Bisection method

The *bisection method* (also known as *binary chopping* or *half-interval method*) is one of the simplest and most reliable of iterative methods for the solution of nonlinear equations. This method based on the repeated application of the *intermediate value theorem*.

- **Intermediate Value Theorem**: If $f(x)$ is a continuous function is some interval [a, b] and $f(a)$ and $f(b)$ are of opposite signs, then the equation $f(x) = 0$ has at least one real root or an odd number of real roots in (a, b) if $f(a)\, f(b) < 0$.

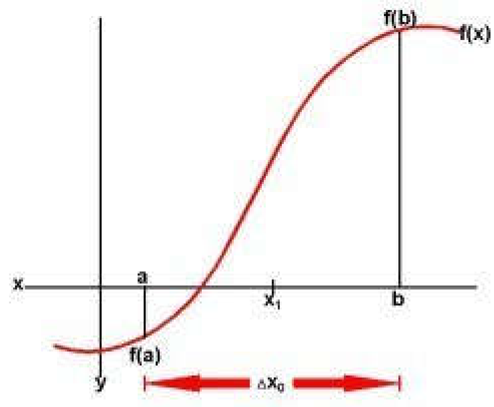

Figure 1: Illustration of Bisection Methods

This theorem is evident from the above Figure, that if $f(a)$ and $f(b)$ have opposite signs the graph must cross the x-axis at least once between x = a and x = b.

- Suppose $f(x)$ has a real root in the interval [a, b]. The midpoint between a and b is $x_0 = (a+b)/2$.

  Now, there exist the following three conditions:

  1.if $f(x_0) = 0$, then x_0 is the root of the equation $f(x) = 0$.

  2.if $f(x_0) * f(a) < 0$, then the root lies between x_0 and a.

  3.if $f(x_0) * f(b) < 0$, then the root lies between x_0 and b.

  It states that by testing the sign of the function at midpoint, we can conclude which part of the interval contains the root. We can further separation this subinterval into equal parts to find another subinterval containing the root. This procedure can be rehashed until the interim containing the root is as little as we want.

Algorithm: Bisection method

1. Decide initial values for lower limit a and upper limit b and stopping criterion e_s.
2. Compute $f_1 = f$ (a) and $f2 = f$ (b).
3. If $f_1 * f_2 > 0$, a and b do not bracket any root and go to step 1.
4. Compute $x_0 = (a + b) / 2$ and compute $f_0 = f(x_0)$.
5. If $f_0 = 0$ then x_0 is the root of the equation, print the root
6. If $f_1 * f_0 < 0$ then set $b = x_0$ else set $a = x_0$.
7. If $|(b – a)/b| < e_s$ then root = (a + b) / 2, print the root and go to step 8
   Else go to step 4
8. Stop.

Example 1: Find real root of the equation $f(x) = x^2 – 4x – 10 = 0$ using Bisection Method.
Solution:

$$|x_{\max}| = \sqrt{\{(\frac{a_{n-1}}{a_n})^2 - 2(\frac{a_{n-2}}{a_n})\}} = \sqrt{\{(\frac{-4}{1})^2 - 2(\frac{-10}{1})\}} = \pm 6$$

Table 1:

X	– 6	– 5	– 4	– 3	– 2	– 1	0	1	2	3	4	5	6
***f*(x)**	50	35	22	11	2	– 4	– 10	– 13	– 14	– 13	– 10	– 5	2

From Table 1 let lower limit a = – 2 and upper limit b = – 1

Iteration 1:
$f(-2) = 2$ and $f(– 1) = – 4$
so, $f(a) f(b) <0$
root lies between – 2 and – 1
$x_0 = a+b/2 =– 2 + (– 1)/2 = –1.5$
$f(x_0) = f(–1.5) = –1.75$

