# RAICES DE FUNCIONES

## ASPECTOS GENERALES

La determinación de las condiciones en las cuales una función se anula, es una tarea habitual en el trabajo del ingeniero. Existe una gran cantidad de métodos numéricos para resolver este problema; sin embargo, se verá un solo método.

En este tipo de métodos numéricos se hace uso del teorema de Bolzano, que dice:

Sea $f$ una función real continua en un intervalo cerrado $[a,b]$ con $f(a)$ y $f(b)$ de signos contrarios. Entonces existe al menos un punto $c$ del intervalo abierto $(a, b)$ donde $f(c) = 0$.

Es importante recalcar que el teorema indica que existe al menos un punto interior, lo que implica que puede haber más puntos, siempre en cantidad impar, tal como se observa en la figura 1.

Los métodos para determinar los ceros de una función, trabajan sobre una raíz a la vez, por este motivo es importante dividir el intervalo en estudio en sub-intervalos, dentro de los cuales se encuentre una única raíz.

Esta tarea previa se hará visualmente. Por tal motivo se deberá graficar la función, y analizando el resultado se determinarán de manera grosera los distintos intervalos, tal que haya una única raíz en cada uno de ellos. Posteriormente se resolverá cada raíz por separado.

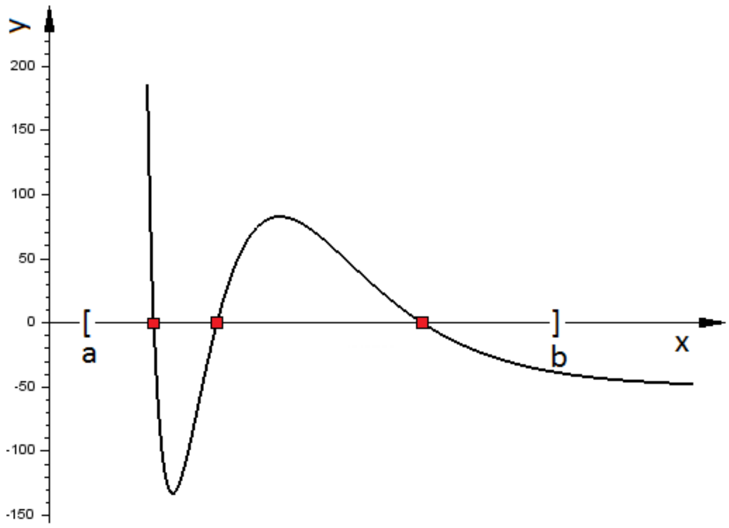


Fig.1.Ejemplo de múltiples raíces de una función continua en [a,b].

## MÉTODO DE LA REGULA FALSI

Sea la función dada en la figura 2, que contiene tres raíces reales distintas en el intervalo (0, 7).

Se desea calcular cada una de las mismas, para lo cual se subdivide en los tres subintervalos dados a continuación, asegurando de esta manera la existencia de una única raíz en cada uno de ellos:

$I_1$=[0.5, 1.5]

$I_2$=[1.5, 2.5]

$I_3$=[5.5, 6.5]

Se calculará la primera de las raíces en el intervalo $I_1$ haciendo uso del método de la regula falsi o de la falsa posición. Las dos restantes se las deja al lector como ejercicio.

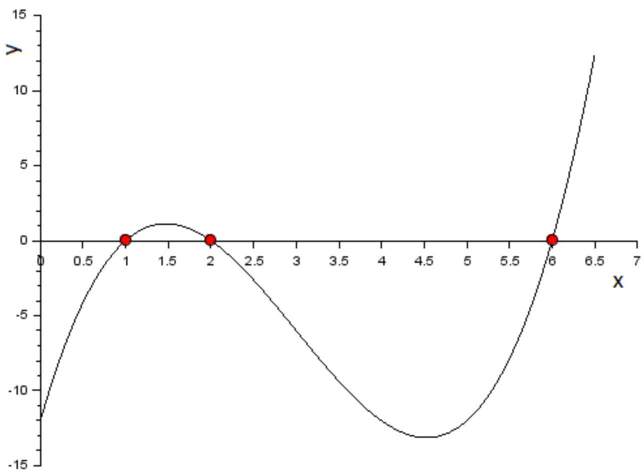


Fig. 2. Función con tres raíces reales distintas.

