Método de Muller

Juan Páez[1], Santiago Zuñiga[2]

Resumen
In this paper we will explain the Muller method used to solve equations in one variable with complex roots, we will also present an application problem with the computational solution and the respective error analysis.

Keywords
Muller — Solution — Complex Roots — One Variable — Error

[1] *Facultad de ingeniería, Pontificia Universidad Javeriana*
[2] *Facultad de ingeniería, Pontificia Universidad Javeriana*
Autor correspondiente: jd.paez@javeriana.edu.co

Introducción

A la hora de solucionar ecuaciones no lineales en una variable, es decir encontrar o aproximar sus raíces, se pueden utilizar distintos métodos, como el despeje directo, Newton, posición falsa, entre otros. Sin embargo, se presenta un problema cuando las funciones que queramos evaluar tengan raíces complejas, ya que los métodos mencionados no son los indicados para este tipo planteamientos.
Es por eso que el matemático estadounidense, David Eugene **Muller**, propuso un método para poder obtener un aproximación de este tipo de funciones el cual será explicado y desarrollado a lo largo de este documento.

1. Definición

El método de Muller, esta muy relacionado con el método de la secante, teniendo en cuenta que este consiste en tener una aproximación de la raíz a partir de dos puntos en la función $f(x)$.

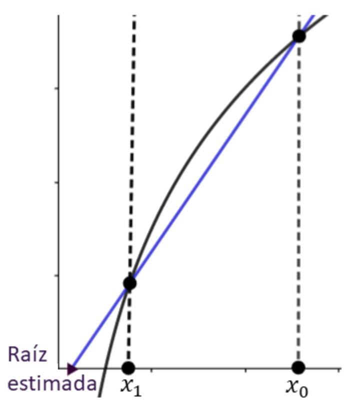

Figura 1. Método de la secante

En el caso de Muller consiste en tener 3 puntos sobre la gráfica de la función, siendo estos una composición cuadrática, la cual da una aproximación de la solución o raíz de $f(x)$.

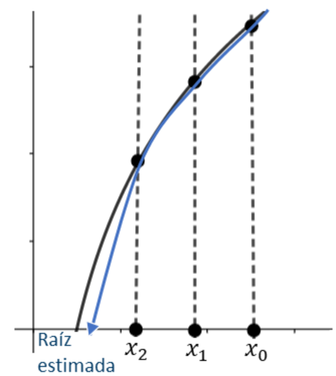

Figura 2. Método de Muller

1.1 Definición matemática

Teniendo en cuenta la forma general de la ecuación cuadrática y la aproximación de la raíz, el metodo comienza planteando el siguiente polinomio cuadrático:

$$g(x) = a(x-x_2)^2 + b(x-x_2) + c \tag{1}$$

Queremos que la función $g(x)$, representada de color azul, pase por los puntos presentados en la figura 2, es decir $[x_0, f(x_0)]$, $[x_1, f(x_1)]$ y $[x_2, f(x_2)]$,

por lo tanto sustituyendo en la ecuación (1), nos da las respectivas imagenes de las funciones en $g(x_k)$.

$$g(x_0) = a(x_0-x_2)^2 + b(x_0-x_2) + c \tag{2}$$

$$g(x_1) = a(x_1-x_2)^2 + b(x_1-x_2) + c \tag{3}$$

$$g(x_2) = a(x_2-x_2)^2 + b(x_2-x_2) + c \tag{4}$$

Al haber tres ecuaciones, podemos hallar los valores de los tres coeficientes a, b y c. Empezando por el valor de c, que utilizando la ecuación (4) sería:

$$\begin{aligned} g(x_2) &= a(0)^2 + b(0) + c \\ g(x_2) &= c \end{aligned} \tag{5}$$

Por lo tanto al reemplazar c en las ecuaciones (2) y (3) tenemos que:

$$g(x_0) - g(x_2) = a(x_0 - x_2)^2 + b(x_0 - x_2) \tag{6}$$

$$g(x_1) - g(x_2) = a(x_1 - x_2)^2 + b(x_1 - x_2) \tag{7}$$

Empleando la fórmula para hallar la pendiente de una recta, podemos hallar una aproximación a las derivadas, dando como resultado los coeficientes a y b.

$$\begin{aligned} h_0 &= x_1 - x_0 \\ h_1 &= x_2 - x_1 \end{aligned} \tag{8}$$

$$\begin{aligned} \delta_0 &= \frac{f(x_1) - f(x_0)}{x_1 - x_0} \\ \delta_1 &= \frac{f(x_2) - f(x_1)}{x_2 - x_1} \end{aligned} \tag{9}$$

Utilizando las ecuaciones (9) y (8), reemplazándolas en (6) y (7)

$$\begin{aligned} h_0\delta_0 + h_1\delta_1 = b(h_0 + h_1) - a(h_0 + h_1)^2 \\ h_1\delta_1 = bh_1 - ah_1^2 \end{aligned} \tag{10}$$

