Chapter 8
Basic Descent Methods

We turn now to a description of the basic techniques used for iteratively solving unconstrained minimization problems. These techniques are, of course, important for practical application since they often offer the simplest, most direct alternatives for obtaining solutions; but perhaps their greatest importance is that they establish certain reference plateaus with respect to difficulty of implementation and speed of convergence. Thus in later chapters as more efficient techniques and techniques capable of handling constraints are developed, reference is continually made to the basic techniques of this chapter both for guidance and as points of comparison.

There is a fundamental underlying structure for almost all the descent algorithms we discuss. One starts at an initial point; determines, according to a fixed rule, a direction of movement; and then moves in that direction to a (relative) minimum of the objective function on that line. At the new point a new direction is determined and the process is repeated. The primary differences between algorithms (steepest descent, Newton's method, etc.) rest with the rule by which successive directions of movement are selected. Once the selection is made, all algorithms call for movement to the minimum point on the corresponding line.

The process of determining the minimum point on a given line (one variable only) is called *line search*. For general nonlinear functions that cannot be minimized analytically, this process actually is accomplished by searching, in an intelligent manner, along the line for the minimum point. These line search techniques, which are really procedures for solving one-dimensional minimization problems, form the backbone of nonlinear programming algorithms, since higher dimensional problems are ultimately solved by executing a sequence of successive line searches. There are a number of different approaches to this important phase of minimization and the first half of this chapter is devoted to their, discussion.

The last sections of the chapter are devoted to a description and analysis of the basic descent algorithms for unconstrained problems; steepest descent, coordinate descent, and Newton's method. These algorithms serve as primary models for the development and analysis of all others discussed in the book.

© Springer International Publishing Switzerland 2016

D.G. Luenberger, Y. Ye, *Linear and Nonlinear Programming*, International Series in Operations Research & Management Science 228,
DOI 10.1007/978-3-319-18842-3_8

8.1 Line Search Algorithms

These algorithms are classified by the order of information of the objective functions $f(x)$ being evaluated.

0*th-Order Method: Golden Section Search and Curve Fitting*

A very popular method for resolving the line search problem is the Fibonacci search method described in this section. The method has a certain degree of theoretical elegance, which no doubt partially accounts for its popularity, but on the whole, as we shall see, there are other procedures which in most circumstances are superior.

The method determines the minimum value of a function f over a closed interval $[c_1,\ c_2]$. In applications, f may in fact be defined over a broader domain, but for this method a fixed interval of search must be specified. The only property that is assumed of f is that it is *unimodal*, that is, it has a single relative minimum (see Fig. 8.1). The minimum point of f is to be determined, at least approximately, by measuring the value of f at a certain number of points. It should be imagined, as is indeed the case in the setting of nonlinear programming, that each measurement of f is somewhat costly—of time if nothing more.

To develop an appropriate search strategy, that is, a strategy for selecting measurement points based on the previously obtained values, we pose the following problem: Find how to successively select N measurement points so that, without explicit knowledge of f, we can determine the smallest possible region of uncertainty in which the minimum must lie. In this problem the region of uncertainty is determined in any particular case by the relative values of the measured points in conjunction with our assumption that f is unimodal. Thus, after values are known at N points $x_1,\ x_2,\ \ldots,\ x_N$ with

$$c_1 \leqslant x_1 < x_2 \ldots < x_{N-1} < x_N \leqslant c_2,$$

the region of uncertainty is the interval $[x_{k-1},\ x_{k+1}]$ where x_k is the minimum point among the N, and we define $x_0 = c_1,\ x_{N+1} = c_2$ for consistency. The minimum of f must lie somewhere in this interval.

The derivation of the optimal strategy for successively selecting measurement points to obtain the smallest region of uncertainty is fairly straight-forward but somewhat tedious. We simply state the result and give an example.

Let

$$\begin{aligned} d_1 &= c_2 - c_1, \text{ the initial width of uncertainty} \\ d_k &= \text{ width of uncertainty after } k \text{ measurements} \end{aligned}$$

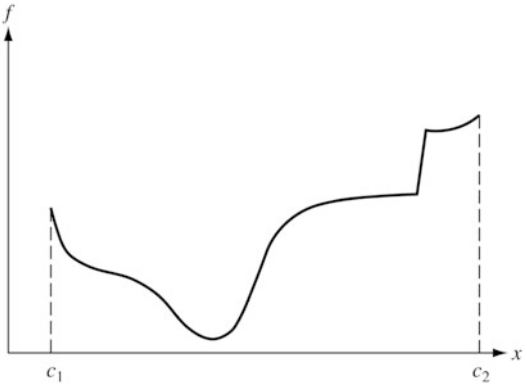

Fig. 8.1 A unimodal function

Then, if a total of N measurements are to be made, we have

$$d_k = \left(\frac{F_{N-k+1}}{F_N}\right) d_1, \tag{8.1}$$

where the integers F_k are members of the Fibonacci sequence generated by the recurrence relation

$$F_N = F_{N-1} + F_{N-2}, \qquad F_0 = F_1 = 1. \tag{8.2}$$

The resulting sequence is 1, 1, 2, 3, 5, 8, 13, ...

The procedure for reducing the width of uncertainty to d_N is this: The first two measurements are made symmetrically at a distance of $(F_{N-1}/F_N)d_1$ from the ends of the initial intervals; according to which of these is of lesser value, an uncertainty interval of width $d_2 = (F_{N-1}/F_N)d_1$ is determined. The third measurement point is placed symmetrically in this new interval of uncertainty with respect to the measurement already in the interval. The result of this third measurement gives an interval of uncertainty $d_3 = (F_{N-2}/F_N)d_1$. In general, each successive measurement point is placed in the current interval of uncertainty symmetrically with the point already existing in that interval.

Some examples are shown in Fig. 8.2. In these examples the sequence of measurement points is determined in accordance with the assumption that each measurement is of lower value than its predecessors. Note that the procedure always calls for the last two measurements to be made at the midpoint of the semifinal interval of uncertainty. We are to imagine that these two points are actually separated a small distance so that a comparison of their respective values will reduce the interval to nearly half. This terminal anomaly of the Fibonacci search process is, of course, of no great practical consequence.

Search by Golden Section

If the number N of allowed measurement points in a Fibonacci search is made to approach infinity, we obtain the golden section method. It can be argued, based on the optimal property of the finite Fibonacci method, that the corresponding infinite version yields a sequence of intervals of uncertainty whose widths tend to zero faster than that which would be obtained by other methods.

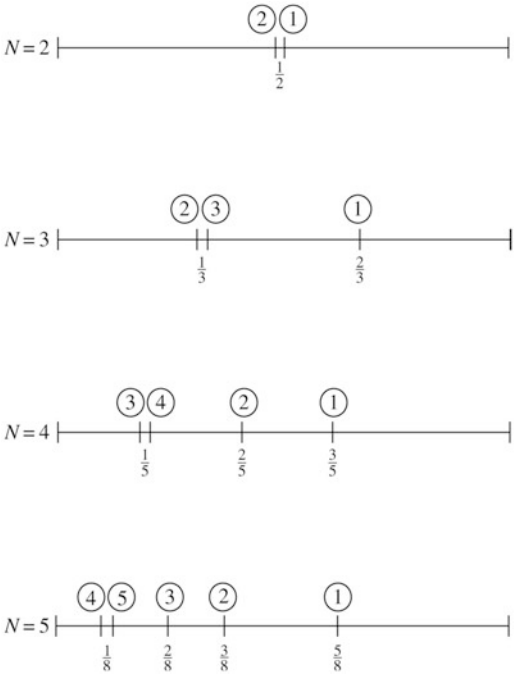

Fig. 8.2 Fibonacci search

The solution to the Fibonacci difference equation

$$F_N = F_{N-1} + F_{N-2} \tag{8.3}$$

is of the form

$$F_N = A\tau_1^N + B\tau_2^N, \tag{8.4}$$

where τ_1 and τ_2 are roots of the characteristic equation

$$\tau^2 = \tau + 1.$$

Explicitly,

$$\tau_1 = \frac{1 + \sqrt{5}}{2},\ \tau_2 = \frac{1 - \sqrt{5}}{2}.$$

(The number $\tau_1 \simeq 1.618$ is known as the *golden section* ratio and was considered by early Greeks to be the most aesthetic value for the ratio of two adjacent sides of a rectangle.) For large N the first term on the right side of (8.4) dominates the second, and hence

$$\lim_{N\to\infty} \frac{F_{N-1}}{F_N} = \frac{1}{\tau_1} \simeq 0.618.$$

It follows from (8.1) that the interval of uncertainty at any point in the process has width

$$d_k = \left(\frac{1}{\tau_1}\right)^{k-1} d_1, \tag{8.5}$$

and from this it follows that

$$\frac{d_{k+1}}{d_k} = \frac{1}{\tau_1} = 0.618. \tag{8.6}$$

Therefore, we conclude that, with respect to the width of the uncertainty interval, the search by golden section converges linearly (see Sect. 7.8) to the overall minimum of the function f with convergence ratio $1/\tau_1 = 0.618$.

The Fibonacci search method has a certain amount of theoretical appeal, since it assumes only that the function being searched is unimodal and with respect to this broad class of functions the method is, in some sense, optimal. In most problems, however, it can be safely assumed that the function being searched, as well as being unimodal, possesses a certain degree of smoothness, and one might, therefore, expect that more efficient search techniques exploiting this smoothness can be devised; and indeed they can. Techniques of this nature are usually based on curve fitting procedures where a smooth curve is passed through the previously measured points in order to determine an estimate of the minimum point. A variety of such techniques can be devised depending on whether or not derivatives of the function as well as the values can be measured, how many previous points are used to determine the fit, and the criterion used to determine the fit. In this section a number of possibilities are outlined and analyzed. All of them have orders of convergence greater than unity.

Quadratic Fit

The scheme that is often most useful in line searching is that of fitting a quadratic through three given points. This has the advantage of not requiring any derivative information. Given x_1, x_2, x_3 and corresponding values $f(x_1) = f_1$, $f(x_2) = f_2$, $f(x_3) = f_3$ we construct the quadratic passing through these points

$$q(x) = \sum_{i=1}^{3} f_i \frac{\prod_{j\neq i}(x - x_j)}{\prod_{j\neq i}(x_i - x_j)}, \tag{8.7}$$

and determine a new point x_4 as the point where the derivative of q vanishes. Thus

$$x_4 = \frac{1}{2}\frac{b_{23}f_1 + b_{31}f_2 + b_{12}f_3}{a_{23}f_1 + a_{31}f_2 + a_{12}f_3}, \tag{8.8}$$

where $a_{ij} = x_i - x_j,\ b_{ij} = x_i^2 - x_j^2$.

Define the errors $\varepsilon_i = x^* - x_i,\ i = 1, 2, 3, 4$. The expression for ε_4 must be a polynomial in $\varepsilon_1,\ \varepsilon_2,\ \varepsilon_3$. It must be second order (since it is a quadratic fit). It must go to zero if any two of the errors $\varepsilon_1,\ \varepsilon_2,\ \varepsilon_3$ is zero. (The reader should check this.) Finally, it must be symmetric (since the order of points is relevant). It follows that near a minimum point x^* of f, the errors are related approximately by

$$\varepsilon_4 = M(\varepsilon_1\varepsilon_2 + \varepsilon_2\varepsilon_3 + \varepsilon_1\varepsilon_3), \tag{8.9}$$

where M depends on the values of the second and third derivatives of f at x^*.

If we assume that $\varepsilon_k \to 0$ with an order greater than unity, then for large k the error is governed approximately by

$$\varepsilon_{k+2} = M\varepsilon_k\varepsilon_{k-1}.$$

Letting $y_k = \log M\varepsilon_k$ this becomes

$$y_{k+2} = y_k + y_{k-1}$$

with characteristic equation

$$\lambda^3 - \lambda\ - 1 = 0.$$

The largest root of this equation is $\lambda \simeq 1.3$ which thus determines the rate of growth of y_k and is the order of convergence of the quadratic fit method.

1*st-Order Method: Curve Fitting and Methods of False Position*

In this section a number fitting methods using the first derivative information are described. All of them have orders of convergence greater than unity.

Quadratic Fit: Method of False Position

Suppose that at two points x_k and x_{k-1} where measurements $f(x_k),\ f'(x_k),\ f'(x_{k-1})$ are available, it is possible to fit the quadratic

$$q(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{f'(x_{k-1}) - f'(x_k)}{x_{k-1} - x_k} \cdot \frac{(x - x_k)^2}{2},$$

which has the same corresponding values. An estimate x_{k+1} can then be determined by finding the point where the derivative of q vanishes; thus

$$x_{k+1} = x_k - f'(x_k)\left[\frac{x_{k-1} - x_k}{f'(x_{k-1}) - f'(x_k)}\right]. \tag{8.10}$$

(See Fig. 8.3.) Comparing this formula with Newton's method, we see again that the value $f(x_k)$ does not enter; hence, our fit could have been passed through either $f(x_k)$ or $f(x_{k-1})$. Also the formula can be regarded as an approximation to Newton's method where the second derivative is replaced by the difference of two first derivatives.

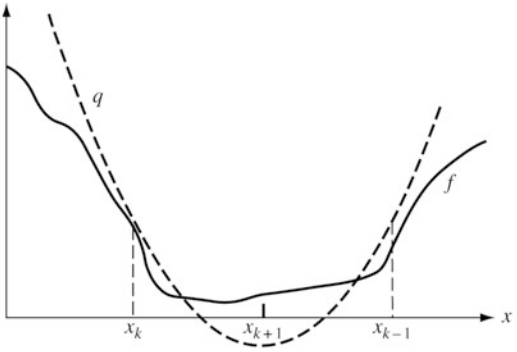

Fig. 8.3 False position for minimization

Again, since this method does not depend on values of f directly, it can be regarded as a method for solving $f'(x) \equiv g(x) = 0$. Viewed in this way the method, which is illustrated in Fig. 8.4, takes the form

$$x_{k+1} = x_k - g(x_k)\left[\frac{x_k - x_{k-1}}{g(x_k) - g(x_{k-1})}\right]. \tag{8.11}$$

We next investigate the order of convergence of the method of false position and discover that it is order $\tau_1 \simeq 1.618$, the golden mean.

Proposition. *Let g have a continuous second derivative and suppose x^* is such that $g(x^*) = 0$, $g'(x^*) \neq 0$. Then for x_0 sufficiently close to x^*, the sequence $\{x_k\}_{k=0}^{\infty}$ generated by the method of false position (8.11) converges to x^* with order $\tau_1 \simeq 1.618$.*

Proof. Introducing the notation

$$g[a,\ b] = \frac{g(b) - g(a)}{b - a}, \tag{8.12}$$

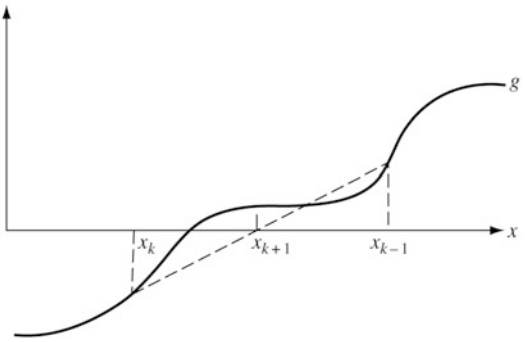

Fig. 8.4 False position for solving equations

we have

$$x_{k-1} - x^* = x_k - x^* - g(x_k)\left[\frac{x_k - x_{k-1}}{g(x_k) - g(x_{k-1})}\right]$$
$$= (x_k - x^*)\left\{\frac{g[x_{k-1}, x_k] - g[x_k, x^*]}{g[x_{k-1}, x_k]}\right\}. \tag{8.13}$$

Further, upon the introduction of the notation

$$g[a,\ b,\ c] = \frac{g[a,b] - g[b,c]}{a - c},$$

we may write (8.13) as

$$x_{k+1} - x^* = (x_k - x^*)(x_{k-1} - x^*)\left\{\frac{g[x_{k-1}, x_k, x^*]}{g[x_{k-1}, x_k]}\right\}.$$

Now, by the mean value theorem with remainder, we have (see Exercise 2)

$$g[x_{k-1},\ x_k] = g'(\xi_k) \tag{8.14}$$

and

$$g[x_{k-1},\ x_k,\ x^*] = \frac{1}{2}g''(\eta k), \tag{8.15}$$

where ξ_k and η_k are convex combinations of $x_k,\ x_{k-1}$ and $x_k,\ x_{k-1},\ x^*$, respectively. Thus

$$x_{k+1} - x^* = \frac{g''(\eta_k)}{2g'(\xi_k)}(x_k - x^*)(x_{k-1} - x^*). \tag{8.16}$$

It follows immediately that the process converges if it is started sufficiently close to x^*.

To determine the order of convergence, we note that for large k Eq. (8.16) becomes approximately

$$x_{k+1} - x^* = M(x_k - x^*)(x_{k-1} - x^*),$$

where

$$M = \frac{g''(x^*)}{2g'(x^*)}.$$

Thus defining $\varepsilon_k = (x_k - x^*)$ we have, in the limit,

$$\varepsilon_{k+1} = M\varepsilon_k\varepsilon_{k-1}. \tag{8.17}$$

Taking the logarithm of this equation we have, with $y_k = \log M\varepsilon_k$,

$$y_{k+1} = y_k + y_{k-1}, \tag{8.18}$$

which is the Fibonacci difference equation discussed in Sect. 7.1. A solution to this equation will satisfy

$$y_{k+1} - \tau_1 y_k \to 0.$$

Thus

$$\log M\varepsilon_{k+1} - \tau_1 \log M\varepsilon_k \to 0 \text{ or } \log \frac{M\varepsilon_{k+1}}{(M\varepsilon_k)^{\tau_1}} \to 0,$$

and hence

$$\frac{\varepsilon_{k+1}}{\varepsilon_k^{\tau_1}} \to M^{(\tau_1 - 1)}. \blacksquare$$

Having derived the error formula (8.17) by direct analysis, it is now appropriate to point out a short-cut technique, based on symmetry and other considerations, that can sometimes be used in even more complicated situations. The right side of error formula (8.17) must be a polynomial in ε_k and ε_{k-1}, since it is derived from approximations based on Taylor's theorem. Furthermore, it must be second order, since the method reduces to Newton's method when $x_k = x_{k-1}$. Also, it must go to zero if either ε_k or ε_{k-1} go to zero, since the method clearly yields $\varepsilon_{k+1} = 0$ in that case. Finally, it must be symmetric in ε_k and ε_{k-1}, since the order of points is irrelevant. The only formula satisfying these requirements is $\varepsilon_{k+1} = M\varepsilon_k\varepsilon_{k-1}$.

