Chapter 3

Solution of Non-linear equation

3.1 Introduction

The primary objective of this chapter is to acquaint you with different methods for finding the root of a single nonlinear equation. The value of m that makes $f(m) = 0$ is, the root of the equation.

3.2 Graphical Methods

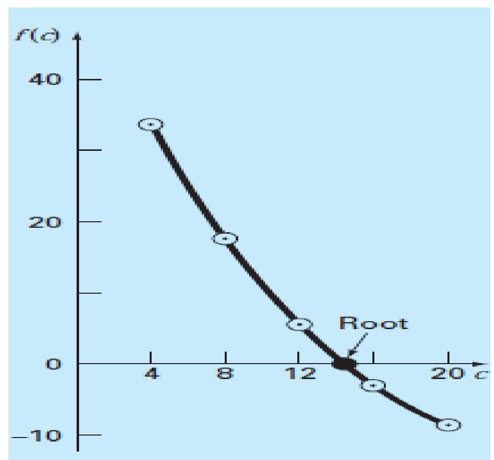
A simple method for obtaining an estimate of the root of the equation $f(x) = 0$ is to make a plot of the function and observe where it crosses the x axis. This point, which represents the x value for which $f(x) = 0$, provides a rough approximation of the root.

Example 1: Use the graphical approach to determine the drag coefficient c needed for a parachutist of mass $m = 68.1kg$ to have a velocity of $40m/s$ after free-falling for time $t = 10\ s$. Note: The acceleration due to gravity is $9.8\ m/s^2$.

Solution:

$$f(c) = \frac{667.38}{c}(1 - e^{-0.146843c}) - 40$$

c	4	8	12	16	20
$f(c)$	34.115	17.653	6.067	−2.269	−8.401



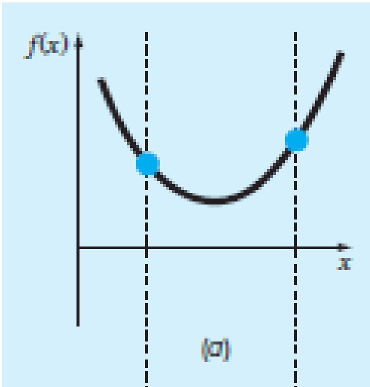
These points are plotted. The resulting curve crosses the c axis between 12 and 16.

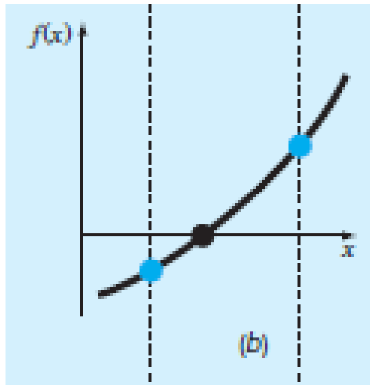
Visual inspection of the plot provides a rough estimate of the root of 14.75.

$$f(14.75) = \frac{667.38}{14.75}\left(1 - e^{-0.146843(14.75)}\right) - 40 = 0.059\ (close\ to\ zero)$$

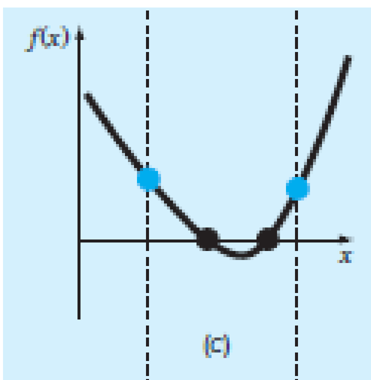
The Figures show a number of general ways that a root mayoccur in an interval prescribed by a lower bound x_l and an upper bound x_u. Parts (a) and (c) indicate that if both $f(x_l)$ and $f(x_u)$have the same sign, either there will be no roots or there will be an even number of roots within

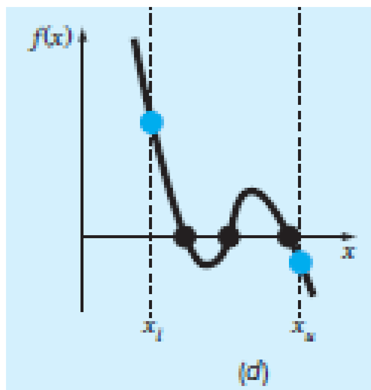
the interval.





