

## Introducción

¿Sabes como calcula una calculadora la raíz cuadrada de un valor, por ejemplo  $\sqrt{2}$ ? Se parte del hecho de que esta operación es equivalente a resolver la siguiente ecuación algebraica

$$x^2 = 2$$

Y de allí se procede a resolverla; para lo cual se debe igualar la ecuación a cero.

Entonces ahora vamos a generalizar esta situación considerando una simple ecuación no lineal  $f: \mathbb{R} \rightarrow \mathbb{R}$ . Se tiene entonces, una función escalar de un argumento escalar, y se pretende encontrar sus raíces. Es decir, hallar un  $x_0$  tal que  $f(x_0) = 0$ .

Obviamente, para resolver este problema se necesita especificar aspectos adicionales ya que surgen las siguientes preguntas:

- ¿La función posee realmente raíces?
- ¿Se está buscando una o todas las raíces que la función pueda tener? Por ejemplo, ¿Cuántas raíces tiene la ecuación  $x - \text{sen}(x) = 1$ ?
- ¿Se pretende hallar solo raíces reales o también complejas? (Por ejemplo, los polinomios son funciones no lineales que pueden tener raíces complejas)

Resulta claro que se necesita especificar el problema. Este aspecto concierne al especialista del dominio del problema. Entonces, suponga que de alguna manera ese trabajo se ha realizado, y como fruto de este se sabe que la función tiene una raíz y debe concentrarse en hallarla.

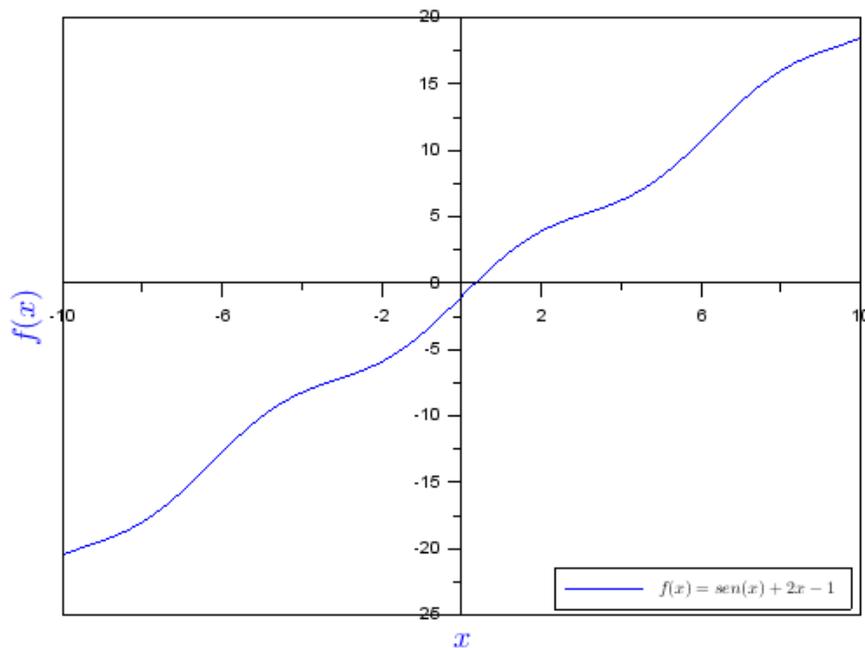
Ahora bien, en el mejor de los casos es posible determinar el valor de la raíz usando simplemente operaciones algebraicas para “despejar  $x$ ”. Sin embargo, muchas funciones adquieren una forma tan particular que esta operatoria no será posible de aplicar. Por ejemplo, intente despejar la variable independiente en esta ecuación

$$\text{sen}(x) + 2x - 1 = 0$$

¿Pudo? Sin embargo, sí que tiene una raíz, si observamos el siguiente código en Scilab podremos determinar que permite visualizar el gráfico de la función en el intervalo  $-10 \leq x \leq 10$

```
1 x=[-10:0.01:10];
2 y=sin(x)+2*x-1;
3 plot(x,y)
4 xgrid(0)
5 //manejador de los ejes de manera manual
6 ejes=gca();
7 //borramos los ejes por defecto
8 set(ejes,'axes_visible',['off','off','on'])
9 //visualizamos el nuevo eje x-e-y
10 drawaxis(x=[-10,10,5],y=0,dir='d',tics='r')
11 drawaxis(x=0,y=[-25,20,9],dir='l',tics='r')
12 //agregamos titulos a los ejes
13 xlabel("$x$", "font_size",4,"color","blue")
14 ylabel("$f(x)$", "font_size",4,"color","blue")
15 //mostramos una leyenda con la forma de la función
16 legend("$f(x)=\text{sen}(x)+2x-1$", "font_size",4,"color","blue",pos="in_lower_right")
```

Que genera el siguiente resultado



Este tipo de situaciones aparecen de manera ubicua en diversos problemas de la Ciencia e Ingeniería. Es decir que existen una gran variedad de problemas de ingeniería cuya solución pasa por determinar las raíces de una ecuación algebraica.

Hallar la raíz de una ecuación no lineal puede llegar a ser imposible de resolver en forma analítica. En estos casos (y en aquellos en los cuales la expresión a resolver es tan complicada que no es práctica utilizarla) usualmente el ingeniero estará obligado a utilizar métodos numéricos.

