} & \frac{\partial f_2(x(\lambda))}{\partial x_2} & \cdots & \frac{\partial f_2(x(\lambda))}{\partial x_n}\\ \vdots & \vdots & & \\ \frac{\partial f_n(x(\lambda))}{\partial x_1} & \frac{\partial f_n(x(\lambda))}{\partial x_2} & \cdots & \frac{\partial f_n(x(\lambda))}{\partial x_n}\end{bmatrix}=J(x(\lambda)),$$

y $\frac{\partial G(\lambda,x(\lambda))}{\partial \lambda}=F\left(x^{(0)}\right).$

Por tanto, el sistema de ecuaciones diferenciales se convierte en $x'(\lambda)=-\left[J(x(\lambda))\right]^{-1}F\left(x^{(0)}\right)$, para $0\leq\lambda\leq 1$, con la condición inicial $x^{(0)}$.

Teorema 2.4

Sea $F(x)$ una función continuamente diferenciable para $x \in \mathbb{R}^n$. Suponga que la matriz jacobiana $J(x)$ es no singular para cada $x \in \mathbb{R}^n$ y que existe una constante M tal que $\|J(x)^{-1}\| \leq M$, para cada $x \in \mathbb{R}^n$. Entonces, para cualquier $x^{(0)}$ en $\mathbb{R}^n$, existe una única función $x(\lambda)$, tal que $G(\lambda, x(\lambda)) = 0$ para cada λ en $[0,1]$. Además, $x(\lambda)$ es continuamente diferenciable y $x'(\lambda) = -J(x(\lambda))^{-1} F(x^{(0)})$, para cada $\lambda \in [0,1]$.

Para resolver un sistema no lineal de ecuaciones $F(x) = 0$ mediante un algoritmo de continuación, le asociaremos un sistema de ecuaciones diferenciales cuya solución se conoce al aplicar un método dado (método de Runge-Kutta).

Ejemplo

Considere el sistema no lineal

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Encuentre un sistema de ecuaciones diferenciales asociado al mismo.

Solución:

La matriz jacobiana es

$$J(x) = \begin{bmatrix} 2x_1 - 10 & 2x_2 \\ x_2^2 + 1 & 2x_1 x_2 - 10 \end{bmatrix}$$

Sea $x^{(0)} = (0,0)^t$, de modo que

$$F\left(x^{(0)}\right) = \begin{bmatrix} 8 \\ 8 \end{bmatrix}$$

El sistema de ecuaciones diferenciales es

$$\begin{bmatrix} x_1'(\lambda) \\ x_2'(\lambda) \end{bmatrix} = -\begin{bmatrix} 2x_1 - 10 & 2x_2 \\ x_2^2 + 1 & 2x_1x_2 - 10 \end{bmatrix}^{-1} \begin{bmatrix} 8 \\ 8 \end{bmatrix}$$

Caracterizaremos a continuación el sistema de ecuaciones diferenciales a resolver mediante el método de Runge-Kutta.

El sistema de ecuaciones diferenciales $x'(\lambda) = -J\left(x(\lambda)\right)^{-1} F\left(x^{(0)}\right)$, para $0 \leq \lambda \leq 1$, con la condición inicial $x^{(0)}$ tiene la forma

$$\frac{dx_1}{d\lambda} = \phi_1(\lambda, x_1, x_2, \dots, x_n),$$

$$\frac{dx_2}{d\lambda} = \phi_2(\lambda, x_1, x_2, \dots, x_n),$$

$$\vdots$$

$$\frac{dx_n}{d\lambda} = \phi_n(\lambda, x_1, x_2, \dots, x_n),$$

donde

$$\begin{bmatrix} \phi_1(\lambda, x_1, x_2, \dots, x_n) \\ \phi_2(\lambda, x_1, x_2, \dots, x_n) \\ \vdots \\ \phi_n(\lambda, x_1, x_2, \dots, x_n) \end{bmatrix} = -J(x_1, x_2, \dots, x_n)^{-1} \begin{bmatrix} f_1\left(x^{(0)}\right) \\ f_2\left(x^{(0)}\right) \\ \vdots \\ f_n\left(x^{(0)}\right) \end{bmatrix}$$

y para resolverlo mediante el método de Runge-Kutta descrito en la sección (1.6.3) elegimos un entero $N > 0$ y hacemos $h = (1-0)/N$. Dividimos el intervalo $[0,1]$ en N subintervalos con los puntos de red $\lambda_j = jh$, para cada $j = 0,1,2,\cdots,N$.

Con $w_{i,j}$ denotamos una aproximación a $x_i(\lambda j)$ para $j = 0,1,2,\cdots,N$ e $i = 1,2,\cdots,n$.

Para las condiciones iniciales tomamos $w_{i,0} = x_i^{(0)}, i = 1,2,\cdots,n$ y calcularemos los valores $w_{i,j+1}$ para cada $j = 0,1,2,\cdots,N$ e $i = 1,2,\cdots,n$ usando los valores $w_{i,j}$ y las ecuaciones siguientes

$$k_{1,i} = h\phi_i\left(\lambda_j, w_{1,j}, w_{2,j}, \dots, w_{m,j}\right) para\ cada\ i = 1,2,\dots,n\ ;$$

$$k_{2,i} = h\phi_i\left(\lambda_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{1,i}, w_{2,j} + \frac{1}{2}k_{1,2}, \dots, w_{n,j} + \frac{1}{2}k_{1,n}\right) para\ cada\ i = 1,2,\dots,n;$$

$$k_{3,i} = h\phi_i\left(\lambda_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{2,1}, w_{2,j} + \frac{1}{2}k_{2,2}, \dots, w_{n,j} + \frac{1}{2}k_{2,n}\right) para\ cada\ i = 1,2,\dots,n;$$

$$k_{4,i} = h\phi_i\left(\lambda_j + h, w_{1,j} + k_{3,1}, w_{2,j} + k_{3,2}, \dots, w_{n,j} + k_{3,n}\right) para\ cada\ i = 1,2,\dots,n;$$

y por último

$$w_{i,j+1} = w_{i,j} + \frac{1}{6}\left[k_{1,i} + 2k_{2,i} + 3k_{3,i} + k_{4,i}\right] para\ cada\ i = 1,2,\dots,n.$$

