5

OPTIMIZATION OF UNCONSTRAINED FUNCTIONS: ONE-DIMENSIONAL SEARCH

A GOOD TECHNIQUE for the optimization of a function of just one variable is essential for two reasons:

1. Some unconstrained problems inherently involve only one variable
2. Techniques for unconstrained and constrained optimization problems generally involve repeated use of a one-dimensional search as described in Chapters 6 and 8.

Prior to the advent of high-speed computers, methods of optimization were limited primarily to analytical methods, that is, methods of calculating a potential extremum were based on using the necessary conditions and analytical derivatives as well as values of the objective function. Modern computers have made possible iterative, or numerical, methods that search for an extremum by using function and sometimes derivative values of $f(\mathbf{x})$ at a sequence of trial points $\mathbf{x}^1, \mathbf{x}^2, \ldots$.

As an example consider the following function of a single variable x (see Figure 5.1).

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

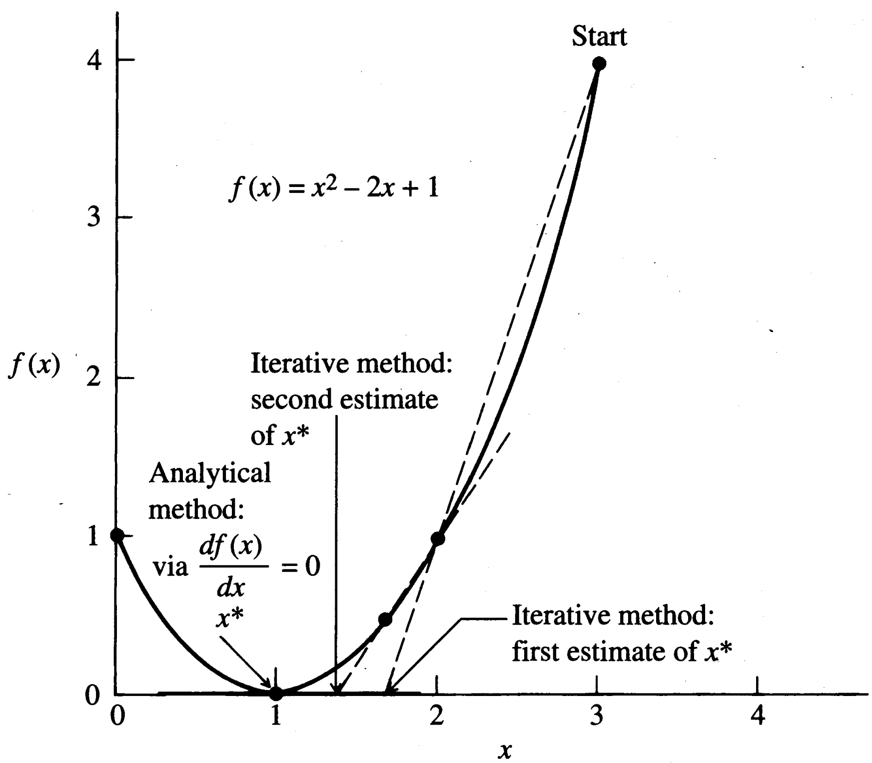


FIGURE 5.1
Iterative versus analytical methods of finding a minimum.

An analytical method of finding x^* at the minimum of $f(x)$ is to set the gradient of $f(x)$ equal to zero

$$\frac{df(x)}{dx} = 0 = 2x - 2$$

and solve the resulting equation to get $x^* = 1$; x^* can be tested for the sufficient conditions to ascertain that it is indeed a minimum:

$$\frac{d^2f(1)}{dx^2} = 2 > 0$$

To carry out an iterative method of numerical minimization, start with some initial value of x, say $x^0 = 0$, and calculate successive values of $f(x) = x^2 - 2x + 1$ and possibly df/dx for other values of x, values selected according to whatever strategy is to be employed. A number of different strategies are discussed in subsequent sections of this chapter. Stop when $f(x^{k+1}) - f(x^k) < \varepsilon_1$ or when

$$\left.\frac{df}{dx}\right|_{x^k} < \varepsilon_2$$

where the superscript k designates the iteration number and ε_1 and ε_2 are the prespecified tolerances or criteria of precision.

If $f(x)$ has a simple closed-form expression, analytical methods yield an exact solution, a closed form expression for the optimal x, x^*. If $f(x)$ is more complex, for example, if it requires several steps to compute, then a numerical approach must be used. Software for nonlinear optimization is now so widely available that the numerical approach is almost always used. For example, the "Solver" in the Microsoft Excel spreadsheet solves linear and nonlinear optimization problems, and many FORTRAN and C optimizers are available as well. General optimization software is discussed in Section 8.9.

Analytical methods are usually difficult to apply for nonlinear objective functions with more than one variable. For example, suppose that the nonlinear function $f(\mathbf{x}) = f(x_1, x_2, \ldots, x_n)$ is to be minimized. The necessary conditions to be used are

$$\frac{\partial f(\mathbf{x})}{\partial x_1} = 0$$

$$\frac{\partial f(\mathbf{x})}{\partial x_2} = 0$$

$$\vdots$$

$$\frac{\partial f(\mathbf{x})}{\partial x_n} = 0$$

Each of the partial derivatives when equated to zero may well yield a nonlinear equation. Hence, the minimization of $f(\mathbf{x})$ is converted into a problem of solving a set of nonlinear equations in n variables, a problem that can be just as difficult to solve as the original problem. Thus, most engineers prefer to attack the minimization problem directly by one of the numerical methods described in Chapter 6, rather than to use an indirect method. Even when minimizing a function of one variable by an indirect method, using the necessary conditions can lead to having to find the real roots of a nonlinear equation.

5.1 NUMERICAL METHODS FOR OPTIMIZING A FUNCTION OF ONE VARIABLE

Most algorithms for unconstrained and constrained optimization make use of an efficient unidimensional optimization technique to locate a local minimum of a function of one variable. Nash and Soter (1996) and other general optimization books (e.g., Dennis and Schnabel, 1983) have reviewed one-dimensional search techniques that calculate the interval in which the minimum of a function lies. To apply these methods you initially need to know an initial bracket Δ^0 that contains the minimum of the objective function $f(x)$, and that $f(x)$ is unimodal in the interval. This can be done by coding the function in a spreadsheet or in a programming language like Visual Basic, Fortran, or C, choosing an interval, and evaluating $f(x)$ at a grid of points in that interval. The interval is extended if the minimum is at an end point. There are various methods of varying the initial interval to reach a final interval Δ^n. In the next section we describe a few of the methods that prove to be the most effective in practice.

One method of optimization for a function of a single variable is to set up as fine a grid as you wish for the values of x and calculate the function value for every point on the grid. An approximation to the optimum is the best value of $f(x)$. Although this is not a very efficient method for finding the optimum, it can yield acceptable results. On the other hand, if we were to utilize this approach in optimizing a multivariable function of more than, say, five variables, the computer time is quite likely to become prohibitive, and the accuracy is usually not satisfactory.

In selecting a search method to minimize or maximize a function of a single variable, the most important concerns are software availability, ease of use, and efficiency. Sometimes the function may take a long time to compute, and then efficiency becomes more important. For example, in some problems a simulation may be required to generate the function values, such as in determining the optimal number of trays in a distillation column. In other cases you have no functional description of the physical–chemical model of the process to be optimized and are forced to operate the process at various input levels to evaluate the value of the process output. The generation of a new value of the objective function in such circumstances may be extremely costly, and no doubt the number of plant tests would be limited and have to be quite judiciously designed. In such circumstances, efficiency is a key criterion in selecting a minimization strategy.

5.2 SCANNING AND BRACKETING PROCEDURES

Some unidimensional search procedures require that a bracket of the minimum be obtained as the first part of the strategy, and then the bracket is narrowed. Along with the statement of the objective function $f(x)$ there must be some statement of bounds on x or else the implicit assumption that x is unbounded $(-\infty < x < \infty)$. For example, the problem

$$\text{Minimize: } f(x) = (x - 100)^2$$

has an optimal value of $x^* = 100$. Clearly you would not want to start at $-\infty$ (i.e., a large negative number) and try to bracket the minimum. Common sense suggests estimating the minimum x and setting up a sufficiently wide bracket to contain the true minimum. Clearly, if you make a mistake and set up a bracket of $0 \leq x \leq 10$, you will find that the minimum occurs at one of the bounds, hence the bracket must be revised. In engineering and scientific work physical limits on temperature, pressure, concentration, and other physically meaningful variables place practical bounds on the region of search that might be used as an initial bracket.

Several strategies exist for scanning the independent variable space and determining an acceptable range for search for the minimum of $f(x)$. As an example, in the above function, if we discretize the independent variable by a grid spacing of 0.01, and then initiate the search at zero, proceeding with consecutively higher values of x, much time and effort would be consumed in order to set up the initial bracket for x. Therefore, acceleration procedures are used to scan rapidly for a suitable range of x. One technique might involve using a functional transformation (e.g., $\log x$) in order to look at wide ranges of the independent variable. Another method might be to use a variable grid spacing. Consider a sequence in x given by the following formula:

$$x^{k+1} = x^k + \delta \cdot 2^{k-1} \tag{5.1}$$

Equation (5.1) allows for successively wider-spaced values, given some base increment (delta). Table 5.1 lists the values of x and $f(x) = (x - 100)^2$ for Equation (5.1) with $\delta = 1$.Note that in nine calculations we have bounded the minimum of $f(x)$. Another scanning procedure could be initiated between $x = 63$ and $x = 255$, with δ reduced, and so on to find the minimum of $f(x)$. However, more efficient techniques are discussed in subsequent sections of this chapter.