Al despejar a y b, de la anterior ecuación nos da el resultado de los coeficientes

$$a = \frac{\delta_1 - \delta_0}{h_1 - h_0} \tag{11}$$

$$b = ah_1 + \delta_1 \tag{12}$$

Teniendo como calcular los coeficientes, podemos hallar la raíz x_3 por medio de la fórmula cuadrática en (1) con $x = x_3$, sin embargo, como se comenta en las referencias [1, 2], no se puede emplear la formula tal cual, teniendo en cuenta los posibles errores de redondeo al usarla, por lo tanto hay que usar una variante:

$$x_3 - x_2 = \frac{-2c}{b(signo(b))\sqrt{b^2 - 4ac}} \tag{13}$$

Despejando x_3

$$x_3 = \frac{-2c}{b(signo(b))\sqrt{b^2 - 4ac}} + x_2 \tag{14}$$

Teniendo en cuenta que x_2 de manera iterativa es el punto más aproximado a la raíz, el error se calcula:

$$\varepsilon_n = |\frac{x_3 - x_2}{x_3}| \tag{15}$$

Cabe mencionar que el signo que se escoge para este método, en la variante de la formula cuadrática, es el mismo signo de b, para que el denominador sea más grande, dando un número más aproximado a la raíz, por eso se usa la notación $signo(b)$.

2. Pseudo código

A continuación se presenta como sería la implementación del método, expresado en palabras naturales:

PSEUDO CÓDIGO ACÁ

Ahora que ya se definió como es el algoritmo del método de Muller, a continuación se presentará un problema extraído del libro de Burden, el cual será solucionado a partir de un algoritmo desarrollado en lenguaje de programación *Python*.

3. Problema de aplicación

El problema a evaluar es de la sección 2.6, ejercicio numero 10[1]:

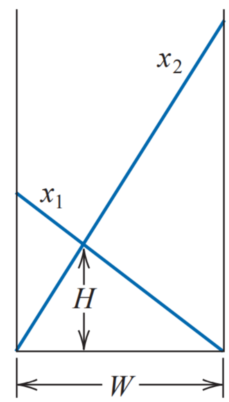

Figura 3. Dibujo problema

Dos escaleras atraviesan un callejón de ancho w. Cada escalera llega desde la base de una pared hasta algún punto de la pared opuesta. Las escaleras se cruzan a una altura H sobre la acera. Halla W dado que las longitudes de las escaleras son $x_1 = 20$ pies y $x_2 = 30$ pies, y que $H = 8$ pies.

4. Planteamiento del problema

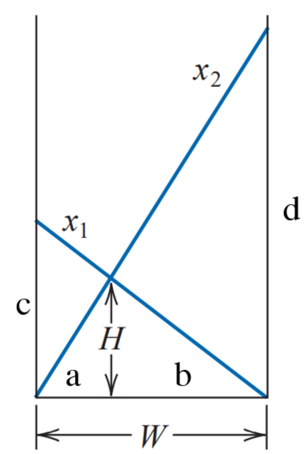

Figura 4. Planteamiento problema

Empezamos creando unas nuevas variables, siendo a la distancia de la pared de la izquierda hasta donde se cruzan las escaleras, y b es parecido la distancia de a, solo que lo mide desde el cruce hasta la pared de la derecha. Generando c, se crea un triangulo rectángulo entre w y x_1, y con d se crea otro triangulo de noventa grados con w y x_2
Por el teorema de pitágoras

$$\begin{aligned} w^2 + c^2 &= x_1^2 \\ w^2 + d^2 &= x_2^2 \end{aligned} \tag{16}$$

Despejando c y d tenemos

$$\begin{aligned} c &= \sqrt{x_1^2 - w^2} \\ d &= \sqrt{x_2^2 - w^2} \end{aligned} \tag{17}$$

Siguiendo con el teorema de tales

$$\begin{aligned} \frac{a}{h} &= \frac{w}{d} \\ \frac{b}{h} &= \frac{w}{c} \end{aligned} \tag{18}$$

Reemplazando 17 y despejando a y b en la anterior ecuación

$$\begin{aligned} a &= \frac{w \cdot h}{\sqrt{x_2^2 - w^2}} \\ b &= \frac{w \cdot h}{\sqrt{x_1^2 - w^2}} \end{aligned} \tag{19}$$

Sabemos entonces, que la suma de a y b da la distancia de w, entonces, usando 19 tenemos

$$w = \frac{w \cdot h}{\sqrt{x_2^2 - w^2}} + \frac{w \cdot h}{\sqrt{x_1^2 - w^2}} \tag{20}$$

Moviendo w al otro lado de la ecuación tenemos que

$$1 = \frac{h}{\sqrt{x_2^2 - w^2}} + \frac{h}{\sqrt{x_1^2 - w^2}} \tag{21}$$

Reemplazando h, x_1 y x_2 da la ecuación

$$0 = \frac{8}{\sqrt{30^2 - w^2}} + \frac{8}{\sqrt{20^2 - w^2}} - 1 \tag{22}$$

Usando el programa desarrollado en *Python*, nos da como resultado, con sus respectivas iteraciones

FOTO DEL RESULTADO DE CÓDIGO MULLER

Comparando con la implementación en *Python* del método de la secante obtenido de internet

FOTO DEL RESULTADO DE CÓDIGO SECANTE

5. Conclusiones

Referencias

[1] Richard L Burden et al. Análisis numérico. 10, 2017.

[2] Steven C Chapra, Raymond P Canale, et al. *Métodos numéricos para ingenieros*, volume 5. McGraw-Hill, New York, NY, 2011.