En forma abreviada $x^{(\lambda_0)} = x^{(0)} = w_0$ y para cada $j = 0,1,2,\cdots,n$

$x^{(\lambda_{j+1})} = x^{(\lambda_j)} + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4] = w_j + \frac{1}{6}[k_1 + 2k_2 + 2k_3 + k_4]$ donde

$$k_1 = h[-J(w_j)]^{-1} F(x^{(0)});$$

$$k_2 = h\left[-J\left(w_j + \frac{1}{2}k_1\right)\right]^{-1} F(x^{(0)});$$

$$k_3 = h\left[-J\left(w_j + \frac{1}{2}k_2\right)\right]^{-1} F(x^{(0)});$$

$$k_4 = h[-J(w_j + k_3)]^{-1} F(x^{(0)}).$$

Observe que $x^{(\lambda_n)} = x^{(1)}$ es nuestra aproximación a la solución del sistema de ecuaciones diferenciales.

Algoritmo de Continuación

Para aproximar la solución del sistema no lineal $F(x) = 0$ dada una aproximación inicial $x0$:

ENTRADA numero n de ecuaciones y variables; entero $N > 0$; aproximación inicial $x0 = (x0_1, x0_2, \dots, x0_n)^t$.

SALIDA solución aproximada $x = (x_1, x_2, \dots, x_n)^t$

Paso 1 Tome $h = 1/N$;

$$-b = F(x0)$$

Paso 2 Para i= $1,2,\cdots,N$ haga los pasos 3-8.

Paso 3 Tome $A = J(x0)$;

Resuelva el sistema lineal $Ak_1 = b$.

Paso 4 Tome $A = J(x0 + 1/2k_1)$;

Resuelva el sistema lineal $Ak_2 = b$.

Paso 5 Tome $A = J(x0 + 1/2k_2)$;

Resuelva el sistema lineal $Ak_3 = b$.

Paso 6 Tome $A = J(x0 + k_3)$;

Resuelva el sistema lineal $Ak_4 = b$.

Paso 7 Tome $x = x0 + (k_1 + 2k_2 + 2k_3 + k_4)/6$.

Paso 8 Tome $x0 = x$.

Paso 9 SALIDA $(x_1, x_2, \ldots, x_n)$;

PARAR

En el anexo 6, se presenta el programa en MATLAB correspondiente al algoritmo del método de Homotopía y Continuación para sistemas no lineales.

El método de Homotopía y Continuación se puede utilizar cuando se tiene una no muy buena aproximación o bien para obtener una aproximación inicial para cualquiera de los métodos más eficientes (Newton y Broyden), ya que estos requieren menos cálculos

Ejemplo

Encuentre una aproximación a la solución del siguiente sistema no lineal usando el método de Homotopía y Continuación con $x^{(0)} = (0,0)^t$.

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

Solución:

La matriz jacobiana es

$$J(x) = \begin{bmatrix} 2x_1 - 10 & 2x_2 \\ x_2^2 + 1 & 2x_1 x_2 - 10 \end{bmatrix}$$

Sea $x^{(0)} = (0,0)^t$, de modo que

$$F\left(x^{(0)}\right) = \begin{bmatrix} 8 \\ 8 \end{bmatrix}$$

Con N=4 y h=0.25, tenemos

$$k_1 = h\left[-J\left(x^{(0)}\right)\right]^{-1} F\left(x^{(0)}\right) = 0.25 \begin{bmatrix} 0.1 & 0 \\ 0.01 & 0.1 \end{bmatrix} \begin{bmatrix} 8 \\ 8 \end{bmatrix} = \begin{bmatrix} 0.2 \\ 0.22 \end{bmatrix}$$

$$k_2 = h\left[-J\left(x^{(0)} + \frac{1}{2}k_1\right)\right]^{-1} F\left(x^{(0)}\right)$$

$$k_3 = h\left[-J\left(x^{(0)} + \frac{1}{2}k_2\right)\right]^{-1} F\left(x^{(0)}\right)$$

$$k_4 = h\left[-J\left(x^{(0)} + k_3\right)\right]^{-1} F\left(x^{(0)}\right)$$

$$x^{(1)} = x^{(0)} + (k_1 + 2k_2 + 2k_3 + k_4)/6$$

Los resultados se muestran en la siguiente tabla

N	$x_1^{(k)}$	$x_2^{(k)}$
1	0.209308	0.221962
2	0.439871	0.453015
3	0.698412	0.704510
4	1.000013	1.000027

Un punto que se aproxima a la solución del sistema no lineal es $x = (1.000013, 1.000027)$

2.8 Análisis de los Métodos

Comparando los resultados obtenidos en los diversos métodos para el sistema de ecuaciones no lineales

$$x_1^2 - 10x_1 + x_2^2 + 8 = 0$$

$$x_1 x_2^2 + x_1 - 10x_2 + 8 = 0$$

considerando como aproximación inicial $x^{(0)} = (0,0)$ y tolerancia de 0.0001, observamos que:

Método	Solución Aproximada	No. de Iteraciones
Iteración de Punto fijo	(0.999731 , 0.999731)	8
Iterativo de Seidel	(0.999833 , 0.999907)	7
Newton	(1.000000 , 1.000000)	4
Broyden	(1.000000 , 1.000000)	4
Descenso más rápido	(0.999891 , 0.999577)	7
Homotopía y Continuación	(1.000013 , 1.000027)	4

El método de Iteración de Punto Fijo necesita de 8 iteraciones para tener una buena aproximación mientras que el método Iterativo de Seidel necesita de 7 iteraciones.

Los métodos más efectivos son los métodos de Newton y Broyden ya que se realizan 4 iteraciones, siempre que tengamos una buena aproximación inicial.

El método del Descenso más rápido realiza 7 iteraciones para tener una aproximación a la solución, este método puede encontrar una aproximación que sirva como valor inicial para Newton o Broyden ya que no requiere de una buena aproximación inicial.