In optimization of a function of a single variable, we recognize (as for general multivariable problems) that there is no substitute for a good first guess for the starting point in the search. Insight into the problem as well as previous experience

TABLE 5.1
Acceleration in fixing an initial bracket

x	0	1	3	7	15	31	63	127	255
$f(x)$	10^4	9801	9409	8649	7225	4761	1369	729	2325

are therefore often very important factors influencing the amount of time and effort required to solve a given optimization problem.

The methods considered in the rest of this chapter are generally termed descent methods for minimization because a given step is pursued only if it yields an improved value for the objective function. First we cover methods that use function values or first or second derivatives in Section 5.3, followed by a review of several methods that use only function values in Section 5.4.

5.3 NEWTON AND QUASI-NEWTON METHODS OF UNIDIMENSIONAL SEARCH

Three basic procedures for finding an extremum of a function of one variable have evolved from applying the necessary optimality conditions to the function:

1. Newton's method
2. Finite difference approximation of Newton's method
3. Quasi-Newton methods

In comparing the effectiveness of these techniques, it is useful to examine the rate of convergence for each method. Rates of convergence can be expressed in various ways, but a common classification is as follows:[a]

Linear

$$\frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^{k} - \mathbf{x}^*\|} \leq c \qquad 0 \leq c < 1, k \text{ large} \tag{5.2}$$

(rate usually slow in practice)

Order p

$$\frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^{k} - \mathbf{x}^*\|^p} \leq c \qquad c \geq 0, p \geq 1, k \text{ large} \tag{5.3}$$

(rate fastest in practice if $p > 1$)

If $p = 2$, the order of convergence is said to be quadratic.

To understand these definitions, assume that the algorithm generating the sequence of points $\mathbf{x}^k$ is converging to $\mathbf{x}^*$, that is, as $k \rightarrow \infty$, if Equation (5.2) holds for large k, $\mathbf{x}^k \rightarrow \mathbf{x}^*$. Then

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \leq c\,\|\mathbf{x}^{k} - \mathbf{x}^*\| \qquad k \text{ large}$$

[a]The symbols $\mathbf{x}^k$, $\mathbf{x}^{k+1}$, and so on refer to the kth or $(k + 1)$st stage of iteration and not to powers of $\mathbf{x}$.

so the error at iteration $k + 1$ is bounded by c times the error at iteration k, where $c < 1$. If $c = 0.1$, then the error is reduced by a factor of 10 at each iteration, at least for the later iterations. The constant c is called the convergence ratio.

If Equation (5.3) holds for large k, then $\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \leq c\,\|\mathbf{x}^k - \mathbf{x}\|^p$, k large enough. If $p = 2$, and $\|\mathbf{x}^k - \mathbf{x}^0\| = 10^{-1}$ for some k, then

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \leq c \cdot 10^{-2}$$

$$\|\mathbf{x}^{k+2} - \mathbf{x}^*\| \leq c^2 \cdot 10^{-4}$$

$$\|\mathbf{x}^{k+3} - \mathbf{x}^*\| \leq c^3 \cdot 10^{-6}$$

and so on.

Hence, if c is around 1.0, the error decreases very rapidly, the number of correct digits in $\mathbf{x}^k$ doubling with each iteration. Because all real numbers in double precision arithmetic have about 16 significant decimal digits, only a few iterations are needed before the limits of accuracy of Equation (5.3) are reached.

Superlinear

$$\lim_{k\to\infty} \frac{\|\mathbf{x}^{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}^k - \mathbf{x}^*\|} \to 0 \quad (\text{or} < c_k \text{ and } c_k \to 0 \text{ as } k \to \infty) \tag{5.4}$$

(rate usually fast in practice)

For a function of a single variable $\|\mathbf{x}\| = |x|$ itself.

5.3.1 Newton's Method

Recall that the first-order necessary condition for a local minimum is $f'(x) = 0$. Consequently, you can solve the equation $f'(x) = 0$ by Newton's method to get

$$x^{k+1} = x^k - \frac{f'(x^k)}{f''(x^k)} \tag{5.5}$$

making sure on each stage k that $f(x^{k+1}) < f(x^k)$ for a minimum. Examine Figure 5.2.

To see what Newton's method implies about $f(x)$, suppose $f(x)$ is approximated by a quadratic function at x^k

$$f(x) = f(x^k) + f'(x^k)(x - x^k) + \tfrac{1}{2}f''(x^k)(x - x^k)^2 \tag{5.6}$$

Find $df(x)/dx = 0$, a stationary point of the quadratic model of the function. The result obtained by differentiating Equation (5.6) with respect to x is

$$f'(x^k) + (\tfrac{1}{2})(2)f''(x^k)(x - x^k) = 0 \tag{5.7}$$

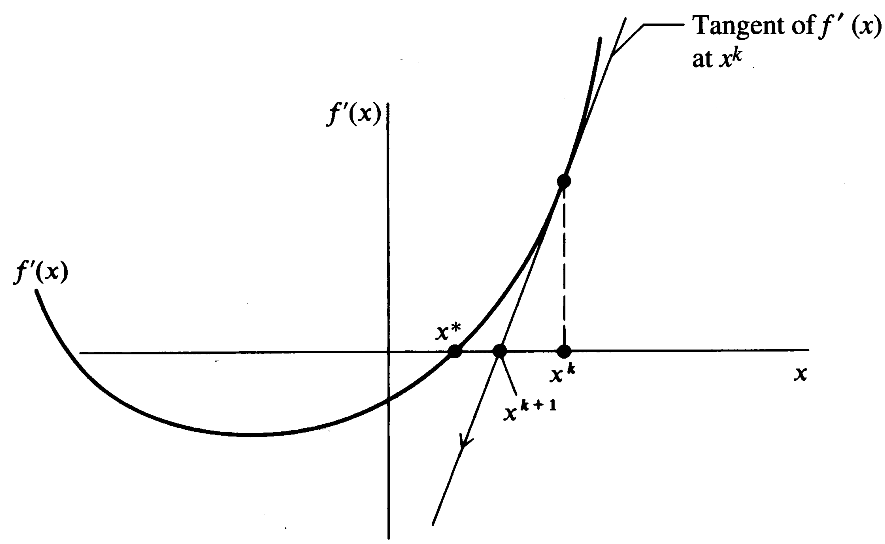


FIGURE 5.2
Newton's method applied to the solution of $f'(x) = 0$.

which can be rearranged to yield Equation (5.5). Consequently, Newton's method is equivalent to using a quadratic model for a function in minimization (or maximization) and applying the necessary conditions.

The advantages of Newton's method are

1. The procedure is locally quadratically convergent [$p = 2$ in Equation (5.3)] to the extremum as long as $f''(x) \neq 0$.
2. For a quadratic function, the minimum is obtained in one iteration.

The disadvantages of the method are

1. You have to calculate both $f'(x)$ and $f''(x)$.
2. If $f''(x) \to 0$, the method converges slowly.
3. If the initial point is not close enough to the minimum, the method as described earlier will not converge. Modified versions that guarantee convergence from poor starting points are described in Bazarra et al. (1993) and Nash and Sofer (1996).

5.3.2 Finite Difference Approximations to Derivatives

If $f(x)$ is not given by a formula, or the formula is so complicated that analytical derivatives cannot be formulated, you can replace Equation (5.5) with a finite difference approximation

$$x^{k+1} = x^k - \frac{[f(x+h) - f(x-h)]/2h}{[f(x+h) - 2f(x) + f(x-h)]/h^2} \tag{5.8}$$

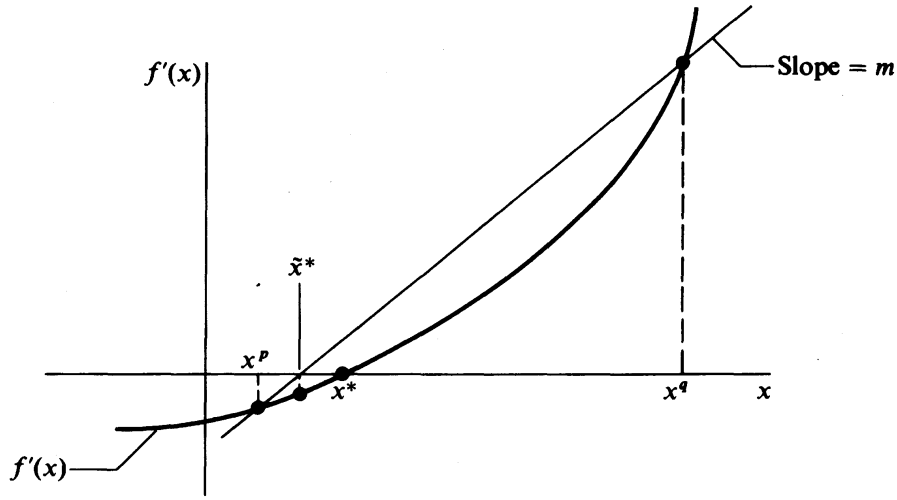


FIGURE 5.3
Quasi-Newton method for solution of $f'(x) = 0$.

Central differences were used in Equation (5.8), but forward differences or any other difference scheme would suffice as long as the step size h is selected to match the difference formula and the computer (machine) precision with which the calculations are to be executed. The main disadvantage is the error introduced by the finite differencing.

