Chapter 1

The Solution of Nonlinear Equations $f(x) = 0$

Consider the physical problem that involves a spherical ball of radius r that is submerged to a depth d in water(see Figure 1.1). Assume that the ball is constructed from a variety of longleaf pine that has a density of $\rho = 0.638$ and that its radius measures $r = 10$ cm. How much of the ball will be submerged when it is placed in water?

The mass M_w of water displaced when a sphere is submerged to a depth d is

$$M_w = \int_0^d \pi(r^2 - (x - r)^2)dx = \frac{\pi d^2(3r - d)}{3}.$$

and the mass of the ball is $M_b = 4\pi r^3\rho/3$. Applying Archimedes' law $M_w = M_b$, produces the following equation that must be solved:

$$\frac{\pi(d^3 - 3d^2r + 4r^3\rho)}{3} = 0.$$

Figure 1.1 The portion of a sphere of radius r that is to be submerged to depth d.

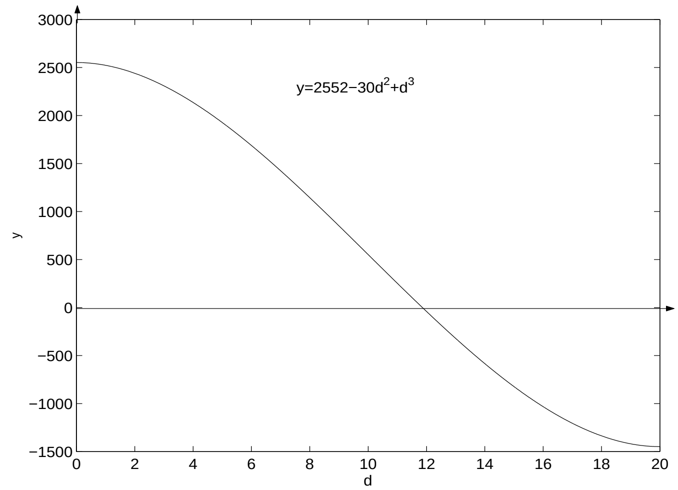


Figure 1.2 The cubic $y = 2552 - 30d^2 + d^3$.

In our case (with $r = 10$ and $\rho = 0.638$) this equation becomes

$$\frac{\pi(2552 - 30d^2 + d^3)}{3} = 0.$$

The graph of the cubic polynomial $y = 2552 - 30d^2 + d^3$ is shown in Figure 1.2 and from it one can see that the solution lies near the value $d = 12$.

The goal of this chapter is to develop a variety of methods for finding numerical approximations for the roots of an equation. For example, the bisection method could be applied to obtain the three roots $d_1 = -8.17607212, d_2 = 11.86150151$, and $d_3 = 26.31457061$. The first root d_1 is not a feasible solution for this problem, because d cannot be negative. The third root d_3 is larger than the diameter of the sphere and it is not the desired solution. The root $d_2 = 11.86150151$ lies in the interval $[0, 20]$ and is the proper solution. Its magnitude is reasonable because a little more than one-half of the sphere must be submerged.

1.1 Iteration for Solving $x = g(x)$

A fundamental principle in computer science is *iteration*. As the name suggests, a process is repeated until an answer is achieved. Iterative techniques are used to find roots equations, solutions of linear and nonlinear systems of equations, and solutions of differential equations. In this section we study the process of iteration using repeated substitution.

A rule or function $g(x)$ for computing successive terms is needed, together with a starting value p_0. Then a sequence of values $\{p_k\}$ is obtained using the iterative rule

$p_{k+1} = g(p_k)$. The sequence has the pattern

$$\begin{aligned} &p_0 && \text{(starting value)} \\ &p_1 = g(p_0) \\ &p_2 = g(p_1) \\ &\vdots \\ &p_k = g(p_{k-1}) \\ &p_{k+1} = g(p_k) \\ &\vdots \end{aligned} \tag{1.1}$$

What can we learn from an unending sequence of numbers? If the numbers tend to a limit, we feel that something has been achieved. But what if the numbers diverge or are periodic? The next example addresses this situation.

Example 1.1. The iterative rule $p_0 = 1$ and $p_{k+1} = 1.001p_k$ for $k = 0, 1, \ldots$ produces a divergent sequence. The first 100 terms look as follows:

$$p_1 = 1.001p_0 = (1.001)(1.000000) = 1.001000,$$

$$p_2 = 1.001p_1 = (1.001)(1.001000) = 1.002001,$$

$$p_3 = 1.001p_2 = (1.001)(1.002001) = 1.003003,$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$p_{100} = 1.001p_{99} = (1.001)(1.104012) = 1.105116.$$

The process can be continued indefinitely, and it is easily shown that $\lim_{n\to\infty} p_n = +\infty$. In Chapter 9 we will see that the sequence $\{p_k\}$ is a numerical solution to the differential equation $y' = 0.001y$. The solution is known to be $y(x) = e^{0.001x}$. Indeed, if we compare the 100th term in the sequence with $y(100)$, we see that $p_{100} = 1.105116 \approx 1.105171 = e^{0.1} = y(100)$.

In this section we are concerned with the types of functions $g(x)$ that produce convergent sequences $\{p_k\}$.

1.1.1 Finding Fixed Points

Definition 1.1 (Fixed Point). A ***fixed point*** of a function $g(x)$ is a real number P such that $P = g(P)$.

Geometrically, the fixed points of a function $y = g(x)$ are the points of intersection of $y = g(x)$ and $y = x$.

Definition 1.2 (Fixed-point Iteration). The iteration $p_{n+1} = g(p_n)$ for $n = 0, 1, \ldots$ is called ***fixed-point iteration***.

Theorem 1.1. Assume that g is a continuous function and that $\{p_n\}_{n=0}^{\infty}$ is a sequence generated by fixed-point iteration. If $\lim_{n\to\infty} p_n = P$, then P is a fixed point of $g(x)$.

Proof. If $\lim_{n\to\infty} p_n = P$, then $\lim_{n\to\infty} p_{n+1} = P$. It follows from this result, the continuity of g, and the relation $p_{n+1} = g(p_n)$ that

$$g(P) = g(\lim_{n\to\infty} p_n) = \lim_{n\to\infty} g(p_n) = \lim_{n\to\infty} p_{n+1} = P. \tag{1.2}$$

Therefore, P is a fixed point of $g(x)$.

Example 1.2. Consider the convergent iteration

$$p_0 = 0.5 \quad \text{and} \quad p_{k+1} = e^{-p_k} \quad \text{for } k = 0, 1, \ldots .$$

The first 10 terms are obtained by the calculations

$$\begin{aligned}
p_1 &= e^{-0.500000} = 0.606531 \\
p_2 &= e^{-0.606531} = 0.545239 \\
p_3 &= e^{-0.545239} = 0.579703 \\
&\vdots \qquad\quad \vdots \\
p_9 &= e^{-0.566409} = 0.567560 \\
p_{10} &= e^{-0.567560} = 0.566907
\end{aligned}$$

The sequence is converging, and further calculations reveal that

$$\lim_{n\to\infty} p_n = 0.567143 \ldots .$$

Thus we have found an approximation for the fixed point of the function $y = e^{-x}$.

The following two theorems establish conditions for the existence of a fixed point and the convergence of the fixed-point iteration process to a fixed point.

Theorem 1.2 Assume that $g \in C[a, b]$.

If the range of the mapping $y = g(x)$ satisfies $y \in [a, b]$ for all $x \in [a, b]$, then g has a fixed point in $[a, b]$. (1.3)

Furthermore, suppose that $g'(x)$ is defined over (a, b) and that a positive constant $K < 1$ exists with $|g'(x)| \leq K < 1$ for all $x \in (a, b)$, then g has a unique fixed point P in $[a, b]$. (1.4)

Proof of (1.3). If $g(a) = a$ or $g(b) = b$, the assertion is true. Otherwise, the values of $g(a)$ and $g(b)$ must satisfy $g(a) \in (a, b]$ and $g(b) \in [a, b)$. The function $f(x) = x - g(x)$ has the property that

$$f(a) = a - g(a) < 0 \quad \text{and} \quad f(b) = b - g(b) > 0.$$

Now apply Theorem 0.2, the Intermediate Value Theorem, to $f(x)$, with the constant $L = 0$, and conclude that there exists a number P with $P \in (a, b)$ so that $f(P) = 0$. Therefore, $P = g(P)$ and P is the desired fixed point of $g(x)$.

Proof of (1.4). Now we must show that this solution is unique. By way of contradiction, let us make the additional assumption that there exist two fixed points P_1 and P_2. Now apply Theorem 0.6, the Mean Value Theorem, and conclude that there exists a number $d \in (a, b)$ so that

$$g'(d) = \frac{g(P_2) - g(P_1)}{P_2 - P_1}. \tag{1.5}$$

Next, use the facts that $g(P_1) = P_1$ and $g(P_2) = P_2$ to simplify the right side of equation (1.5) and obtain

$$g'(d) = \frac{P_2 - P_1}{P_2 - P_1} = 1.$$

But this contradicts the hypothesis in (1.4) that$|g'(x)| < 1$ over (a, b), so it is not possible for two fixed points to exist. Therefore, $g(x)$ has a unique fixed point P in $[a, b]$ under the conditions given in (1.4).

Example 1.3. Apply Theorem 1.2 to rigorously show that $g(x) = \cos(x)$ has a unique fixed point in $[0, 1]$.

Clearly, $g \in C[0, 1]$ secondly, $g(x) = \cos(x)$ is a decreasing function on $[0, 1]$, thus its range on $[0, 1]$ is $[\cos(1), 1] \subseteq [0, 1]$. Thus condition (3) of Theorem 2.2 is satisfied and g has a fixed point in $[0, 1]$. Finally, if $x \in (0, 1)$, then $|g'(x)| = |-\sin(x)| = \sin(x) \leq \sin(1) < 0.8415 < 1$. Thus $K = \sin(1) < 1$, condition (1.4) of Theorem 1.2 is satisfied, and g has a unique fixed point in $[0, 1]$.

We can now state a theorem that can be used to determine whether the fixed-point iteration process given in (1.1) will produce a convergent or divergent sequence.

Theorem 1.3 (Fixed-point Theorem). Assume that (i) $g, g' \in C[a, b]$, (ii) K is a positive constant, (iii) $P_0 \in (a, b)$, and (iv) $g(x) \in [a, b]$ for all $x \in [a, b]$.

> If $|g'(x)| \leq K < 1$ for all $x \in [a, b]$, then the iteration $p_n = g(p_{n-1})$ will converge to the unique fixed point $P \in [a, b]$. In this case, P is said to be an attractive fixed point. (1.6)

> If $|g'(x)| > 1$ for all $x \in [a, b]$,then the iteration $p_n = g(p_{n-1})$ will not converge to P. In this case, P is said to be a repelling fixed point and the iteration exhibits local divergence. (1.7)

Figure 1.3 The relationship among $P, p_0, p_1, |P - p_0|$ and $|P - p_1$

Remark 1. It is assumed that $p_0 \neq P$ in statement (1.7)
Remark 2. Because g is continuous on an interval containing P, it is permissible to use the simpler criterion $|g'(P)| \leq K < 1$ and $|g'(P)| > 1$ in (1.6) and (1,7), respectively.
Proof. We first show that the points $\{p_n\}_{n=0}^{\infty}$ all lie in (a, b). Starting with p_0, we apply Theorem 0.6, the Mean Value Theorem. There exists a value $c_0 \in (a, b)$ so that

$$\begin{aligned} |P - p_1| &= |g(P) - g(p_0)| = |g'(c_0)(P - p_0)| \\ &= |g'(c_0)||P - p_0| \leq K|P - p_0| < |P - p_0|. \end{aligned} \tag{1.8}$$

Therefore, p_1 is no further from P than P_0 was, and it follows that $p_1 \in (a, b)$ (see Figure 1.3). In general, suppose that $p_{n-1} \in (a, b)$; then

$$\begin{aligned} |P - p_n| &= |g(P) - g(p_{n-1})| = |g'(c_{n-1})(P - p_{n-1})| \\ &= |g'(c_{n-1})||P - p_{n-1})| \leq K|P - p_{n-1}| < |P - p_{n-1})|. \end{aligned} \tag{1.9}$$

Therefore, $p_n \in (a, b)$ and hence, by induction, all the points $\{p_n\}_{n=0}^{\infty}$ lie in (a, b).

To complete the proof of (1.6), we will show that

$$\lim_{n \to \infty} |P - p_n| = 0. \tag{1.10}$$

First, a proof by induction will establish the inequality

$$|P - p_n| \leq K^n |P - p_0|. \tag{1.11}$$

The case $n = 1$ follows from the details in relation (1.8). Using the induction hypothesis $|P - p_{n-1}| \leq K^{n-1}|P - p_0|$ and the ideas in (1.9), we obtain

$$|P - p_n| \leq K|P - p_{n-1}| \leq KK^{n-1}|P - p_0| = K^n|P - p_0|.$$

Thus, by induction, inequality (1.11) holds for all n, Since $0 < K < 1$, the term K^n goes to zero as n goes to infinity. Hence

$$0 \leq \lim_{n \to \infty} |P - p_n| \leq \lim_{n \to \infty} K^n|P - p_0| = 0. \tag{1.12}$$

The limit of $|P - p_n|$ is squeezed between zero on the left and zero on the right, so we can conclude that $\lim_{n \to \infty} |P - p_n| = 0$. Thus $\lim_{n \to \infty} p_n = P$ and, by Theorem 1.1, the iteration $p_n = g(p_{n-1})$ converges to the fixed point P. Therefore, statement (1.6) of Theorem 1.3 is proved. We leave statement (1.7) for the reader to investigate.

Figure 1.4 (a) Monotone convergence when $0 < g'(P) < 1$.

Figure 1.4 (a) Monotone convergence when $-1 < g'(P) < 0$.

