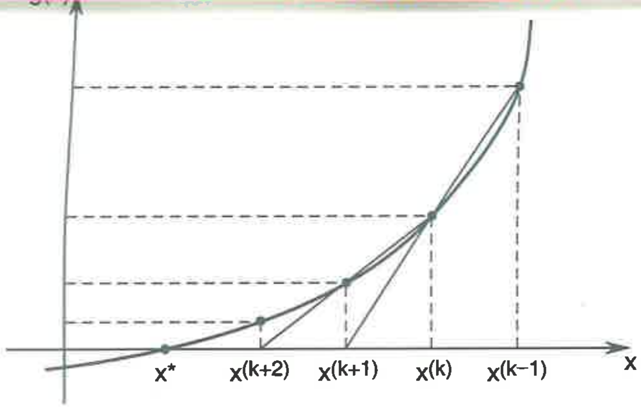

Figure 7.10 Secant method for root finding

of R that minimizes the total squared error between the measured voltages and the voltages predicted by the model.

We derive an algorithm to find the best estimate of R using the secant method. The objective function is:

$$f(R) = \sum_{i=1}^{n} (V_i - e^{-Rt_i})^2.$$

Hence, we have

$$f'(R) = 2\sum_{i=1}^{n} (V_i - e^{-Rt_i})e^{-Rt_i}t_i.$$

The secant algorithm for the problem is:

$$\begin{aligned} R_{k+1} &= R_k - \left(\frac{R_k - R_{k-1}}{\sum_{i=1}^{n}(V_i - e^{-R_k t_i})e^{-R_k t_i}t_i - (V_i - e^{-R_{k-1}t_i})e^{-R_{k-1}t_i}t_i} \right) \\ &\quad \times \sum_{i=1}^{n} (V_i - e^{-R_k t_i})e^{-R_k t_i}t_i. \end{aligned}$$

■

For further reading on the secant method, see [20].

7.5 REMARKS ON LINE SEARCH METHODS

One-dimensional search methods play an important role in multidimensional optimization problems. In particular, iterative algorithms for solving such optimization at every iteration. To be specific, let $f : \mathbb{R}^n \to \mathbb{R}$ be a function that we wish to minimize. Iterative algorithms for finding a minimizer of f are of the form

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)},$$

where $\boldsymbol{x}^{(0)}$ is a given initial point, and $\alpha_k \geq 0$ is chosen to minimize $\phi_k(\alpha) = f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)})$. The vector $\boldsymbol{d}^{(k)}$ is called the search direction. Note that choice of α_k involves a one-dimensional minimization. This choice ensures that, under appropriate conditions,

$$f(\boldsymbol{x}^{(k+1)}) < f(\boldsymbol{x}^{(k)}).$$

We may, for example, use the secant method to find α_k. In this case, we need the derivative of ϕ_k, which is

$$\phi_k'(\alpha) = \boldsymbol{d}^{(k)T} \nabla f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}).$$

The above is obtained using the chain rule. Therefore, applying the secant method for the line search requires the gradient ∇f, the initial line search point $\boldsymbol{x}^{(k)}$, and the search direction $\boldsymbol{d}^{(k)}$ (see Exercise 7.9). Of course, other one-dimensional search methods may be used for line search (see, e.g., [29] and [64]).

Line search algorithms used in practice are much more involved than the one-dimensional search methods presented in this chapter. The reason for this stems from several practical considerations. First, determining the value of α_k that exactly minimizes ϕ_k may be computationally demanding; even worse, the minimizer of ϕ_k may not even exist. Second, practical experience suggests that it is better to allocate more computational time on iterating the optimization algorithm rather than performing exact line searches. These considerations led to the development of conditions for terminating line search algorithms that would result in low-accuracy line searches while still securing a decrease in the value of the f from one iteration to the next. For more information on practical line search methods, we refer the reader to [29, pp. 26–40], [34], and [35][1].

EXERCISES

7.1 Suppose that we have a unimodal function over the interval $[5, 8]$. Give an example of a desired final uncertainty range where the Golden Section method requires at least 4 iterations, whereas the Fibonacci method requires only 3. You may choose an arbitrarily small value of ε for the Fibonacci method.

7.2 Let $f(x) = x^2 + 4\cos x$, $x \in \mathbb{R}$. We wish to find the minimizer x^* of f over the interval $[1, 2]$. (*Calculator users:* Note that in $\cos x$, the argument x is in radians).