Cubic Fit

Given the points x_{k-1} and x_k together with the values $f(x_{k-1}), f'(x_{k-1}), f(x_k), f'(x_k)$, it is also possible to fit a cubic equation to the points having corresponding values. The next point x_{k+1} can then be determined as the relative minimum point of this cubic. This leads to

$$x_{k+1} = x_k - (x_k - x_{k-1})\left[\frac{f'(x_k) + u_2 - u_1}{f'(x_k) - f'(x_{k-1}) + 2u_2}\right], \tag{8.19}$$

where

$$u_1 = f'(x_{k-1}) + f'(x_k) - 3\frac{f(x_{k-1}) - f(x_k)}{x_{k-1} - x_k}$$
$$u_2 = [u_1^2 - f'(x_{k-1})f'(x_k)]^{1/2},$$

which is easily implementable for computations.

It can be shown (see Exercise 3) that the order of convergence of the cubic fit method is 2.0. Thus, although the method is exact for cubic functions indicating that its order might be three, its order is actually only two.

2nd-Order Method: Newton's Method

Suppose that the function f of a single variable x is to be minimized, and suppose that at a point x_k where a measurement is made it is possible to evaluate the three numbers $f(x_k)$, $f'(x_k)$, $f''(x_k)$. It is then possible to construct a quadratic function q which at x_k agrees with f up to second derivatives, that is

$$q(x) = f(x_k) + f'(x_k)(x - x_k) + \frac{1}{2}f''(x_k)(x - x_k)^2. \tag{8.20}$$

We may then calculate an estimate x_{k+1} of the minimum point of f by finding the point where the derivative of q vanishes. Thus setting

$$0 = q'(x_{k+1}) = f'(x_k) + f''(x_k)(x_{k+1} - x_k),$$

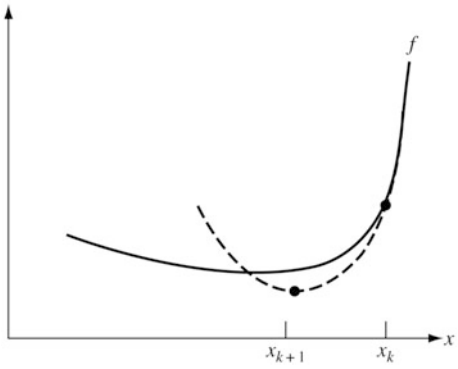

Fig. 8.5 Newton's method for minimization

we find

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}. \tag{8.21}$$

This process, which is illustrated in Fig. 8.5, can then be repeated at x_{k+1}.

We note immediately that the new point x_{k+1} resulting from Newton's method does not depend on the value $f(x_k)$. The method can more simply be viewed as a technique for iteratively solving equations of the form

$$g(x) = 0,$$

where, when applied to minimization, we put $g(x) \equiv f'(x)$. In this notation Newton's method takes the form

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}. \tag{8.22}$$

This form is illustrated in Fig. 8.6.

We now show that Newton's method has order two convergence:

Proposition. *Let the function g have a continuous second derivative, and let x^* satisfy $g(x^*) = 0$, $g'(x^*) \neq 0$. Then, provided x_0 is sufficiently close to x^*, the sequence $\{x_k\}_{k=0}^{\infty}$ generated by Newton's method (8.22) converges to x^* with an order of convergence at least two.*

Proof. For points ξ in a region near x^* there is a k_1 such that $|g''(\xi)| < k_1$ and a k_2 such that $|g'(\xi)| > k_2$. Then since $g(x^*) = 0$ we can write

$$\begin{aligned} x_{k+1} - x^* &= x_k - x^* - \frac{g(x_k) - g(x^*)}{g'(x_k)} \\ &= -[g(x_k) - g(x^*) + g'(x_k)(x^* - x_k)]/g'(x_k). \end{aligned}$$

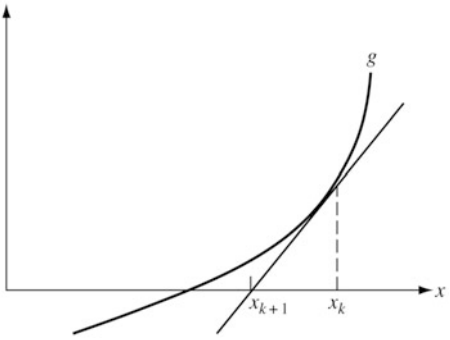

Fig. 8.6 Newton's method for solving equations

The term in brackets is, by Taylor's theorem, zero to first-order. In fact, using the remainder term in a Taylor series expansion about x_k, we obtain

$$x_{k+1} - x^* = \frac{1}{2}\frac{g''(\xi)}{g'(x_k)}(x_k - x^*)^2$$

for some ξ between x^* and x_k. Thus in the region near x^*,

$$|x_{k+1} - x^*| \leqslant \frac{k_1}{2k_2}|x_k - x^*|^2.$$

We see that if $|x_k - x^*|k_1/2k_2 < 1$, then $|x_{k+1} - x^*| < |x_k - x^*|$ and thus we conclude that if started close enough to the solution, the method will converge to x^* with an order of convergence at least two. ∎

Global Convergence of Curve Fitting

Above, we analyzed the convergence of various curve fitting procedures in the neighborhood of the solution point. If, however, any of these procedures were applied in pure form to search a line for a minimum, there is the danger—alas, the most likely possibility—that the process would diverge or wander about meaninglessly. In other words, the process may never get close enough to the solution for our detailed local convergence analysis to be applicable. It is therefore important to artfully combine our knowledge of the local behavior with conditions guaranteeing global convergence to yield a workable and effective procedure.

The key to guaranteeing global convergence is the Global Convergence Theorem of Chap. 7. Application of this theorem in turn hinges on the construction of a suitable descent function and minor modifications of a pure curve fitting algorithm. We offer below a particular blend of this kind of construction and analysis, taking as departure point the quadratic fit procedure discussed in Sect. 8.1 above.

Let us assume that the function f that we wish to minimize is strictly unimodal and has continuous second partial derivatives. We initiate our search procedure by searching along the line until we find three points $x_1,\ x_2,\ x_3$ with $x_1 < x_2 < x_3$ such that $f(x_1) \geqslant f(x_2) \leqslant f(x_3)$. In other words, the value at the middle of these three points is less than that at either end. Such a sequence of points can be determined in a number of ways—see Exercise 7.

The main reason for using points having this pattern is that a quadratic fit to these points will have a minimum (rather than a maximum) and the minimum point will lie in the interval $[x_1,\ x_3]$. See Fig. 8.7. We modify the pure quadratic fit algorithm so that it always works with points in this basic *three-point pattern*.

The point x_4 is calculated from the quadratic fit in the standard way and $f(x_4)$ is measured. Assuming (as in the figure) that $x_2 < x_4 < x_3$, and accounting for the unimodal nature of f, there are but two possibilities:

1. $f(x_4) \leqslant f(x_2)$
2. $f(x_2) < f(x_4) \leqslant f(x_3)$.

In either case a new three-point pattern, $\bar{x}_1, \bar{x}_2, \bar{x}_3$, involving x_4 and two of the old points, can be determined: In case (8.1) it is

$$(\bar{x}_1, \bar{x}_2, \bar{x}_3) = (x_2,\ x_4,\ x_3),$$

while in case (8.2) it is

$$(\bar{x}_1, \bar{x}_2, \bar{x}_3) = (x_1,\ x_2,\ x_4).$$

We then use this three-point pattern to fit another quadratic and continue. The pure quadratic fit procedure determines the next point from the current point and the previous two points. In the modification above, the next point is determined from the current point and the two out of three last points that form a three-point pattern with it. This simple modification leads to global convergence.

To prove convergence, we note that each three-point pattern can be thought of as defining a vector $\mathbf{x}$ in E^3. Corresponding to an $\mathbf{x} = (x_1, x_2, x_3)$ such that (x_1, x_2, x_3) form a three-point pattern with respect to f, we define $\mathbf{A}(\mathbf{x}) = (\bar{x}_1, \bar{x}_2, \bar{x}_3)$ as discussed above. For completeness we must consider the case where two or more of the x_i, $i = 1, 2, 3$ are equal, since this may occur. The appropriate definitions are simply limiting cases of the earlier ones. For example, if $x_1 = x_2$, then (x_1, x_2, x_3) form a three-point pattern if $f(x_2) \leqslant f(x_3)$ and $f'(x_2) < 0$ (which is the limiting case of $f(x_2) < f(x_1)$). A quadratic is fit in this case by using the values at the two distinct points and the derivative at the duplicated point. In case $x_1 = x_2 = x_3$, (x_1, x_2, x_3)forms a three-point pattern if $f'(x_2) = 0$ and $f''(x_2) \geqslant 0$.

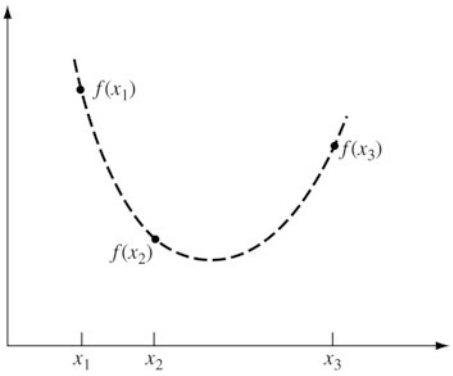

Fig. 8.7 Three-point pattern

With these definitions, the map $\mathbf{A}$ is well defined. It is also continuous, since curve fitting depends continuously on the data.

We next define the solution set $\Gamma \subset E^3$ as the points $\mathbf{x}^* = (x^*, x^*, x^*)$ where $f'(x^*) = 0$.

Finally, we let $Z(\mathbf{x}) = f(x_1) + f(x_2) + f(x_3)$. It is easy to see that Z is a descent function for $\mathbf{A}$. After application of $\mathbf{A}$ one of the values $f(x_1)$, $f(x_2)$, $f(x_3)$ will be replaced by $f(x_4)$, and by construction, and the assumption that f is unimodal, it will replace a strictly larger value. Of course, at $\mathbf{x}^* = (x^*, x^*, x^*)$ we have $\mathbf{A}(\mathbf{x}^*) = \mathbf{x}^*$ and hence $Z(\mathbf{A}(\mathbf{x}^*)) = Z(\mathbf{x}^*)$.

Since all points are contained in the initial interval, we have all the requirements for the Global Convergence Theorem. Thus the process converges to the solution. The order of convergence may not be destroyed by this modification, if near the solution the three-point pattern is always formed from the previous three points. In this case we would still have convergence of order 1.3. This cannot be guaranteed, however.

It has often been implicitly suggested, and accepted, that when using the quadratic fit technique one should require

$$f(x_{k+1}) < f(x_k)$$

so as to guarantee convergence. If the inequality is not satisfied at some cycle, then a special local search is used to find a better x_{k+1} that does satisfy it. This philosophy amounts to taking $Z(\mathbf{x}) = f(x_3)$ in our general framework and, unfortunately, this is not a descent function even for unimodal functions, and hence the special local search is likely to be necessary several times. It is true, of course, that a similar special local search may, occasionally, be required for the technique we suggest in regions of multiple minima, but it is never required in a unimodal region.

The above construction, based on the pure quadratic fit technique, can be emulated to produce effective procedures based on other curve fitting techniques. For application to smooth functions these techniques seem to be the best available in terms of flexibility to accommodate as much derivative information as is available, fast convergence, and a guarantee of global convergence.

*Closedness of Line Search Algorithms

Since searching along a line for a minimum point is a component part of most nonlinear programming algorithms, it is desirable to establish at once that this procedure is closed; that is, that the end product of the iterative procedures outlined above, when viewed as a single algorithmic step finding a minimum along a line, define closed algorithms. That is the objective of this section.

To initiate a line search with respect to a function f, two vectors must be specified: the initial point $\mathbf{x}$ and the direction $\mathbf{d}$ in which the search is to be made. The result of the search is a new point. Thus we define the search algorithm $\mathbf{S}$ as a mapping from E^{2n} to E^n.

We assume that the search is to be made over the semi-infinite line emanating from $\mathbf{x}$ in the direction $\mathbf{d}$. We also assume, for simplicity, that the search is not made in vain; that is, we assume that there is a minimum point along the line. This will be the case, for instance, if f is continuous and increases without bound as $\mathbf{x}$ tends toward infinity.

Definition. The mapping $\mathbf{S} : E^{2n} \to E^n$ is defined by

$$\mathbf{S}(\mathbf{x},\ \mathbf{d}) = \{\mathbf{y} : \mathbf{y} = \mathbf{x} + \alpha\mathbf{d} \text{ for some } \alpha \geqslant 0,\ f(\mathbf{y}) = \min_{0\leqslant\alpha\leqslant\infty} f(\mathbf{x} + \alpha\mathbf{d})\}. \tag{8.23}$$

In some cases there may be many vectors $\mathbf{y}$ yielding the minimum, so $\mathbf{S}$ is a set-valued mapping. We must verify that $\mathbf{S}$ is closed.

Theorem. *Let f be continuous on E^n. Then the mapping defined by (8.23) is closed at $(\boldsymbol{x}, \boldsymbol{d})$ if $\mathbf{d} \neq \mathbf{0}$.*

Proof. Suppose $\{\mathbf{x}_k\}$ and $\{\mathbf{d}_k\}$ are sequences with $\mathbf{x}_k \to \mathbf{x},\ \mathbf{d}_k \to \mathbf{d} \neq \mathbf{0}$. Suppose also that $\mathbf{y}_k \in \mathbf{S}(\mathbf{x}_k,\ \mathbf{d}_k)$ and that $\mathbf{y}_k \to \mathbf{y}$. We must show that $\mathbf{y} \in \mathbf{S}(\mathbf{x},\ \mathbf{d})$.

For each k we have $\mathbf{y}_k = \mathbf{x}_k + \alpha_k\mathbf{d}_k$ for some α_k. From this we may write

$$\alpha_k = \frac{|\mathbf{y}_k - \mathbf{x}_k|}{|\mathbf{d}_k|}.$$

Taking the limit of the right-hand side of the above, we see that

$$\alpha_k \to \overline{\alpha} \equiv \frac{|\mathbf{y} - \mathbf{x}|}{|\mathbf{d}|}.$$

It then follows that $\mathbf{y} = \mathbf{x} + \overline{\alpha}\mathbf{d}$. It still remains to be shown that $\mathbf{y} \in \mathbf{S}(\mathbf{x},\ \mathbf{d})$.

For each k and each $\alpha,\ 0 \leqslant \alpha < \infty$,

$$f(\mathbf{y}_k) \leqslant f(\mathbf{x}_k + \alpha\mathbf{d}_k).$$

Letting $k \to \infty$ we obtain

$$f(\mathbf{y}) \leqslant f(\mathbf{x} + \alpha\mathbf{d}).$$

Thus

$$f(\mathbf{y}) \leqslant \min_{0 \leqslant \alpha < \infty} f(\mathbf{x} + \alpha\mathbf{d}),$$

and hence $\mathbf{y} \in \mathbf{S}(\mathbf{x},\ \mathbf{d})$. ∎

The requirement that $\mathbf{d} \neq \mathbf{0}$ is natural both theoretically and practically. From a practical point of view this condition implies that, when constructing algorithms, the choice $\mathbf{d} = \mathbf{0}$ had better occur only in the solution set; but it is clear that if $\mathbf{d} = \mathbf{0}$, no search will be made. Theoretically, the map $\mathbf{S}$ can fail to be closed at $\mathbf{d} = \mathbf{0}$, as illustrated below.

Example. On E^1 define $f(x) = (x-1)^2$. Then $S(x,\ d)$ is not closed at $x = 0,\ d = 0$. To see this we note that for any $d > 0$

$$\min_{0 \leqslant \alpha < \infty} f(\alpha d) = f(1),$$

and hence

$$S(0,\ d) = 1;$$

but

$$\min_{0 \leqslant \alpha < \infty} f(\alpha \cdot 0) = f(0)$$

so that

$$S(0,\ 0) = 0.$$

Thus as $d \to 0,\ S(0,\ d) \nrightarrow S(0,\ 0)$.

Inaccurate Line Search

In practice, of course, it is impossible to obtain the exact minimum point called for by the ideal line search algorithm $\mathbf{S}$ described above. As a matter of fact, it is often desirable to sacrifice accuracy in the line search routine in order to conserve

overall computation time. Because of these factors we must, to be realistic, be certain, at every stage of development, that our theory does not crumble if inaccurate line searches are introduced.

Inaccuracy generally is introduced in a line search algorithm by simply terminating the search procedure before it has converged. The exact nature of the inaccuracy introduced may therefore depend on the particular search technique employed and the criterion used for terminating the search. We cannot develop a theory that simultaneously covers every important version of inaccuracy without seriously detracting from the underlying simplicity of the algorithms discussed later. For this reason our general approach, which is admittedly more free-wheeling in spirit than necessary but thereby more transparent and less encumbered than a detailed account of inaccuracy, will be to analyze algorithms as if an accurate line search were made at every step, and then point out in side remarks and exercises the effect of inaccuracy.