Corollary 1.1. Assume that g satisfies the hypothesis given in (1.6) of Theorem 1.3. Bounds for the error involved when using p_n to approximate P are given by

$$|P - p_n| \leq K^n |P - p_0| \quad \text{for all } n \geq 1, \tag{1.13}$$

and

$$|P - p_n| \leq \frac{K^n |P - p_0|}{1 - K} \quad \text{for all } n \geq 1, \tag{1.14}$$

Figure 1.5 (a) Monntone divergence when $1 < g'(P)$.

Figure 1.5 (b) Divergent oscillation when $g'(P) < -1$.

1.1.2 Graphical Interpretation of Fixed-point Iteration

Since we seek a fixed point P to $g(x)$, it is necessary that the graph of the curve $y = g(x)$ and the line $y = x$ intersect at the point (P, P). Two simple types of convergent iteration, monotone and oscillating, are illustrated in Figure 1.4(a) and(b), respectively.

To visualize the process, start at p_0 on the x-axis and move vertically to the point $(p_0, p_1) = (p_0, g(p_0))$ on the curve $y = g(x)$. Then move horizontally from (p_0, p_1) to the point (p_1, p_1) on the line $y = x$. Finally, move vertically downward to p_1 on the x-axis. The recursion $p_{n+1} = g(p_n)$ is used to construct the point (p_n, p_{n+1}) on the graph, then a horizontal motion locates (p_{n+1}, p_{n+1}) on the line $y = x$, and then a vertical movement ends up at p_{n+1} on the x-axis. The situation is shown in Figure1.4.

If $|g'(P)| > 1$, then the iteration $p_{n+1} = g(p_n)$ produces a sequence that diverges away from P. The two simple types of divergent iteration, monotone and oscillating, are illustrated in Figure 1.5 (a) and (b), respectively.

Example 1.4. Consider the iteration $p_{n+1} = g(p_n)$ when the function $g(x) = 1 + x - x^2/4$ is used. The fixed points can be found by solving the equation $x = g(x)$. The two solutions (fixed points of g) are $x = -2$ and $x = 2$. The derivative of the function is $g'(x) = 1 - x/2$, and there are only two cases to consider.

Case(i) $P = -2$

Start with $p_0 = -2.05$

then get

$$p_1 = -2.100625$$
$$p_2 = -2.20378135$$
$$p_3 = -2.41794441$$
$$\vdots$$
$$\lim_{n\to\infty} p_n = -\infty$$

Since $|g'(x)| > \frac{3}{2}$ on $[-3, -1]$, by Theorem 1.3, the sequence will not converge to $P = -2$

Case(ii): $P = 2$

Start with $p_0 = 1.6$

then get

$$p_1 = 1.96$$
$$p_2 = 1.9996$$
$$p_3 = 1.99999996$$
$$\vdots$$
$$\lim_{n\to\infty} p_n = 2.$$

Since $|g'(x)| < \frac{1}{2}$ on $[1, 3]$, by Theorem 1.3, the sequence will converge to $P = 2$.

Theorem 1.3 does not state what will happen when $g'(P) = 1$. The next example has been specially constructed so that the sequence $[p_n]$ converges whenever $p_0 > P$ and it diverges if we choose $p_0 < P$.

Example 1.5. Consider the iteration $p_{n+1} = g(p_n)$ when the function $g(x) = 2(x - 1)^{1/2}$ for $x \geq 1$ is used. Only one fixed point $P = 2$ exists. The derivative is $g'(x) = 1/(x - 1)^{1/2}$ and $g'(2) = 1$, so Theorem 3.3 does not apply. There are two cases to consider when the starting value lies to the left or right of $P = 2$.

Case(i) Start with $p_0 = 1.5$

then get

$$p_1 = 1.41421356$$
$$p_2 = 1.28718851$$
$$p_3 = 1.07179943$$
$$p_4 = 0.53590832$$
$$\vdots$$
$$p_5 = 2(-0.46409168)^{1/2}.$$

Since p_4 lies outside the domain of $g(x)$, the term p_5 cannot be computed

Case(ii): Start with $p_0 = 2.5$

then get

$$p_1 = 2.44948974$$
$$p_2 = 2.40789513$$
$$p_3 = 2.37309514$$
$$p_4 = 2.34358284$$
$$\vdots$$
$$\lim_{n\to\infty} p_n = 2.$$

This sequence is converging too slowly to the value $P = 2$; indeed, $P_{1000} = 2.00398714$.

1.1.3 Absolute and Relative Error Considerations

In Example 1.5, case (ii), the sequence converges slowly, and after 1000 iterations the three consecutive terms are

$$P_{1000} = 2.00398714, \quad P_{1001} = 2.00398317, \quad \text{and} \quad P_{1002} = 2.00397921.$$

This should not be disturbing; after all, we could compute a few thousand more terms and find a better approximation! But what about a criterion for stopping the iteration? Notice that if we use the difference between consecutive terms,

$$|p_{1001} - p_{1002}| = |2.00398317 - 2.00397921| = 0.00000396.$$

Yet the absolute error in the approximation P_{1000} is known to be

$$|P - p_{1000}| = |2.00000000 - 2.00398714| = 0.00398714.$$

This is about 1000 times larger than $|p_{1001} - p_{1002}|$ and it shows that closeness of consecutive terms does not guarantee that accuracy has been achieved. But it is usually the only criterion available and is often used to terminate an iterative procedure.

Program 1.1 (Fixed-Point Iteration). To approximate a solution to the equation $x = g(x)$ starting with the initial guess p_0 and iterating $p_{n+1} = g(p_n)$.

```
Function [k, p, err, P] =fixpt(g, po, tol, max1)
% Input - g is the iteration function input as a string 'g'
%        - po is the initial guess for the fixed point
%        - tol is the tolerance
%        - max1 is the maximum number of iterations
%Output- k is the number of iterations that were carried out
%        - p is the approximation to the fixed point
%        - err is the error in the approximation
%        - P contains the sequence {pn}
P(1)= po;
for k=2:max1
    P(k)=feval(g, P(k-1));
    err=abs(P(k)-P(k-1));
    relerr=err/(abs(P(k))+eps);
    p=P(k);
    if (err<tol) | (relerr<tol), break; end
end
if k == max1
    disp('maximum number of iterations exceeded')
end
P=P';
```

Remark. When using the user-defined function fixpt, it is necessary to input the M-file g.m as a string: 'g' (see MATLAB Appendix).

1.1.4 Exercises for Iteration for Solving $x = g(x)$

1. Determine rigorously if each function has a unique fixed point on the given interval (follow Example 1.3).
(a) $g(x) = 1 - x^2/4$ on $[0, 1]$
(b) $g(x) = 2^{-x}$ on $[0, 1]$
(c) $g(x) = 1/x$ on $[0.5, 5.2]$

2. Investigate the nature of the fixed-point iteration when

$$g(x) = -4 + 4x - \frac{1}{2}x^2.$$

(a) Solve $g(x) = x$ and show that $P = 2$ and $P = 4$ are fixed points.
(b) Use the starting value $p_0 = 1.9$ and compute p_1, p_2, and p_3.
(c) Use the starting value $p_0 = 3.8$ and compute p_1, p_2, and p_3.
(d) Find the errors E_k and relative errors R_k for the values p_k in parts (b) and (c).
(e) What conclusions can be drawn from Theorem 1.3?

3. Graph $g(x)$, the line $y = x$, and the given fixed point P on the same coordinate system. Using the given starting value p_0, compute p_1 and p_2. Construct figures similar to Figures 1.4 and 1.5. Based on your graph, determine geometrically if fixed-point iteration converges.
(a) $g(x) = (6 + x)^{1/2}, P = 3$, and $p_0 = 7$
(b) $g(x) = 1 + 2/x, P = 2$, and $p_0 = 4$
(c) $g(x) = x^2/3, P = 3$, and $p_0 = 3.5$
(d) $g(x) = -x^2 + 2x + 2, P = 2$, and $p_0 = 2.5$

4. Let $g(x) = x^2 + x - 4$. Can fixed-point iteration be used to find the solution(s) to the equation $x = g(x)$? Why?
5. Let $g(x) = x\cos(x)$. Solve $x = g(x)$ and find all the fixed points of g (there are infinitely many). Can fixed-point iteration be used to find the solution(s) to the equation $x = g(x)$? Why?
6. Suppose that $g(x)$ and $g'(x)$ are defined and continuous on (a, b); $p_0, p_1, p_2 \in (a, b)$; and $p_1 = g(p_0)$ and $p_2 = g(p_1)$. Also, assume that there exists a constant K such that $|g'(x)| < K$. Show that $|p_2 - p_1| < K|p_1 - p_0|$. *Hint.* Use the Mean Value Theorem.
7. Suppose that $g(x)$ and $g'(x)$ are continuous on (a, b) and that $|g'(x)| > 1$ on this interval. If the fixed point P and the initial approximations p_0 and p_1 lie in the interval (a, b), then show that $p_1 = g(p_0)$ implies that $|E_1| = |P - p_1| > |P - p_0| = |E_0|$. Hence statement (1.7) of Theorem 1.3 is established (local divergence).
8. Let $g(x) = -0.0001x^2 + x$ and $p_0 = 1$, and consider fixed-point iteration.
(a) Show that $p_0 > p_1 > \cdots > p_n > p_{n+1} > \cdots$.
(b) Show that $p_n > 0$ for all n.

(c) Since the sequence $\{p_n\}$ is decreasing and bounded below, it has a limit. What is the limit?

9. Let $g(x) = 0.5x + 1.5$ and $p_0 = 4$, and consider fixed-point iteration.
(a) Show that the fixed point is $P = 3$.
(b) Show that $|P - p_n| = |P - p_{n-1}|/2$ for $n = 1, 2, 3, \ldots$.
(c) Show that $|P - p_n| = |P - p_0|/2^n$ for $n = 1, 2, 3, \ldots$.

10. Let $g(x) = x/2$, and consider fixed-point iteration.
(a) Find the quantity $|p_{k+1} - p_k|/p_{k+1}$.
(b) Discuss what will happen if only the relative error stopping criterion were used in Program 1.1.
11. For fixed-point iteration, discuss why it is an advantage to have $g'(P) \approx 0$.

1.1.5 Algorithms and Programs

1. Use Program 1.1 to approximate the fixed points (if any) of each function. Answers should be accurate to 12 decimal places. Produce a graph of each function and the line $y = x$ that clearly shows any fixed points.
(a) $g(x) = x^5 - 3x^3 - 2x^2 + 2$
(b) $g(x) = \cos(\sin(x))$
(c) $g(x) = x^2 - \sin(x + 0.15)$
(d) $g(x) = x^{x-\cos(x)}$

1.2 Bracketing Methods for Locating a Root

Consider a familiar topic of interest. Suppose that you save money by making regular monthly deposits P and the annual interest rate is I; then the total amount A after N deposits is

$$A = P + P\left(1 + \frac{I}{12}\right) + P\left(1 + \frac{I}{12}\right)^2 + \cdots + P\left(1 + \frac{I}{12}\right)^{N-1}. \tag{1.15}$$

The first term on the right side of equation (1.15) is the last payment. Then the next-to-last payment, which has earned one period of interest, contributes $P(1 + \frac{I}{12})$. The second-from-last payment has earned two periods of interest and contributes $P(1+\frac{I}{12})^2$, and so on. Finally, the last payment, which has earned interest for $N - 1$ periods, contributes $P(1 + \frac{I}{12})^{N-1}$ toward the total. Recall that the formula for the sum of the N terms of a geometric series is

$$1 + r + r^2 + r^3 + \cdots + r^{N-1} = \frac{1 - r^N}{1 - r}. \tag{1.16}$$

We can write (1.15) in the form

$$A = P\left(1 + \left(1+\frac{I}{12}\right) + \left(1+\frac{I}{12}\right)^2 + \cdots + \left(1+\frac{I}{12}\right)^{N-1}\right).$$

and use the substitution $r = (1 + I/12)$ in (1.16) to obtain

$$A = P\frac{1-(1+\frac{I}{12})^N}{1-(1+\frac{I}{12})}.$$

This can be simplified to obtain the annuity-due equation,

$$A = \frac{P}{I/12}\left(\left(1+\frac{I}{12}\right)^N - 1\right). \tag{1.17}$$

The following example uses the annuity-due equation and requires a sequence of repeated calculations to find an answer.

Example 1.6. You save \$250 per month for 20 years and desire that the total value of all payments and interest is \$250,000 at the end of the 20 years. What interest rate I is needed to achieve your goal? If we hold $N = 240$ fixed, then A is a function of I alone; that is $A = A(I)$. We will start with two guesses, $I_0 = 0.12$ and $I_1 = 0.13$, and perform a sequence of calculations to narrow down the final answer. Starting with $I_0 = 0.12$ yields

$$A(0.12) = \frac{250}{0.12/12}\left(\left(1+\frac{0.12}{12}\right)^{240} - 1\right) = 247,314.$$

Since this value is a little short of the goal, we next try $I_1 = 0.13$:

$$A(0.13) = \frac{250}{0.13/12}\left(\left(1+\frac{0.13}{12}\right)^{240} - 1\right) = 282,311.$$

This is a little high, so we try the value in the middle $I_2 = 0.125$:

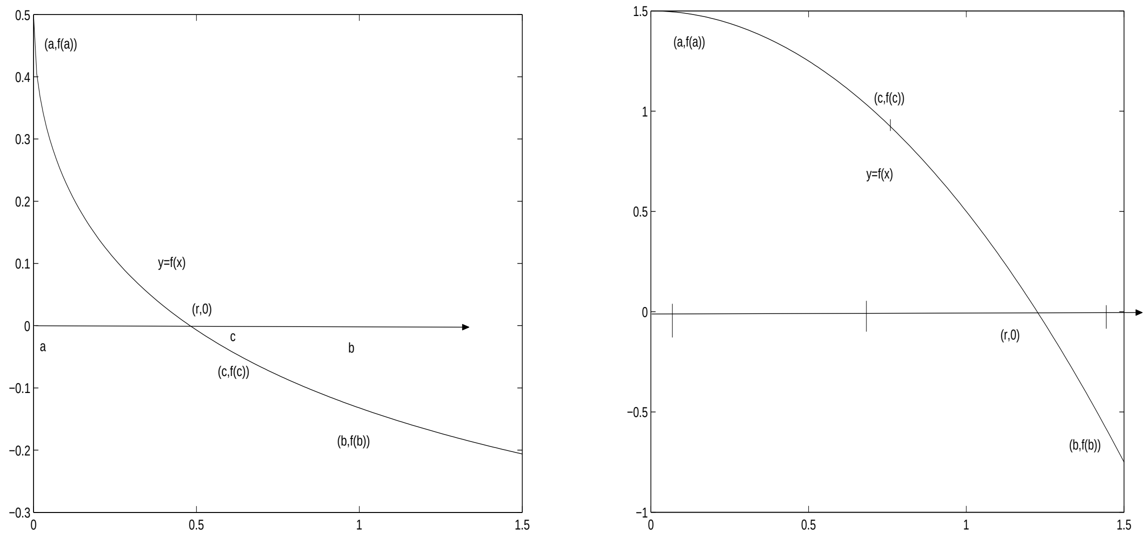
$$A(0.125) = \frac{250}{0.125/12}\left(\left(1+\frac{0.125}{12}\right)^{240} - 1\right) = 264,623.$$

This is again high and we conclude that the desired rate lies in the interval $[0.12, 0.125]$. The next guess is the midpoint $I_3 = 0.1225$:

$$A(0.1225) = \frac{250}{0.1225/12}\left(\left(1+\frac{0.1225}{12}\right)^{240} - 1\right) = 255,803.$$

This is high and the interval is now narrowed to $[0.12, 0.1225]$. Our last calculation uses the midpoint approximation $I_4 = 0.12125$:

$$A(0.12125) = \frac{250}{0.12125/12}\left(\left(1+\frac{0.12125}{12}\right)^{240} - 1\right) = 251,518.$$



(a) If $f(a)$ and $f(c)$ have opposite signs then squeeze from the right.
(b) If $f(c)$ and $f(b)$ have opposite signs then squeeze from the left.