Para comenzar, se determina la recta que pasa por los puntos extremos del intervalo, de manera tal que al haber un cambio de signo en la función, la recta corta al eje de abscisas.

Sean las coordenadas de los puntos extremos (a, y(a)) y (b, y(b)), entonces la recta que pasa por ellos queda definida como

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

$$y - y(a) = m \cdot (x - a)$$

Con esta recta se calcula el punto de intersección con el eje de abscisas, resolviendo la ecuación para y=0, quedando

$$x_i = a - \frac{y(a)}{m}$$

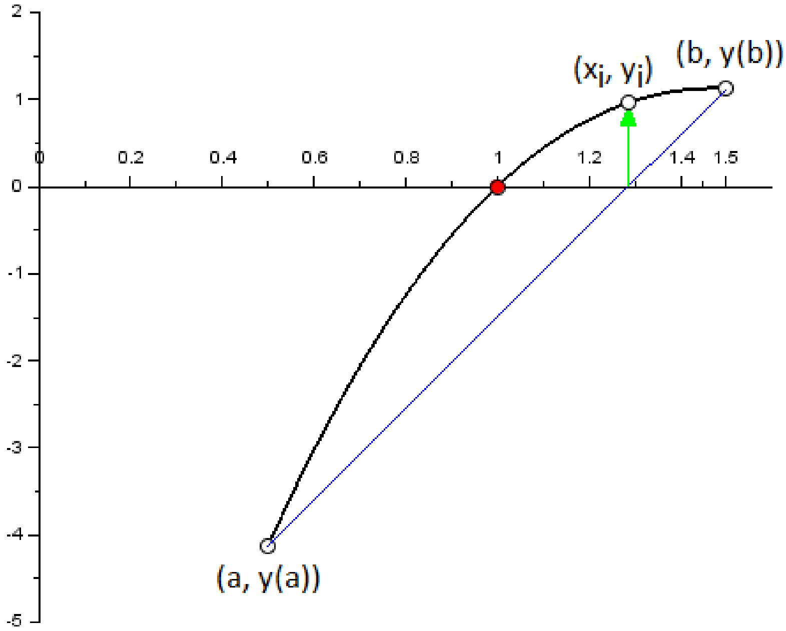


Fig. 3. Primera estimación con la recta secante.

Se calcula el valor de la función para este valor de x

$$y_i = y(x_i)$$

En la figura 3 se observa gráficamente este proceso, donde se aprecia como el nuevo punto se ha acercado a la raíz. Para realizar una nueva iteración, y mejorar la precisión del resultado, el punto obtenido $(x_i;\ y_i)$ reemplazará al punto que tiene el mismo signo funcional, en este caso será el punto $(x_b;\ y_b)$. Por lo tanto, corresponden realizar las siguientes asignaciones:

$b = x_i \qquad\qquad y(b) = y_i$

Una vez ajustado el nuevo intervalo, que resulta ser más estrecho que el original, se vuelve a repetir el procedimiento.

En la figura 4 se aprecia el segundo punto estimado, que se acerca aún más a la raíz.

El procedimiento se continua hasta lograr que el valor de la función $y_i$ cumpla con una tolerancia predeterminada, dada por

$$|y_i| < \varepsilon$$

En ese momento se establece el valor de $x_i$ como el valor de la raíz.