Parts (b) and (d) indicate that if the function has different signs at the end points, there will be an odd number of roots in the interval.





The two major classes of methods available are distinguished by the type of initial guess. They are

- Bracketing methods: these are based on two initial guesses that “bracket” the root—that is, are on either side of the root.
- Open methods. These methods can involve one or more initial guesses, but there is no need for them to bracket the root.

For well-posed problems, the bracketing methods always work but converge slowly (i.e., they typically take more iterations to home in on the answer). In contrast, the open methods do not always work (i.e., they can diverge), but when they do they usually converge quicker.

3.3 Bracketing Methods

3.3.1 Bisection

The bisection method is alternatively called binary chopping, interval halving, or Bolzano's method, in which the interval is always divided in half. If a function changes sign over an interval, the function value at the midpoint is evaluated. The location of the root is then determined as lying at the midpoint of the subinterval within which the sign change occurs. The subinterval then becomes the interval for the next iteration. The process is repeated until the root is known to the requred precision.

An approximate percent relative error ε_a can be calculated as

$$\varepsilon_a = \left| \frac{x_r^{new} - x_r^{old}}{x_r^{new}} \right| 100\%$$

where x_r^{new} is the root for the present iteration and x_r^{old} is the root from the previous iteration. The absolute value is used because we are usually concerned with the magnitude of ε_a rather than with its sign. When ε_a becomes less than a prespecified stopping criterion ε_s , the computation is terminated.

An alternative formulation for the approximate percent relative error

$$\varepsilon_a = \left| \frac{x_u - x_l}{x_u + x_l} \right| 100\%$$

This equation allows us to calculate an error estimate on the basis of our initial guesses.

If E is the desired error, the number of iterations, n, is

$$n = \frac{\log(\Delta x^0 / E_{a,d})}{\log 2} = \log_2 \left(\frac{\Delta x^0}{E_{a,d}} \right)$$

Example 2: Use bisection to solve the same previous problem until the approximate error falls below a stopping criterion of ε_s = 0.5%.

$$f(c) = \frac{667.38}{c}(1 - e^{-0.146843c}) - 40$$

Note that the true value of the root is 14.7802

The first step in bisection is to guess two values of the unknown that give values for $f(c)$ with different signs. The function changes sign between values of 12 and 16 then there is at least one real root between x_l and x_u. Therefore, the initial estimate of the root x_r lies at the midpoint of the interval

c	4	8	**12**	**16**	20
$f(c)$	34.115	17.653	**6.067**	**−2.269**	−8.401

$$x_r = \frac{12+16}{2} = 14$$

$$\Delta x^0 = 16 - 12 = 4, E_{a,d} = 0.0628$$

$$n = \frac{\log(\Delta x^0 / E_{a,d})}{\log 2} = \frac{\log(4/0.0628)}{\log 2} = 6$$