Armijo's Rule

A practical and popular criterion for terminating a line search is Armijo's rule. The essential idea is that the rule should first guarantee that the selected α is not too large, and next it should not be too small. Let us define the function

$$\phi(\alpha) = f(\mathbf{x}_k + \alpha \mathbf{d}_k).$$

Armijo's rule is implemented by consideration of the function $\phi(0) + \varepsilon\phi'(0)\alpha$ for fixed ε, $0 < \varepsilon < 1$. This function is shown in Fig. 8.8a as the dashed line. A value of α is considered to be not too large if the corresponding function value lies below the dashed line; that is, if

$$\phi(\alpha) \leqslant \phi(0) + \varepsilon\phi'(0)\alpha. \tag{8.24}$$

To insure that α is not too small, a value $\eta > 1$ is selected, and α is then considered to be not too small if

$$\phi(\eta\alpha) > \phi(0) + \varepsilon\phi'(0)\eta\alpha.$$

This means that if α is increased by the factor η, it will fail to meet the test (8.24). The acceptable region defined by the Armijo rule is shown in Fig. 8.8a when $\eta = 2$ (there are also other rules can be adapted).

Sometimes in practice, the Armijo test is used to define a simplified line search technique that does not employ curve fitting methods. One begins with an arbitrary α. If it satisfies (8.24), it is repeatedly increased by $\eta(\eta = 2$ or $\eta = 10$ and $\varepsilon = .2$ are often used) until (8.24) is not satisfied, and then the penultimate α is selected. If, on the other hand, the original α does not satisfy (8.24), it is repeatedly divided by η until the resulting α does satisfy (8.24).

8.2 The Method of Steepest Descent

One of the oldest and most widely known methods for minimizing a function of several variables is the method of steepest descent (often referred to as the gradient method). The method is extremely important from a theoretical viewpoint, since it is one of the simplest for which a satisfactory analysis exists. More advanced algorithms are often motivated by an attempt to modify the basic steepest descent technique in such a way that the new algorithm will have superior convergence properties. The method of steepest descent remains, therefore, not only the technique most often first tried on a new problem but also the standard of reference against which other techniques are measured. The principles used for its analysis will be used throughout this book.

The Method

Let f have continuous first partial derivatives on E^n. We will frequently have need for the gradient vector of f and therefore we introduce some simplifying notation. The gradient $\nabla f(\mathbf{x})$ is, according to our conventions, defined as a n-dimensional *row* vector. For convenience we define the n-dimensional *column* vector $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})^T$. When there is no chance for ambiguity, we sometimes suppress the argument $\mathbf{x}$ and, for example, write $\mathbf{g}_k$ for $\mathbf{g}(\mathbf{x}_k) = \nabla f(\mathbf{x}_k)^T$.

The method of steepest descent is defined by the iterative algorithm

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k,$$

where stepsize α_k is a nonnegative scalar possibly minimizing $f(\mathbf{x}_k - \alpha \mathbf{g}_k)$. In words, from the point $\mathbf{x}_k$ we search along the direction of the negative gradient $-\mathbf{g}_k$ to a minimum point on this line; this minimum point is taken to be $\mathbf{x}_{k+1}$.

In formal terms, the overall algorithm $\mathbf{A} : E^n \to E^n$ which gives $\mathbf{x}_{k+1} \in \mathbf{A}(\mathbf{x}_k)$ can be decomposed in the form $\mathbf{A} = \mathbf{SG}$. Here $\mathbf{G} : E^n \to E^{2n}$ is defined by $\mathbf{G}(\mathbf{x}) = (\mathbf{x}, -\mathbf{g}(\mathbf{x}))$, giving the initial point and direction of a line search. This is followed by the line search $\mathbf{S} : E^{2n} \to E^n$ defined in Sect. 8.1.

Global Convergence and Convergence Speed

It was shown in Sect. 8.1 that $\mathbf{S}$ is closed if $\nabla f(\mathbf{x}) \neq \mathbf{0}$, and it is clear that $\mathbf{G}$ is continuous. Therefore, by Corollary 2 in Sect. 7.7 $\mathbf{A}$ is closed.

We define the solution set to be the points $\mathbf{x}$ where $\nabla f(\mathbf{x}) = \mathbf{0}$. Then $Z(\mathbf{x}) = f(\mathbf{x})$ is a descent function for $\mathbf{A}$, since for $\nabla f(\mathbf{x}) \neq \mathbf{0}$

$$\lim_{0 \leqslant \alpha < \infty} f(\mathbf{x} - \alpha \mathbf{g}(\mathbf{x})) < f(\mathbf{x}).$$

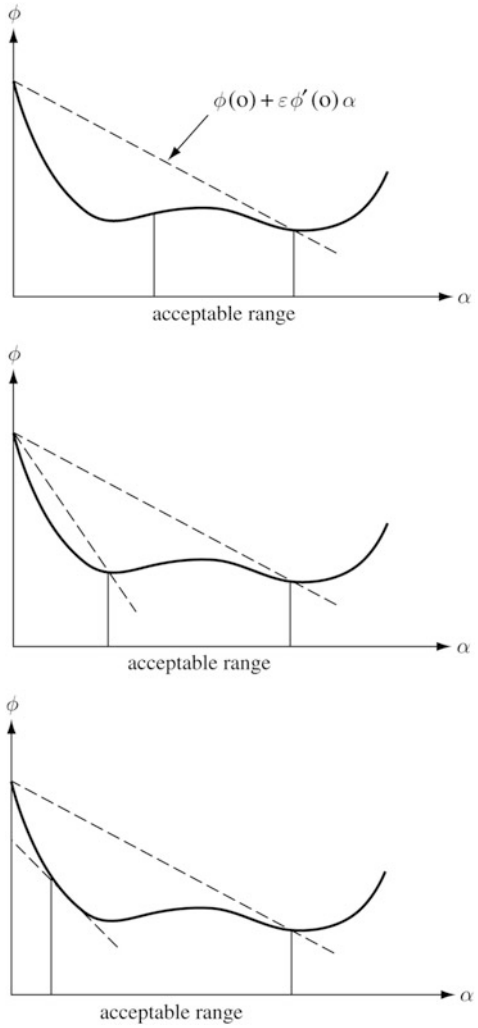

Fig. 8.8 Stopping rules. (**a**) Armijo rule. (**b**) Golden test. (**c**) Wolfe test

Thus by the Global Convergence Theorem, if the sequence $\{\mathbf{x}_k\}$ is bounded, it will have limit points and each of these is a solution. What about the convergence speed? Assume that $f(\mathbf{x})$ is convex and differentiable everywhere, admits a minimizer $\mathbf{x}^*$, and satisfies the (first-order) β-Lipschitz condition, that is, for any two points $\mathbf{x}$ and $\mathbf{y}$

$$|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})| \leq \beta|\mathbf{x} - \mathbf{y}|$$

for a positive real number β. Starting from any point $\mathbf{x}_0$, we consider the method of steepest descent with a fixed step size $\alpha_k = \frac{1}{\beta}$ for all k:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{\beta}\mathbf{g}_k = \mathbf{x}_k - \frac{1}{\beta}\nabla f(\mathbf{x}_k)^T. \tag{8.25}$$

We first prove a lemma.

Lemma 1. *Let $f(\boldsymbol{x})$ be differentiable everywhere and satisfy the (first-order) β-Lipschitz condition. Then, for any two points* $\mathbf{x}$ *and* $\mathbf{y}$

$$f(\boldsymbol{x}) - f(\boldsymbol{y}) - \nabla f(\boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y}) \le \frac{\beta}{2}|\boldsymbol{x} - \boldsymbol{y}|^2.$$

Then we prove

Theorem 1 (Steepest Descent—Lipschitz Convex Case). *Let $f(\boldsymbol{x})$ be convex and differentiable everywhere, satisfy the (first-order) β-Lipschitz condition, and admit a minimizer $\mathbf{x}^*$. Then, the method of steepest descent (8.25) generates a sequence of solutions $\mathbf{x}_k$ such that*

$$|\nabla f(\boldsymbol{x}_k)| \le \frac{\beta^2}{\sqrt{k(k+1)}}|\boldsymbol{x}_0 - \boldsymbol{x}^*|,$$

and

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \le \frac{\beta}{2(k+1)}|\boldsymbol{x}_0 - \boldsymbol{x}^*|^2.$$

Proof. Consider the function $g_x(\mathbf{y}) = f(\mathbf{y}) - \nabla f(\mathbf{x})\mathbf{y}$ for any given $\mathbf{x}$. Note that g_x is also convex and satisfies the β-Lipschitz condition. Moreover, $\mathbf{x}$ is the minimizer of $g_x(\mathbf{y})$ and $\nabla g_x(\mathbf{y}) = \nabla f(\mathbf{y}) - \nabla f(\mathbf{x})$.

Applying Lemma 1 to g_x and noting the relations of g_x and $f(\mathbf{x})$, we have

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{y}) - \nabla f(\mathbf{x})(\mathbf{x} - \mathbf{y}) &= g_x(\mathbf{x}) - g_x(\mathbf{y}) \\ &\le g_x(\mathbf{y} - \tfrac{1}{\beta}\nabla g_x(\mathbf{y})) - g_x(\mathbf{y}) \\ &\le \nabla g_x(\mathbf{y})(-\tfrac{1}{\beta}\nabla g_x(\mathbf{y})^T) + \tfrac{\beta}{2}\tfrac{1}{\beta^2}|\nabla g_x(\mathbf{y})|^2 \\ &= -\tfrac{1}{2\beta}|\nabla g_x(\mathbf{y})|^2 \\ &= -\tfrac{1}{2\beta}|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})|^2. \end{aligned} \tag{8.26}$$

Similarly, we have

$$f(\mathbf{y}) - f(\mathbf{x}) - \nabla f(\mathbf{y})(\mathbf{y} - \mathbf{x}) \le -\frac{1}{2\beta}|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})|^2.$$

Adding the above two derived inequalities, we have for any $\mathbf{x}$ and $\mathbf{y}$:

$$(\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}))(\mathbf{x} - \mathbf{y}) \ge \frac{1}{\beta}|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})|^2. \tag{8.27}$$

For simplification, in what follows let $\mathbf{d}_k = \mathbf{x}_k - \mathbf{x}^*$ and $\delta_k = [f(\mathbf{x}_k) - f(\mathbf{x}^*)] \ge 0$.

Now let $\mathbf{x} = \mathbf{x}_{k+1}$ and $\mathbf{y} = \mathbf{x}_k$ in (8.27). Then

$$-\frac{1}{\beta}(\mathbf{g}_k)^T(\mathbf{g}_{k+1} - \mathbf{g}_k) = (\mathbf{x}_{k+1} - \mathbf{x}_k)^T(\mathbf{g}_{k+1} - \mathbf{g}_k) \geq \frac{1}{\beta}|\mathbf{g}_{k+1} - \mathbf{g}_k|^2,$$

which leads to

$$|\mathbf{g}_{k+1}|^2 \leq (\mathbf{g}_{k+1})^T\mathbf{g}_k \leq |\mathbf{g}_{k+1}||\mathbf{g}_k|, \quad \text{that is} \quad |\mathbf{g}_{k+1}| \leq |\mathbf{g}_k|. \tag{8.28}$$

Inequality (8.28) implies that $|\mathbf{g}_k| = |\nabla f(\mathbf{x}_k)|$ is monotonically decreasing.

Applying inequality (8.26) for $\mathbf{x} = \mathbf{x}_k$ and $\mathbf{y} = \mathbf{x}^*$ and noting $\mathbf{g}^* = \mathbf{0}$ we have

$$\begin{aligned}
\delta_k &\leq (\mathbf{g}_k)^T\mathbf{d}_k - \tfrac{1}{2\beta}|\mathbf{g}_k|^2 \\
&= -\beta(\mathbf{x}_{k+1} - \mathbf{x}_k)\mathbf{d}_k - \tfrac{\beta}{2}|\mathbf{x}_{k+1} - \mathbf{x}_k|^2 \\
&= -\tfrac{\beta}{2}(|\mathbf{x}_{k+1} - \mathbf{x}_k|^2 + 2(\mathbf{x}_{k+1} - \mathbf{x}_k)^T\mathbf{d}_k) \\
&= -\tfrac{\beta}{2}(|\mathbf{d}_{k+1} - \mathbf{d}_k|^2 + 2(\mathbf{d}_{k+1} - \mathbf{d}_k)^T\mathbf{d}_k) \\
&= \tfrac{\beta}{2}(|\mathbf{d}_k|^2 - |\mathbf{d}_{k+1}|^2).
\end{aligned} \tag{8.29}$$

Summing up (8.29) from 0 to k, we have

$$\sum_{l=0}^{k} \delta_l \leq \frac{\beta}{2}(|\mathbf{d}_0|^2 - |\mathbf{d}_{k+1}|^2) \leq \frac{\beta}{2}|\mathbf{d}_0|^2. \tag{8.30}$$

Using (8.26) again for $\mathbf{x} = \mathbf{x}_{k+1}$ and $\mathbf{y} = \mathbf{x}_k$ and noting (8.25) we have

$$\begin{aligned}
\delta_{k+1} - \delta_k &= f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) \\
&\leq \mathbf{g}_{k+1}^T(-\tfrac{1}{\beta}\mathbf{g}_k) - \tfrac{1}{2\beta}|\mathbf{g}_{k+1} - \mathbf{g}_k|^2 \\
&= -\tfrac{1}{2\beta}(|\mathbf{g}_{k+1}|^2 + |\mathbf{g}_k|^2).
\end{aligned} \tag{8.31}$$

Noting (8.31) holts for all k, we have

$$\begin{aligned}
\textstyle\sum_{l=0}^{k} \delta_l &= \textstyle\sum_{l=0}^{k} \delta_l(l + 1 - l) \\
&= \textstyle\sum_{l=0}^{k} \delta_l(l + 1) - \sum_{l=0}^{k} \delta_l l \\
&= \textstyle\sum_{l=1}^{k+1} \delta_{l-1} l - \sum_{l=1}^{k} \delta_l l \\
&= \delta_k(k + 1) + \textstyle\sum_{l=1}^{k} (\delta_{l-1} - \delta_l) l \\
&\geq \delta_k(k + 1) + \textstyle\sum_{l=1}^{k} \frac{l}{2\beta}(|\mathbf{g}_l|^2 + |\mathbf{g}_{l-1}|^2) \\
&\geq \delta_k(k + 1) + \tfrac{k(k+1)}{2\beta}|\mathbf{g}_k|^2
\end{aligned}$$

where the last inequality comes $|\mathbf{g}_k| = |\nabla f(\mathbf{x}_k)|$ is *monotonically* decreasing.

Using (8.30) we finally have

$$(k + 1)\delta_k + \frac{k(k + 1)}{2\beta}|\mathbf{g}_k|^2 \leq \frac{\beta}{2}|\mathbf{d}_0|^2. \tag{8.32}$$

Inequality (8.32), from $\delta_k = f(\mathbf{x}_k) - f(\mathbf{x}^*) \geq 0$ and $\mathbf{d}_0 = \mathbf{x}_0 - \mathbf{x}^*$, proves the desired bounds. ∎

Theorem 1 implies that the convergence speed of the steepest descent method is arithmetic.

The Quadratic Case

When $f(\mathbf{x})$ is strongly convex, the convergence speed can be increased from arithmetic to geometric or linear convergence. Since all of the important convergence characteristics of the method of steepest descent are revealed by an investigation of the method when applied to quadratic problems, we focus here on

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{x}^T\mathbf{b}, \tag{8.33}$$

where $\mathbf{Q}$ is a positive definite symmetric $n \times n$ matrix. Since $\mathbf{Q}$ is positive definite, all of its eigenvalues are positive. We assume that these eigenvalues are ordered: $0 < a = \lambda_1 \leqslant \lambda_2 \ldots \leqslant \lambda_n = A$. With $\mathbf{Q}$ positive definite, it follows (from Proposition 5, Sect. 7.4) that f is strictly convex.

The unique minimum point of f can be found directly, by setting the gradient to zero, as the vector $\mathbf{x}^*$ satisfying

$$\mathbf{Q}\mathbf{x}^* = \mathbf{b}. \tag{8.34}$$

Moreover, introducing the function

$$E(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T\mathbf{Q}(\mathbf{x} - \mathbf{x}^*), \tag{8.35}$$

we have $E(\mathbf{x}) = f(\mathbf{x}) + (1/2)\mathbf{x}^{*T}\mathbf{Q}\mathbf{x}^*$, which shows that the function E differs from f only by a constant. For many purposes then, it will be convenient to consider that we are minimizing E rather than f.

The gradient (of both f and E) is given explicitly by

$$\mathbf{g}(\mathbf{x}) = \mathbf{Q}\mathbf{x} - \mathbf{b}. \tag{8.36}$$

Thus the method of steepest descent can be expressed as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k\mathbf{g}_k, \tag{8.37}$$

where $\mathbf{g}_k = \mathbf{Q}\mathbf{x}_k - \mathbf{b}$ and where α_k minimizes $f(\mathbf{x}_k - \alpha\mathbf{g}_k)$. We can, however, in this special case, determine the value of α_k explicitly. We have, by definition (8.33),

$$f(\mathbf{x}_k - \alpha\mathbf{g}_k) = \frac{1}{2}(\mathbf{x}_k - \alpha\mathbf{g}_k)^T\mathbf{Q}(\mathbf{x}_k - \alpha\mathbf{g}_k) - (\mathbf{x}_k - \alpha\mathbf{g}_k)^T\mathbf{b},$$

which (as can be found by differentiating with respect to α) is minimized at

$$\alpha_k = \frac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T\mathbf{Q}\mathbf{g}_k}. \tag{8.38}$$

Hence the method of steepest descent (8.37) takes the explicit form

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \left(\frac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T\mathbf{Q}\mathbf{g}_k}\right)\mathbf{g}_k, \tag{8.39}$$

where $\mathbf{g}_k = \mathbf{Q}\mathbf{x}_k - \mathbf{b}$.