5.3.3 Quasi-Newton Method

In the quasi-Newton method (secant method) the approximate model analogous to Equation (5.7) to be solved is

$$f'(x^k) + m(x - x^k) = 0 \tag{5.9}$$

where m is the slope of the line connecting the point x^p and a second point x^q, given by

$$m = \frac{f'(x^q) - f'(x^p)}{x^q - x^p}$$

The quasi-Newton approximates $f'(x)$ as a straight line (examine Figure 5.3); as $x^q \rightarrow x^p$, m approaches the second derivative of $f(x)$. Thus Equation (5.9) imitates Newton's method

$$\tilde{x} = x^q - \frac{f'(x^q)}{[f'(x^q) - f'(x^p)]/(x^q - x^p)} \tag{5.10}$$

where $\tilde{x}$ is the approximation to x^* achieved on one iteration k. Note that $f'(x)$ can itself be approximated by finite differencing.

Quasi-Newton methods start out by using two points x^p and x^q spanning the interval of x, points at which the first derivatives of $f(x)$ are of opposite sign. The zero of Equation (5.9) is predicted by Equation (5.10), and the derivative of the function is then evaluated at the new point. The two points retained for the next step are $\tilde{x}$ and either x^q or x^p. This choice is made so that the pair of derivatives $f'(\tilde{x})$, and either $f'(x^p)$ or $f'(x^q)$, have opposite signs to maintain the bracket on x^*. This variation is called "regula falsi" or the method of false position. In Figure 5.3, for the $(k + 1)$st search, $\tilde{x}$ and x^q would be selected as the end points of the secant line.

Quasi-Newton methods may seem crude, but they work well in practice. The order of convergence is $(1 + \sqrt{5})/2 \approx 1.6$ for a single variable. Their convergence is slightly slower than a properly chosen finite difference Newton method, but they are usually more efficient in terms of total function evaluations to achieve a specified accuracy (see Dennis and Schnabel, 1983, Chapter 2).

For any of the three procedures outlined in this section, in minimization you assume the function is unimodal, bracket the minimum, pick a starting point, apply the iteration formula to get x^{k+1} (or $\tilde{x}$) from x^k (or x^p and x^q), and make sure that $f(x^{k+1}) < f(x^k)$ on each iteration so that progress is made toward the minimum. As long as $f''(x^k)$ or its approximation is positive, $f(x)$ decreases.

Of course, you must start in the correct direction to reduce $f(x)$ (for a minimum) by testing an initial perturbation in x. For maximization, minimize $-f(x)$.

EXAMPLE 5.1 COMPARISON OF NEWTON, FINITE DIFFERENCE NEWTON, AND QUASI-NEWTON METHODS APPLIED TO A QUADRATIC FUNCTION

In this example, we minimize a simple quadratic function $f(x) = x^2 - x$ that is illustrated in Figure E5.1a using one iteration of each of the methods presented in Section 5.3.

Solution. By inspection we can pick a bracket on the minimum, say $x = -3$ to $x = 3$. Assume $x^0 = 3$ is the starting point for the minimization.

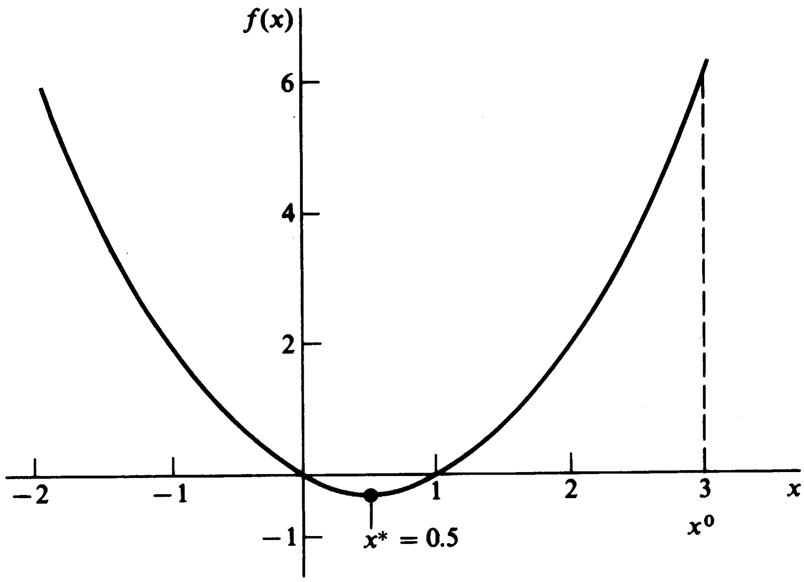
Newton's method. For Newton's method sequentially apply Equation (5.5). Examine Figure 5.1b for $f(x) = x^2 - x$ and $f'(x) = 2x - 1$; $f''(x) = 2$. Note $f''(x)$ is always positive-definite. For this example Equation (5.5) is

$$x^1 = x^0 - \frac{f'(x^0)}{f''(x^0)} \tag{a}$$

and

$$x^1 = 3 - \tfrac{5}{2} = 0.5$$

Because the function is quadratic and hence $f'(x)$ is linear, the minimum is obtained in one step. If the function were not quadratic, then additional iterations using Equation (5.5) would take place.



FIIGURE E5.1a

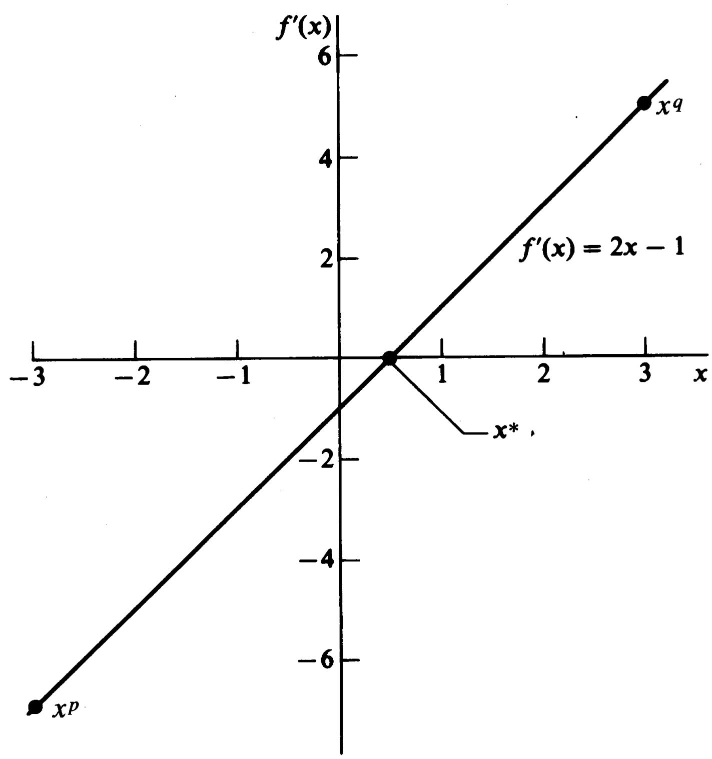


FIGURE E5.1b

Finite difference Newton method. Application of Equation (5.8) to $f(x) = x^2 - x$ is illustrated here. However, we use a forward difference formula for $f'(x)$ and a three-point central difference formula for $f''(x)$

$$x^{k+1} = x^k - \frac{[f(x+h) - f(x)]/h}{[f(x+h) - 2f(x) + f(x-h)]/h^2} \tag{b}$$

with $h = 10^{-3}$:

$$x^1 = 3 - \frac{[f(3.001) - f(3.0)]/10^{-3}}{[f(3.001) - 2f(3.0) + f(2.999)]/(10^{-3})^2}$$

$$= 3 - (10^{-3})\frac{(6.005001 - 6.000000)}{(6.005001 - 12.000000 + 5.995001)}$$

$$= 3 - (10^{-3})\frac{0.005001}{0.000002} = 3 - 2.500500$$

$$= 0.499500$$

One more iteration could be taken to improve the estimate of x^*, perhaps with a smaller value of h (if desired).

Quasi-Newton method. The application of Equation (5.10) to $f(x) = x^2 - x$ starts with the two points $x = -3$ and $x = 3$ corresponding to the x^p and x^q, respectively, in Figure 5.3:

$$f'(-3) = -7 \qquad f'(3) = 5$$

$$x^1 = 3 - \frac{5}{[5 - (-7)]/[3 - (-3)]} = 3 - 2.5 = 0.5$$

As before, the optimum is reached in one step because $f'(x)$ is linear, and the linear extrapolation is valid.

EXAMPLE 5.2 MINIMIZING A MORE DIFFICULT FUNCTION

In this example we minimize a nonquadratic function $f(x) = x^4 - x + 1$ that is illustrated in Figure E5.2a, using the same three methods as in Example 5.1. For a starting point of $x = 3$, minimize $f(x)$ until the change in x is less than 10^{-7}. Use $h = 0.1$ for the finite-difference method. For the quasi-Newton method, use $x^q = 3$ and $x^p = -3$.