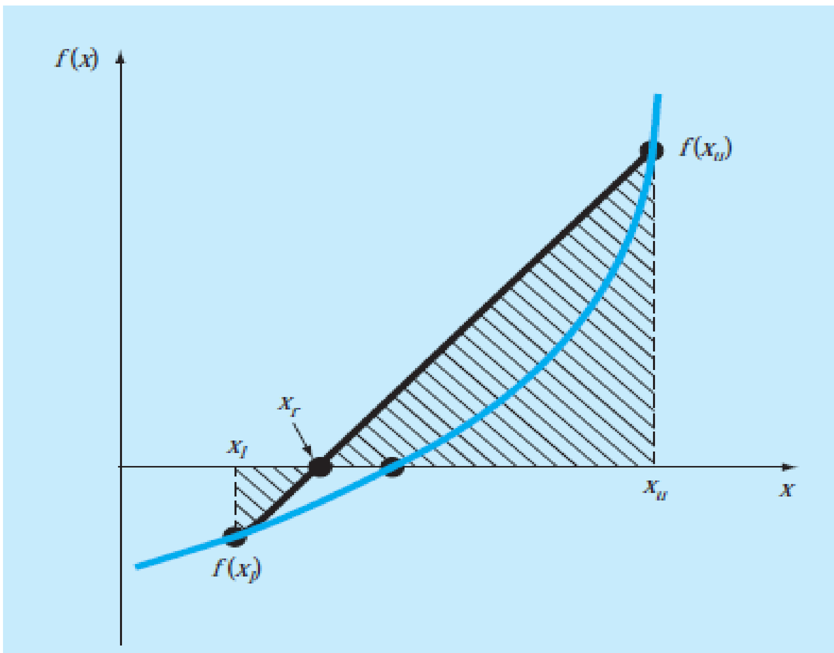
$$\varepsilon_a = \left|\frac{x_u - x_l}{x_u + x_l}\right| 100\% = \left|\frac{16-12}{16+12}\right| 100\%$$

$$\varepsilon_a = 14.29\%$$

Iteration	x_l	x_u	x_r	$f(x_r)$	ε_a (%)	ε_t (%)
1	12	16	14	$f(14) = 1.568$	14.29	5.279
2	14	16	15	$f(15) = -0.425$	6.667	1.487
3	14	15	14.5	$f(14.5) = 0.5523$	3.446	1.896
4	14.5	15	14.75	$f(14.75) = 0.0589$	1.695	0.204
5	14.75	15	14.875	$f(14.875) = -0.184$	0.840	0.641
6	14.75	14.875	14.8125	$f(14.8125) = -0.0628$	0.422	0.219

3.3.2 The False-Position Method

False position is an alternative based on a graphical insight to join $f(x_l)$ and $f(x_u)$ by a straight line. The intersection of this line with the x axis represents an improved estimate of the root. The fact that the replacement of the curve by a straight line gives a "false position" of the root is the origin of the name, method of false position. It is also called the linear interpolation method.



Using similar triangles, the intersection of the straight line with the x axis can be estimated as

$$\frac{f(x_l)}{x_r - x_l} = \frac{f(x_u)}{x_r - x_u} \rightarrow x_r = x_u \frac{f(x_u)(x_l - x_u)}{f(x_l) - f(x_u)}$$

This is the false-position formula. The value of x_r computed then replaces whichever of the two initial guesses, x_l or x_u, yields a function value with the same sign as $f(x_r)$. In this way, the values of x_l and x_u always bracket the true root. The process is repeated until the root is estimated adequately. The same stopping criterion is used to terminate the computation.

$$\varepsilon_a = \left| \frac{x_r^{new} - x_r^{old}}{x_r^{new}} \right| 100\%$$

Example 3: Use the false-position method to determine the root of the same equation investigated in previous Example.

$$f(c) = \frac{667.38}{c}(1 - e^{-0.146843c}) - 40$$

Solution:

Initiate the computation with guesses of $x_l = 12$ and $x_u = 16$.

First iteration:

$$x_l = 12, f(x_l) = 6.0699$$

$$x_u = 16, f(x_u) = -2.2688$$

$$x_r = 16 - \frac{-2.2688(12 - 16)}{6.0669 - (-2.2688)} = 14.913$$

which has a true relative error of 0.89 percent.