The function f and the steepest descent process can be illustrated as in Fig. 8.9 by showing contours of constant values of f and a typical sequence developed by the process. The contours of f are n-dimensional ellipsoids with axes in the directions of the n-mutually orthogonal eigenvectors of $\mathbf{Q}$. The axis corresponding to the ith eigenvector has length proportional to $1/\lambda_i$. We now analyze this process and show that the rate of convergence depends on the ratio of the lengths of the axes of the elliptical contours of f, that is, on the eccentricity of the ellipsoids.

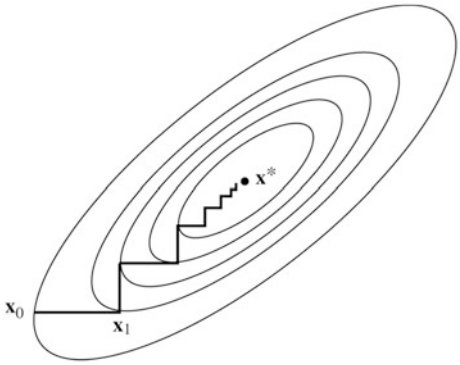

Fig. 8.9 Steepest descent

Lemma 2. *The iterative process (8.39) satisfies*

$$E(\boldsymbol{x}_{k+1}) = \left\{1 - \frac{(\boldsymbol{g}_k^T \boldsymbol{g}_k)^2}{(\boldsymbol{g}_k^T \boldsymbol{Q} \boldsymbol{g}_k)(\boldsymbol{g}_k^T \boldsymbol{Q}^{-1} \boldsymbol{g}_k)}\right\} E(\boldsymbol{x}_k). \tag{8.40}$$

Proof. The proof is by direct computation. We have, setting $\mathbf{y}_k = \mathbf{x}_k - \mathbf{x}^*$,

$$\frac{E(\mathbf{x}_k) - E(\mathbf{x}_{k+1})}{E(\mathbf{x}_k)} = \frac{2\alpha_k \mathbf{g}_k^T \mathbf{Q} \mathbf{y}_k - \alpha_k^2 \mathbf{g}_k^T \mathbf{Q} \mathbf{g}_k}{\mathbf{y}_k^T \mathbf{Q} \mathbf{y}_k}.$$

Using $\mathbf{g}_k = \mathbf{Q}\mathbf{y}_k$ we have

$$\begin{aligned} \frac{E(\mathbf{x}_k) - E(\mathbf{x}_{k+1})}{E(\mathbf{x}_k)} &= \frac{\frac{2(\mathbf{g}_k^T \mathbf{g}_k)^2}{(\mathbf{g}_k^T \mathbf{Q} \mathbf{g}_k)} - \frac{(\mathbf{g}_k^T \mathbf{g}_k)^2}{(\mathbf{g}_k^T \mathbf{Q} \mathbf{g}_k)}}{\mathbf{g}_k^T \mathbf{Q}^{-1} \mathbf{g}_k} \\ &= \frac{(\mathbf{g}_k^T \mathbf{g}_k)^2}{(\mathbf{g}_k^T \mathbf{Q} \mathbf{g}_k)(\mathbf{g}_k^T \mathbf{Q}^{-1} \mathbf{g}_k)}. \end{aligned}$$

∎

In order to obtain a bound on the rate of convergence, we need a bound on the right-hand side of (8.40). The best bound is due to Kantorovich and his lemma, stated below, is a useful general tool in convergence analysis.

Kantorovich inequality: *Let* $\mathbf{Q}$ *be a positive definite symmetric* $n \times n$ *matrix. For any vector* $\mathbf{x}$ *there holds*

$$\frac{(\mathbf{x}^T\mathbf{x})^2}{(\mathbf{x}^T\mathbf{Q}\mathbf{x})(\mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x})} \geqslant \frac{4aA}{(a+A)^2}, \tag{8.41}$$

where a *and* $\mathbf{A}$ *are, respectively, the smallest and largest eigenvalues of* $\mathbf{Q}$.

Proof. Let the eigenvalues $\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n$ of $\mathbf{Q}$ satisfy

$$0 < a = \lambda_1 \leqslant \lambda_2 \ldots \leqslant \lambda_n = A.$$

By an appropriate change of coordinates the matrix $\mathbf{Q}$ becomes diagonal with diagonal $(\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_n)$. In this coordinate system we have

$$\frac{(\mathbf{x}^T\mathbf{x})^2}{(\mathbf{x}^T\mathbf{Q}\mathbf{x})(\mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x})} = \frac{(\sum_{i=1}^n x_i^2)^2}{(\sum_{i=1}^n \lambda_i x_i^2)(\sum_{i=1}^n (x_i^2/\lambda_i))},$$

which can be written as

$$\frac{(\mathbf{x}^T\mathbf{x})^2}{(\mathbf{x}^T\mathbf{Q}\mathbf{x})(\mathbf{x}^T\mathbf{Q}^{-1}\mathbf{x})} = \frac{1/\sum_{i=1}^n \xi_i\lambda_i}{\sum_{i=1}^n (\xi_i/\lambda_i)} \equiv \frac{\phi(\xi)}{\psi(\xi)},$$

where $\xi_i = x_i^2/\sum_{i=1}^n x_i^2$. We have converted the expression to the ratio of two functions involving convex combinations; one a combination of λ_i's; the other a combination of $1/\lambda_i$'s. The situation is shown pictorially in Fig. 8.10. The curve in the figure represents the function $1/\lambda$. Since $\sum_{i=1}^n \xi_i\lambda_i$ is a point between λ_1 and λ_n, the value of $\phi(\xi)$ is a point on the curve. On the other hand, the value of $\psi(\xi)$ is a convex combination of points on the curve and its value corresponds to a point in the shaded region. For the same vector ξ both functions are represented by points on the same vertical line. The minimum value of this ratio is achieved for some $\lambda = \xi_1\lambda_1 + \xi_n\lambda_n$, with $\xi_1 + \xi_n = 1$. Using the relation $\xi_1/\lambda_1 + \xi_n/\lambda_n = (\lambda_1 + \lambda_n - \xi_1\lambda_1 - \xi_n\lambda_n)/\lambda_1\lambda_n$, an appropriate bound is

$$\frac{\phi(\xi)}{\psi(\xi)} \geqslant \lim_{\lambda_1 \leqslant \lambda \leqslant \lambda_n} \frac{(1/\lambda)}{(\lambda_1 + \lambda_n - \lambda)/(\lambda_1\lambda_n)}.$$

The minimum is achieved at $\lambda = (\lambda_1 + \lambda_n)/2$, yielding

$$\frac{\phi(\xi)}{\psi(\xi)} \geqslant \frac{4\lambda_1\lambda_n}{(\lambda_1 + \lambda_n)^2}. \blacksquare$$

Combining the above two lemmas, we obtain the central result on the convergence of the method of steepest descent.

Theorem 2 (Steepest Descent—Quadratic Case). *For any* $\mathbf{x}_0 \in E^n$ *the method of steepest descent (8.39) converges to the unique minimum point* $\mathbf{x}^*$ *of* f *Furthermore, with* $E(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}^*)^T\boldsymbol{Q}(\boldsymbol{x} - \boldsymbol{x}^*)$, *there holds at every step* k

$$E(\boldsymbol{x}_{k+1}) \leqslant \left(\frac{A-a}{A+a}\right)^2 E(\boldsymbol{x}_k). \tag{8.42}$$

Proof. By Lemma 2 and the Kantorovich inequality

$$E(\mathbf{x}_{k+1}) \leqslant \left\{1 - \frac{4aA}{(A+a)^2}\right\} E(\mathbf{x}_k) = \left(\frac{A-a}{A+a}\right)^2 E(\mathbf{x}_k).$$

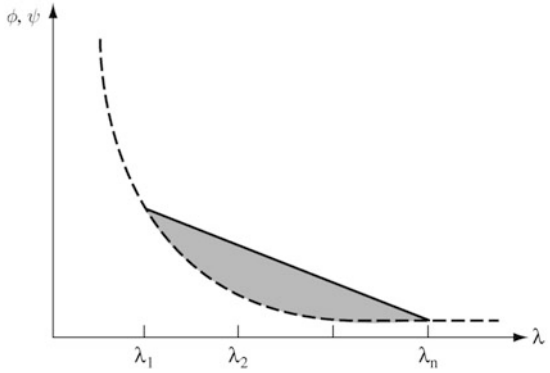

Fig. 8.10 Kantorovich inequality

It follows immediately that $E(\mathbf{x}_k) \to 0$ and hence, since $\mathbf{Q}$ is positive definite, that $\mathbf{x}_k \to \mathbf{x}^*$. ∎

Roughly speaking, the above theorem says that the convergence rate of steepest descent is slowed as the contours of f become more eccentric. If $a = A$, corresponding to circular contours, convergence occurs in a single step. Note, however, that even if $n-1$ of the n eigenvalues are equal and the remaining one is a great distance from these, convergence will be slow, and hence a single abnormal eigenvalue can destroy the effectiveness of steepest descent.

In the terminology introduced in Sect. 7.8, the above theorem states that with respect to the error function E (or equivalently f) the method of steepest descent converges linearly with a ratio no greater than $[(A-a)/(A+a)]^2$. The actual rate depends on the initial point $\mathbf{x}_0$. However, for some initial points the bound is actually achieved. Furthermore, it has been shown by Akaike that, if the ratio is unfavorable, the process is very likely to converge at a rate close to the bound. Thus, somewhat loosely but with reasonable justification, we say that the convergence ratio of steepest descent *is* $[(A-a)/(A+a)]^2$.

It should be noted that the convergence rate actually depends only on the ratio $r = A/a$ of the largest to the smallest eigenvalue. Thus the convergence ratio is

$$\left(\frac{A-a}{A+a}\right)^2 = \left(\frac{r-1}{r+1}\right)^2,$$

which clearly shows that convergence is slowed as r increases. The ratio r, which is the single number associated with the matrix $\mathbf{Q}$ that characterizes convergence, is often called the *condition number* of the matrix.

Example. Let us take

$$\mathbf{Q} = \begin{bmatrix} 0.78 & -0.02 & -0.12 & -0.14 \\ -0.02 & 0.86 & -0.04 & 0.06 \\ -0.12 & -0.04 & 0.72 & -0.08 \\ -0.14 & 0.06 & -0.08 & 0.74 \end{bmatrix}$$

$$\mathbf{b} = (0.76,\ 0.08,\ 1.12,\ 0.68).$$

For this matrix it can be calculated that $a = 0.52$, $A = 0.94$ and hence $r = 1.8$. This is a very favorable condition number and leads to the convergence ratio $[(A - a)/(A + a)]^2 = 0.081$. Thus each iteration will reduce the error in the objective by more than a factor of ten; or, equivalently, each iteration will add about one more digit of accuracy. Indeed, starting from the origin the sequence of values obtained by steepest descent as shown in Table 8.1 is consistent with this estimate.

The Nonquadratic Case

For nonquadratic functions, we expect that steepest descent will also do reasonably well if the condition number is modest. Fortunately, we are able to establish estimates of the progress of the method when the Hessian matrix is always positive definite. Specifically, we assume that the Hessian matrix is bounded above and below as $\mathbf{aI} \leqslant \mathbf{F}(\bar{\mathbf{x}}) \leqslant A\mathbf{I}$. (Thus f is *strongly* convex.) We present three analyses:

Table 8.1 Solution to Example

Step k	$f(\mathbf{x}_k)$
0	0
1	−2.1563625
2	−2.1744062
3	−2.1746440
4	−2.1746585
5	−2.1746595
6	−2.1746595

Solution point $\mathbf{x}^*$ = (1.534965, 0.1220097, 1.975156, 1.412954)

1. **Exact Line Search.** Given a point $\mathbf{x_k}$, we have for any α

$$f(\mathbf{x}_k - \alpha\mathbf{g}(\mathbf{x}_k)) \leqslant f(\mathbf{x}_k)) - \alpha\mathbf{g}(\mathbf{x}_k)^T\mathbf{g}(\mathbf{x}_k) + \frac{A\alpha^2}{2}\mathbf{g}(\mathbf{x}_k)^T\mathbf{g}(\mathbf{x}_k). \tag{8.43}$$

Minimizing both sides separately with respect to α the inequality will hold for the two minima. The minimum of the left hand side is $f(\mathbf{x}_{k+1})$. The minimum of the right hand side occurs at $\alpha = 1/A$, yielding the result

$$f(\mathbf{x}_{k+1}) \leqslant f(\mathbf{x}_k) - \frac{1}{2A}|\mathbf{g}(\mathbf{x}_k)|^2.$$

where $|\mathbf{g}(\mathbf{x_k})|^2 \equiv \mathbf{g}(\mathbf{x_k})^T\mathbf{g}(\mathbf{x_k})$. Subtracting the optimal value $f^* = f(\mathbf{x}^*)$ from both sides produces

$$f(\mathbf{x_{k+1}}) - \mathbf{f}^* \leqslant \mathbf{f}(\mathbf{x}_k) - \mathbf{f}^* - \frac{1}{2\mathbf{A}}|\mathbf{g}(\mathbf{x}_k)|^2. \tag{8.44}$$

In a similar way, for any $\mathbf{x}$ there holds

$$f(\mathbf{x}) \geqslant f(\mathbf{x_k}) + \mathbf{g}(\mathbf{x_k})^T(\mathbf{x} - \mathbf{x_k}) + \frac{\mathbf{a}}{2}|\mathbf{x} - \mathbf{x_k}|^2.$$

Again we can minimize both sides separately. The minimum of the left hand side is f^* the optimal solution value. Minimizing the right hand side leads to the quadratic optimization problem. The solution is $\overline{\mathbf{x}} = \mathbf{x}_k - \mathbf{g}(\mathbf{x}_k)/a$. Substituting this $\overline{\mathbf{x}}$ in the right hand side of the inequality gives

$$f^* \geqslant f(\mathbf{x}_k) - \frac{1}{2a}|\mathbf{g}(\mathbf{x}_k)|^2. \tag{8.45}$$

From (8.45) we have

$$-|\mathbf{g}(\mathbf{x}_k)|^2 \leqslant 2a[f^* - f(\mathbf{x}_k)]. \tag{8.46}$$

Substituting this in (8.44) gives

$$f(\mathbf{x}_{k+1}) - f^* \leqslant (1 - a/A)[f(\mathbf{x}_k) - f^*]. \tag{8.47}$$

This shows that the method of steepest descent makes progress even when it is not close to the solution.

2. **Other Stopping Criteria.** As an example of how other stopping criteria can be treated, we examine the rate of convergence when using Amijo's rule with $\varepsilon < 0.5$ and $\eta > 1$. Note first that the inequality $t \geqslant t^2$ for $0 \leqslant t \leqslant 1$ implies by a change of variable that

$$-\alpha + \frac{\alpha^2 A}{2} \leq -\alpha/2$$

for $0 \leqslant \alpha \leqslant 1/A$. Then using (8.43) we have that for $\alpha < 1/A$

$$\begin{aligned} f(\mathbf{x}_k - \alpha\mathbf{g}(\mathbf{x}_k)) &\leq f(\mathbf{x}_k) - \alpha|\mathbf{g}(\mathbf{x}_k)|^2 + 0.5\alpha^2 A|\mathbf{g}(\mathbf{x}_l)|^2 \\ &\leq f(\mathbf{x}_k) - 0.5\alpha|\mathbf{g}(\mathbf{x}_k)|^2 \\ &< f(\mathbf{x}_k) - \varepsilon\alpha|\mathbf{g}(\mathbf{x}_k)|^2 \end{aligned}$$

since $\varepsilon < 0.5$. This means that the first part of the stopping criterion is satisfied for $\alpha < 1/A$.

The second part of the stopping criterion states that $\eta\alpha$ does not satisfy the first criterion and thus the final α must satisfy $\alpha \geq 1/(\eta A)$. Therefore the inequality of the first part of the criterion implies

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k) - \frac{\varepsilon}{\eta A}|\mathbf{g}(\mathbf{x}_k)|^2.$$

Subtracting f^* from both sides,

$$f(\mathbf{x}_{k+1}) - f^* \leq f(\mathbf{x}_k) - f^* - \frac{\varepsilon}{\eta A}|\mathbf{g}(\mathbf{x}_k)|^2.$$

Finally, using (8.46) we obtain

$$f(\mathbf{x}_{k+1}) - f^* \leq [1 - (2\varepsilon a/\eta A)](f(\mathbf{x}_k) - f^*).$$

Clearly $2\varepsilon a/\eta A < 1$ and hence there is linear convergence. Notice if that in fact ε is chosen very close to 0.5 and η is chosen very close to 1, then the stopping condition demands that the α be restricted to a very small range, and the estimated rate of convergence is very close to the estimate obtained above for exact line search.

3. **Asymptotic Convergence.** We expect that as the points generated by steepest descent approach the solution point, the convergence characteristics will be close to those inherent for quadratic functions. This is indeed the case.

The general procedure for proving such a result, which is applicable to most methods having unity order of convergence, is to use the Hessian of the objective at the solution point as if it were the **Q** matrix of a quadratic problem. The particular theorem stated below is a special case of a theorem in Sect. 12.5 so we do not prove it here; but it illustrates the generalizability of an analysis of quadratic problems.

Theorem. *Suppose f is defined on E^n, has continuous second partial derivatives, and has a relative minimum at $\mathbf{x}^*$. Suppose further that the Hessian matrix of f, $\boldsymbol{F}(\boldsymbol{x}^*)$, has smallest eigenvalue $a > 0$ and largest eigenvalue $A > 0$. If $\{\boldsymbol{x}_k\}$ is a sequence generated by the method of steepest descent that converges to $\mathbf{x}^*$, then the sequence of objective values $\{f(\boldsymbol{x}_k)\}$ converges to $f(\mathbf{x}^*)$ linearly with a convergence ratio no greater than $[(A - a)/(A + a)]^2$.*

8.3 Applications of the Convergence Theory

Now that the basic convergence theory, as represented by the formula (8.42) for the rate of convergence, has been developed and demonstrated to actually characterize the behavior of steepest descent, it is appropriate to illustrate how the theory can be used. Generally, we do *not* suggest that one compute the numerical value of the formula—since it involves eigenvalues, or ratios of eigenvalues, that are not easily determined. Nevertheless, the formula itself is of immense practical importance, since it allows one to theoretically compare various situations. Without such a theory, one would be forced to rely completely on experimental comparisons.