En las secciones posteriores nos centraremos en la siguiente cuestión:

¿Qué significa encontrar la raíz de una función de forma numérica?

Y en menor medida en

¿Qué significa en un ordenador que algo sea igual a cero en coma flotante?

Los métodos numéricos para este tipo de problemas se clasifican tradicionalmente en dos tipos: **los métodos cerrados** (que garantizan la convergencia si se conoce un intervalo  $[a, b]$  que contenga la raíz. Además, la función en ese intervalo debe ser monótona y continua) y **los métodos abiertos** (que requieren una aproximación inicial lo suficientemente cercana a la raíz deseada. Conseguir esto genera un trabajo extra, pero en compensación el método convergerá más rápido que el método cerrado). Independientemente del tipo de método a aplicar siempre es conveniente previamente utilizar el Método Gráfico para aproximar la raíz; por lo cual iniciaremos nuestro viaje por esta técnica para luego enmarcarnos en el estudio de los métodos más comunes que ofrecen cada una de las clasificaciones previamente indicadas.

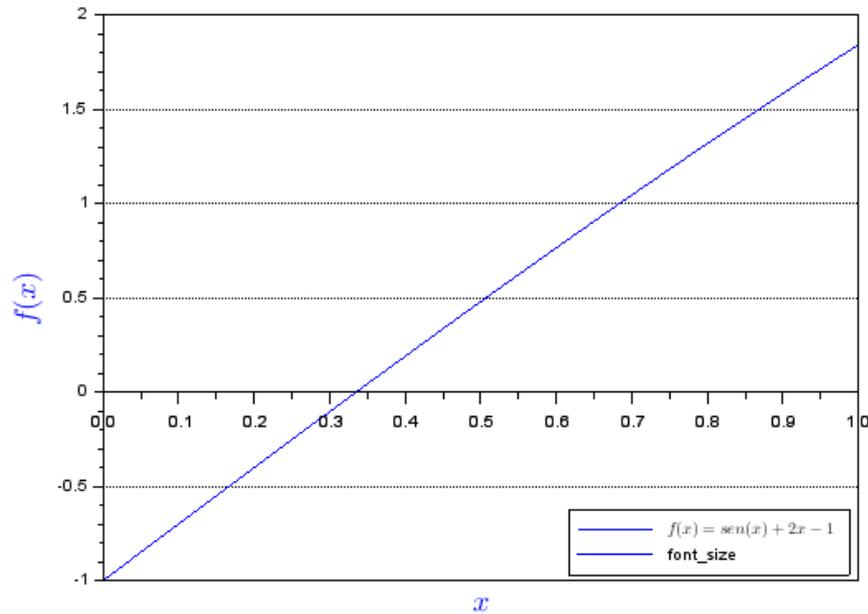
## El Método Gráfico

El método gráfico es útil por varias razones:

1. Permite determinar una primera aproximación suficientemente cercana de la raíz.
2. Si graficamos cada aproximación generada podemos estudiar cómo evoluciona el comportamiento particular del método respecto de la función sobre la cual se desea hallar una raíz.

### Ejemplos de uso del Método Gráfico con Scilab

- 1) Graficar para determina una raíz aproximada para la función  $\text{sen}(x) + 2x - 1 = 0$   
 Si se observa el gráfico anterior podemos deducir que la raíz se halla en  $0 < x < 1$ , por lo cual, si refinamos el gráfico, quedaría de esta manera



Con estos datos podemos establecer que entre la raíz se halla en el intervalo  $[0,3; 0,35]$ . Entonces es posible generar en Scilab una tabla que permita visualizar en ese intervalo posibles valores de la función para determinar una mejor aproximación. Por ejemplo, para este caso podría ser algo similar a esto

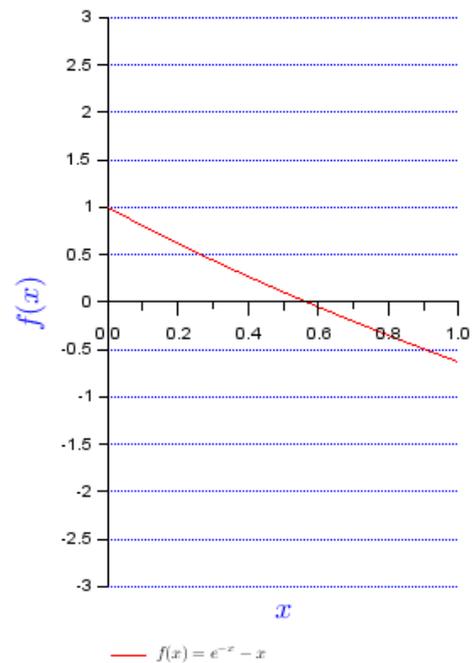
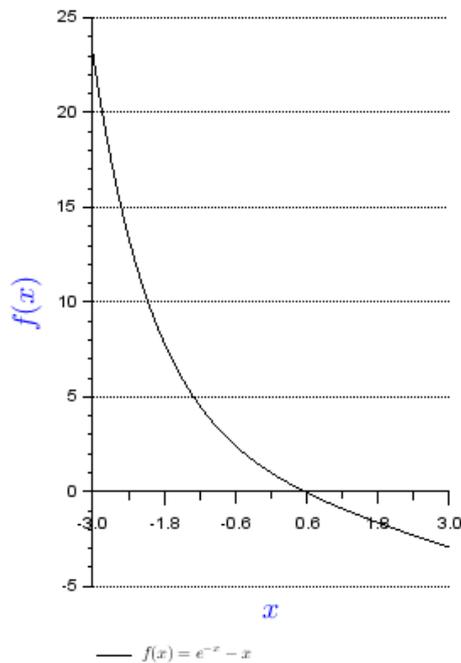
```
printf("x\t\tf(x)\n");
for i=0.3:0.01:0.35
    fx=sin(i)+2*i-1;
    printf("%f\t\t%f\n",i,fx);
end
```