Figure 1.6 The decision process for the bisection process.

Further iterations can be done to obtain as many significant digits as required. The purpose of this example was to find the value of I that produced a specified level L of the function value, that is to find a solution to $A(I) = L$. It is standard practice to place the constant L on the left and solve the equation $A(I) - L = 0$.

Definition 1.3 (Root of an Equation, Zero of a Function). Assume that $f(x)$ is a continuous function. Any number r for which $f(r) = 0$ is called a ***root of the equation*** $f(x) = 0$. Also, we say r is a ***zero of the function*** $f(x)$.

For example, the equation $2x^2+5x-3 = 0$ has two real roots $r_1 = 0.5$ and $r_3 = -3$, whereas the corresponding function $f(x) = 2x^2+5x-3 = (2x-1)(x+3)$ has two real zeros, $r_1 = 0.5$ and $r_2 = -3$.

1.2.1 The Bisection Method of Bolzano

In this section we develop our first bracketing method for finding a zero of a continuous function. We must start with an initial interval $[a, b]$, where $f(a)$ and $f(b)$ have opposite signs. Since the graph $y = f(x)$ of a continuous function is unbroken, it will cross the x-axis at a zero $x = r$ that lies somewhere in the interval (see Figure 1.6). The bisection method systematically moves the end points of the interval closer and closer together until we obtain an interval of arbitrarily small width that brackets the zero. The decision step for this process of interval halving is first to choose the midpoint

$c = (a+b)/2$ and then to analyze the three possibilities that might arise:

$$\text{If } f(a) \text{ and } f(c) \text{ have opposite signs, a zero lies in } [a,c]. \tag{1.18}$$

$$\text{If } f(c) \text{ and } f(b) \text{ have opposite signs, a zero lies in } [c,b]. \tag{1.19}$$

$$\text{If } f(c) = 0, \text{ then the zero is } c. \tag{1.20}$$

If either case (1.18) or (1.19) occurs, we have found an interval half as wide as the original interval that contains the root, and we are "squeezing down on it" (see Figure 1.6). To continue the process, relabel the new smaller interval $[a,b]$ and repeat the process until the interval is as small as desired. Since the bisection process involves sequences of nested intervals and their midpoints, we will use the following notation to keep track of the details in the process:

$[a_0, b_0]$ is the starting interval and$c_0 = \frac{a_0+b_0}{2}$ is the midpoint.
$[a_1, b_1]$ is the second interval, which brackets the zero r, and c_1 is its midpoint; the interval $[a_1, b_1]$ is half as wide as $[a_0, b_0]$.
After arriving at the$n$th interval $[a_n, b_n]$, which brackets r and has midpoint c_n, the interval $[a_{n+1}, b_{n+1}]$ is constructed, which also brackets r and is half as wide as $[a_n, b_n]$.

(1.21)

It is left as an exercise for the reader to show that the sequence of left end points is increasing and the sequence of right end points is decreasing; that is,

$$a_0 \le a_1 \le \cdots \le a_n \le \cdots \le r \le \cdots \le b_n \le \cdots \le b_1 \le b_0, \tag{1.22}$$

where $c_n = \frac{a_n+b_n}{2}$, and if $f(a_{n+1})f(b_{n+1}) < 0$, then

$$[a_{n+1}, b_{n+1}] = [a_n, c_n] \quad \text{or} \quad [a_{n+1}, b_{n+1}] = [c_n, b_n] \quad \text{for all } n \tag{1.23}$$

Theorem 1.4 (Bisection Theorem). Assume that $f \in C[a,b]$ and that there exists a number $r \in [a,b]$ such that $f(r) = 0$. If $f(a)$ and $f(b)$ have opposite signs, and $\{c_n\}_{n=0}^{\infty}$ represents the sequence of midpoints generated by the bisection process of (1.22) and (1.23), then

$$|r - c_n| \le \frac{b-a}{2^{n+1}} \qquad n = 0, 1, \ldots, \tag{1.24}$$

and there fore the sequence $\{c_n\}_{n=0}^{\infty}$ converges to the zero $x = r$; that is,

$$\lim_{n\to\infty} c_n = r. \tag{1.25}$$

Proof. Since both the zero r and the midpoint lie in the interval $[a_n, b_n]$, the distance between c_n and r cannot be greater than half the width of this interval (see Figure 1.7). Thus

$$|r - c_n| \le \frac{b_n - a_n}{2} \quad \text{for all } n \tag{1.26}$$

Observe that the successive interval widths form the pattern

$$b_1 - a_1 = \frac{b_0 - a_0}{2^1}$$

$$b_2 - a_2 = \frac{b_1 - a_1}{2} = \frac{b_0 - a_0}{2}.$$

It is left as an exercise for the reader to use mathematical induction and show that

$$b_n - a_n = \frac{b_0 - a_0}{2^n}. \tag{1.27}$$

Combining (1.26) and (1.27) results in

$$|r - c_n| \leq \frac{b_0 - a_0}{2^{n+1}} \quad \text{for all } n. \tag{1.28}$$

Now an argument similar to the one given in Theorem 1.3 can be used to show that (1.28) implies that the sequence $\{c_n\}_{n=0}^{\infty}$ converges to r and the proof of the theorem is complete.

Example 1.7. The function $h(x) = x\sin(x)$ occurs in the study of undamped forced oscillations. Find the value of x that lies in the interval $[0, 2]$, where the function takes on the value $h(x) = 1$ (the function $\sin(x)$ is evaluated in radians).

We use the bisection method to find a zero of the function $f(x) = x\sin(x) - 1$. Starting with $a_0 = 0$ and $b_0 = 2$, we compute

$$f(0) = -1.000000 \quad \text{and} \quad f(2) = 0.818595,$$

so a root of $f(x) = 0$ lies in the interval $[0, 2]$. At the midpoint $c_0 = 1$, we find that $f(1) = 0.158529$. Hence the function changes sign on $[c_0, b_0] = [1, 2]$.

To continue, we squeeze from the left and set $a_1 = c_0$ and $b_1 = b_0$. The midpoint is $c_1 = 1.5$ and $f(c_1) = 0.496242$. Now, $f(1) = -0.158529$ and $f(1.5) = 0.496242$ imply that the root lies in the interval $[a_1, c_1] = [1.0, 1.5]$. The next decision is to squeeze from the right and set $a_2 = a_1$ and $b_2 = c_1$. In this manner we obtain a sequence $\{c_k\}$ that converges to $r \approx 1.114157141$. A sample calculation is given in Table 1.1.

Table 1.1 Bisection Method Solution of $x\sin(x) - 1 = 0$

k	Left end point, a_k	Midpoint, c_k	Right end point, b_k	Function value, $f(c_k)$
0	0	1	2	−0.158529
1	1.0	1.5	2.0	0.496242
2	1.00	1.25	1.50	0.186231
3	1.000	1.125	1.250	0.015051
4	1.0000	1.0615	1.1250	−0.071827
5	1.06250	1.09375	1.12500	−0.028362
6	1.093750	1.109375	1.125000	−0.006643
7	1.1093750	1.1171875	1.1250000	0.004208
8	1.10937500	1.11328125	1.11718750	−0.001216
⋮	⋮	⋮	⋮	⋮

A virtue of the bisection method is that formula (1.24) provides a predetermined estimate for the accuracy of the computed solution. In Example 1.7 the width of the starting interval was $b_0 - a_0 = 2$. Suppose that Table 1.1 were continued to the thirty-first iterate; then, by (1.24), the error bound would be $|E_{31}| \leq (2-0)/2^{32} \approx 4.656613 \times 10^{-10}$. Hence c_{31} would be an approximation to r with nine decimal places of accuracy. The number N of repeated bisections needed to guarantee that the Nth midpoint c_N is an approximation to a zero and has an error less than the preassigned value δ is

$$N = \text{int}\left(\frac{\ln(b-a) - \ln(\delta)}{\ln(2)}\right) \tag{1.29}$$

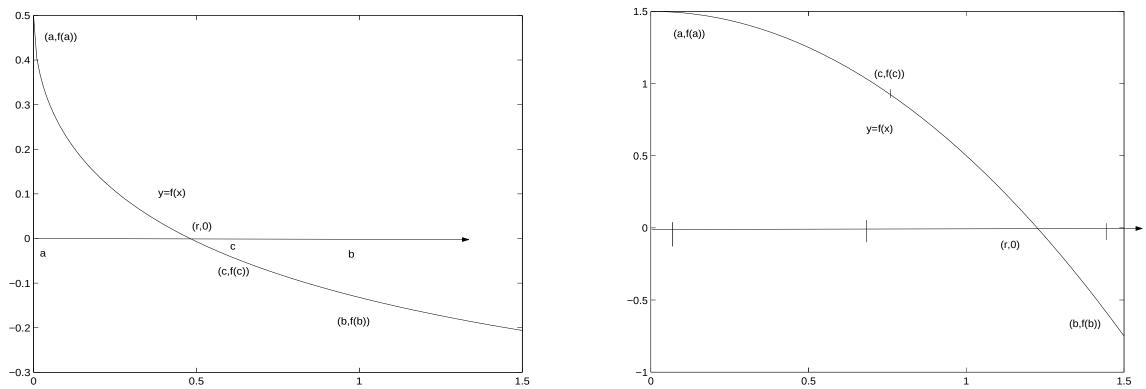
The proof of this formula is left as an exercise.

Another popular algorithm is the ***method of false position*** or the ***regula falsi method***. It was developed because the bisection method converges at a fairly slow speed. As before, we assume that $f(a)$ and $f(b)$ have opposite sings. The bisection method used the midpoint of the interval $[a, b]$ as the next iterate. A better approximation is obtained if we find the point $(c, 0)$ where the secant line L joining the points $(a, f(a))$ and $(b, f(b))$ crosses the x-axis (see Figure 1.8). To find the value c, we write down two versions of the slope m of the line L:

$$m = \frac{f(b) - f(a)}{b - a}, \tag{1.30}$$

where the points $(a, f(a))$ and $(b, f(b))$ are used, and

$$m = \frac{0 - f(b)}{c - b}, \tag{1.31}$$



(a) If $f(a)$ and $f(c)$ have opposite signs then squeeze from the right.
(b) If $f(c)$ and $f(b)$ have opposite signs then squeeze from the left.

Figure 1.8 The decision process for the false position method.

where the points $(c, 0)$ and $(b, f(b))$ are used.

Equating the slopes in (1.30) and (1.31), we have

$$\frac{f(b) - f(a)}{b - a} = \frac{0 - f(b)}{c - b},$$

which is easily solved for c to get

$$c = b - \frac{f(b)(b - a)}{f(b) - f(a)}. \tag{1.32}$$

The three possibilities are the same as before:

$$\text{If } f(a) \text{ and } f(c) \text{ have opposite signs, a zero lies in } [a, c]. \tag{1.33}$$

$$\text{If } f(c) \text{ and } f(b) \text{ have opposite signs, a zero lies in } [c, b]. \tag{1.34}$$

$$\text{If } f(c) = 0, \text{ then the zero is } c. \tag{1.35}$$

1.2.2 Convergence of the False Position Method

The decision process implied by (1.33) and (1.34) along with (1.32) is used to construct a sequence of intervals $\{[a_n, b_n]\}$ each of which brackets the zero. At each step the approximation of the zero r is

$$c_n = b_n - \frac{f(b_n)(b_n - a_n)}{f(b_n) - f(a_n)}, \tag{1.36}$$

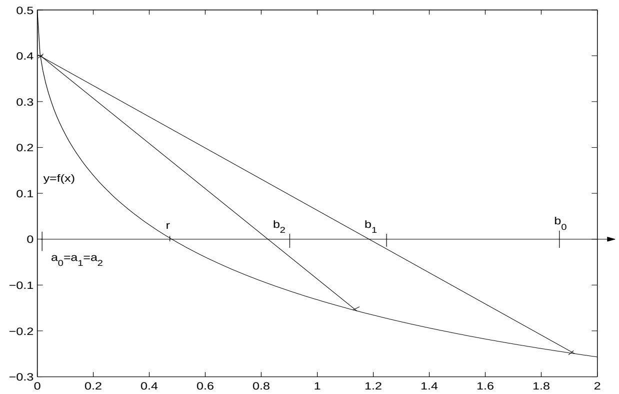


Figure 1.9 The stationary endpoint for the false position method.

and it can be proved that the sequence $\{c_n\}$ will converge to r. But beware, although the interval width $b_n - a_n$ is getting smaller, it is possible that it may not go to zero. If the graph of $y = f(x)$ is concave near $(r, 0)$, one of the end points becomes fixed and the other one marches into the solution (see Figure 1.9).