El Método de Homotopía y Continuación con N=4 da una buena aproximación a la solución, aunque emplea más cálculos para encontrar esta aproximación.

CAPÍTULO 3

APLICACIONES

3.1 Introducción

En este capítulo se presentan algunas aplicaciones de la solución numérica de sistemas de ecuaciones no lineales.

Revisaremos cuatro tipos de aplicaciones: El problema de encontrar los puntos de intersección de dos curvas en el plano xy, el problema de determinar los ceros comunes de varias funciones no lineales, el problema de encontrar un mínimo de una función de varias variables y el problema de determinar el punto de equilibrio de dos o más fenómenos dados.

3.2 Puntos de intersección de dos curvas en el plano xy

Caso 1:

Dadas las curvas en el plano xy.

$x^2 - y - 0.2 = 0 \qquad (parábola)$

$y^2 - x - 0.3 = 0 \qquad (parábola)$

Encontrar un punto donde ambas curvas se cruzan.

Solución:

Aplicando el método de Iteración de Punto Fijo con el esquema

$x = \sqrt{y + 0.2}$

$y = \sqrt{x + 0.3}$.

tomaremos $D = \{(x,y)/-2 \leq x,y \leq 2\}$.

Para demostrar la existencia de un único punto fijo se tiene que

$$|g_1(x,y)| = \left|\sqrt{y+0.2}\right| \leq 1.48$$

$$|g_2(x,y)| = \left|\sqrt{x+0.3}\right| \leq 1.51$$

así que $-2 \leq g_i(x,y) \leq 2$, por lo tanto $G(x) \in D$ siempre que $x \in D$

Para las cotas de las derivadas parciales en D se obtiene:

$$\left|\frac{\partial g_1}{\partial x}\right| = 0, \left|\frac{\partial g_2}{\partial y}\right| = 0,$$

mientras que

$$\left|\frac{\partial g_1}{\partial y}\right| = \frac{1}{2\sqrt{y+0.2}} \leq 0.33, \left|\frac{\partial g_2}{\partial x}\right| = \frac{1}{2\sqrt{x+0.3}} \leq 0.32,$$

como las derivadas parciales de g_1, g_2 están acotadas en D, entonces por el teorema 1.11 implica que estas funciones son continuas en D . Luego G es continua en D . Por lo tanto G tiene un punto fijo en D.

Además, para toda $x \in D$ $\left|\frac{\partial g_i}{\partial x_j}(x)\right| \leq 0.33$ $siempre\ que\ x \in D,$ $para\ cada\ i = 1,2$ y $j = 1,2$. Por tanto el valor que satisface a K es 0.66.

Se obtienen los resultados aproximados corriendo los programas correspondientes a los métodos de Iteración de Punto fijo e Iterativo de Seidel, con aproximación inicial $x^{(0)} = (1.2,1.2)$ y tolerancia 0.0001 las tablas siguientes:

Punto Fijo

k	x(1)	x(2)	err
1	1.183216	1.224745	0.029900
2	1.193627	1.217874	0.012474
3	1.190745	1.222140	0.005149
4	1.192535	1.220961	0.002144
5	1.192041	1.221694	0.000884
6	1.192348	1.221491	0.000368
7	1.192263	1.221617	0.000152
8	1.192316	1.221582	0.000063

Seidel

k	x(1)	x(2)	err
1	1.183216	1.217874	0.024519
2	1.190745	1.220961	0.008137
3	1.192041	1.221491	0.001400
4	1.192263	1.221582	0.000240
5	1.192301	1.221598	0.000041

Por tanto un punto de intersección de las curvas dadas es (1.192316,1.221582) para el método de Iteración de Punto fijo y (1.192301,1.221598) para el método Iterativo de Seidel. Se puede notar que para el método de Iteración de Punto Fijo se necesitan 8 iteraciones mientras que para el método Iterativo de Seidel se realizan 5 iteraciones. Se tiene más precisión con menos iteraciones con el método Iterativo de Seidel.

3.3 Ceros comunes de varias funciones no lineales

Caso 1:

Dadas las funciones no lineales

$$f(x,y) = x^2 + 4y^2 - 16$$

$$g(x,y) = xy^2 - 4$$

Encuentre un cero común a ambas funciones.

Solución:

Supongamos que el sistema no lineal

$$f(x,y) = x^2 + 4y^2 - 16 = 0$$

$$g(x,y) = xy^2 - 4 = 0$$

tiene una solución p en $D = \{(x,y): 0.5 \leq x, y \leq 2\}$.

Para las derivadas parciales de ambas funciones se tiene que:

$$\left|\frac{\partial f}{\partial x}\right| = 2x \neq 0, \left|\frac{\partial f}{\partial y}\right| = 8y \neq 0, \left|\frac{\partial g}{\partial x}\right| = y^2 \neq 0, \left|\frac{\partial g}{\partial y}\right| = 2xy \neq 0.$$

Corriendo los programas correspondientes a los métodos de Newton y Broyden, con aproximación inicial $x^{(0)} = (1,1)$ y tolerancia 0.0001 se obtienen las tablas siguientes:

Newton

k	x(1)	x(2)	err
1	1.500000	2.250000	1.346291
2	1.206349	1.937831	0.428580
3	1.078687	1.926993	0.128121
4	1.078378	1.925948	0.001089
5	1.078378	1.925948	0.000000

Broyden

k	x(1)	x(2)	err
1	-0.501095	2.337349	2.003001
2	1.232695	1.750629	1.830374
3	1.135054	1.918452	0.194160
4	1.099978	1.916523	0.035129
5	1.071322	1.929037	0.031270
6	1.078472	1.925917	0.007802
7	1.078382	1.925946	0.000094

Por tanto el cero común que satisface a ambas funciones es (1.078378,1.925948) al aplicar el método de Newton y (1.078382,1.925946) al aplicar el método de Broyden.
Se puede notar que para el método de Newton se necesitan 5 iteraciones mientras que para el método de Broyden se realizan 7 iteraciones. Se tiene más precisión con menos iteraciones con el método de Newton.

