Numerical Methods for Engineers

Steven C. Chapra
Raymond P. Canale
Fifth Edition

Chapter 5

by Martin Mendez,
UASLP

Roots of Equations

Part 2

- Why?

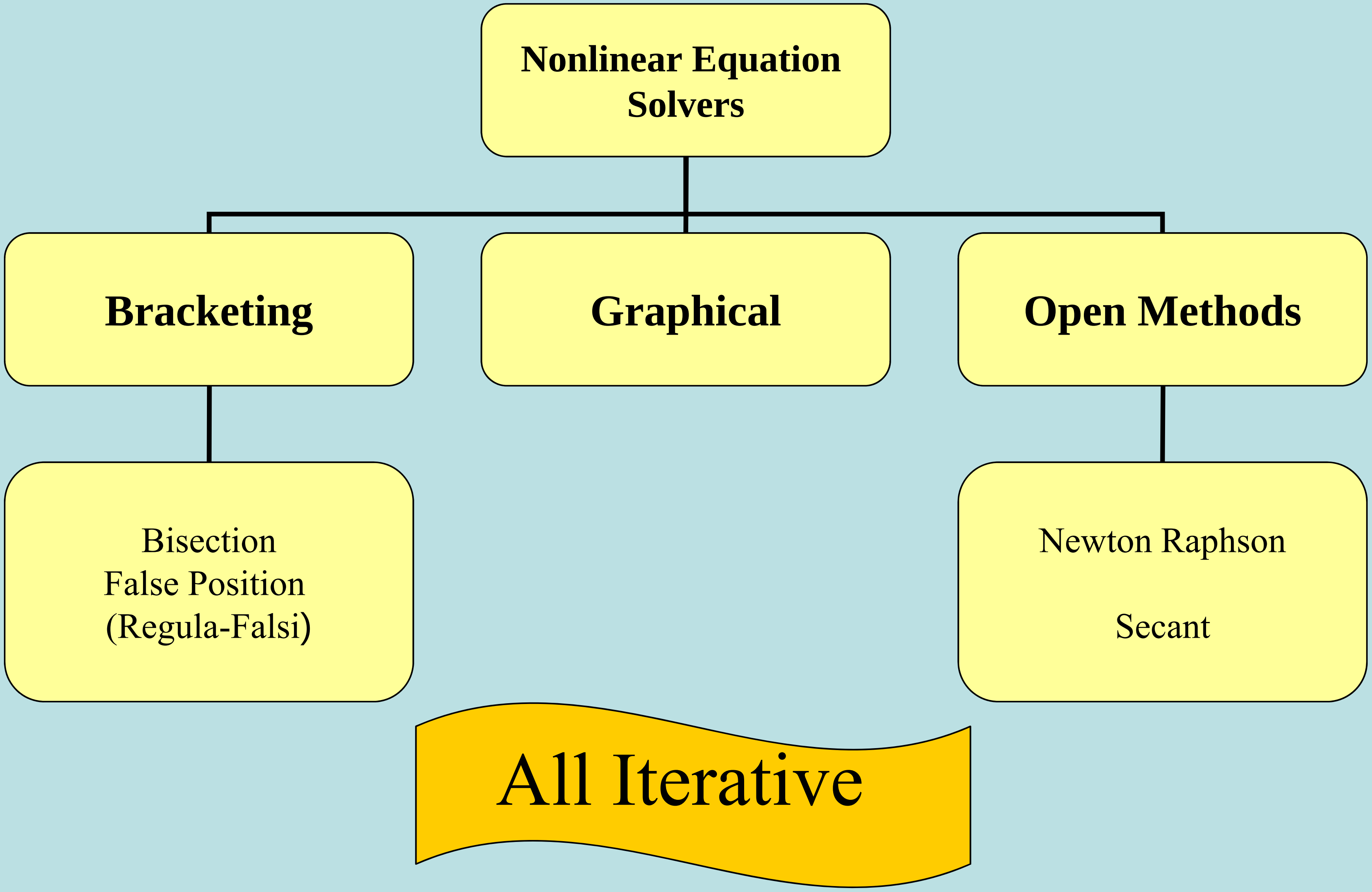
$$ax^2 + bx + c = 0 \quad \Rightarrow \quad x = \frac{-b \mp \sqrt{b^2 - 4ac}}{2a}$$

- But

$$ax^5 + bx^4 + cx^3 + dx^2 + ex + f = 0 \quad \Rightarrow x = ?$$

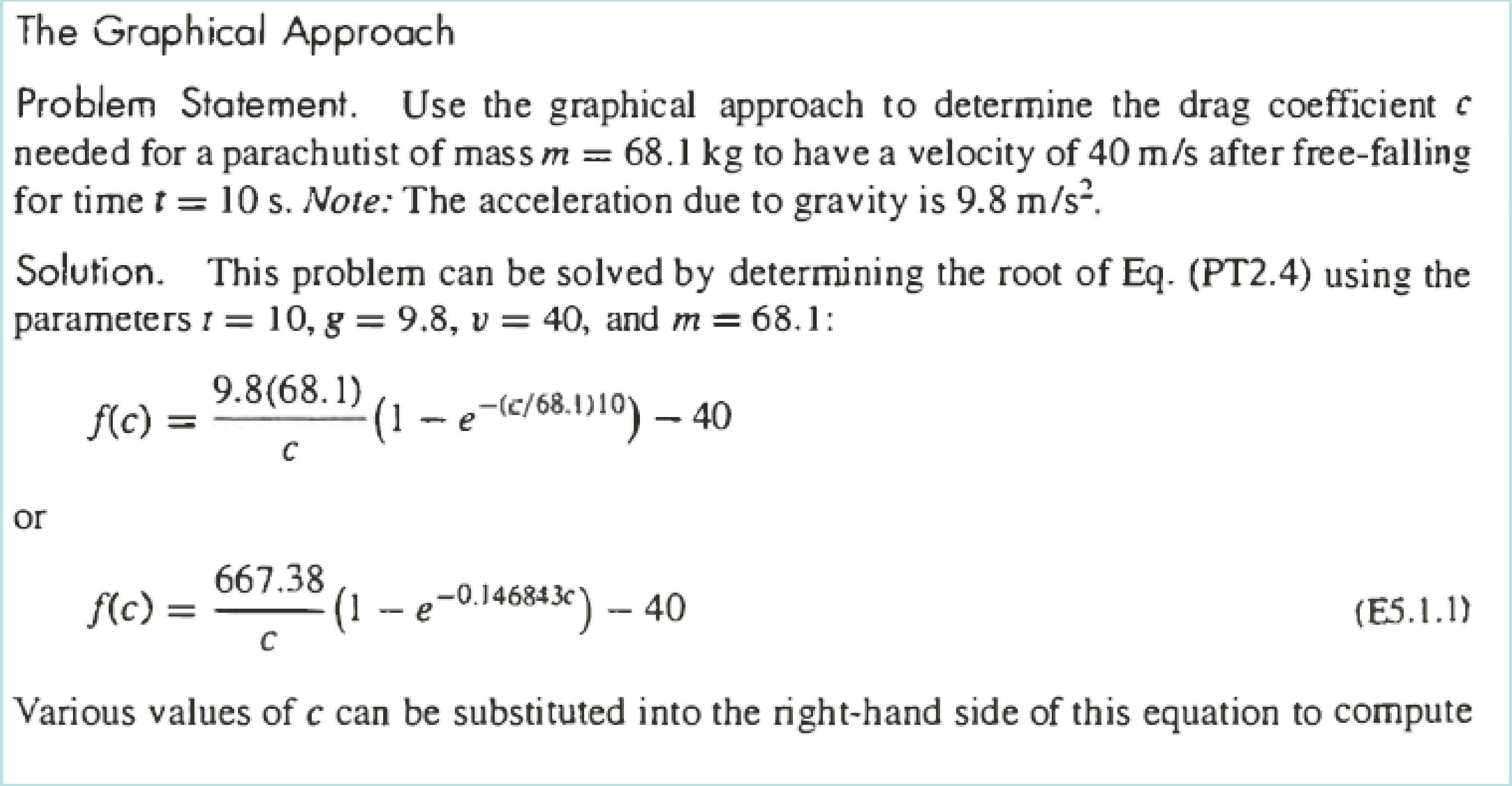
$$\sin x + x = 0 \quad \Rightarrow x = ?$$

by Martin Mendez,
UASLP



by Martin Mendez,
UASLP

Example 5.1



The Graphical Approach

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

Solution. This problem can be solved by determining the root of Eq. (PT2.4) using the parameters $t = 10$, $g = 9.8$, $v = 40$, and $m = 68.1$:

$$f(c) = \frac{9.8(68.1)}{c}\left(1 - e^{-(c/68.1)10}\right) - 40$$

or

$$f(c) = \frac{667.38}{c}\left(1 - e^{-0.146843c}\right) - 40 \tag{E5.1.1}$$

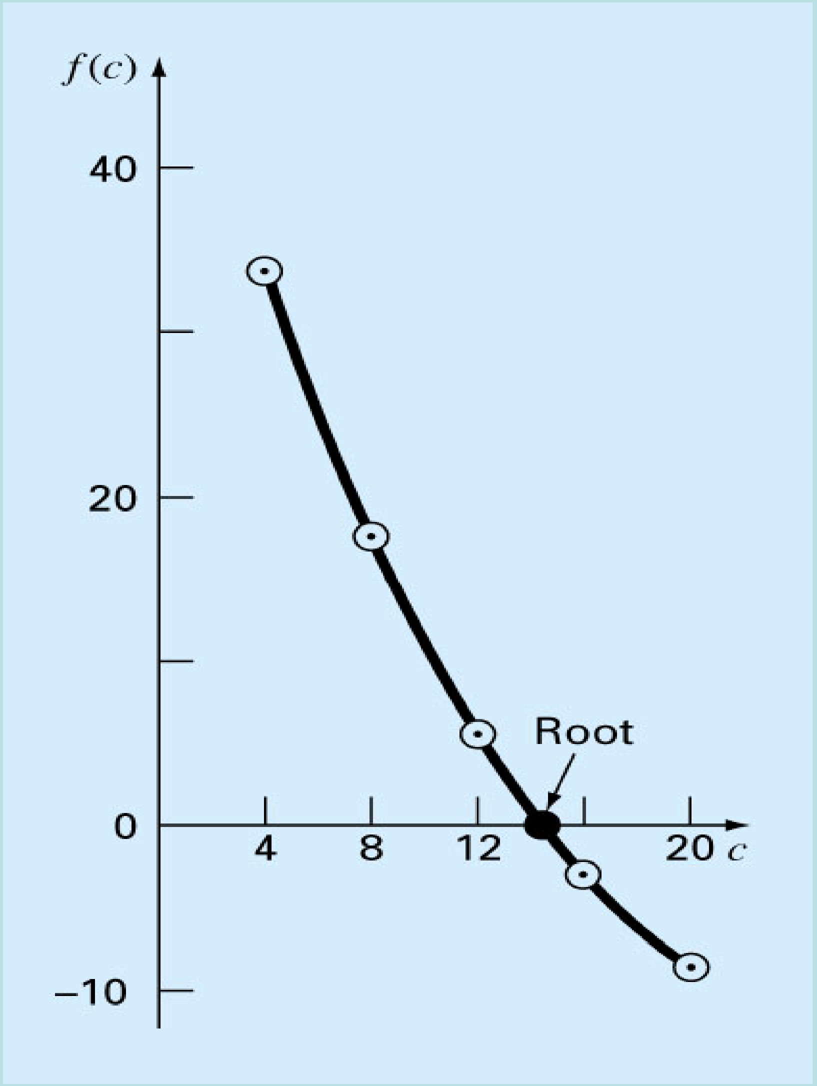
Various values of c can be substituted into the right-hand side of this equation to compute

Bracketing Methods

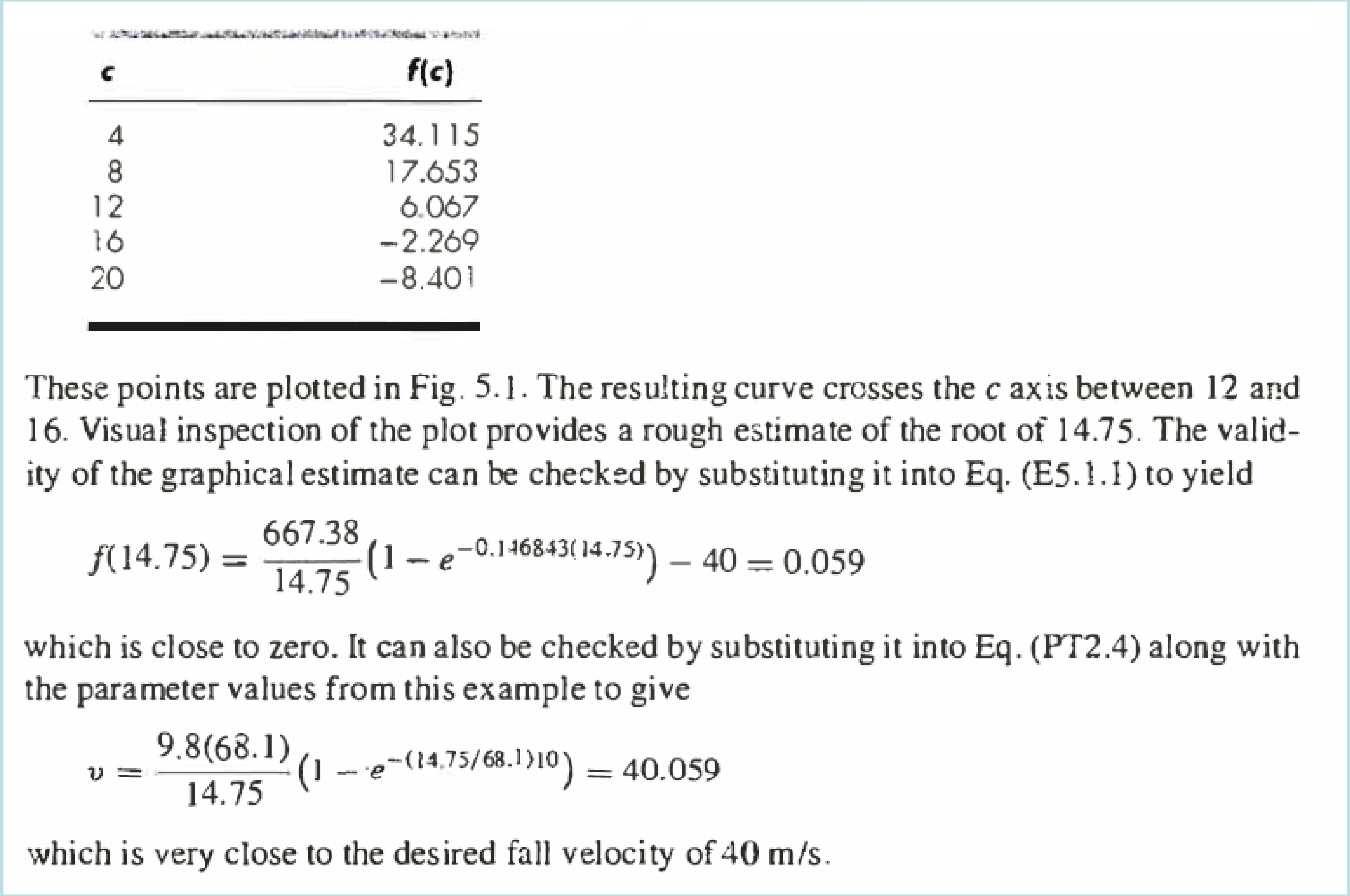
(Or, two point methods for finding roots)

- Two initial guesses for the root are required. These guesses must “bracket” or be on either side of the root.

- If one root of a real and continuous function, $f(x)=0$, is bounded by values $x=x_l$, $x=x_u$ then
  $f(x_l) \,.\, f(x_u) < 0$. (The function changes sign on opposite sides of the root)



by Martin Mendez, UASLP



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

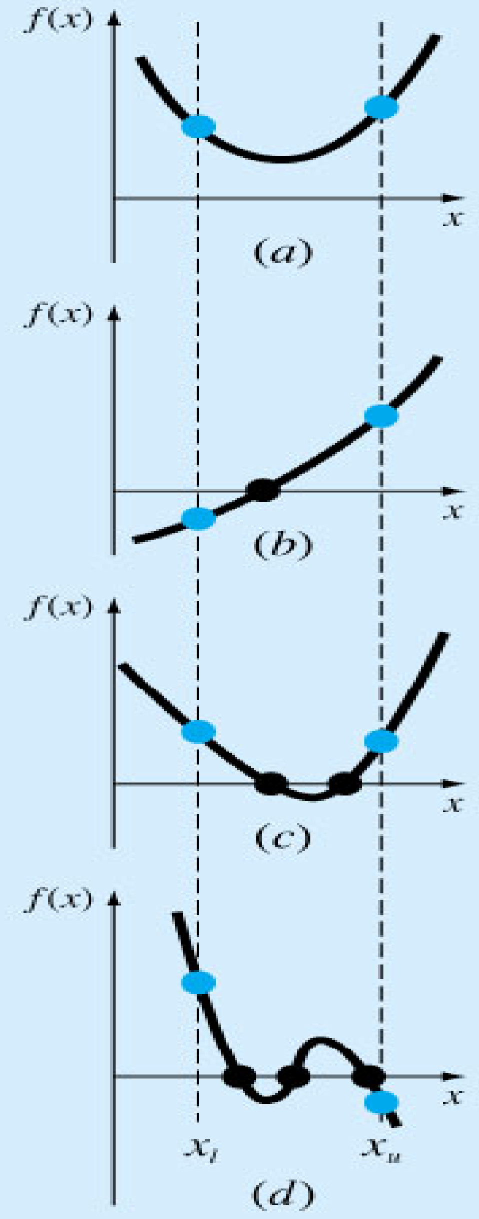
These points are plotted in Fig. 5.1. The resulting curve crosses the c axis between 12 and 16. Visual inspection of the plot provides a rough estimate of the root of 14.75. The validity of the graphical estimate can be checked by substituting it into Eq. (E5.1.1) to yield

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

which is close to zero. It can also be checked by substituting it into Eq. (PT2.4) along with the parameter values from this example to give

$$v = \frac{9.8(68.1)}{14.75}\left(1 - e^{-(14.75/68.1)10}\right) = 40.059$$

which is very close to the desired fall velocity of 40 m/s.



No answer (No root)

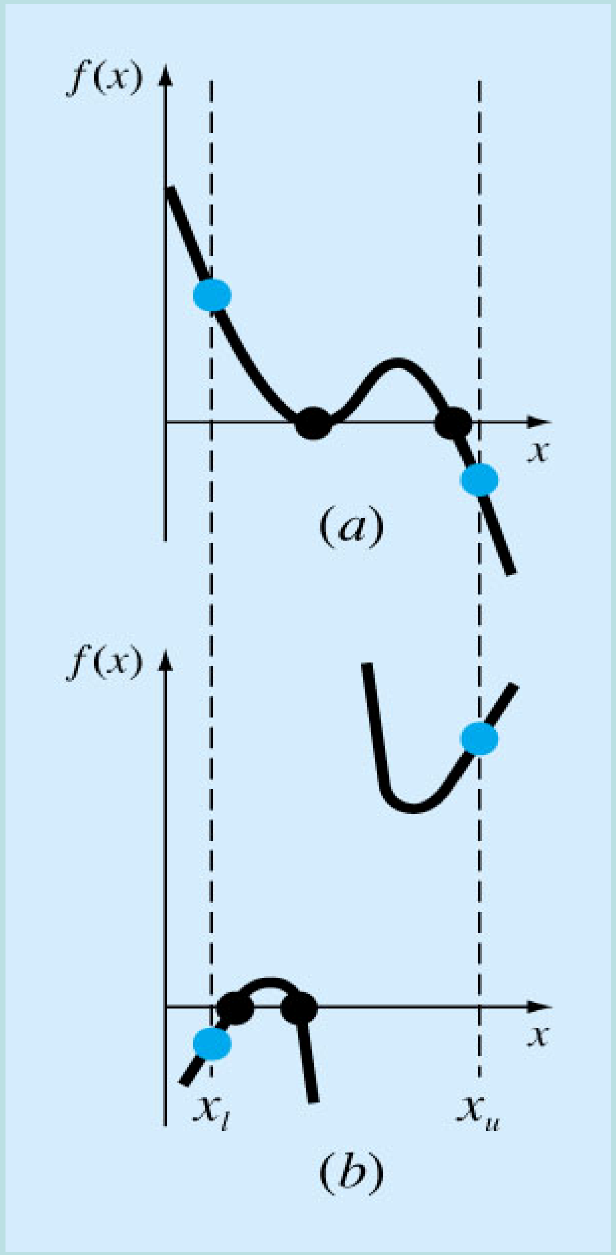
Nice case (one root)

Oops!! (two roots!!)

Three roots(Might work for a while!!)

by Martin Mendez, UASLP

Figure 5.3

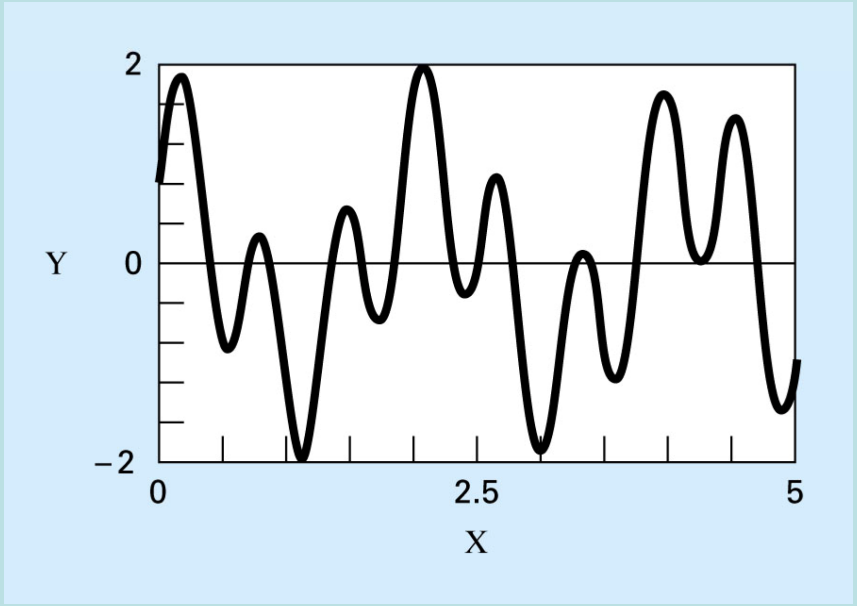


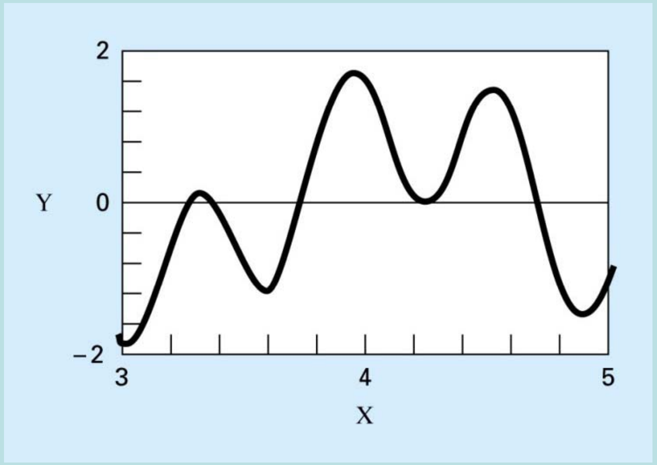
Two roots(Might work for a while!!)

