Universidad Nacional Autónoma de México

Escuela Nacional de Estudios Superiores

Unidad Juriquilla

Computación II

# Practica 2

# Método de bisección

Bruno Arturo López Pacheco

Dr. Ulises Olivares Pinto

No. De Cuenta 317347140

Licenciatura en Tecnología

**Objetivo**

Implementar el método de bisección y el de exploración exhaustiva para realizar una comparación de tiempos de ejecución.

**Marco teórico**

Este método se usa para resolver ecuaciones y se basa en el Teorema de los Valores Intermedios, el cual dice que toda función continua f, en un intervalo cerrado *[a,b],* toma todos los valores que se hallan entre *f(a)* y *f(b),* de tal forma que la ecuación $f(x) = 0$ tiene una sola raíz que verifica $f(a) \cdot f(\mathrm{b}) < 0$

Consiste en partir de un intervalo $[x_0, x_1]$ tal que $f(a) \cdot f(b) < 0$, por lo que se sabe que existe al menos una raíz real. A partir de ese punto se va reduciendo el intervalo sucesivamente hasta hacerlo tan pequeño como se diga la precisión que se haya decidido usar.

El método consiste en:

- Debe existir seguridad sobre la continuidad de la función $f(x)$ en el intervalo $[a, b]$.
- Se verifica que $f(a) \cdot f(b) < 0$
- Se calcula el punto medio m del intervalo $[a, b]$ y se evalúa $f(m)$; si ese valor es igual a cero, es la raíz buscada.
- En caso de que no lo sea, se verifica si $f(m)$ tiene signo opuesto con $f(a)$ o con $f(b)$.
- Se define un nuevo intervalo $[a, b]$ siendo ahora $[a, m]$ o $[m, b]$ según se haya determinado en cuál de estos intervalos ocurre un cambio de signo,
- Con este nuevo intervalo se continúa sucesivamente encerrando la solución en un intervalo cada vez más pequeño hasta alcanzar la precisión deseada.

Este método es muy simple y al mismo tiempo muy robusto.

**Metodología**

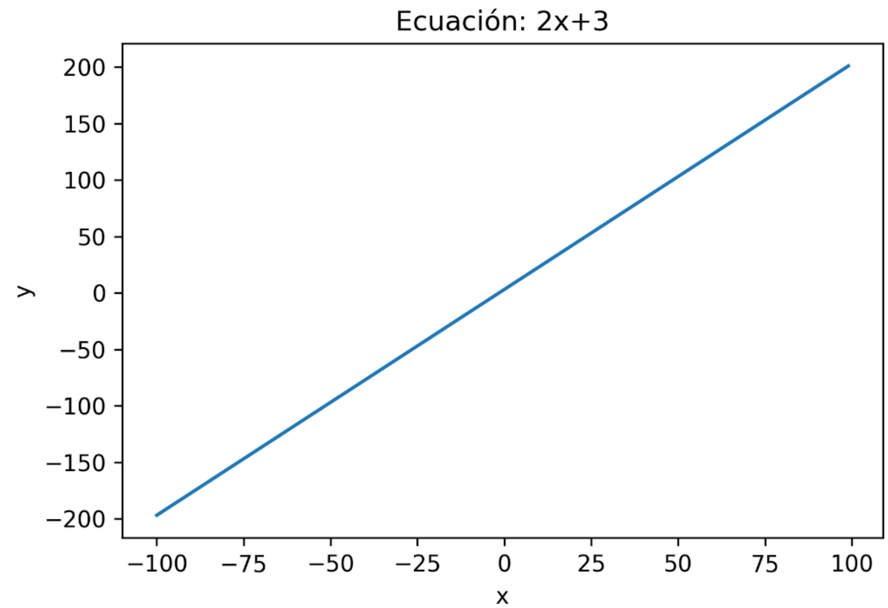
*Pseudocódigo del método de bisección*

```
    Si (f(a)*f(b) < 0):
      Existe un cambio de signo
    Sino:
      No existen raíces reales en el intervalo
      Salir

    error = inf
    iteración=1

    mientras (error>=tolerancia o iteración==máximaIteración):
      c=(a+b)/2
      Si (f(a)*f(c) < 0):
        Hay una raíz en [a, c]
        a=a
        b=c
      Si (f(a)*f(c) > 0):
        Hay una raíz en [c, b]
        a=c
        b=b
      Si (f(a)*f(c) == 0):
        La raíz real se encuentra en: c
        break
      error=abs(b-a)
      iteración=iteración+1
```