Solution

Newton's method. For Newton's method, $f' = 4x^3 - 1$ and $f'' = 12x^2$, and the sequence of steps is

$$x_1 = x_0 - \frac{4x_0^3 - 1}{12x_0^2} \qquad (a)$$

$$= 3 - \frac{107}{108} = 2.009259$$

$$x_2 = 2.00926 - \frac{31.4465}{48.4454} = 1.36015$$

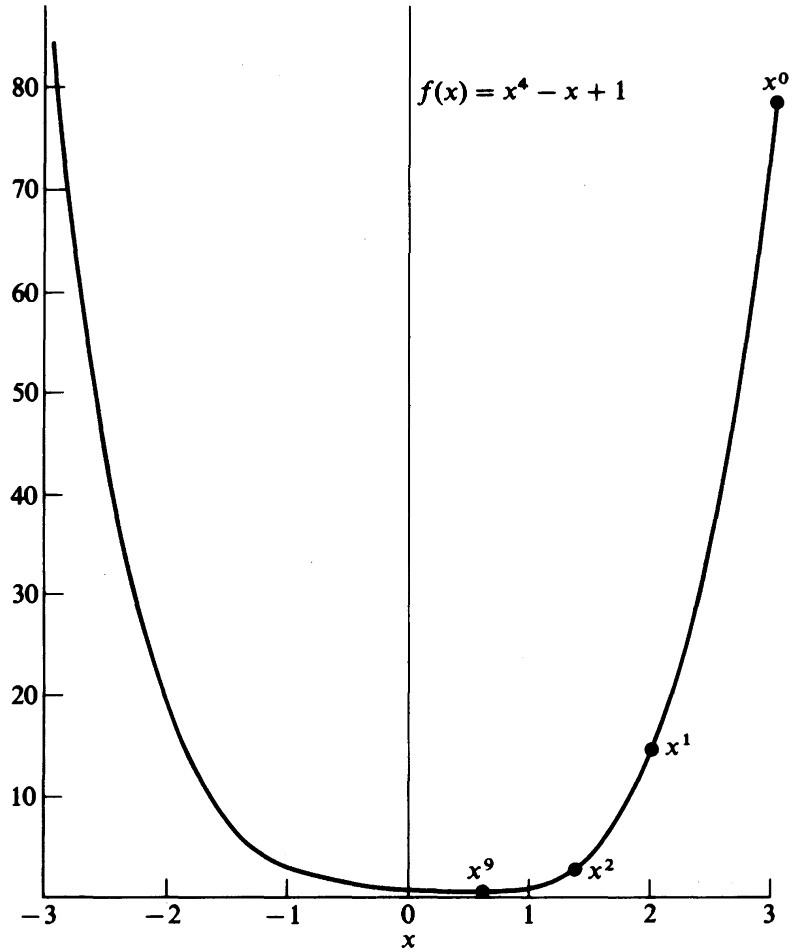


FIGURE E5.2a
Newton iterates for fourth order function.

Additional iterations yield the following values for x:

k	x^k	$\dfrac{x^{k+1} - x^*}{x^k - x^*}$	$\dfrac{x^{k+1} - x^*}{\lvert x^k - x^* \rvert^2}$
0	3.00000		
1	2.009259	0.582	0.246
2	1.3601480	0.529	0.384
3	0.9518103	0.441	0.604
4	0.7265254	0.300	0.932
5	0.6422266	0.127	1.315
6	0.6301933	0.019	1.547
7	0.6299606	0.000	1.587
8	0.6299605		
9	0.6299605		

As you can see from the third and fourth columns in the table the rate of convergence of Newton's method is superlinear (and in fact quadratic) for this function.

Finite Difference Newton. Equation (5.8) for this example is

$$x^{k+1} = x^k - \frac{h}{2} \frac{[f(x+h) - f(x-h)]}{[f(x+h) - 2f(x) + f(x-h)]} \tag{b}$$

For the same problem as used in Newton's method, the first iteration using (*b*) for $h = 10^{-4}$ is

$$x^1 = 3 - \left[\frac{10^{-4}}{2}\right] \frac{[f(3.0001) - f(2.9999)]}{[f(3.0001) - 2f(3.000) + f(2.999)]}$$

Other values of h give

	x^k		
k	$h = 0.10$	$h = 10^{-4}$	$h = 10^{-7}$
0	3.00000	2.00926	3.00000
1	2.00833	1.36015	2.21568
2	1.35816	0.951811	1.46785
3	0.948531	0.726526	0.955459
4	0.721882	0.642227	0.736528
5	0.636823	0.630193	0.642986
6	0.624849	0.629960	0.631846
7	0.624668	0.6299605191	0.630035
8	0.624669	....	0.629964
9	0.624669313	....	0.629961
10	....	....	0.629961
11	....	....	0.629960525

For $h = 10^{-8}$, the procedure diverged after the second iteration.

Quasi-Newton. The application of Equation (5.10) yields the following results (examine Figure E5.2b). Note how the shape of $f'(x)$ implies that a large number of iterations are needed to reach x^*. Some of the values of $f'(x)$ and x during the search are shown in the following table; notice that x^q remains unchanged in order to maintain the bracket with $f'(x) > 0$.

k	x^q	x^p	$f'(x^p)$
0	3.0	−3.0	−109.0000
1	3.0	0.0277	−0.9991
2	3.0	0.0552	−0.9992
3	3.0	0.0825	−0.9977
4	3.0	0.1094	−0.9899
5	3.0	0.1361	−0.9899
20	3.0	0.4593	−0.6124
50	3.0	0.6223	−0.0360
100	3.0	0.6299	-1.399×10^{-4}
132	3.0	0.6299	-3.952×10^{-6}

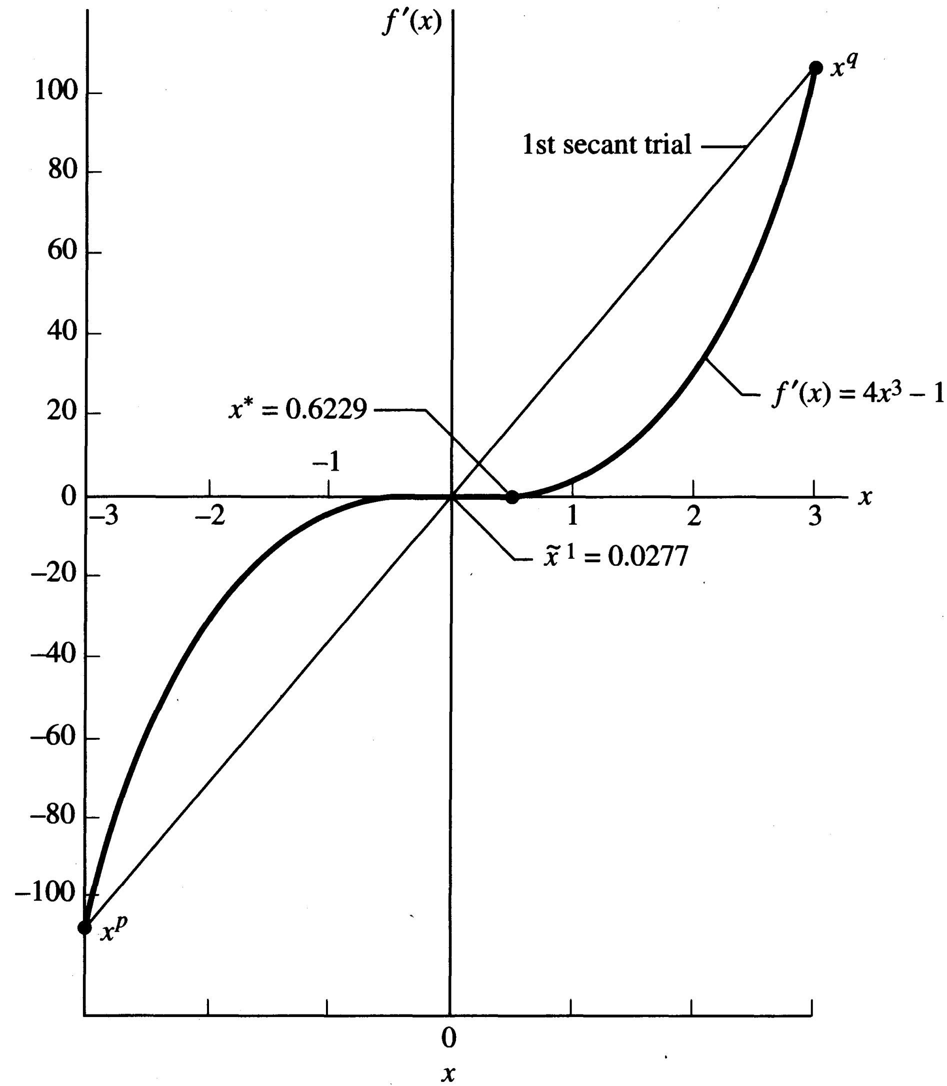


FIGURE E5.2b
Quazi-Newton method applied to $f'(x)$.

5.4 POLYNOMIAL APPROXIMATION METHODS

Another class of methods of unidimensional minimization locates a point x near x^*, the value of the independent variable corresponding to the minimum of $f(x)$, by extrapolation and interpolation using polynomial approximations as models of $f(x)$. Both quadratic and cubic approximation have been proposed using function values only and using both function and derivative values. In functions where $f'(x)$ is continuous, these methods are much more efficient than other methods and are now widely used to do line searches within multivariable optimizers.