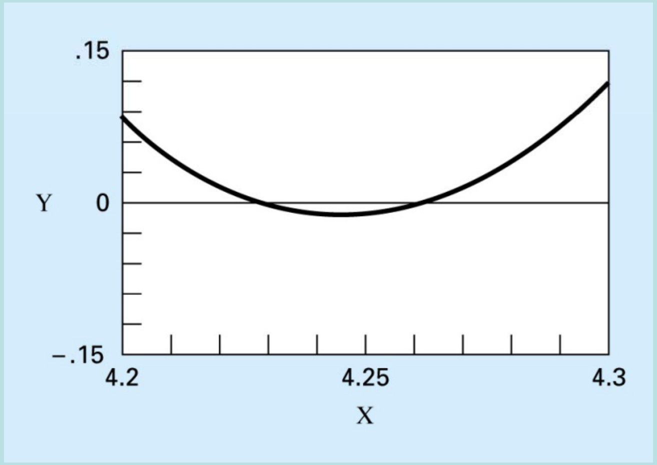
Discontinuous function. Need special method

MANY-MANY roots. What do we do?

$f(x)=\sin 10x+\cos 3x$







by Martin Mendez, UASLP

The Bisection Method

For the arbitrary equation of one variable, $f(x)=0$

1. Pick x_l and x_u such that they bound the root of interest, check if $f(x_l).f(x_u) <0$.

2. Estimate the root by evaluating $f[(x_l+x_u)/2]$.

3. Find the pair
   - If $f(x_l).\ f[(x_l+x_u)/2]<0$, root lies in the lower interval, then $x_u=(x_l+x_u)/2$ and go to step 2.

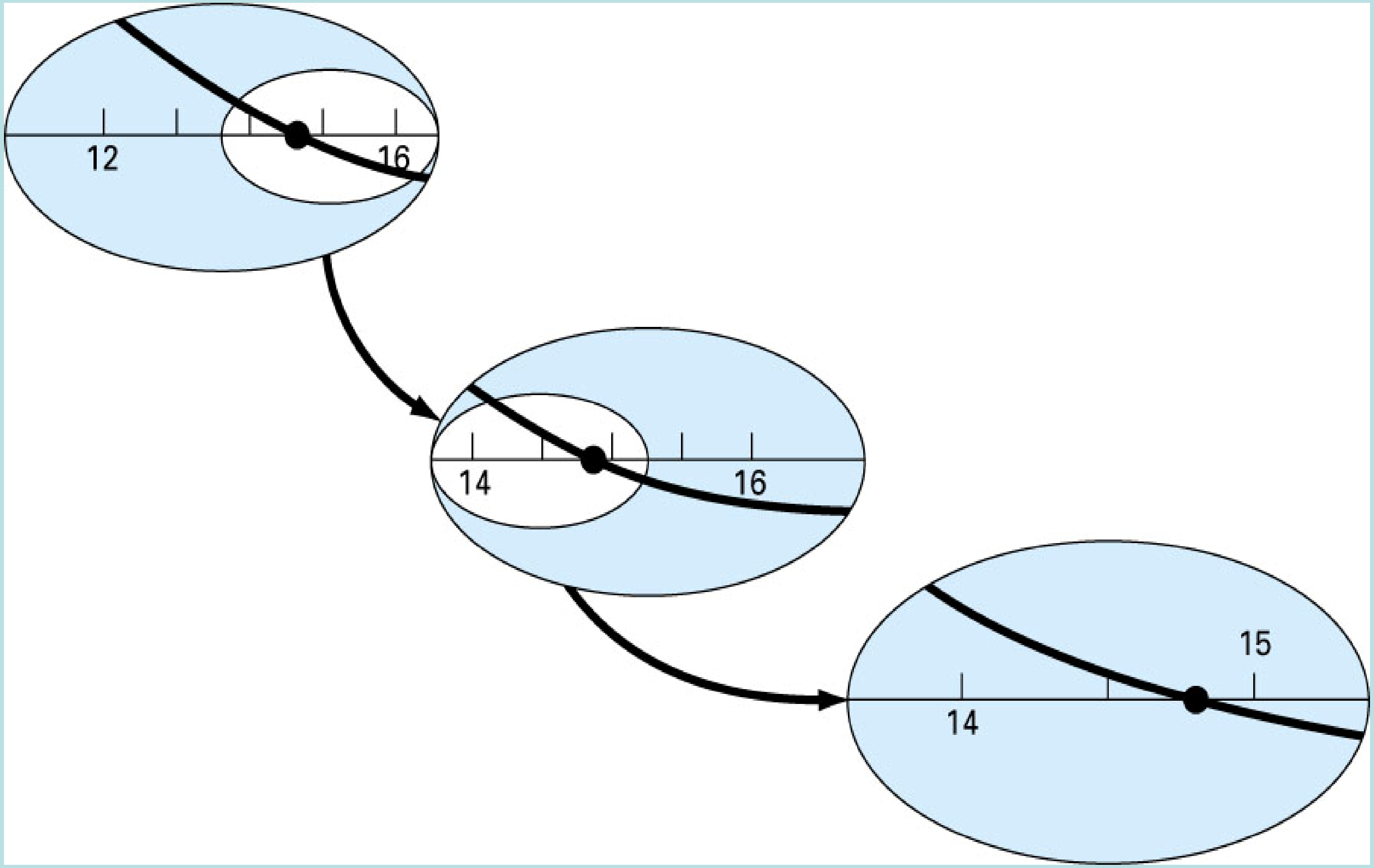
- If $f(x_l) . f[(x_l+x_u)/2]>0$, root lies in the upper interval, then $x_l= [(x_l+x_u)/2$, go to step 2.

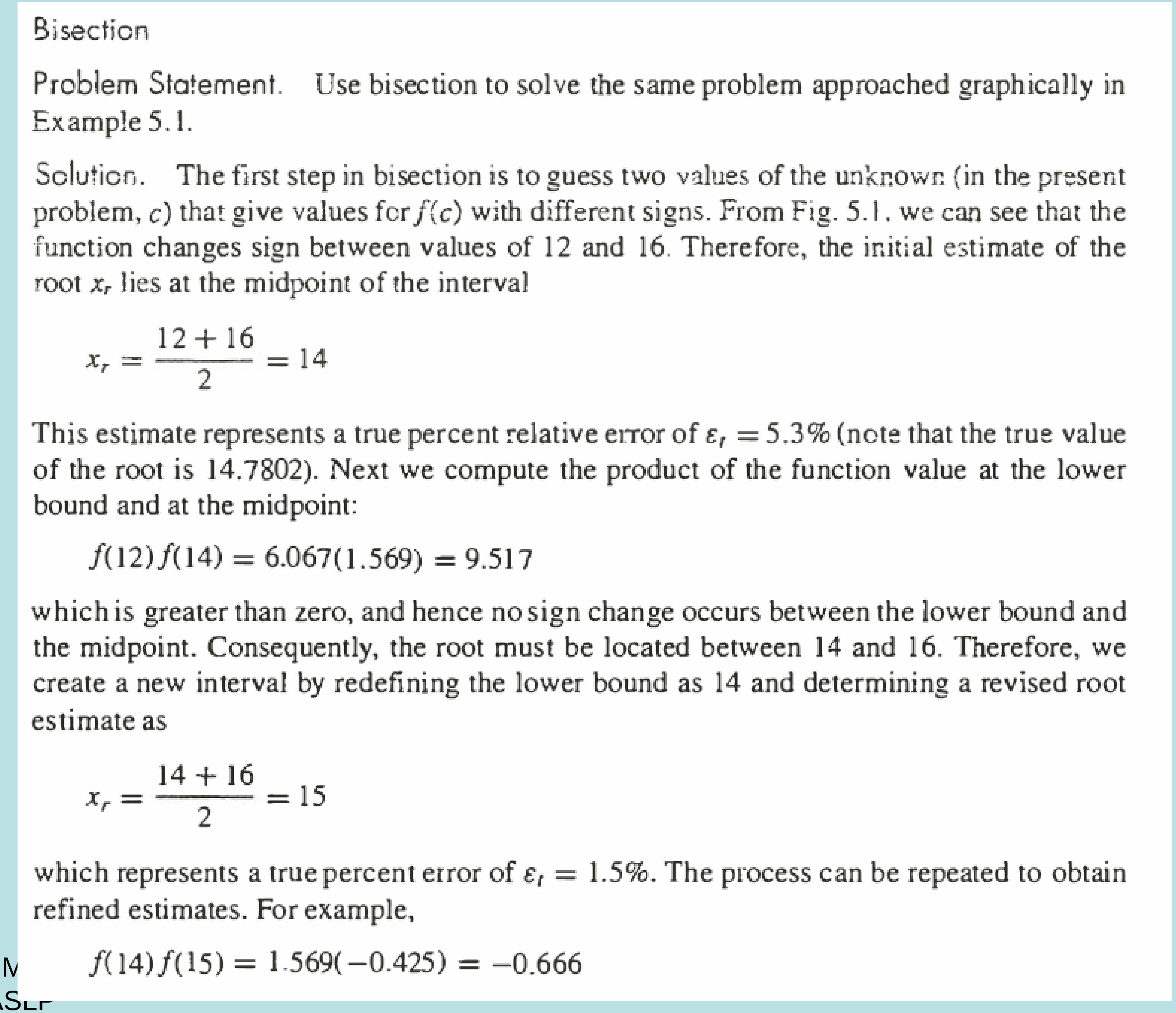
- If $f(x_l) . f[(x_l+x_u)/2]=0$, then root is $(x_l+x_u)/2$ and terminate.

4. Compare ε_s with ε_a

5. If $\varepsilon_a<\varepsilon_s$, stop. Otherwise repeat the process.

$$\left\{ \begin{array}{l} \dfrac{\left| x_l - \dfrac{x_l + x_u}{2} \right|}{\left| \dfrac{x_l + x_u}{2} \right|} \prec 100\% \\ \textit{or} \\ \dfrac{\left| x_u - \dfrac{x_l + x_u}{2} \right|}{\left| \dfrac{x_l + x_u}{2} \right|} \prec 100\% \end{array} \right\}$$


12
16
14
16
14
15



Bisection

Problem Statement. Use bisection to solve the same problem approached graphically in Example 5.1.

Solution. The first step in bisection is to guess two values of the unknown (in the present problem, c) that give values for $f(c)$ with different signs. From Fig. 5.1, we can see that the function changes sign between values of 12 and 16. Therefore, the initial estimate of the root x_r lies at the midpoint of the interval

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

This estimate represents a true percent relative error of $\varepsilon_t = 5.3\%$ (note that the true value of the root is 14.7802). Next we compute the product of the function value at the lower bound and at the midpoint:

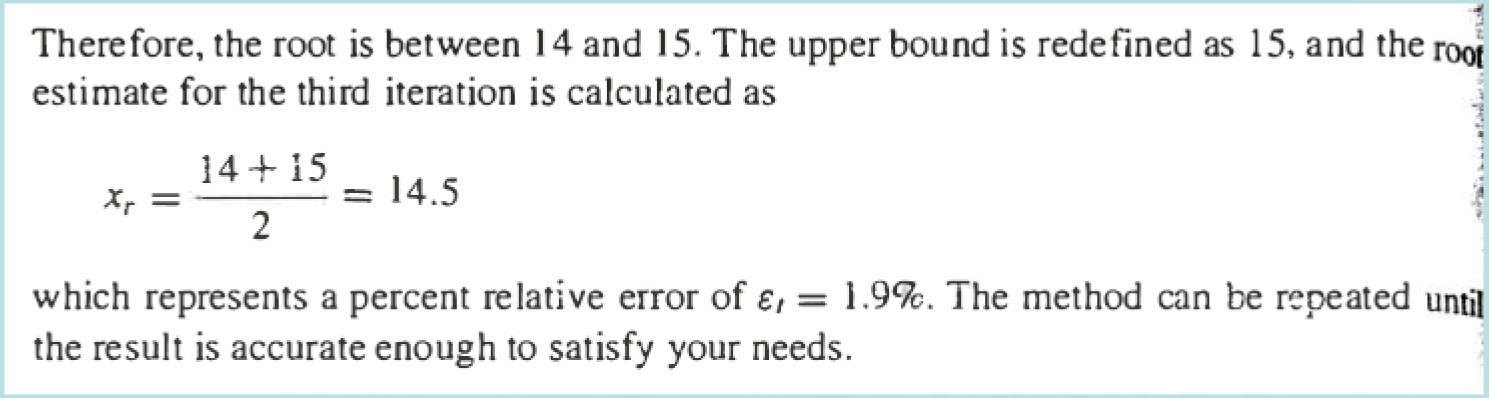
$$f(12)f(14) = 6.067(1.569) = 9.517$$

which is greater than zero, and hence no sign change occurs between the lower bound and the midpoint. Consequently, the root must be located between 14 and 16. Therefore, we create a new interval by redefining the lower bound as 14 and determining a revised root estimate as

$$x_r = \frac{14 + 16}{2} = 15$$

which represents a true percent error of $\varepsilon_t = 1.5\%$. The process can be repeated to obtain refined estimates. For example,

$$f(14)f(15) = 1.569(-0.425) = -0.666$$


Therefore, the root is between 14 and 15. The upper bound is redefined as 15, and the root estimate for the third iteration is calculated as

$$x_r = \frac{14 + 15}{2} = 14.5$$

which represents a percent relative error of $\varepsilon_t = 1.9\%$. The method can be repeated until the result is accurate enough to satisfy your needs.

Evaluation of Method

Pros

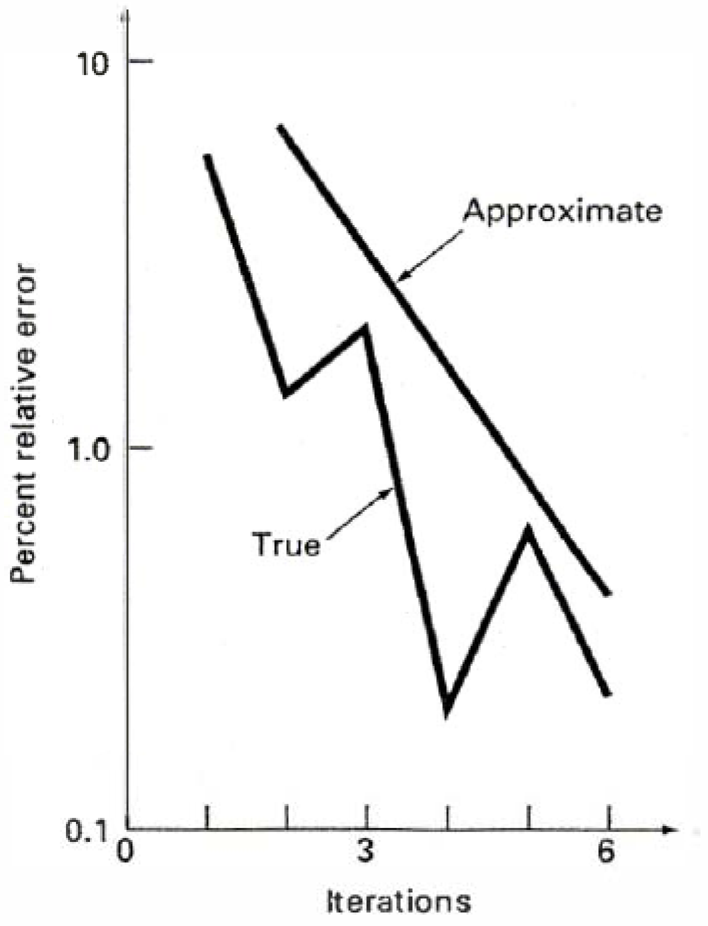
- Easy
- Always find root
- Number of iterations required to attain an absolute error can be computed a priori.

Cons

- Slow
- Know a and b that bound root
- Multiple roots
- No account is taken of $f(x_l)$ and $f(x_u)$, if $f(x_l)$ is closer to zero, it is likely that root is closer to x_l .

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

FIGURE 5.7
Errors for the bisection method. True and estimated errors are plotted versus the number of iterations.



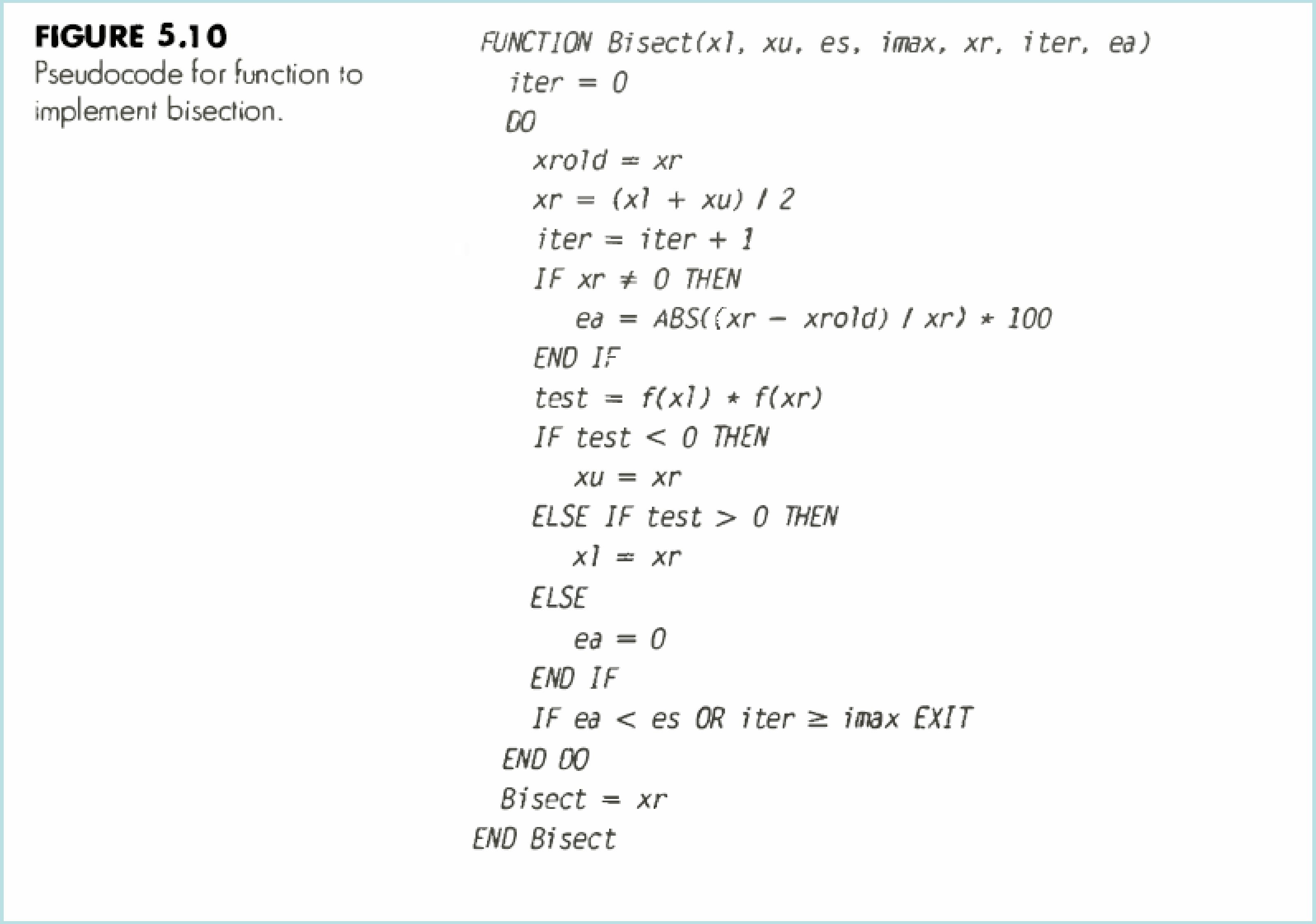


FIGURE 5.10
Pseudocode for function to implement bisection.

```
FUNCTION Bisect(xl, xu, es, imax, xr, iter, ea)
  iter = 0
  DO
    xrold = xr
    xr = (xl + xu) / 2
    iter = iter + 1
    IF xr ≠ 0 THEN
      ea = ABS((xr − xrold) / xr) * 100
    END IF
    test = f(xl) * f(xr)
    IF test < 0 THEN
      xu = xr
    ELSE IF test > 0 THEN
      xl = xr
    ELSE
      ea = 0
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  Bisect = xr
END Bisect
```

by Martin Mendez,
UASLP

How Many Iterations will It Take?

