# Chapter2: Solving Nonlinear Equations

## 2.1 BACKGROUND

Equations need to be solved in all areas of science and engineering. An equation of one variable can be written in the form:

$$f(x) = 0 \qquad (2.1)$$

A solution to the equation (also called a root of the equation) is a numerical value of x that satisfies the equation. Graphically, as shown in Fig. 2-1, the solution is the point where the function $f(x)$ crosses or touches the x-axis. An equation might have no solution or can have one or several (possibly many) roots. When the equation is simple, the value of $x$ can be determined analytically. This is the case when x can be written explicitly by applying mathematical operations, or when a known formula (such as the formula for solving a quadratic equation) can be used to determine the exact value of x. In many situations, however, it is impossible to determine the root of an equation analytically.

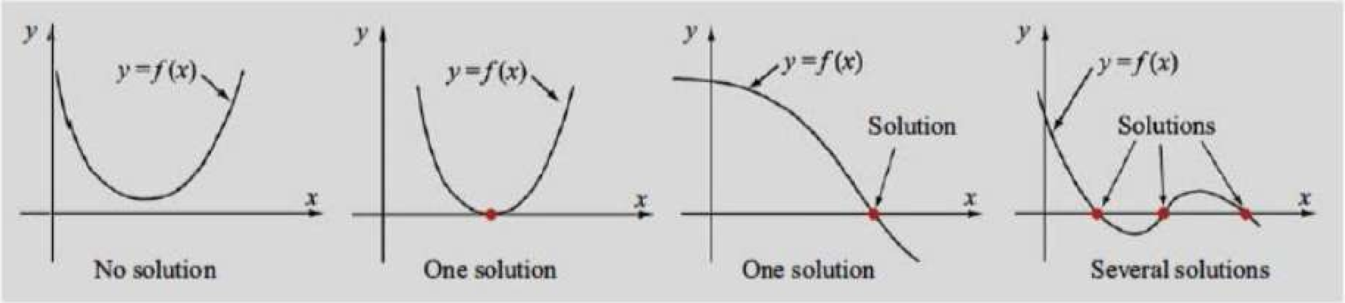


**Figure 2-1: Illustration of equations with no, one, or several solutions.**

### Overview of approaches in solving equations numerically

The process of solving an equation numerically is different from the procedure used to find an analytical solution. An analytical solution is obtained by deriving an expression that has an exact numerical value. A numerical solution is obtained in a process that starts by finding an approximate solution and is followed by a numerical procedure in which a better (more accurate) solution is determined.

An initial numerical solution of an equation $f(x) = 0$ can be estimated by plotting $f(x)$ versus x and looking for the point where the graph crosses the $x$-axis.

It is also possible to write and execute a computer program that looks for a domain that contains a solution. Such a program looks for a solution by evaluating $f(x)$ at different values of x. It starts at one value of x and then changes the value of $x$ in small increments. A change in the sign of $f(x)$ indicates that there is a root within the last increment. In most cases, when the equation that is solved is related to an application in science or engineering, the range of $x$ that includes the solution can be estimated and used in the initial plot of $f(x)$, or for a numerical search of a small domain that contains a solution. When an equation has more than one root, a numerical solution is obtained one root at a time.

The methods used for solving equations numerically can be divided into two groups: bracketing methods and open methods.

In bracketing methods, illustrated in Fig. 2-2, an interval that includes the solution is identified. By definition, the endpoints of the interval are the upper bound and lower bound of the solution. Then, by using a numerical scheme, the size of the interval is successively reduced until the distance between the endpoints is less than the desired accuracy of the solution. In open methods, illustrated in Fig. 2-3, an initial estimate (one point) for the solution is assumed. The value of this initial guess for the solution should be close to the actual solution. Then, by using a numerical scheme, better (more accurate) values for the solution are calculated. Bracketing methods always converge to the solution. Open methods are usually more efficient but sometimes might not yield the solution.