Second iteration:

$$f(x_l)f(x_r) = -1.5426$$

Therefore, the root lies in the first subinterval, and x_r becomes the upper limit for the next iteration, $x_u = 14.9113$

$$x_l = 12, f(x_l) = 6.0699$$

$$x_u = 14.9113, f(x_u) = -0.2543$$

$$x_r = 14.9113 - \frac{-0.2543(12 - 14.9113)}{6.0669 - (-0.2543)} = 14.7942$$

$$\varepsilon_a = \left|\frac{14.7942 - 14.9113}{14.7942}\right| 100\% = 0.79\%$$

which has true relative errors of 0.09 percent

Third iteration:

$$f(x_l)f(x_r) = -0.171$$

Therefore, the root lies in the first subinterval, and x_r becomes the upper limit for the next iteration, $x_u = 14.7942$

$$x_l = 12, f(x_l) = 6.0699$$

$$x_u = 14.7942, f(x_u) = -0.02726$$

$$x_r = 14.7942 - \frac{-0.02726(12 - 14.7942)}{6.0669 - (-0.02726)} = 14.7817$$

$$\varepsilon_a = \left|\frac{14.7817 - 14.7942}{14.7817}\right| 100\% = 0.084\ \%$$

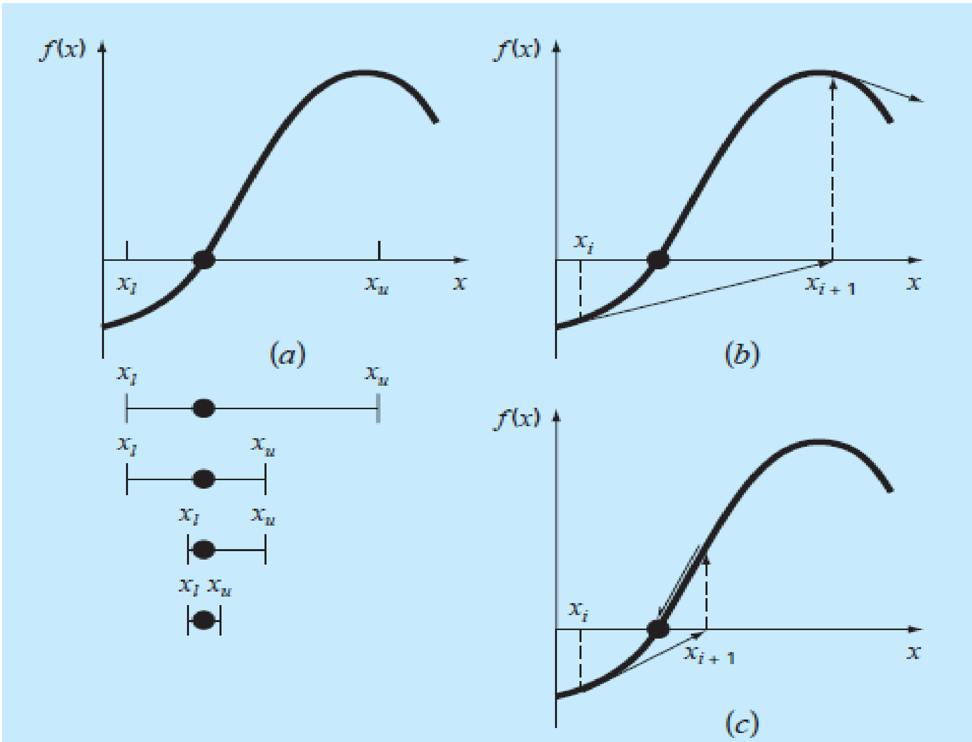
which has true relative errors of 0.01 percent

Note how the error for false position decreases much faster than for bisection because of the more efficient scheme for root location in the false-position method.

Recall in the bisection method that the interval between x_l and x_u grew smaller during the course of a computation. This is not the case for the method of false position because one of the initial guesses may stay fixed throughout the computation as the other guess converges on the root.