5.4.1 Quadratic Interpolation

We start with three points x_1, x_2, and x_3 in increasing order that might be equally spaced, but the extreme points must bracket the minimum. From the analysis in Chapter 2, we know that a quadratic function $f(x) = a + bx + cx^2$ can be passed exactly through the three points, and that the function can be differentiated and the derivative set equal to 0 to yield the minimum of the approximating function

$$\tilde{x} = -\frac{b}{2c} \tag{5.11}$$

Suppose that $f(x)$ is evaluated at x_1, x_2, and x_3 to yield $f(x_1) \equiv f_1, f(x_2) \equiv f_2$, and $f(x_3) \equiv f_3$. The coefficients b and c can be evaluated from the solution of the three linear equations

$$f(x_1) = a + bx_1 + cx_1^2$$

$$f(x_2) = a + bx_2 + cx_2^2$$

$$f(x_3) = a + bx_3 + cx_3^2$$

via determinants or matrix algebra. Introduction of b and c expressed in terms of x_1, x_2, x_3, f_1, f_2, and f_3 into Equation (5.11) gives

$$\tilde{x}^* = \frac{1}{2}\left[\frac{(x_2^2 - x_3^2)f_1 + (x_3^2 - x_1^2)f_2 + (x_1^2 - x_2^2)f_3}{(x_2 - x_3)f_1 + (x_3 - x_1)f_2 + (x_1 - x_2)f_3}\right] \tag{5.12}$$

To illustrate the first stage in the search procedure, examine the four points in Figure 5.4 for stage 1. We want to reduce the initial interval $[x_1, x_3]$. By examining the values of $f(x)$ [with the assumptions that $f(x)$ is unimodal and has a minimum], we can discard the interval from x_1 to x_2 and use the region (x_2, x_3) as the new interval. The new interval contains three points, $(x_2, \tilde{x}, x_3)$ that can be introduced into Equation (5.12) to estimate a x^*, and so on. In general, you evaluate $f(x^*)$ and discard from the set $\{x_1, x_2, x_3\}$ the point that corresponds to the greatest value of $f(x)$, unless

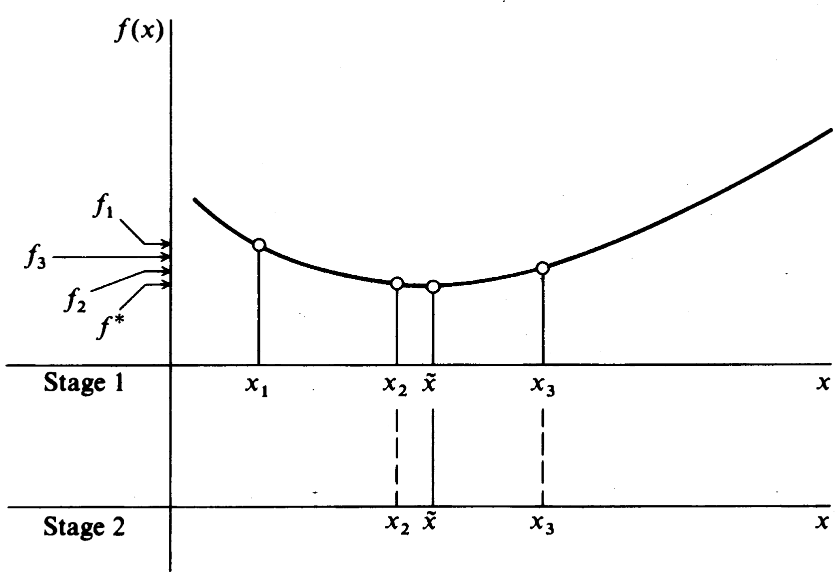
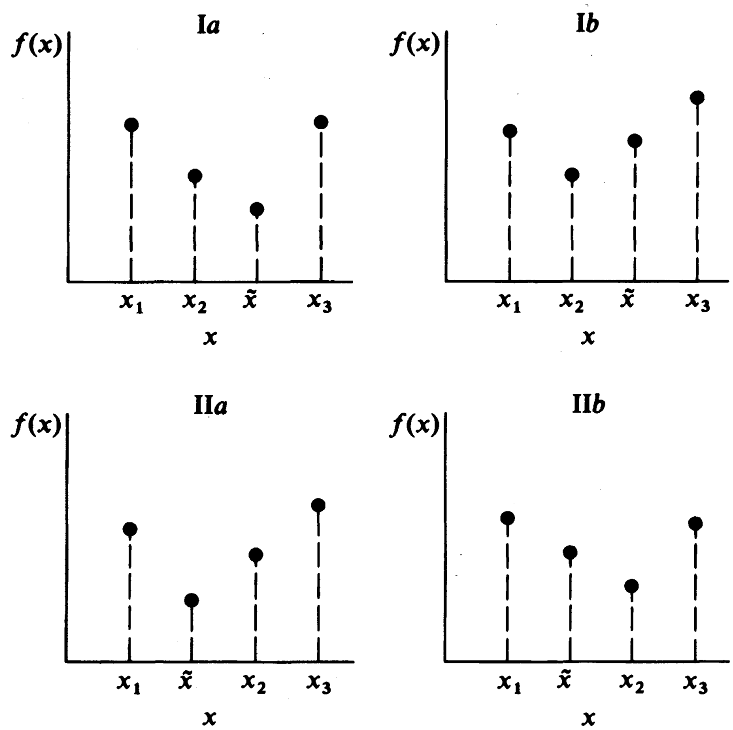


FIGURE 5.4
Two stages of quadratic interpolation.



I. If $\tilde{x}$ lies between x_2 and x_3:

(*a*) $f^* < f_2$, $f^* < f_3$ Pick $x_2, \tilde{x}, x_3$

(*b*) $f^* > f_2$, $f^* < f_3$ Pick $x_1, x_2, \tilde{x}$

II. If $\tilde{x}$ lies between x_1 and x_2:

(*a*) $f^* < f_2$, $f^* < f_1$ Pick $x_1, \tilde{x}, x_2$

(b) $f^* > f_2$, $f^* < f_1$ Pick $\tilde{x}, x_2, x_3$

FIGURE 5.5
How to maintain a bracket on the minimum in quadratic interpolation.

a bracket on the minimum of $f(x)$ is lost by so doing, in which case you discard the x so as to maintain the bracket. The specific tests and choices of x_i to maintain the bracket are illustrated in Figure 5.5. In Figure 5.5, $f^* \equiv f(\tilde{x})$. If x^* and whichever of $\{x_1, x_2, x_3\}$ corresponding to the smallest $f(x)$ differ by less than the prescribed accuracy in x, or the prescribed accuracy in the corresponding values of $f(x)$ is achieved, terminate the search. Note that only function evaluations are used in the search and that only one new function evaluation (for $\tilde{x}$) has to be carried out at each new iteration.

EXAMPLE 5.3 APPLICATION OF QUADRATIC INTERPOLATION

The function to be minimized is $f(x) = x^2 - x$ and is illustrated in Figure E5.1a. Three points bracketing the minimum $(-1.7, -0.1, 1.5)$ are used to start the search for the minimum of $f(x)$; we use equally spaced points here but that is not a requirement of the method.

Solution

$$x_1 = -1.7 \qquad x_2 = -0.1 \qquad x_3 = 1.5$$

$$f(x_1) = 4.59 \qquad f(x_2) = 0.11 \qquad f(x_3) = 0.75$$

$$\Delta x = 1.6$$

Two different formulas for quadratic interpolation can be compared: Equation (5.8), the finite difference method, and Equation (5.12).

$$\tilde{x}^* = x_2 - \frac{\Delta x[f(x_3) - f(x_1)]}{2[f(x_3) - 2f(x_2) + f(x_1)]} \tag{5.8}$$

$$= -0.1 - \frac{1.6(0.75 - 4.59)}{2(0.75 - 2(0.11) + 4.59)} = 0.50 \tag{a}$$

$$\tilde{x}^* = \frac{1}{2}\frac{[x_2^2 - x_3^2]f(x_1) + [x_3^2 - x_1^2]f(x_2) + [x_1^2 - x_2^2]f(x_3)}{(x_2 - x_3)f(x_1) + (x_3 - x_1)f(x_2) + (x_1 - x_2)f(x_3)} \tag{5.12}$$

$$= \frac{1}{2}\frac{\begin{array}{c}[(-0.1)^2 - (1.5)^2](4.59) + [(1.5)^2 - (-1.7)^2](0.11) \\ + [(-1.7)^2 - (-0.1)^2](0.75)\end{array}}{\begin{array}{c}[(-0.1) - (1.5)](4.59) + [(1.5) - (-1.7)](0.11) \\ + [(-1.7) - (-0.1)](0.75)\end{array}} \tag{b}$$

$$= 0.50$$

Note that a solution on the first iteration seems to be remarkable, but keep in mind that the function is quadratic so that quadratic interpolation should be good even if approximate formulas are used for derivatives.

5.4.2 Cubic Interpolation

Cubic interpolation to find the minimum of $f(x)$ is based on approximating the objective function by a third-degree polynomial within the interval of interest and then determining the associated stationary point of the polynomial

$$f(x) = a_1x^3 + a_2x^2 + a_3x + a_4$$

Four points must be computed (that bracket the minimum) to estimate the minimum, either four values of $f(x)$, or the values of $f(x)$ and the derivative of $f(x)$, each at two points.

In the former case four linear equations are obtained with the four unknowns being the desired coefficients. Let the matrix $\mathbf{X}$ be

$$\mathbf{X} = \begin{bmatrix} x_1^3 & x_1^2 & x_1 & 1 \\ x_2^3 & x_2^2 & x_2 & 1 \\ x_3^3 & x_3^2 & x_3 & 1 \\ x_4^3 & x_4^2 & x_4 & 1 \end{bmatrix}$$

$$\mathbf{F}^T = [f(x_1) \quad f(x_2) \quad f(x_3) \quad f(x_4)]$$

$$\mathbf{A}^T = [a_1 \quad a_2 \quad a_3 \quad a_4]$$

$$\mathbf{F} = \mathbf{XA} \tag{5.13}$$

Then the extremum of $f(x)$ is obtained by setting the derivative of $f(x)$ equal to zero and solving for $\tilde{x}$

$$\frac{df(x)}{dx} = 3a_1x^2 + 2a_2x + a_3 = 0$$

so that

$$\tilde{x} = \frac{-2a_2 \pm \sqrt{4a_2^2 - 12a_1a_3}}{6a_1} \tag{5.14}$$

The sign to use before the square root is governed by the sign of the second derivative of $f(\tilde{x})$, that is, whether a minimum or maximum is sought. The vector **A** can be computed from **XA** = **F** or

$$\mathbf{A} = \mathbf{X}^{-1}\mathbf{F} \tag{5.15}$$

After the optimum point $\tilde{x}$ is predicted, it is used as a new point in the next iteration and the point with the highest [lowest value of $f(x)$ for maximization] value of $f(x)$ is discarded.