Application 1 (Solution of Gradient Equation). One approach to the minimization of a function f is to consider solving the equations $\nabla f(\mathbf{x}) = \mathbf{0}$ that represent the necessary conditions. It has been proposed that these equations could be solved by applying steepest descent to the function $h(\mathbf{x}) = |\nabla f(\mathbf{x})|^2$. One advantage of this method is that the minimum value is known. We ask whether this method is likely to be faster or slower than the application of steepest descent to the original function f itself.

For simplicity we consider only the case where f is quadratic. Thus let $f(\mathbf{x}) = (1/2)\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{b}^T\mathbf{x}$. Then the gradient of f is $\mathbf{g}(\mathbf{x}) = \mathbf{Q}\mathbf{x} - \mathbf{b}$, and $h(\mathbf{x}) = |\mathbf{g}(\mathbf{x})|^2 = \mathbf{x}^T\mathbf{Q}^2\mathbf{x} - 2\mathbf{x}^T\mathbf{Q}\mathbf{b} + \mathbf{b}^T\mathbf{b}$. Thus $h(\mathbf{x})$ is itself a quadratic function. The rate of convergence of steepest descent applied to h will be governed by the eigenvalues of the matrix $\mathbf{Q}^2$. In particular the rate will be

$$\left(\frac{\bar{r}-1}{\bar{r}+1}\right)^2,$$

where $\bar{r}$ is the condition number of the matrix $\mathbf{Q}^2$. However, the eigenvalues of $\mathbf{Q}^2$ are the squares of those of $\mathbf{Q}$ itself, so $\bar{r} = r^2$, where r is the condition number of $\mathbf{Q}$, and it is clear that the convergence rate for the proposed method will be worse than for steepest descent applied to the original function.

We can go further and actually estimate how much slower the proposed method is likely to be. If r is large, we have

$$\text{steepest descent rate} = \left(\frac{r-1}{r+1}\right)^2 \simeq (1-1/r)^4$$

$$\text{proposed method rate} = \left(\frac{r^2-1}{r^2+1}\right)^2 \simeq (1-1/r^2)^4.$$

Since $(1-1/r^2)^r \simeq 1-1/r$, it follows that it takes about r steps of the new method to equal one step of ordinary steepest descent. We conclude that if the original problem is difficult to solve with steepest descent, the proposed method will be quite a bit worse.

Application 2 (Penalty Methods). Let us briefly consider a problem with a single constraint:

$$\begin{aligned} &\text{minimize } f(\mathbf{x}) \\ &\text{subject to } h(\mathbf{x}) = 0. \end{aligned} \tag{8.48}$$

One method for approaching this problem is to convert it (at least approximately) to the unconstrained problem

$$\text{minimize } f(\mathbf{x}) + \frac{1}{2}\mu h(\mathbf{x})^2, \tag{8.49}$$

where μ is a (large) penalty coefficient. Because of the penalty, the solution to (8.49) will tend to have a small $h(\mathbf{x})$. Problem (8.49) can be solved as an unconstrained problem by the method of steepest descent. How will this behave?

For simplicity let us consider the case where f is quadratic and h is linear. Specifically, we consider the problem

$$\begin{aligned} &\text{minimize } \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{b}^T\mathbf{x} \\ &\text{subject to } \mathbf{c}^T\mathbf{x} = 0. \end{aligned} \tag{8.50}$$

The objective of the associated penalty problem is $(1/2)\{\mathbf{x}^T\mathbf{Q}\mathbf{x}+\mu\mathbf{x}^T\mathbf{c}\mathbf{c}^T\mathbf{x}\}-\mathbf{b}^T\mathbf{x}$. The quadratic form associated with this objective is defined by the matrix $\mathbf{Q}+\mu\mathbf{c}\mathbf{c}^T$ and, accordingly, the convergence rate of steepest descent will be governed by the condition number of this matrix. This matrix is the original matrix $\mathbf{Q}$ with a large rank-one matrix added. It should be fairly clear[†] that this addition will cause one eigenvalue of the matrix to be large (on the order of μ). Thus the condition number is roughly proportional to μ. Therefore, as one increases μ in order to get an accurate solution to the original constrained problem, the rate of convergence becomes extremely poor. We conclude that the penalty function method used in this simplistic way with steepest descent will not be very effective. (Penalty functions, and how to minimize them more rapidly, are considered in detail in Chap. 11.)

Scaling

The performance of the method of steepest descent is dependent on the particular choice of variables $\mathbf{x}$ used to define the problem. A new choice may substantially alter the convergence characteristics.

Suppose that $\mathbf{T}$ is an invertible $n \times n$ matrix. We can then represent points in E^n either by the standard vector $\mathbf{x}$ or by $\mathbf{y}$ where $\mathbf{T}\mathbf{y} = \mathbf{x}$. The problem of finding $\mathbf{x}$ to minimize $f(\mathbf{x})$ is equivalent to that of finding $\mathbf{y}$ to minimize $h(\mathbf{y}) = f(\mathbf{T}\mathbf{y})$. Using $\mathbf{y}$ as the underlying set of variables, we then have

$$\nabla h = \nabla f \mathbf{T}, \tag{8.51}$$

where ∇f is the gradient of f with respect to $\mathbf{x}$. Thus, using steepest descent, the direction of search will be

$$\nabla \mathbf{y} = -\mathbf{T}^T \nabla f^T, \tag{8.52}$$

which in the original variables is

$$\Delta \mathbf{x} = -\mathbf{T}\mathbf{T}^T \nabla f^T. \tag{8.53}$$

[†]See the Interlocking Eigenvalues Lemma in Sect. 10.6 for a proof that only one eigenvalue becomes large.

Thus we see that the change of variables changes the direction of search.

The rate of convergence of steepest descent with respect to $\mathbf{y}$ will be determined by the eigenvalues of the Hessian of the objective, taken with respect to $\mathbf{y}$. That Hessian is

$$\nabla^2 h(\mathbf{y}) \equiv \mathbf{H}(\mathbf{y}) = \mathbf{T}^T \mathbf{F}(\mathbf{T}\mathbf{y})\mathbf{T}.$$

Thus, if $\mathbf{x}^* = \mathbf{T}\mathbf{y}^*$ is the solution point, the rate of convergence is governed by the matrix

$$\mathbf{H}(\mathbf{y}^*) = \mathbf{T}^T \mathbf{F}(\mathbf{x}^*)\mathbf{T}. \tag{8.54}$$

Very little can be said in comparison of the convergence ratio associated with $\mathbf{H}$ and that of $\mathbf{F}$. If $\mathbf{T}$ is an orthonormal matrix, corresponding to $\mathbf{y}$ being defined from $\mathbf{x}$ by a simple rotation of coordinates, then $\mathbf{T}^T\mathbf{T} = \mathbf{I}$, and we see from (8.48) that the directions remain unchanged and the eigenvalues of $\mathbf{H}$ are the same as those of $\mathbf{F}$.

In general, before attacking a problem with steepest descent, it is desirable, if it is feasible, to introduce a change of variables that leads to a more favorable eigenvalue structure. Usually the only kind of transformation that is at all practical is one having $\mathbf{T}$ equal to a diagonal matrix, corresponding to the introduction of scale factors on each of the variables. One should strive, in doing this, to make the second derivatives with respect to each variable roughly the same. Although appropriate scaling can potentially lead to substantial payoff in terms of enhanced convergence rate, we largely ignore this possibility in our discussions of steepest descent. However, see the next application for a situation that frequently occurs.

Application 3 (Program Design). In applied work it is extremely rare that one solves just a single optimization problem of a given type. It is far more usual that once a problem is coded for computer solution, it will be solved repeatedly for various parameter values. Thus, for example, if one is seeking to find the optimal production plan (as in Example 2 of Sect. 7.2), the problem will be solved for the different values of the input prices. Similarly, other optimization problems will be solved under various assumptions and constraint values. It is for this reason that speed of convergence and convergence analysis is so important. One wants a program that can be used efficiently. In many such situations, the effort devoted to proper scaling repays itself, not with the first execution, but in the long run.

As a simple illustration consider the problem of minimizing the function

$$f(x) = x^2 - 5xy + y^4 - ax - by.$$

It is desirable to obtain solutions quickly for different values of the parameters a and b. We begin with the values $a = 25,\ b = 8$.

The result of steepest descent applied to this problem directly is shown in Table 8.2, column (a). It requires eighty iterations for convergence, which could be regarded as disappointing.

The reason for this poor performance is revealed by examining the Hessian matrix

$$\mathbf{F} = \begin{bmatrix} 2 & -5 \\ -5 & 12y^2 \end{bmatrix}$$

Using the results of our first experiment, we know that $y = 3$. Hence the diagonal elements of the Hessian, at the solution, differ by a factor of 54. (In fact, the condition number is about 61.) As a simple remedy we scale the problem by replacing the variable y by $z = ty$. The new lower right-corner term of the Hessian then becomes $12z^2/t^4$, which has magnitude $12 \times t^2 \times 3^2/t^4 = 108/t^2$. Thus we might put $t = 7$ in order to make the two diagonal terms approximately equal. The result of applying steepest descent to the problem scaled this way is shown in Table 8.2, column (b). (This superior performance is in accordance with our general theory, since the condition number of the scaled problem is about two.) For other nearby values of a and b, similar speeds will be attained.

Table 8.2 Solution to scaling application

Iteration no.	Value of f (a) Unscaled	(b) Scaled
0	0.0000	0.0000
1	−230.9958	−162.2000
2	−256.4042	−289.3124
4	−293.1705	−341.9802
6	−313.3619	−342.9865
8	−324.9978	−342.9998
9	−329.0408	−343.0000
15	−339.6124	
20	−341.9022	
25	−342.6004	
30	−342.8372	
35	−342.9275	
40	−342.9650	
45	−342.9825	
50	−342.9909	
55	−342.9951	
60	−342.9971	Solution
65	−342.9883	$x = 20.0$
70	−342.9990	$y = 3.0$
75	−342.9994	
80	−342.9997	

8.4 Accelerated Steepest Descent

There is an *accelerated* steepest descent method that works as follows:

$$\lambda^0 = 0,\ \lambda_{k+1} = \frac{1+\sqrt{1+4(\lambda_k)^2}}{2},\ \alpha_k = \frac{1-\lambda_k}{\lambda_{k+1}}, \tag{8.55}$$

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{x}_k - \tfrac{1}{\beta}\nabla f(\mathbf{x}_k)^T, \quad \mathbf{x}_{k+1} = (1-\alpha_k)\tilde{\mathbf{x}}_{k+1} + \alpha_k \tilde{\mathbf{x}}_k. \tag{8.56}$$

Note that $(\lambda_k)^2 = \lambda_{k+1}(\lambda_{k+1} - 1)$, $\lambda_k > k/2$ and $\alpha_k \leq 0$. One can prove:

Theorem (Accelerated Steepest Descent). *Let $f(\boldsymbol{x})$ be convex and differentiable everywhere, satisfies the (first-order) β-Lipschitz condition, and admits a minimizer $\mathbf{x}^*$. Then, the method of accelerated steepest descent generates a sequence of solutions such that*

$$f(\tilde{\boldsymbol{x}}_{k+1}) - f(\boldsymbol{x}^*) \leq \frac{2\beta}{k^2}|\boldsymbol{x}^0 - \boldsymbol{x}^*|^2, \ \forall k \geq 1.$$

Proof. We now let $\mathbf{d}_k = \lambda_k \mathbf{x}_k - (\lambda_k - 1)\tilde{\mathbf{x}}_k - \mathbf{x}^*$, and $\delta_k = f(\tilde{\mathbf{x}}_k) - f(\mathbf{x}^*)(\geq 0)$.

Applying Lemma 1 for $\mathbf{x} = \tilde{\mathbf{x}}_{k+1}$ and $\mathbf{y} = \tilde{\mathbf{x}}_k$, convexity of f and (8.56), we have

$$\begin{aligned}
\delta_{k+1} - \delta_k &= f(\tilde{\mathbf{x}}_{k+1}) - f(\mathbf{x}_k) + f(\mathbf{x}_k) - f(\tilde{\mathbf{x}}_k) \\
&\leq -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 + f(\mathbf{x}_k) - f(\tilde{\mathbf{x}}_k) \\
&\leq -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 + (\mathbf{g}_k)^T(\mathbf{x}_k - \tilde{\mathbf{x}}_k) \\
&= -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 - \beta(\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k)^T(\mathbf{x}_k - \tilde{\mathbf{x}}_k).
\end{aligned} \tag{8.57}$$

Applying Lemma 1 for $\mathbf{x} = \tilde{\mathbf{x}}_{k+1}$ and $\mathbf{y} = \mathbf{x}^*$, convexity of f and (8.56), we have

$$\begin{aligned}
\delta_{k+1} &= f(\tilde{\mathbf{x}}_{k+1}) - f(\mathbf{x}_k) + f(\mathbf{x}_k) - f(\mathbf{x}^*) \\
&\leq -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 + f(\mathbf{x}_k) - f(\mathbf{x}^*) \\
&\leq -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 + (\mathbf{g}_k)^T(\mathbf{x}_k - \mathbf{x}^*) \\
&= -\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 - \beta(\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k)^T(\mathbf{x}_k - \mathbf{x}^*).
\end{aligned} \tag{8.58}$$

Multiplying (8.57) by $\lambda_k(\lambda_k - 1)$ and (8.58) by λ_k respectively, and summing the two, we have

$$\begin{aligned}
&(\lambda_k)^2\delta_{k+1} - (\lambda_{k-1})^2\delta_k \\
&\leq -(\lambda_k)^2\tfrac{\beta}{2}|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 - \lambda_k\beta(\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k)^T\mathbf{d}_k \\
&= -\tfrac{\beta}{2}((\lambda_k)^2|\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k|^2 + 2\lambda_k(\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_k)^T\mathbf{d}_k) \\
&= -\tfrac{\beta}{2}(|\lambda_k\tilde{\mathbf{x}}_{k+1} - (\lambda_k - 1)\tilde{\mathbf{x}}_k - \mathbf{x}^*|^2 - |\mathbf{d}_k|^2) \\
&= \tfrac{\beta}{2}(|\mathbf{d}_k|^2 - |\lambda_k\tilde{\mathbf{x}}_{k+1} - (\lambda_k - 1)\tilde{\mathbf{x}}_k - \mathbf{x}^*|^2).
\end{aligned}$$

Using (8.55) and (8.56) we derive

$$\lambda_k\tilde{\mathbf{x}}_{k+1} - (\lambda_k - 1)\tilde{\mathbf{x}}_k = \lambda_{k+1}\mathbf{x}_{k+1} - (\lambda_{k+1} - 1)\tilde{\mathbf{x}}_{k+1}.$$

Thus,

$$(\lambda_k)^2\delta_{k+1} - (\lambda_{k-1})^2\delta_k \leq \frac{\beta}{2}(|\mathbf{d}_k|^2 - |\mathbf{d}_{k+1}|^2.) \tag{8.59}$$

Summing up (8.59) from 1 to k we have

$$\delta_{k+1} \leq \frac{\beta}{2(\lambda_k)^2}|\mathbf{d}_1|^2 \leq \frac{2\beta}{k^2}|\mathbf{d}_0|^2$$

where we used facts $\lambda_k \geq k/2$ and $|\mathbf{d}_1| \leq |\mathbf{d}_0|$. ∎

The Method of False Position

Yet there is another steepest descent method, commonly called the BB method, that works as follows:

$$\Delta_k^x = \mathbf{x}_k - \mathbf{x}_{k-1} \quad \text{and} \quad \Delta_k^g = \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1}), \tag{8.60}$$

$$\alpha_k = \frac{(\Delta_k^x)^T \Delta_k^g}{(\Delta_k^g)^T \Delta_k^g} \quad \text{or} \quad \alpha_k = \frac{(\Delta_k^x)^T \Delta_k^x}{(\Delta_k^x)^T \Delta_k^g},$$

Then

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k = \mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k)^T. \tag{8.61}$$

The step size of the BB method resembles the one used in quadratic curve fitting discussed for line search. There, the step size of (8.10) is given as $\frac{x_{k-1}-x_k}{f'(x_{k-1})-f'(x_k)}$. If we let $\delta_k^x = x_k - x_{k-1}$ and $\delta_k^g = f'(x_k) - f'(x_{k-1})$, this quantity can be written as $\frac{\delta_k^x \delta_k^g}{(\delta_k^g)^2}$ or $\frac{(\delta_k^x)^2}{\delta_k^x \delta_k^g}$. In the vector case, multiplication is replaced by inner product.

There was another explanation on the step size of the BB method. Consider convex quadratic minimization, and let the distinct positive eigenvalues of the Hessian $\mathbf{Q}$ be $\lambda_1, \lambda_2, \ldots \lambda_K$. Then, if we let the step size in the method of steepest descent be $\alpha_k = \frac{1}{\lambda_k}$, $k = 1, \ldots, K$, the method terminates in K iterations (which we leave as an exercise). In the BB method, α_k minimizes

$$|\Delta_k^x - \alpha \Delta_k^g| = |\Delta_k^x - \alpha \mathbf{Q} \Delta_k^x|.$$

If the error becomes 0 plus $|\Delta_k^x| \neq 0$, $\frac{1}{\alpha_k}$ will be a positive eigenvalue of $\mathbf{Q}$. Notice that the objective values of the iterates generated by the BB method is not monotonically decreasing; the method may overshoot in order to have a better position in the long run.