[1]We thank Dennis M. Goodman for furnishing us with references [34] and [35].

a. Plot $f(x)$ versus x over the interval $[1, 2]$.

b. Use the Golden Section method to locate x^* to within an uncertainty of 0.2. Display all intermediate steps using a table as follows:

Iteration k	a_k	b_k	$f(a_k)$	$f(b_k)$	New uncertainty interval
1	?	?	?	?	[?,?]
2	?	?	?	?	[?,?]
⋮	⋮	⋮	⋮	⋮	⋮

c. Repeat part b using the Fibonacci method, with $\varepsilon = 0.05$. Display all intermediate steps using a table as follows:

Iteration k	ρ_k	a_k	b_k	$f(a_k)$	$f(b_k)$	New uncertainty interval
1	?	?	?	?	?	[?,?]
2	?	?	?	?	?	[?,?]
⋮	⋮	⋮	⋮	⋮	⋮	⋮

d. Apply Newton's method, using the same number of iterations as in part b, with $x^{(0)} = 1$.

7.3 Let $f(x) = 8e^{1-x} + 7\log(x)$, where $\log(\cdot)$ represents the natural logarithm function.

a. Use MATLAB to plot $f(x)$ versus x over the interval $[1, 2]$, and verify that f is unimodal over $[1, 2]$.

b. Write a simple MATLAB routine to implement the Golden Section method that locates the minimizer of f over $[1, 2]$ to within an uncertainty of 0.23. Display all intermediate steps using a table as in Exercise 7.2.

c. Repeat part b using the Fibonacci method, with $\varepsilon = 0.05$. Display all intermediate steps using a table as in Exercise 7.2.

7.4 Suppose that $\rho_1, \ldots, \rho_N$ are the values used in the Fibonacci search method. Show that for each $k = 1, \ldots, N$, $0 \le \rho_k \le 1/2$, and for each $k = 1, \ldots, N-1$,

$$\rho_{k+1} = 1 - \frac{\rho_k}{1-\rho_k}.$$

7.5 Show that if $F_0, F_1, \ldots$ is the Fibonacci sequence, then for each $k = 2, 3, \ldots$,

$$F_{k-2}F_{k+1} - F_{k-1}F_k = (-1)^k.$$

7.6 Show that the Fibonacci sequence can be calculated using the formula

$$F_n = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{n+1} - \left(\frac{1-\sqrt{5}}{2}\right)^{n+1}\right).$$

7.7 Suppose that we have an efficient way of calculating exponentials. Based on this, use Newton's method to devise a method to approximate $\log(2)$ (where $\log(\cdot)$ is the natural logarithm function). Use an initial point of $x^{(0)} = 1$, and perform 2 iterations.

7.8 The objective of this exercise is to implement the secant method using MATLAB.

a. Write a simple MATLAB routine to implement the secant method to locate the root of the equation $g(x) = 0$. For the stopping criterion, use the condition $|x^{(k+1)} - x^{(k)}| < |x^{(k)}|\varepsilon$, where $\varepsilon > 0$ is a given constant.

b. Let $g(x) = (2x-1)^2 + 4(4 - 1024x)^4$. Find the root of $g(x) = 0$ using the secant method with $x^{(-1)} = 0$, $x^{(0)} = 1$, and $\varepsilon = 10^{-5}$. Also determine the value of g at the obtained solution.

7.9 Write a MATLAB function that implements a line search algorithm using the secant method. The arguments to this function are the name of the M-file for the gradient, the current point, and the search direction. For example, the function may be called `linesearch_secant`, and used by the function call `alpha=linesearch_secant('grad',x,d)`, where `grad.m` is the M-file containing the gradient, `x` is the starting line search point, `d` is the search direction, and `alpha` is the value returned by the function (which we use in the following chapters as the step size for iterative algorithms (see, e.g., Exercises 8.18, 10.8)).

Note: In the solutions manual, we used the stopping criterion $|\boldsymbol{d}^T\nabla f(\boldsymbol{x}+\alpha\boldsymbol{d})| \le \varepsilon|\boldsymbol{d}^T\nabla f(\boldsymbol{x})|$, where $\varepsilon > 0$ is a prespecified number, ∇f is the gradient, $\boldsymbol{x}$ is the starting line search point, and $\boldsymbol{d}$ is the search direction. The rationale for the above stopping criterion is that we want to reduce the directional derivative of f in the direction $\boldsymbol{d}$ by the specified fraction ε. We used a value of $\varepsilon = 10^{-4}$, and initial conditions of 0 and 0.001.