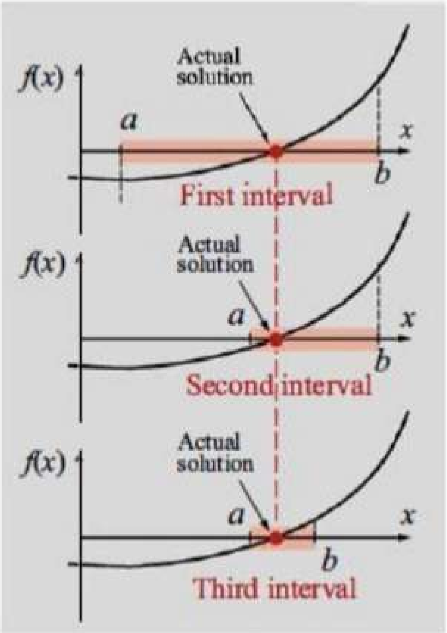


Figure 2-2: Illustration of a bracketing method.

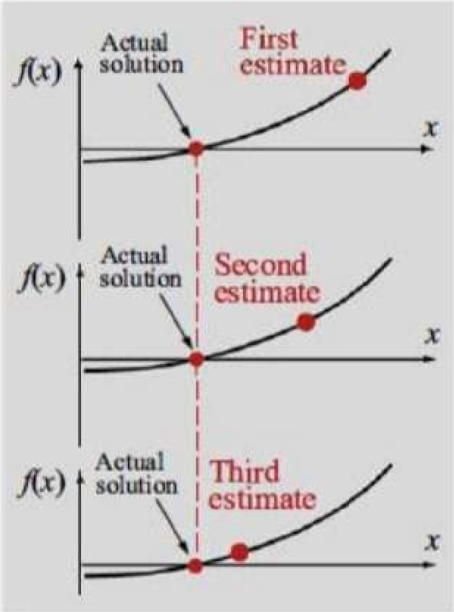


Figure 2-3: Illustration of an open method.

## 2.2 ESTIMATION OF ERRORS IN NUMERICAL SOLUTIONS

Since numerical solutions are not exact, some criterion has to be applied in order to determine whether an estimated solution is accurate enough. Several measures can be used to estimate the accuracy of an approximate solution. The decision as to which measure to use depends on the application and has to be made by the person solving the equation. Let $x_{rs}$ be the true (exact) solution such that $f(x_{rs}) = 0$, and let $x_{Ns}$ be a numerically approximated solution such that $f(x_{Ns})$ = E (where E is a small number). Four measures that can be considered for estimating the error are:

### 2.2.1 True error

The true error is the difference between the true solution, $X_{rs}$ and a numerical solution, $X_{Ns}$:

$$TrueError = X_{rs}\text{-}X_{Ns} \qquad (2.2)$$

Unfortunately, however, the true error cannot be calculated because the true solution is generally not known.

### 2.2.2 Tolerance in *f(x)*:

Instead of considering the error in the solution, it is possible to consider the deviation of $f(x_{Ns})$ from zero (the value of $f(x)$ at $x_{rs}$ is obviously zero). The tolerance in $f(x)$ is defined as the absolute value of the difference between $f(x_{rs})$ and $f(x_{Ns})$:

$$ToleranceInf = |f(x_{rs}) - f \text{[illegible]} - \varepsilon| = |\varepsilon| \qquad (2.3)$$

The tolerance in $f(x)$ then is the absolute value of [illegible] $x_{Ns}$.

### 2.2.3 Tolerance in the solution:

Tolerance is the maximum amount by which the true solution can deviate from an approximate numerical solution. A tolerance is useful for estimating the error when bracketing methods are used for determining the numerical solution. In this case, if it is known that the solution is within the domain [a, b] ,

### 2.2.1 True error

The true error is the difference between the true solution, $X_{rs}$ and a numerical solution, $X_{Ns}$:

$$TrueError = X_{rs}\text{-}X_{Ns} \qquad (2.2)$$

nfortunately, however, the true error cannot be calculated because the true solution is generally not known.