8.5 Newton's Method

The idea behind Newton's method is that the function f being minimized is approximated locally by a quadratic function, and this approximate function is minimized exactly. Thus near $\mathbf{x}_k$ we can approximate f by the truncated Taylor series

$$f(\mathbf{x}) \simeq f(\mathbf{x}_k) + \boldsymbol{\nabla} f(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k).$$

The right-hand side is minimized at

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\mathbf{F}(\mathbf{x}_k)]^{-1} \boldsymbol{\nabla} f(\mathbf{x}_k)^T, \tag{8.62}$$

and this equation is the pure form of Newton's method.

In view of the second-order sufficiency conditions for a minimum point, we assume that at a relative minimum point, $\mathbf{x}^*$, the Hessian matrix, $\mathbf{F}(\mathbf{x}^*)$, is positive definite. We can then argue that if f has continuous second partial derivatives, $\mathbf{F}(\mathbf{x})$ is positive definite near $\mathbf{x}^*$ and hence the method is well defined near the solution.

Order Two Convergence

Newton's method has very desirable properties if started sufficiently close to the solution point. Its order of convergence is two.

> ***Theorem (Newton's Method).*** *Let $f \in C^3$ on E^n, and assume that at the local minimum point $\mathbf{x}^*$, the Hessian $\mathbf{F}(\mathbf{x}^*)$ is positive definite. Then if started sufficiently close to $\mathbf{x}^*$, the points generated by Newton's method converge to $\mathbf{x}^*$. The order of convergence is at least two.*

Proof. There are $\rho > 0$, $\beta_1 > 0$, $\beta_2 > 0$ such that for all $\mathbf{x}$ with $|\mathbf{x} - \mathbf{x}^*| < \rho$, there holds $|\mathbf{F}(\mathbf{x})^{-1}| < \beta_1$ (see Appendix **A** for the definition of the norm of a matrix) and $|\nabla f(\mathbf{x}^*)^T - \nabla f(\mathbf{x})^T - \mathbf{F}(\mathbf{x})(\mathbf{x}^* - \mathbf{x})| \leqslant \beta_2 |\mathbf{x} - \mathbf{x}^*|^2$. Now suppose $\mathbf{x}_k$ is selected with $\beta_1\beta_2|\mathbf{x}_k - \mathbf{x}^*| < 1$ and $|\mathbf{x}_k - \mathbf{x}^*| < \rho$. Then

$$\begin{aligned}
|\mathbf{x}_{k+1} - \mathbf{x}^*| &= |\mathbf{x}_k - \mathbf{x}^* - \mathbf{F}(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)^T| \\
&= |\mathbf{F}(\mathbf{x}_k)^{-1}[\nabla f(\mathbf{x}^*)^T - \nabla f(\mathbf{x}_k)^T - \mathbf{F}(\mathbf{x}_k)(\mathbf{x}^* - \mathbf{x}_k)]| \\
&\leqslant |\mathbf{F}(\mathbf{x}_k)^{-1}|\beta_2|\mathbf{x}_k - \mathbf{x}^*|^2 \\
&\leqslant \beta_1\beta_2|\mathbf{x}_k - \mathbf{x}^*|^2 < |\mathbf{x}_k - \mathbf{x}^*|.
\end{aligned}$$

The final inequality shows that the new point is closer to $\mathbf{x}^*$ than the old point, and hence all conditions apply again to $\mathbf{x}_{k+1}$. The previous inequality establishes that convergence is second order. ∎

Modifications

Although Newton's method is very attractive in terms of its convergence properties near the solution, it requires modification before it can be used at points that are remote from the solution. The general nature of these modifications is discussed in the remainder of this section.

1. **Damping.** The first modification is that usually a search parameter α is introduced so that the method takes the form

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k[\mathbf{F}(\mathbf{x}_k)]^{-1}\nabla f(\mathbf{x}_k)^T,$$

where α_k is selected to minimize f. Near the solution we expect, on the basis of how Newton's method was derived, that $\alpha_k \simeq 1$. Introducing the parameter for general points, however, guards against the possibility that the objective might increase with $\alpha_k = 1$, due to nonquadratic terms in the objective function.

2. **Positive Definiteness.** A basic consideration for Newton's method can be seen most clearly by a brief examination of the general class of algorithms

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{M}_k \mathbf{g}_k, \tag{8.63}$$

where $\mathbf{M}_k$ is an $n \times n$ matrix, α is a positive search parameter, and $\mathbf{g}_k = \nabla f(\mathbf{x}_k)^T$. We note that both steepest descent ($\mathbf{M}_k = \mathbf{I}$) and Newton's method ($\mathbf{M}_k = [\mathbf{F}(\mathbf{x}_k)]^{-1}$) belong to this class. The direction vector $\mathbf{d}_k = -\mathbf{M}_k \mathbf{g}_k$ obtained in this way is a direction of descent if for small α the value of f decreases as α increases from zero. For small α we can say

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) + O(|\mathbf{x}_{k+1} - \mathbf{x}_k|^2).$$

Employing (8.51) this can be written as

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k) - \alpha \mathbf{g}_k^T \mathbf{M}_k \mathbf{g}_k + O(\alpha^2).$$

As $\alpha \to 0$, the second term on the right dominates the third. Hence if one is to guarantee a decrease in f for small α, we must have $\mathbf{g}_k^T \mathbf{M}_k \mathbf{g}_k > 0$. The simplest way to insure this is to require that $\mathbf{M}_k$ be positive definite.

The best circumstance is that where $\mathbf{F}(\mathbf{x})$ is itself positive definite throughout the search region. The objective function of many important optimization problems have this property, including for example interior-point approaches to linear programming using the logarithm as a barrier function. Indeed, it can be argued that convexity is an inherent property of the majority of well-formulated optimization problems.

Therefore, assume that the Hessian matrix $\mathbf{F}(\mathbf{x})$ is positive definite throughout the search region and that f has continuous third derivatives. At a given $\mathbf{x}_k$ define the symmetric matrix $\mathbf{T} = \mathbf{F}(\mathbf{x}_k)^{-1/2}$. As in Sect. 8.3 introduce the change of variable $\mathbf{T}\mathbf{y} = \mathbf{x}$. Then according to (8.48) a steepest descent direction with respect to $\mathbf{y}$ is equivalent to a direction with respect to $\mathbf{x}$ of $\mathbf{d} = -\mathbf{T}\mathbf{T}^T \mathbf{g}(\mathbf{x}_k)$, where $\mathbf{g}(\mathbf{x}_k)$ is the gradient of f with respect to $\mathbf{x}$ at $\mathbf{x}_k$. Thus, $\mathbf{d} = \mathbf{F}^{-1}\mathbf{g}(\mathbf{x}_k)$. In other words, a steepest descent direction in $\mathbf{y}$ is equivalent to a Newton direction in $\mathbf{x}$.

We can turn this relation around to analyze Newton steps in $\mathbf{x}$ as equivalent to gradient steps in $\mathbf{y}$. We know that convergence properties in $\mathbf{y}$ depend on the bounds on the Hessian matrix given by (8.49) as

$$\mathbf{H}(\mathbf{y}) = \mathbf{T}^T \mathbf{F}(\mathbf{x}) \mathbf{T} = \mathbf{F}^{-1/2} \mathbf{F}(\mathbf{x}) \mathbf{F}^{-1/2}. \tag{8.64}$$

Recall that $\mathbf{F} = \mathbf{F}(\mathbf{x}_k)$ which is fixed, whereas $\mathbf{F}(\mathbf{x})$ denotes the general Hessian matrix with respect to $\mathbf{x}$ near $\mathbf{x}_k$. The product (8.64) is the identity matrix at $\mathbf{y}_k$

but the rate of convergence of steepest descent in $\mathbf{y}$ depends on the bounds of the smallest and largest eigenvalues of $\mathbf{H}(\mathbf{y})$ in a region near $\mathbf{y}_k$.

These observations tell us that the damped method of Newton's method will converge at a linear rate at least as fast as $c = (1 - a/A)$ where a and A are lower and upper bounds on the eigenvalues of $\mathbf{F}(\mathbf{x}_0)^{-1/2}\mathbf{F}(\mathbf{x}^0)\mathbf{F}(\mathbf{x}_0)^{-1/2}$, where $\mathbf{x}_0$ and $\mathbf{x}^0$ are arbitrary points in the local search region. These bounds depend, in turn, on the bounds of the third-order derivatives of f. It is clear, however, by continuity of $\mathbf{F}(\mathbf{x})$ and its derivatives, that the rate becomes very fast near the solution, becoming superlinear, and in fact, as we know, quadratic.

3. **Backtracking.** The backtracking method of line search, using $\alpha = 1$ as the initial guess, is an attractive procedure for use with Newton's method. Using this method the overall progress of Newton's method divides naturally into two phases: first a damping phase where backtracking may require $\alpha < 1$, and second a quadratic phase where $\alpha = 1$ satisfies the backtracking criterion at every step. The damping phase was discussed above.

Let us now examine the situation when close to the solution. We assume that all derivatives of f through the third are continuous and uniformly bounded. We also assume that in the region close to the solution, $\mathbf{F}(\mathbf{x})$ is positive definite with $\overline{a} > 0$ and $\overline{A} > 0$ being, respectively, uniform lower and upper bounds on the eigenvalues of $\mathbf{F}(\mathbf{x})$. Using $\alpha = 1$ and $\varepsilon < 0.5$ we have for $\mathbf{d}_k = -\mathbf{F}(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k)$

$$\begin{aligned} f(\mathbf{x}_k + \mathbf{d}_k) &= f(\mathbf{x}_k) - \mathbf{g}(\mathbf{x}_k)^T\mathbf{F}(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) + \frac{1}{2}\mathbf{g}(\mathbf{x}_k)^T\mathbf{F}(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) + o(|\mathbf{g}(\mathbf{x}_k)|^2) \\ &= f(\mathbf{x}_k) - \frac{1}{2}\mathbf{g}(\mathbf{x}_k)^T\mathbf{F}(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) + o(|\mathbf{g}(\mathbf{x}_k)|^2) \\ &< f(\mathbf{x}_k) - \varepsilon\mathbf{g}(\mathbf{x}_k)^T\mathbf{F}(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) + o(|\mathbf{g}(\mathbf{x}_k)|^2), \end{aligned}$$

where the o bound is uniform for all $\mathbf{x}_k$. Since $|\mathbf{g}(\mathbf{x}_k)| \to 0$ (uniformly) as $\mathbf{x}_k \to \mathbf{x}^*$, it follows that once $\mathbf{x}_k$ is sufficiently close to $\mathbf{x}^*$, then $f(\mathbf{x}_k+\mathbf{d}_k) < f(\mathbf{x}_k)-\varepsilon\mathbf{g}(\mathbf{x}_k)^T\mathbf{d}_\mathbf{k}$ and hence the backtracking test (the first part of Amijo's rule) is satisfied. This means that $\alpha = 1$ will be used throughout the final phase.

4. **General Problems.** In practice, Newton's method must be modified to accommodate the possible nonpositive definiteness at regions remote from the solution.

A common approach is to take $\mathbf{M}_k = [\varepsilon_k\mathbf{I}+\mathbf{F}(\mathbf{x}_k)]^{-1}$ for some non-negative value of ε_k. This can be regarded as a kind of compromise between steepest descent (ε_k very large) and Newton's method ($\varepsilon_k = 0$). There is always an ε_k that makes $\mathbf{M}_k$ positive definite. We shall present one modification of this type.

Let $\mathbf{F}_k \equiv \mathbf{F}(\mathbf{x}_k)$. Fix a constant $\delta > 0$. Given $\mathbf{x}_k$, calculate the eigenvalues of $\mathbf{F}_k$ and let ε_k be the smallest nonnegative constant for which the matrix $\varepsilon_k\mathbf{I} + \mathbf{F}_k$ has eigenvalues greater than or equal to δ. Then define

$$\mathbf{d}_k = -(\varepsilon_k\mathbf{I} + \mathbf{F}_k)^{-1}\mathbf{g}_k \tag{8.65}$$

and iterate according to

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k, \tag{8.66}$$

where α_k minimizes $f(\mathbf{x}_k + \alpha \mathbf{d}_k)$, $\alpha \geqslant 0$.

This algorithm has the desired global and local properties. First, since the eigenvalues of a matrix depend continuously on its elements, ε_k is a continuous function of $\mathbf{x}_k$ and hence the mapping $\mathbf{D} : E^n \rightarrow E^{2n}$ defined by $\mathbf{D}(\mathbf{x}_k) = (\mathbf{x}_k, \mathbf{d}_k)$ is continuous. Thus the algorithm $\mathbf{A} = \mathbf{SD}$ is closed at points outside the solution set $\Omega = \{\mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}\}$. Second, since $\varepsilon_k \mathbf{I} + \mathbf{F}_k$ is positive definite, $\mathbf{d}_k$ is a descent direction and thus $Z(\mathbf{x}) \equiv f(\mathbf{x})$ is a continuous descent function for A. Therefore, assuming the generated sequence is bounded, the Global Convergence Theorem applies. Furthermore, if $\delta > 0$ is smaller than the smallest eigenvalue of $\mathbf{F}(\mathbf{x}^*)$, then for $\mathbf{x}_k$ sufficiently close to $\mathbf{x}^*$ we will have $\varepsilon_k = 0$, and the method reduces to Newton's method. Thus this revised method also has order of convergence equal to two.

The selection of an appropriate δ is somewhat of an art. A small δ means that nearly singular matrices must be inverted, while a large δ means that the order two convergence may be lost. Experimentation and familiarity with a given class of problems are often required to find the best δ.

The utility of the above algorithm is hampered by the necessity to calculate the eigenvalues of $\mathbf{F}(\mathbf{x}_k)$, and in practice an alternate procedure is used. In one class of methods (Levenberg–Marquardt type methods), for a given value of ε_k, Cholesky factorization of the form $\varepsilon_\mathbf{k} \mathbf{I} + \mathbf{F}(\mathbf{x}_\mathbf{k}) = \mathbf{G}\mathbf{G}^T$ (see Exercise 6 of Chap. 7) is employed to check for positive definiteness. If the factorization breaks down, ε_k is increased. The factorization then also provides the direction vector through solution of the equations $\mathbf{G}\mathbf{G}^T \mathbf{d}_k = \mathbf{g}_k$, which are easily solved, since $\mathbf{G}$ is triangular. Then the value $f(\mathbf{x}_k + \mathbf{d}_k)$ is examined. If it is sufficiently below $f(\mathbf{x}_k)$, then $\mathbf{x}_{k+1}$ is accepted and a new ε_{k+1} is determined. Essentially, ε serves as a search parameter in these methods. It should be clear from this discussion that the simplicity that Newton's method first seemed to promise is not fully realized in practice.

Newton's Method and Logarithms

Interior point methods of linear and nonlinear programming use barrier functions, which usually are based on the logarithm. For linear programming especially, this means that the only nonlinear terms are logarithms. Newton's method enjoys some special properties in this case,

To illustrate, let us apply Newton's method to the one-dimensional problem

$$\min_x [tx - \ln x] \tag{8.67}$$

where t is a positive parameter. The derivative at x is

$$f'(x) = t - \frac{1}{x},$$

and of course the solution is $x^* = 1/t$, or equivalently $1 - tx^* = 0$. The second derivative is $f''(x) = 1/x^2$. Denoting by x^+ the result of one step of a pure Newton's method (with step length equal to 1) applied to the point x, we find

$$x^+ = x - [f''(x)]^{-1} f'(x) = x - x^2\left(t - \frac{1}{x}\right) = x - tx^2 + x$$
$$= 2x - tx^2.$$

Thus

$$1 - tx^+ = 1 - 2tx + x^2t^2 = (1 - tx)^2 \tag{8.68}$$

Therefore, rather surprisingly, the quadratic nature of convergence of $(1 - tx) \to 0$ is directly evident and exact. Expression (8.68) represents a reduction in the error magnitude only if $|(1 - tx)| < 1$, or equivalently, $0 < x < 2/t$. If x is too large, then Newton's method must be used with damping until the region $0 < x < 2/t$ is reached. From then on, a step size of 1 will exhibit pure quadratic error reduction.

The situation is shown in Fig. 8.11. The graph is that of $f'(x) = t - 1/x$. The root-finding form of Newton's method (Sect. 8.1) is then applied to this function. At each point, the tangent line is followed to the x axis to find the new point. The starting value marked x_1 is far from the solution $1/t$ and hence following the tangent would lead to a new point that was negative. Damping must be applied at that starting point. Once a point x is reached with $0 < x < 1/t$, all further points will remain to the left of $1/t$ and move toward it quadratically.

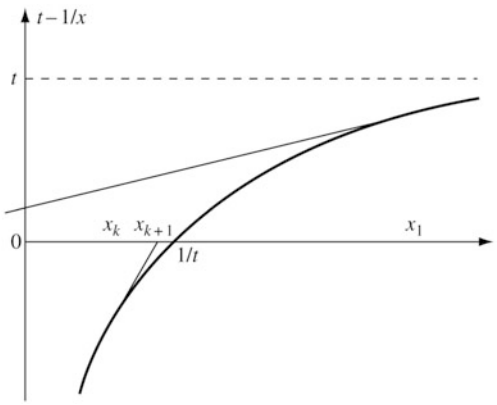

Fig. 8.11 Newton's method applied to minimization of $tx - \ln x$

In interior point methods for linear programming, a logarithmic barrier function is applied separately to the variables that must remain positive. The convergence analysis in these situations is an extension of that for the simple case given here, allowing for estimates of the rate of convergence that do not require knowledge of bounds of third-order derivatives.

Self-concordant Functions

The special properties exhibited above for the logarithm have been extended to the general class of *self-concordant functions* of which the logarithm is the primary example. A function f defined on the real line is self-concordant if it satisfies

$$|f'''(x)| \leq 2f''(x)^{3/2}, \tag{8.69}$$

throughout its domain. It is easily verified that $f(x) = -\ln x$ satisfies this inequality with equality for $x > 0$.