Caso 2:

Dadas las funciones no lineales

$$g_1(x,y,z) = x^2 + y^2 + z^2 + 10x - 4$$

$$g_2(x,y,z) = x^2 - y^2 + z^2 + 10y - 5$$

$$g_3(x,y,z) = x^2 + y^2 - z^2 + 10z - 6$$

Encuentre un cero común a las tres funciones.

Solución:

Supongamos el sistema no lineal

$$g_1(x,y,z) = x^2 + y^2 + z^2 + 10x - 4 = 0$$

$$g_2(x,y,z) = x^2 - y^2 + z^2 + 10y - 5 = 0$$

$$g_3(x,y,z) = x^2 + y^2 - z^2 + 10z - 6 = 0$$.

Tomamos $D = \{(x,y.z): 0.3 \leq x,y,z \leq 1\}$.

Para las derivadas parciales tenemos:

$$\left|\frac{\partial g_1}{\partial x}\right| = 2x + 10 \neq 0, \left|\frac{\partial g_1}{\partial y}\right| = 2y \neq 0, \left|\frac{\partial g_1}{\partial z}\right| = 2z \neq 0, \left|\frac{\partial g_2}{\partial x}\right| = 2x \neq 0,$$

$$\left|\frac{\partial g_2}{\partial y}\right| = -2y + 10 \neq 0, \left|\frac{\partial g_2}{\partial z}\right| = 2z \neq 0, \left|\frac{\partial g_3}{\partial x}\right| = 2x \neq 0, \left|\frac{\partial g_3}{\partial y}\right| = 2y \neq 0,$$

$$\left|\frac{\partial g_3}{\partial z}\right| = -2z + 10 \neq 0.$$

Usando como aproximación inicial $x^{(0)} = (0,0,0)$ y tolerancia 0.0001 y corriendo los programas correspondientes a los métodos de Newton y Broyden, se obtienen las tablas siguientes:

Newton

k	x(1)	x(2)	x(3)	err
1	0.400000	0.500000	0.600000	0.877496
2	0.330575	0.475719	0.603389	0.073626
3	0.330161	0.475353	0.602846	0.000775
4	0.330161	0.475353	0.602846	0.000000

Broyden

k	x(1)	x(2)	x(3)	err
1	0.327456	0.474563	0.595289	0.077018
2	0.330539	0.476066	0.601814	0.007371
3	0.330179	0.475487	0.602730	0.001141
4	0.330159	0.475360	0.602842	0.000171
5	0.330161	0.475353	0.602845	0.000008

Por tanto el cero común que satisface a las funciones es (0.330161, 0.475353,0.602846) aplicando el método de Newton y (0.330161, 0.475353,0.602845) aplicando el método de Broyden. Se puede notar que para el método de Newton se necesitan 4 iteraciones mientras que para el método de Broyden se realizan 5 iteraciones. Se tiene más precisión con menos iteraciones con el método de Newton

3.4 Mínimo de una función de varias variables

Como en el caso de las funciones de una variable, los mínimos locales de una función de dos variables se pueden alcanzar en los puntos críticos.

Si $z = f(x,y)$ tiene primeras derivadas parciales $f_x(x,y)$ y $f_y(x,y)$, entonces las soluciones del sistema formado por las ecuaciones:

$f_x(x,y) = 0$ y $f_y(x,y) = 0$ se llaman puntos críticos.

Para determinar los mínimos locales de una función de dos variables se usa el criterio de la segunda derivada que establece lo siguiente:

Si (p,q) es un punto crítico, $f_{xx}(p,q)f_{yy}(p,q) - f_{xy}^2(p,q) > 0$ y $f_{xx}(p,q) > 0$, entonces f tiene un mínimo local en (p,q).

Caso 1:

Encuentre el mínimo de la función siguiente

$$f(x,y) = (x^2 + 4y^2 - 16)^2 + (xy^2 - 4)^2$$

Solución:

Calculando las derivadas parciales f_x y f_y e igualándolas a cero se obtiene el sistema no lineal

$$f_x = 4x^3 + 16xy^2 - 64x + 2xy^4 - 8y^2$$

$$f_y = 416x^2y + 64y^3 - 256y + 4x^2y^3 - 16xy$$

Usando los métodos de Broyden y Homotopía y Continuación y tomando $x^{(0)} = (1,2)$ como aproximación inicial.

Broyden

k	x(1)	x(2)	err
1	0.877192	2.011147	0.074022
2	1.062462	1.924820	0.204395
3	1.055820	1.935053	0.012200
4	1.062879	1.931708	0.007812
5	1.074510	1.927326	0.012429
6	1.078068	1.926051	0.003779
7	1.078350	1.925958	0.000298
8	1.078381	1.925947	0.000033

Homotopía y Continuación

N	x(1)	x(2)
1	0.986380	1.995715
2	0.972648	1.991248
3	0.958800	1.986588
4	0.944830	1.981720

El punto (1.078381, 1.925947) aplicando el método de Broyden y (0.944830, 1.981720) aplicando el método de Homotopía y Continuación, es el punto crítico de la función anterior y aplicando el criterio de la segunda derivada se tiene que ese punto es el mínimo de la función $f(x,y) = (x^2 + 4y^2 - 16)^2 + (xy^2 - 4)^2$.

El método de Broyden es el que mejor calcula la solución aproximada con tolerancia de 0.0001 y realizando 8 iteraciones. El Método de Homotopía y Continuación aplicando Runge-Kutta de orden 4 con N=4 da una aproximación menos precisa.