### .2.2 Tolerance in *f(x)*:

Instead of considering the error in the solution, it is possible to consider the deviation of $f(x_{Ns})$ from zero (the value of $f(x)$ at $x_{rs}$ is obviously zero). The tolerance in $f(x)$ is defined as the absolute value of the difference between $f(x_{rs})$ and $f(x_{Ns})$:

$$Tolerancelnf = |f(x_{rs}) - f(x_{Ns})| = |0 - \varepsilon| = |\varepsilon| \qquad (2.3)$$

The tolerance in $f(x)$ then is the absolute value of the function at $x_{Ns}$.

### 2.2.3 Tolerance in the solution:

Tolerance is the maximum amount by which the true solution can deviate from an approximate numerical solution. A tolerance is useful for estimating the error when bracketing methods are used for determining the numerical solution. In this case, if it is known that the solution is within the domain [a, b] , then the numerical solution can be taken as the midpoint between a and b:

$$x_{Ns} = \frac{a+b}{2} \qquad (2.4)$$

plus or minus a tolerance that is equal to half the distance between a and b:

$$\text{Tolerance}=\frac{b-a}{2} \qquad (2.5)$$

### 2.2.4 Relative error:

If $x_{Ns}$ is an estimated numerical solution, then the True Relative Error is given by:

$$\text{TrueRelativeError} = \left|\frac{x_{rs}-x_{Ns}}{x_{Ns}}\right| \qquad (2.6)$$

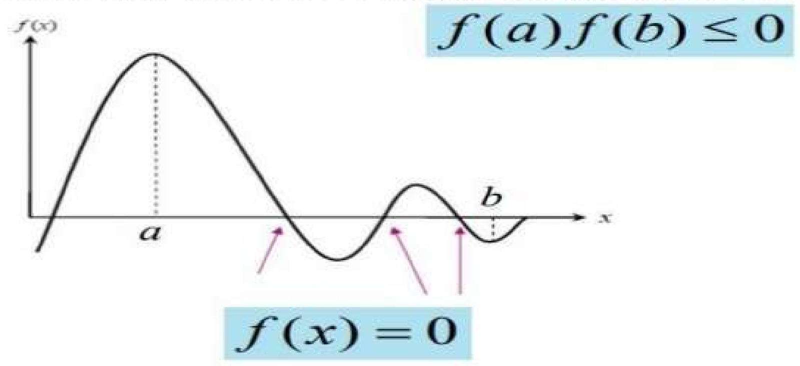
This True Relative Error cannot be calculated since the true solution $x_{rs}$ is not known. Instead, it is possible to calculate an Estimated Relative Error when two numerical estimates for the solution are known. This is the case when numerical solutions are calculated iteratively, wherein each new iteration a more accurate solution is calculated. If $x_{Ns}^{(n)}$is the estimated numerical solution in the last iteration and $x_{Ns}^{(n-1)}$ is the estimated numerical solution in the preceding iteration, then an Estimated Relative Error can be defined by:

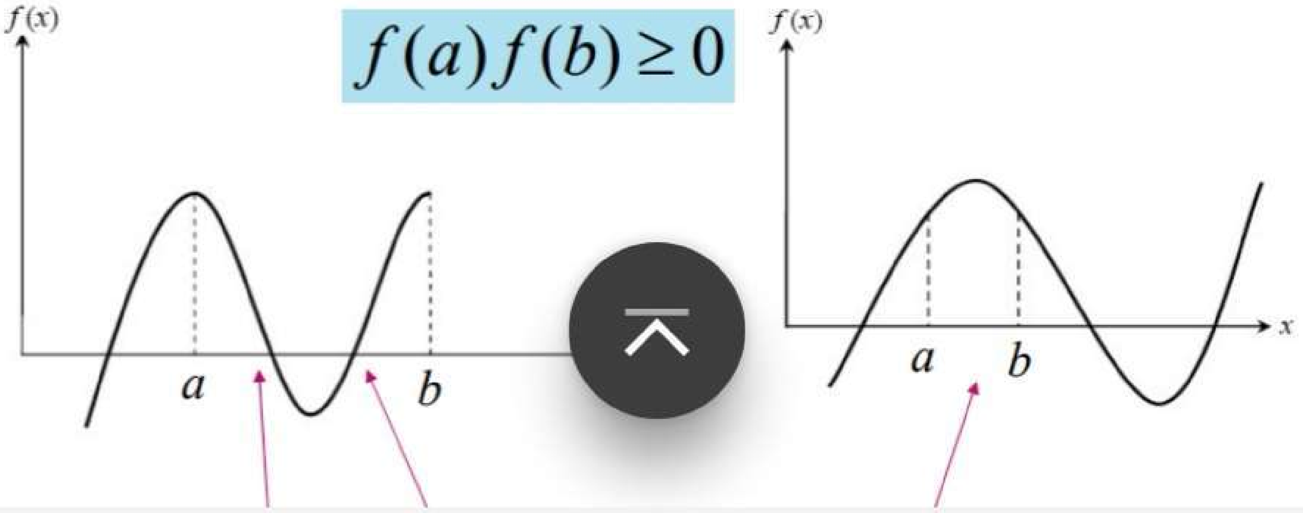
$$\text{Estimated Relative Error}=\left|\frac{x_{Ns}^{(n)}-x_{Ns}^{(n-1)}}{x_{Ns}^{(n-1)}}\right| \qquad (2.7)$$

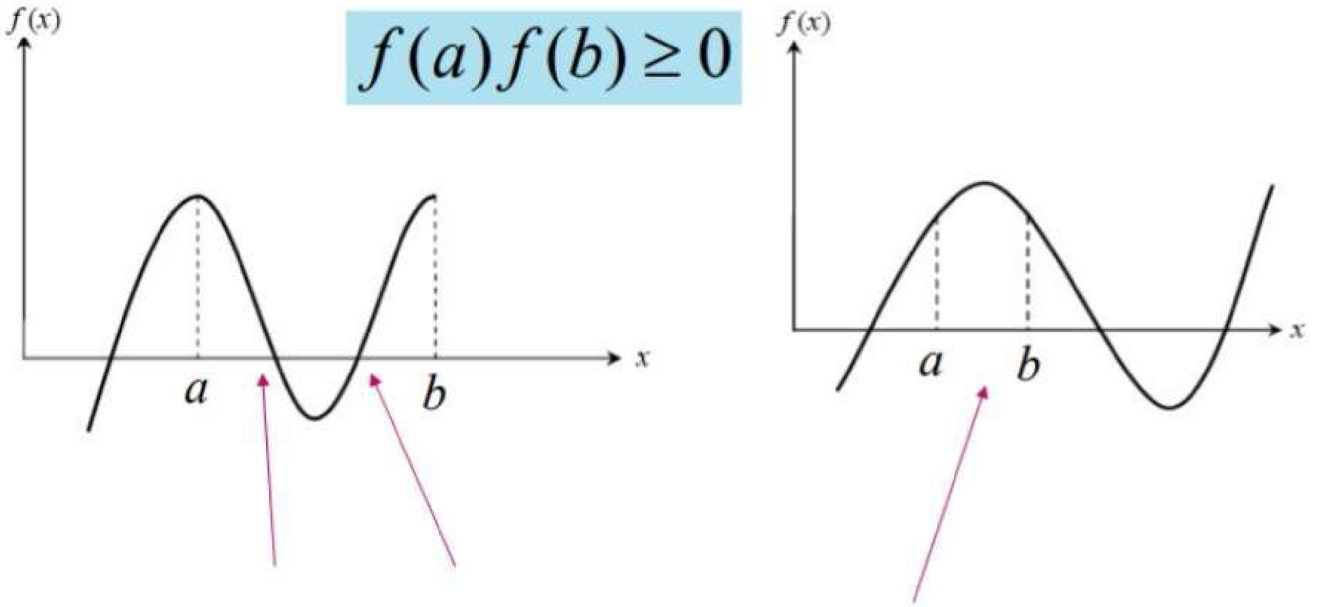
When the estimated numerical solutions are close to the true solution it is anticipated that the difference $x_{Ns}^{(n)} - x_{Ns}^{(n-1)}$ is small compared to the value of $x_{Ns}^{(n)}$, and the Estimated Relative Error is approximately the same as the True Relative Error.