Self-concordancy is preserved by the addition of an affine term since such a term does not affect the second or third derivatives.

A function defined on E^n is said to be *self-concordant* if it is self-concordant in every direction: that is if $f(\mathbf{x} + \alpha\mathbf{d})$ is self-concordant with respect to α for every $\mathbf{d}$ throughout the domain of f.

Self-concordant functions can be combined by addition and even by composition with affine functions to yield other self-concordant functions. (See Exercise 29.) For example the function

$$f(\mathbf{x}) = -\sum_{i=1}^{m} \ln(b_j - \mathbf{a}_i^T \mathbf{x}),$$

often used in interior point methods for linear programming, is self-concordant.

When a self-concordant function is subjected to Newton's method, the quadratic convergence of final phase can be measured in terms of the function

$$\lambda(\mathbf{x}) = [\nabla f(\mathbf{x})\mathbf{F}(\mathbf{x})^{-1}\nabla f(\mathbf{x})^T]^{1/2},$$

where as usual $\mathbf{F}(\mathbf{x})$ is the Hessian matrix of f at $\mathbf{x}$. Then it can be shown that close to the solution

$$2\lambda(\mathbf{x}_{k+1}) \leq [2\lambda(\mathbf{x}_k)]^2. \tag{8.70}$$

Furthermore, in a backtracking procedure, estimates of both the stepwise progress in the damping phase and the point at which the quadratic phase begins can be expressed in terms of parameters that depend only on the backtracking parameters. Although, this knowledge does not generally influence practice, it is theoretically quite interesting.

Example 1 (The Logarithmic Case). Consider the earlier example of $f(x) = tx - \ln x$. There

$$\lambda(x) = [f'(x)^2/f''(x)]^{\frac{1}{2}} = |(t - 1/x)x| = |1 - tx|.$$

Then (8.70) gives

$$(1 - tx^+) \leq 2(1 - tx)^2.$$

Actually, for this example, as we found in (8.68), the factor of 2 is not required.

There is a relation between the analysis of self-concordant functions and our earlier convergence analysis.

Recall that one way to analyze Newton's method is to change variables from $\mathbf{x}$ to $\mathbf{y}$ according to $\tilde{\mathbf{y}} = [\mathbf{F}(\mathbf{x})]^{-(1/2)}\tilde{\mathbf{x}}$, where here $\mathbf{x}$ is a reference point and $\tilde{\mathbf{x}}$ is variable. The gradient with respect to $\mathbf{y}$ at $\tilde{\mathbf{y}}$ is then $\mathbf{F}(\mathbf{x})^{-(1/2)}\nabla f(\tilde{\mathbf{x}})$, and hence the norm of the gradient at $\mathbf{y}$ is $[\nabla f(\mathbf{x})\mathbf{F}(\mathbf{x})^{-1}\nabla f(\mathbf{x})^T]^{(1/2)} \equiv \lambda(\mathbf{x})$. Hence it is perhaps not surprising that $\lambda(\mathbf{x})$ plays a role analogous to the role played by the norm of the gradient in the analysis of steepest descent.

8.6 Coordinate Descent Methods

The algorithms discussed in this section are sometimes attractive because of their easy implementation. Generally, however, their convergence properties are poorer than steepest descent.

Let f be a function on E^n having continuous first partial derivatives. Given a point $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, descent with respect to the coordinate x_i (i fixed) means that one solves

$$\underset{x_i}{\text{minimize}}\ f(x_1, x_2, \ldots, x_n).$$

Thus only changes in the single component x_i are allowed in seeking a new and better vector $\mathbf{x}$ (one can also consider $\mathbf{x}_i$ the ith block of variables, called the block-coordinate method). In our general terminology, each such descent can be regarded as a descent in the direction $\mathbf{e}_i$(or $-\mathbf{e}_i$) where $\mathbf{e}_i$ is the ith unit vector. By sequentially minimizing with respect to different components, a relative minimum of f might ultimately be determined.

There are a number of ways that this concept can be developed into a full algorithm. The *cyclic coordinate descent* algorithm minimizes f cyclically with respect to the coordinate variables. Thus x_1 is changed first, then x_2 and so forth through x_n. The process is then repeated starting with x_1 again. A variation of this is the *Aitken double sweep method*. In this procedure one searches over $x_1, x_2, \ldots, x_n$, in that order, and then comes back in the order $x_{n-1}, x_{n-2}, \ldots, x_1$. These cyclic methods have the advantage of not requiring any information about ∇f to determine the descent directions.

If the gradient of f is available, then it is possible to select the order of descent coordinates on the basis of the gradient. A popular technique is the *Gauss–Southwell Method* where at each stage the coordinate corresponding to the largest (in absolute value) component of the gradient vector is selected for descent. A *randomized strategy* can be also adapted in which one randomly chooses a coordinate to optimize in each step; see more discussions later.

Global Convergence

It is simple to prove global convergence for cyclic coordinate descent. The algorithmic map $\mathbf{A}$ is the composition of $2n$ maps

$$\mathbf{A} = \mathbf{S}\mathbf{C}^n\mathbf{S}\mathbf{C}^{n-1}\dots\mathbf{S}\mathbf{C}^1,$$

where $\mathbf{C}^i(\mathbf{x}) = (\mathbf{x},\ \mathbf{e}_i)$ with $\mathbf{e}_i$ equal to the ith unit vector, and $\mathbf{S}$ is the usual line search algorithm but over the doubly infinite line rather than the semi-infinite line. The map $\mathbf{C}^i$ is obviously continuous and $\mathbf{S}$ is closed. If we assume that points are restricted to a compact set, then $\mathbf{A}$ is closed by Corollary 1, Sect. 7.7. We define the solution set $\Gamma = \{\mathbf{x} : \nabla f(\mathbf{x}) = \mathbf{0}\}$. If we impose the mild assumption on f that a search along any coordinate direction yields a unique minimum point, then the function $Z(\mathbf{x}) \equiv f(\mathbf{x})$ serves as a continuous descent function for $\mathbf{A}$ with respect to Γ. This is because a search along any coordinate direction either must yield a decrease or, by the uniqueness assumption, it cannot change position. Therefore, if at a point $\mathbf{x}$ we have $\nabla f(\mathbf{x}) \neq \mathbf{0}$, then at least one component of $\nabla f(\mathbf{x})$ does not vanish and a search along the corresponding coordinate direction must yield a decrease.

Local Convergence Rate

It is difficult to compare the rates of convergence of these algorithms with the rates of others that we analyze. This is partly because coordinate descent algorithms are from an entirely different general class of algorithms than, for example, steepest descent and Newton's method, since coordinate descent algorithms are unaffected by (diagonal) scale factor changes but are affected by rotation of coordinates—the opposite being true for steepest descent. Nevertheless, some comparison is possible.

It can be shown (see Exercise 20) that for the same quadratic problem as treated in Sect. 8.2, there holds for the Gauss−Southwell method

$$E(\mathbf{x}_{k+1}) \leqslant \left(1 - \frac{a}{A(n-1)}\right) E(\mathbf{x}_k), \tag{8.71}$$

where $a,\ A$ are as in Sect. 8.2 and n is the dimension of the problem. Since

$$\left(\frac{A-a}{A+a}\right)^2 \leqslant \left(1 - \frac{a}{A}\right) \leqslant \left(1 - \frac{a}{A(n-1)}\right)^{n-1}, \tag{8.72}$$

we see that the bound we have for steepest descent is better than the bound we have for $n-1$ applications of the Gauss−Southwell scheme. Hence we might argue that it takes essentially $n-1$ coordinate searches to be as effective as a single gradient search. This is admittedly a crude guess, since (8.54) is generally not a tight bound, but the overall conclusion is consistent with the results of many experiments. Indeed, unless the variables of a problem are essentially uncoupled from each other

(corresponding to a nearly diagonal Hessian matrix) coordinate descent methods seem to require about n line searches to equal the effect of one step of steepest descent.

The above discussion again illustrates the general objective that we seek in convergence analysis. By comparing the formula giving the rate of convergence for steepest descent with a bound for coordinate descent, we are able to draw some general conclusions on the relative performance of the two methods that are not dependent on specific values of a and A. Our analyses of local convergence properties, which usually involve specific formulae, are always guided by this objective of obtaining general qualitative comparisons.

Example. The quadratic problem considered in Sect. 8.2 with

$$\mathbf{Q} = \begin{bmatrix} 0.78 & -0.02 & -0.12 & -0.14 \\ -0.02 & 0.86 & -0.04 & 0.06 \\ -0.12 & -0.04 & 0.72 & -0.08 \\ -0.14 & 0.06 & -0.08 & 0.74 \end{bmatrix}$$

$$\mathbf{b} = (0.76,\ 0.08,\ 1.12,\ 0.68)$$

was solved by the various coordinate search methods. The corresponding values of the objective function are shown in Table 8.3. Observe that the convergence rates of the three coordinate search methods are approximately equal but that they all converge about three times slower than steepest descent. This is in accord with the estimate given above for the Gauss−Southwell method, since in this case $n - 1 = 3$.

Convergence Speed of a Randomized Coordinate Descent Method

We now describe a *randomized strategy* in selecting x_i in each step of the coordinate descent method for f that is differentiable and Lipschitz continuous; that is, there exist some constants $\beta_i > 0$, $i = 1, \ldots, n$, such that

$$|\nabla_i f(\mathbf{x} + h\mathbf{e}_i) - \nabla_i f(\mathbf{x})| \le \beta_i |h|, \quad \forall\, h \in E,\ \mathbf{x} \in E^n, \tag{8.73}$$

where $\nabla_i f(\mathbf{x})$ denotes the ith partial derivative of f at $\mathbf{x}$, and $\mathbf{e}_i$ is the ith unit vector with the ith entry equal 1 and everywhere else equal 0.

Randomized coordinate decent method. Given an initial point $\mathbf{x}_0$; repeat for $k = 0, 1, 2, \ldots$

1. Choose $i_k \in \{1, \ldots, n\}$ randomly with a uniform distribution.
2. Update $\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{\beta_{i_k}} \nabla_{i_k} f(\mathbf{x}_k) \mathbf{e}_{i_k}$.

Note that after k iterations, the randomized coordinate descent method generates a random sequence of $\mathbf{x}_k$, which depends on the observed realization of the random variable

$$\xi_{k-1} = \{i_0, i_1, \ldots, i_{k-1}\}.$$

Table 8.3 Solutions to Example

Iteration no.	Value of f for various methods		
	Gauss-Southwell	Cyclic	Double sweep
0	0.0	0.0	0.0
1	−0.871111	−0.370256	−0.370256
2	−1.445584	−0.376011	−0.376011
3	−2.087054	−1.446460	−1.446460
4	−2.130796	−2.052949	−2.052949
5	−2.163586	−2.149690	−2.060234
6	−2.170272	−2.149693	−2.060237
7	−2.172786	−2.167983	−2.165641
8	−2.174279	−2.173169	−2.165704
9	−2.174583	−2.174392	−2.168440
10	−2.174638	−2.174397	−2.173981
11	−2.174651	−2.174582	−2.174048
12	−2.174655	−2.174643	−2.174054
13	−2.174658	−2.174656	−2.174608
14	−2.174659	−2.174656	−2.174608
15	−2.174659	−2.174658	−2.174622
16		−2.174659	−2.174655
17		−2.174659	−2.174656
18			−2.174656
19			−2.174659
20			−2.174659

Theorem 3 (Randomized Coordinate Descent—Lipschitz Convex Case). *Let $f(\boldsymbol{x})$ be convex and differentiable everywhere, satisfy the Lipschitz condition (8.73), and admit a minimizer $\mathbf{x}^*$. Then, the randomized coordinate decent method generates a sequence of solutions $\mathbf{x}_k$ such that for any $k \geq 1$, the iterate $\mathbf{x}_k$ satisfies*

$$\mathrm{E}_{\xi_{k-1}}[f(\mathbf{x}_k)] - f(\mathbf{x}^*) \leq \frac{n}{n+k}\left(\frac{1}{2}|\mathbf{x}_0 - \mathbf{x}^*|_\beta^2 + f(\mathbf{x}_0) - f(\mathbf{x}^*)\right),$$

where $|\mathbf{x}|_\beta = \left(\sum_i \beta_i x_i^2\right)^{1/2}$ *for all* $\mathbf{x} \in E^n$.

Proof. Let $r_k^2 = |\mathbf{x}_k - \mathbf{x}^*|_\beta^2 = \sum_{i=1}^n \beta_i((\mathbf{x}_k)_i - x_i^*)^2$ for any $k \geq 0$. Since $\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{\beta_{i_k}}\nabla_{i_k} f(\mathbf{x}_k)\mathbf{e}_{i_k}$, we have

$$r_{k+1}^2 = r_k^2 - 2\nabla_{i_k} f(\mathbf{x}_k)((\mathbf{x}_k)_{i_k} - x_{i_k}^*) + \frac{1}{\beta_{i_k}}(\nabla_{i_k} f(\mathbf{x}_k))^2.$$

It follows from (8.73), Lemma 1, and $\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{\beta_{i_k}}\nabla_{i_k} f(\mathbf{x}_k)\mathbf{e}_{i_k}$ that

$$\begin{aligned} f(\mathbf{x}_{k+1}) &\leq f(\mathbf{x}_k) + \nabla_{i_k} f(\mathbf{x}_k)((\mathbf{x}_{k+1})_{i_k} - (\mathbf{x}_k)_{i_k}) + \frac{\beta_{i_k}}{2}((\mathbf{x}_{k+1})_{i_k} - (\mathbf{x}_k)_{i_k})^2 \\ &= f(\mathbf{x}_k) - \frac{1}{2\beta_{i_k}}(\nabla_{i_k} f(\mathbf{x}_k))^2. \end{aligned} \tag{8.74}$$

Combining the above two relations, one has

$$r_{k+1}^2 \;\le\; r_k^2 - 2\nabla_{i_k} f(\mathbf{x}_k)((\mathbf{x}_k)_{i_k} - \mathbf{x}_{i_k}^*) + 2(f(\mathbf{x}_k) - f(\mathbf{x}_{k+1})).$$

Multiplying both sides by $1/2$ and taking expectation with respect to i_k yields

$$\mathrm{E}_{i_k}\left[\frac{1}{2}r_{k+1}^2\right] \le \frac{1}{2}r_k^2 - \frac{1}{n}\nabla f(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{x}^*) + f(\mathbf{x}_k) - \mathrm{E}_{i_k}\left[f(\mathbf{x}_{k+1})\right],$$

which together with the fact that $\nabla f(\mathbf{x}_k)(\mathbf{x}^* - \mathbf{x}_k) \le f(\mathbf{x}^*) - f(\mathbf{x}_k)$ yields

$$\mathrm{E}_{i_k}\left[\frac{1}{2}r_{k+1}^2\right] \;\le\; \frac{1}{2}r_k^2 + \frac{1}{n}f(\mathbf{x}^*) + \frac{n-1}{n}f(\mathbf{x}_k) - \mathrm{E}_{i_k}\left[f(\mathbf{x}_{k+1})\right].$$

By rearranging terms, we obtain that for each $k \ge 0$,

$$\mathrm{E}_{i_k}\left[\frac{1}{2}r_{k+1}^2 + f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)\right] \;\le\; \left(\frac{1}{2}r_k^2 + f(\mathbf{x}_k) - f(\mathbf{x}^*)\right) - \frac{1}{n}\left(f(\mathbf{x}_k) - f(\mathbf{x}^*)\right).$$

Let $f^* = f(\mathbf{x}^*)$. Then, taking expectation with respect to ξ_{k-1} on both sides of the above relation, we have

$$\mathrm{E}_{\xi_k}\left[\frac{1}{2}r_{k+1}^2 + f(\mathbf{x}_{k+1}) - f^*\right] \;\le\; \mathrm{E}_{\xi_{k-1}}\left[\frac{1}{2}r_k^2 + f(\mathbf{x}_k) - f^*\right] - \frac{\mathrm{E}_{\xi_{k-1}}\left[f(\mathbf{x}_k) - f^*\right]}{n}. \quad (8.75)$$

In addition, it follows from (8.74) that $\mathrm{E}_{\xi_j}[f(\mathbf{x}_{j+1})] \le \mathrm{E}_{\xi_{j-1}}[f(\mathbf{x}_j)]$ for all $j \ge 0$. Using this relation and applying the inequality (8.75) recursively, we further obtain that

$$\begin{aligned}
\mathrm{E}_{\xi_k}\left[f(\mathbf{x}_{k+1})\right] - f^* &\le \mathrm{E}_{\xi_k}\left[\frac{1}{2}r_{k+1}^2 + f(\mathbf{x}_{k+1}) - f^*\right] \\
&\le \frac{1}{2}r_0^2 + f(\mathbf{x}_0) - f^* - \frac{1}{n}\sum_{j=0}^{k}\left(\mathrm{E}_{\xi_{j-1}}\left[f(\mathbf{x}_j)\right] - f^*\right) \\
&\le \frac{1}{2}r_0^2 + f(\mathbf{x}_0) - f^* - \frac{k+1}{n}\left(\mathrm{E}_{\xi_k}\left[f(\mathbf{x}_{k+1})\right] - f^*\right).
\end{aligned}$$

This leads to the desired result by moving the last term on the right to the left side. ∎

If f is a strongly convex quadratic function, the randomized coordinate decent method would have an expected average convergence rate $(1 - \frac{a}{An})$. However, each step of the method does $\frac{1}{n}$ amount of work of the full steepest descent update; see an exercise.

8.7 Summary

Most iterative algorithms for minimization require a line search at every stage of the process. By employing any one of a variety of curve fitting techniques, however, the order of convergence of the line search process can be made greater than unity, which means that as compared to the linear convergence that accompanies most full descent algorithms (such as steepest descent) the individual line searches are rapid. Indeed, in common practice, only about three search points are required in any one line search. If the first derivatives are available, then two search points are required (method of false position); and if both first and second derivatives are available, then one search point is required (Newton's method).