Lo cual generará la siguiente salida

x	f(x)
0.300000	-0.104480
0.310000	-0.074941
0.320000	-0.045433
0.330000	-0.015957
0.340000	0.013487
0.350000	0.042898

Como puede observar el método gráfico combina la parte gráfica con pruebas en menores rango con el objetivo de mejorar la aproximación, dado que donde se produce el cambio del signo representa una aproximación de raíz o también si se lo desea una acotación del intervalo donde se halla la raíz. En este caso en la raíz se halla en el intervalo  $[0,33; 0,34]$

- 2) Graficar para determinar una raíz aproximada para la función  $f(x) = e^{-x} - x$   
 De manera similar que el anterior, pero en este caso se grafica en subgráficos. En el primer gráfico se puede observar que la función posee una raíz. Luego se crea un segundo gráfico con intervalos más restringidos para poder indicar el valor inicial de la raíz de una manera más precisa. Como se puede observar un posible valor de la raíz podría ser 0,55; o en su defecto definir un intervalo entre  $[0,56;0,57]$



Se hace notar que el siguiente es un extracto, donde se puede visualizar donde cambia el signo.

0.540000	0.042748
0.550000	0.026950
0.560000	0.011209
0.570000	-0.004475
0.580000	-0.020102
0.590000	-0.035673
0.600000	-0.051188

Finalmente se comparte el código utilizado

```

1 x=[-3:0.1:3];
2 y=exp(-x)-x;
3
4 subplot(121)
5
6 plot2d(x,y,leg="$f(x)=e^{-x}-x$")
7 ejes=gca();
8 set(ejes,'axes_visible',['off','on','on'])
9 drawaxis(x=[-3,3,5],y=0,dir='d',tics='r')
10 xgrid(1)
11 xlabel("$x$", "font_size",4,"color","blue")
12 ylabel("$f(x)$", "font_size",4,"color","blue")
13 subplot(122)
14
15 plot2d(x,y,5,leg="$f(x)=e^{-x}-x$",rect=[0,-3,1,3])
16 ejes=gca();
17 set(ejes,'axes_visible',['off','on','on'])
18 drawaxis(x=[0,1,5],y=0,dir='d',tics='r')
19
20 xgrid(2)
21 xlabel("$x$", "font_size",4,"color","blue")
22 ylabel("$f(x)$", "font_size",4,"color","blue")
23
24 printf("x\t\tf(x)\n");
25 for i=0:0.01:1
26     fx=exp(-i)-i;
27     printf("%f\t\t%f\n",i,fx);
28 end
    
```

## Métodos cerrados

En esta sección estudiaremos los denominados métodos iterativos para resolver ecuaciones no lineales. Estos métodos se basan en la siguiente idea: parten de una o más aproximaciones iniciales a la raíz (comúnmente denominada  $x_0$ ), luego se aplica un algoritmo que genera una nueva aproximación. Este proceso se repite generándose entonces una secuencia de aproximaciones  $x_0, x_1, x_2, \dots$  que presumiblemente converge hacia la raíz deseada.

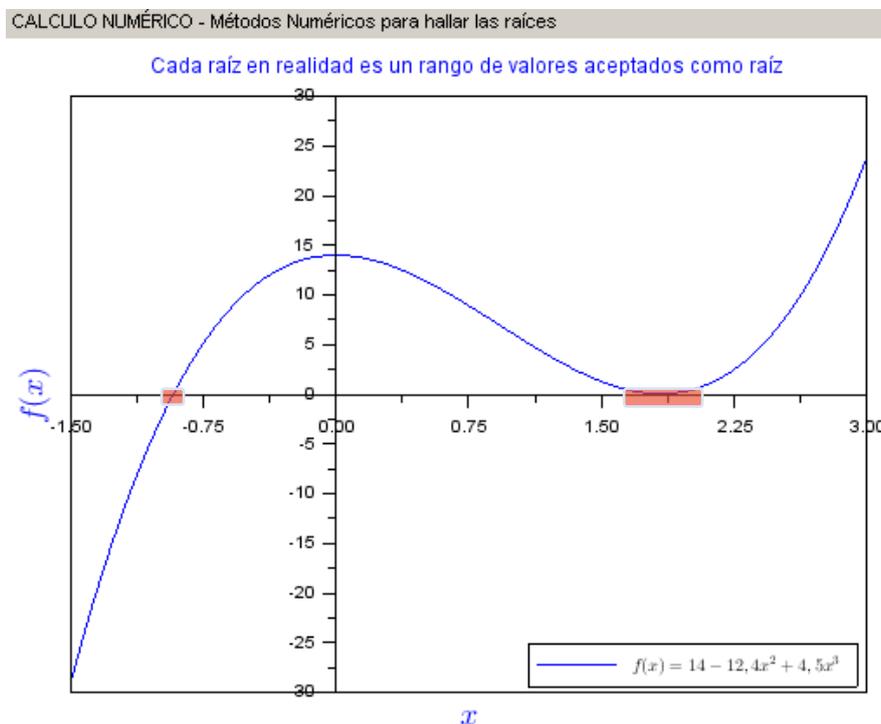
En particular se abordará la determinación de las raíces de una ecuación algebraica no lineal en un intervalo cerrado  $[a, b]$ , esto es, hallar un valor  $x / f(x_*) = 0$ .