3.4 Open methods

The open methods are based on formulas that require only a single starting value of x or two starting values that do not necessarily bracket. Graphical depiction of the fundamental difference between the (a) bracketing and (b) and (c) open methods for root location



. In (a), which is the bisection method, the root is constrained within the interval prescribed by x_l and x_u. In contrast, for the open method depicted in (b) and (c), a formula is used to project from $x_i$to x_{l+1}in an iterative fashion. Thus, the method can either (b) diverge or (c) converge rapidly, depending on the value of the initial guess.

3.4.1 Simple Fixed-Point Iteration

Open methods employ a formula to predict the root. Such a formula can be developed for simple fixed-point iteration (or, as it is also called, one-point iteration or successive substitution) by rearranging the function $f(x) = 0$ so that x is on the left-hand side of the equation:

$$x = g(x)$$

This transformation can be accomplished either by algebraic manipulation or by simply adding x to both sides of the original equation. For example,

$$x^2 - 2x + 3 = 0$$

can be simply manipulated to yield

$$x = \frac{x^2 + 3}{2}$$

whereas $\sin x = 0$ could be put into the form by adding x to both sides to yield

$$x = x + \sin x$$

It provides a formula to predict a new value of x as a function of an old value of x. Thus, given an initial guess at the root x_i, can be used to compute a new estimate x_{i+1} as expressed by the iterative formula

$$x_{i+1} = g(x_i)$$

The approximate error for this equation can be determined using the error estimator:

$$\varepsilon_a = \left| \frac{x_{i+1} - x_i}{x_{i+1}} \right| 100\%$$

Example 4: Use simple fixed-point iteration to locate the root of

$$f(x) = e^{-x} - x$$

Solution. The function can be separated directly and expressed as

$$x_{i+1} = e^{-x_i}$$

Starting with an initial guess of $x_0 = 0$, this iterative equation can be applied to compute

i	x_i	$\varepsilon_a(\%)$	$\varepsilon_t(\%)$
0	0		100
1	1	100	76.3
2	0.3679	171.8	35.1
3	0.6922	46.9	22.1
4	0.5004	38.3	11.8
5	0.6062	17.4	6.89

6	0.5454	11.2	3.83
7	0.5796	5.90	2.2
8	0.5601	3.48	1.24
9	0.5711	1.93	0.705
10	0.5649	1.11	0.399

Thus, each iteration brings the estimate closer to the true value of the root: 0.56714329

➢ Convergence

The two-curve method can now be used to illustrate the convergence and divergence of fixed-point iteration. First,