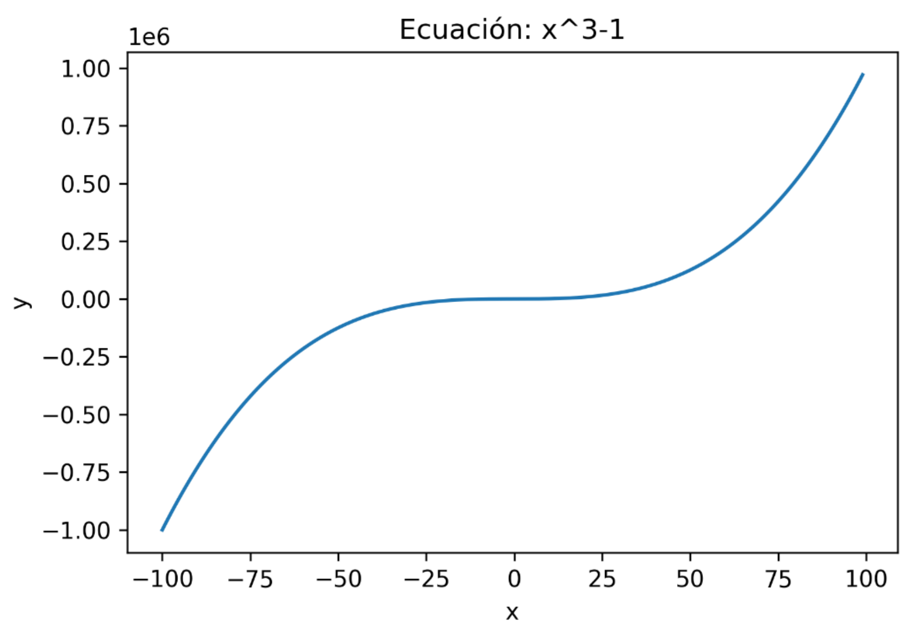
*Pruebas para comparación*

Al momento de comparar los tiempos entre métodos (bisección e ingenuo), se analizaron las respuestas de 3 ecuaciones específicas, las cuales son:

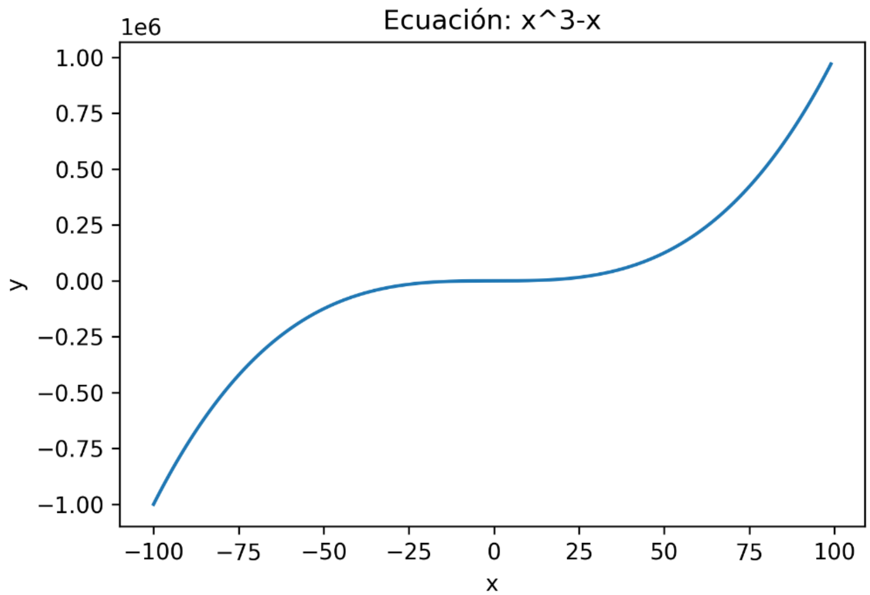
$f_{1(x)} = 2x + 3$ con raíz real de -1.5




$f_2(x) = x^3 - 1$ con raíz real de 1



$f_3(x) = x^3 - x$ con 3 raíces reales; -1,0 y 1



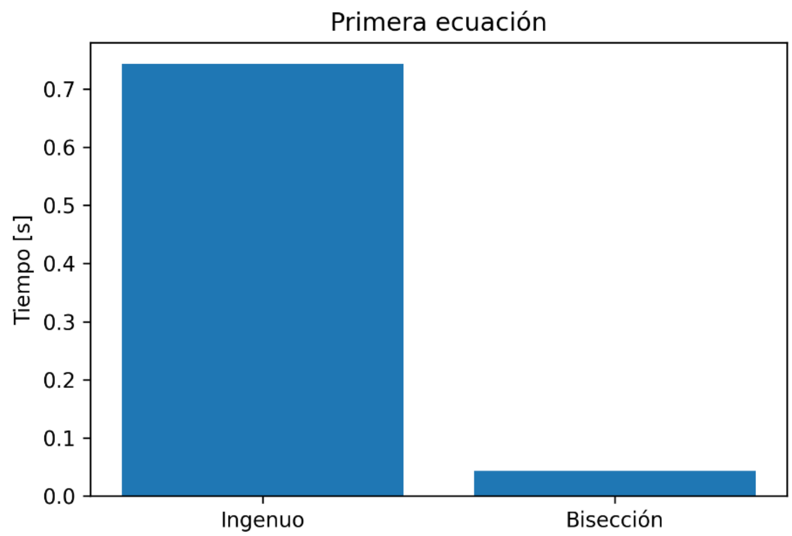

Durante este proceso de pruebas, se realizaron pruebas de tiempo haciendo uso de la librería time para saber el tiempo de ejecución

