Área Académica de Ingeniería en Computadores
Licenciatura en Ingeniería en Computadores
CE-3102: Análisis Numérico para Ingeniería

Método Iterativo Para Encontrar Ceros De Funciones

Tarea 1 - Parte 2 : Pregunta 1

Brayan Alfaro González
alfabrayan12@gmail.com
2019380074

Sebastián Alva Vives
sebasjosue84@gmail.com
2017097108

Kevin Zeledón Salazar
kev_sala@estudiantec.cr
2018076244

Daniel Camacho González
ddcamachog1501@estudiantec.cr
2017114058

Cartago, Costa Rica
Marzo 2021

1. Formulación Matemática

El método iterativo presentado en el actual trabajo para la solución de ecuaciones no lineales corresponde al determinado por las ecuaciones del sistema 1. Este procedimiento tiene el objetivo de facilitar la obtención de raíces reales α a ecuaciones no lineales, de manera que $f(\alpha) = 0$ para una ecuación no lineal de la forma $f(x) = 0$. En particular, el método aquí presentado corresponde a un método iterativo, sin memoria, de tres pasos, óptimo de octavo orden, el cual es sugerido por Diyashvir et al. en [1] e implementa mejoras en la convergencia y eficiencia a partir del método dado por Chun y Neta en [2].

$$\begin{cases} y_n = x_n - \frac{f(x_n)}{f'(x_n)}(1 - (\frac{f(x_n)}{f'(x_n)})^5) \\ z_n = y_n - \frac{f(y_n)}{f'(x_n)}(1 - \frac{f(y_n)}{f(x_n)})^{-2} \\ x_{n+1} = z_n - \frac{f(z_n)}{f'(x_n)} \frac{(1+(\frac{f(y_n)}{f(x_n)})^2 + \frac{f(z_n)}{f(y_n)})}{(1-\frac{f(y_n)}{f(x_n)} - \frac{f(z_n)}{f(x_n)})^2} \end{cases} \tag{1}$$

2. Valores Iniciales

Es recomendable usar valores iniciales cercanos al cero buscado, ya que de otra manera el método podría divergir o encontrar un cero diferente al que se está buscando. Como ejemplo, se presentan los resultados obtenidos en el cuadro 1. En este se usaron funciones las cuales tienen ceros cercanos a la unidad [1]. En dicha tabla se puede apreciar como el error crece grandemente y las aproximaciones no convergen cuando el valor inicial se aleja mucho del valor de la raíz. Esto sucede en todas las funciones, excepto en la más derecha, ya que es una función sinusoidal y tiene infinitas raíces.

Cuadro 1: Errores para diferentes funciones de acuerdo a su valor inicial

Función / **Val. Inicial**	$10xe^{-x^2} - 1$	$x^5 + x^4 + 4x^2 - 15$	$11x^{11} - 1$	$sin(3x) + xcos(x)$
1	3.2580e-11	-1.7763e-15	-1.2212e-15	2.2204e-16
1.5	-2.2204e-16	-1.7763e-15	4.440e-16	2.2204e-16
10	-1.0	∞	1.9899e-12	-2.8466e-13
15	-1.0	-2.8489e+42	∞	-1.6620e-13

3. Ventajas

- Al ser un método de octavo orden presenta una rápida convergencia, en particular cuando la raíz se encuentra cerca del valor inicial.
- El método es óptimo, esto debido a que su índice de eficiencia se acerca a la máxima eficiencia teórica. Esto tomando en cuenta que el índice de eficiencia viene dado por $\rho^{1/\alpha}$ [3]. Dónde ρ es el orden de convergencia y α es la cantidad de calculos por iteración.
- Este método presenta poca divergencia comparado con otros en la literatura.

4. Desventajas

- Al buscar la solución de polinomios de alto grado el método comienza a requerir de más cantidad de iteraciones.
- Cuando se utilizan valores iniciales lejanos a las raíces el método no converge.
- Debido a que se realizan múltiples divisiones, aumenta la posibilidad de una indefinición.
- Se requiere del calculo de una derivada, lo cual no es tan preferible.
- Se requiere del calculo de múltiples valores en cada iteración, lo que significa un mayor costo computacional por iteración.

5. Pseudocódigo

Algoritmo 1: Pseudocódigo del método implementado

Entradas: La función, valor inicial, la tolerancia, el límite de iteraciones
Salida **:** La raíz encontrada, la cantidad de iteraciones, el error

Procedimiento `método_nuevo(`$funcion$, $valorInicial$, $tolerancia$, $iterMax$`):`
  $derivada \leftarrow$ derivar($funcion$)
  crearMétodo($funcion$)
  crearMétodo($derivada$)
  $x_n \leftarrow valorInicial$
  $k \leftarrow 0$
  mientras *funcion(x_n)* $>$ *tolerancia* **Y** $k < iterMax$ **hacer**
    `// Se realiza un calculo auxiliar`
    $xdx \leftarrow$ funcion(x_n)/derivada(x_n)
    `// Se calcula` y_n
    $y_n \leftarrow x_n - xdx(1-(xdx)^5)$
    `// Se realiza un calculo auxiliar`
    $ydx \leftarrow$ funcion(y_n)/derivada(x_n)
    $yx \leftarrow$ funcion(y_n)/funcion(x_n)
    `// Se calcula` z_n
    $z_n \leftarrow y_n - ydx(1-yx)^{-2}$
    `// Se realiza un calculo auxiliar`
    $zdx \leftarrow$ funcion(z_n)/derivada(x_n)
    $zy \leftarrow$ funcion(z_n)/funcion(y_n)
    $zx \leftarrow$ funcion(z_n)/funcion(x_n)
    `// Se calcula` x_n
    $x_n \leftarrow z_n - zdx(1+(yx)^2+zy)/(1-yx-zx)^2$
    `// Se aumenta el número de iteraciones`
    $k \leftarrow k$+1
  fin
  devolver [x_n, k, funcion(x_n)]
fin

Referencias

[1] D. Babajee, A. Cordero, F. Soleymani and J. Torregrosa, On improved three-step schemes with high efficiency index and their dynamics, Numerical Algorithms, vol. 65, no. 1, pp. 153-169, 2013. Disponible: 10.1007/s11075-013-9699-6 [Accesado 5 Marzo 2021].

[2] Chun, C., Neta, B.: A new sixth-order scheme for nonlinear equations. Appl. Math. Lett. 25, 185–189, 2012.

[3] M. Grau-Sánchez, M. Noguera and J. Diaz-Barrero, Note on the efficiency of some iterative methods for solving nonlinear equation. 2000.