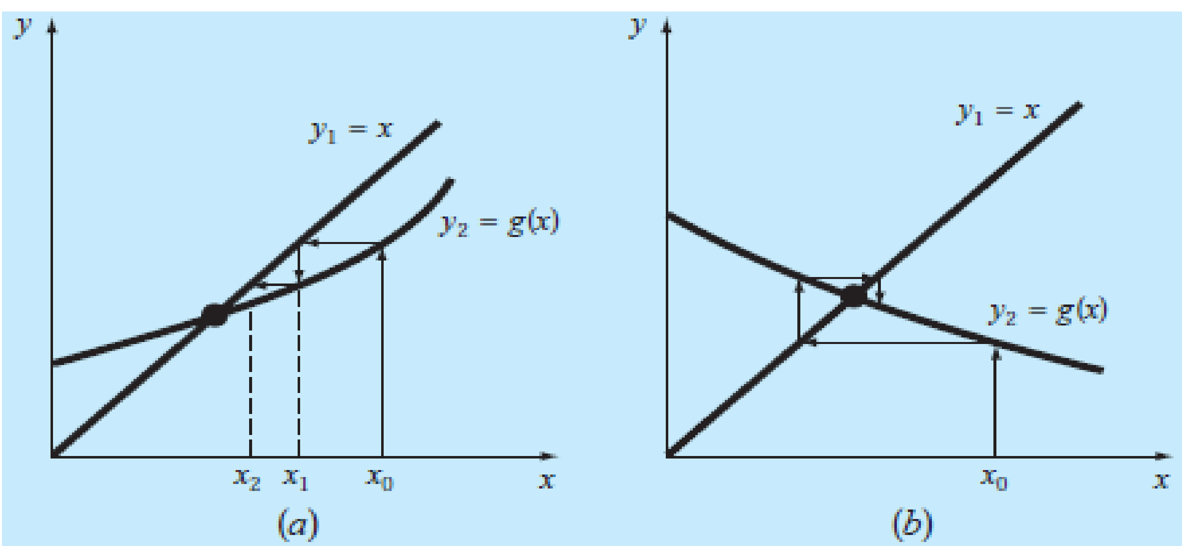
$$x = g(x)$$

can be re-expressed as a pair of equations

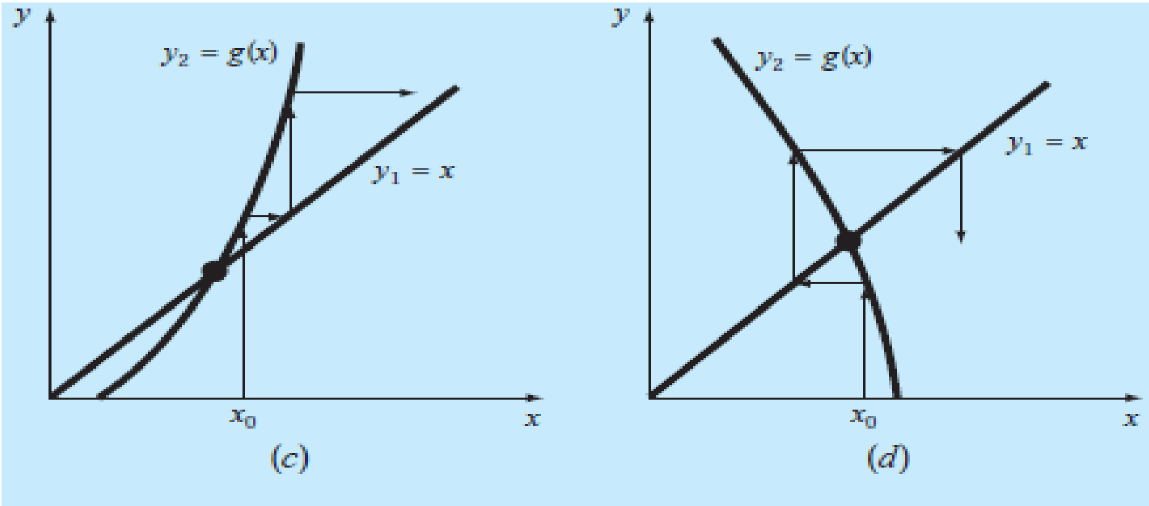
$$y_1 = x \text{ and } y_2 = g(x)$$

These two equations can then be plotted separately. The roots of $f(x) = 0$ correspond to the value at the intersection of the two curves. The function y_1 and four different shapes for y_2 are plotted in the following Figure. Graphical depiction of (*a*) and (*b*) convergence and (*c*) and (*d*) divergence of simple fixed-point iteration. Graphs (*a*) and (*c*) are called monotone patterns, whereas (*b*) and (*d*) are called oscillating or spiral patterns.

The solution in *(a)* is *convergent* because the estimates of *x* move closer to the root with each iteration. The same is true for *(b)*.