## 2.3 Root-finding algorithms

***Theorem:*** If the function $f(x)$ is defined and continuous in the range [a,b] and function changes sign at the ends of the interval that is $f(a)f(b) < 0$ then there is at least one single root in the range [a,b].







If the function does not change the sign between two points, there may not be or there may exist roots for this equation between the two points.

***Root-finding strategy***

- Plot the function (the plot provides an initial guess, and indication of potential problems).
- Isolate single roots in separate intervals (bracketing).
- Select an initial guess.
- Iteratively refine the initial guess with a root-finding algorithm, i.e. generate the sequence :

$$\{x_i\}_{i=0}^{n} : \lim_{n\to\infty}(x_n - \alpha) = 0$$

***<u>EXAMPLE 2.1</u>***

Find the largest root of $f(x) = x^6 - x - 1 = 0$.

| $x$ | -2 | -1 | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|---|---|
| $f(x)$ | 65 | 1 | -1 | -1 | 61 | 725 | 4091 |

It is obvious that the largest root of this equation is in the interval [1.2].

## 2.4 BISECTION METHOD

The bisection method is a bracketing method for finding a numerical solution of an equation of the form $f(x) = 0$ when it is known that within a given interval [a, b], $f(x)$ is continuous and the equation has a solution. When this is the case, $f(x)$ will have opposite signs at the endpoints of the interval. As shown in Fig. 2-4, if $f(x)$ is continuous and has a solution between the points $x$ = a and $x$ = b , then either $f(a) > 0$ and $f(b) < 0$ or $f(a) < 0$ and $f(b) > 0$. In other words, if there is a solution between $x$=a and $x$ = b, then $f(a)f(b) < 0$.

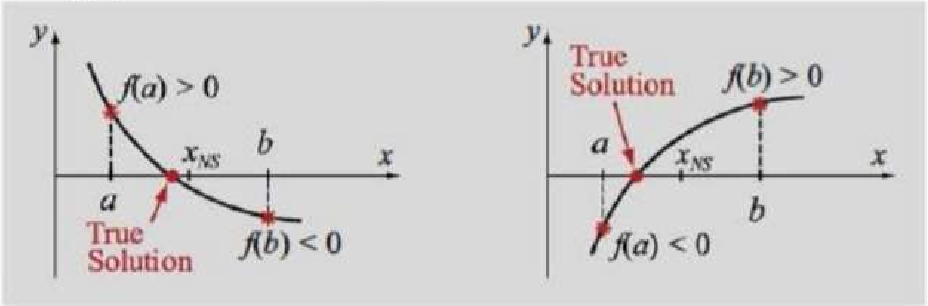


**Figure 2-4: Solution of $f(x) = 0$ between $x$ =a and $x$ = b.**

## *Algorithm for the bisection method*

1. Choose the first interval by finding points a and b such that a solution exists between them. This means that *f(a)* and *f(b)* have different signs such that $f(a)f(b) < 0$. The points can be determined by examining the plot of f(x) versus x.