If the first derivatives of $f(x)$ are available, only two points are needed, and the cubic function can be fitted to the two pairs of the slope and function values. These four pieces of information can be uniquely related to the four coefficients in the cubic equation, which can be optimized for predicting the new, nearly optimal data point. If (x_1, f_1, f'_1) and (x_2, f_2, f'_2) are available, then the optimum $\tilde{x}$ is

$$\tilde{x} = x_2 - \left[\frac{f'_2 + w - z}{f'_2 - f'_1 + 2w}\right](x_2 - x_1) \tag{5.18}$$

where $$z = \frac{3[f_1 - f_2]}{[x_2 - x_1]} + f_1' + f_2'$$

$$w = [z^2 - f'_1 \cdot f'_2]^{1/2}$$

In a minimization problem, you require $x_1 < x_2, f'_1 < 0$, and $f'_2 > 0$ (x_1 and x_2 bracket the minimum). For the new point $(\tilde{x})$, calculate $f'(\tilde{x})$ to determine which of the previous two points to replace. The application of this method in nonlinear programming algorithms that use gradient information is straightforward and effective.

If the function being minimized is not unimodal locally, as has been assumed to be true in the preceding discussion, extra logic must be added to the unidimensional search code to ensure that the step size is adjusted to the neighborhood of the local optimum actually sought. For example, Figure 5.6 illustrates how a large initial step can lead to an unbounded solution to a problem when, in fact, a local minimum is sought.

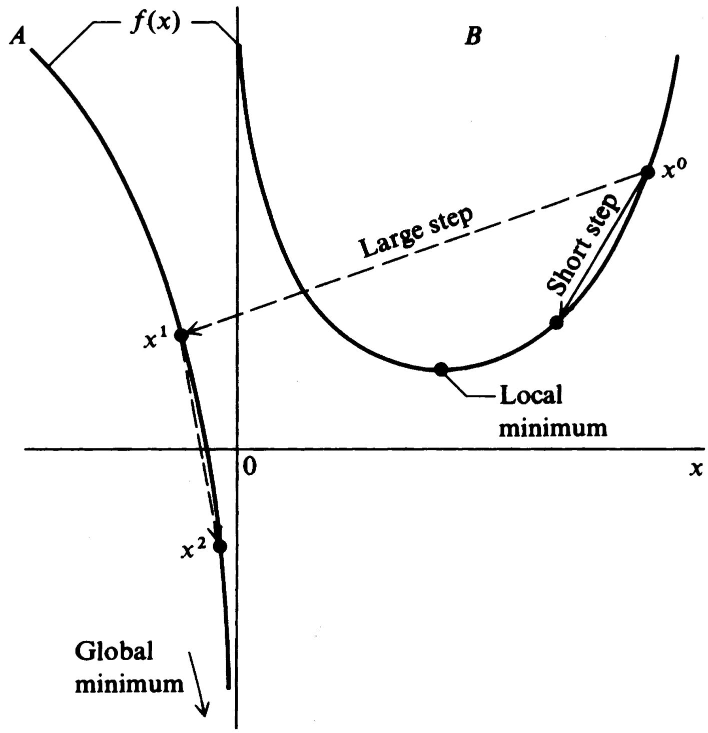


FIGURE 5.6
A unidimensional search for a local minimum of a multimodal objective function leads to an unbounded solution.

EXAMPLE 5.4 OPTIMIZATION OF A MICROELECTRONICS PRODUCTION LINE FOR LITHOGRAPHY

You are to optimize the thickness of resist used in a production lithographic process. There are a number of competing effects in lithography.

1. As the thickness t (measured in micrometers) grows smaller, the defect density grows larger. The number of defects per square centimeter of resist is given by

$$D_0 = 1.5t^{-3}$$

2. The chip yield in fraction of good chips for each layer is given by

$$\eta = \frac{1}{1 + \alpha D_0 a}$$

where a is the active area of the chip. Assume that 50 percent of the defects are "fatal" defects ($\alpha = 0.5$) detected after manufacturing the chip.

Assume four layers are required for the device. The overall yield is based on a series formula:

$$\eta = \frac{1}{(1 + \alpha D_0 a)^4}$$

3. Throughput decreases as resist thickness increases. A typical relationship is

$$V(\text{wafers/h}) = 125 - 50t + 5t^2$$

Each wafer has 100 chip sites with 0.25 cm^2 active area. The daily production level is to be 2500 finished wafers. Find the resist thickness to be used to maximize the number of good chips per hour. Assume $0.5 \le t \le 2.5$ as the expected range. First use cubic interpolation to find the optimal value of t, t^*. How many parallel production lines are required for t^*, assuming 20 h/day operation each? How many iterations are needed to reach the optimum if you use quadratic interpolation?

Solution. The objective function to be maximized is the number of good chips per hour, which is found by multiplying the yield, the throughput, and the number of chips per wafer (= 100):

$$f = V\eta = (125 - 50t + 5t^2)\frac{100}{[1 + 0.5(1.5t^{-3})(0.25)]^4}$$

Using initial guesses of $t = 1.0$ and 2.0, cubic interpolation yielded the following values of f:

t	f	f'
1.0	4023.05	5611.10
2.0	4101.73	−2170.89
1.414	4973.22	−148.70
1.395	4974.60	3.68 (optimum)

Because f is multiplied by 100, f' after two iterations is small enough. Figure E5.4 is a plot of the objective function $f(t)$.

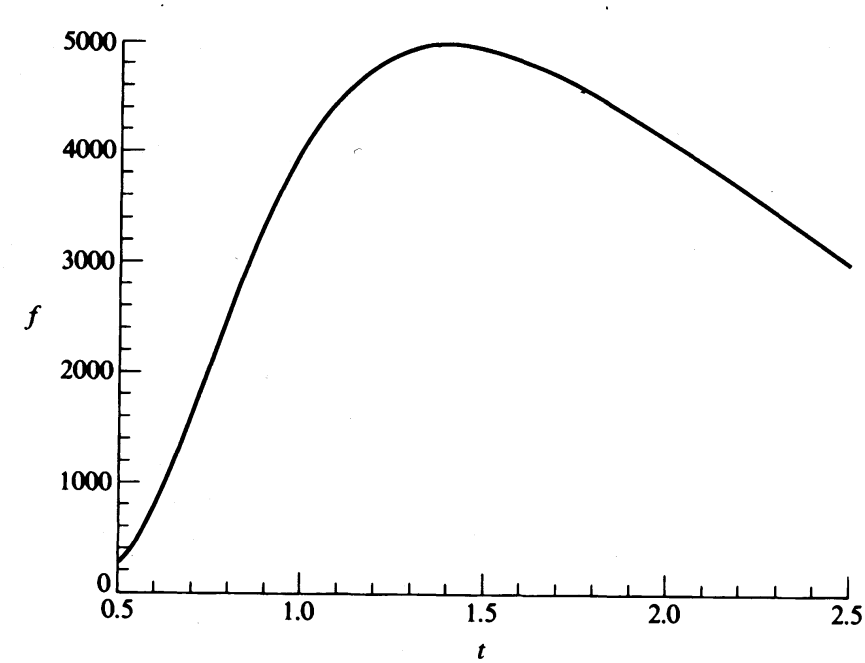


FIGURE E5.4
Plot of objective function (number of good chips per hour) versus resist thickness, $t(\mu\text{m})$.

The throughput for $t^* = 1.395$ is

$$V = 65.02 \text{ wafers/h}$$

If a production line is operated 20 h/day, two lines are needed to achieve 2500 wafers/day.
If quadratic interpolation is used with starting points of $t = 1, 2,$ and 3, the following iterative sequence results:

t	f	f'
1.0	4023.05	5611.10
2.0	4101.73	−2170.89
3.0	1945.40	−1891.73
1.535	4904.08	−942.28
1.511	4924.73	−810.91
1.434	4968.58	−304.19
1.420	4972.17	−196.10
1.406	4974.20	−81.98
1.401	4974.48	−44.78
1.398	4974.58	−20.24
1.397	4974.60	−10.76
1.396	4974.61	−5.01

5.5 HOW ONE-DIMENSIONAL SEARCH IS APPLIED IN A MULTIDIMENSIONAL PROBLEM

In minimizing a function $f(\mathbf{x})$ of several variables, the general procedure is to (a) calculate a search direction and (b) reduce the value of $f(\mathbf{x})$ by taking one or more steps in that search direction. Chapter 6 describes in detail how to select search directions. Here we explain how to take steps in the search direction as a function of a single variable, the step length α. The process of choosing α is called a *unidimensional search* or *line search.*

Examine Figure 5.7 in which contours of a function of two variables are displayed:

$$f(\mathbf{x}) = x_1^4 - 2x_2x_1^2 + x_2^2 + x_1^2 - 2x_1 + 5$$

Suppose that the negative gradient of $f(\mathbf{x})$, $-\nabla f(\mathbf{x})$, is selected as the search direction starting at the point $\mathbf{x}^T = [1\ 2]$. The negative gradient is the direction that maximizes the rate of change of $f(\mathbf{x})$ in moving toward the minimum. To move in this direction we want to calculate a new $\mathbf{x}$

$$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \alpha\mathbf{s}$$

where $\mathbf{s}$ is the search direction, a vector, and α is a scalar denoting the distance moved along the search direction. Note $\alpha\mathbf{s} \equiv \Delta\mathbf{x}$, the vector for the step to be taken (encompassing both direction and distance).