Now we rework the solution to $x\sin(x) - 1 = 0$ using the method of false position and observe that it converges faster than the bisection method. Also, notice that $\{b_n, a_n\}_{n=0}^{\infty}$ does not go to zero.

Example 1.8. Use the false position method to find the root of $x\sin(x) - 1 = 0$ that is located in the interval $[0, 2]$ (the function $\sin(x)$ is evaluated in radians).

Starting with $a_0 = 0$ and $b_0 = 2$, we have $f(0) = -1.00000000$ and $f(2) = 0.81859485$, so a root lies in the interval $[0, 2]$. Using formula (1.36), we get

$$c_0 = 2 - \frac{0.81859485(2 - 1.09975017)}{0.81859485} = 1.09975017 \quad \text{and} \quad f(c_0) = -0.02001921.$$

The function changes sign on the interval $[c_0, b_0] = [1.09975017, 2]$, so we squeeze from the left and set $a_1 = c_0$ and $b_1 = b_0$. Formula (1.36) produces the next approximation:

$$c_1 = 2 - \frac{0.81859485(2 - 1.9975017)}{0.8185948 - (-0.02001921)} = 1.12124074$$

and

$$f(c_1) = 0.00983461.$$

Next $f(x)$ changes sign on $[a_1, c_1] = [1.09975017, 1.12124074]$, and the next decision is to squeeze from the right and set $a_2 = a_1$ and $b_2 = c_1$. A summary of the calculations is given in Table 1.2.

The termination criterion used in the bisection method is not useful for the false position method and may result in an infinite loop. The closeness of consecutive iterates and the size of $|f(c_n)|$ are both used in the termination criterion for Program 1.3. In section 1.3 we discuss the reasons for this choice.

Table 1.2 False Position Method Solution of $x\sin(x)-1=0$

k	Left end point, a_k	Midpoint, c_k	Right end point, b_k	Function value $f(c_k)$
0	0.00000000	1.09975017	2.000000	-0.02001921
1	1.09975017	1.12124074	2.00000000	0.00983461
2	1.09975017	1.11416120	1.12124074	0.00000563
3	1.09975017	1.11415714	1.11416120	0.00000000

Program 1.2 (Bisection Method). To approximate a root of the equation $f(x)=0$ in the interval $[a,b]$. Proceed with the method only if $f(x)$ is continuous and $f(a)$ and $f(b)$ have opposite signs.

```
function [c,err,yc]=bisect(f,a,b,delta)
%Input   - f is the function input as a string 'f'
%        - a and b are the left and right end points
%        - delta is the tolerance
%Output - c is the zero
%        - yc=f(c)
%        - err is the error estimate for c
ya=feva1(f,a);
yb=feva1(f,b);
if ya*yb>0, break, end
max1=1+round((1og (b-a)-1og (delta))/1og(2));
for k=1:max1
    c=(a+b)/2;
    yc=feva1(f,c);

    if yc==0
      a=c;
      b=c;
elseif yb*yc>0
      b=c; yb=yc;
    else
      a=c;
      ya=yc;

    end
    if b-a<delta, break, end
end
c=(a+b)/2;
err=abs(b-a);
yc=feva1(f,c);
```

Program 1.3 (False Position or Regula Falsi Method). To approximate a root of the equation $f(x) = 0$ in the interval $[a, b]$. Proceed with the method only if $f(x)$ is continuous and $f(a)$ and $f(b)$ have opposite signs.

```
function [c, err, yc] = regula (f, a, b, delta, epsilon, maxl)
%Input     - f is the function input as a string 'f'
%          - a and b are the left and right end points
%          - delta is the tolerance for the zero
%          - epsilon is the tolerance for the value of f at the zero
%          - max1 is the maximum number of iterations
%Output    - c is the zero
%          - yc = f(c)
%          - err is the error estimate for c
ya=feva1(f,a);
yb=feva1(f,a);
if ya*yb>0
    disp ('Note: f(a)*f(b)>0'),
    break,
end
for k=1:max1
    dx=yb*(b-a)/(yb-ba);
    c=b-dx;
    ac=c-a;
    yc=feval(f,c);
    if yc==0, break;
    elseif yb*yc>0
         b=c;
         yb=yc;
    end
    dx=min (abs (dx), ac);
    if abs (dx)<delta, break, end
    if abs (yc)<epsilon, break, end
    end
    c;
    err=abs (b-a)/2;
    yc=feval(f,c);
```

1.2.3 Exercises for Bracketing Methods

In Exercises l and 2, find an approximation for the interest rate I that will yield the total annuity value A if 240 monthly payments P are made. Use the two starting values for I and compute the next three approximations using the bisection method.

1. $P = \$275, A = \$250,000, I_0 = 0.11, I_1 = 0.12$

2. $P = \$325, A = \$400,000, I_0 = 0.13, I_1 = 0.14$

3. For each function, find an interval $[a, b]$ so that $f(a)$ and $f(b)$ have opposite signs.

(a) $f(x) = e^x - 2 - x$

(b) $f(x) = \cos(x) + 1 - x$

(c) $f(x) = ln(x) - 5 + x$

(d) $f(x) = x^2 - 10x + 23$

In Exercises 4 through 7 start with $[a_0, b_0]$ and use the false position method to compute c_0, c_1, c_2 and c_3.

4. $e^x - 2 - x = 0, [a_0, b_0] = [-2.4, -1.6]$

5. $\cos(x) + 1 - x = 0, [a_0, b_0] = [0.8, 1.6]$

6. $\ln(x) - 5 + x = 0, [a_0, b_0] = [3.2, 4.0]$

7. $x^2 - 10x + 23 = 0, [a_0, b_0] = [6.0, 6.8]$

8. Denote the intervals that arise in the bisection method by $[a_0, b_0]$, $[a_1, b_1]$, $\ldots, [a_n, b_n]$.

(a) Show that $a_0 \le a_1 \le \cdots \le a_n \le \cdots$ and that $\cdots \le b_n \le \cdots \le b_1 \le b_0$.

(b) Show that $b_n - a_n = (b_0 - a_0)/2^n$.

(c) Let the midpoint of each interval be $c_n = (a_n + b_n)/2$. Show that

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty} c_n = \lim_{n\to\infty} b_n.$$

Hint. Review convergence of monotone sequences in your calculus book.

9. What will happen if the bisection method is used with the function $f(x) = 1/(x - 2)$ and

(a) the interval is $[3, 7]$? **(b)** the interval is $[1, 7]$?

10. What will happen if the bisection method is used with the function $f(x) = \tan(x)$ and

(a) the interval is $[3, 4]$? **(b)** the interval is $[1, 3]$?

11. Suppose that the bisection method is used to find a zero of $f(x)$ in the interval $[2, 7]$. How many times must this interval be bisected to guarantee that the approximation c_N has an accuracy of 5×10^{-9}?

12. Show that formula (1.36) for the false position method is algebraically equivalent to

$$c_n = \frac{a_n f(b_n) - b_n f(a_n)}{f(b_n) - f(a_n)}$$

13. Establish formula (1.29) for determining the number of iterations required in the bisection method. *Hint.* Use $|b-a|/2^{n+1} < \delta$ and take logarithms.

14. The polynomial $f(x) = (x-1)^3(x-2)(x-3)$ has three zeros: $x = 1$ of multiplicity 3 and $x = 2$ and $x = 3$, each of multiplicity 1. If a_0 and b_0 are any two real numbers such that $a_0 < 1$ and $b_0 > 3$, then $f(a_0)f(b_0) < 0$. Thus, on the interval $[a_0, b_0]$ the bisection method will converge to one of the three zeros. If $a_0 < 1$ and $b_0 > 3$ are selected such that $c_n = \frac{a_n+b_n}{2}$ is not equal to $1, 2$, or 3 for any $n \geq 1$, then the bisection method will never converge to which zero(s)? Why?

15. If a polynomial, $f(x)$, has an odd number of real zeal zeros in the interval $[a_0, b_0]$, and each of the zeros is of odd multiplicity, then $f(a_0)f(b_0) < 0$, and the bisection method will converge to one of the zeros. If $a_0 < 1$ and $b_0 > 3$ are selected such that $c_n = \frac{a_n+b_n}{2}$ is not equal to any of the zeros of $f(x)$ for any $n \geq 1$, then the bisection method will never converge to which zero(s)? Why?

1.2.4 Algorithms and Programs

1. Find an approximation (accurate to 10 decimal places) for the interest rate I that will yield a total annuity value of \$500,000 if 240 monthly payments of \$300 are made.

2. Consider a spherical ball of radius $r = 15$ cm that is constructed from a variety of white oak that has a density of $\rho = 0.710$. How much of the ball (accurate to 8 decimal places) will be submerged when it is placed in water?

3. Modify Programs 1.2 and 1.3 to output a matrix analogous to Tables 1.1 and 1.2, respectively (i.e., the first row of the matrix would be $[0\ a_0\ c_0\ b_0\ f(c_0)]$).

4. Use your programs from Problem 3 to approximate the three smallest positive roots of $x = \tan(x)$ (accurate to 8 decimal places).

5. A unit sphere is cut into two segments by a plane. One segment has three times the volume of the other. Determine the distance x of the plane from the center of the sphere (accurate to 10 decimal places).

1.3 Initial Approximation and Convergence Criteria

The bracketing methods depend on finding an interval $[a, b]$ so that $f(a)$ and $f(b)$ have opposite signs. Once the interval has been found, no matter how large, the iterations will proceed until a root is found. Hence these methods are called ***globally convergent***. However, if $f(x) = 0$ has several roots in $[a, b]$, then a different starting interval must be used to find each root. It is not easy to locate these smaller intervals on which $f(x)$ changes sign.

In Section 1.4 we develop the Newton-Raphson method and the secant method for solving $f(x) = 0$. Both of these methods require that a close approximation to the root be given to guarantee convergence. Hence these methods are called ***locally convergent***. They usually converge more rapidly than do global ones. Some hybrid algorithms start with a globally convergent method and switch to a locally convergent method when the iteration gets close to a root.

If the computation of roots is one part of a larger project, then a leisurely pace is suggested and the first thing to do is graph the function. We can view the graph $y = f(x)$ and make decisions based on what it looks like (concavity, slope, oscillatory behavior, local extrema, inflection points, etc.). But more important, if the coordinates of points on the graph are available, they can be analyzed and the approximate location of roots determined. These approximations can then be used as starting values in our root-finding algorithms.

We must proceed carefully. Computer software packages use graphics software of varying sophistication. Suppose that a computer is used to graph $y = f(x)$ on $[a, b]$. Typically, the interval is partitioned into $N + 1$ equally spaced points: $a = x_0 < x_1 < \cdots < x_N = b$ and the function values $y_k = f(x_k)$ computed. Then either a line segment or a "fitted curve" are plotted between consecutive points (x_{k-1}, y_{k-1}) and (x_k, y_k) for $k = 1, 2, \cdots, N$. There must be enough points so that we do not miss a root in a portion of the curve where the function is changing rapidly. If $f(x)$ is continuous and two adjacent points (x_{k-1}, y_{k-1}) and (x_k, y_k) lie on opposite sides of the x-axis, then the Intermediate Value Theorem implies that at least one root lies in the interval $[x_{k-1}, x_k]$. But if there is a root, even several closely spaced roots, in the interval $[x_{k-1}, x_k]$ and the two adjacent points (x_{k-1}, y_{k-1}) and (x_k, y_k) lie on the same side of the x-axis, then the computer-generated graph would not indicate a situation where the Intermediate Value Theorem is applicable. The graph produced by the computer will not be a true representation of the actual graph of the function f. It is not unusual for functions to have "closely" spaced roots; that is, roots where the graph touches but does not cross the x-axis, or roots "close" to a vertical asymptote. Such characteristics of a function need to be considered when applying any numerical root-finding algorithm.

Finally, near two closely spaced roots or near a double root, the computer-generated curve between (x_{k-1}, y_{k-1}) and (x_k, y_k) may fail to cross or touch the x-axis. If $|f(x_k)|$ is smaller than a preassigned value ε, (i.e., $f(x) \approx 0$), then x_k is a tentative approximate root. But the graph may be close to zero over a wide range of values near x_k, and thus x_k may not be close to an actual root. Hence we add the requirement that the slope change sign near (x_k, y_k); that is, $m_{k-1} = \frac{y_k - y_{k-1}}{x_k - x_{k-1}}$ and $m_k = \frac{y_{k+1} - y_k}{x_{k+1} - x_k}$ must have opposite signs. Since $x_k - x_{k-1} > 0$ and $x_{k+1} - x_k > 0$, it is not necessary to use the difference quotients, and it will suffice to check to see if the differences $y_k - y_{k-1}$ and $y_{k+1} - y_k$ Change sign. In this case, x_k is the approximate root. Unfortunately, we cannot guarantee that this starting value will produce a convergent sequence. If the graph of $y = f(x)$ has a local minimum (or maximum) that is extremely close to zero, then it is possible that x_k will be reported as an approximate root when $f(x_k) \approx 0$, although x_k may not be close to a root.