- Length of the first Interval $L_o = b - a$
- After 1 iteration $L_1 = L_o/2$
- After 2 iterations $L_2 = L_o/4$
- After k iterations $L_k = L_o/2^k$

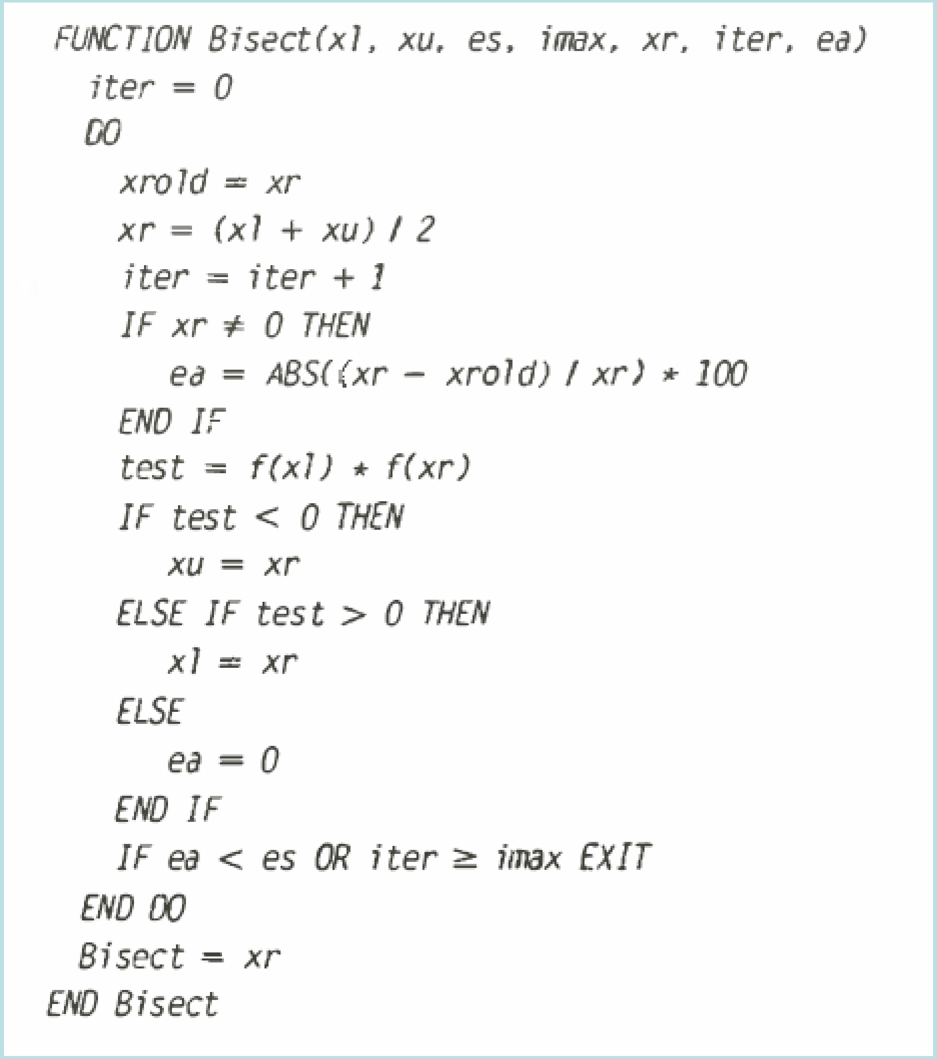
$$\varepsilon_a \le \frac{L_k}{x} \times 100\% \qquad \varepsilon_a \le \varepsilon_s$$

- If the absolute magnitude of the error is

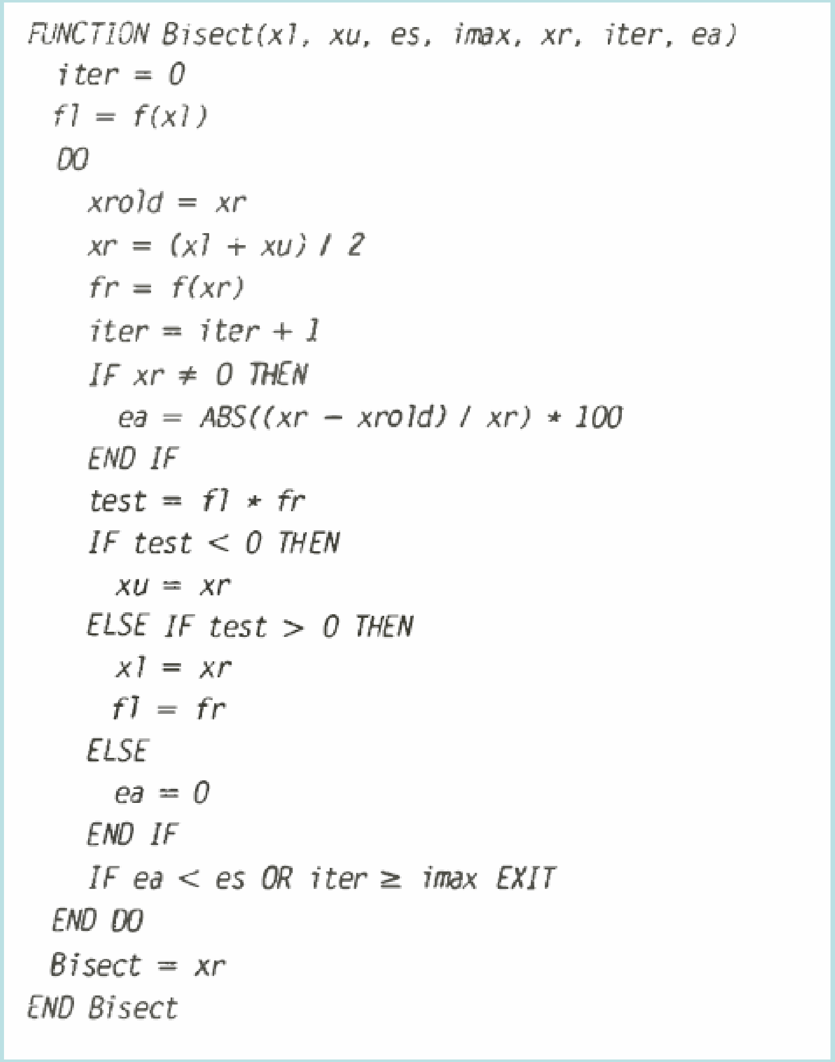
$$\frac{\varepsilon_s \cdot x}{100\%} = 10^{-4}$$

and L_o=2, how many iterations will you have to do to get the required accuracy in the solution?

$$10^{-4} = \frac{2}{2^k} \quad \Rightarrow 2^k = 2 \times 10^4 \quad \Rightarrow k \cong 14.3 = 15$$



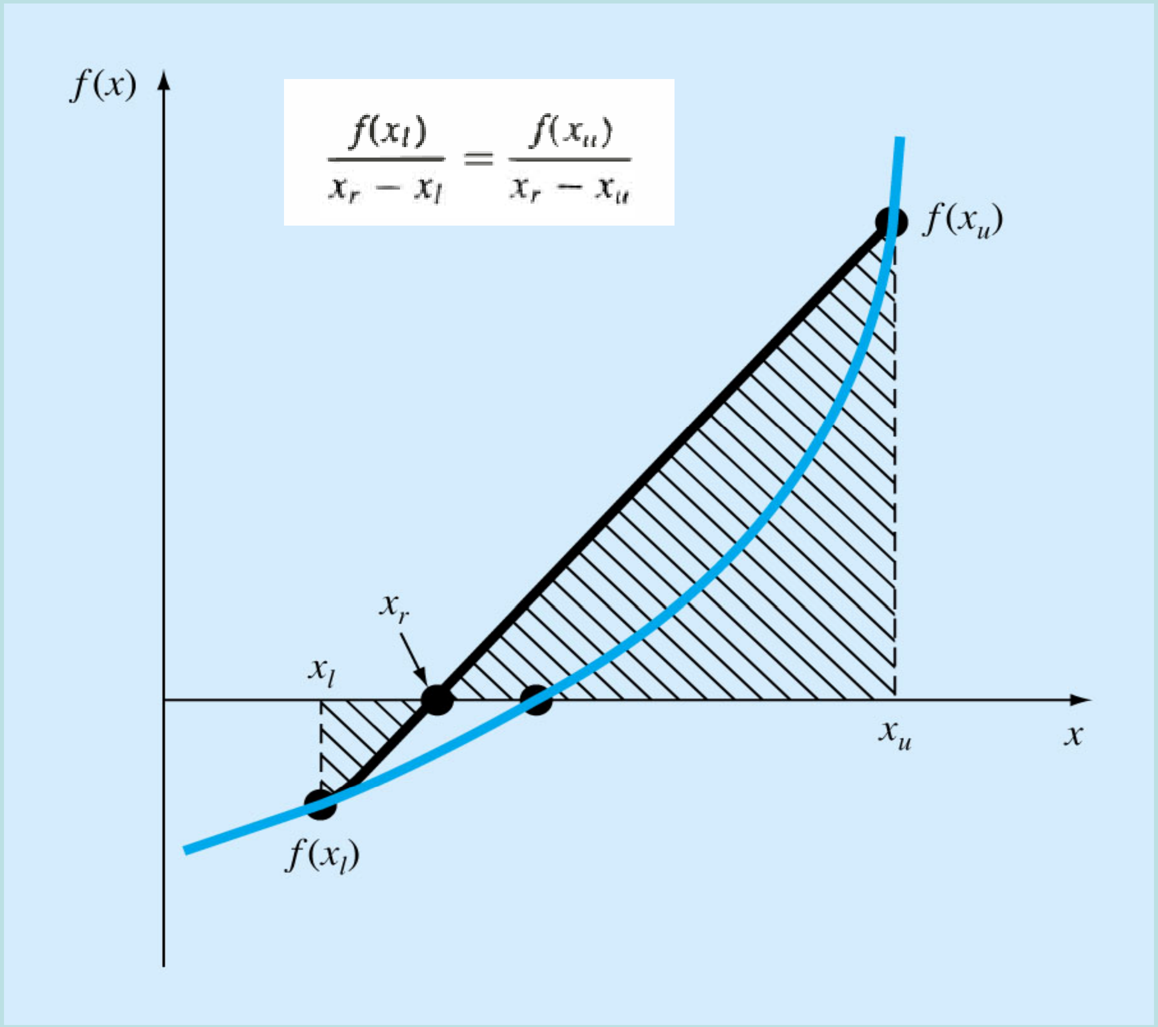
```
FUNCTION Bisect(xl, xu, es, imax, xr, iter, ea)
  iter = 0
  DO
    xrold = xr
    xr = (xl + xu) / 2
    iter = iter + 1
    IF xr ≠ 0 THEN
      ea = ABS((xr − xrold) / xr) * 100
    END IF
    test = f(xl) * f(xr)
    IF test < 0 THEN
      xu = xr
    ELSE IF test > 0 THEN
      xl = xr
    ELSE
      ea = 0
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  Bisect = xr
END Bisect
```

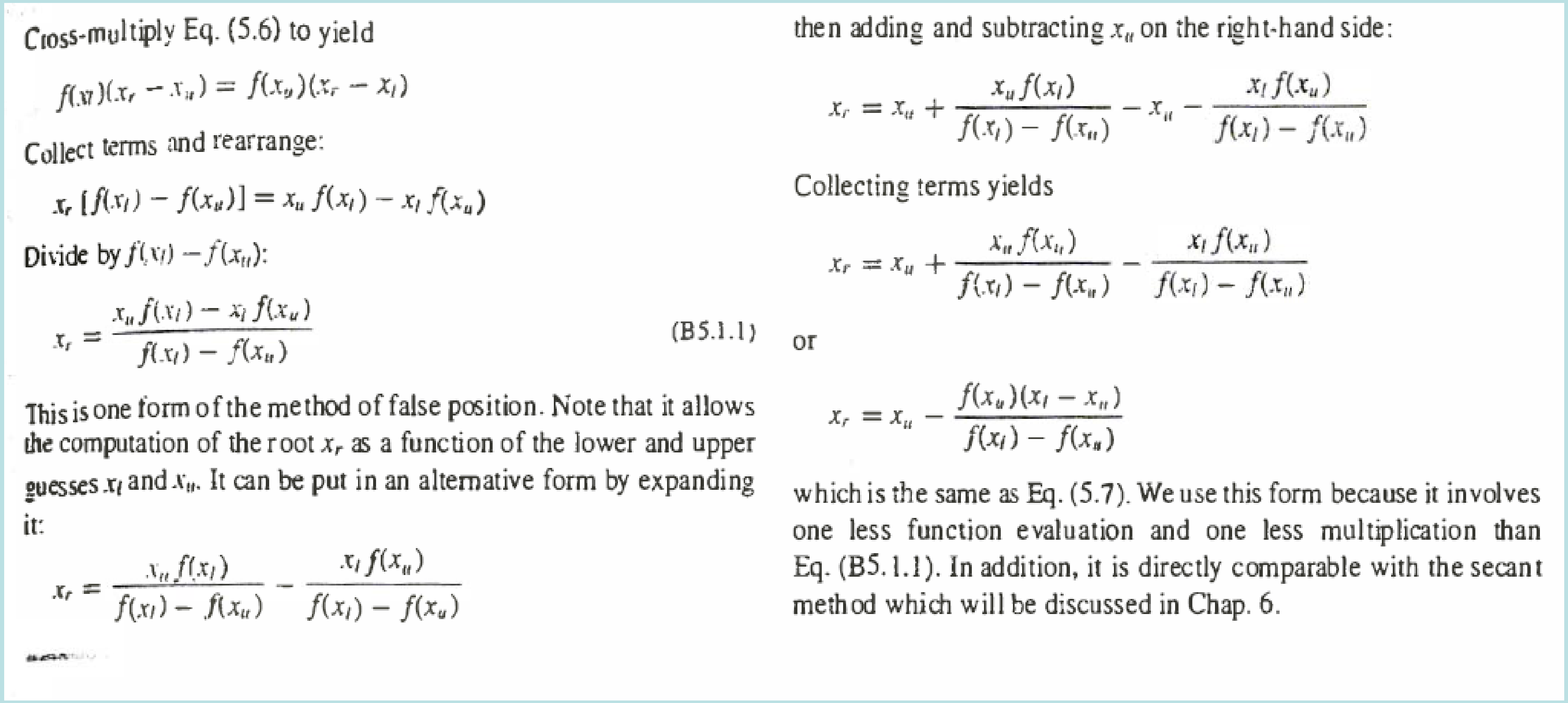


```
FUNCTION Bisect(xl, xu, es, imax, xr, iter, ea)
  iter = 0
  fl = f(xl)
  DO
    xrold = xr
    xr = (xl + xu) / 2
    fr = f(xr)
    iter = iter + 1
    IF xr ≠ 0 THEN
      ea = ABS((xr − xrold) / xr) * 100
    END IF
    test = fl * fr
    IF test < 0 THEN
      xu = xr
    ELSE IF test > 0 THEN
      xl = xr
      fl = fr
    ELSE
      ea = 0
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  Bisect = xr
END Bisect
```

The False-Position Method (Regula-Falsi)

- If a real root is bounded by x_l and x_u of $f(x)=0$,
- Then we can approximate the solution by doing a linear interpolation between the points $[x_l, f(x_l)]$ and $[x_u, f(x_u)]$
- to find the x_r value such that $l(x_r)=0$, $l(x)$ is the linear approximation of $f(x)$.





Cross-multiply Eq. (5.6) to yield

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

Collect terms and rearrange:

$$x_r\,[f(x_l) - f(x_u)] = x_u\, f(x_l) - x_l\, f(x_u)$$

Divide by $f(x_l) - f(x_u)$:

$$x_r = \frac{x_u\, f(x_l) - x_l\, f(x_u)}{f(x_l) - f(x_u)} \qquad \text{(B5.1.1)}$$

This is one form of the method of false position. Note that it allows the computation of the root x_r as a function of the lower and upper guesses x_l and x_u. It can be put in an alternative form by expanding it:

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

then adding and subtracting x_u on the right-hand side:

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

Collecting terms yields

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

or

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

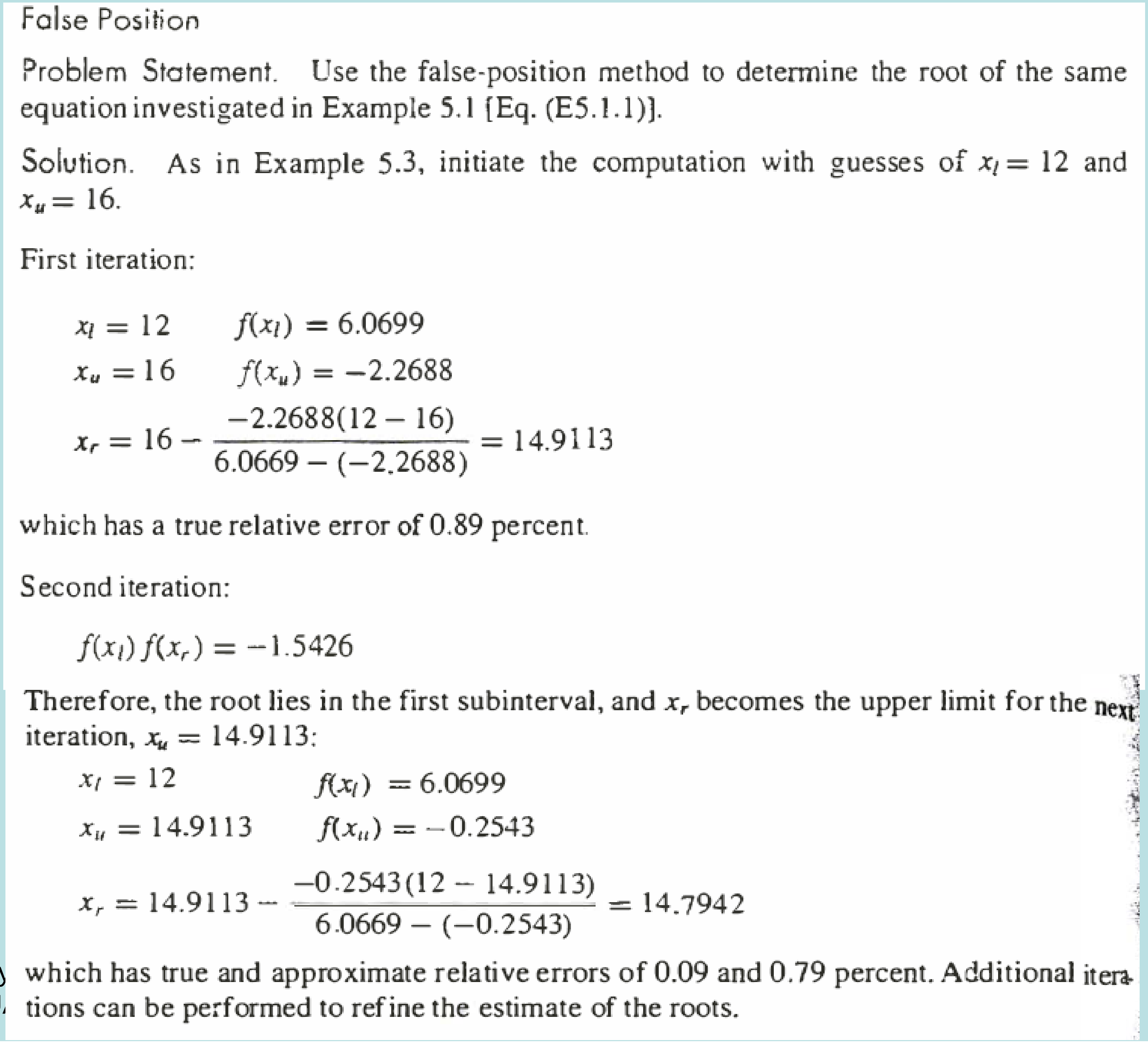
which is the same as Eq. (5.7). We use this form because it involves one less function evaluation and one less multiplication than Eq. (B5.1.1). In addition, it is directly comparable with the secant method which will be discussed in Chap. 6.

Procedure

1. Find a pair of values of x, x_l and x_u such that $f_l=f(x_l) <0$ and $f_u=f(x_u) >0$.
2. Estimate the value of the root from the following formula (Refer to Box 5.1)

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

and evaluate $f(x_r)$.



False Position

Problem Statement. Use the false-position method to determine the root of the same equation investigated in Example 5.1 [Eq. (E5.1.1)].

Solution. As in Example 5.3, initiate the computation with guesses of $x_l = 12$ and $x_u = 16$.

First iteration:

$$x_l = 12 \qquad f(x_l) = 6.0699$$
$$x_u = 16 \qquad f(x_u) = -2.2688$$
$$x_r = 16 - \frac{-2.2688(12 - 16)}{6.0669 - (-2.2688)} = 14.9113$$

which has a true relative error of 0.89 percent.

Second iteration:

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

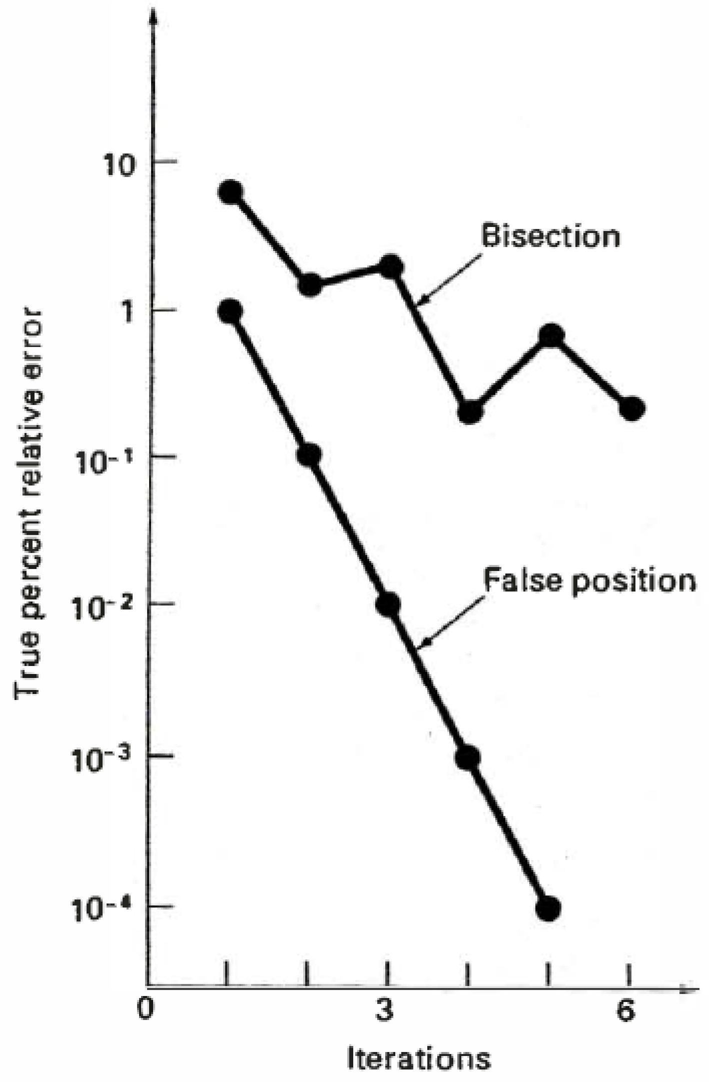
Therefore, the root lies in the first subinterval, and x_r becomes the upper limit for the next iteration, $x_u = 14.9113$:

$$x_l = 12 \qquad f(x_l) = 6.0699$$
$$x_u = 14.9113 \qquad f(x_u) = -0.2543$$
$$x_r = 14.9113 - \frac{-0.2543(12 - 14.9113)}{6.0669 - (-0.2543)} = 14.7942$$

which has true and approximate relative errors of 0.09 and 0.79 percent. Additional iterations can be performed to refine the estimate of the roots.

FIGURE 5.13

Comparison of the relative errors of the bisection and the false-position methods.