2. Calculate the first estimate of the numerical solution $x_{Ns1}$ by:

$$x_{Ns1} = \frac{a+b}{2}$$

Determine whether the true solution is between a and $x_{NS1}$ or between $x_{Ns1}$ and b. This is done by checking e sign of the product $f(a) \cdot f(x_{Ns1})$ :

$f(a) \cdot f(x_{Ns1}) < 0$, the true solution is between a and $x_{Ns1}$.

If $f(a) \cdot f(x_{Ns1}) > 0$, the true solution is between $x_{Ns1}$ and b.

4. Select the subinterval that contains the true solution (a to $x_{Ns1}$, or $x_{Ns1}$ to b) as the new interval [a, b], and go back to step 2.

Steps 2 through 4 are repeated until a specified tolerance or error bound is attained.

## When should the bisection process be stopped?

Ideally, the bisection process should be stopped when the true solution is obtained. This means that the value of $x_{Ns}$ is such that $f(x_{Ns}) = 0$. In reality, as discussed in Section 2.1, this true solution generally cannot be found computationally. In practice, therefore, the process is stopped when the estimated error, according to one of the measures listed in Section 2.2, is smaller than some predetermined value. The choice of termination criteria may depend on the problem that is actually solved.

## Additional notes on the bisection method

- The method always converges to an answer, provided a root was trapped in the interval [a, b] to begin with.
- The method may fail when the function is tangent to the axis and does not cross the x-axis at f(x) = 0.
- The method converges slowly relative to other methods.

***EXAMPLE 2.2***

Find the largest root of $f(x) = x^6 - x - 1 = 0$ accurate to within $\in = 0.001$.

***Solution*** With a graph, it is easy to check that $1 < \alpha < 2$. We choose a = 1, b =2; then $f(a) = -1, f(b) = 61$, and the requirement $f(a)f(b) < 0$ is satisfied. The results from Bisect are shown in the table. The entry n indicates the iteration number n.

| $n$ | $a$ | $b$ | $c$ | $b - c$ | $f(c)$ |
|---|---|---|---|---|---|
| 1 | 1.0000 | 2.0000 | 1.5000 | 0.5000 | 8.8906 |
| 2 | 1.0000 | 1.5000 | 1.2500 | 0.2500 | 1.5647 |
| 3 | 1.0000 | 1.2500 | 1.1250 | 0.1250 | −0.0977 |
| 4 | 1.1250 | 1.2500 | 1.1875 | 0.0625 | 0.6167 |
| 5 | 1.1250 | 1.1875 | 1.1562 | 0.0312 | 0.2333 |
| 6 | 1.1250 | 1.1562 | 1.1406 | 0.0156 | 0.0616 |
| 7 | 1.1250 | 1.1406 | 1.1328 | 0.0078 | −0.0196 |
| 8 | 1.1328 | 1.1406 | 1.1367 | 0.0039 | 0.0206 |
| 9 | 1.1328 | 1.1367 | 1.1348 | 0.0020 | 0.0004 |
| 10 | 1.1328 | 1.1348 | 1.1338 | 0.00098 | −0.0096 |

**Example 2.3** Show that $f(x) = x^3 + 4x^2 - 10 = 0$ has a root in [1, 2], and use the Bisection method to determine an approximation to the root that is accurate to at least within$10^{-4}$.

**Solution** Because $f(1) = -5$ and $f(2) = 14$, the Intermediate Value Theorem ensures that this continuous function has a root in [1, 2].