Caso 2:

Supongamos que queremos ajustar una curva exponencial de la forma

$$y = Ce^{Ax_k}$$

a un conjunto de puntos $(x_1,y_1),(x_2,y_2),\dots,(x_N,y_N)$ dado de antemano. El método no lineal de los mínimos cuadrados consiste en hallar el mínimo de la función de error

$$E(A,C) = \sum_{k=1}^{N} (Ce^{Ax_k} - y_k)^2$$

Para ello, hallamos las derivadas parciales de $E(A,C)$ respecto de A y C,

$$\frac{\partial E}{\partial A} = 2\sum_{k=1}^{N} (Ce^{Ax_k} - y_k)\,(Cx_k e^{Ax_k})$$

$$\frac{\partial E}{\partial C} = 2\sum_{k=1}^{N} (Ce^{Ax_k} - y_k)(e^{Ax_k})$$

Al igualar a cero estas derivadas parciales obtenemos, después de simplificar, las correspondientes ecuaciones normales

$$C\sum_{k=1}^{N} x_k e^{2Ax_k} - \sum_{k=1}^{N} x_k y_k e^{Ax_k} = 0$$

$$C\sum_{k=1}^{N} e^{2Ax_k} - \sum_{k=1}^{N} y_k e^{Ax_k} = 0$$

Utilicemos ahora el método no lineal de los mínimos cuadrados para hallar el ajuste exponencial optimo $y = Ce^{Ax_k}$ de los cinco datos (0,1.5), (1,2.5), (2,3.5), (3,5.0) y (4,7.5) y encontrar el punto (A, C) que minimice el error

$$E(A,C) = \sum_{k=1}^{N} (Ce^{Ax_k} - y_k)^2$$

Solución:

Sustituyendo los cinco datos en las ecuaciones normales obtenemos el sistema de ecuaciones no lineales

$$Ce^{2A} + 2Ce^{4A} + 3Ce^{6A} + 4Ce^{8A} - 2.5e^{A} - 7e^{2A} - 15e^{3A} - 30e^{4A} = 0$$

$$C + Ce^{2A} + Ce^{4A} + Ce^{6A} + Ce^{8A} - 1.5 - 2.5e^{A} - 3.5e^{2A} - 5e^{3A} - 7.5e^{4A} = 0$$

Resolviendo el sistema anterior por los métodos de Newton y Homotopía y Continuación con aproximación inicial $x^{(0)} = (1,1)$ obtenemos los siguientes resultados.

	Newton			Homotopía y continuación		
k	**x(1)**	**x(2)**	**err**	**N**	**x(1)**	**x(2)**
1	0.930897	0.649003	0.357735	**1**	0.978839	0.914896
2	0.815002	0.694426	0.124479	**2**	0.944816	0.839733
3	0.681472	0.868830	0.219652	**3**	0.876759	0.797948
4	0.551481	1.128844	0.290697	**4**	0.512557	1.251105
5	0.449232	1.400637	0.290390			

6	0.395894	1.568773	0.176393
7	0.384062	1.609142	0.042067
8	0.383576	1.610866	0.001792
9	0.383575	1.610869	0.000003

El punto (0.383575, 1.610869) es una solución del sistema no lineal cuando aplicamos el Método de Newton con aproximación inicial de (1,1), tolerancia de 0.0001 realizando 9 iteraciones y se obtiene el punto (0.512557, 1.251105) al aplicar el Método de Homotopía y Continuación.

Tomando como aproximación inicial $x^{(0)} = (0.5,1.5)$ y aplicando el método de Newton obtenemos el siguiente resultado

Newton

k	**x(1)**	**x(2)**	**err**
1	0.436600	1.509544	0.064115
2	0.395354	1.583785	0.084929
3	0.384202	1.609304	0.027849
4	0.383577	1.610864	0.001681
5	0.383575	1.610869	0.000005

Observemos que el número de iteraciones se reduce de 9 iteraciones a 5 iteraciones.

El punto (0.383575,1.610869) que es solución del sistema, es también el punto crítico de la función de error $E(A,C) = \sum_{k=1}^{N}(Ce^{Ax_k} - y_k)^2$. Aplicando el criterio de la segunda derivada se tiene que este punto es el que minimiza el error.

3.5 Punto de equilibrio de dos o más fenómenos dados

Caso 1:

Consideremos el problema de predecir la población de dos especies que compiten por un mismo suministro de alimento. Si el número de individuos de las especies en el tiempo t se denota por x_1 y x_2, se supone frecuentemente que mientras la razón de natalidad de cada especie es simplemente proporcional al número de individuos de esa especie en ese tiempo, la razón de mortalidad de cada especie depende de la población de ambas especies. Supondremos que la población de una pareja particular de especies se describe por las ecuaciones

$$\frac{dx_1(t)}{dt} = x_2(t)(2 - 0.0002x_1(t) - 0.0001x_2(t))$$

y

$$\frac{dx_2(t)}{dt} = x_1(t)(4 - 0.0003x_1(t) - 0.0004x_2(t))$$

Determine el punto de equilibrio de las poblaciones de las dos especies.

Solución:

El criterio matemático que debe satisfacerse para que las poblaciones estén en equilibrio es que, simultáneamente

$$\frac{dx_1}{dt}(t) = 0$$

y

$$\frac{dx_2}{dt}(t) = 0$$

Esto es, debemos encontrar valores de x_1 y x_2 que satisfagan simultáneamente el sistema no lineal

$$\frac{dx_1(t)}{dt} = 2x_2(t) - 0.0002x_1(t)x_2(t) - 0.0001x_2^2(t) = 0$$

$$\frac{dx_2(t)}{dt} = 4x_1(t) - 0.0003x_1^2(t) - 0.0004x_1(t)x_2(t) = 0$$

Observe que claramente se satisface cuando la primera especie está extinta y la segunda tiene una población de 20,000, o cuando la segunda especie está extinta y la primera tiene una población de 13,333. ¿Puede haber equilibrio en alguna otra situación?

Solución:

Resolviendo el sistema anterior por el método Iterativo de Seidel con aproximación inicial de (1000, 1000) y tolerancia de 0.01 obtenemos los siguientes resultados

k	x(1)	x(2)	err
1	9500.000000	2875.000000	8704.345179
2	8562.500000	3578.125000	1171.875000
3	8210.937500	3841.796875	439.453125
4	8079.101562	3940.673828	164.794922
5	8029.663086	3977.752686	61.798096
6	8011.123657	3991.657257	23.174286
7	8004.171371	3996.871471	8.690357
8	8001.564264	3998.826802	3.258884

9	8000.586599	3999.560051	1.222081
10	8000.219975	3999.835019	0.458281
11	8000.082491	3999.938132	0.171855
12	8000.030934	3999.976800	0.064446
13	8000.011600	3999.991300	0.024167
14	8000.004350	3999.996737	0.009063

Otro punto de equilibrio para ambas especies es cuando la primera población es de 8000 y la segunda población de 4000.