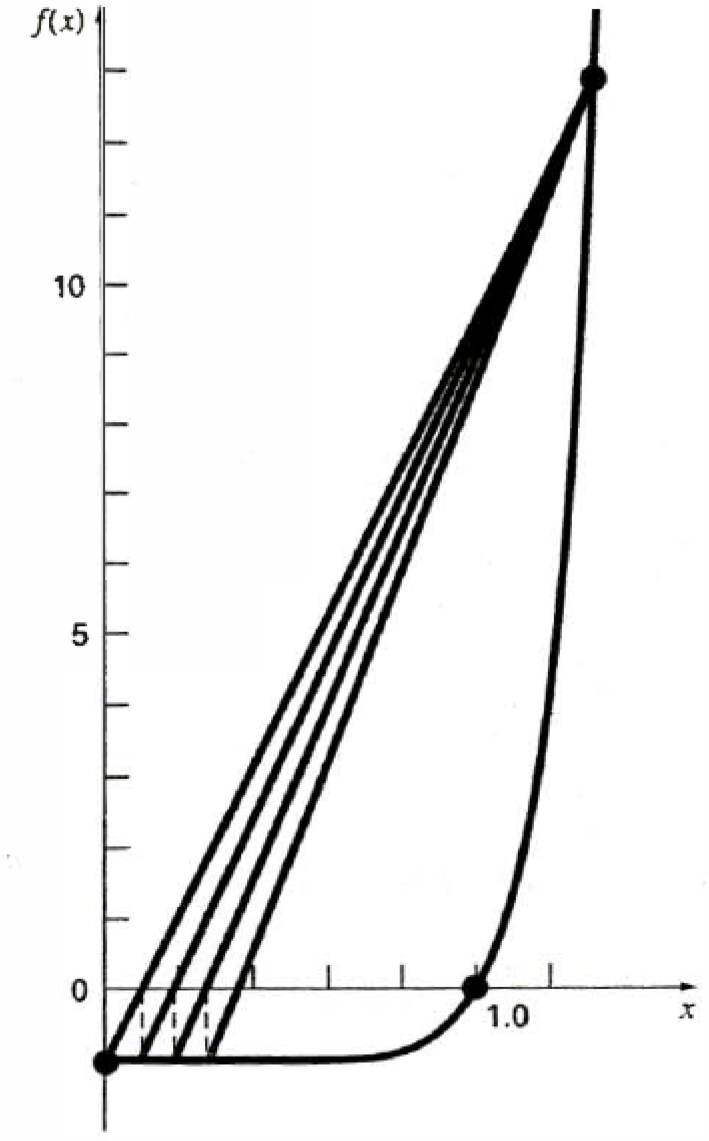
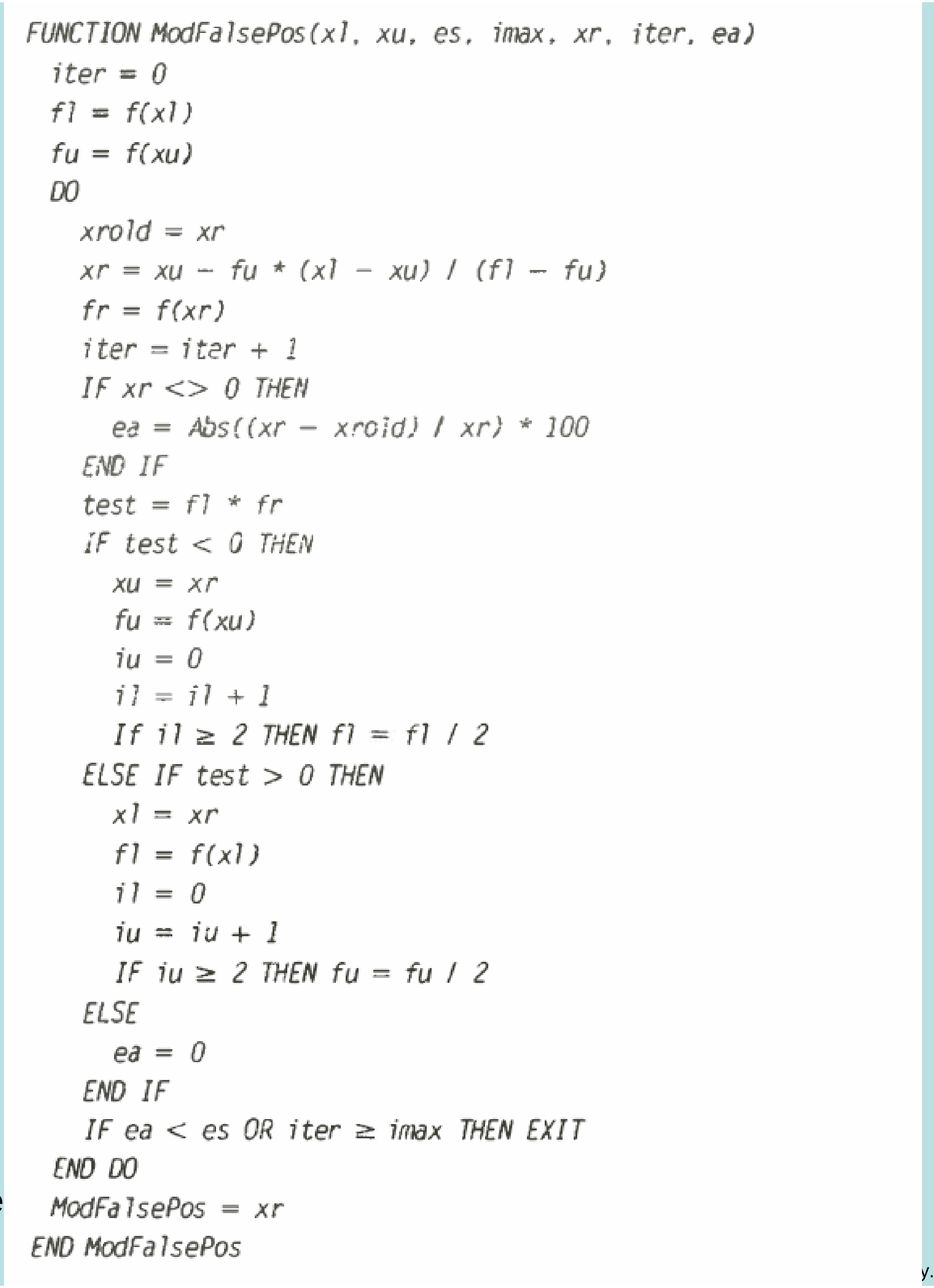


FIGURE 5.14
Plot of $f(x) = x^{10} - 1$, illustrating slow convergence of the false-position method.



```
FUNCTION ModFalsePos(xl, xu, es, imax, xr, iter, ea)
  iter = 0
  fl = f(xl)
  fu = f(xu)
  DO
    xrold = xr
    xr = xu - fu * (xl - xu) / (fl - fu)
    fr = f(xr)
    iter = iter + 1
    IF xr <> 0 THEN
      ea = Abs((xr - xrold) / xr) * 100
    END IF
    test = fl * fr
    IF test < 0 THEN
      xu = xr
      fu = f(xu)
      iu = 0
      il = il + 1
      If il ≥ 2 THEN fl = fl / 2
    ELSE IF test > 0 THEN
      xl = xr
      fl = f(xl)
      il = 0
      iu = iu + 1
      IF iu ≥ 2 THEN fu = fu / 2
    ELSE
      ea = 0
    END IF
    IF ea < es OR iter ≥ imax THEN EXIT
  END DO
  ModFalsePos = xr
END ModFalsePos
```

3. Use the new point to replace one of the original points, keeping the two points on opposite sides of the x axis.

If $f(x_r)<0$ then $x_l=x_r$ $==>$ $f_l=f(x_r)$

If $f(x_r)>0$ then $x_u=x_r$ $==>$ $f_u=f(x_r)$

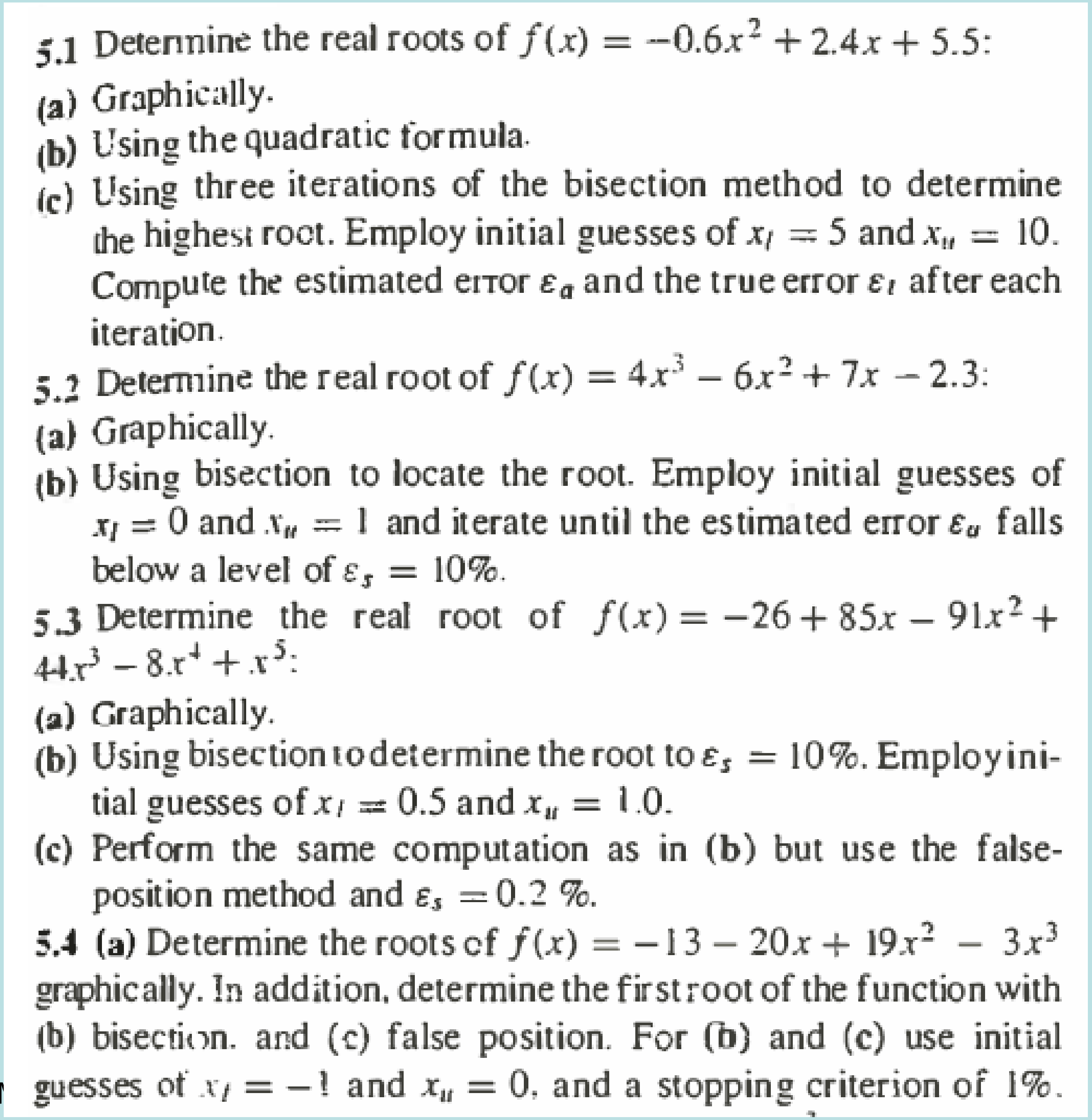
If $f(x_r)=0$ then you have found the root and need go no further!

4. See if the new x_l and x_u are close enough for convergence to be declared. If they are not go back to step 2.

- Why this method?
  - Faster
  - Always converges for a single root.

➔See Sec.5.3.1, Pitfalls of the False-Position Method

Note: Always check by substituting estimated root in the original equation to determine whether $f(x_r) \approx 0$.


5.1 Determine the real roots of $f(x) = -0.6x^2 + 2.4x + 5.5$:

(a) Graphically.

(b) Using the quadratic formula.

(c) Using three iterations of the bisection method to determine the highest root. Employ initial guesses of $x_l = 5$ and $x_u = 10$. Compute the estimated error ε_a and the true error ε_t after each iteration.

5.2 Determine the real root of $f(x) = 4x^3 - 6x^2 + 7x - 2.3$:

(a) Graphically.

(b) Using bisection to locate the root. Employ initial guesses of $x_l = 0$ and $x_u = 1$ and iterate until the estimated error ε_a falls below a level of $\varepsilon_s = 10\%$.

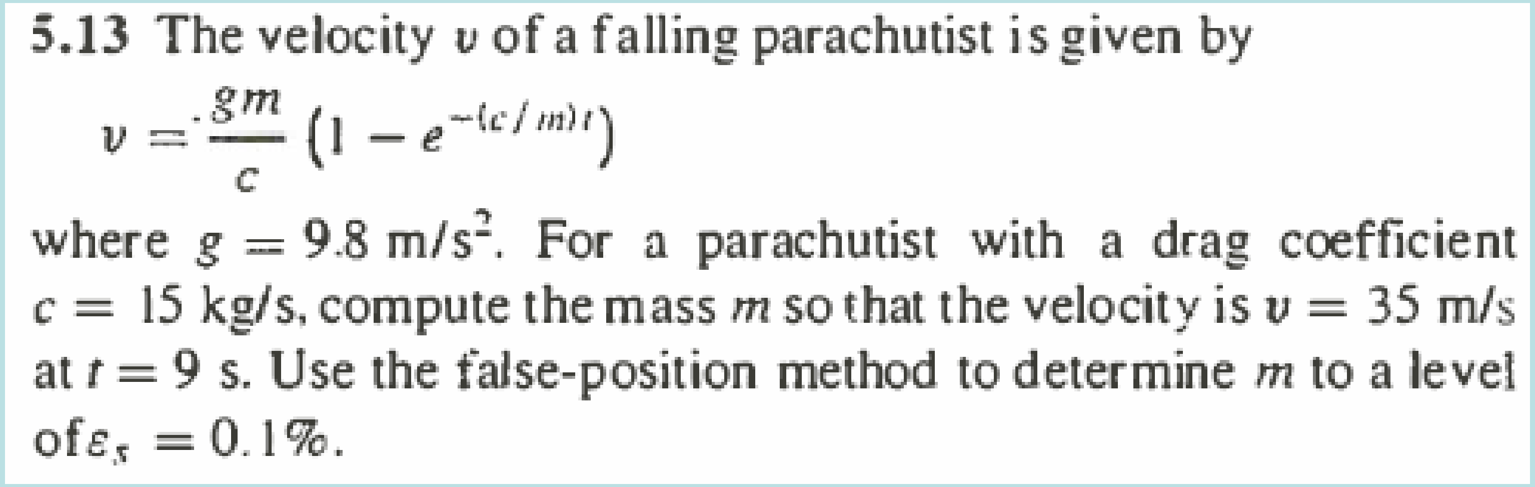
5.3 Determine the real root of $f(x) = -26 + 85x - 91x^2 + 44x^3 - 8x^4 + x^5$:

(a) Graphically.

(b) Using bisection to determine the root to $\varepsilon_s = 10\%$. Employ initial guesses of $x_l = 0.5$ and $x_u = 1.0$.

(c) Perform the same computation as in (b) but use the false-position method and $\varepsilon_s = 0.2\ \%$.

5.4 **(a)** Determine the roots of $f(x) = -13 - 20x + 19x^2 - 3x^3$ graphically. In addition, determine the first root of the function with **(b)** bisection, and **(c)** false position. For **(b)** and **(c)** use initial guesses of $x_l = -1$ and $x_u = 0$, and a stopping criterion of 1%.



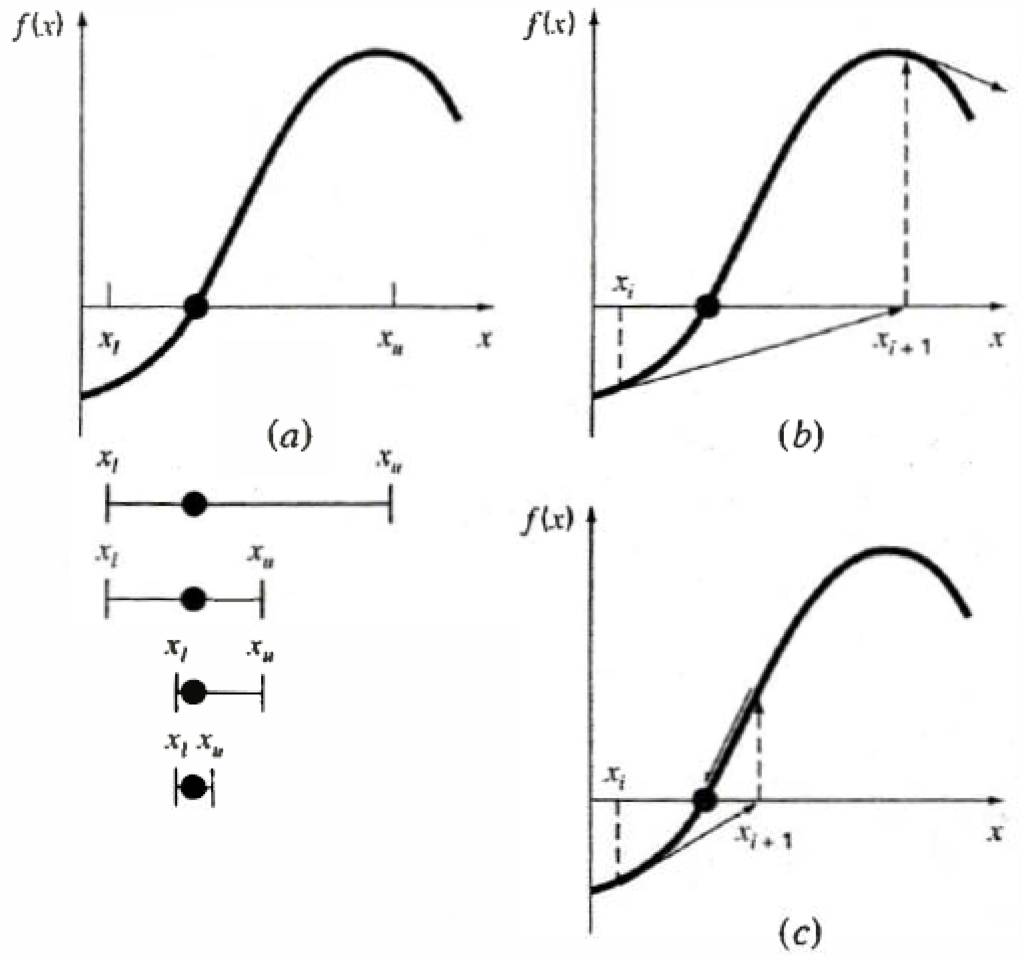
5.13 The velocity v of a falling parachutist is given by

$$v = \frac{gm}{c}\left(1 - e^{-(c/m)t}\right)$$

where $g = 9.8\text{ m/s}^2$. For a parachutist with a drag coefficient $c = 15$ kg/s, compute the mass m so that the velocity is $v = 35$ m/s at $t = 9$ s. Use the false-position method to determine m to a level of $\varepsilon_s = 0.1\%$.

OPEN METHODS

FIGURE 6.1
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_{i+1} in an iterative fashion. Thus, the method can either (*b*) diverge or (*c*) converge rapidly, depending on the value of the initial guess.



by Martin Mendez, UASLP

Simple Fixed-point Iteration

•Rearrange the function so that x is on the left side of the equation:

$$f(x) = 0 \quad \Rightarrow \quad g(x) = x$$

$$x_k = g(x_{k-1}) \qquad x_o \text{ given, k} = 1, 2, \ldots$$

•Bracketing methods are “convergent”.

•Fixed-point methods may sometime “diverge”, depending on the stating point (initial guess) and how the function behaves.

Example:

$$f(x) = x^2 - x - 2 \qquad\qquad x \succ 0$$

$$g(x) = x^2 - 2$$

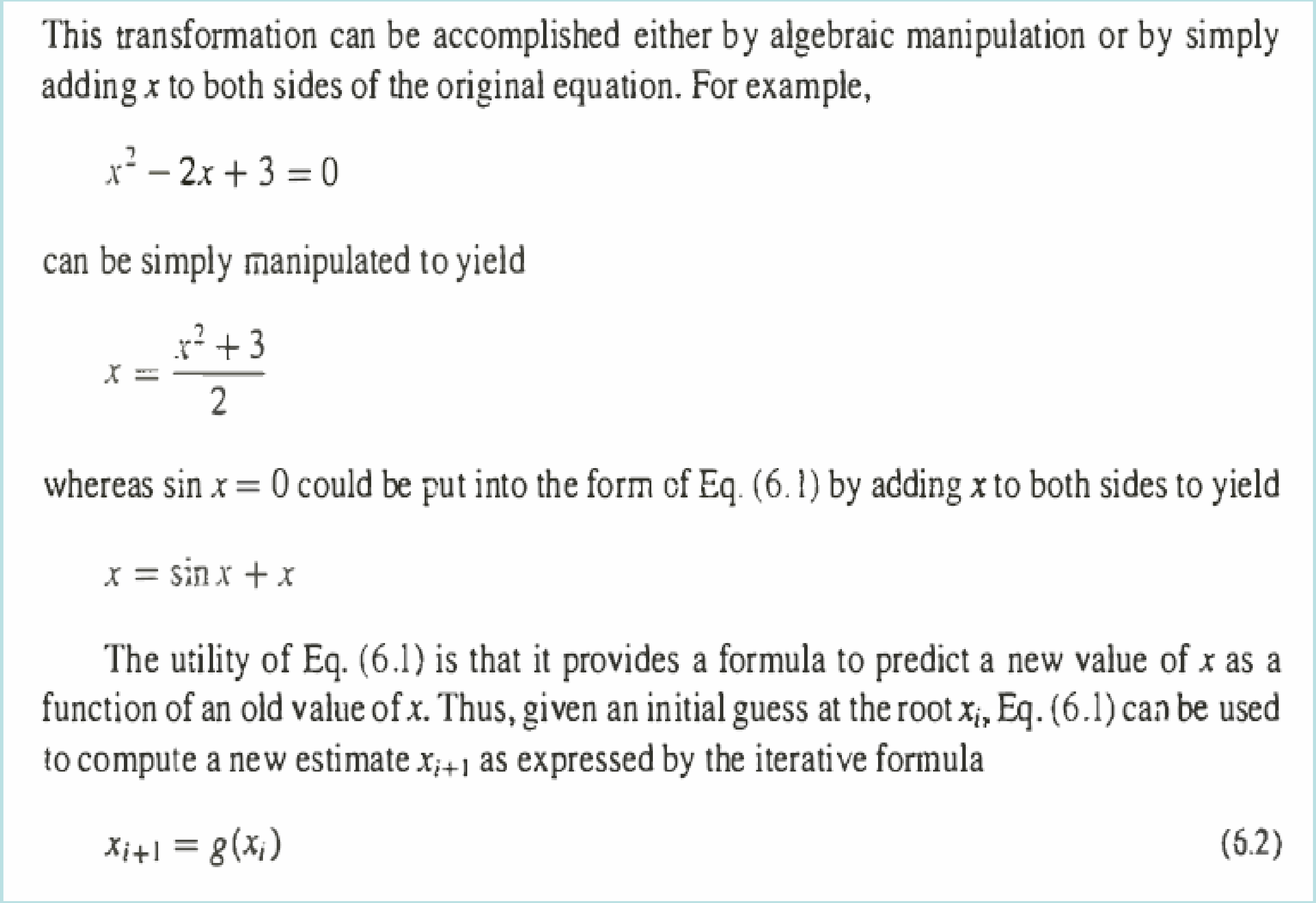
or

$$g(x) = \sqrt{x+2}$$

or

$$g(x) = 1 + \frac{2}{x}$$

$\vdots$



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

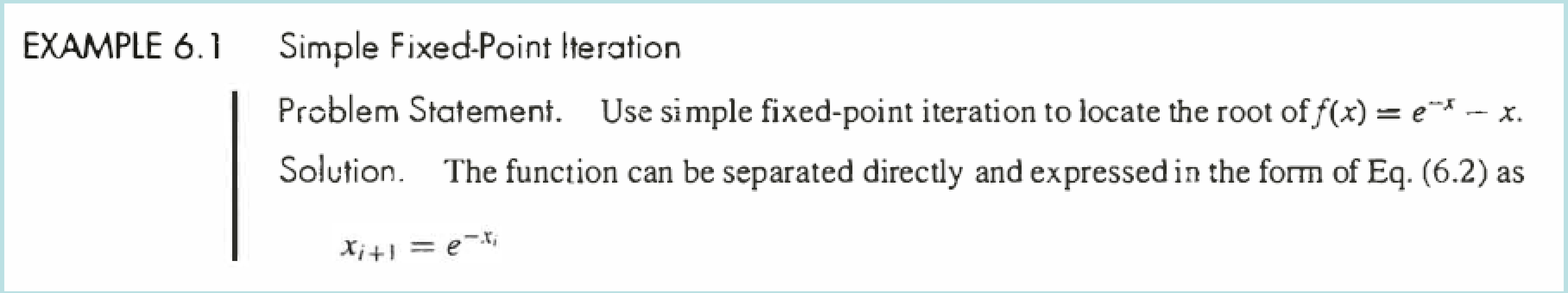
$$x = \frac{x^2 + 3}{2}$$

whereas $\sin x = 0$ could be put into the form of Eq. (6.1) by adding x to both sides to yield

$$x = \sin x + x$$

The utility of Eq. (6.1) is that 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, Eq. (6.1) can be used to compute a new estimate x_{i+1} as expressed by the iterative formula

$$x_{i+1} = g(x_i) \tag{6.2}$$

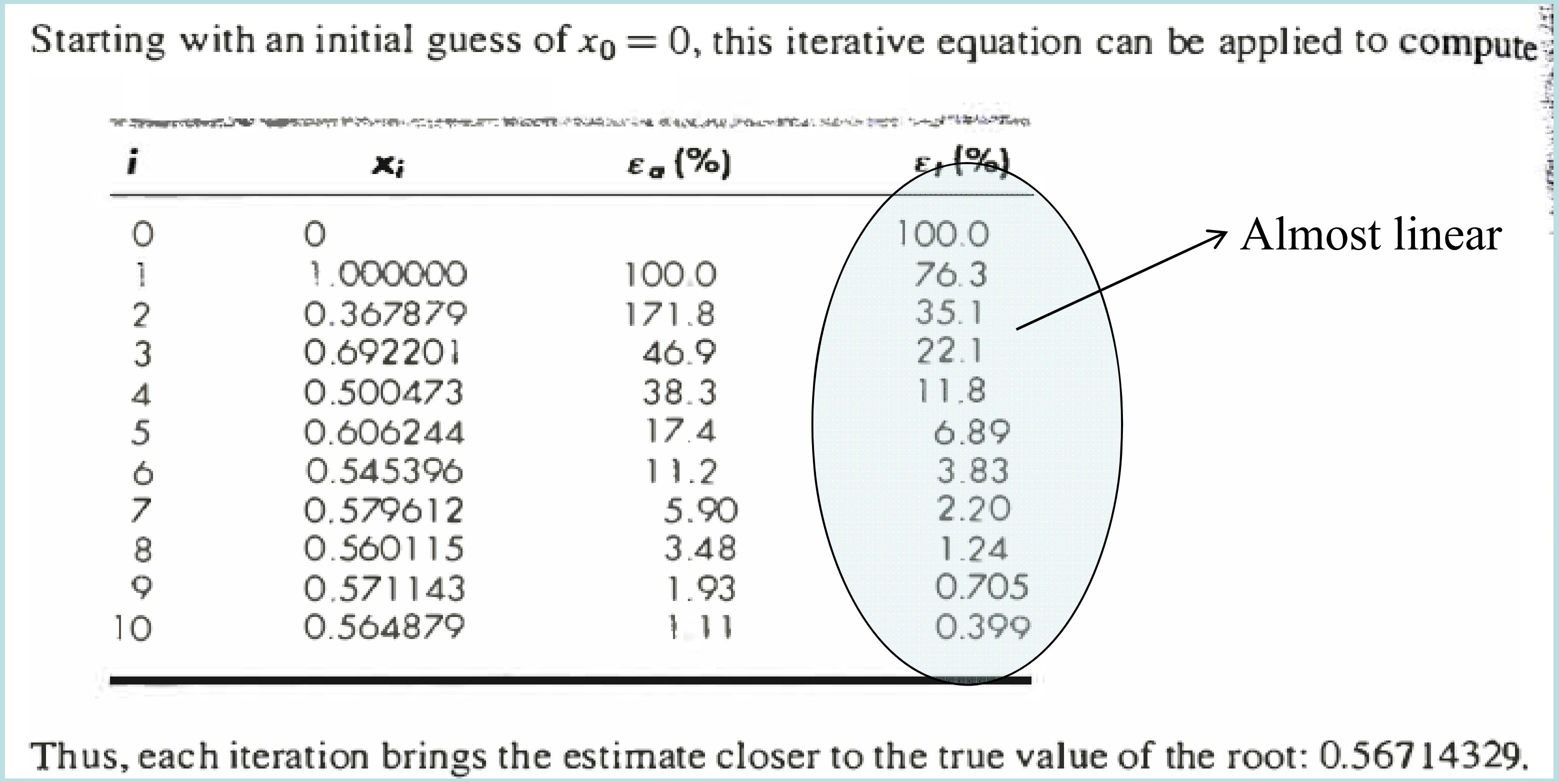


EXAMPLE 6.1 Simple Fixed-Point Iteration

Problem Statement. Use simple fixed-point iteration to locate the root of $f(x) = e^{-x} - x$.