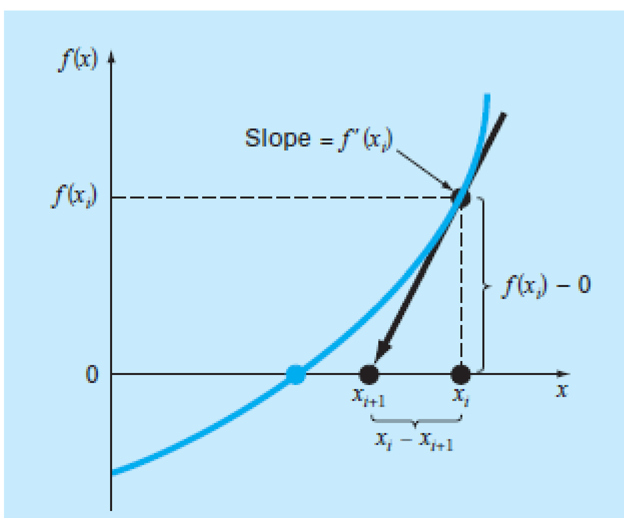
However, this is not the case for *(c)* and *(d)*, where the iterations diverge from the root.



3.4.2 The Newton-Raphson Method

Perhaps the most widely used of all root-locating formulas is the Newton-Raphson equation. If the initial guess at the root is x_i, a tangent can be extended from the point $[x_i, f(x_i)]$ The point where this tangent crosses the x axis usually represents an improved estimate of the root.

The following Figure is Graphical depiction of the Newton-Raphson method. A tangent to the function of x_i $[f'(x_i)]$ is extrapolated down to the x axis to provide an estimate of the root at x_{i+1}



$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}}$$

which can be rearranged to yield

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

which is called the Newton-Raphson formula.

Example 6: Use the Newton-Raphson method to estimate the root of

$$f(x) = e^{-x} - x$$

employing an initial guess of $x_0 = 0$

Solution: The first derivative of the function can be evaluated as

$$f'(x) = -e^{-x} - 1$$

which can be substituted along with Newton-Raphson formula to give

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \rightarrow x_{i+1} = x_i - \frac{e^{-x_i} - x}{-e^{-x_i} - 1}$$

Starting with an initial guess of $x_0 = 0$, this iterative equation can be applied to compute

i	x_i	$\varepsilon_t(\%)$
0	0	100
1	0.5	11.8
2	0.5663110	0.147
3	0.56714316	0.000022
4	0.56714329	$< 10^{-8}$