Caso 2:

Las ecuaciones del esquema optimal de diferencia finita de dos etapas para una ecuación diferencial parcial parabólica son

$$12rq = 6r - 1$$

$$60r^2 - 180r^2q - 30rq = 1$$

y su solución exacta es

$$r = \frac{\sqrt{5}}{10} \quad , \qquad q = \frac{3 - \sqrt{5}}{6}$$

Encontrar el punto de equilibrio de las dos ecuaciones y compare con su solución exacta.

Solución:

Resolviendo el sistema no lineal

$$12rq - 6r + 1 = 0$$

$$60r^2 - 180r^2q - 30rq - 1 = 0$$

Resolviendo por los métodos de Homotopía y Continuación y del Descenso más rápido, con aproximación inicial de (0.2, 0.1), obtenemos los siguientes resultados.

Homotopía y Continuación

N	x(1)	x(2)
1	0.205939	0.107489
2	0.211852	0.114512
3	0.217741	0.121110
4	0.223607	0.127322

Descenso más rápido

k	x(1)	x(2)	g
1	0.197644	0.102480	0.003513
2	0.219452	0.124863	0.000368
3	0.220033	0.124245	0.000066
4	0.222243	0.126394	0.000022

El punto de equilibrio de las dos ecuaciones con el método de Homotopía y Continuación es (0.223607,0.127322) y con el método del Descenso más rápido con tolerancia de 0.0001 y 4 iteraciones es (0.22243, 0.126394). Comparando con la solución exacta (0.223606798, 0.127322004) notamos que con el Método de Homotopía y Continuación nos acercamos más al punto de equilibrio

CONCLUSIONES

- **Existe una variedad de métodos para resolver sistemas de ecuaciones no lineales algunos de ellos son: Iteración de Punto Fijo, Iterativo de Seidel, Newton, Broyden, Descenso más rápido y Homotopía y Continuación.**

- **En el Método de Punto Fijo se genera una sucesión $\{x^{(k)}\}_{k=0}^{\infty}$ con $x^{(k)} = G(x^{(k-1)})$ convergente a una de las soluciones del sistema no lineal. En este método se analiza G y las derivadas parciales en su dominio para garantizar la convergencia.**

- **El Método Iterativo de Seidel es una forma de acelerar la convergencia del método de Punto fijo empleando las últimas estimaciones $x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}$ en vez de $x_1^{(k-1)}, x_2^{(k-1)}, \dots, x_n^{(k-1)}$ para calcular $x_i^{(k)}$.**

- **En el Método de Newton se espera generalmente que dé convergencia cuadrática siempre y cuando se conozca un valor inicial lo suficientemente cerca de la solución y que la matriz jacobiana asociada al sistema no lineal exista. Este método converge rápidamente a la solución.**

- **El Método de Broyden reemplaza la matriz jacobiana del método de Newton por una matriz cuya inversa se determina directamente en cada paso, también necesita una buena aproximación inicial. En este método la convergencia cambia de cuadrática a superlineal.**

- **El Método del Descenso más rápido resulta una buena forma para obtener una aproximación inicial para otro método más efectivo, como Newton y Broyden, ya que este método no necesita de una buena aproximación inicial y no genera una sucesión rápidamente convergente.**

- **En el Método de Homotopía y Continuación se introduce el problema a resolver dentro de una colección de problemas de un parámetro λ. El problema original corresponde a $\lambda=1$ y un problema con una solución conocida corresponde a $\lambda=0$. Al igual que el Método del Descenso más rápido puede utilizarse para obtener un valor inicial para los métodos de Newton y Broyden.**

ANEXOS

ANEXO 1

PROGRAMA DEL MÉTODO DE ITERACIÓN DE PUNTO FIJO PARA SISTEMAS NO LINEALES

```
% Programa del método de Iteración de Punto Fijo para sistemas no lineales de ecuaciones

function x=iter(G,x0,tol,max)

%Datos

%G es el sistema no lineal archivado como G.m

%x0 es la aproximación inicial

%tol es la tolerancia

%max es el número máximo de iteraciones

%Resultados

%x es la aproximación a la solución

x0=input('Introduzca el punto inicial x0:   ')

tol=input('Introduzca la tolerancia:   ')

max=input('Introduzca el número máximo de iteraciones:   ')

n=length(x0);

k=1;

fprintf('\n k    x(1)     x(2)    err \n');

while(k<=max)

    x=feval('G',x0);

       err=norm(x-x0);

       fprintf('%d',k);

       fprintf('%10f',x(1));

       fprintf('%10f',x(2));
```

```
        fprintf('%10f',err);
        fprintf('\n');
             if err<=tol
          fprintf('\nProcedimiento completado \n');
          return;
       end
       k=k+1;
       x0=x;
 end
fprintf('\nNúmero máximo de iteraciones excedido \n');
```

SALIDA

```
iter2
Introduzca el punto inicial x0:   [0 0]
x0 =
     0     0
Introduzca la tolerancia:   0.0001
tol =
  1.0000e-004
Introduzca el número máximo de iteraciones:   30
max =
    30
```

```
k          x(1)         x(2)       err
1    0.800000    0.800000    1.131371
2    0.928000    0.931200    0.183296
3    0.972832    0.973270    0.061480
4    0.989366    0.989435    0.023123
5    0.995783    0.995794    0.009034
6    0.998319    0.998321    0.003580
7    0.999328    0.999329    0.001427
8    0.999732    0.999732    0.000570
9    0.999893    0.999893    0.000228
10   0.999957    0.999957    0.000091
Procedimiento completado
ans =
 1.000  1.0000
```