En general, esta tarea requiere un conocimiento general de la función que se está analizando, pues no existe ningún método universal que nos proporcione el resultado automáticamente. Por este motivo el objetivo inicial es poner a la disposición de ustedes un conjunto de métodos numéricos para determinar una sola raíz, aunque pueden aplicarse a la determinación de más raíces subdividiendo el dominio de la función en intervalos que contengan una sola. Recuerde además que, geoméricamente hablando, hallar la raíz de una ecuación consiste en encontrar el punto de corte de la gráfica de la función con el eje de las abscisas.

¿Tiene sentido intentar aproximar la raíz por medio de un ordenador que de por sí generará un error?

La aritmética del ordenador no es exacta, no es precisa. Sin embargo, ciertamente es posible acotar ese error. Por otro lado, realizar los cálculos de forma manual probablemente podría generar errores. Entonces, si se utiliza el ordenador como herramienta científica para realizar los cálculos, se asume el cálculo de  $f(x)$  con cierta incertidumbre representada por el incremento  $\Delta$  (donde su principal aportante es el error de redondeo).

En este contexto, si se busca la raíz de una función de forma numérica lo que se hace en realidad es determinar un intervalo alrededor de la misma y considerar todo valor en ese intervalo como una raíz válida. Observe esta situación en el siguiente gráfico



Evidentemente esta incertidumbre provoca que el rango del intervalo es indeterminado. Para poder expresar este rango en un intervalo determinado podemos expresar la función de la siguiente manera: por medio de los métodos numéricos se calcula el valor de la función con una determinada incertidumbre  $\delta$  que representa la cota de error. Esto es

$$\exists \delta \mid |f(x)| < \Delta \forall |x - x_0| \leq \delta$$

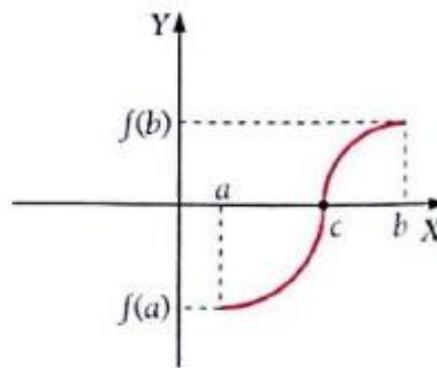
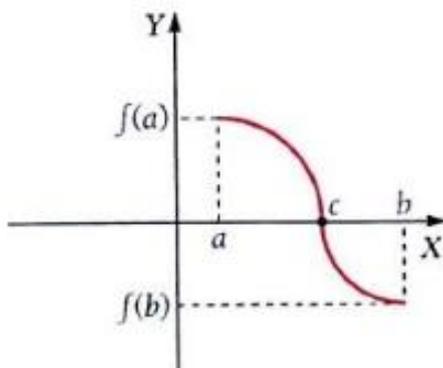
Entonces, cuando tenemos esos intervalos donde el valor de la función es más pequeño que Delta, cualquier punto en ese intervalo pueden declararse una raíz.

Por esto es válido expresar que los métodos cerrados requieren un intervalo  $[a, b]$  en el cual se presume o asegura que contiene un valor que cumple  $f(x) = 0$ , que es lo mismo que decir que ese valor es la raíz. Si el valor de la raíz realmente existe, estos métodos siempre convergerán, pero la velocidad de convergencia puede resultar excesivamente lenta. Finalmente, estos métodos se basan en la aplicabilidad del Teorema de Bolzano

### Teorema de Bolzano

Si  $f(x)$  es una función continua en un intervalo  $[a, b]$  y además es monótona, es decir  $f(a) \cdot f(b) < 0$ ; entonces  $f(x)$  tiene una raíz en el intervalo  $(a, b)$ .

Este teorema refleja el hecho evidente de que, si una función es continua y monótona con signos diferentes en los extremos, entonces necesariamente tiene que pasar por una raíz en algún punto intermedio dentro del intervalo considerado, lo cual puede ser demostrado fácilmente de forma gráfica definiendo diferentes funciones que cumplan los requisitos y observando el resultado, por ejemplo:

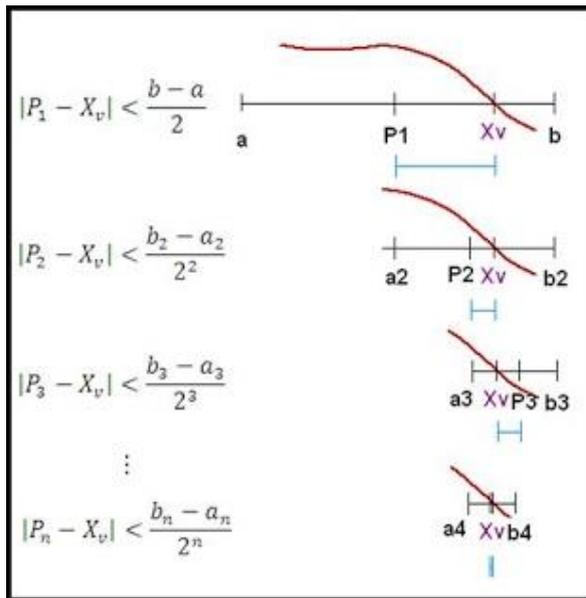


### Método de Bisección (o del Intervalo Medio o del Corte Binario)

Este método procede de manera sencilla: asumiendo que en el intervalo  $[a, b]$  se cumple el Teorema de Bolzano, entonces ese intervalo se dividirá sistemáticamente desechando sucesivamente la sección que no contenga la raíz. Para lograr esto, se vuelve a aplicar el Teorema de Bolzano para determinar en cuales de los dos segmentos obtenidos se halla la raíz. Así, en algún momento el punto del corte coincidirá con la raíz buscada. Estos tipos de procedimientos se conocen con el nombre de Búsqueda Incremental de la Raíz.

En particular cuando la búsqueda incremental consiste en dividir sistemáticamente el intervalo que contiene a la raíz por la mitad (proceso de bisección) y luego evaluar si ese punto de corte es la raíz, se dice que el método aplicado es el de la Bisección.

El análisis gráfico de la aplicación del método sería el siguiente:



Dado un intervalo  $[a, b]$  donde se comprueba el Teorema de Bolzano. Se busca la raíz  $X_v$ . Se divide el intervalo original en dos subintervalos:  $[a, P_1]$  y  $[P_1, b]$ . Si  $P_1$  no es la raíz por Bolzano determinamos que  $[P_1, b]$  la contiene.

Se vuelve a dividir el subintervalo que contiene la raíz quedando  $[P_1 = a_2, P_2]$  y  $[P_2, b_2 = b]$ .

Si el punto de corte no es la raíz se vuelve a aplicar el proceso. Observe que el intervalo se hace cada vez más pequeño, más precisamente cada vez que se aplica la división, el intervalo resultante tiene una longitud igual a la mitad del anterior.

El algoritmo del Método de la Bisección se puede expresar de la siguiente manera:

- 1) Verificar el teorema de Bolzano ( $f(a) \cdot f(b) < 0$ ). Si no se verifica el teorema no significa que la función no posea raíces [no las posee en el intervalo seleccionado]. Por tanto, se sugiere seleccionar otro intervalo  $[a, b]$ .
- 2) Establecer tolerancia de error  $\xi$ .
- 3) Calcular una aproximación a la raíz en el punto medio del intervalo:

$$x_i = \frac{a + b}{2}$$

- 4) Calcular  $f(x_i)$ . Si  $f(x_i) = 0$  entonces se ha hallado la raíz y por lo tanto se termina el algoritmo devolviendo el valor hallado
- 5) Calcular

$$f(a) \cdot f(x_i)$$

y verificar si

$$f(a) \cdot f(x_i) < 0$$

De cumplirse, significa que la raíz se encuentra en  $[a, x_i]$ . Entonces se reemplaza el extremo superior de la siguiente manera

$$b = x_i$$

- 6) Calcular

$$f(a) \cdot f(x_i)$$

y verificar si

$$f(a) \cdot f(x_i) > 0$$

De cumplirse, significa que la raíz se encuentra en  $[x_i, b]$ . Entonces se reemplaza el extremo inferior de la siguiente manera

$$a = x_i$$

- 7) Verificar el criterio de parada. Esto es calcular

$$\left| \frac{x_i - x_{i-1}}{x_i} \right| < \xi$$

Si esta condición se verifica quiere decir que la raíz aproximada es aceptable (en términos de la tolerancia de error definida en el paso 2). Así el proceso termina.

Por el contrario, si no se verifica la condición de parada se vuelve al paso 3).

En el algoritmo presentado se ha introducidos un concepto muy importante:

Criterio de parada: ¿Qué sucedería si la cantidad de iteraciones que se necesitan para hallar la raíz provoca que la aplicabilidad del método sea inaceptable? Sería interesante contar con un mecanismo que permita aceptar una aproximación. Esto es lo que se busca al aplicar la fórmula:

$$\left| \frac{x_i - x_{i-1}}{x_i} \right| < \xi$$

Donde

$x_i$  = valor aproximado hallado en la actual iteración

$x_{i-1}$  = valor aproximado hallado en la anterior iteración

Se aplica el valor absoluto debido a que en este tipo de problemas solo nos interesa la magnitud del error cometido. Entonces en el criterio de parada se calcula el error relativo generado en la actual iteración.

En muchas situaciones también se puede requerir conocer el error relativo porcentual de la iteración, es decir

$$\left| \frac{x_i - x_{i-1}}{x_i} \right| \cdot 100\%$$

### Ejemplo Práctico

Obtenga, si existe; la raíz de la siguiente función:  $f(x) = e^{-x} - x$  con una tolerancia de error  $\xi = 0,001$ .