Los códigos para poder utilizar este método se muestran a continuación:

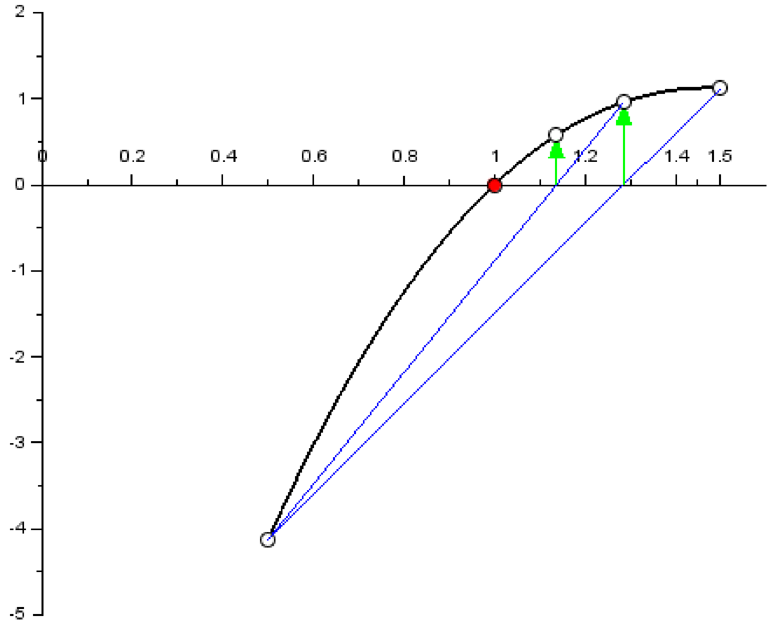


Fig. 4. Segunda estimación con la recta secante.

```
// Método de la Regula Falsi
//
// Variables de entrada
// a,b: intervalo del dominio de la función donde hay una única raíz
// tol: precisión con la que se desea conocer el resultado
// Variables de salida
// r: valor de la raíz
// er: código de error
//      0: no hubieron errores
//    801: los valores de abscisa son iguales
//    802: la tolerancia es nula o negativa
//    803: la función en a y b tienen el mismo signo
//    804: se excedió la cantidad de iteraciones máximas sin hallar una raíz
function [r, er]=regulafalsi(a, b, tol)
    er=0
    r=0
    if a==b then
        er=801
        return
    end
    if tol<=0 then
        er=802
        return
    end
    ya=func(a)
    yb=func(b)
    if sign(ya)==sign(yb) then
        er=803
        return
    end
    yi=tol
    n=0
    while abs(yi)>=tol & n<100
        m=(yb-ya)/(b-a)
        xi=a-ya/m
        yi=func(xi)
        if sign(yi)==sign(ya) then
            a=xi
            ya=yi
        else
            b=xi
            yb=yi
        end
        n=n+1
    end
    if abs(yi)<tol then
        r=xi
    else
        er=804
    end
endfunction
```

Al analizar la función se observan las siguientes líneas:

```
ya=func(a)
yb=func(b)
yi=func(xi)
```

donde se hace una llamada a una función denominada **func()** la cual contiene la expresión matemática de la función que está siendo analizada. Esto se hace así para mantener inalterado el código de la función **regulafalsi()**, ya que cualquier modificación se hace en **func()**.

Este tipo de estrategia es muy utilizada al escribir algoritmos de métodos numéricos, por este motivo la función `func()` o cualquier otra equivalente, debe ser escrita al principio del archivo de scinotes o al final, con el objeto que se pueda ubicar fácilmente.

**CASO DE ESTUDIO**

Sea una reacción química elemental, del tipo

$$A \rightarrow B$$

Tal que se carga un reactor con una concentración inicial de A igual a 3 mol/l. La concentración del reactivo evoluciona según se observa en la figura 5.

Si se desea saber en cuánto tiempo la concentración del reactivo alcanza los 0.5 mol/l, se trazaría una horizontal desde la ordenada 0.5, hasta la curva, y de allí se leería en abscisas el tiempo, en las unidades correspondientes, que sería cercano a 1.8 unidades de tiempo.

La curva de la figura 5 es del tipo exponencial decreciente, y es muy útil para representar el consumo de un reactivo.