Solution. The function can be separated directly and expressed in the form of Eq. (6.2) 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	ε_a (%)	ε_t (%)
0	0		100.0
1	1.000000	100.0	76.3
2	0.367879	171.8	35.1
3	0.692201	46.9	22.1
4	0.500473	38.3	11.8
5	0.606244	17.4	6.89
6	0.545396	11.2	3.83
7	0.579612	5.90	2.20
8	0.560115	3.48	1.24
9	0.571143	1.93	0.705
10	0.564879	1.11	0.399

Almost linear

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

Convergence

Figure 6.2

- $x=g(x)$ can be expressed as a pair of equations:

  $y_1=x$

  $y_2=g(x)$ (component equations)
- Plot them separately.

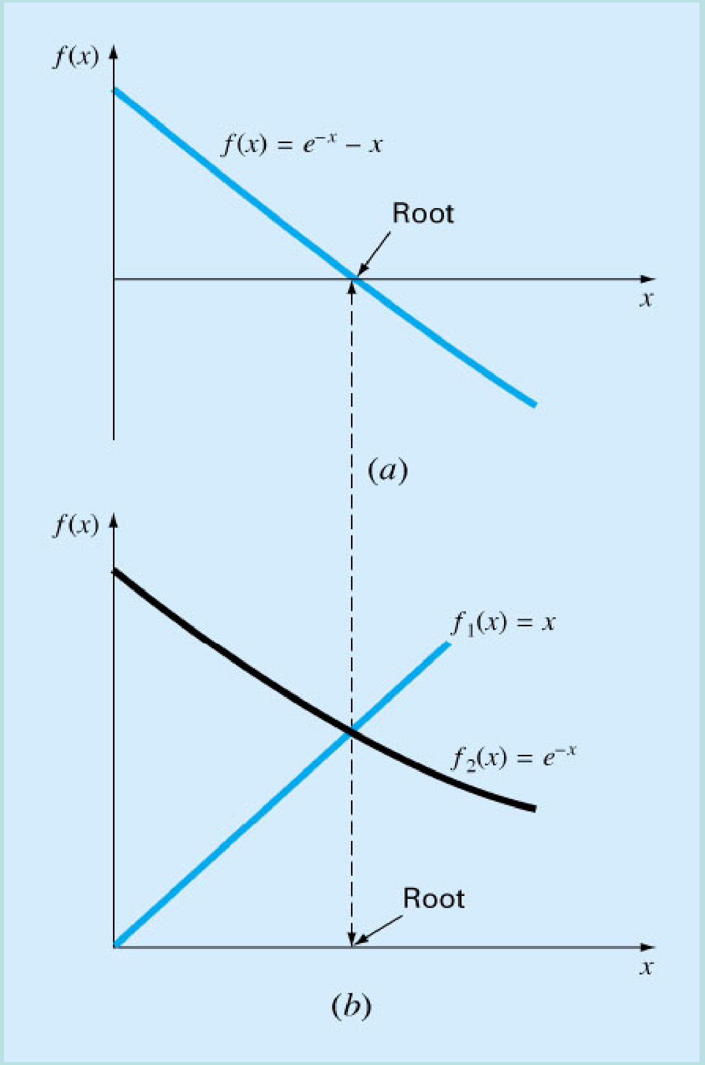
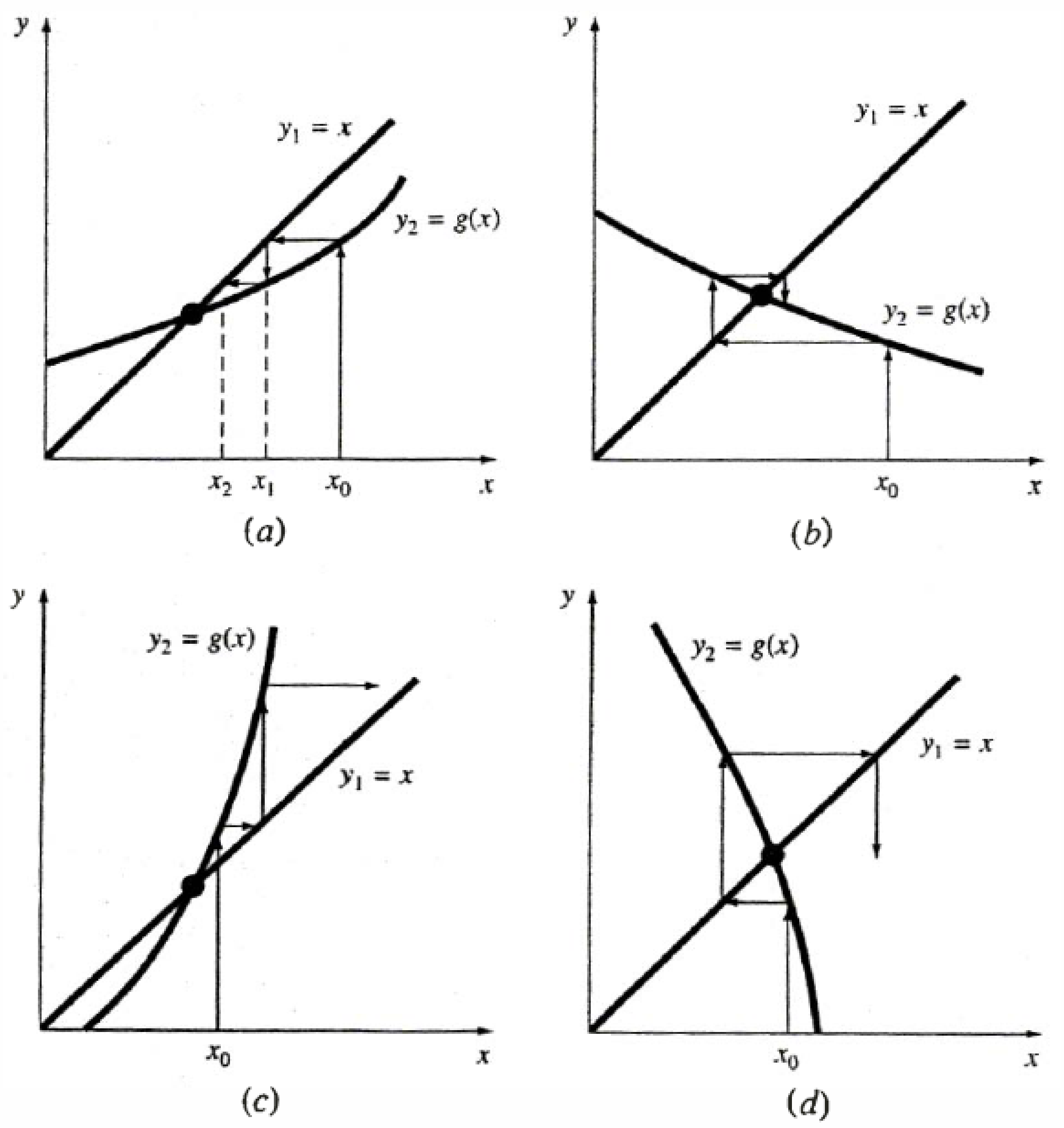


FIGURE 6.3
Graphical depiction of (*a*) and (*b*) convergence and (*c*) and (*d*) divergence of simple fixed-point iteration. Graphs (*a*) and (*c*) are called monotone patterns, whereas (*b*) and (*d*) are called oscillating or spiral patterns. Note that convergence occurs when $|g'(x)| < 1$.

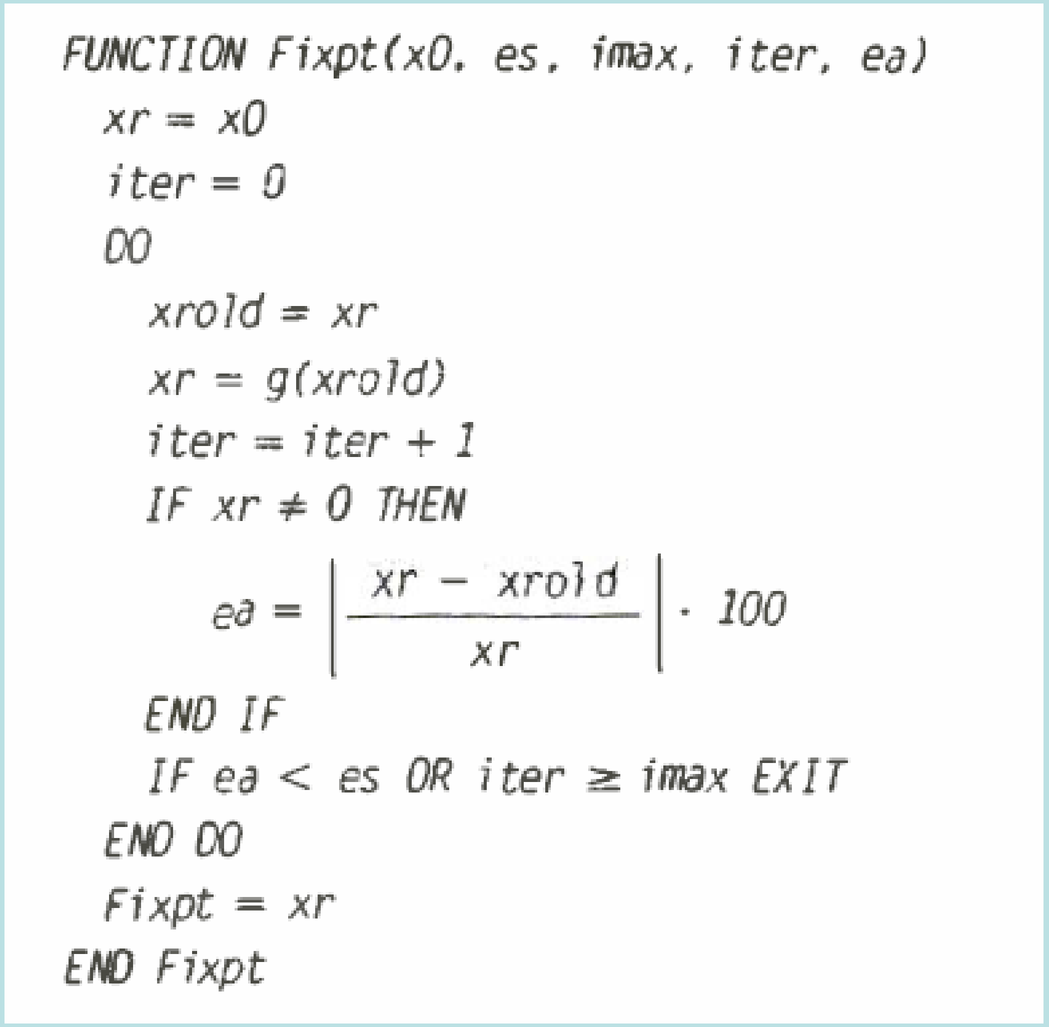


Conclusion

- Fixed-point iteration converges if

$$\left|g'(x)\right| \prec 1 \quad \text{(slope of the line f(x) = x)}$$

•When the method converges, the error is roughly proportional to or less than the error of the previous step, therefore it is called “linearly convergent.”



```
FUNCTION Fixpt(x0, es, imax, iter, ea)
  xr = x0
  iter = 0
  DO
    xrold = xr
    xr = g(xrold)
    iter = iter + 1
    IF xr ≠ 0 THEN
      $ea = \left| \frac{xr - xrold}{xr} \right| \cdot 100$
    END IF
    IF ea < es OR iter ≥ imax EXIT
  END DO
  Fixpt = xr
END Fixpt
```

Newton-Raphson Method

- Most widely used method.
- Based on Taylor series expansion:

$$f(x_{i+1}) = f(x_i) + f'(x_i)\Delta x + f''(x_i)\frac{\Delta x^2}{2!} + O\Delta x^3$$

The root is the value of $\mathrm{x}_{\mathrm{i+1}}$ when $\mathrm{f(x_{i+1}) = 0}$

Rearranging,

Solve for

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

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

Newton-Raphson formula

Fig. 6.5

- A convenient method for functions whose derivatives can be evaluated analytically. It may not be convenient for functions whose derivatives cannot be evaluated analytically.

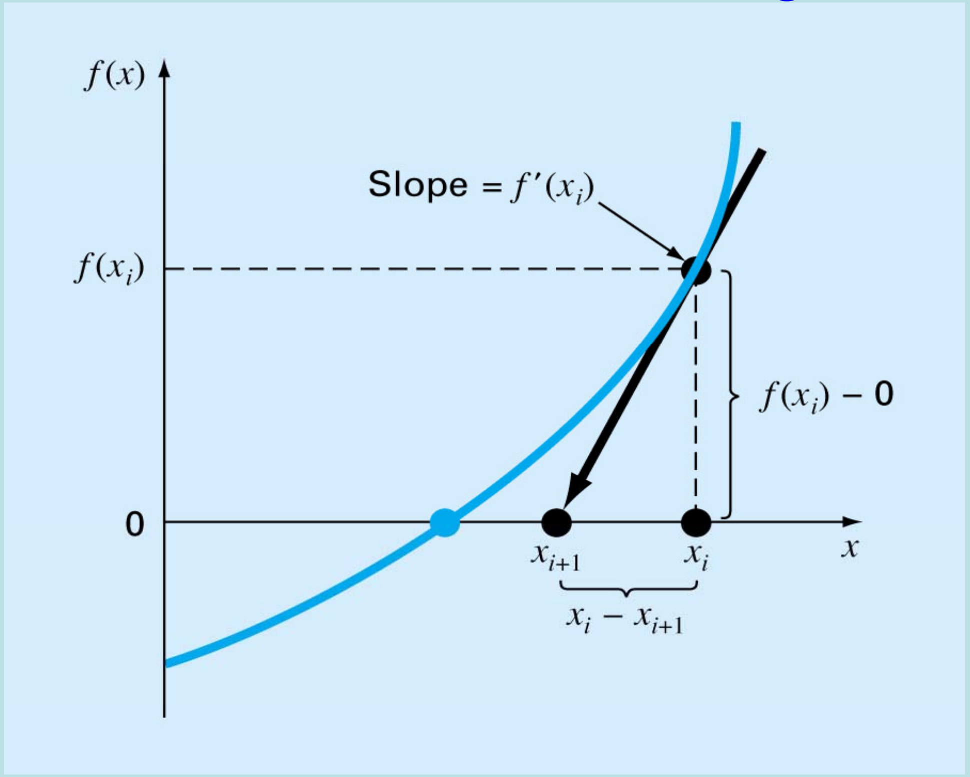
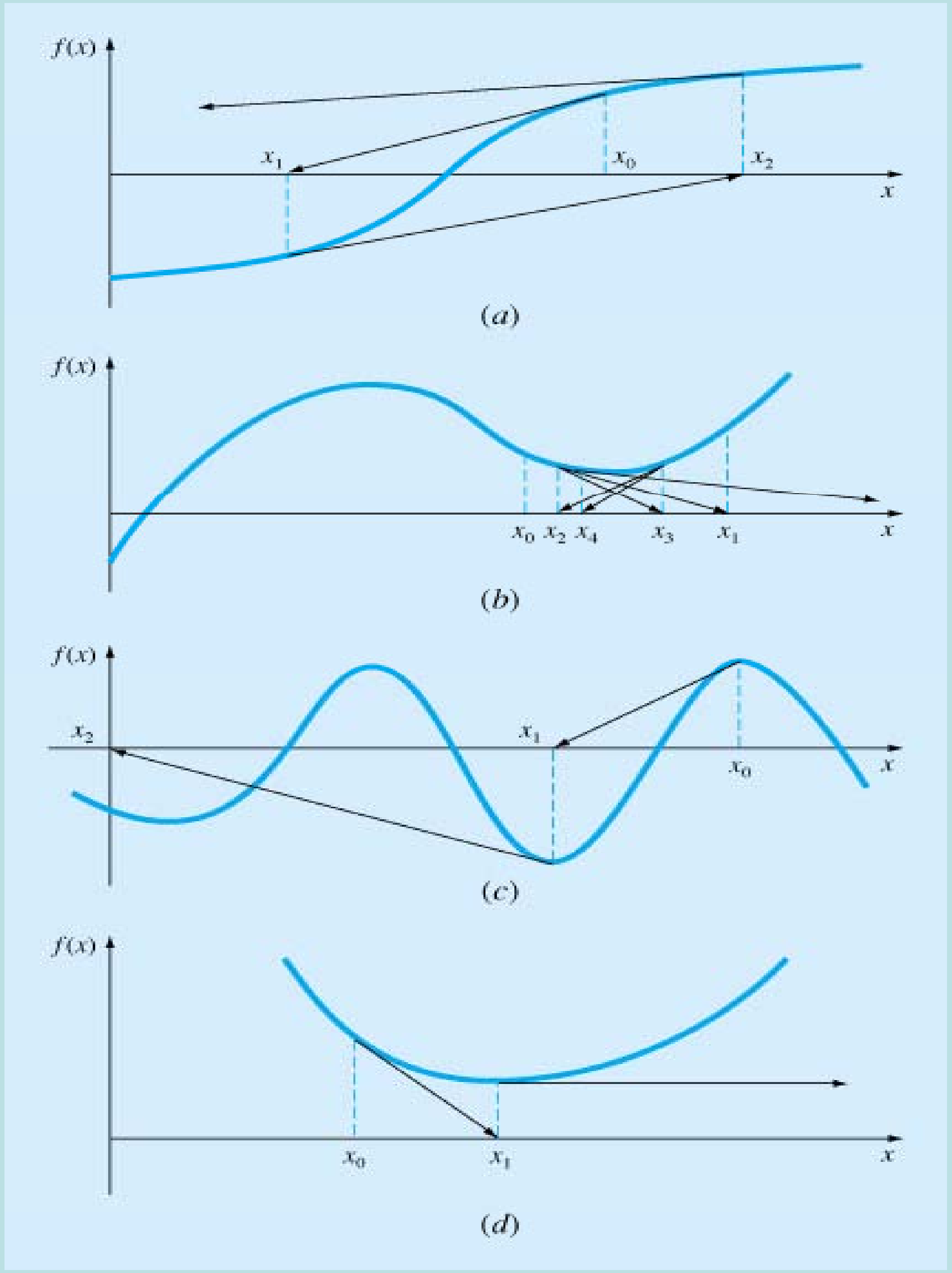
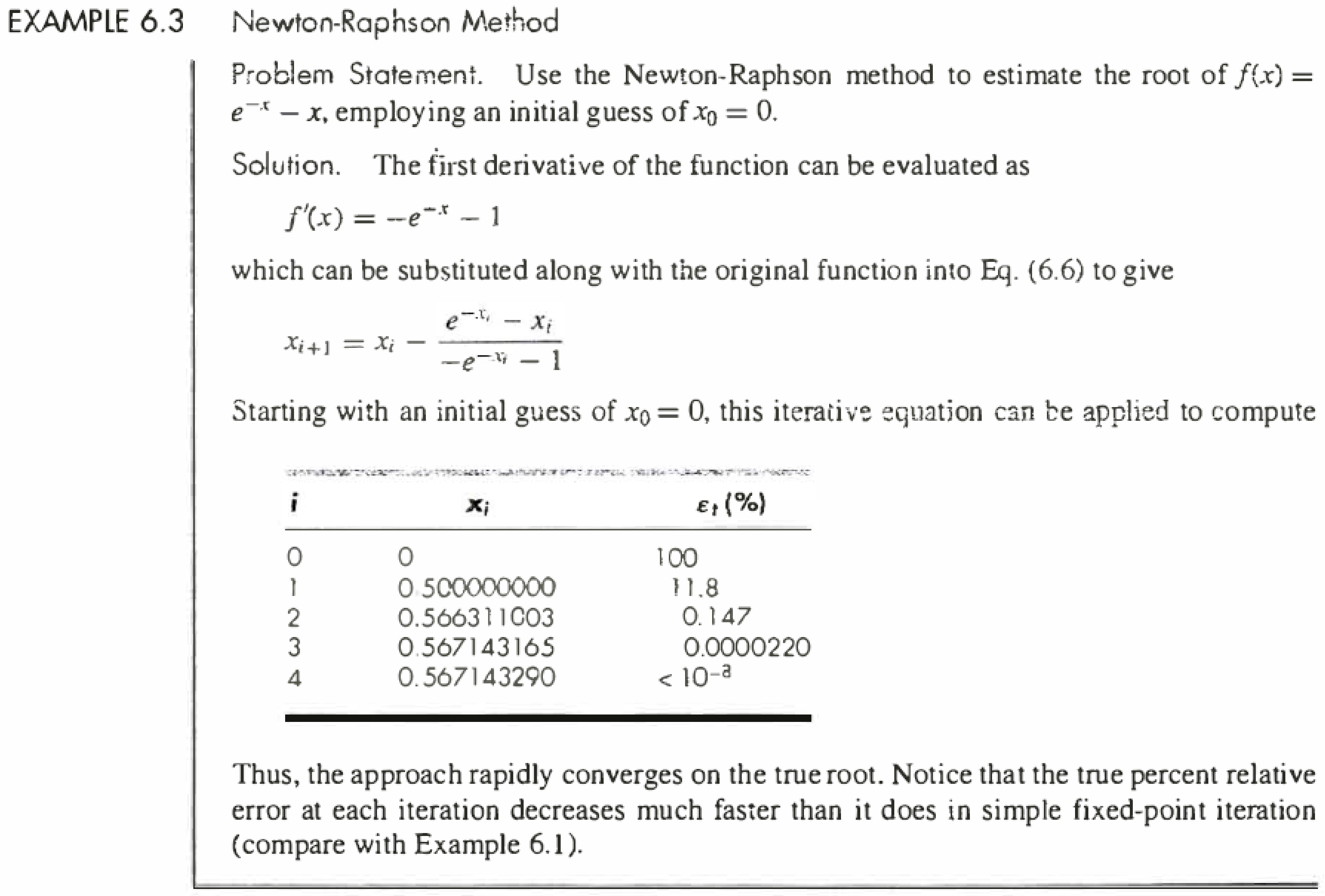