*Respuesta*

Luego de usar el método gráfico se pudo observar que el intervalo propuesto es usando el algoritmo en el intervalo  $[0,56 ; 0,57]$ .

Si se propone desarrollar el método de manera manual, suele ser conveniente armar una tabla con la información resultante de la aplicación del algoritmo en todas las iteraciones

							$\xi$	
$i$	$a$	$x_i$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?

Con esta tabla como base se siguen todos los pasos del Algoritmo.

- Determinar si se cumple el Teorema de Bolzano en el intervalo  $[a, b]$ .  
 Sabemos que se cumple el teorema gracias a la previa aplicación del método gráfico, pero se realiza el mismo; en ocasión del ejemplo. Entonces con

$$a = 0,56 \text{ y } b = 0,57$$

$$f(a) = 1 \text{ y } f(b) = -0,10653066$$

Se anotan los resultados en la tabla y se observa que se cumple el Teorema porque los valores de las funciones en los extremos tienen diferente signo

							$\xi$	
$i$	$a$	$x_i$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?
1	0,56		0,57	0,011209		- 0,004475		

2) Establecer la tolerancia de error

								$\xi$	0,001
$i$	$a$	$x_i$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?	
1	0,56		0,57	0,011209		- 0,004475			

3) Calcular la aproximación de la raíz. Recuerde que

$$x_i = \frac{a + b}{2}$$

								$\xi$	0,001
$i$	$a$	$x_i = \frac{a + b}{2}$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?	
1	0,56	0,565	0,57	0,011209		- 0,004475			

4) Verificar si  $x_i$  es la raíz. En caso de serlo se detiene el proceso

								$\xi$	0,001
$i$	$a$	$x_i = \frac{a + b}{2}$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?	
1	0,56	0,565	0,57	0,011209	0,003360	- 0,004475			

Como se puede observar no es la raíz, entonces continuamos con el algoritmo

5) Verificar el subintervalo donde se halla la raíz usando el Teorema de Bolzano y sustituir el extremo correspondiente con el valor  $x_i$

								$\xi$	0,001
$i$	$a$	$x_i = \frac{a + b}{2}$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?	
1	0,56	0,565	0,57	0,011209	0,003360	- 0,004475			
	0,565		0,57	0,003360		- 0,004475			

Como se observa  $f(x_i) \cdot f(b) < 0$  por lo tanto  $a = x_i$  mientras que el otro extremo se mantiene igual.

Observe además que en la tabla se han unido la verificación de los pasos 5 y 6 del algoritmo por cuanto resulta directo (si la raíz no está en un intervalo, se halla en el otro)

6) Verificar el criterio de parada. Para ello se calcula

$$\left| \frac{x_i - x_{i-1}}{x_i} \right|$$

								$\xi$	0,001
$i$	$a$	$x_i = \frac{a + b}{2}$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\left  \frac{x_i - x_{i-1}}{x_i} \right $	¿Continuar?	
1	0,56	0,565	0,57	0,011209	0,003360	- 0,004475	1	SI	
	0,565		0,57	0,003360		- 0,004475			

Observe que el valor del error relativo obtenido es mayor que la tolerancia de error, por lo tanto, el criterio de parada no se cumple y se debe continuar

7) En base a la forma que hemos llenado la tabla en los pasos anteriores completamos la misma hasta hallar una iteración en la cual se halle la raíz o se cumpla la tolerancia de error.

$i$	$a$	$x_i = \frac{a+b}{2}$	$b$	$f(a)$	$f(x_i)$	$f(b)$	$\frac{ x_i - x_{i-1} }{x_i}$	$\xi$	¿Continuar?
1	0,56	0,565	0,57	0,011209	0,003360	- 0,004475	1	0,001	SI
2	0,565	0,5675	0,57	0,003360	-0,000559	- 0,004475	0,004405		SI
3	0,565	0,566250	0,5675	0,003360	0,001400	-0,000559	0,002208		SI
4	0,566250	0,566875	0,5675	0,001400	0,000420	-0,000559	0,001103		SI
5	0,566875	0,567187	0,5675	0,000420	-0,000069	-0,000559	0,000551		NO

Entonces resulta que la raíz aproximada es 0,567187

A continuación, se presenta una codificación del método en Scilab altamente mejorable

```

0002 //Definición de la función
0003 def('y=f(x)', 'y=exp(-x)-x;');
0004 //Se inicializan la coa superior e inferior del intervalo
0005 xa=.56;
0006 xb=.57;
0007 // Se define un error muy grande (Scilab requiere inicializar var)
0008 error=100;
0009 // Se define e inicializa un contador de iteraciones
0010 it=1;
0011 // xraiz representa a las aproximaciones de la raíz obtenidas
0012 xraiz=0;
0013 //Tolerancia de error
0014 tolerancia=0.001;
0015 // Imprime por consola los encabezados de la tabla de iteraciones
0016 mprintf('Iteraciones\ta\txi\tb\tf(a)\tf(xi)\tf(b)\tError relativo\n')
0017 while error>tolerancia
0018     //Almacena el valor de la aproximación anterior
0019     aux=xraiz;
0020     //Se evalúa la función en la cota inferior
0021     fxa=f(xa);
0022     //Se evalúa la función en la cota superior
0023     fxb=f(xb);
0024     //Se aplica la fórmula de recurrencia y se obtiene una aproximación
0025     xraiz=(xa+xb)/2;
0026     // Se evalúa la función en la raíz aproximada
0027     fxr=f(xraiz);
0028     // lo usaremos para evaluar si xraiz es la raíz o en que
0029     // intervalo se halla
0030     producto=fxa*fxr;
0031     if producto == 0 then //significa que fxr es la raíz
0032         error = 0;
0033     else
0034         // error relativo porcentual en la iteración
0035         error=abs(xraiz-aux)/abs(xraiz);
0036         mprintf('%i\t\tf\t\tf\t\tf\t\tf\t\tf\t\tf\n', it, xa, xraiz, xb, fxa, fxr, fxb, error)
0037         if producto<0
0038             xb=xraiz; // la nueva cota superior es xi
0039         elseif producto>0
0040             xa=xraiz; // la nueva cota inferior es xi
0041         end
0042     end
0043     // mostramos la salida
0044     //mprintf('%i\t\tf\t\tf\t\tf\t\tf\t\tf\t\tf\n', it, xa, xraiz, xb, fxa, fxr, fxb, error)
0045     it=it+1;
0046 end
    
```