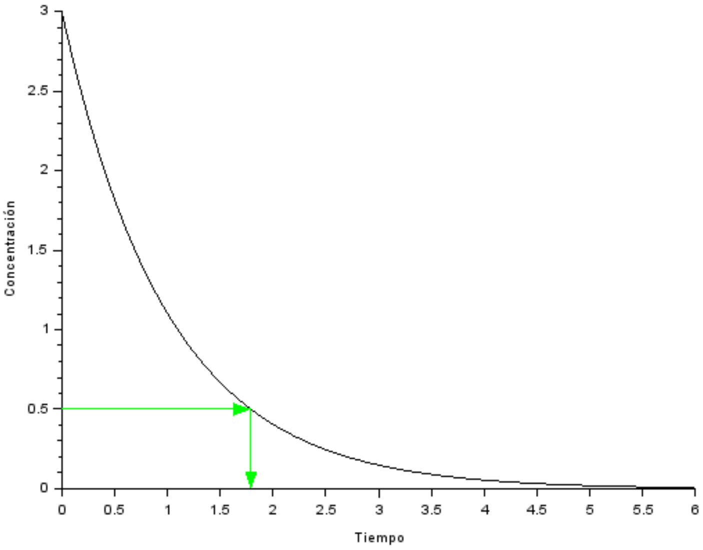

Fig. 5. Concentración de A en función del tiempo.

Para el ejemplo se utilizó la siguiente expresión:

$$C_A = 3 \cdot e^{-t}$$

Si se desea realizar un cálculo preciso, se puede representar la concentración final (0.5 mol/l) mediante una recta horizontal, tal como se observa en la figura 6.

Para poder determinar el punto de corte mediante la técnica de la regula falsi, se transforma la intersección de las dos curvas en una raíz de la primera, mediante la siguiente traslación vertical:

$$C_A = 3 \cdot e^{-t} - 0.5$$

En la figura 7 se aprecia como la curva queda 0.5 unidades desplazada hacia abajo (curva azul), transformando de esta manera el punto de intersección en una raíz.

Para poder utilizar el código de la regula falsi de Scilab dado anteriormente, se debe definir la función `func()` como se indica a continuación:

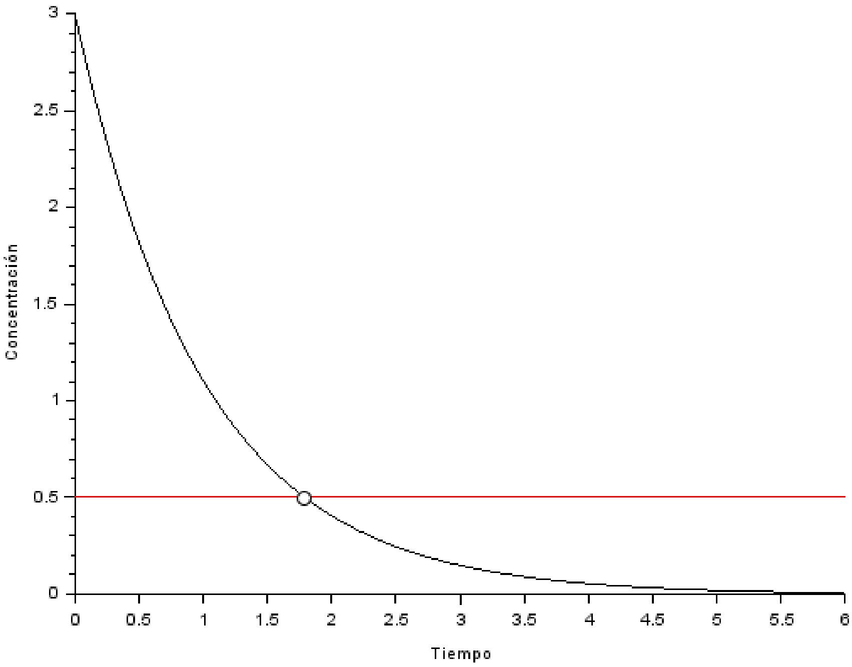

Fig. 6. La curva debe ser desplazada hasta hacer coincidir el eje de abscisas con el valor de ordenada deseado.

```
// Función de la cual se quieren conocen las raíces
function [y]=func(x)
    A=-1
    C=3
    y=C*exp(A*x)-0.5
endfunction
```

Para determinar el valor de la misma, es sencillo establecer el intervalo de la raíz en

I=[1.5, 2]

Si se desea una tolerancia de $10^{-5}$, la instrucción de cálculo será la siguiente

```
-->[r1,er]=regulafalsi(1.5,2,1d-5)
 er  =

    0.
 r1  =

    1.7917706
```

Es importante destacar que la tolerancia se mide en ordenadas, mientras que el resultado corresponde a un valor de abscisas.