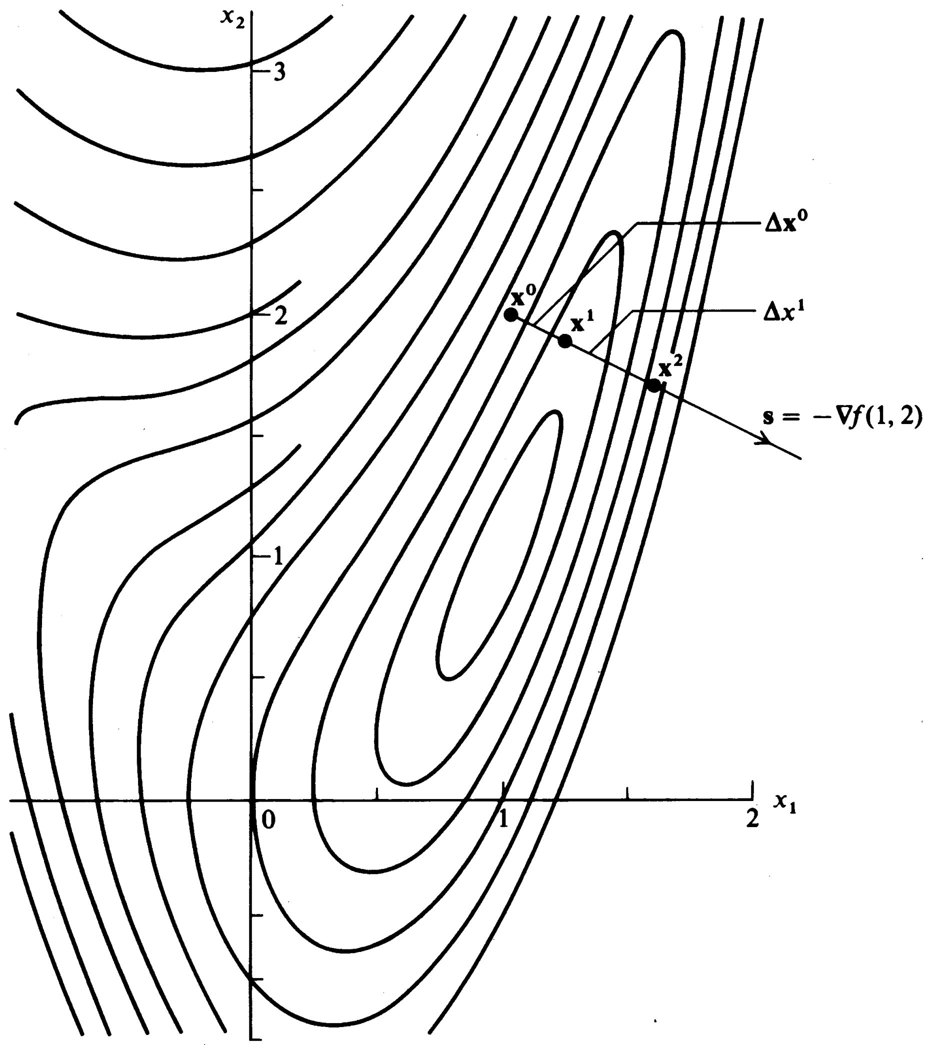


FIGURE 5.7
Unidimensional search to bracket the minimum.

Execution of a unidimensional search involves calculating a value of α and then taking steps in each of the coordinate directions as follows:

$$\text{In the } x_1 \text{ direction: } x_{1,\text{new}} = x_{1,\text{old}} + \alpha s_1$$

$$\text{In the } x_2 \text{ direction: } x_{2,\text{new}} = x_{2,\text{old}} + \alpha s_2$$

where s_1 and s_2 are the two components of $\mathbf{s}$ in the x_1 and x_2 directions, respectively. Repetition of this procedure accomplishes the unidimensional search.

EXAMPLE 5.5 EXECUTION OF A UNIDIMENSIONAL SEARCH

We illustrate two stages in bracketing the minimum in minimizing the function from Fox (1971)

$$f(\mathbf{x}) = x_1^4 - 2x_2x_1^2 + x_2^2 + x_1^2 - 2x_1 + 5$$

in the negative gradient direction

$$-\nabla f(\mathbf{x}) = -\begin{bmatrix} 4x_1^3 - 4x_2x_1 + 2x_1 - 2 \\ -2x_1^2 + 2x_2 \end{bmatrix}$$

starting at $\mathbf{x}^T = [1\ 2]$ where $f(\mathbf{x}) = 5$. Here

$$s = -\nabla f(1, 2) = -\begin{bmatrix} -4 \\ 2 \end{bmatrix}$$

We start to bracket the minimum by taking $\alpha^0 = 0.05$

$$x_1^1 = x_1^0 + (0.05)(4) = 1.2 \qquad (a)$$

$$x_2^1 = x_2^0 + (0.05)(-2) = 1.9 \qquad (b)$$

Steps (a) and (b) consist of one overall step in the direction $\mathbf{s} = [4\ -2]^T$, and yield $\Delta\mathbf{x}^T = [0.2\ -0.1]$. At $\mathbf{x}^1$, $f(1.2, 1.9) = 4.25$, an improvement.

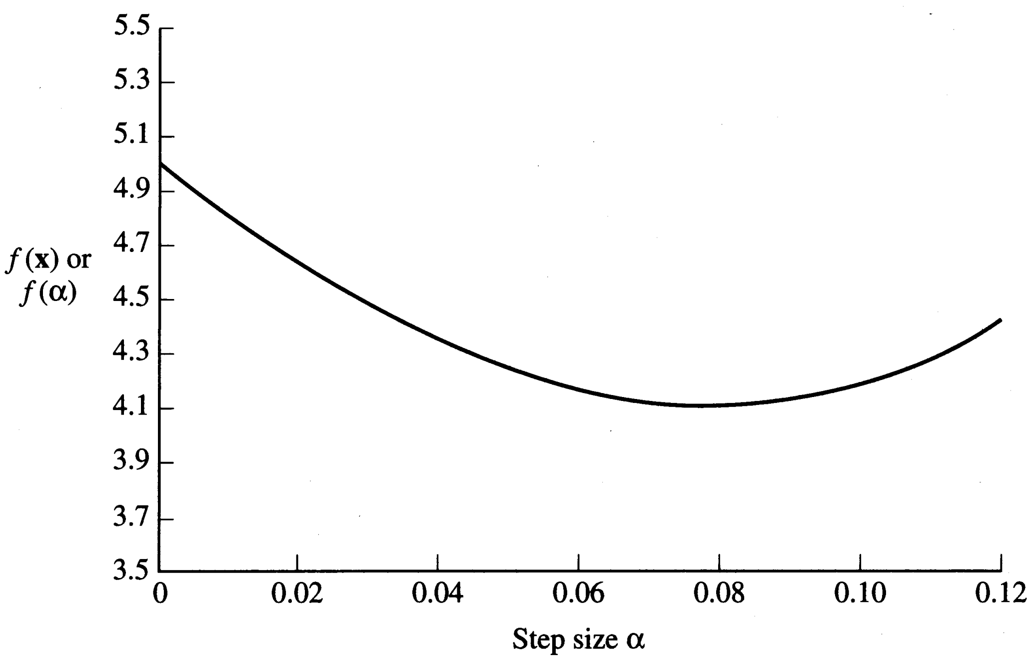


FIGURE E5.5
Values of $f(x)$ along the gradient vector $[4\ \ -2]^T$ starting at $[1\ \ 2]^T$.

For the next step, we let $\alpha^1 = 2\alpha^0 = 0.1$, and take another step in the same direction:

$$x_1^2 = x_1^1 + 0.1(4) = 1.6$$

$$x_2^2 = x_2^1 + 0.1(-2) = 1.7$$

$$\Delta\mathbf{x}^1 = [0.4 \quad -0.2]^T$$

At $\mathbf{x}^2$, $f(1.6, 1.7) = 5.10$, so that the minimum of $f(x)$ in direction **s** has been bracketed. Examine Figure 5.7. The optimal value of α along the search direction can be found to be $\widetilde{\alpha}^* = 0.0797$ by one of the methods described in this chapter. Figure E5.5 shows a plot of f versus α along the search direction.

5.6 EVALUATION OF UNIDIMENSIONAL SEARCH METHODS

In this chapter we described and illustrated only a few unidimensional search methods. Refer to Luenberger (1984), Bazarra et al. (1993), or Nash and Sofer (1996) for many others. Naturally, you can ask which unidimensional search method is best to use, most robust, most efficient, and so on. Unfortunately, the various algorithms are problem-dependent even if used alone, and if used as subroutines in optimization codes, also depend on how well they mesh with the particular code. Most codes simply take one or a few steps in the search direction, or in more than one direction, with no requirement for accuracy—only that $f(x)$ be reduced by a sufficient amount.

REFERENCES

Bazarra, M. S.; H. D. Sherali; and C. M. Shetly. *Nonlinear Programming: Theory and Algorithms*. Wiley, New York (1993).

Becker, H. A.; P. L. Douglas; and S. Ilias. "Development of Optimization Strategies for Industrial Grain Dryer Systems." *Can J Chem Eng,* **62:** 738–745 (1984).

Beveridge, G. S. G.; and R. S. Schechter. *Optimization: Theory and Practice*. McGraw-Hill, New York (1970).

Cook, L. N. "Laboratory Approach Optimizes Filter-Aid Addition." *Chem Eng,* July 23, 1984: 45–50.

Dennis, J. E.; and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, New Jersey (1983) chapter 2.

Fox, R. L. *Optimization Methods for Engineering Design*. Addison-Wesley, Reading, Massachusetts (1971) p. 42.

Luenberger, D. G. *Linear and Nonlinear Programming*. 2nd ed. Addison-Wesley, Menlo Park, CA (1984).

Nash, S. G.; and A. Sofer. *Linear and Nonlinear Programming*. McGraw-Hill, New York (1996).