ANEXO 2

PROGRAMA DEL MÉTODO ITERATIVO DE SEIDEL PARA SISTEMAS NO LINEALES

```
% Programa del método de Seidel para funciones de ecuaciones
no lineales

function x=seidel(G,x0,tol,max)

x0=input('Introduzca el punto inicial x0:   ')

tol=input('Introduzca la tolerancia:   ')

max=input('Introduzca el número máximo de iteraciones:   ')

n=length(x0);

k=1;

fprintf('\n k   x(1)   x(2)  err \n');

for k=1:max

   x=x0;

   for j=1:n

      A=feval('G',x);

      x(j)=A(j);

   end

   dif=norm(x-x0);

       fprintf('%d',k);

       fprintf('%10f',x(1));

       fprintf('%10f',x(2));

       fprintf('%10f',dif);

       fprintf('\n');
```

```
            if dif<=tol
          fprintf('\nEl método converge \n');
          return;
       end
       k=k+1;
       x0=x;
 end
   fprintf('\nEl método no converge \n');
```

SALIDA

```
Seidel

Introduzca el punto inicial x0:    [0 0]
x0 =
     0     0

Introduzca la tolerancia:   0.0001
tol =
  1.0000e-004

Introduzca el número máximo de iteraciones:   30
max =
    30
```

```
k     x(1)        x(2)        err
1    0.800000    0.880000    1.189285
2    0.941440    0.967049    0.166081
3    0.982149    0.990064    0.046765
4    0.994484    0.996930    0.014117
5    0.998287    0.999045    0.004351
6    0.999467    0.999703    0.001351
7    0.999834    0.999907    0.000420
8    0.999948    0.999971    0.000131
9    0.999984    0.999991    0.000041
 Procedimiento completado
ans =
    1.0000    1.0000
```

ANEXO 3

PROGRAMA DEL MÉTODO DE NEWTON PARA SISTEMAS NO LINEALES

```
% Programa del método de Newton para sistemas no lineales de ecuaciones

function x=newdim(F,JF,x0,tol,max)

% Datos

% F es el sistema no lineal, archivado como F.m

% JF es la matriz jacobiana de F, archivada como JF.m

% x0 es la aproximación inicial

% tol es la tolerancia

% max es el número máximo de iteraciones

% Resultados

% x es la aproximación a la solución que se obtiene

x0=input('Introduzca el punto inicial x0:   ')

tol=input('Introduzca la tolerancia:   ')

max=input('Introduzca el número máximo de iteraciones:   ')

n=length(x0);

k=1;

fprintf('\n k    x(1)     x(2)    err \n');

while(k<=max)

   B=feval('F',x0);

   C=-1*B;

   J=feval('JF',x0);
```

```
   Y=J\C;  %Resuelve el sistema lineal J(x0)Y=-F(x0)
   x=x0+Y;
   err=norm(x-x0);
   fprintf('%d',k);
   fprintf('%10f',x(1));
   fprintf('%10f',x(2));
   fprintf('%10f',err);
   fprintf('\n');
 if(err<=tol)
   fprintf('\nProcedimiento completado \n');
   return;
end
k=k+1;
x0=x;
end
fprintf('\n Número máximo de iteraciones excedido \n');
```

SALIDA

```
newton
Introduzca el punto inicial x0:   [0 0]'
x0 =
     0
     0
Introduzca la tolerancia:   0.0001
tol
```

```
1.0000e-004

Introduzca el número máximo de iteraciones:   30
max =
    30

k      x(1)      x(2)       err
1   0.800000   0.880000   1.189285
2   0.991787   0.991712   0.221950
3   0.999975   0.999969   0.011628
4   1.000000   1.000000   0.000040
Procedimiento completado
ans =
    1.0000
    1.0000
```

ANEXO 4

PROGRAMA DEL MÉTODO DE BROYDEN PARA SISTEMAS NO LINEALES

```
% Programa del método de Broyden para sistemas no lineales de
ecuaciones

function x=broyden(F,JF,x0,tol,max)

% Datos

% F es el sistema no lineal, archivado como F.m

% JF es la matriz jacobiana de F, archivada como JF.m

% x0 es la aproximación inicial

% tol es la tolerancia

% max es el número máximo de iteraciones

% Resultados

% x es la aproximación a la solución que se obtiene

x0=input('Introduzca el punto inicial x0:   ')

tol=input('Introduzca la tolerancia:   ')

max=input('Introduzca el número máximo de iteraciones:   ')

n=length(x0);

fprintf('\n k      x(1)      x(2)    err \n');

A0=feval('JF',x0);

v=feval('F',x0);

A=inv(A0);

k=1;

s=-1*A*v; %s=s1
```

```
x=x0+s;    %x=x1
while (k<=max)
   w=v;
   v=feval('F',x);
   y=v-w;
   z=-1*A*y;
   p=-1*s'*z;
   I=eye(2);
   c=p*I+(s+z)*s';
   A=(1/p)*c*A;
   s=-1*A*v;
   x=x+s;
    err=norm(s);
   fprintf('%d',k);
   fprintf('%10f',x(1));
   fprintf('%10f',x(2));
   fprintf('%10f',err);
   fprintf('\n');
if(err<=tol)
   fprintf('\nProcedimiento completado \n');
   return;
end
k=k+1;
x0=x;
end
```

```
fprintf('\n Número máximo de iteraciones excedido \n')
```

SALIDA

```
Broyden
Introduzca el punto inicial x0:   [0 0]'
x0 =
     0
     0
Introduzca la tolerancia:   0.0001
tol =
  1.0000e-004
Introduzca el número máximo de iteraciones:   30
max =
    30
k    x(1)      x(2)      err
1  0.962080  0.967201  0.184049
2  0.997434  0.996786  0.046100
3  0.999904  0.999845  0.003932
4  0.999998  0.999997  0.000179
5  1.000000  1.000000  0.000004
Procedimiento completado
ans =
    1.0000
    1.0000
```