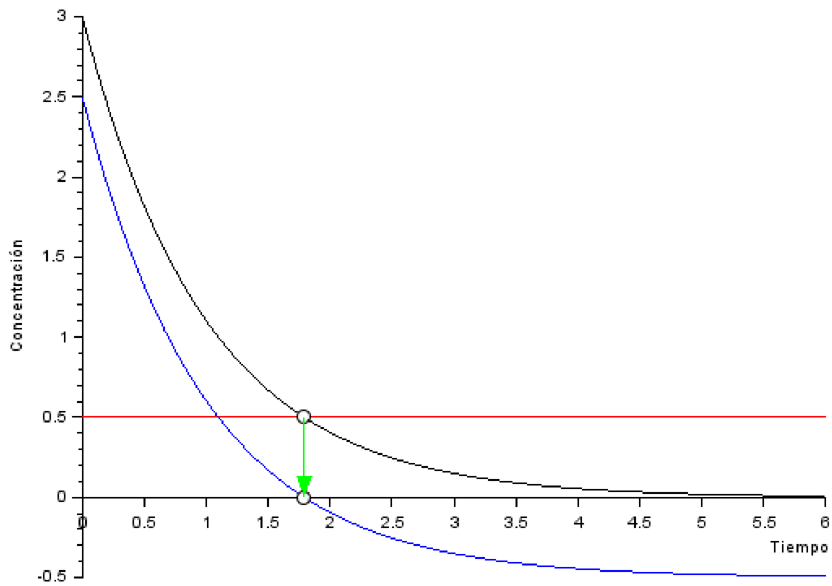


Fig. 7. Curva original desplazada para convertir el punto de interés en una raíz.

Por este motivo se debe analizar la precisión del resultado obtenido. Una manera simple es resolver el problema nuevamente con una tolerancia menor, y comparar las dos raíces para determinar la cantidad de cifras significativas exactas.

Para el caso del ejemplo, se puede utilizar una tolerancia con un orden de magnitud menor, tal cual se observa a continuación:

```
-->[r2,er]=regulafalsi(1.5,2,1d-6)
 er  =

   0.

 r2  =

   1.791761
```

Como se observa en la siguiente tabla, el primer resultado obtenido tiene 5 cifras significativas de precisión.

| Tolerancia | Resultado | Precisión |
|---|---|---|
| $10^{-5}$ | 1.7917706 | 5 cifras significativas |
| $10^{-6}$ | 1.791761 | |

Otra forma de obtener esta información relativa a la precisión, es calculando la diferencia de ambos resultados en valor absoluto, mediante el siguiente cálculo:

```
-->abs(r1-r2)
```

```
ans  =

   0.0000096
```

Es importante destacar que el valor calculado no es el error absoluto, ya que para obtener el mismo se debe conocer el resultado exacto o valor verdadero.

Se debe tener en cuenta que la precisión mencionada es relativa a la resolución numérica, no a la precisión en el mundo real. Es responsabilidad excluyente del ingeniero validar el resultado, para garantizar la correspondencia de los modelos matemáticos utilizados con la realidad. Aquí se observa la importancia de las pruebas de laboratorio y de planta piloto, para realizar la mencionada validación.

**INTERSECCIÓN DE CURVAS**

Otro caso interesante en ingeniería es el de determinar los puntos de intersección de dos curvas. Analíticamente se arman sistemas de ecuaciones, que al ser resueltos permiten conocer los puntos correspondientes. Sin embargo, no siempre dichos sistemas tienen solución analítica, por lo que deben ser abordados mediante métodos numéricos.