Iteration 2:
$f(-2) = 2$ and $f(– 1.5) = – 4$
so, $f(a) f(b) <0$
root lies between – 2 and – 1.5
$x_0 = a+b/2 =– 2 + (– 1.5)/2 = –1.75$
$f(x_0) = f(–1.75) = 0.0625$

Iteration 3:
$f(-1.5) = 1.75$ and $f(– 1.75) = – 0.0625$
so, $f(a) f(b) <0$
root lies between – 1.75 and – 1.5
$x_0 = a+b/2 =– 1.75 + (– 1.5)/2 = –1.625$
$f(x_0) = f(–1.75) = – 0.859$

Iteration 4:
$f(–1.75) = 0.0625$ and $f(– 1.625) = – 0.859$
so, $f(a) f(b) <0$

root lies between – 1.75 and – 1.625
$x_0 = a+b/2 = -1.75 + (-1.625)/2 = -1.6875$
$f(x_0) = f(-1.6875) = -0.40$

Iteration	x_1	x_2	x_0	$f(x_1)$	$f(x_2)$	$f(x_0)$
1	– 2	– 1	–1.5	2	–4	c
2	– 2	–1.5	–1.75	2	–1.75	0.0625
3	–1.75	–1.5	–1.625	0.0625	–1.75	–0.859
4	–1.75	–1.625	–1.6875	0.0625	–0.859	–0.40

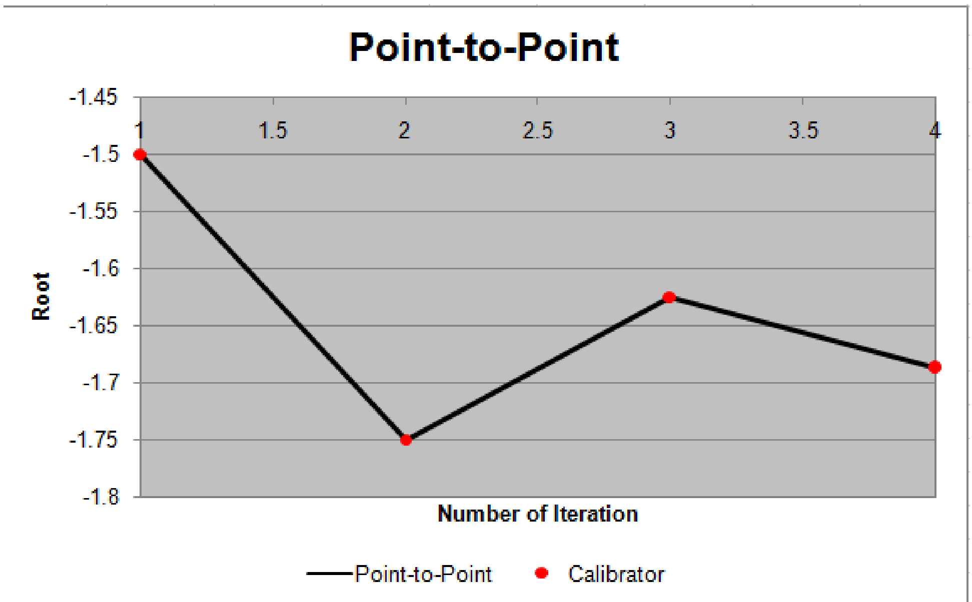

Figure 2: Illustration of Example 1

- The bisection method is *linearly convergent*. Since the convergence is slow to achieve a high degree of accuracy, a large number of iterations may be needed. However, the bisection method is guaranteed to converge.
- **A convenient stopping criterion is to compute the percentage error** ε_r **defined by**
  where $\varepsilon r = \left|\frac{x'_r - x_r}{x'_r}\right| * 100$
  x'_r is the new value of x_r. The computations can be terminated when ε_r becomes less than a prescribed tolerance, say ε_p. In addition, the maximum number of iterations may also be specified in advance.