Table 1.3 Finding Approximate Locations for Roots

	Function value		Difference in y		significant changes
x_k	y_{k-1}	y_k	$y_k - y_{k-1}$	$y_{k+1} - y_k$	in $f(x)$ or $f'(x)$
-1.2	-3.125	-0.968	2.157	1.329	f changes sign
-0.9	-0.968	0.361	1.329	0.663	in $[x_{k-1}, x_k]$
-0.6	0.361	1.024	0.663	0.159	
-0.3	1.024	1.183	0.159	-0.183	f' changes sign
0.0	1.183	1.000	-0.183	-0.363	near x_k
0.3	1.000	0.637	-0.363	-0.381	
0.6	0.637	0.256	-0.381	-0.237	f' changes sign
0.9	0.256	0.019	-0.237	0.069	near x_k
1.2	0.019	0.088	0.069	0.537	

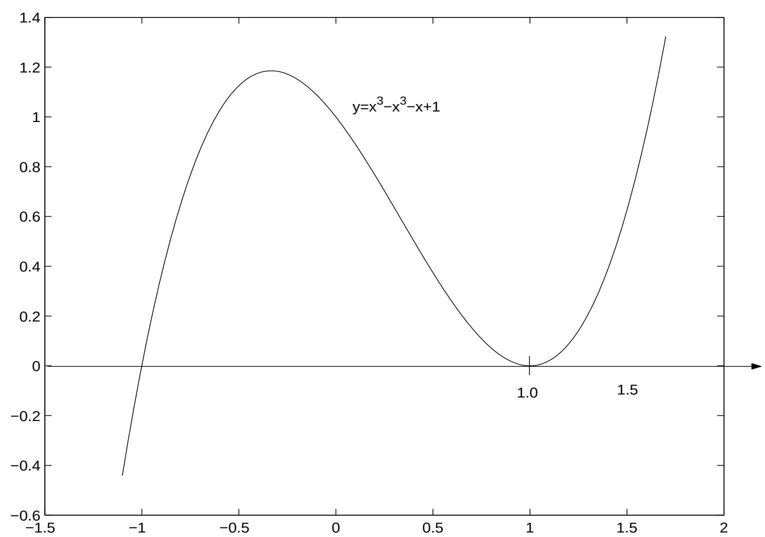


Figure 1.10 The graph of the cubic polynomial $y = x^3 - x^2 - x + 1$.

Example 1.9. Find the approximate location of the roots of $x^3 - x^2 - x + 1 = 0$ on the interval $[-1.2, 1.2]$. For illustration, choose $N = 8$ and look at Table 1.3.

The three abscissas for consideration are $-1.05, 0.3$, and 0.9. Because $f(x)$ changes sign on the interval $[-1.2, 0.9]$, the value 1.05 is an approximate root; indeed, $f(-1.05) = 0.210$.

Although the slope changes sign near -0.3, we find that $f(-0.3) = 1.183$; hence -0.3 is not near a root. Finally, the slope changes sign near 0.9 and $f(0.9) = 0.019$, so 0.9 is an approximate root (see Figure 1.10).

Figure 1.11 (a) The horizontal convergence band for locating a solution to $f(x) = 0$.

Figure 1.11 (b) The vertical convergence band for locating a solution to $f(x) = 0$.

1.3.1 Checking for Convergence

A graph can be used to see the approximate location of a root, but an algorithm must be used to compute a value p_n that is an acceptable computer solution. Iteration is often used to produce a sequence $\{p_k\}$ that converges to a root P, and a termination criterion or strategy must be designed ahead of time so that the computer will stop when an accurate approximation is reached. Since the goal is to solve $f(x) = 0$, the final value p_n should have the property that $|f(p_n)| < \varepsilon$.

The user can supply a tolerance value ε for the size of $|f(p_n)|$ and then an iterative process produces points $P_k = (p_k, f(p_k))$ until the last point P_n lies horizontal band bounded by the lines $y = +\varepsilon$ and, $y = -\varepsilon$, as shown in Figure 1.11(a). This criterion is useful if the user is trying to solve $h(x) = L$ by applying a root-finding algorithm to the function $f(x) = h(x) - L$.

Another termination criterion involves the abscissas, and we can try to determine if the sequence $\{p_k\}$ is converging. If we draw the vertical lines $x = p + \delta$ and $x = p - \delta$ on each side of $x = p$, we could decide to stop the iteration when the point P_n lies between these two vertical lines, as shown in Figure 1.11 (b).

The latter criterion is often desired, but it is difficult to implement because it involves the unknown solution p. We adapt this idea terminate further calculations when the consecutive iterates p_{n-1} and p_n are sufficiently close or if they agree within M significant digits.

Sometimes the user of an algorithm will be satisfied if $p_n \approx p_{n-1}$ and other times when $f(p_n) \approx 0$. Correct logical reasoning is required to understand the consequences. If we require that $|p_n - p| < \delta$ **and** $|f(p_n)| < \varepsilon$, and $|f(p_n)| < \varepsilon$, the point P_n will be located in the rectangular region about the solution $(p, 0)$, as shown in Figure 1.12 (a). If we stipulate that $p_n - p < \delta$ **or** $|f(p_n)| < \varepsilon$, the point p_n could be located anywhere in the region formed by the union of the horizontal and vertical stripes, as shown in Figure 1.12(b). The size of the tolerances δ and ε are crucial. If the tolerances are chosen too small, iteration may continue forever. They should be chosen about 100 times larger than 10^{-M}, where M is the number of decimal digits in the computer's floating-point numbers. The closeness of the abscissas is checked with one of the criteria

$$|p_0 - p_{n-1}| < \delta \quad \text{(estimate for the absolute error)}$$

or

$$\frac{2|p_n - p_{n-1}|}{|p_n| + |p_{n-1}|} < \delta \quad \text{(estimate for the relation error)}$$

The closeness of the ordinate is usually checked by $|f(p_n)| < \varepsilon$.

1.3.2 Troublesome Function

A computer solution to $f(x) = 0$ will almost always be in error due to roundoff and/or instability in the calculations. If the graph $y = f(x)$ is steep near the root $(p, 0)$, then the root-finding problem is well conditioned (i.e., a solution with several significant digits is easy to obtain). If the graph $y = f(x)$ is shallow near $(p, 0)$, then the root-finding problem is ill conditioned (i.e., the computed root may have only a few significant digits). This occurs when $f(x)$ has a multiple root at p. This is discussed further in the next section.

Figure 1.12 (a)The rectangular region defined by $|x - p| < \delta$ AND $|y| < \varepsilon$.

Figure 1.12 (b)The unbounded region defined by $|x - p| < \delta$ OR $y < \varepsilon$.

Program 1.4 (Approximate Location of Roots). To roughly estimate the locations of the roots of the equation $f(x) = 0$ over the interval $[a, b]$, by using the equally spaced sample points $(x_k, f(x_k))$ and the following criteria:
(i) $(y_{k-1})(y_k) < 0$, or
(ii) $|y_k| < \varepsilon$ and $(y_k - y_{k-1}(y_{k+1} - y_k) < 0$.
That is, either $f(x_{k-1})$ and $f(x_k)$ have opposite signs or $|f(x_k)|$ is small and the slope of the curve $y = f(x)$ changes sign near $(x_k, f(x_k))$.

```
function R = approot (x, epsilon)
% Input – f is the object function saved as an M-file named f.m
%        – X is the vector of abscissas
%         – epsilon is the tolerance
% Output – R is the vector of approximate roots
Y=f(x);
yrange = max(Y)–min(Y);
epsilon2 = yrange*epsilon;
n=length(X);
m=0;
X(n+1)=X(n);
Y(n+1)=Y(n);
for k=2:n,
    if Y(k-1)*Y(k)¡=0,
       m=m+1;
       R(m)=(X(k-1)+X(b))/2;
    end
    s=(Y(k)-Y(k-1))*(Y(k+1)-Y(k));
    if (abs(Y(k))¡epsilon2)&(s¡=0),
       m=m+1;
       R(m)=X(k);
    end
end
```

Example 1.10. Use approot to find approximate locations for the roots of $f(x) = \sin(\cos(x^3))$ in the interval $[-2, 2]$. First save f as an M-file named f.m. Since the results will be used as initial approximations for a root-finding algorithm, we will construct X so that the approximations will be accurate to 4 decimal places.

```
>>X=–2:.001:2;
>>approot (X,0.00001)
ans=
–1.9875    –1.6765    –1.1625    1.1625    1.6765    1.9875
```

Comparing the results with the graph of f, we now have good initial approximations for one of our root-finding algorithms.

1.3.3 Exercises for Initial Approximation

In Exercises 1 through 6 use a computer or graphics calculator to graphically determine the approximate location of the roots of $f(x) = 0$ in the given interval. In each case, determine an interval $[a, b]$ over which Programs 1.2 and 1.3 could be used to determine the roots (i.e., $f(a)f(b) < 0$).

1. $f(x) = x^2 - e^x$ for $-2 \leq x \leq 2$

2. $f(x) = x^{-} \cos(x)$ for $-2 \leq x \leq 2$

3. $f(x) = \sin(x) - 2\cos(x)$ for $-2 \leq x \leq 2$

4. $f(x) = \cos(x) + (1 + x^2)^{-1}$ for $-2 \leq x \leq 2$

5. $f(x) = (x - 2)^2 - \ln(x)$ for $0.5 \leq x \leq 4.5$

6. $f(x) = 2x - \tan(x)$ for $-1.4 \leq x \leq 1.4$

1.3.4 Algorithms and Programs

In Problems 1 and 2 use a computer or graphics calculator and Program 1.4 to approximate the real roots, to 4 decimal places, of each function over the given interval. Then use Program 1.2 or Program 1.3 to approximate each root to 12 decimal places.

1. $f(x) = 1,000,000x^3 - x^3 - 111,000x^2 + 1110x$ for $-2 \leq x \leq 2$

2. $f(x) = 5x^{10} - 38x^9 + 21x^8 - 5\pi x^6 - 3\pi x^5 - 5x^2 + 8x - 3$ for $-15 \leq x \leq 15$.

3. A computer program that plots the graph of $y = f(x)$ over the interval $[a, b]$ using the points $(x_0, y_0), (x_1, y_1), \ldots,$ and (x_N, y_N) usually scales the vertical height of the graph, and a procedure must be written to determine the minimum and maximum values of f over the interval.

(a) Construct an algorithm that will find the values $Y_{\max} = \max_k\{y_k\}$ and $Y_{\min} = \min_k\{y_k\}$.

(b) Write a MATLAB program that will find the approximate location and value of the extreme values of $f(x)$ on the interval $[a, b]$.

(c) Use your program from part (b) to find the approximate location and value of the extreme values of the functions in Problems 1 and 2. Compare your approximations with the actual values.

1.4 Newton-Raphson and Secant Methods

1.4.1 Slope Methods for Finding Roots

If $f(x), f'(x)$, and $f''(x)$ are continuous near a root p, then this extra information regarding the nature of $f(x)$ can be used to develop algorithms that will produce sequenced $\{p_k\}$ that converge faster to p than either the bisection of false position method. The Newton-Raphson(or simply Newton's) method is one of the most useful and best known algorithms that relies on the continuity of $f'(x)$ and $f''(x)$. We shall introduce it graphically and then give a more rigorous treatment based on the Taylor polynomial.

Assume that the initial approximation p_0 is near the root p. Then the graph of $y = f(x)$ intersects the x-axis at the point $(p, 0)$, and the point $(p_0, f(p_0))$ lies on the curve neat the point $(p, 0)$ (see Figure 1.13). Define p_1 to be the point of intersection of the x-axis and the line tangent to the curve at the point $(p_0, f(p_0))$. Then Figure 1.13 shows that p_1 will be closer to p than p_0 in this case. An equation relating p_1 and p_0 can be found if we write down two versions for the slope of the tangent line L;

$$m = \frac{0 - f(p_0)}{p_1 - p_0}, \tag{1.37}$$

which is the slope of the line through $(p_1, 0)$ and $(p_0, f(p_0))$, and

$$m = f'(p_0), \tag{1.38}$$

which is the slope at the point $(p_0, f(p_0))$. Equating the values of the slope m in equations (1.37) and (1.38) and solving for p_1 results in

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)}. \tag{1.39}$$

Figure 1.13 The geometric construction of p_1 and p_2 for the Newton-Raphson method.

The process above can be repeated to obtain a sequence $\{p_k\}$ that converges to p. We now make these ideas more precise.

Theorem 1.5 (Newton-Raphson Theorem). Assume that $f \in C^2[a,b]$ and there exists a number $p \in [a,b]$, where $f(p) = 0$. If $f'(p) \neq 0$, then there exists a $\delta > 0$ such that the sequence $\{p_k\}_{k=0}^{\infty}$ defined by the iteration

$$p_k = g(p_{k-1}) = p_{k-1} - \frac{f(p_{k-1})}{f'(p_{k-1})} \quad \text{for } k = 1, 2, \dots \tag{1.40}$$

will converge to p for any initial approximation $p_0 \in [p-\delta, p+\delta]$.
Remark. The function $g(x)$ defined by formula

$$g(x) = x - \frac{f(x)}{f'(x)} \tag{1.41}$$

is called the ***Newton-Raphson iteration function***. Since $f(p) = 0$, it is easy to see that $g(p) = p$. Thus the Newton-Raphson iteration for finding the root of the equation $f(x) = 0$ is accomplished by finding a fixed point the function $g(x)$.