It was also shown in Sect. 8.1 and the exercises that line search algorithms of varying degrees of accuracy are all closed. Thus line searching is not only rapid enough to be practical but also behaves in such a way as to make analysis of global convergence simple.

The most important results of this chapter are the arithmetic convergence of the method of steepest descent for solving convex minimization, the improved arithmetic convergence of the accelerated steepest descent method, and the geometric convergence of the method for solving strongly convex minimization. The fact that the method of steepest descent converges linearly with a convergence ratio equal to $[(A - a)/(A + a)]^2$, where a and A are, respectively, the smallest and largest eigenvalues of the Hessian of the objective function evaluated at the solution point. This formula, which arises frequently throughout the remainder of the book, serves as a fundamental reference point for other algorithms. It is, however, important to understand that it is the *formula* and not its *value* that serves as the reference. We rarely advocate that the formula be evaluated since it involves quantities (namely eigenvalues) that are generally not computable until after the optimal solution is known. The formula itself, however, even though its value is unknown, can be used to make significant comparisons of the effectiveness of steepest descent versus other algorithms.

Newton's method has order two convergence. However, it must be modified to insure global convergence, and evaluation of the Hessian at every point can be costly. Nevertheless, Newton's method provides another valuable reference point in the study of algorithms, and is frequently employed in interior point methods using a logarithmic barrier function.

As optimization problem sizes become bigger and bigger, various coordinate descent algorithms are extremely popular. They are valuable especially in situations where the variables are essentially uncoupled or there is special structure that makes searching in the coordinate directions particularly easy. Typically, steepest descent can be expected to be faster. Even if the gradient is not directly available, it would probably be better to evaluate a finite-difference approximation to the gradient, by taking a single step in each coordinate direction, and use this approximation in a steepest descent algorithm, rather than executing a full line search in each coordinate direction.

8.8 Exercises

1. Show that $g[a,\ b,\ c]$ defined by (8.14) is symmetric, that is, interchange of the arguments does not affect its value.
2. Prove (8.14) and (8.15).
   Hint: To prove (8.15) expand it, and subtract and add $g'(x_k)$ to the numerator.
3. Argue using symmetry that the error in the cubic fit method approximately satisfies an equation of the form

$$\varepsilon_{k+1} = M(\varepsilon_k^2 \varepsilon_{k-1} + \varepsilon_k \varepsilon_{k-1}^2)$$

   and then find the order of convergence.
4. What conditions on the values and derivatives at two points guarantee that a cubic polynomial fit to this data will have a minimum between the two points? Use your answer to develop a search scheme, based on cubic fit, that is globally convergent for unimodal functions.
5. Using a symmetry argument, find the order of convergence for a line search method that fits a cubic to $x_{k-3},\ x_{k-2},\ x_{k-1},\ x_k$ in order to find x_{k+1}.
6. Consider the iterative process

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{a}{x_k}\right),$$

   where $a > 0$. Assuming the process converges, to what does it converge? What is the order of convergence?
7. Suppose the continuous real-valued function f of a single variable satisfies

$$\min_{x \geqslant 0} f(x) \ < f(0).$$

   Starting at any $x > 0$ show that, through a series of halvings and doublings of x and evaluation of the corresponding $f(x)$'s, a three-point pattern can be determined.
8. For $\delta > 0$ define the map $\mathbf{S}^\delta$ by

$$\mathbf{S}^\delta(\mathbf{x},\ \mathbf{d}) = \{\mathbf{y} : \mathbf{y} = \mathbf{x} + \alpha\mathbf{d},\ 0 \leqslant \alpha \leqslant \delta;\ f(\mathbf{y}) = \min_{0 \leqslant \beta \leqslant \delta} f(\mathbf{x} + \beta\mathbf{d})\}.$$

   Thus $\mathbf{S}^\delta$ searches the interval $[0,\ \delta]$ for a minimum of $f(\mathbf{x} + \alpha\mathbf{d})$, representing a "limited range" line search. Show that if f is continuous, $\mathbf{S}^\delta$ is closed at all $(\mathbf{x},\ \mathbf{d})$.
9. For $\varepsilon > 0$ define the map ${}^\varepsilon\mathbf{S}$ by

$${}^\varepsilon\mathbf{S}(\mathbf{x},\ \mathbf{d}) = \{\mathbf{y} : \mathbf{y} = \mathbf{x} + \alpha\mathbf{d},\ \alpha \geqslant 0,\ f(\mathbf{y}) \leqslant \min_{0 \leqslant \beta} f(\mathbf{x} + \beta\mathbf{d}) + \varepsilon\}.$$

   Show that if f is continuous, ${}^\varepsilon\mathbf{S}$ is closed at $(\mathbf{x},\ \mathbf{d})$ if $\mathbf{d} \neq \mathbf{0}$. This map corresponds to an "inaccurate" line search.

10. Referring to the previous two exercises, define and prove a result for ${}^{\varepsilon}\mathbf{S}^{\delta}$.
11. Define $\bar{\mathbf{S}}$ as the line search algorithm that finds the first relative minimum of $f(\mathbf{x} + \alpha\mathbf{d})$ for $\alpha \geqslant 0$. If f is continuous and $\mathbf{d} \neq \mathbf{0}$, is $\bar{\mathbf{S}}$ closed?
12. Consider the problem

$$\text{minimize } 5x^2 + 5y^2 - xy - 11x + 11y + 11.$$

    (a) Find a point satisfying the first-order necessary conditions for a solution.
    (b) Show that this point is a global minimum.
    (c) What would be the rate of convergence of steepest descent for this problem?
    (d) Starting at $x = y = 0$, how many steepest descent iterations would it take (at most) to reduce the function value to 10^{-11}?

13. Define the search mapping $\mathbf{F}$ that determines the parameter α to within a given fraction c, $0 \leqslant c \leqslant 1$, by

$$\mathbf{F}(\mathbf{x},\ \mathbf{d}) = \{\mathbf{y} : \mathbf{y} = \mathbf{x} + \alpha\mathbf{d},\ 0 \leqslant \alpha < \infty,\ |\overline{\alpha}| \leqslant c\overline{\alpha},\ \text{where } \frac{d}{d\alpha} f(\mathbf{x} + \overline{\alpha}\mathbf{d}) = 0\}.$$

    Show that if $\mathbf{d} \neq \mathbf{0}$ and $(d/d\alpha)f(\mathbf{x}+\alpha\mathbf{d})$ is continuous, then $\mathbf{F}$ is closed at $(\mathbf{x},\ \mathbf{d})$.
14. Let $\mathbf{e}_1,\ \mathbf{e}_2, \ldots, \mathbf{e}_n$ denote the eigenvectors of the symmetric positive definite $n \times n$ matrix $\mathbf{Q}$. For the quadratic problem considered in Sect. 8.2, suppose $\mathbf{x}_0$ is chosen so that $\mathbf{g}_0$ belongs to a subspace M spanned by a subset of the $\mathbf{e}_i$'s. Show that for the method of steepest descent $\mathbf{g}_k \in M$ for all k. Find the rate of convergence in this case.
15. Suppose we use the method of steepest descent to minimize the quadratic function $f(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T\mathbf{Q}(\mathbf{x} - \mathbf{x}^*)$ but we allow a tolerance $\pm\delta\alpha_k$ ($\delta \geqslant 0$) in the line search, that is $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k\mathbf{g}_k$, where

$$(1 - \delta)\overline{\alpha}_k \leqslant \alpha_k \leqslant (1 + \delta)\overline{\alpha}_k$$

    and $\overline{\alpha}_k$ minimizes $f(\mathbf{x}_k - \alpha\mathbf{g}_k)$ over α.

    (a) Find the convergence rate of the algorithm in terms of a and A, the smallest and largest eigenvalues of $\mathbf{Q}$, and the tolerance δ.
    Hint: Assume the extreme case $\alpha_k = (1 + \delta)\overline{\alpha}_k$.
    (b) What is the largest δ that guarantees convergence of the algorithm? Explain this result geometrically.
    (c) Does the sign of δ make any difference?

16. Show that for a quadratic objective function the percentage test and the Goldstein test are equivalent.
17. Suppose in the method of steepest descent for the quadratic problem, the value of α_k is not determined to minimize $E(\mathbf{x}_{k+1})$ exactly but instead only satisfies

$$\frac{E(\mathbf{x}_k) - E(\mathbf{x}_{k+1})}{E(\mathbf{x}_k)} \geqslant \beta\frac{E(\mathbf{x}_k) - \overline{E}}{E(\mathbf{x}_k)}$$

for some $\beta,\ 0 < \beta < 1$, where $\overline{E}$ is the value that corresponds to the best α_k. Find the best estimate for the rate of convergence in this case.

18. Suppose an iterative algorithm of the form $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ is applied to the quadratic problem with matrix $\mathbf{Q}$, where α_k as usual is chosen as the minimum point of the line search and where $\mathbf{d}_k$ is a vector satisfying $\mathbf{d}_k^T \mathbf{g}_k < 0$ and $(\mathbf{d}_k^T \mathbf{g}_k)^2 \geqslant \beta(\mathbf{d}_k^T \mathbf{Q}\mathbf{d}_k)(\mathbf{g}_k^T \mathbf{Q}^{-1}\mathbf{g}_k)$, where $0 < \beta \leqslant 1$. This corresponds to a steepest descent algorithm with "sloppy" choice of direction. Estimate the rate of convergence of this algorithm.
19. Repeat Exercise 18 with the condition on $(\mathbf{d}_k^T \mathbf{g}_k)^2$ replaced by

$$(\mathbf{d}_k^T \mathbf{g}_k)^2 \geqslant \beta(\mathbf{d}_k^T \mathbf{d}_k)(\mathbf{g}_k^T \mathbf{g}_k),\ 0 < \beta \leqslant 1.$$

20. Use the result of Exercise 19 to derive (8.71) for the Gauss-Southwell method.
21. Let $f(x,\ y) = s^2 + y^2 + xy - 3x$.

    (a) Find an unconstrained local minimum point of f.
    (b) Why is the solution to (a) actually a global minimum point?
    (c) Find the minimum point of f subject to $x \geqslant 0,\ y \geqslant 0$.
    (d) If the method of steepest descent were applied to (a), what would be the rate of convergence of the objective function?

22. Find an estimate for the rate of convergence for the modified Newton method

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k(\varepsilon_k \mathbf{I} + \mathbf{F}_k)^{-1}\mathbf{g}_k$$

given by (8.65) and (8.66) when δ is larger than the smallest eigenvalue of $\mathbf{F}(\mathbf{x}^*)$.
23. Prove global convergence of the Gauss-Southwell method.
24. Consider a problem of the form

$$\begin{aligned} &\text{minimize } f(\mathbf{x}) \\ &\text{subject to } \mathbf{x} \geqslant \mathbf{0}, \end{aligned}$$

where $\mathbf{x} \in E^n$. A gradient-type procedure has been suggested for this kind of problem that accounts for the constraint. At a given point $\mathbf{x} = (x_1,\ x_2, \ldots,\ x_n)$, the direction $\mathbf{d} = (d_1,\ d_2,\ \ldots,\ d_n)$ is determined from the gradient $\nabla f(\mathbf{x})^T = \mathbf{g} = (g_1,\ g_2,\ \ldots,\ g_n)$ by

$$d_i = \begin{cases} -g_i \text{ if } x_i > 0 \text{ or } g_i < 0 \\ 0 \text{ if } x_i = 0 \text{ and } g_i \geqslant 0. \end{cases}$$

This direction is then used as a direction of search in the usual manner.

    (a) What are the first-order necessary conditions for a minimum point of this problem?
    (b) Show that $\mathbf{d}$, as determined by the algorithm, is zero only at a point satisfying the first-order conditions.
    (c) Show that if $\mathbf{d} \neq \mathbf{0}$, it is possible to decrease the value of f by movement along $\mathbf{d}$.

(d) If restricted to a compact region, does the Global Convergence Theorem apply? Why?

25. Consider the quadratic problem and suppose $\mathbf{Q}$ has unity diagonal. Consider a coordinate descent procedure in which the coordinate to be searched is at every stage selected randomly, each coordinate being equally likely. Let $\boldsymbol{\varepsilon}_k = \mathbf{x}_k - \mathbf{x}^*$. Assuming $\boldsymbol{\varepsilon}_k$ is known, show that $\overline{\boldsymbol{\varepsilon}_{k+1}^T \mathbf{Q} \boldsymbol{\varepsilon}_{k+1}}$, the expected value of $\boldsymbol{\varepsilon}_{k+1}^T \mathbf{Q} \boldsymbol{\varepsilon}_{k+1}$, satisfies

$$\overline{\boldsymbol{\varepsilon}_{k+1}^T \mathbf{Q} \boldsymbol{\varepsilon}_{k+1}} = \left(1 - \frac{\boldsymbol{\varepsilon}_k^T \mathbf{Q}^2 \boldsymbol{\varepsilon}_k}{n \boldsymbol{\varepsilon}_k^T \mathbf{Q} \boldsymbol{\varepsilon}_k}\right) \boldsymbol{\varepsilon}_k^T \mathbf{Q} \boldsymbol{\varepsilon}_k \leqslant \left(1 - \frac{a^2}{nA}\right) \boldsymbol{\varepsilon}_k^T \mathbf{Q} \boldsymbol{\varepsilon}_k.$$

26. If the matrix $\mathbf{Q}$ has a condition number of 10, how many iterations of steepest descent would be required to get six place accuracy in the minimum value of the objective function of the corresponding quadratic problem?

27. *Stopping criterion.* A question that arises in using an algorithm such as steepest descent to minimize an objective function f is when to stop the iterative process, or, in other words, how can one tell when the current point is close to a solution. If, as with steepest descent, it is known that convergence is linear, this knowledge can be used to develop a stopping criterion. Let $\{f_k\}_{k=0}^{\infty}$ be the sequence of values obtained by the algorithm. We assume that $f_k \to f^*$ linearly, but both f^* and the convergence ratio β are unknown. However we know that, at least approximately,

$$f_{k+1} - f^* = \beta(f_k - f^*)$$

and

$$f_k - f^* = \beta(f_{k-1} - f^*).$$

These two equations can be solved for β and f^*.

(a) Show that

$$f^* = \frac{f_k^2 - f_{k-1} f_{k+1}}{2f_k - f_{k-1} - f_{k+1}}, \quad \beta = \frac{f_{k+1} - f_k}{f_k - f_{k-1}}.$$

(b) Motivated by the above we form the sequence $\{f_k^*\}$ defined by

$$f_k^* = \frac{f_k^2 - f_{k-1} f_{k+1}}{2f_k - f_{k-1} - f_{k+1}}$$

as the original sequence is generated. (This procedure of generating $\{f_k^*\}$ from $\{f_k\}$ is called the Aitken δ^2-process.) If $|f_k - f^*| = \beta^k + o(\beta^k)$ show that $|f_k^* - f^*| = o(\beta^k)$ which means that $\{f_k^*\}$ converges to f^* faster than $\{f_k\}$ does. The iterative search for the minimum of f can then be terminated when $f_k - f_k^*$ is smaller than some prescribed tolerance.

28. Show that the concordant requirement (8.69) can be expressed as

$$\left| \frac{\mathrm{d}}{\mathrm{d}x} f''(x)^{-\frac{1}{2}} \right| \leq 1.$$

29. Assume $f(x)$ and $g(x)$ are self-concordant. Show that the following functions are also self-concordant.

    (a) $af(x)$ for $a > 1$
    (b) $ax + b + f(x)$
    (c) $f(ax + b)$
    (d) $f(x) + g(x)$

1. Prove Lemma 1
2. Consider convex quadratic minimization with matrix $\mathbf{Q}$, and let its distinct positive eigenvalues be $\lambda_1, \lambda_2, \ldots \lambda_K$. Then, if we let the step size in the method of steepest descent be $\alpha_k = \frac{1}{\lambda_k}, k = 1, \ldots, K$, the method terminates in K iterations.
3. Show that the *randomized coordinate descent method* has the expected average convergence rate $(1 - \frac{a}{An})$ for solving strongly convex quadratic programs where a and A are smallest and largest eigenvalues of the Hessian matrix.

References

8.1 For a detailed exposition of Fibonacci search techniques, see Wilde and Beightler [W1]. For an introductory discussion of difference equations, see Lanczos [L1]. Many of these techniques are standard among numerical analysts. See, for example, Kowalik and Osborne [K9], or Traub [T9]. Also see Tamir [T1] for an analysis of high-order fit methods. The use of symmetry arguments to shortcut the analysis is new. The closedness of line search algorithms was established by Zangwill [Z2]. For the line search stopping criteria, see Armijo [A8], Goldstein [G12], and Wolfe [W6].

8.2 For an alternate exposition of this well-known method, see Antosiewicz and Rheinboldt [A7] or Luenberger [L8]. For a proof that the estimate (8.42) is essentially exact, see Akaike [A2]. For early work on the nonquadratic case, see Curry [C10]. For recent work reports in this section see Boyd and Vandenberghe [B23]. The numerical problem considered in the example is a standard one. See Faddeev and Faddeeva [F1].

8.4 The accelerated method of steepest descent is due to Nesterov [190], also see Beck and Teboulle [23]. The BB method is due to Barzilai and Borwein [17], also see Dai and Fletcher [58].

8.5 For good reviews of modern Newton methods, see Fletcher [F9] and Gill, Murray, and Wright [G7]. The theory of self-concordant functions was developed by Nesterov and Nemirovskri, see [N2], [N4], there is a nice reformulation by Renegar [R2] and an introduction in Boyd and Vandenberghe [B23].

8.6 A detailed analysis of coordinate algorithms can be found in Fox [F17] and Isaacson and Keller [I1]. For a discussion of the Gauss-Southwell method, see Forsythe and Wasow [F16]. The proof of convergence speed of the randomized coordinate descent method is essentially due to Nesterov [188] and Lu and Lin [160].