For the first iteration of the Bisection method we use the fact that at the midpoint of [1,2] we have $f(1.5) = 2.375 > 0$. This indicates that we should select the interval [1,1.5] for our second iteration. Then we find that $f(1.25) = -1.796875$ so our new interval becomes [1.25, 1.5], whose midpoint is 1.375. Continuing in this manner gives the values in the following table. After 13 iterations, $p_{13} = 1.365112305$ approximates the root $p$ with an error:

$|p - p_{13}| < |b_{14} - a_{14}| = |1.365234375 - 1.365112305| = 0.000122070.$

Since $|a_{14}| < |p|$, we have $|p - p_{13}|/|p| < |b_{14} - a_{14}|/|a_{14}| \leq 9.0 \times 10^{-5}$,

| $n$ | $a_n$ | $b_n$ | $p_n$ | $f(p_n)$ |
|---|---|---|---|---|
| 1 | 1.0 | 2.0 | 1.5 | 2.375 |
| 2 | 1.0 | 1.5 | 1.25 | −1.79687 |
| 3 | 1.25 | 1.5 | 1.375 | 0.16211 |
| 4 | 1.25 | 1.375 | 1.3125 | −0.84839 |
| 5 | 1.3125 | 1.375 | 1.34375 | −0.35098 |
| 6 | 1.34375 | 1.375 | 1.359375 | −0.09641 |
| 7 | 1.359375 | 1.375 | 1.3671875 | 0.03236 |
| 8 | 1.359375 | 1.3671875 | 1.36328125 | −0.03215 |
| 9 | 1.36328125 | 1.3671875 | 1.365234375 | 0.000072 |
| 10 | 1.36328125 | 1.365234375 | 1.364257813 | −0.01605 |
| 11 | 1.364257813 | 1.365234375 | 1.364746094 | −0.00799 |
| 12 | 1.364746094 | 1.365234375 | 1.364990235 | −0.00396 |
| 13 | 1.364990235 | 1.365234375 | 1.365112305 | −0.00194 |

so the approximation is correct to at least within $10^{-4}$. The correct value of $p$ to nine decimal places is $p = 1.365230013$. Note that $p_9$ is closer to $p$ than is the final approximation $p_{13}$. You might suspect this is true because $|f(p_9)| < |f(p_{13})|$, but we cannot be sure of this unless the true answer is known.

***Example 2.4*** Use the Bisection method to find a root of the equation $x^3 - 4x - 8.95 = 0$ accurate to three decimal places using the Bisection Method.

***Solution***

Here, $f(x) = x^3 - 4x - 8.95 = 0$

$f(2) = 2^3 - 4(2) - 8.95 = -8.95 < 0$

$f(3) = 3^3 - 4(3) - 8.95 = 6.05 > 0$

Hence, a root lies between 2 and 3.

Hence, we have $a = 2$ and $b = 3$. The results of the algorithm for Bisection method are shown in Table.

| n | a | b | $xs_1$ | b-$xs_1$ | f($xs_1$) |
|---|---|---|---|---|---|
| 0 | 2 | 3 | 2.5 | 0.5 | -3.3249999999999999 |
| 1 | 2.5 | 3 | 2.75 | 0.25 | 0.846875000000001 |
| 2 | 2.5 | 2.75 | 2.625 | 0.125 | -1.362109374999999 |
| 3 | 2.625 | 2.75 | 2.6875 | 0.0625 | -0.289111328124999 |
| 4 | | | 2.71875 | 0.03125 | 0.270916748046876 |
| 5 | | | 2.703125 | 0.015625 | -0.011077117919921 |
| 6 | | | 2.7109375 | 0.007812 | 0.129423427581788 |
| 7 | | | 2.7070312 | 0.003906 | 0.059049236774445 |
| 8 | | | 2.7050781 | 0.001953 | 0.023955102264882 |
| 9 | | | 2.7041016 | 0.000976 | 0.006431255675853 |
| 10 | | | 2.7036133 | 0.000488 | -0.002324864896945 |
| 11 | | | 2.7038574 | 0.000244 | 0.002052711902071 |
| 12 | | | 2.7037354 | 0.000122 | -0.000136197363826 |
| 13 | | | 2.7037964 | 0.000061 | 0.000958227051843 |

Hence the root is 2.703 accurate to three decimal places.