Fig. 6.6



Ej.
f(x)=exp(-x)-x



EXAMPLE 6.3 Newton-Raphson Method

Problem Statement. 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

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

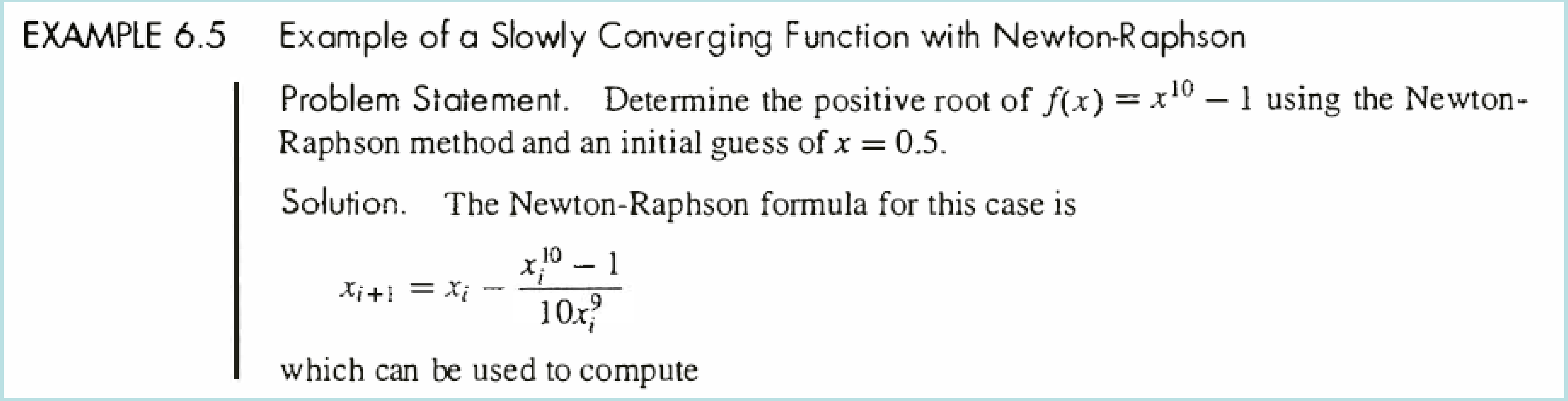
which can be substituted along with the original function into Eq. (6.6) to give

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

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

i	x_i	ε_t (%)
0	0	100
1	0.500000000	11.8
2	0.566311003	0.147
3	0.567143165	0.0000220
4	0.567143290	$< 10^{-8}$

Thus, the approach rapidly converges on the true root. Notice that the true percent relative error at each iteration decreases much faster than it does in simple fixed-point iteration (compare with Example 6.1).



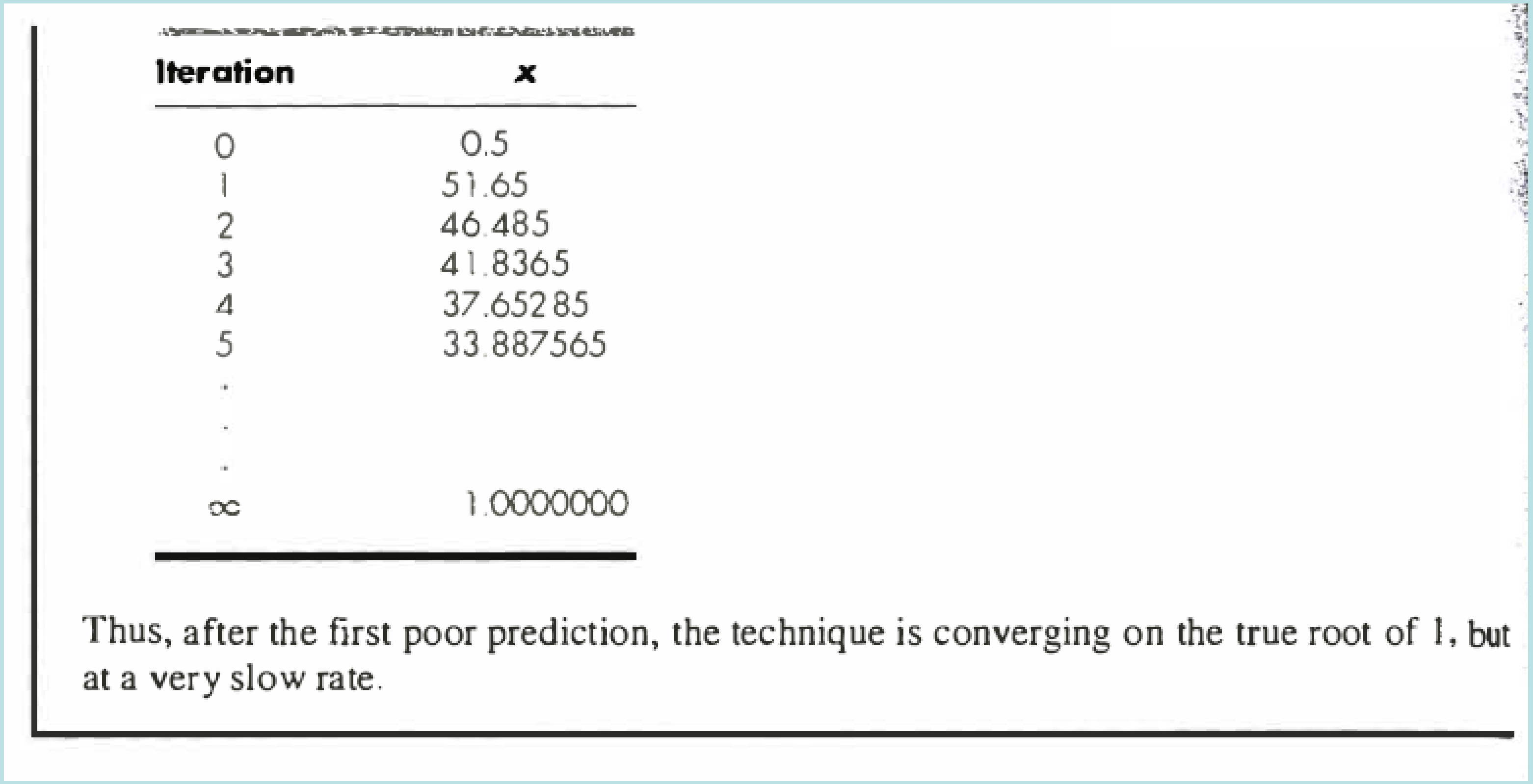
EXAMPLE 6.5 Example of a Slowly Converging Function with Newton-Raphson

Problem Statement. Determine the positive root of $f(x) = x^{10} - 1$ using the Newton-Raphson method and an initial guess of $x = 0.5$.

Solution. The Newton-Raphson formula for this case is

$$x_{i+1} = x_i - \frac{x_i^{10} - 1}{10x_i^9}$$

which can be used to compute



Iteration	x
0	0.5
1	51.65
2	46.485
3	41.8365
4	37.65285
5	33.887565
.	
.	
.	
∞	1.0000000

Thus, after the first poor prediction, the technique is converging on the true root of 1, but at a very slow rate.

The Secant Method

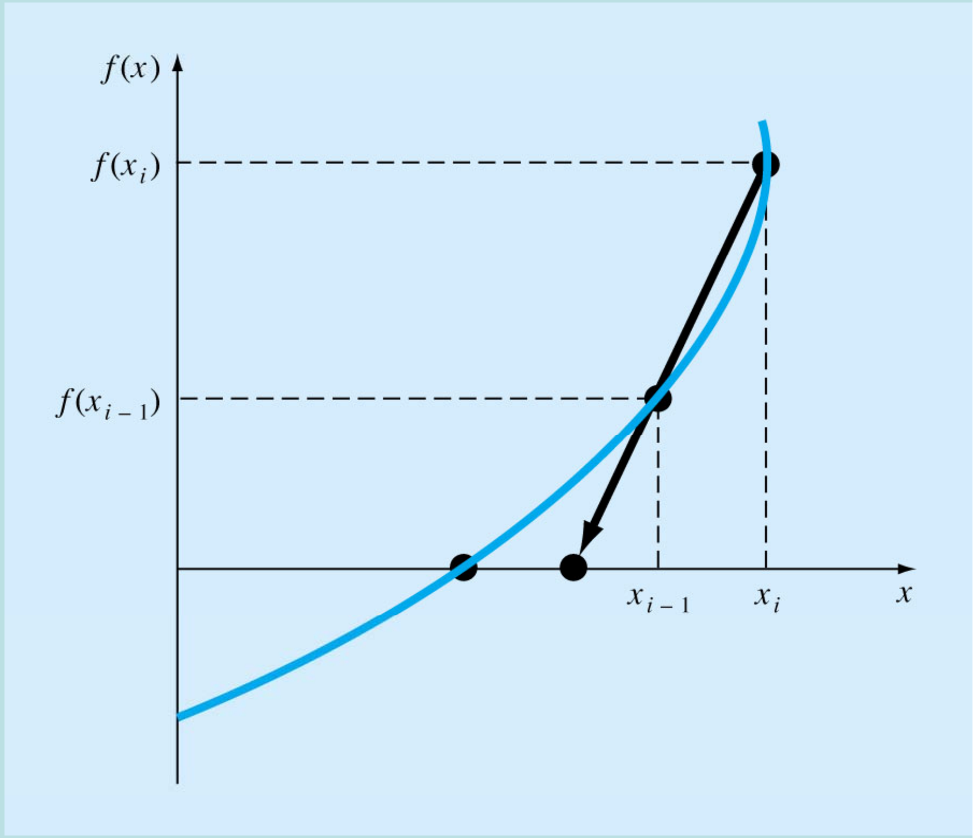
- A slight variation of Newton's method for functions whose derivatives are difficult to evaluate. For these cases the derivative can be approximated by a backward finite divided difference.

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

$$x_{i+1} = x_i - f(x_i)\frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} \qquad i = 1,2,3,\ldots$$

Fig. 6.7

- Requires two initial estimates of x , e.g, x_o, x_1. However, because f(x) is not required to change signs between estimates, it is not classified as a “bracketing” method.
- The secant method has the same properties as Newton’s method. Convergence is not guaranteed for all x_o, f(x).



EXAMPLE 6.6 The Secant Method

Problem Statement. Use the secant method to estimate the root of $f(x) = e^{-x} - x$. Start with initial estimates of $x_{-1} = 0$ and $x_0 = 1.0$.

Solution. Recall that the true root is 0.56714329. . . .

First iteration:

$$x_{-1} = 0 \qquad f(x_{-1}) = 1.00000$$

$$x_0 = 1 \qquad f(x_0) = -0.63212$$

$$x_1 = 1 - \frac{-0.63212(0-1)}{1-(-0.63212)} = 0.61270 \qquad \varepsilon_t = 8.0\%$$

Second iteration:

$$x_0 = 1 \qquad f(x_0) = -0.63212$$

$$x_1 = 0.61270 \qquad f(x_1) = -0.07081$$

(Note that both estimates are now on the same side of the root.)

$$x_2 = 0.61270 - \frac{-0.07081(1-0.61270)}{-0.63212-(-0.07081)} = 0.56384 \qquad \varepsilon_t = 0.58\%$$

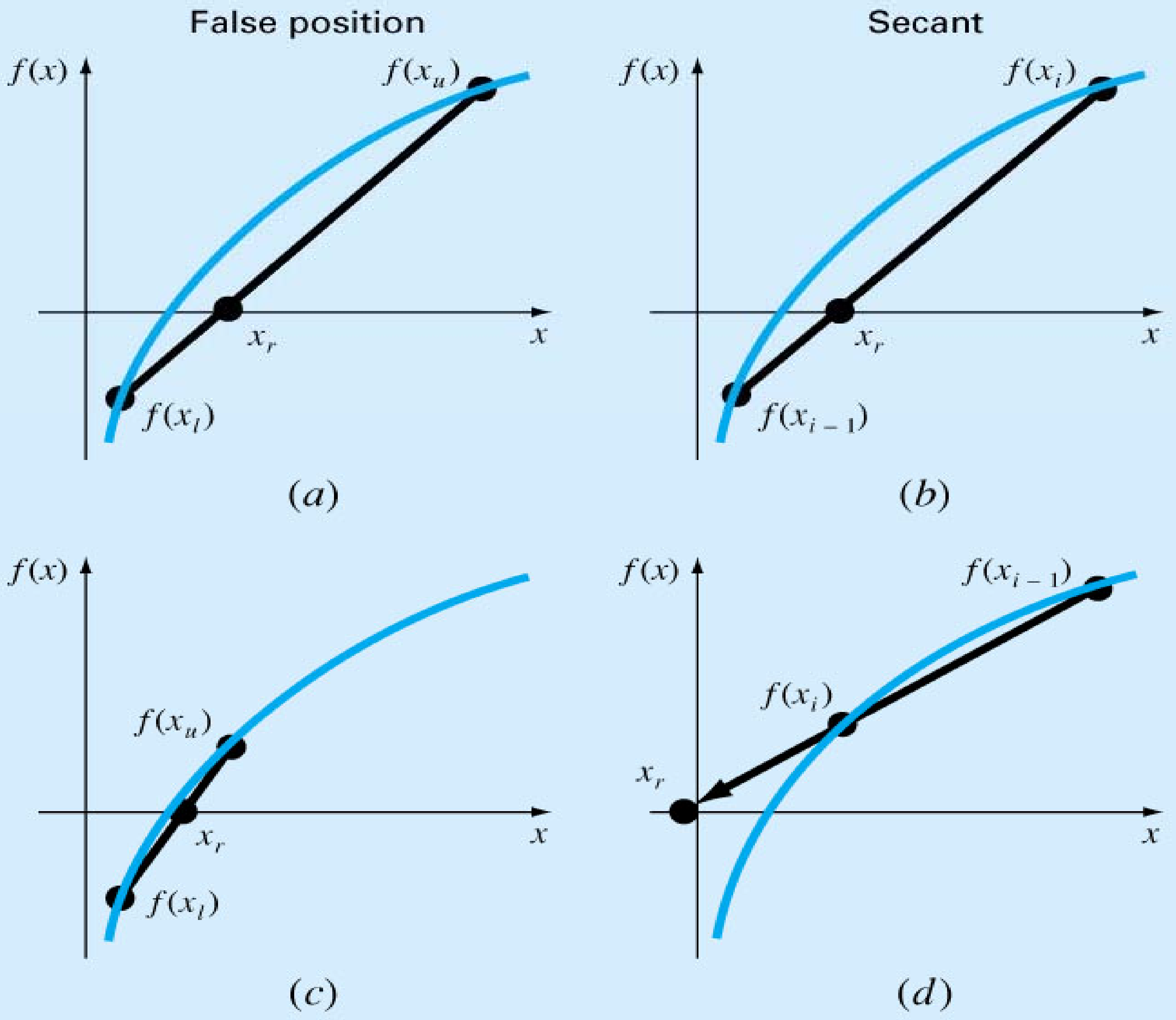
Third iteration:

$$x_1 = 0.61270 \qquad f(x_1) = -0.07081$$

$$x_2 = 0.56384 \qquad f(x_2) = 0.00518$$

$$x_3 = 0.56384 - \frac{0.00518(0.61270-0.56384)}{-0.07081-(-0.00518)} = 0.56717 \qquad \varepsilon_t = 0.0048\%$$

Fig. 6.8



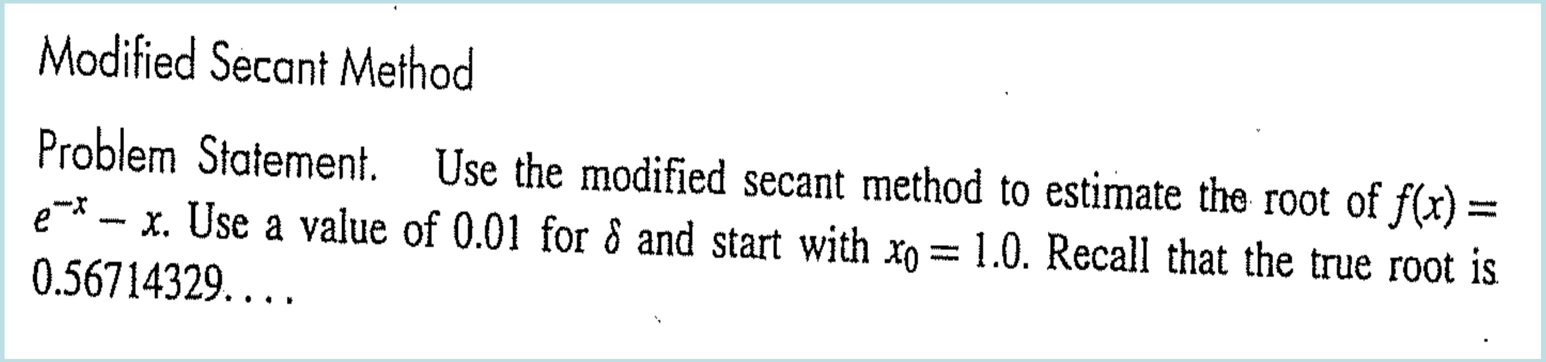
The Modified Secant Method

- Rather than using two arbitrary values to estimate the derivative, an alternative approach involves a fractional perturbation of the independent variable to estimate f`(x).

$$f'(x_i) \cong \frac{f(x_i + \delta x_i)}{\delta x_i}$$

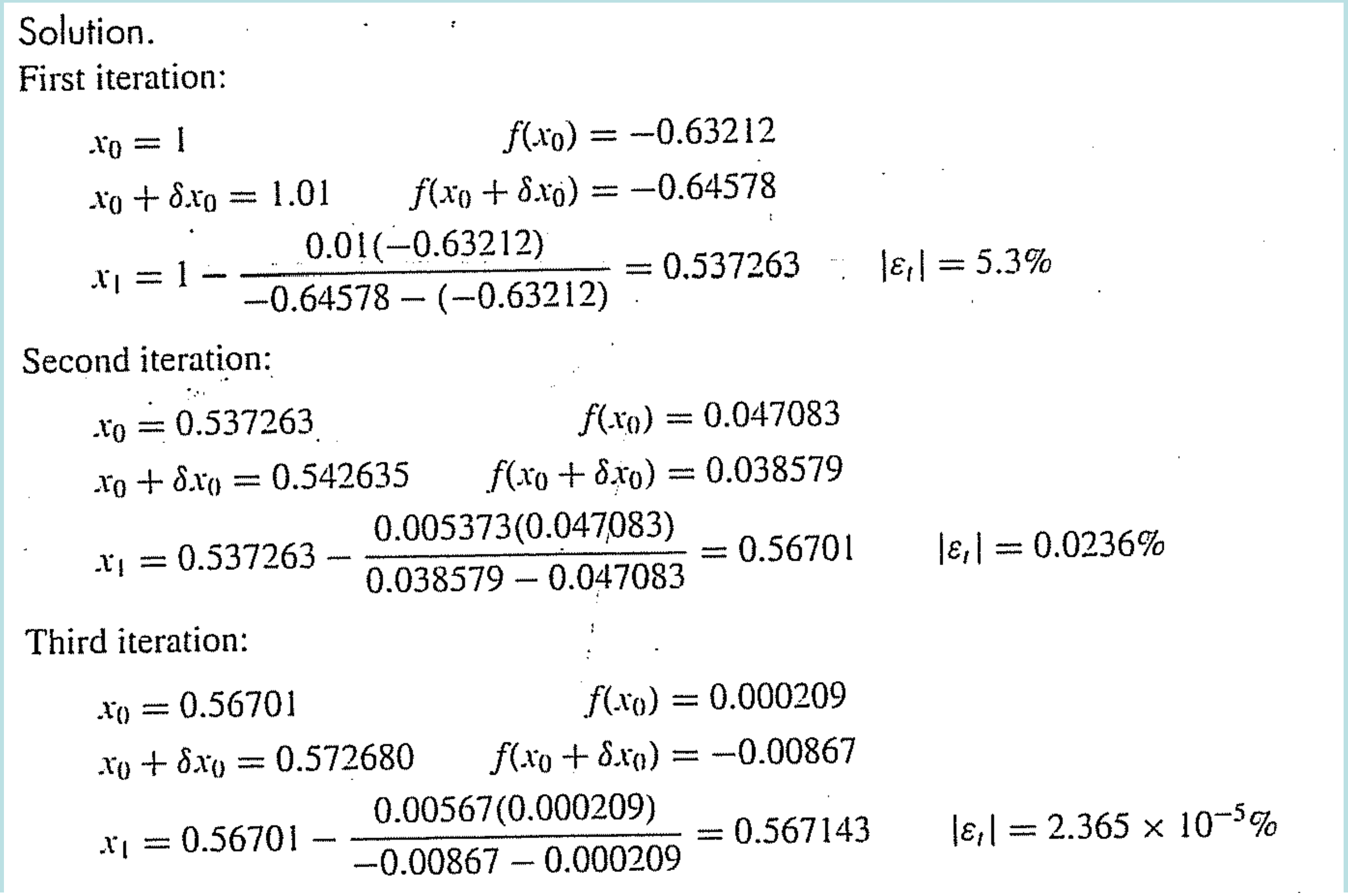
$$x_{i+1} = x_i - \frac{\delta x_i f(x_i)}{f(x_i + \delta x_i) - f(x_i)}$$

Where ‘delta’ is a small fraction change



Modified Secant Method

Problem Statement. Use the modified secant method to estimate the root of $f(x) = e^{-x} - x$. Use a value of 0.01 for δ and start with $x_0 = 1.0$. Recall that the true root is 0.56714329. . . .



Solution.