Que genera la siguiente salida

Iteraciones	a	xi	b	f(a)	f(xi)	f(b)	Error relativo
1	0.560000	0.565000	0.570000	0.011209	0.003360	-0.004475	1.000000
2	0.565000	0.567500	0.570000	0.003360	-0.000559	-0.004475	0.004405
3	0.565000	0.566250	0.567500	0.003360	0.001400	-0.000559	0.002208
4	0.566250	0.566875	0.567500	0.001400	0.000420	-0.000559	0.001103
5	0.566875	0.567187	0.567500	0.000420	-0.000069	-0.000559	0.000551

### Comentarios del Método de la Bisección

El método va generando una sucesión de intervalos cerrados, siendo la longitud de cada uno de ellos la mitad de la del anterior. El punto medio de  $[a_n, b_n]$  aproxima a la raíz con un error menor o igual que  $\frac{b_0 - a_0}{2^n}$ .

En esta breve descripción queda claro que la raíz  $\alpha$  y la aproximación  $x_{n+1}$  pertenecen a  $I_n$  y a  $I_{n+1}$ ; que la longitud de  $I_{n+1}$  es la mitad que de la de  $I_n$  y que el error absoluto de  $x_{n+1}$  está acotado por la longitud de  $I_{n+1}$ . Más concretamente para todo  $n$  tenemos que:

$$b_n - a_n = \frac{b_{n-1} - a_{n-1}}{2} = \frac{(b_0 - a_0)}{2^n}$$

lo que nos permite relacionar la longitud del intervalo  $I_n$  con el intervalo inicial  $I_0 = [a_0, b_0]$ . Además, tenemos que:

$$|x_{n+1} - \alpha| \leq |b_{n+1} - a_{n+1}| = \frac{|b_0 - a_0|}{2^{n+1}}$$

Y por lo tanto el método siempre es convergente.

El criterio de parada puede ser el error absoluto o el error relativo. Para una precisión  $\xi > 0$  prefijada el criterio de parada para el error relativo sería

$$\left| \frac{x_{i+1} - \alpha}{x_{i+1}} \right| \leq \left| \frac{b_{n+1} - a_{n+1}}{b_{n+1}} \right| < \xi$$

siempre que  $a_{n+1}, b_{n+1}$  y  $\alpha$  sean distintos de 0.

Con toda la información indicada en este apartado y el hecho de que el método se basa en el Teorema de Bolzano se puede estimar el número de iteraciones que harían falta para cumplir con la condición de parada

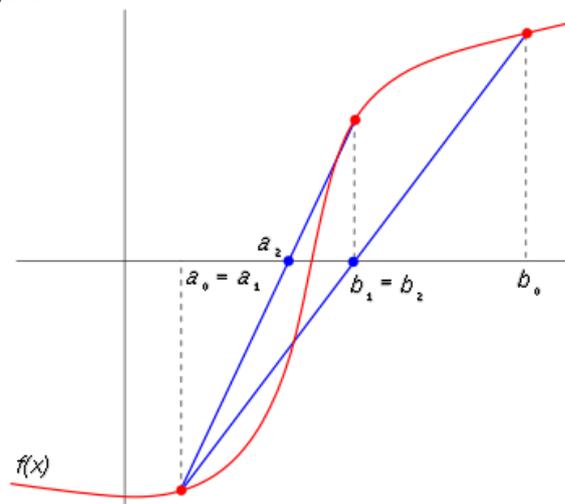
$$\frac{b_0 - a_0}{2^{n+1}} < \xi \Rightarrow n + 1 > \frac{\log(|b_0 - a_0|) - \log(\xi)}{\log(2)}$$

Si suponemos que  $|b_0 - a_0| \approx 1$ , entonces el número de iteraciones dependerá únicamente de  $\xi$  y como  $2^{10} = 1024 \approx 10^3$ , necesitaremos unas 10 iteraciones cada vez que dividamos por 1000. Lo cual es coherente con el resultado de los ejemplos vistos hasta ahora.

### Método Regula Falsi (o de la Falsa Posición)

Este método aprovecha la idea de unir los puntos con una línea recta. La intersección de esta línea con el eje x proporciona una estimación de la raíz. El reemplazo de la curva por una línea recta da una "posición falsa" de la raíz, de aquí el nombre de método de la regla falsa o en latín regula falsi.