- Flow Chart of bisection Methods:

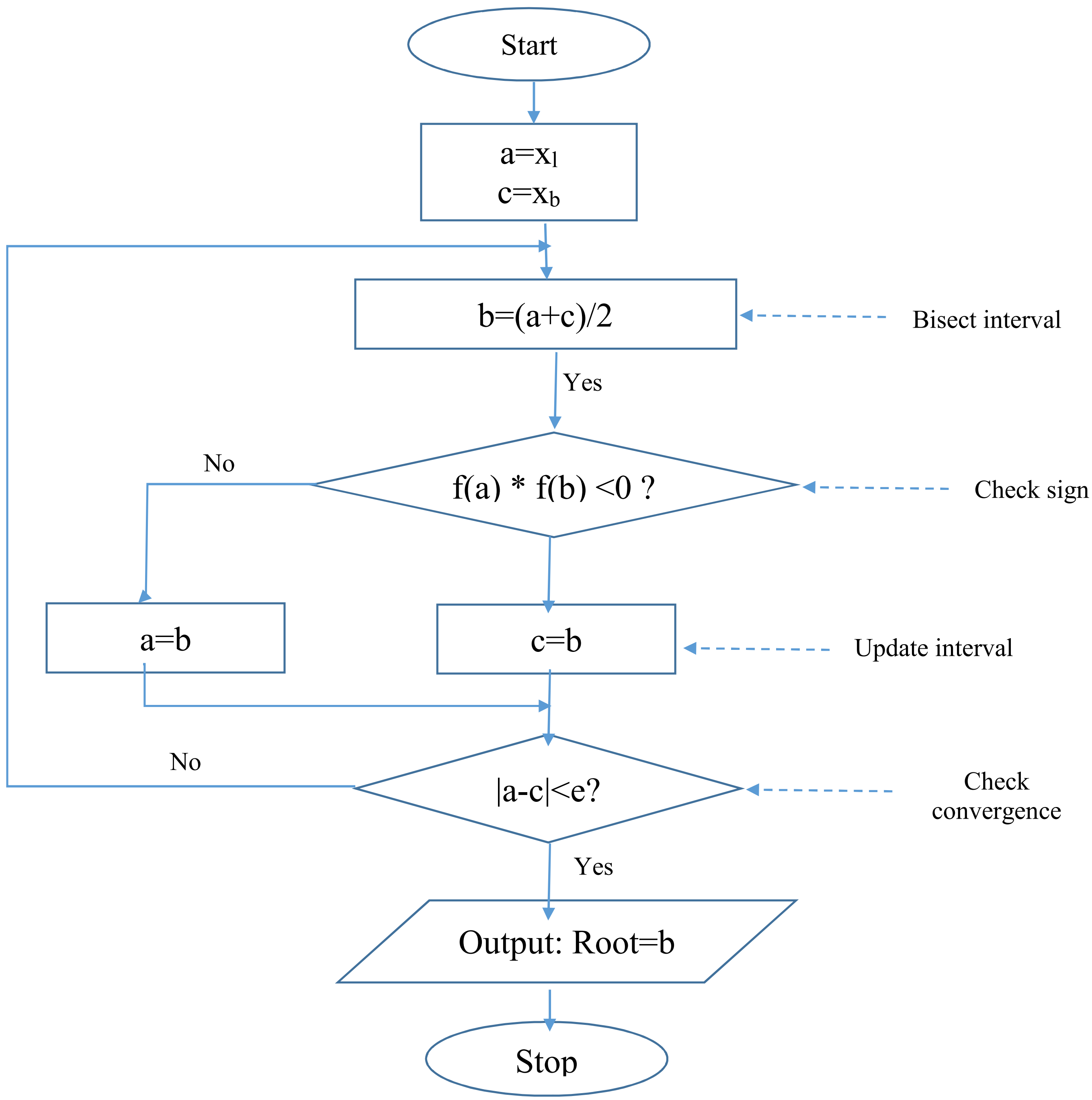

Figure 4: Flowchart of the Bisection Method

Convergence of Bisection Method:
In Bisection Method, we choose a mid-point x_0 in the interval between x_p and x_n. Depending upon the sign of $f(x_0)$, x_p or x_n is set equal to x_0 such that the root lies in the interval. In other case the interval containing the root is reduced by a function of 2. The same process is repeated for the new interval. If the procedure is repeated 'n' times, then the interval containing the root is reduced to the size:

$$\frac{x_p - x_n}{2^n} = \frac{\Delta x}{2^n}$$

After iteration the root must lies within of our estimate, this mean the error bound at n_{th} iteration is:

$$E_n = \frac{\Delta x}{2^n}$$

Similarly,

$$E_{n+1} = \frac{\Delta x}{2^{n+1}} = \frac{E_n}{2^n}$$

Thus, the error decreases linearly with each step by a factor of 0.5.This method is therefore linearly convergent.

Merits of bisection method

A. A root is always produced by the iteration using the bisection method, as the method brackets the root between two values.
B. The length of the interval is halved as iterations are performed. So in case of the equation solution, one can guarantee the convergence.
C. Programming the Bisection Method on a machine is easy.

Demerits of bisection method

A. The bisection process convergence is sluggish because it is based on halving the interval.
B. In case of discontinuity, the bisection approach can not be extended over an interval.
C. The method of bisection can not be extended at intervals where the function still takes the same sign values.
D. Complex roots are not defined by the system.
E. If one of the initial assumptions a_0 or b_0 is closer to the exact solution, further iterations will be required to reach the root.

False Position Method

The interval between x_1 and x_2 is divided into two equal halves in the bisection process, regardless of the root position. As shown in Figure 4, the root may be closer to one end than the other. The root's closer to x_1 in the figure. When we connect the x_1 and x_2 points in a straight line, the intersection point of this line with the x axis (x_0) provides an improved root and is called the *false position* of the root. This point then replaces one of the initial guesses that has a function value of the same sign as $f(x_0)$. The process is repeated with the new values of x_1 and x_2. Since this method uses the false position of the root repeatedly, it is called the *false position method* (or *regula falsi* in Latin). It is also called the *linear interpolation* method.

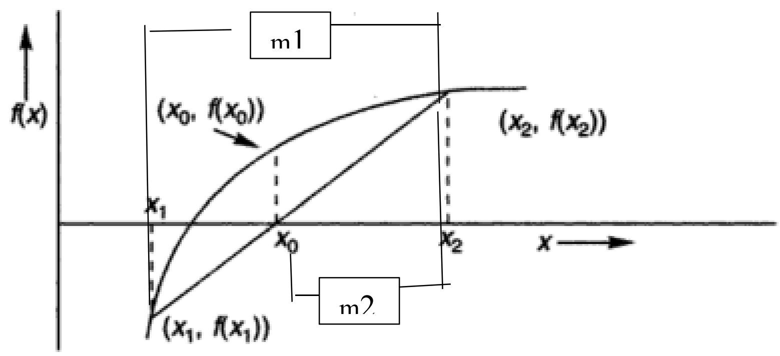

Figure 4: Illustration of Bisection Methods [1]

Derivation of False Position Formula

From figure 4: Slop of straight line m = $\frac{\Delta y}{\Delta x}$

Slop at point $(x_1, f(x_1))$ m1 = Slop at point $(x_2, f(x_2))$ m2

So,

The points $(x_1, f(x_1))$ and $(x_2, f(x_2))$ is given by

$$(f(x_2) - f(x_1)) / (x_2 - x_1) = (y - f(x_1)) / (x - x_1)$$

Since the line intersects the x-axis at x_0, when $x = x_0$, $y = 0$, we have,

$$(f(x_2) - f(x_1)) / (x_2 - x_1) = (-f(x_1)) / (x_0 - x_1)$$

or $$x_0 - x_1 = -f(x_1)(x_2 - x_1) / (f(x_2) - f(x_1))$$

Then we have,

$$x_0 = x_1 - (f(x_1)(x_2 - x_1)) / (f(x_2) - f(x_1))$$

This equation is known as the *false position formula*.