First iteration:

$$x_0 = 1 \qquad f(x_0) = -0.63212$$

$$x_0 + \delta x_0 = 1.01 \qquad f(x_0 + \delta x_0) = -0.64578$$

$$x_1 = 1 - \frac{0.01(-0.63212)}{-0.64578 - (-0.63212)} = 0.537263 \qquad |\varepsilon_t| = 5.3\%$$

Second iteration:

$$x_0 = 0.537263 \qquad f(x_0) = 0.047083$$

$$x_0 + \delta x_0 = 0.542635 \qquad f(x_0 + \delta x_0) = 0.038579$$

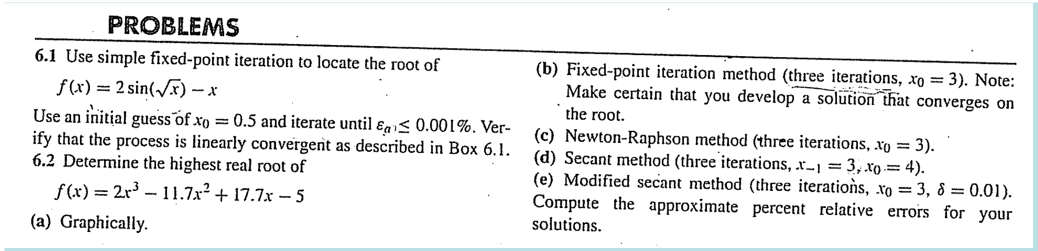
$$x_1 = 0.537263 - \frac{0.005373(0.047083)}{0.038579 - 0.047083} = 0.56701 \qquad |\varepsilon_t| = 0.0236\%$$

Third iteration:

$$x_0 = 0.56701 \qquad f(x_0) = 0.000209$$

$$x_0 + \delta x_0 = 0.572680 \qquad f(x_0 + \delta x_0) = -0.00867$$

$$x_1 = 0.56701 - \frac{0.00567(0.000209)}{-0.00867 - 0.000209} = 0.567143 \qquad |\varepsilon_t| = 2.365 \times 10^{-5}\%$$



PROBLEMS

6.1 Use simple fixed-point iteration to locate the root of

$$f(x) = 2\sin(\sqrt{x}) - x$$

Use an initial guess of $x_0 = 0.5$ and iterate until $\varepsilon_a \le 0.001\%$. Verify that the process is linearly convergent as described in Box 6.1.

6.2 Determine the highest real root of

$$f(x) = 2x^3 - 11.7x^2 + 17.7x - 5$$

(a) Graphically.

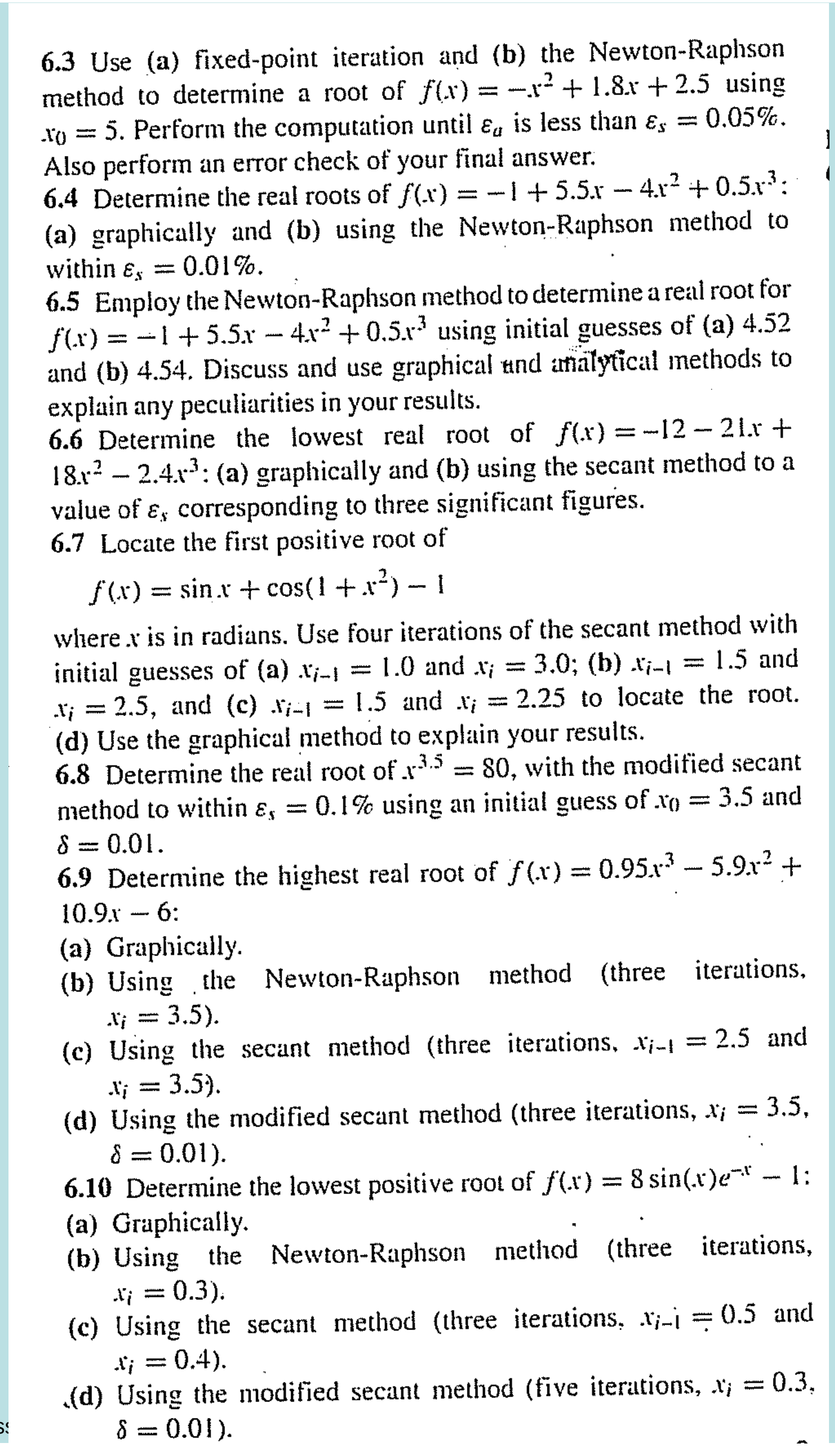
(b) Fixed-point iteration method (three iterations, $x_0 = 3$). Note: Make certain that you develop a solution that converges on the root.

(c) Newton-Raphson method (three iterations, $x_0 = 3$).

(d) Secant method (three iterations, $x_{-1} = 3$, $x_0 = 4$).

(e) Modified secant method (three iterations, $x_0 = 3$, $\delta = 0.01$).

Compute the approximate percent relative errors for your solutions.



6.3 Use **(a)** fixed-point iteration and **(b)** the Newton-Raphson method to determine a root of $f(x) = -x^2 + 1.8x + 2.5$ using $x_0 = 5$. Perform the computation until ε_a is less than $\varepsilon_s = 0.05\%$. Also perform an error check of your final answer.

6.4 Determine the real roots of $f(x) = -1 + 5.5x - 4x^2 + 0.5x^3$: **(a)** graphically and **(b)** using the Newton-Raphson method to within $\varepsilon_s = 0.01\%$.

6.5 Employ the Newton-Raphson method to determine a real root for $f(x) = -1 + 5.5x - 4x^2 + 0.5x^3$ using initial guesses of **(a)** 4.52 and **(b)** 4.54. Discuss and use graphical and analytical methods to explain any peculiarities in your results.

6.6 Determine the lowest real root of $f(x) = -12 - 21x + 18x^2 - 2.4x^3$: **(a)** graphically and **(b)** using the secant method to a value of ε_s corresponding to three significant figures.

6.7 Locate the first positive root of

$$f(x) = \sin x + \cos(1 + x^2) - 1$$

where x is in radians. Use four iterations of the secant method with initial guesses of **(a)** $x_{i-1} = 1.0$ and $x_i = 3.0$; **(b)** $x_{i-1} = 1.5$ and $x_i = 2.5$, and **(c)** $x_{i-1} = 1.5$ and $x_i = 2.25$ to locate the root. **(d)** Use the graphical method to explain your results.

6.8 Determine the real root of $x^{3.5} = 80$, with the modified secant method to within $\varepsilon_s = 0.1\%$ using an initial guess of $x_0 = 3.5$ and $\delta = 0.01$.

6.9 Determine the highest real root of $f(x) = 0.95x^3 - 5.9x^2 + 10.9x - 6$:

(a) Graphically.

(b) Using the Newton-Raphson method (three iterations, $x_i = 3.5$).

(c) Using the secant method (three iterations, $x_{i-1} = 2.5$ and $x_i = 3.5$).

(d) Using the modified secant method (three iterations, $x_i = 3.5$, $\delta = 0.01$).

6.10 Determine the lowest positive root of $f(x) = 8\sin(x)e^{-x} - 1$:

(a) Graphically.

(b) Using the Newton-Raphson method (three iterations, $x_i = 0.3$).

(c) Using the secant method (three iterations, $x_{i-1} = 0.5$ and $x_i = 0.4$).

(d) Using the modified secant method (five iterations, $x_i = 0.3$, $\delta = 0.01$).

Chapter 6

Roots of Polynomials

Chapter 7

- The roots of polynomials such as

$$f_n(x) = a_o + a_1 x + a_2 x^2 + \ldots + a_n x^n$$

Follow these rules:

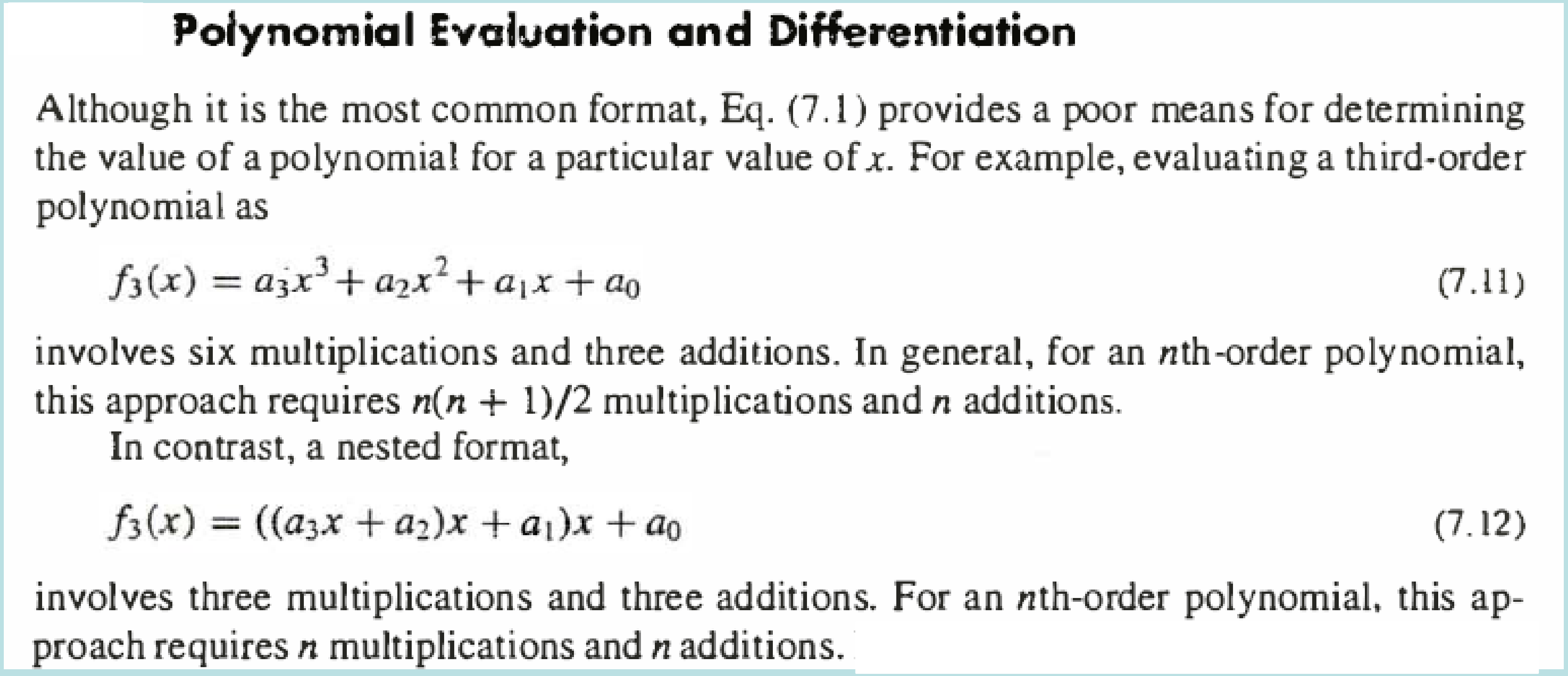
1. For an *n*th order equation, there are n real or complex roots.

2. If *n* is odd, there is at least one real root.

3. If complex root exist in conjugate pairs (that is, $\lambda+\mu i$ and $\lambda-\mu i$), where $i=sqrt(-1)$.

Conventional Methods

- The efficacy of bracketing and open methods depends on whether the problem being solved involves complex roots. If only real roots exist, these methods could be used. However,
  - Finding good initial guesses complicates both the open and bracketing methods, also the open methods could be susceptible to divergence.
- Special methods have been developed to find the real and complex roots of polynomials – Müller and Bairstow methods.



Polynomial Evaluation and Differentiation

Although it is the most common format, Eq. (7.1) provides a poor means for determining the value of a polynomial for a particular value of x. For example, evaluating a third-order polynomial as

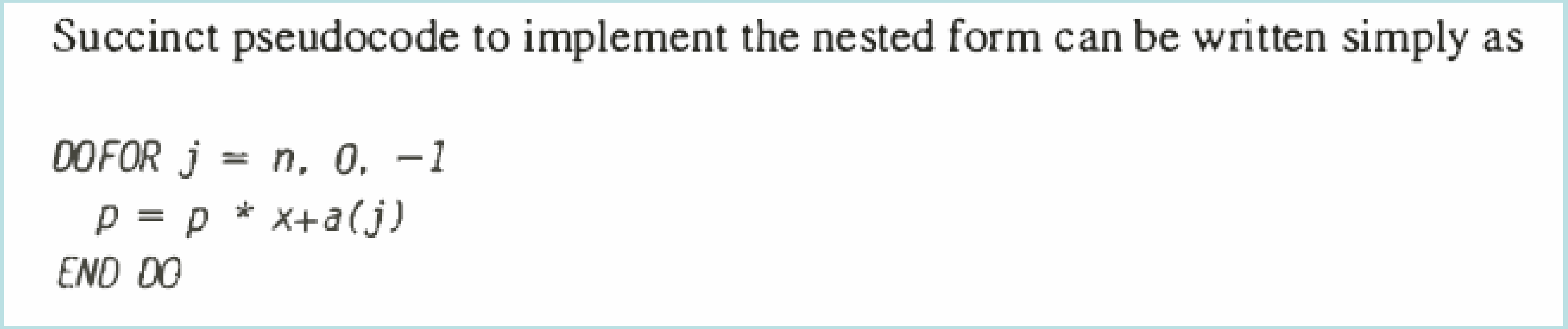
$$f_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0 \tag{7.11}$$

involves six multiplications and three additions. In general, for an nth-order polynomial, this approach requires $n(n+1)/2$ multiplications and n additions.

In contrast, a nested format,

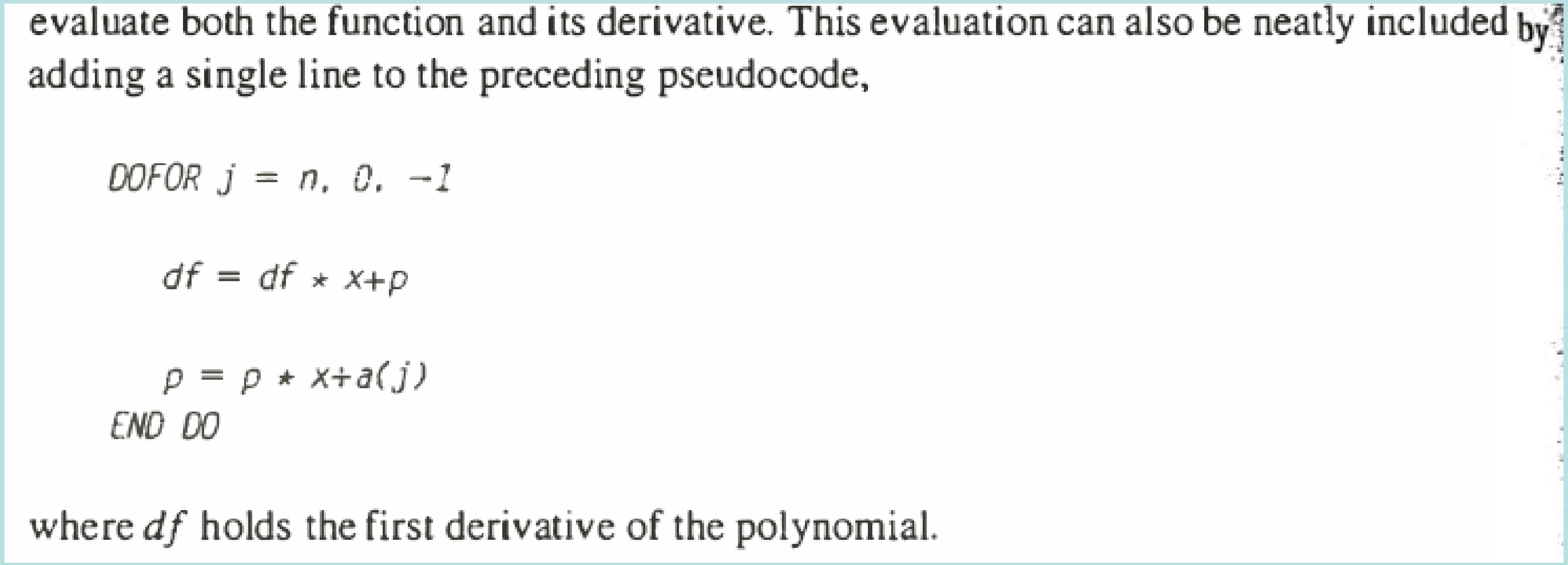
$$f_3(x) = ((a_3x + a_2)x + a_1)x + a_0 \tag{7.12}$$

involves three multiplications and three additions. For an nth-order polynomial, this approach requires n multiplications and n additions.



Succinct pseudocode to implement the nested form can be written simply as

```
DOFOR j = n, 0, -1
  p = p * x+a(j)
END DO
```



evaluate both the function and its derivative. This evaluation can also be neatly included by adding a single line to the preceding pseudocode,

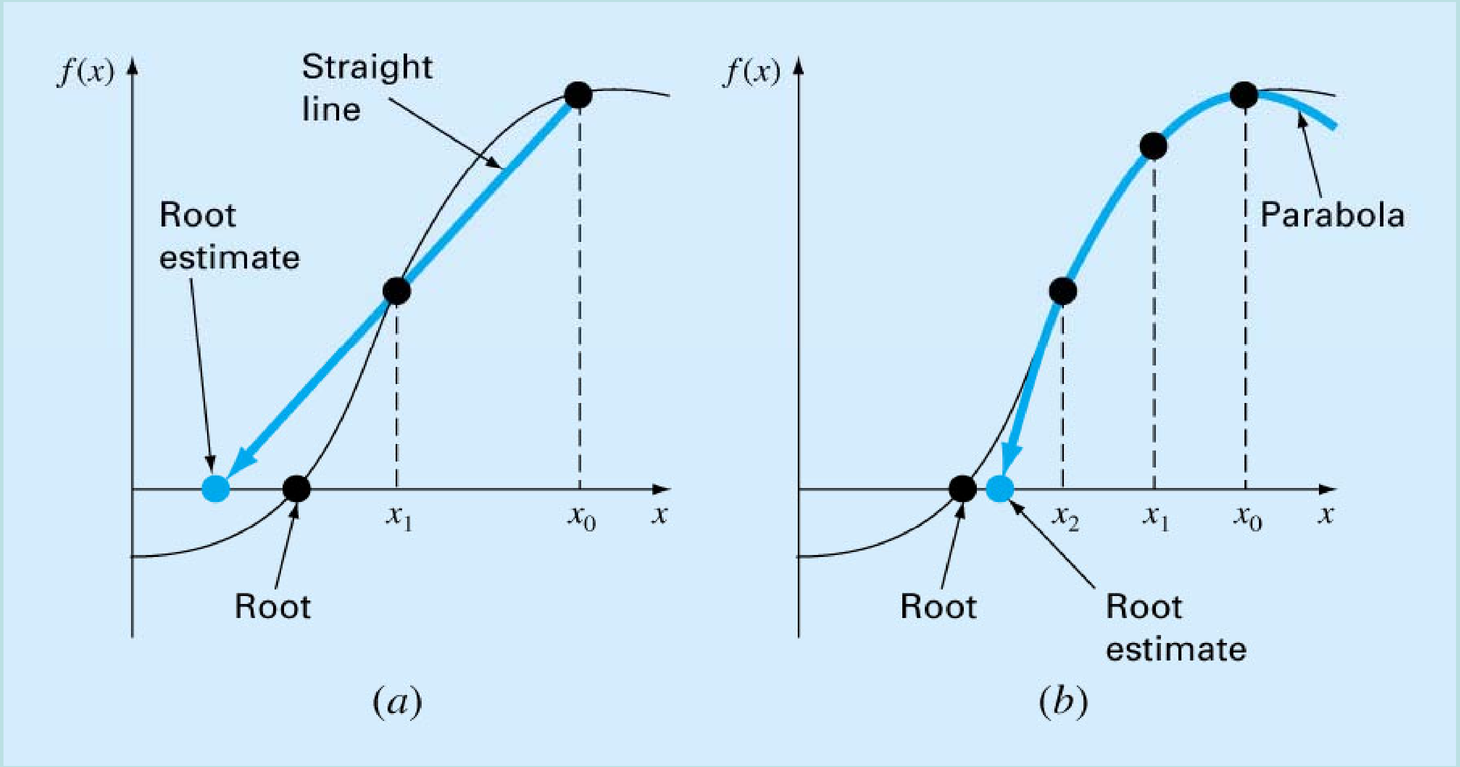
```
DOFOR j = n, 0, -1
   df = df * x+p
   p = p * x+a(j)
END DO
```

where *df* holds the first derivative of the polynomial.

Müller Method

- Müller's method obtains a root estimate by projecting a parabola to the x axis through three function values.