Proof. The geometric construction of p_1 shown in Figure 1.13 does not help in understanding why p_0 needs to be close to p or why the continuity of $f''(x)$ is essential. Our analysis starts with the Taylor polynomial of degree $n = 1$ and its remainder term:

$$f(x) = f(p_0) + f'(p_0)(x - p_0) + \frac{f''(c)(x-p_0)^2}{2!}. \tag{1.42}$$

where c lies somewhere between p_0 and x. Substituting $x = p$ into equation (1.42) and using the fact that $f(p) = 0$ produces

$$0 = f(p_0) + f'(p_0)(p - p_0) + \frac{f''(c)(p-p_0)^2}{2!} \tag{1.43}$$

If p_0 is close enough to p, the last term on the right side of (1.43) will be small compared to the sum of the first two terms. Hence it can be neglected and we can use the approximation

$$0 \approx f(p_0) + f'(p_0)(p - p_0). \tag{1.44}$$

Solving for p in equation (1.44), we get $p \approx p_0 - f(p_0)/f'(p_0)$. This is used to define the next approximation p_1 to the root

$$p_1 = p_0 - \frac{f(p_0)}{f'(p_0)}. \tag{1.45}$$

When p_{k-1} is used in place of p_0 in equation (1.45), the general rule (1.40) is established. For most applications this is all that needs to be understood. However, to fully comprehend what is happening, we need to consider the fixed-point iteration function and apply Theorem 1.2 in our situation.The key is in the analysis of $g'(x)$:

$$g'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{(f'(x))^2} = \frac{f(x)f''(x)}{(f'(x))^2}.$$

By hypothesis, $f(p) = 0$; thus $g'(p) = 0$. Since $g'(p) = 0$ and $g'(x)$ is continuous, it is possible to find a $\delta > 0$ so that the hypothesis $|g'(x)| < 1$ of Theorem 1.2 is satisfied on $(p-\delta, p+\delta)$. Therefore, a sufficient condition for p_0 to initialize a convergent sequence $\{p_k\}_{k=0}^{\infty}$, which converges to a root of $f(x) = 0$, is that $p_0 \in (p-\delta, p+\delta)$ and that δ be chosen so that

$$\frac{|f(x)f''(x)|}{|f'(x)|^2} < 1 \quad \text{for all } x \in (p-\delta, p+\delta) \tag{1.46}$$

Corollary 1.2 (Newton's Iteration for Finding Square Roots). Assume that $A > 0$ is a real number and let $p_0 > 0$ be an initial approximation to $\sqrt{A}$. Define the sequence $\{p_k\}_{k=0}^{\infty}$ using the recursive rule.

$$p_k = \frac{p_{k-1} + \frac{A}{p_{k-1}}}{2} \quad \text{for } k = 1, 2, \ldots. \tag{1.47}$$

Then the sequence $\{p_k\}_{k=0}^{\infty}$ converges to $\sqrt{A}$; that is, $\lim_{n\to\infty} p_k = \sqrt{A}$.
Outline of Proof. Start with the function $f(x) = x^2 - A$, and notice that the roots of the equation $x^2 - A = 0$ are $\pm\sqrt{A}$. Now use $f(x)$ and the derivative $f'(x)$ in formula (1.41) and write down the Newton-Raphson iteration formula

$$g(x) = x - \frac{f(x)}{f'(x)} = x - \frac{x^2 - A}{2x}. \tag{1.48}$$

This formula can be simplified to obtain

$$g(x) = \frac{x + \frac{A}{x}}{2}. \tag{1.49}$$

When $g(x)$ in (1.49) is used to define the recursive iteration in (1.30), the result is formula (1.47). It can be proved that the sequence that is generated in (1.47) will converge for any starting value $p_0 > 0$. The details are left for the exercises.

An important point of Corollary 1.2 is the fact that the iteration function $g(x)$ involved only the arithmetic operations $+, -, \times$, and $/$. If $g(x)$ had involved the calculation of a square root, we would be caught in the circular reasoning that being able to calculate the square root would permit you to recursively define a sequence that will converge to $\sqrt{A}$. For this reason, $f(x) = x^2 - A$ was chosen, because it involved only the arithmetic operations.

Example 1.11. Use Newton's square-root algorithm to find $\sqrt{5}$.

Starting with $p_0 = 2$ and using formula(1.47), we compute

$$p_1 = \frac{2 + 5/2}{2} = 2.25$$

$$p_2 = \frac{2.25 + 5/2.25}{2} = 2.236111111$$

$$p_3 = \frac{2.236111111 + 5/2.236111111}{2} = 2.236067978$$

$$p_4 = \frac{2.36067978 + 5/2.236067978}{2} = 2.236067978.$$

Further iterations produce $p_k \approx 2.236067978$ for $k > 4$, so we see that convergence accurate to nine decimal places hes been achieved.

Now let us turn to a familiar problem from elementary physics and see why determining the location of a root is an important task. Suppose that a projectile is fired the origin with an angle of elevation b_0 and initial velocity v_0. In elementary courses, air resistance is neglected and we learn that the height $y = y(t)$ and the distance traveled $x = x(t)$, measured in feet, obey the rules

$$y = v_y t - 16t^2 \quad \text{and} \quad x = v_x t, \tag{1.50}$$

where the horizontal and vertical components of the initial velocity are $v_x = v_0 \cos(b_0)$ and $v_y = v_0 \sin(b_0)$, respectively. The mathematical model expressed by the rules in (1.50) is easy to work with, but tends to give too high an altitude and too long a range for the projectile' s path. If we make the additional assumption that the air resistance is proportional to the velocity, the equations of motion become

$$y = f(t) = (Cv_y + 32C^2)(1 - e^{-t/C}) - 32Ct \tag{1.51}$$

and

$$x = r(t) = Cv_x(1 - e^{-t/C}), \tag{1.52}$$

where $C = m/k$ and k is the coefficient of air resistance and m is the mass of the projectile. A larger value of C will result in a higher maximum altitude and a longer range for the projectile. The graph of a flight path of a projectile when air resistance is considered is shown in Figure 2.14. This improved model is more realistic, but requires the use of a root-finding algorithm for solving $f(t) = 0$ to determine the elapsed time until the projectile his the ground. The elementary model in (1.50) does not require a sophisticated procedure to find the elapsed time.

Figure 1.14 Path of a projectile with air resistance considered.

Table 1.4 Finding the Time When the Height $f(t)$ Is Zero

k	Time, p_k	$p_{k+1} - p_k$	Height, $f(p_k)$
0	8.00000000	0.79773101	83.22097200
1	8.79773101	−0.05530160	−6.68369700
2	8.74242941	−0.00025475	−0.03050700
3	8.74217467	−0.00000001	−0.00000100
4	8.74217466	0.00000000	0.00000000

Example 1.12. A projectile is fired with an angle of elevation $b_0 = 45°, v_y = v_x =$ 160ft/sec, and $C = 10$. Find the elapsed time until impact and find the range.

Using formulas (1.51) and (1.52), the equations of motion are $y = f(t) = 4800(1 - e^{-t/10}) - 320t$ and $x = r(t) = 1600(1 - e^{-t/10})$. Since $f(8) = 83.220972$ and $f(9) = -31.534367$, we will use the initial guess $p_0 = 8$. The derivative is $f'(t) = 480e^{-t/10} - 320$, and its value $f'(p_0) = f'(8) = -104.3220972$ is used in formula (1.40) to get

$$p_1 = 8 - \frac{83.22097200}{-104.3220972} = 8.797731010.$$

A summary of the calculation is given in Table 1.4.

The value p_4 has eight decimal places of accuracy, and the time until impact is $t \approx 8.74217466$ seconds. The range can now be computed using $r(t)$; and we get

$$r(8.74217466) = 1600(1 - e^{-0.847217466}) = 932.4986302\text{ft}.$$

1.4.2 The Division-by-Zero Error

One obvious pitfall of the Newton-Raphson method is the possibility of division by zero in formula (1.40), which would occur if $f'(p_{k-1}) = 0$. Program 1.5 has a procedure to check for this situation, but what use is the last calculated approximation p_{k-1} in this case? It is quite possible that $f(p_{k-1})$ is sufficiently close to zero and that p_{k-1} is an acceptable approximation to the root. We now investigate this situation and will uncover an interesting fact, that is, how fast the iteration converges.

Definition 1.4 (Order of a Root). Assume that $f(x)$ and its derivatives $f'(x)$, $\ldots, f^{(M)}(x)$ are defined and continuous on an interval about $x = p$. We say that $f(x) = 0$ has a root of order M at $x = p$ if and only if

$$f(p) = 0, \quad f'(p), \quad \ldots, \quad f^{(M-1)}(p) = 0, \quad \text{and} \quad f^{(M)}(p) \neq 0. \tag{1.53}$$

A root of order $M = 1$ is often called a ***simple root***, and if $M > 1$, it is called a ***multiple root***. A root of order $M = 2$ is sometimes called a ***double root***, and so on. The next result will illuminate these concepts.

Lemma 1.1. If the equation $f(x) = 0$ has a root of order M at $x = p$, then there exists a continuous function $h(x)$ so that $f(x)$ can be expressed as the product

$$f(x) = (x - p)^M h(x), \quad \text{where } h(p) \neq 0. \tag{1.54}$$

Example 1.13. The function $f(x) = x^3 - 3x + 2$ has a simple root at $p = -2$ and a double root at $p = 1$. This can be verified by considering the derivatives $f'(x) = 3x^2 - 3$ and $f''(x) = 6x$. At the value $p = -2$, we have $f(-2) = 0$ and $f'(-2) = 9$, so $M = 1$ in Definition 1.4; hence $p = -2$ is a simple root. For the value $p = 1$, we have $f(1) = 0, f'(1) = 0$, and $f''(1) = 6$, so $M = 2$ in Definition 1.4; hence $p = 1$ is a double root. Also, notice that $f(x)$ has the factorization $f(x) = (x + 2)(x - 1)^2$.

1.4.3 Speed of Convergence

The distinguishing property we seek is the following. If p is a simple root of $f(x) = 0$, Newton's method will converge rapidly, and the number of accurate decimal places (roughly) doubles with each iteration. On the other hand, if p is a multiple root, the error in each successive approximation is a fraction of the previous error. To make this precise, we define the ***order of convergence***. The is a measure of how rapidly a sequence converges.

Definition 1.5 (Order of Convergence). Assume that $\{p_n\}_{n=0}^{\infty}$ converges to p and set $E_n = p - p_n$ for $n \geq 0$. If two positive constants $A \neq 0$ and $R > 0$ exist, and

$$\lim_{n\to\infty} \frac{|p - p_{n+1}|}{|p - p_n|^R} = \lim_{n\to\infty} \frac{|E_{n+1}|}{|E_n|^R} = A. \tag{1.55}$$

Table 1.5 Newton's Method Converges Quadratically at a Simple Root

k	p_k	$p_{k+1} - p_k$	$E_k = p - p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert^2}$
0	−2.400000000	0.323809524	0.400000000	0.476190475
1	−2.076190476	0.072594465	0.076190476	0.619469086
2	−2.003596011	0.003587422	0.003596011	0.664202613
3	−2.000008589	0.000008589	0.000008589	
4	−2.000000000	0.000000000	0.000000000	

then the sequence is said to converge to p with order of convergence R. The number A is called the asymptotic error constant. The cases $R = 1, 2$ are given special consideration.

$$\text{If } R = 1, \text{ the convergence of } \{p_n\}_{n=0}^{\infty} \text{ is called } \textbf{\textit{linear}} \tag{1.56}$$

$$\text{If } R = 1 \text{ , the convergence of } \{p_n\}_{n=0}^{\infty} \text{ is called } \textbf{\textit{quadratic}}. \tag{1.57}$$

If R is large, the sequence $\{p_n\}$ converges rapidly to p; that is, relation (1.55) implies that for large values of n we have the approximation $|E_{n+1}| \approx A|E_n|^R$. For example, suppose that $R = 2$ and $|E_n| \approx 10^{-2}$; then we would expect that $|E_{n+1}| \approx A \times 10^{-4}$.

Some sequences converge at a rate that is not an integer, and we will see that the order of convergence of the secant method is $R = (1 + \sqrt{5})/2 \approx 1.618033989$.

Example 1.14 (Quadratic Convergence at a Simple Root). Start with $p_0 = -2.4$ and use Newton-Raphson iteration to find the root $p = -2$ of the polynomial $f(x) = x^3 - 3x + 2$. The iteration formula for computing $\{p_k\}$ is

$$p_k = g(p_{k-1}) = \frac{2p_{k-1}^3 - 2}{3p_{k-1}^2 - 3}. \tag{1.58}$$

Using formula (1.55) to check for quadratic convergence, we get the values in Table 1.5.

A detailed look at the rate of convergence in Example 1.14 will reveal that the error in each successive iteration is proportional to the square of the error in the previous iteration. That is,

$$|p - p_{k+1}| \approx A|p - p_k|^2,$$

where $A \approx 2/3$. To check this. we use

$$|p - p_3| = 0.000008589 \quad \text{and} \quad |p - p_2|^2 = |0.003596011|^2 = 0.000012931$$

and it is easy to see that

$$|p - p_3| = 0.000008589 \approx 0.000008621 = \frac{2}{3}|p - p_2|^2.$$

Table 1.6 Newton' Method Converges Linearly at a Double Root

k	p_k	$p_{k+1} - p_k$	$E_k = p - p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert}$
0	1.200000000	−0.096969697	−0.200000000	0.515151515
1	1.103030303	−0.050673883	−0.103030303	0.508165253
2	1.052356420	−0.025955609	−0.052356420	0.496751115
3	1.026400811	−0.013143081	−0.026400811	0.509753688
4	1.013257730	−0.006614311	−0.013257730	0.501097775
5	1.006643419	−0.003318055	−0.006643419	0.500550093
⋮	⋮	⋮	⋮	⋮

Example 1.15 (Linear Convergence at a Double Root). Start with $p_0 = 1.2$ and use Newton-Raphson iteration to find the double root $p = 1$ of the polynomial $f(x) = x^3 - 3x + 2$.

Using formula (1.56) to check for linear convergence, we get the values in Table 1.6.

Notice that the Newton-Raphson method is converging to the double root, but at a slow rate. The values of $f(p_k)$ in Example 1.15 go to zero faster than the values of $f'(p_k)$, so the quotient $f(p_k)/f'(p_k)$ in formula (1.40) is defined when $p_k \neq p$. The sequence is converging linearly, and the error is decreasing by a factor of approximately 1/2 with each successive iteration. The following theorem summarizes the performance of Newton's method on simple and double roots.

Theorem 1.6 (Convergence Rate for Newton-Raphson Iteration). Assume that Newton-Raphson iteration produces a sequence $\{p_n\}_{n=0}^{\infty}$ that converges to the root p of the function $f(x)$. If p is a simple root, convergence is quadratic and

$$|E_{n+1}| \approx \frac{|f''(p)|}{2|f'(p)|}|E_n|^2 \quad \text{for } n \text{ sufficiently large.} \tag{1.59}$$

If p is a multiple root of order M, convergence is linear and

$$|E_{n+1}| \approx \frac{M-1}{M}|E_n| \quad \text{for } n \text{ sufficiently large.} \tag{1.60}$$

1.4.4 Pitfalls

The division-by-zero error was easy to anticipate, but there are other difficulties that are not so easy to spot. Suppose that the function is $f(x) = x^2 - 4x + 5$; then the sequence $\{p_k\}$ of real numbers generated by formula (1.37) will wander back and forth from left to right and not converge. A simple analysis of the situation reveals that $f(x) > 0$ and has no real roots.