False Position Algorithm

1. Decide initial values for x_1 and x_2 and stopping criterion E.
2. Compute $x_0 = x_1 - (f(x_1)(x_2 - x_1)) / (f(x_2) - f(x_1))$
3. If $f(x_0) * f(x_1) < 0$ set $x_2 = x_0$ otherwise set $x_1 = x_0$
4. If the absolute difference of two successive x_0 is less then E, then root = x_0 and stop. Else go to step 2.

- False position method converges *linearly*.

Example 2: Use the false position method to find a root of the function $f(x) = x^2 - x - 2 = 0$ in the range $1 < x < 3$.

Solution: Given $x_1 = 1$ and $x_2 = 3$

$f(x_1) = f(1) = -2 \qquad f(x_2) = f(3) = 4$

$x_0 = x_1 - (f(x_1)\,(x_2 - x_1)) / (f(x_2) - f(x_1))$

$= 1 + 2 * (3 - 1) / (4 + 2)$

$= 1.6667$

Iteration 2: $f(x_0) * f(x_1) = f(1.6667) * f(1) = 1.7778$

Therefore, the root lies in the interval between x_0 and x_2. Then,

$x_1 = x_0 = 1.6667$

$f(x_1) = f(1.6667) = -0.8889$

$f(x_2) = f(3) = 4$

$x_0 = 1.6667 + 0.8889 * (3 - 1.6667) / (4 + 0.8889) = 1.909$

Iteration 3: $f(x_0) * f(x_1) = f(1.909) * f(1.6667) = 0.2345$

Therefore, the root lies in the interval between $x_0 = 1.909$ and $x_2 = 3$. Then,

$x_1 = x_0 = 1.909$

$x_0 = 1.909 - 0.2647 * (3 - 1.909) / (4 - 0.2647) = 1.986$

The estimated root after third iteration is 1.986. The interval contains a root $x = 2$. We can perform additional iterations to refine this estimate further.

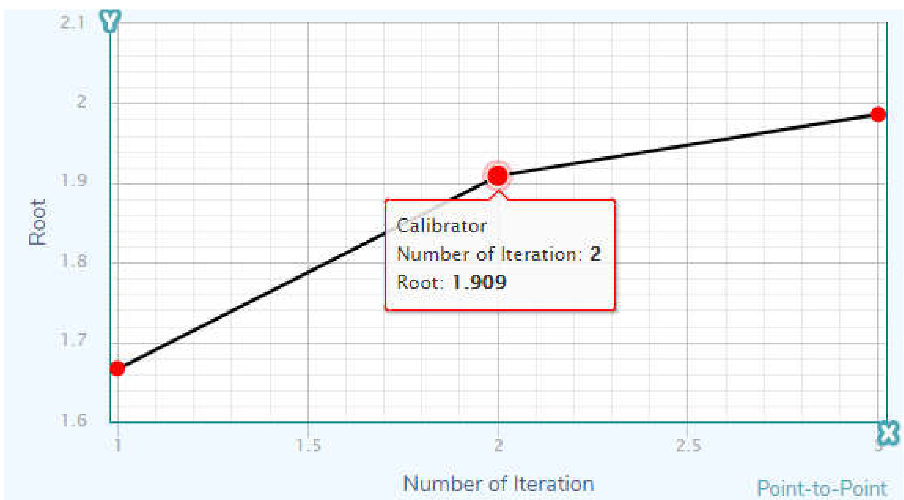

Figure 5: Illustration of Example 2

References:

1. **BalaGurushamy, E.** *Numerical Methods.* New Delhi : Tata McGraw-Hill, 2000.

2. **Steven C.Chapra, Raymon P. Cannale.** *Numerical Methods for Engineers.* New Delhi : Tata McGRAW-HILL, 2003. ISMN 0-07-047437-0.