Figure 7.3



Müller Method

- The method consists of deriving the coefficients of parabola that goes through the three points:

1. Write the equation in a convenient form:

$$f_2(x) = a(x - x_2)^2 + b(x - x_2) + c$$

2. The parabola should intersect the three points *[x_o, $f(x_o)$], [x_1, $f(x_1)$], [x_2, $f(x_2)$].* The coefficients of the polynomial can be estimated by substituting three points to give

$$f(x_o) = a(x_o - x_2)^2 + b(x_o - x_2) + c$$
$$f(x_1) = a(x_1 - x_2)^2 + b(x_1 - x_2) + c$$
$$f(x_2) = a(x_2 - x_2)^2 + b(x_2 - x_2) + c$$

3. Three equations can be solved for three unknowns, *a, b, c*. Since two of the terms in the 3rd equation are zero, it can be immediately solved for $c=f(x_2)$.

$$f(x_o) - f(x_2) = a(x_o - x_2)^2 + b(x_o - x_2)$$
$$f(x_1) - f(x_2) = a(x_1 - x_2)^2 + b(x_1 - x_2)$$

If

$$h_o = x_1 - x_o \qquad h_1 = x_2 - x_1$$

$$\delta_o = \frac{f(x_1) - f(x_o)}{x_1 - x_o} \qquad \delta_1 = \frac{f(x_2) - f(x_1)}{x_2 - x_1}$$

$$\left.\begin{array}{l} (h_o + h_1)b - (h_o + h_1)^2 a = h_o\delta_o + h_1\delta_1 \\ h_1 b - h_1^2 a = h_1\delta_1 \end{array}\right\}$$

Solved for a and *b*

$$a = \frac{\delta_1 - \delta_o}{h_1 + h_o} \qquad b = ah_1 + \delta_1 \qquad c = f(x_2)$$

- Roots can be found by applying an alternative form of quadratic formula:

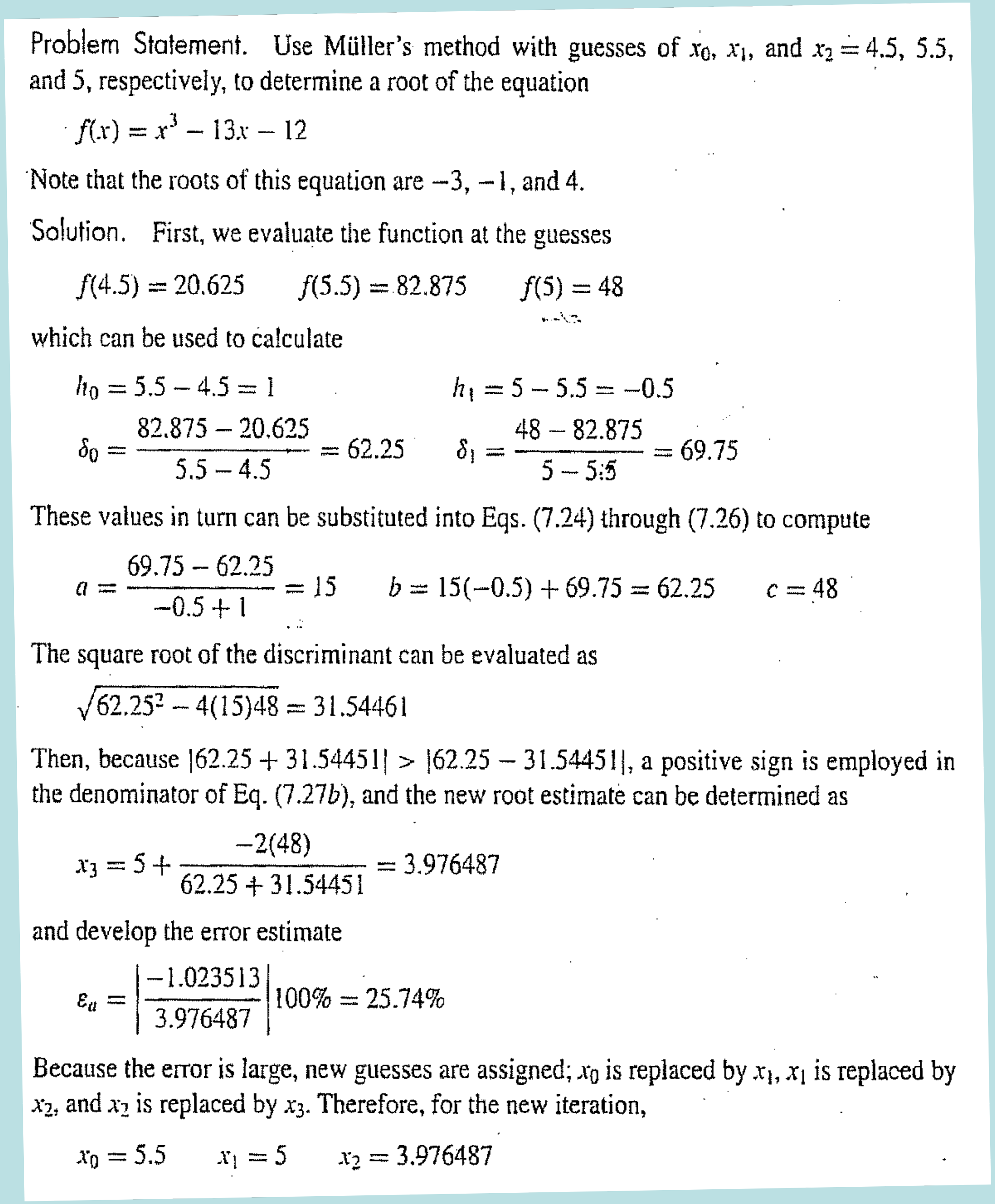
$$x_3 = x_2 + \frac{-2c}{b \pm \sqrt{b^2 - 4ac}}$$

- The error can be calculated as

$$\varepsilon_a = \left| \frac{x_3 - x_2}{x_3} \right| 100\%$$

- ±term yields two roots, the sign is chosen to agree with *b*. This will result in a largest denominator, and will give root estimate that is closest to x_2.

- Once x_3 is determined, the process is repeated using the following guidelines:
  1. If only real roots are being located, choose the two original points that are nearest the new root estimate, x_3.
  2. If both real and complex roots are estimated, employ a sequential approach just like in secant method, x_1, x_2, and x_3 to replace x_o, x_1, and x_2.



Problem Statement. Use Müller's method with guesses of x_0, x_1, and $x_2 = 4.5$, 5.5, and 5, respectively, to determine a root of the equation

$$f(x) = x^3 - 13x - 12$$

Note that the roots of this equation are −3, −1, and 4.

Solution. First, we evaluate the function at the guesses

$$f(4.5) = 20.625 \qquad f(5.5) = 82.875 \qquad f(5) = 48$$

which can be used to calculate

$$h_0 = 5.5 - 4.5 = 1 \qquad h_1 = 5 - 5.5 = -0.5$$

$$\delta_0 = \frac{82.875 - 20.625}{5.5 - 4.5} = 62.25 \qquad \delta_1 = \frac{48 - 82.875}{5 - 5.5} = 69.75$$

These values in turn can be substituted into Eqs. (7.24) through (7.26) to compute

$$a = \frac{69.75 - 62.25}{-0.5 + 1} = 15 \qquad b = 15(-0.5) + 69.75 = 62.25 \qquad c = 48$$

The square root of the discriminant can be evaluated as

$$\sqrt{62.25^2 - 4(15)48} = 31.54461$$

Then, because $|62.25 + 31.54451| > |62.25 - 31.54451|$, a positive sign is employed in the denominator of Eq. (7.27*b*), and the new root estimate can be determined as

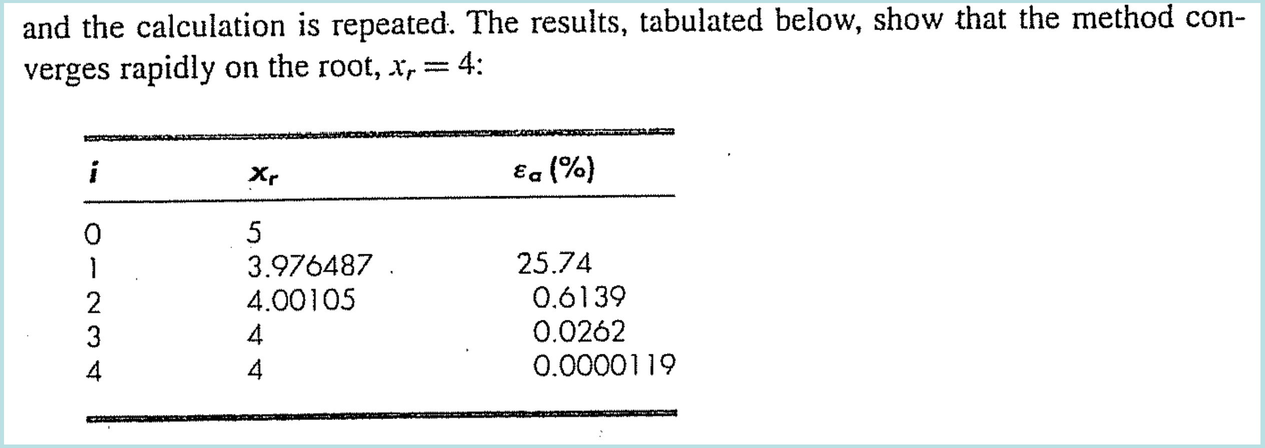
$$x_3 = 5 + \frac{-2(48)}{62.25 + 31.54451} = 3.976487$$

and develop the error estimate

$$\varepsilon_a = \left|\frac{-1.023513}{3.976487}\right| 100\% = 25.74\%$$

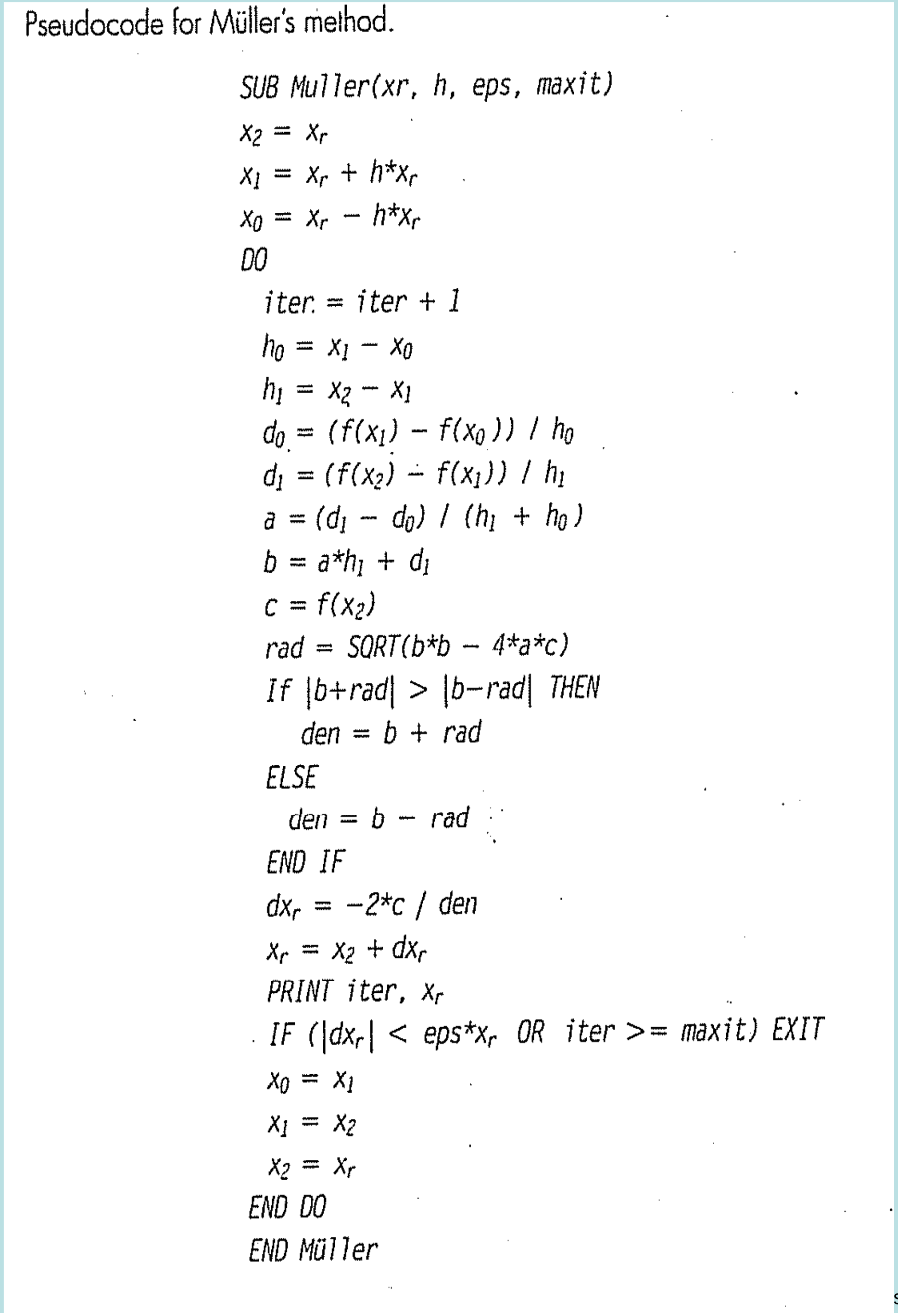
Because the error is large, new guesses are assigned; x_0 is replaced by x_1, x_1 is replaced by x_2, and x_2 is replaced by x_3. Therefore, for the new iteration,

$$x_0 = 5.5 \qquad x_1 = 5 \qquad x_2 = 3.976487$$



and the calculation is repeated. The results, tabulated below, show that the method converges rapidly on the root, $x_r = 4$:

i	x_r	ε_a (%)
0	5	
1	3.976487	25.74
2	4.00105	0.6139
3	4	0.0262
4	4	0.0000119

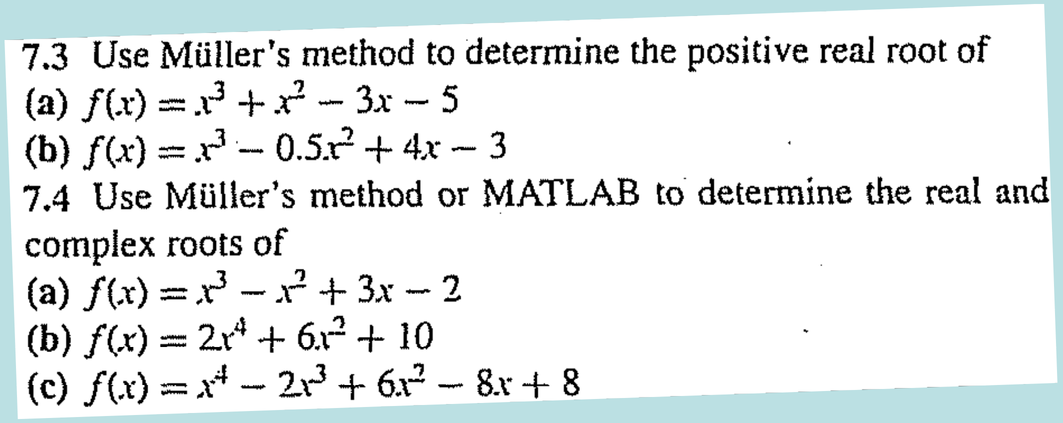


Pseudocode for Müller's method.

```
SUB Muller(xr, h, eps, maxit)
$x_2 = x_r$
$x_1 = x_r + h*x_r$
$x_0 = x_r - h*x_r$
DO
  iter = iter + 1
  $h_0 = x_1 - x_0$
  $h_1 = x_2 - x_1$
  $d_0 = (f(x_1) - f(x_0)) / h_0$
  $d_1 = (f(x_2) - f(x_1)) / h_1$
  $a = (d_1 - d_0) / (h_1 + h_0)$
  $b = a*h_1 + d_1$
  $c = f(x_2)$
  rad = SQRT(b*b - 4*a*c)
  If |b+rad| > |b-rad| THEN
     den = b + rad
  ELSE
     den = b - rad
  END IF
  $dx_r = -2*c / den$
  $x_r = x_2 + dx_r$
  PRINT iter, $x_r$
  IF ($|dx_r| < eps*x_r$ OR iter >= maxit) EXIT
  $x_0 = x_1$
  $x_1 = x_2$
  $x_2 = x_r$
END DO
END Müller
```

splay.

Tarea



7.3 Use Müller's method to determine the positive real root of

(a) $f(x) = x^3 + x^2 - 3x - 5$

(b) $f(x) = x^3 - 0.5x^2 + 4x - 3$

7.4 Use Müller's method or MATLAB to determine the real and complex roots of

(a) $f(x) = x^3 - x^2 + 3x - 2$

(b) $f(x) = 2x^4 + 6x^2 + 10$

(c) $f(x) = x^4 - 2x^3 + 6x^2 - 8x + 8$

Polinomial evaluation and Differenciation

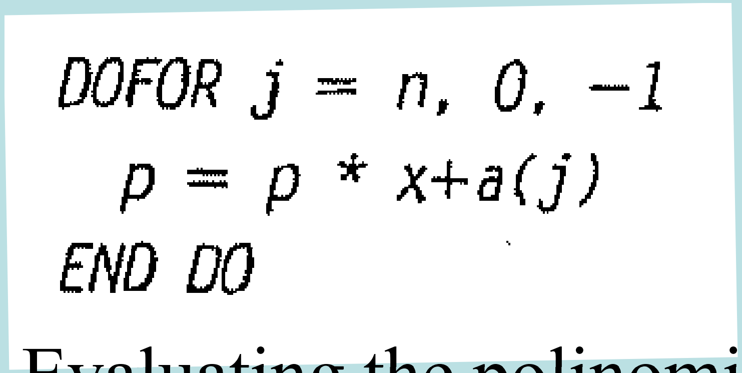
$$f_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

six multiplications and three additions.

$$f_3(x) = ((a_3x + a_2)x + a_1)x + a_0$$

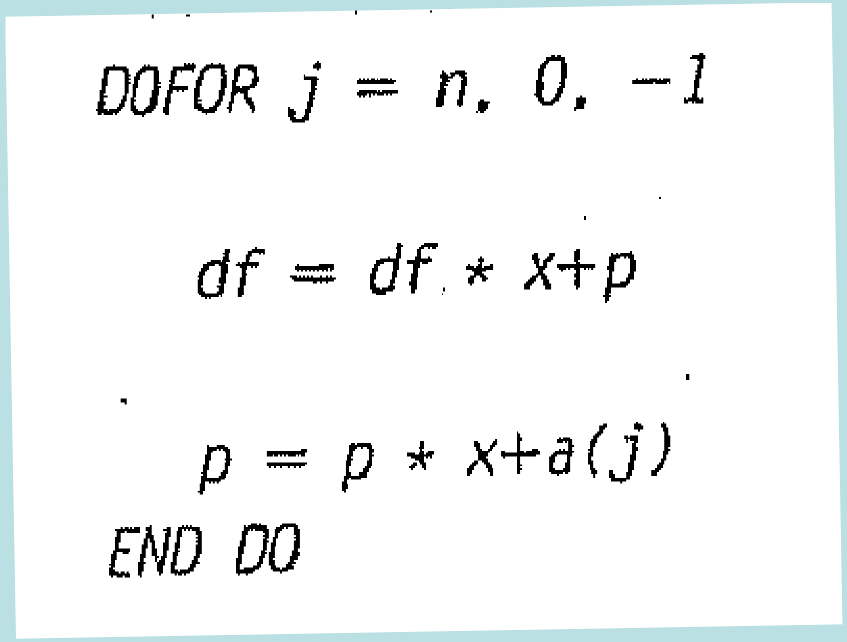
three multiplications and three additions.

Evaluating the polinomial



```
DOFOR j = n, 0, -1
  p = p * x+a(j)
END DO
```

Evaluating the polinomial and its derivative



```
DOFOR j = n, 0, -1
    df = df * x+p
    p = p * x+a(j)
END DO
```

Polynomial deflation

This is to eliminate the found root from the polynomial

$$f_5(x) = -120 - 46x + 79x^2 - 3x^3 - 7x^4 + x^5$$

Forma factorizada

$$f_5(x) = (x + 1)(x - 4)(x - 5)(x + 3)(x - 2)$$

Roots

Si se divide el polinomio entre cualquiera de sus factores, el resultado serà un polinomio de grado 4 con un residuo igual a cero.

Seudocódigo

```
r= a(n)
a(n)=0
DOFOR i = n-1,0,-1
        s=a(i)
        a(i)=r
        R=s+(r*t)
ENDDO
```

Divide un polinomio de n-ésimo grado entre un monomial x-t

Ejemplo:
f(x)=(x-4)(x+6)=x^2+2x-24

Dividiento entre x-4

Usando los parametros de suedocódigo:
n=2,a0=-24,a1=2,a2=1 y t=4

Entonces r=a2=1
a2=0

Iterando en el loop desde i=2-1 hasta 0:

Para i=1

s=a1=2

a1=r=1

r=s+rt=2+1(4)=6

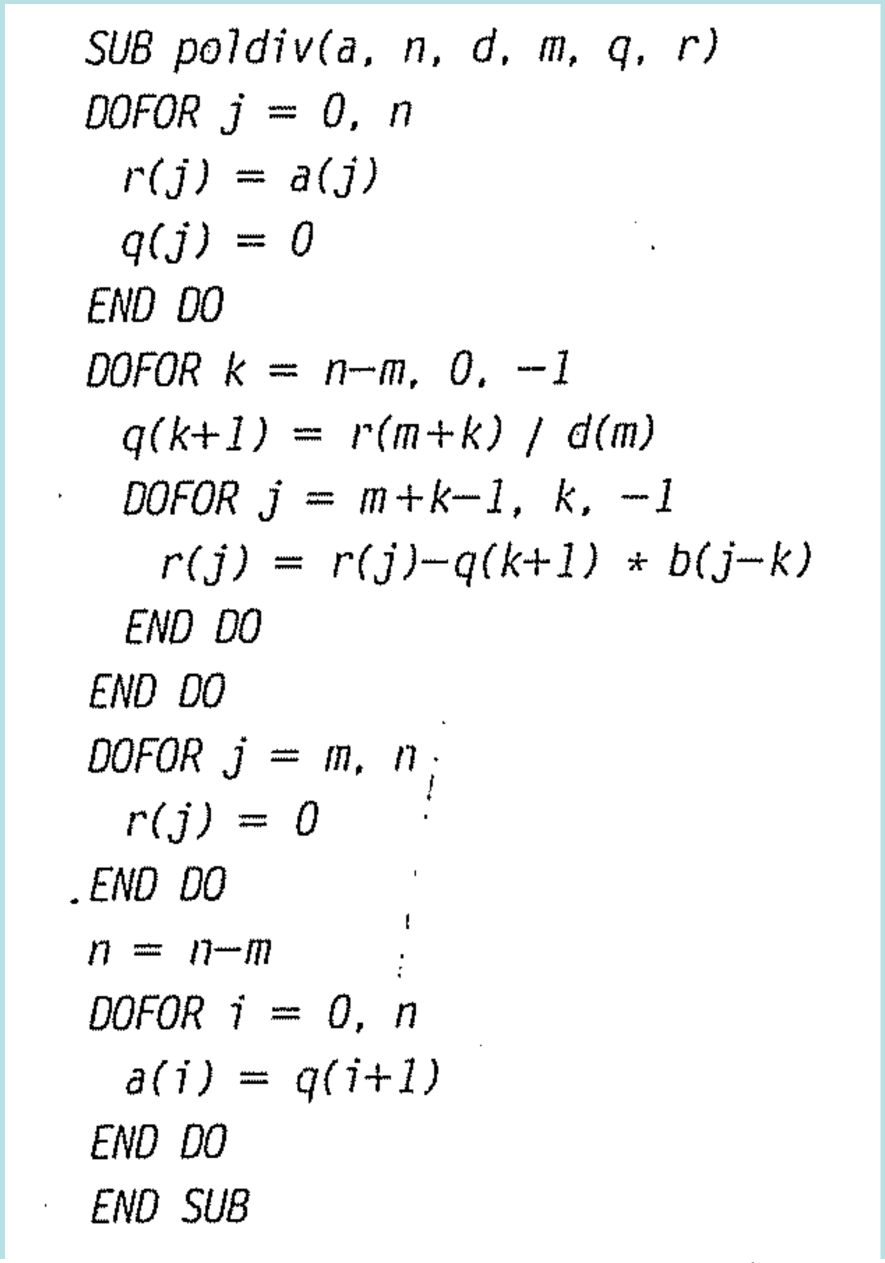
Para i = 0,

s=a0=-24

a0=r=6

r=-24+6(4)=0

Entonces, a0+a1x=6+x



```
SUB poldiv(a, n, d, m, q, r)
DOFOR j = 0, n
  r(j) = a(j)
  q(j) = 0
END DO
DOFOR k = n-m, 0, -1
  q(k+1) = r(m+k) / d(m)
  DOFOR j = m+k-1, k, -1
    r(j) = r(j)-q(k+1) * b(j-k)
  END DO
END DO
DOFOR j = m, n
  r(j) = 0
END DO
n = n-m
DOFOR i = 0, n
  a(i) = q(i+1)
END DO
END SUB
```

Algorithm to divide a polynomial (defined by its coefficients *a*) by a lower-order polynomial *d*.