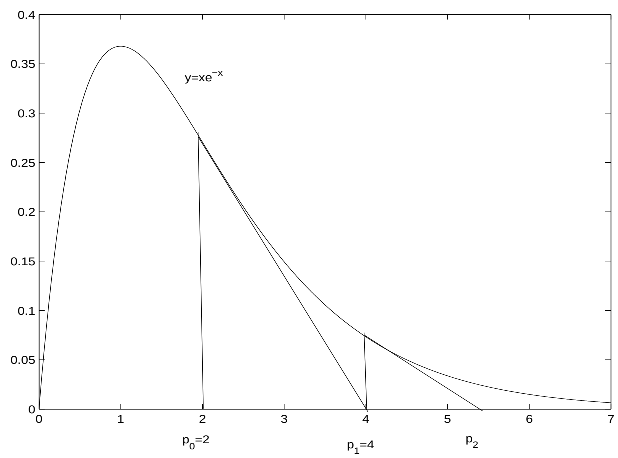


Figure 1.15 (a)Newton-Raphson iteration for $f(x) = xe^x$ can produce a divergent sequence.

Sometimes the initial approximation p_0 is too far away from the desired root and the sequence $\{p_k\}$ converges to some other root. This usually happens when the slope $f'(p_0)$ is small and the tangent line to the curve $y = f(x)$ is nearly horizontal. For example, if $f(x) = \cos(x)$ and we seek the root $p = \pi/2$ and start with $p_0 = 3$, calculation reveals that $p_1 = -4.01525255, p_2 = -4.85265757, \cdots$, and $\{p_k\}$ will converge to a different root $-3\pi/2 \approx -4.71238898$.

Suppose that $f(x)$ is positive and monotone decreasing on the unbounded interval $[a, \infty]$ and $p_0 > a$; then the sequence $\{p_k\}$ might diverge to $+\infty$. For example, if $f(x) = xe^{-x}$ and $p_0 = 2.0$, then

$$p_1 = 4.0, \quad p_2 = 5.333333333, \quad \ldots, \quad p_{15} = 19.723549434, \ldots,$$

and $\{p_k\}$ diverges slowly to $+\infty$ (see Figure 1.15(a)). This particular function has another surprising problem. The value of $f(x)$ goes to zero rapidly as x gets large, for example, $f(p_{15}) = 0.0000000536$, and it is possible that p_{15} could be mistaken for a root. For this reason we designed stopping criterion in Program 1.5 to involve the relative error $2|p_{k+1} - p_k|/(|p_k| + 10^{-6})$, and when $k = 15$, this value is 0.106817, so the tolerance $\delta = 10^{-6}$ will help guard against reporting a false root.

Another phenomenon, ***cycling***, occurs when the terms in the sequence $\{p_k\}$ tend to repeat ro almost repeat. For example, if $f(x) = x^3 - x - 3$ and initial approximation is $p_0 = 0$, then the sequence is

$$p_1 = -3.000000, \quad p_2 = -1.961538, \quad p_3 = -1.147176, \quad p_4 = -0.006579,$$

$$p_5 = -3.000389, \quad p_6 = -1.961818, \quad p_7 = -1.1474430, \quad \ldots$$

and we are stuck in a cycle where $p_{k+4} \approx p_k$ for $k = 0, 1, \ldots$ (see Figure 1.15(b)). But if the starting value p_0 is sufficiently close to the root $p \approx 1.671699881$, then $\{p_k\}$

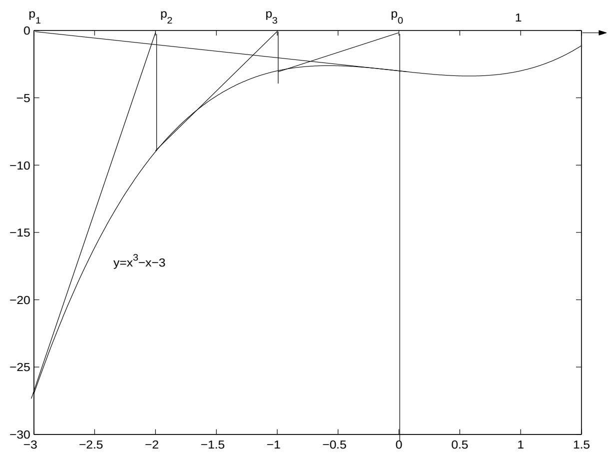


Figure 1.15 (b)Newton-Raphson iteration for $f(x) = x^3 - x - 3$ can produce a cyclic sequence.

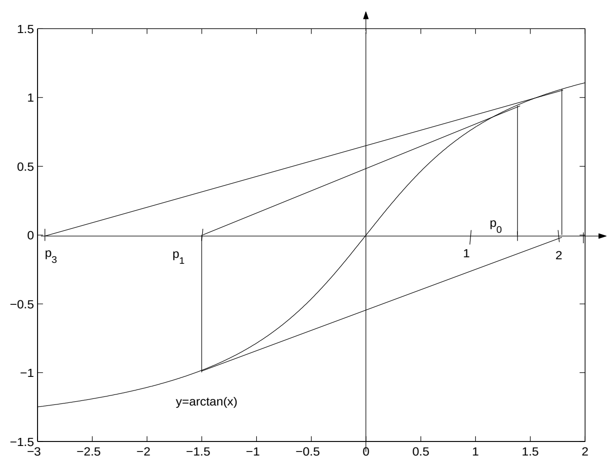


Figure 1.15 (c)Newton-Raphson iteration for $f(x) = \arctan(x)$ can produce a divergent sequence.

converges. If $p_0 = 2$, the sequence converges: $p_1 = 1.72727272, p_2 = 1.67369173, p_3 = 1.671702570$, and $p_4 = 1.671699881$.

When $|g'(x)| \geq 1$ on an interval containing the root p, there is a chance of divergent oscillation. For example, let $f(x) = \arctan(x)$; then the Newton-Raphson iteration function is $g(x) = x - (1 + x^2)\arctan(x)$, and $g'(x) = -2x\arctan(x)$. If the starting value $p_0 = 1.45$ is chosen, then

$$p_1 = -1.550263297, \quad p_2 = 1.845931751, \quad p_3 = -2.889109054,$$

etc, (see Figure 1.15(c)). But if the starting value is sufficiently close to the root $p = 0$,

Figure 1.16 The geometric construction of p_2 for the secant method.

a convergent sequence results. If $p_0 = 0.5$, then

$$p_1 = -0.079559511, \quad p_2 = 0.000335302, \quad p_3 = 0.000000000.$$

The situations above point to the fact that we must be honest in reporting an answer. Sometimes the sequence does not converge. It is not always the case that after N iterations a solution is found. The user of a root-finding algorithm needs to be warned of the situation when a root is not found. If there is other information concerning the context of the problem, then it is less likely that an erroneous root will be found. Sometimes $f(x)$ has a definite interval in which a root is meaningful. If knowledge of the behavior of the function or an "accurate" graph is available, then it is easier to choose p_0.

1.4.5 The Secant Method

The Newton-Raphson algorithm requires the evaluation of two functions per iteration, $f(p_{k-1})$ and $f'(p_{k-1})$. Traditionally, the calculation of derivatives of elementary functions could involve considerable effort. But, with modern computer algebra software packages, this has become less of an issue. Still many functions have nonelementary forms (integrals, sums, etc.), and it is desirable to have a method that converges almost as fast as Newton's method yet involves only evaluation of $f(x)$ and not of $f'(x)$. The secant method will require only one evaluation of $f(x)$ per step and at a simple root has an order of convergence $R \approx 1.618033989$. It is almost as fast as Newton's method, which has order 2.

The formula involved in the secant method is the same one that was used in the regula falsi method, except that the logical decisions regarding how to define each succeeding term are different. Two initial points $(p_0, f(p_0))$ and $(p_1, f(p_1))$ near the point $(p, 0)$ are needed, as shown in Figure 1.16. Define p_2 to be the abscissa

Table 1.7 Convergence of the Secant Method at a Simple Root

k	p_k	$p_{k+1} - p_k$	$E_k = p - p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert^{1.618}}$
0	-2.600000000	0.200000000	0.600000000	0.914152831
1	-2.400000000	0.293401015	0.400000000	0.469497765
2	-2.106598985	0.083957573	0.106598985	0.847290012
3	-2.022641412	0.021130314	0.022641412	0.693608922
4	-2.001511098	0.001488561	0.001511098	0.825841116
5	-2.000022537	0.000022515	0.000022537	0.727100987
6	-2.000000022	0.000000022	0.000000022	
7	-2.000000000	0.000000000	0.000000000	

of the point of intersection of the line through these tow points and the x-axis; then Figure 1.16 shows that p_2 will be closer to p than to either p_0 or p_1. The equation relating p_2, p_1, and p_0 is found by considering the slope

$$m = \frac{f(p_1) - f(p_0)}{p_1 - p_0} \quad \text{and} \quad m = \frac{0 - f(p_1)}{p_2 - p_1}. \tag{1.61}$$

The values of m in (1.61) are the slope of the secant line through the first two approximations and the slope of the line through $(p_1, f(p_1))$ and $(p_2, 0)$, respectively. Set the right-hand sides equal in (1.61) and solve for $p_2 = g(p_1, p_0)$ and get

$$p_2 = g(p_1, p_0) = p_1 - \frac{f(p_1)(p_1 - p_0)}{f(p_1) - f(p_0)}. \tag{1.62}$$

The general term is given by the two-point iteration formula

$$p_{k+1} = g(p_k, p_{k-1}) = p_k - \frac{f(p_k)(p_k - p_{k_1})}{f(p_k) - f(p_{k-1})}. \tag{1.63}$$

Example 1.16 (Secant Method at a Simple Root). Start with $p_0 = -2.6$ and $p_1 = -2.4$ and use the secant method to find the root $p = -2$ of polynomial function $f(x) = x^3 - 3x + 2$.

In this case the iteration formula (1.63) is

$$p_{k+1} = g(p_k, p_{k-1}) = p_k - \frac{(p_k^3 - 3p_k + 2)(p_k - p_{k-1})}{p_k^3 - p_{k-1}^3 - 3p_k + 3p_{k-1}}. \tag{1.64}$$

This can be algebraically manipulated to obtain

$$p_{k+1} = g(p_k, p_{k-1}) = \frac{p_k^2 p_{k-1} + p_k p_{k-1}^2 - 2}{p_k^2 + p_k p_{k-1} + p_{k-1}^2 - 3}. \tag{1.65}$$

The sequence of iterates is given in Table 1.7.

There is a relationship between the secant method and Newton's method. For a polynomial function $f(x)$, the secant method two-point formula $p_{k+1} = g(p_k, p_{k-1})$ will reduce to Newton's one-point formula $p_{k+1} = g(p_k)$ if p_k is replaced by p_{k-1}. Indeed, if we replace p_k by p_{k-1} in (1.65) , then the right side becomes the same as the right side of (1.58) in Example 1.14.

Proofs about the fate of convergence of the secant method can be found in advanced texts on numerical analysis. Let us state that the error terms satisfy the relationship

$$|E_{k+1}| \approx |E_k|^{1.618} \left| \frac{f''(p)}{2f'(p)} \right|^{0.618}, \tag{1.66}$$

where the order of convergence is $R = (1 + \sqrt{5})/2 \approx 1.618$ and the relation in (1.66) is valid only at simple roots.

To check this, we make use of Example 1.16 and the specific values

$$|p - p_5| = 0.000022537$$

$$|p - p_4|^{1.618} = 0.001511098^{1.618} = 0.000027296.$$

and

$$A = |f''(-2)/2f'(-2)|^{0.618} = (2/3)^{0.618} = 0.778351205.$$

Combine these and it is easy to see that

$$|p - p_5| = 0.000022537 \approx 0.000021246 = A|p - p_4|^{1.618}.$$

1.4.6 Accelerated Convergence

We could hope that there are root-finding techniques that converge faster than linearly when p is a root of order M. Our final result shows that a modification can be made to Newton's method so that convergence becomes quadratic at a multiple root.

Theorem 1.7 (Acceleration of Newton-Raphson Iteration). Suppose that the Newton-Raphson algorithm produces a sequence that converges linearly to the root $x = p$ of order $M > 1$. Then the Newton-Raphson iteration formula

$$p_k = p_{k-1} - \frac{Mf(p_{k-1})}{f'(p_{k-1})} \tag{1.67}$$

will produce a sequence $\{p_k\}_{k=0}^{\infty}$ that converges quadratically to p.

Table 1.8 Acceleration of Convergence at a Double Root

k	p_k	$p_{k+1} - p_k$	$E_k = p - p_k$	$\frac{\lvert E_{k+1}\rvert}{\lvert E_k\rvert^2}$
0	1.200000000	−0.193939394	−0.200000000	0.151515150
1	1.006060606	−0.006054517	−0.006060606	0.165718578
2	1.000006087	−0.000006087	−0.000006087	
3	1.000000000	0.000000000	0.000000000	

Table 1.9 Comparison of the speed of Convergence

Method	Special considerations	Relation between Successive error terms
Bisection		$E_{k+1} \approx \frac{1}{2}\lvert E_k\rvert$
Regula falsi		$E_{k+1} \approx A\lvert E_k\rvert$
Secant method	Multiple root	$E_{k+1} \approx A\lvert E_k\rvert$
Newton-Raphson	Multiple root	$E_{k+1} \approx A\lvert E_k\rvert$
Secant method	Simple root	$E_{k+1} \approx A\lvert E_k\rvert^{1.618}$
Newton-Raphson	Simple root	$E_{k+1} \approx A\lvert E_k\rvert^2$
Accelerated Newton-Raphson	Multiple root	$E_{k+1} \approx A\lvert E_k\rvert^2$

Example 1.17 (Acceleration of Convergence at a Double Root). Start with $p_0 = 1.2$ and use accelerated Newton-Raphson iteration to find the double root $p = 1$ of $f(x) = x^3 - 3x + 2$.