Sean las curvas dadas por las siguientes ecuaciones:

$$f_1(x) = 2\,x^3 - 8\,x^2 + 3\,x - 4 \qquad\qquad f_2(x) = e^{x-2} - 10$$

En la figura 8 están representadas las mismas, indicándose los puntos de intersección con sendas marcas rojas.

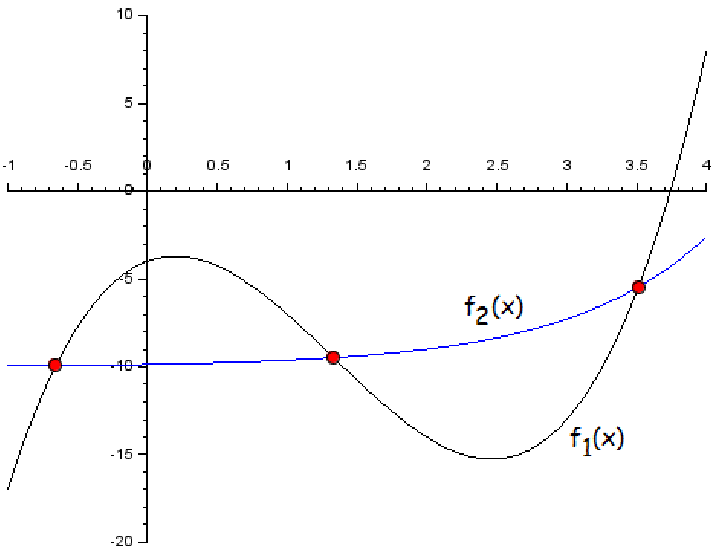

Fig. 8. En rojo están marcados los puntos de intersección de dos curvas.

Analíticamente no es posible resolver el proble–ma, por lo que la única alternativa es utilizar un método numérico, pero para poder hacer esto se deben convertir los puntos mencionados en raíces. Esto se hace fácilmente definiendo una nueva función que sea la diferencia de las funciones dadas; es decir, se define:

$$y(x) = f_1(x) - f_2(x)$$

Cuál función es el minuendo y cuál el sustraen–do no es importante, ya que sólo se buscan los puntos de intersección, lugar donde la diferencia es nula.

De esta forma se escribe **func()** de la siguiente manera:

```
function [y]=func(x)
    y=func1(x)-func2(x)
endfunction

function [f1]=func1(x)
    f1=2*x^3-8*x^2+3*x-4
endfunction

function [f2]=func2(x)
    f2=exp(x-2)-10
endfunction
```

Utilizando el gráfico de la figura 8, se determinan los tres intervalos y se calculan las raíces, de la misma manera que se hizo anteriormente. Para ejemplificar se resolverá para la primer intersección, se deja al lector las dos restantes.

La intersección de la izquierda se encuentra en el intervalo [-1; 0], se recuerda al lector que este intervalo se determina de manera grosera, no es necesario perder tiempo en buscar precisión, ya que ese trabajo corresponde al método numérico. Se utilizará una tolerancia de $10^{-5}$ para iniciar la búsqueda de la raíz, según se observa en las siguientes instrucciones.

```
--> [r1,er]=regulafalsi(-1,0,1d-5)
 er  =
   0.

r1  =
  -0.6531502
```

Para verificar la precisión obtenida se puede realizar el siguiente cálculo:

```
--> abs(func1(r1)-func2(r1))
 ans  =

   0.0000036
```

Pero este último valor corresponde a la precisión en ordenadas, para verificar la precisión en abscisas se realizará nuevamente el cálculo con una tolerancia de $10^{-6}$:

```
--> [r2,er]=regulafalsi(-1,0,1d-6)
 er  =

   0.

 r2  =

  -0.6531504

--> abs(func1(r2)-func2(r2))
 ans  =

   0.0000008

--> abs(r1-r2)
 ans  =

   0.0000002
```

Con estos dos últimos valores se puede observar que se tiene una buena precisión tanto en ordenadas como en abscisas. La precisión en los cálculos la especifica el ingeniero, quien es el que determina, de acuerdo sus necesidades, cuántas cifras significativas exactas se requieren en los resultados.