ANEXO 5

PROGRAMA DEL MÉTODO DEL DESCENSO MÁS RÁPIDO PARA SISTEMAS NO LINEALES

```
% Programa del método del descenso más rápido para sistemas
no lineales de ecuaciones

function x=desc(F,JF,DG,x0,tol,max)

% Datos

% F es el sistema no lineal, archivado como F.m

% JF es la matriz jacobiana de F, archivada como JF.m

% DG es la función a ser minimizada

% x0 es la aproximación inicial

% tol es la tolerancia

% max es el número máximo de iteraciones

% Resultados

% x es la aproximación a la solución que se obtiene

x0=input('Introduzca el punto inicial x0:   ')

tol=input('Introduzca la tolerancia:   ')

max=input('Introduzca el número máximo de iteraciones:   ')

n=length(x0);

k=1;

fprintf('\n k    x(1)     x(2)    g \n');

while k<=max

   g1=feval('DG',x0);

   A=feval('F',x0);  % Empieza cálculo de gradiente
```

```
   B=feval('JF',x0);
   S=B';
   z=2*S*A;            % Termina cálculo de gradiente
   z0=norm(z);
   if z0==0
   fprintf('%d',k);            %Procedimiento  terminado,puede
tener un mínimo
   fprintf('%10f',x0(1));       %Verificar más aún
   fprintf('%10f',x0(2));
   fprintf('%10f',g1);
   fprintf('\n');
   fprintf('\n Gradiente cero\n');
   return;
   end
   z=z/z0;
   a1=0;
   a3=1;
   c=x0-(a3*z);
   g3=feval('DG',c);
   while g3>=g1
      a3=a3/2;
      d=x0-(a3*z);
      g3=feval('DG',d);
      if a3<(tol/2)
  fprintf('%d',k);%Procedimiento  terminado,  puede  tener  un
mínimo
```

```
   fprintf('%10f',x0(1));       %Verificar más aún
   fprintf('%10f',x0(2));
   fprintf('%10f',g1);
   fprintf('\n');
   fprintf('\n No hubo mejora\n');
     return;
      end
   end
   a2=a3/2;
   e=x0-(a2*z);
   g2=feval('DG',e);
   h1=(g2-g1)/(a2);
   h2=(g3-g2)/(a3-a2);
   h3=(h2-h1)/(a3);%polinomiocuadrático P(a)=g1+h1*x+h3*x*(a-
a2)
   a0=0.5*(a2-h1/h3);  %En  a0  hay  un  punto  crítico  del
polinomio cuadrático P
   m=x0-(a0*z);
   g0=feval('DG',m);
   g=min([g0 g3]);    %Para encontrar a
   if g==g0
      a=a0;
      elseif g==g3
         a=a3;
      end
     x=x0-(a*z);
```

```
   err=abs(g-g1);
   fprintf('%d %10f %10f %10f \n',k,x(1),x(2),g);
if err<tol
   fprintf('\nProcedimiento completado \n');
   return;
      end
   k=k+1;
   x0=x;
   end
fprintf('\n Número máximo de iteraciones excedido \n');
```

SALIDA

```
desc

Introduzca el punto inicial x0:   [0 0]'
x0 =
     0
     0

Introduzca la tolerancia:   0.0001
tol =
  1.0000e-004
Introduzca el número máximo de iteraciones:   30
max =
    30
```

```
k     x(1)       x(2)        g
1   0.787983   0.875536   2.678354
2   0.982985   0.923515   0.343754
3   0.969119   0.980500   0.053020
4   0.996770   0.987236   0.009197
5   0.994533   0.996515   0.001644
6   0.999406   0.997690   0.000299
7   0.999002   0.999362   0.000055
8   0.999891   0.999577   0.000010
Procedimiento completado
ans =
    0.9999
```

0.9996

ANEXO 6

PROGRAMA DEL MÉTODO DE HOMOTOPÍA Y CONTINUACIÓN PARA SISTEMAS NO LINEALES

```
% Programa del método de continuación para sistemas no
lineales de ecuaciones

function x=cont(F,JF,x0,N)

% Datos

% F es el sistema no lineal, archivado como F.m

% JF es la matriz jacobiana de F, archivada como JF.m

% x0 es la aproximación inicial

% Entero N>0

% Resultados

% x es la aproximación a la solución que se obtiene

x0=input('Introduzca el punto inicial x0:   ')

N=input('Introduzca el valor N>0:   ')

fprintf('\n x(1)      x(2)    \n');

h=1/N;

a=feval('F',x0);

b=-1*h*a;

for i=1:N

   A=feval('JF',x0);

   k1=A\b;                %Resuelve el sistema lineal Ak1=b

   C=x0+(1/2*k1);

   D=feval('JF',C);
```

```
   k2=D\b;                %Resuelve el sistema lineal Ak2=b
   E=x0+(1/2*k2);
   F=feval('JF',E);
   k3=F\b;                %Resuelve el sistema lineal Ak3=b
   G=x0+k3;
   I=feval('JF',G);
   k4=I\b;                %Resuelve el sistema lineal Ak4=b
   x=x0+(k1+2*k2+2*k3+k4)/6;
   fprintf('%f',x(1));
   fprintf('%10f',x(2));
   fprintf('\n');
   x0=x;
end
```

SALIDA

```
cont
Introduzca el punto inicial x0:   [0 0]'
x0 =
     0
     0
Introduzca el valor N>0:   4
N =
     4
```

```
N     x(1)       x(2)

1   0.209308    0.221962

2   0.439871    0.453015

3   0.698412    0.704510

4   1.000013    1.000027

ans =

    1.0000

    1.0000
```

BIBLIOGRAFÍA

- Burden Richard L. y Faires J. Douglas. Análisis Numérico. Séptima Edición. Editorial Thomson Learning. México D.F. 2002.

- Chapra Steven C. y Canales Raymond P. Métodos Numéricos para Ingenieros. Editorial McGraw-Hill. México. 1998.

- Fausett V. Laurene. Applied Numerical Analisis using MATLAB. Tercera Edición. Editorial Prentice Hall. Madrid. 2000.

- Mathews John H. y Fink Kurtis D. Métodos Numéricos con MATLAB. Tercera Edicion. Editorial Prentice Hall. Madrid. 2000.

- Rudin Walter. Real and Complex Analysis. Editorial Tata McGraw-Hill. New Delhi. 1974.

- Scheid Francis. Análisis Numérico. Editorial McGraw-Hill. México. 1972.