Since $M = 2$, the acceleration formula (1.67) becomes

$$p_k = p_{k-1} - 2\frac{f(p_{k-1})}{f'(p_{k-1})} = \frac{p_{k-1}^3 + 3p_{k-1} - 4}{3p_{k-1}^2 - 3},$$

and we obtain the values in Table 1.8.

Table 1.9 compares the speed of convergence of the various root-finding methods that we have studied so far. The value of the constant A is different for each method.

Program 1.5 (Newton-Raphson Iteration). To approximate a root of $f(x) = 0$ given one initial approximation p_0 and using the iteration

$$p_k = p_{k-1} - \frac{f(p_{k-1})}{f'(p_{k-1})} \quad \text{for} \quad k = 1, 2, \ldots .$$

```
Function [p0,err,k,y]=Newton (f,df,p0,delta,epsilon,max1)
%Input  – f is the object function input as a string 'f'
%       – df is the derivative of f input as a string 'df'
%       – p0 is the initial approximation to a zero of f
%       – delta is the tolerance for p0
%       – epsilon is the tolerance for the function values y
%       – max1 is the maximum number of iterations
%Output – p0 is the Newton-Raphson approximation to the zero
%       – err is the error estimate for p0
%       – k is the number of iterations
%       – y is the function value f(p0)
for k=1:max1
    p1=p0-feval(f,p0)/feval(df,p0);
    err=abs(p1-p0);
    relerr=2*err/(abs(p1)+delta)
    p0=p1;
    y=feval(f,p0);
    if (err<delta)|(relerr<delta)|(abs(y)<epsilon),break,end
end
```

Program 1.6 (Secant Method). To approximate a root of $f(x) = 0$ given two initial approximations p_0 and p_1 and using the iteration

$$p_{k+1} = p_k - \frac{f(p_k)(p_k - p_{k-1})}{f(p_k) - f(p_{k-1})} \quad \text{for} \quad k = 1, 2, \ldots .$$

```
Function [p1,err,k,y]=secant(f,p0,delta,epsilon,max1)
%Input  – f is the object function input as a string 'f'
%       – p0 and p1 are the initial approximations to a zero
%       – delta is the tolerance for p1
%       – epsilon is the tolerance for the function values y
%       – max1 is the maximum number of iterations
%Output – p1 is the secant method approximation to the zero
%       – err is the error estimate for p1
%       – k is the number of iterations
%       – y is the function value f(p1)
for k=1:max1
```

```
    p2=p1-feval(f,p1)*(p1-p0)/(feval (f,p1)-feval(f,p0));
    err=abs(p2-p1);
    relerr=2*err/(abs(p2)+delta);
    y=feval(f,p1);
    if (err<delta) | (relerr<delta) | (abs(y)<epsilon), break,end
end
```

1.4.7 Exercises for Newton-Raphson and Secant Methods

For problems involving calculations, you can use either a calculator or computer.

1. Let $f(x) = x^2 - x + 2$

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) Start with $p_0 = -1.5$ and find p_1, p_2, and p_3.

2. Let $f(x) = x^2 - x - 3$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) Start with $p_0 = 1.6$ and find p_1, p_2, and p_3.

(c) Start with $p_0 = 0.0$ and find p_1, p_2, p_3, and p_4. What do you conjecture about this sequence?

3. Let $f(x) = (x - 2)^4$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) Start with $p_0 = 2.1$ and find p_1, p_2, p_3, and p_4.

(c) Is the sequence converging quadratically or linearly?

4. Let $f(x) = x^3 - 3x - 2$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) Start with $p_0 = 2.1$ and find p_1, p_2, p_3, and p_4.

(c) Is the sequence converging quadratically or linearly?

5. Consider the function $f(x) = \cos(x)$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) We want to find the root $p = 3\pi/2$. Can we use $p_0 = 3$? Why?

(c) We want to find the root $p = 3\pi/2$. Can we use $p_0 = 5$? Why?

6. Consider the function $f(x) = \arctan(x)$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) If $p_0 = 1.0$, the find p_1, p_2, p_3, and p_4. What is $\lim_{n\to\infty} p_k$?

(c) If $p_0 = 2.0$, then find p_1, p_2, p_3, and p_4. What is $\lim_{n\to\infty} p_k$?

7. Consider the function $f(x) = xe^{-x}$.

(a) Find the Newton-Raphson formula $p_k = g(p_{k-1})$.

(b) If $p_0 = 0.2$, then find p_1, p_2, p_3, and p_4. What is $\lim_{n\to\infty} p_k$?

(c) If $p_0 = 20$, then find p_1, p_2, p_3, and p_4. What is $\lim_{n\to\infty} p_k$?

(d) What is the value of $f(p_4)$ in part(c)?

In Exercises 8 through 10, use the secant method and formula (1.59) and compute the next two iterates p_2 and p_3.

8. Let $f(x) = x^2 - 2x - 1$. Start with $p_0 = 2.6$ and $p_1 = 2.5$.

9. Let $f(x) = x^2 - x - 3$. Start with $p_0 = 1.7$ and $p_1 = 1.67$.

10. Let $f(x) = x^3 - x + 2$. Start with $p_0 = 1.5$ and $p_1 = 1.52$.

11. *Cube-root algorithm.* Start with $f(x) = x^3 - A$, where A is any real number, and derive the recursive formula

$$p_k = \frac{2p_{k-1} + A/p_{k-1}^2}{3} \quad \text{for} \quad k = 1, 2, \ldots.$$

12. Consider $f(x) = x^N - A$, where N is a positive integer.

(a) What real values are the solution to $f(x) = 0$ for the various choices of N and A that and can arise?

(b) Derive the recursive formula for finding the Nth root of A.

13. Can Newton-Raphson iteration be used to solve $f(x) = 0$ if $f(x) = x^2 - 14x + 50$? Why?

14. Can Newton-Raphson be used to solve $f(x) = 0$ if $f(x) = x^{1/3}$? Why?

15. Can Newton-Raphson be used to solve $f(x) = 0$ if $f(x) = (x-3)^{1/2}$ and the starting value is $p_0 = 4$? Why?

16. Establish the limit of the sequence in (11).

17. Prove that the sequence $\{p_k\}$ in equation (4) of Theorem 1.5 converges to p. Use the following steps.

(a) Show that if p is a fixed point of $g(x)$ in equation (5) then p is a zero of $f(x)$.

(b) If p is a zero of $f(x)$ and $f'(p) \neq 0$, show that $g'(x) = 0$. Use part (b) and Theorem 1.3 to show that the sequence $\{p_k\}$ in equation (4) converges to p.

18. Prove equation (1.55) of Theorem 1.6. Use the following steps. By Theorem 0.11, we can expand $f(x)$ about $x = p_k$ to get

$$f(x) = f(p_k) + f'(p_k)(x - p_k) + \frac{1}{2}f''(c_k)(x - p_k)^2.$$

Since p is zero of $f(x)$, we set $x = p$ and obtain

$$f(p) = f(p_k) + f'(p_k)(p - p_k) + \frac{1}{2}f''(c_k)(p - p_k)^2.$$

(a) Now assume that $f'(x) \neq 0$ for all x near the root p. Use the facts given above and $f'(p_k) \neq 0$ to show that

$$p - p_k + \frac{f(p_k)}{f'(p_k)} = \frac{-f''(c_k)}{2f'(p_k)}(p - p_k)^2.$$

(b) Assume that $f'(x)$ and $f''(x)$ do not change too rapidly so that we can use the approximations $f''(p_k) \approx f'(p)$ and $f''(c_k) \approx f''(p)$. Now use part (a) to get

$$E_{k+1} \approx \frac{-f''(p)}{2f'(p)}E_k^2.$$

19. Suppose that A is a positive real number.

(a) Show that A has the representation $A = q \times 2^{2m}$, where $1/4 \leq q < 1$ and m is an integer.

(b) Use part (a) to show that the square root is $A^{1/2} = q^{1/2} \times 2^m$. *Remark.* Let $p_0 = (2p + 1)/3$, where $1/4 \leq q \leq 1$, and use Newton's formula (1.47). After three iterations. p_3 will be an approximation to $q^{1/2}$ with a precision of 24 binary digits. This is the algorithm that is often used in the computer's hardware to compute square roots.

20. (a) Show that formula (1.63) for the secant method is algebraically equivalent to

$$p_{k+1} = \frac{p_{k+1}f(p_k) - p_k f(p_{k-1})}{f(p_k) - f(p_{k-1})}.$$

(b) Explain why loss of significance in subtraction makes this formula inferior for computational purposes to the one given in formula (1.63).

21. Suppose that p is a root of order $M = 2$ for $f(x) = 0$. Prove that the accelerated Newton-Raphson iteration

$$p_k = p_{k-1} - \frac{2f(p_{k-1})}{f'(p_{k-1})}$$

converges quadratically (see Exercise 18).

22. *Halley's method* is another way to speed up convergence of Newton's method. The Halley iteration formula is

$$g(x) = x - \frac{f(x)}{f'(x)}\left(1 - \frac{f(x)f''(x)}{2(f'(x))^2}\right)^{-1}.$$

The term in brackets is the modification of the Newton-Raphson formula. Halley's method will yield cubic convergence ($R = 3$) at simple zero of $f(x)$.

(a) Start with $f(x) = x^2 - A$ and find Halley's iteration formula $g(x)$ for finding

$\sqrt{A}$. Use $p_0 = 2$ to approximate $\sqrt{5}$ and compute p_1, p_2, and p_3.

(b) Start with $f(x) = x^3 - 3x + 2$ and find Halley' iteration formula $g(x)$. Use $p_0 = -2.4$ and compute p_1, p_2, and p_3.

23. *A modified Newton-Raphson method for multiple roots.* If p is a root of multiplicity M, then $f(x) = (x-p)^M q(x)$, where $q(p) \neq 0$.

(a) Show that $h(x) = f(x)/f'(x)$ has a simple root at p.

(b) Show that when the Newton-Raphson method is applied to finding the simple root p of $h(x)$ we get $g(x) = x - h(x)/h'(x)$, which becomes

$$g(x) = x - \frac{f(x)f'(x)}{(f'(x))^2 - f(x)f''(x)}.$$

(c) The iteration using $g(x)$ in part (b) converges quadratic ally to p. Explain why this happens.

(d) Zero is a root of multiplicity 3 for the function $f(x) = \sin(x^3)$. Start with $p_0 = 1$ and compute p_1, p_2, and using the modified Newton-Raphson method.

24. Suppose that an iterative method for solving $f(x) = 0$ produce the following four consecutive error terms (see Example 1.11): $E_0 = 0.400000, E_1 = 0.043797$, $E_2 = 0.000062$, and $E_3 = 0.000000$. Estimate the asymptotic error constant A and the order of convergence R of the sequence generated by the iterative method.

1.4.8 Algorithms and Programs

1. Modify Programs 1.5 and 1.6 to display an appropriate error message when (i) division by zero occurs in (4) or (27), respectively, or (ii) the maximum number of iterations, max1, exceeded.

2. It is often instructive to display the terms in the sequences generated by (1.40) and (1.63) (i.e., the second column of Table 1.4) Modify Programs 1.5 and 1.6 to display the sequences generated by (1.40) and (1.63), respectively.

3. Modify Program 1.5 to use Newton's square-root algorithm to approximate each of the following square roots to 10 decimal places.

(a) Start with $p_0 = 3$ and approximate $\sqrt{8}$.

(b) Start with $p_0 = 10$ and approximate $\sqrt{91}$.

(c) Start with $p_0 = -3$ and approximate $-\sqrt{8}$.

4. Modify Program 1.5 to use the cube-root algorithm in Exercise 11 to approximate each of the following cube roots to 10 decimal places.

(a) Start with $p_0 = 2$ and approximate $7^{1/3}$

(b) Start with $p_0 = 6$ and approximate $200^{1/3}$

(c) Start with $p_0 = -2$ and approximate $(-7)^{1/3}$.

5. Modify Program 1.5 to use the accelerated Newton-Raphson algorithm in Theorem 1.7 to find the root p of order M of each of the following functions.

(a) $f(x) = (x-2)^5, M = 5, p = 2$; start with $p_0 = 1$.

(b) $f(x) = \sin(x^3), M = 5, p = 0$; start with $p_0 = 1$.

(c) $f(x) = (x-1)\ln(x), M = 2, p = 1$; start with $p_0 = 2$.

6. Modify Program 1.5 to use Halley's method in Exercise 22 to find the simple zero of $f(x) = x^3 - 3x + 2$, using $p_0 = -2.4$.

7. Suppose that the equations of motion for a projectile are

$$y = f(t) = 9600(1 - e^{-t/15}) - 480t$$

$$x = r(t) = 2400(1 - e^{-t/15}.$$

(a) Find the elapsed time until impact accurate to 10 decimal places.

(b) Find the range accurate to 10 decimal places

8. **(a)** Find the point on the parabola $y = x^2$ that is closest to the point $(3, 1)$ accurate to 10 decimal places.

(b) Find the point on the graph of $y = \sin(x - \sin(x))$ that is closest to the point $(2.1, 0.5)$ accurate to 10 decimal places.

(c) Find the values of at which the minimum vertical distance between the graphs of $f(x) = x^2 + 2$ and $g(x) = (x/5) - \sin(x)$ occurs accurate to 10 decimal places.

9. An open-top box is constructed from a rectangular piece of sheet metal measuring 10 by 16 inches. Squares of what size (accurate to 0.000000001 inch) should be cut from the corners if the volume of the box is to be 100 cubic inches?

10. A catenary is the curve formed by a hanging cable. Assume that the lowest point is $(0, 0)$; then the formula for the catenary is $y = C\cosh(x/C) - C$. To determine the catenary that goes through $(\pm a, b)$ we must solve the equation $b = C\cosh(a/C) - C$ for C.

(a) Show that the catenary through $(\pm 10, 6)$ is $y = 9.1889\cosh(x/9.1889) - 9.1889$.

(b) Find the catenary that passes through $(\pm 12, 5)$.