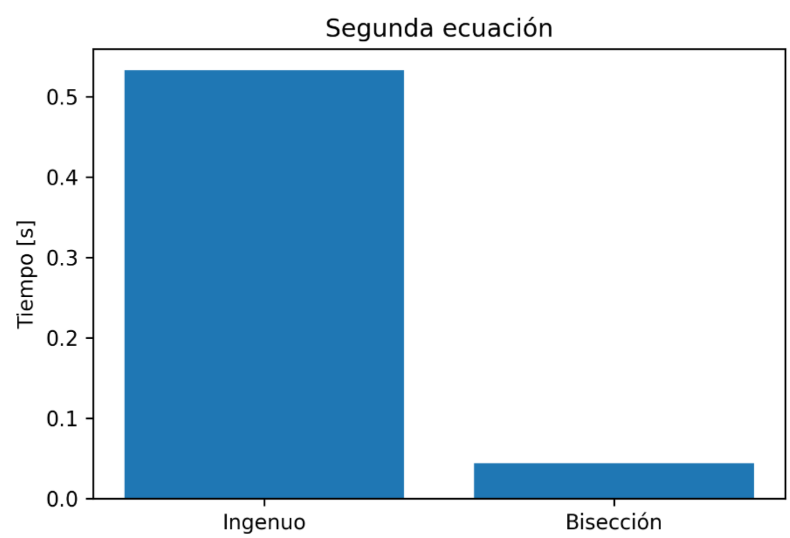
**Resultados**

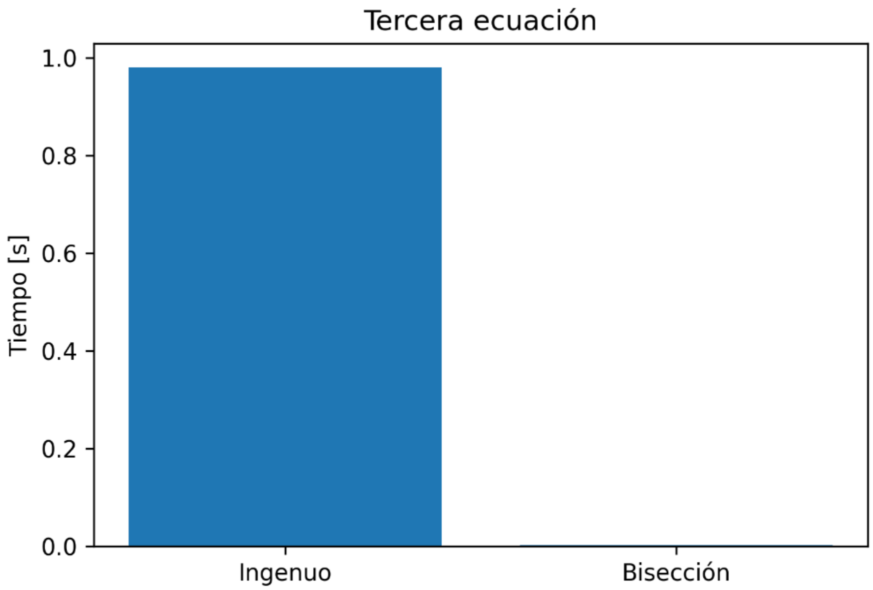
Como primer resultado, es importante resaltar, que el método de bisección funciona muy bien para la mayoría de grados de ecuaciones, pero en las parábolas (de grado 2), como no existen cambios en los valores de y, el código interpreta esto como que no existen raíces en el intervalo dado.

También es importante resaltar, que, para el método de la bisección, solo puede obtener una raíz en cada proceso, ya que cada que va dividiendo el rango, se va acercando a un solo punto. Esto es importante resaltarlo porque para el caso de la tercera ecuación, que tiene 3 raíces reales (-1,0,1) el programa solo encuentra una raíz real, mientras que, en el caso del método ingenuo, puede obtener las 3 pero el tiempo de ejecución es muy alto en comparación al método de la bisección.

A continuación, se muestran las gráficas comparando los tiempos de ejecución en los 3 casos de prueba.






**Conclusión**

Ambos métodos encuentran las raíces de las ecuaciones, pero en el caso del método ingenuo, podemos obtener las raíces teniendo un tiempo de ejecución demasiado alto, si disminuimos el valor de la tolerancia, el tiempo crece demasiado. Del otro lado, tenemos el método de la bisección, el cual es demasiado eficaz, pero se necesitaría hacer una modificación para que pudiera converger a más raíces para así lograr una mejora significativa frente al método de la bisección.

**Referencias**

Chapra, S., & Canale, R. (2011). METODOS NUMERICOS PARA INGENIEROS (Spanish Edition) (6th ed.). McGraw-Hill Interamericana de España S.L.