Observe la siguiente figura



La recta que une los puntos  $(a, f(a))$  con  $(b, f(b))$  puede ser descripta de la siguiente manera:

$$y - f(a) = \frac{f(b) - f(a)}{b - a}(x_1 - a)$$

Dado que  $x_1$  es la raíz entonces

$$x_1 = a - \frac{f(a) \cdot (b - a)}{f(b) - f(a)}$$

La cual se puede generalizar para obtener las aproximaciones a la raíz dentro del método iterativo.

El algoritmo del Método Regula Falsi se puede expresar de la siguiente manera:

- 1) Verificar el teorema de Bolzano ( $f(a) \cdot f(b) < 0$ ). Si no se verifica el teorema no significa que la función no posea raíces [no las posee en el intervalo seleccionado]. Por tanto, se sugiere seleccionar otro intervalo  $[a, b]$ .
- 2) Establecer tolerancia de error  $\xi$ .
- 3) Calcular una aproximación a la raíz en el punto medio del intervalo:

$$x_i = a - \frac{f(a) \cdot (b - a)}{f(b) - f(a)}$$

- 4) Calcular  $f(x_i)$ . Si  $f(x_i) = 0$  entonces se ha hallado la raíz y por lo tanto se termina el algoritmo devolviendo el valor hallado
- 5) Calcular

$$f(a) \cdot f(x_i)$$

y verificar si

$$f(a) \cdot f(x_i) < 0$$

De cumplirse, significa que la raíz se encuentra en  $[a, x_i]$ . Entonces se reemplaza el extremo superior de la siguiente manera

$$b = x_i$$

- 6) Calcular

$$f(a) \cdot f(x_i)$$

y verificar si

$$f(a) \cdot f(x_i) > 0$$

De cumplirse, significa que la raíz se encuentra en  $[x_i, b]$ . Entonces se reemplaza el extremo inferior de la siguiente manera

$$a = x_i$$

- 7) Verificar el criterio de parada. Esto es calcular

$$\left| \frac{x_i - x_{i-1}}{x_i} \right| < \xi$$

Si esta condición se verifica quiere decir que la raíz aproximada es aceptable (en términos de la tolerancia de error definida en el paso 2). Así el proceso termina.

Por el contrario, si no se verifica la condición de parada se vuelve al paso 3).

### **Comentarios del Método Regula Falsi**

Comparte con el método del intervalo medio la ventaja de converger en cualquier circunstancia, y su principal desventaja es la de encontrar sólo un resultado en el caso de raíces múltiples. Sin embargo, este método es más veloz que su análogo, el del intervalo medio o bisección. Adicionalmente, pueden aparecer extremos fijos (como el mostrado en la figura que se usó para

describir gráficamente el funcionamiento de este método), en donde uno de los extremos de la sucesión de intervalos no se mueve del punto original, por lo que las aproximaciones convergen a la raíz exacta solamente, por un lado. En cuyo caso, no siempre, pueden presentarse situaciones de convergencia rápida.

### Ejemplo Práctico

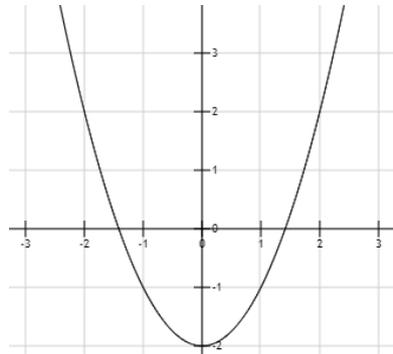
Obtenga la forma en la cual las calculadoras obtienen  $\sqrt{2}$ . Se supone que no se cuenta con calculadora y no puede aplicar el método tradicional para hallar el valor de raíces por diferenciales

#### Respuesta

Las funciones se representan de la forma  $y = f(x)$  por lo tanto podemos expresar el planteamiento del problema de la siguiente manera

$$x = \sqrt{2} \Rightarrow x^2 = 2 \Rightarrow x^2 - 2 = 0$$

Si aplicamos el método de Regula Falsi requerimos un intervalo. Podemos graficar la función para determinar el intervalo cerrado sobre el cual usaremos el método.



Como se puede observar tenemos una raíz en el intervalo  $[1, 1,5]$

Entonces ahora podemos aplicar el método de Regula Falsi para lo cual elegimos una tolerancia de error de 0,001

```
--> regula_falsi()
Iteraciones  a          xi          b          f(a)        f(xi)        f(b)        Error relativo
1            1.000000    1.400000    1.500000   -1.000000   -0.040000    0.250000    1.000000
2            1.000000    1.416667    1.400000   -1.000000    0.006944   -0.040000    0.011765
3            1.416667    1.414201    1.400000    0.006944   -0.000035   -0.040000    0.001743
4            1.416667    1.414214    1.414201    0.006944   -0.000000   -0.000035    0.000009
ans =
1.4142136
```



aproximación de la raíz

Sería interesante comparar los resultados obtenidos por el método de la Bisección y verificar cual converge más rápido y cuál sería la razón. Ya se habrá dado cuenta que el algoritmo del método Regula Falsi solo varía en la ecuación para obtener la