Wilde, D. J. *Optimum Seeking Methods*. Prentice-Hall, Englewood Cliffs, New Jersey (1964).

Wilde, D. J. "Hidden Optima in Engineering Design." In *Constructive Approaches to Mathematical Models*. Academic Press, New York (1979): 243–248.

SUPPLEMENTARY REFERENCES

Beightler, C. S.; D. T. Phillips; and D. J. Wild. *Foundations of Optimization*. 2nd ed. Prentice-Hall, Englewood Cliffs, New Jersey (1979).

Brent, R. P. *Algorithms for Minimization Without Derivatives*. Prentice-Hall, Englewood Cliffs, New Jersey (1973).

Cooper, L.; and D. Steinberg. *Introduction to Methods of Optimization*. W. B. Saunders Co., Philadelphia (1970).

Reklaitis, G. V.; R. A. Ravindran; and K. M. Ragsdell. *Engineering Optimization*. Wiley-Interscience, New York (1983).

Shoup, T. E.; and F. Mistree. *Optimization Methods with Applications for Personal Computers*. Prentice-Hall, Englewood Cliffs, New Jersey (1987).

Weixnan, L. *Optimal Block Search*. Helderman, Berlin (1984).

PROBLEMS

5.1 Can you bracket the minimum of the following function

$$f(x) = e^x - 1.5x^2$$

starting at $x = 0$? Select different step sizes (small and large), and explain your results. If you have trouble in the analysis, you might plot the function.

5.2 Bracket the minimum of the following functions:

(a) $f(x) = e^x + 1.5x^2$

(b) $f(x) = 0.5(x^2 + 1)(x + 1)$

(c) $f(x) = x^3 - 3x$

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

(e) $f(x) = 0.1x^6 - 0.29x^5 + 2.31x^4 - 8.33x^3 + 12.89x^2 - 6.8x + 1$

5.3 Minimize $f = (x - 1)^4$ via (a) Newton's method and (b) the quasi-Newton (secant) method, starting at (1) $x = -1$, (2) $x = -0.5$, and (3) $x = 0.0$.

5.4 Apply a sequential one-dimensional search technique to reduce the interval of uncertainty for the maximum of the function $f = 6.64 + 1.2x - x^2$ from [0,1] to less than 2 percent of its original size. Show all the iterations.

5.5 List three reasons why a quasi-Newton (secant) search for the minimum of a function of one variable will fail to find a local minimum.

5.6 Minimize the function $f = (x - 1)^4$. Use quadratic interpolation but no more than a maximum of ten function evaluations. The initial three points selected are $x_1 = 0$, $x_2 = 0.5$, and $x_3 = 2.0$.

5.7 Repeat Problem 5.6 but use cubic interpolation via function and derivative evaluations. Use $x_1 = 0.5$ and $x_2 = 2.0$ for a first guess.

5.8 Repeat Problem 5.6 for cubic interpolation with four function values: $x_1 = 1.5, x_2 = 3.0$, $x_3 = 4.0$, and $x_4 = 4.5$.

5.9 Carry out the initial and one additional stage of the numerical search for the minimum of

$$f(x) = 2x^3 - 5x^2 - 8 \quad x \geq 1$$

by (a) Newton's method (start at $x = 1$), (b) the quasi-Newton (secant) method (pick a starting point), and (c) polynomial approximation (pick starting points including $x = 1$).

5.10 Find the maximum of the following function

$$f(x) = 1 - 8x + 2x^2 - \tfrac{10}{3}x^3 + \tfrac{1}{4}x^4 + \tfrac{4}{5}x^5 - \tfrac{1}{6}x^6$$

$$\textit{Hint:} \quad f'(x) = (1 + x)^2(2 - x)^3$$

(a) Analytically. (b) By Newton's method (two iterations will suffice). Start at $x = -2$. List each step of the procedure. (c) By quadratic interpolation (two iterations will suffice). Start at $x = -2$. List each step of the procedure.

5.11 Determine the relative rates of convergence for (1) Newton's method, (2) a finite difference Newton method, (3) quasi-Newton method, (4) quadratic interpolation, and (5) cubic interpolation, in minimizing the following functions:
(a) $x^2 - 6x + 3$ (b) sin (x) with $0 < x < 2\pi$ (c) $x^4 - 20x^3 + 0.1x$

5.12 The total annual cost of operating a pump and motor C in a particular piece of equipment is a function of x, the size (horsepower) of the motor, namely

$$C = \$500 + \$0.9x + \frac{\$0.03}{x}(150{,}000)$$

Find the motor size that minimizes the total annual cost.

5.13 A boiler house contains five coal-fired boilers, each with a nominal rating of 300 boiler horsepower (BHP). If economically justified, each boiler can be operated at a rating of 350 percent of nominal. Due to the growth of manufacturing departments, it has become necessary to install additional boilers. Refer to the following data. Determine the percent of nominal rating at which the present boilers should be operated. *Hint*: Minimize total costs per year BHP output.

Data: The cost of fuel, coal, including the cost of handling coal and removing cinders, is \$7 per ton, and the coal has a heating value of 14,000 Btu/lb. The overall efficiency of the boilers, from coal to steam, has been determined from tests of the present boilers operated at various ratings as:

Percent of nominal rating, R	**Percent overall thermal efficiency, E**
100	75
150	76
200	74
225	72
250	69
275	65
300	61

The annual fixed charges C_F in dollars per year on each boiler are given by the equation:

$$C_F = 14{,}000 + 0.04R^2$$

Assume 8550 hours of operation per year.

Hint: You will find it helpful to first obtain a relation between R and E by least squares (refer to Chapter 2) to eliminate the variable E.

5.14 A laboratory filtration study is to be carried out at constant rate. The basic equation (Cook, 1984) comes from the relation

$$\text{Flow-rate} \propto \frac{(\text{Pressure drop})(\text{Filter area})}{(\text{Fluid viscosity})(\text{Cake thickness})}$$

Cook expressed filtration time as

$$t_f = \beta \frac{\Delta P_c A^2}{\mu M^2 c} x_c \exp(-a x_c + b)$$

where t_f = time to build up filter cake, min
ΔP_c = pressure drop across cake, psig (20)
A = filtration area, ft^2 (250)
μ = filtrate viscosity, centipoise (20)
M = mass flow of filtrate, lb$_m$/min (75)
c = solids concentration in feed to filter, lb$_m$/lb$_m$ filtrate (0.01)
x_c = mass fraction solids in dry cake
a = constant relating cake resistance to solids fraction (3.643)
h = constant relating cake resistance to solids fraction (2.680)
β = 3.2×10^{-8} (lb$_m$/ft)^2

Numerical values for each parameter are given in parentheses. Obtain the maximum time for filtration as a function of x_c by a numerical unidimensional search.

5.15 An industrial dryer for granular material can be modeled (Becker et al., 1984) with the total specific cost of drying C(\$/m^3) being

$$C = [1.767 \ln(W_0/W_D)/\beta V_t] \frac{(F_A C_{pA} + UA)\Delta T C'_p}{\Delta H_C + PC'_E + C'_L}$$

where A = heat transfer area of dryer normal to the air flow, m^2 (153.84)
β = constant, function of air plenum temperature and initial moisture level
C'_E = unit cost of electricity, \$/kWh (0.0253)
C'_L = unit cost of labor, \$/h (15)
C'_p = unit cost of propane, \$/kg (0.18)
C_{pA} = specific heat of air, J/kg K (1046.75)
F_A = flow-rate of air, kg/h (3.38×10^5)
ΔH_c = heat combustion of propane, J/kg (4.64×10^7)
P = electrical power, kW (188)
ΔT = temperature difference ($T - T_1$), K; the plenum air temperature T minus the inlet air temperature T_1 (T_1 = 390 K)

U = overall heat transfer coefficient from dryer to atmosphere, W/(m^2)(K)(45)
V_t = total volume of the dryer, m^3 (56)
W_D = final grain moisture content (dry basis), kg/kg (0.1765)
W_0 = initial moisture content (dry basis), kg/kg (0.500)

Numerical values for each parameter are given in parentheses. Values for the coefficient are given by

$$\beta = (-0.263\,1125 + 0.00289587T)\, W_0^{(-0.2368125 + 0.0009966T)}$$

Find the minimum cost as a function of the plenum temperature T (in kelvin).

5.16 The following is an example from D. J. Wilde (1979).

The first example was formulated by Stoecker* to illustrate the steepest descent (gradient) direct search method. It is proposed to attach a vapor recondensation refrigeration system to lower the temperature, and consequently vapor pressure, of liquid ammonia stored in a steel pressure vessel, for this would permit thinner vessel walls. The tank cost saving must be traded off against the refrigeration and thermal insulation cost to find the temperature and insulation thickness minimizing the total annual cost. Stoecker showed the total cost to be the sum of insulation cost $i \equiv 400x^{0.9}$ (x is the insulation thickness, in.), the vessel cost $v \equiv 1000 + 22(p - 14.7)^{1.2}$ (p is the absolute pressure, psia), and the recondensation cost $r \equiv 144(80 - t)/x$ (t is the temperature, °F). The pressure is related to the temperature by

$$\ln p = -3950(t - 460)^{-1} + 11.86$$

By direct gradient search, iterated 16 times from a starting temperature of 50°F, the total annual cost is found to have a local minimum at $x = 5.94$ in. and $t = 6.29$°F, where the cost is \$53,400/yr. The reader can verify, however, that an ambient system (80°F) without any recondensation only costs \$52,000/yr, a saving of 3%.

Is the comment in the example true?

*Stoecker, W. F. In "Design of Thermal Systems." McGraw-Hill, New York (1971), pp. 152–155.