Thus, the approach rapidly converges on the true root

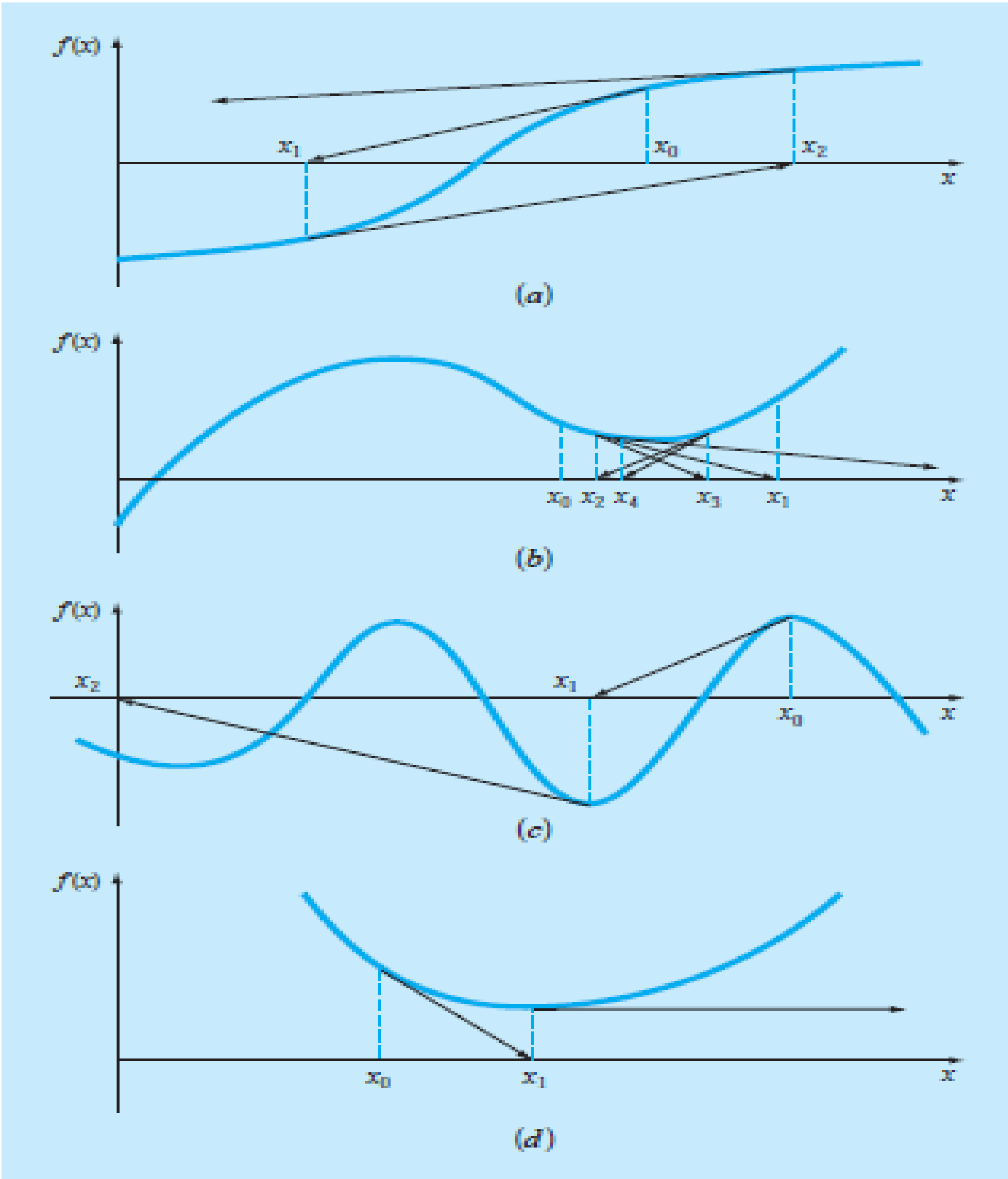
➢ Convergence

There is no general convergence criterion for Newton-Raphson. Its convergence depends on the nature of the function and on the accuracy of the initial guess. Good guesses are usually predicated on knowledge of the physical problem setting or on devices such as graphs that provide insight into the behavior of the solution. The following Figure shows four cases where the Newton-Raphson method exhibits poor convergence due to the nature of the function

For example, (a) depicts the case where an inflection point $[f''(x) = 0]$ occurs in the vicinity of a root. Notice that iterations beginning at x_0 progressively diverge from the root.

(b) illustrates the tendency of the Newton- Raphson technique to oscillate around a local maximum or minimum. Such oscillations may persist, a near-zero slope is reached, whereupon the solution is sent far from the area of interest.

(c) shows how an initial guess that is close to one root can jump to a location several roots away. This tendency to move away from the area of interest is because near-zero slopes are encountered. Obviously, a zero slope $[f'(x) = 0]$ is truly a disaster because it causes division by zero in the Newton-Raphson formula. Graphically (d), it means that the solution shoots off ho



Exercises:

1. Determine the roots of

$f\ (x) = -12 - 21\text{x} + 18x^2 - 2.75x^3$

- **a)** Graphically
- **b)** Bisection
- **c)** False position.

For **(b)** and **(c)** use initial guesses of $x_l = -1$ and $x_u = 0$ and a stopping criterion of 1%

2. Use **(a)** fixed-point iteration and **(b)** the NewtonRaphson method to determine a root of

$f\ (x) = -0.9x^2 + 1.7x + 2.5$

using $x_o = 5$. Perform the computation until ε_a is less than ε_s= 0.01%. Also check your final answer