Escuela de Electrónica

Curso

CE3102 - Análisis Numérico para Ingeniería

Tarea 2 - Parte 2: Método Iterativo Newton-Raphson

Profesor: Juan Pablo Soto Quirós

Elaborado por

José Julián Camacho Hernández - 2019201459
Juan Pablo Carrillo Salazar - 2019380111
José Leonardo Guillén Fernández - 2019031688
Fabián Ramírez Arrieta - 2018099536

Mayo 2022 Cartago, Costa Rica

Introducción

En este documento se explicará el método Newton-Raphson, el cual fue visto en clase para ser modificado y de esta manera resolver ecuaciones no lineales.

Según Smith el método consiste en una serie de pasos sencillos para poder implementar este método.

Primero se debe escogen un valor inicial lo suficientemente cercano a la raíz buscada, con este valor se comienza a ejecutar las iteraciones de manera secuencial hasta aproximar el resultado.

El método original tiene como fórmula matemática la siguiente ecuación:

$$X_{n+1} = X_n - \frac{f(X_n)}{f(X_n)};\ X_0 \in \mathbb{R} \wedge f'(X_n) \neq 0 \qquad (1)$$

Problema a resolver

Para la sección 2 de la tarea 2, se expone el problema de resolver un sistema de ecuaciones de n variables mediante el método iterativo Newton-Raphson, sin embargo, como observarnos en la formula 1 de utiliza la derivada de la función ingresada, pero lo modificaremos utilizando el método de jacobiano, que tiene la siguiente ecuación:

$$J_f(X_n) * y = f(X_n) \qquad (2)$$

Formulación matemática del método

Ahora reescribiendo la formula 1 aplicando el jacobiano de la ecuación 2, obteniendo la siguiente ecuación:

$$X_{n+1} = X_n - [J_f(X_n)]^{-1} * f(X_n) \qquad (3)$$

Valores iniciales

$x^{(0)}$	Vector inicial	R^m
f	Vector string (f1, f2, f3….)	$R^m \to R$
x	Vector string (x1, x2 …) variables a utilizar	R^m
tol	Tolerancia	Mayor que 0
interMax	Cantidad máxima de iteraciones	Mayor que 0

Valores de Salida

$x^{(k)}$	Aproximación de la solución	R^m
k	Cantidad de iteraciones	N
Error	Error de la aproximación	R
Graficas	Iteraciones vr errores	

Pseudocódigo.

A continuación, se mostrarán los pasos para ejecutar el código de manera correcta, se hará un pseudocódigo el cual busca solucionar un sistema de ecuaciones no lineales utilizando el método anterior y su formula 3, recordando que su resultado será mas preciso con un numero de iteraciones más elevado.

Evaluar si tol > 0

Inicializar variables que lleven el control del circuito (k=1, error=tol+1, vectores e y F = [])

Crear un While con las condiciones (error > tol) y (k < interMax)

Calcular jacobiano de la función f

Sustituir valores X0 en la función f

Guardar soluciones en vector F

Calcular el error con la formula || f(Xn) ||

Se actualiza la variable k+=1

Graficar punto calculado, entre interacción vrs error