given by (0.89, –0.45) and (–0.45, –0.89), which are used in the second iteration. The procedure is terminated during the second iteration.

Figure 8.15 displays the progress of Rosenbrock's method, where the points generated are numbered sequentially.

8.6 Multidimensional Search Using Derivatives

In the preceding section we described several minimization procedures that use only functional evaluations during the course of optimization. We now discuss some methods that use derivatives in determining the search directions. In particular, we discuss the steepest descent method and the method of Newton.

Method of Steepest Descent

The method of steepest descent, proposed by Cauchy in 1847, is one of the most fundamental procedures for minimizing a differentiable function of several variables. Recall that a vector $\mathbf{d}$ is called a direction of descent of a function f at $\mathbf{x}$ if there exists a $\delta > 0$ such that $f(\mathbf{x} + \lambda\mathbf{d}) < f(\mathbf{x})$ for all $\lambda \in (0, \delta)$. In particular, if $\lim_{\lambda \to 0^+} [f(\mathbf{x}+\lambda\mathbf{d}) - f(\mathbf{x})]/\lambda < 0$, then $\mathbf{d}$ is a direction of descent. The method of steepest descent moves along the direction $\mathbf{d}$ with $\|\mathbf{d}\| = 1$, which minimizes the above limit. Lemma 8.6.1 shows that if f is differentiable at $\mathbf{x}$ with a nonzero gradient, then $-\nabla f(\mathbf{x})/\|\nabla f(\mathbf{x})\|$ is indeed the direction of steepest descent. For this reason, in the presence of differentiability, the method of steepest descent is sometimes called the *gradient method*; it is also referred to as *Cauchy's method.*

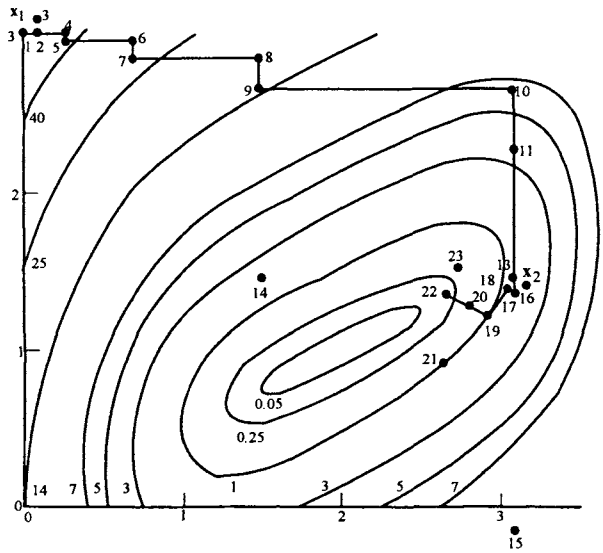

Figure 8.15 Rosenbrock's procedure using discrete steps. (The numbers denote the order in which points are generated.)

8.6.1 Lemma

Suppose that f: $R^n \to R$ is differentiable at $\mathbf{x}$, and suppose that $\nabla f(\mathbf{x}) \neq \mathbf{0}$. Then the optimal solution to the problem to minimize $f'(\mathbf{x}; \mathbf{d})$ subject to $\|\mathbf{d}\| \leq 1$ is given by $\bar{\mathbf{d}} = -\nabla f(\mathbf{x})/\|\nabla f(\mathbf{x})\|$; that is, $-\nabla f(\mathbf{x})/\|\nabla f(\mathbf{x})\|$ is the direction of steepest descent of f at $\mathbf{x}$.

Table 8.10 Summary of Computations for Rosenbrock's Method Using Discrete Steps

Iteration k	$\mathbf{x}_k$ $f(\mathbf{x}_k)$	j	$\mathbf{y}_j$ $f(\mathbf{y}_j)$	Δ_j	$\mathbf{d}_j$	$\mathbf{y}_j + \Delta_j \mathbf{d}_j$ $f(\mathbf{y}_j + \Delta_j \mathbf{d}_j)$
1	(0.00, 3.00) 52.00	1	(0.00, 3.00) 52.00	0.10	(1.00, 0.00)	(0.10, 3.00) 47.84(S)
		2	(0.10, 3.00) 47.84	0.10	(0.00, 1.00)	(0.10, 3.10) 50.24(F)
		1	(0.10, 3.00) 47.84	0.20	(1.00, 0.00)	(0.30, 3.00) 40.84(S)
		2	(0.30, 3.00) 40.84	–0.05	(0.00, 1.00)	(0.30, 2.95) 39.71(S)
		1	(0.30, 2.95) 39.71	0.40	(1.00, 0.00)	(0.70, 2.95) 29.90(S)
		2	(0.70, 2.95) 29.90	–0.10	(0.00, 1.00)	(0.70, 2.85) 27.86(S)
		1	(0.70, 2.85) 27.86	0.80	(1.00, 0.00)	(1.50, 2.85) 17.70(S)
		2	(1.50, 2.85) 17.70	–0.20	(0.00, 1.00)	(1.50, 2.65) 14.50(S)
		1	(1.50, 2.65) 14.50	1.60	(1.00, 0.00)	(3.10, 2.65) 6.30(S)
		2	(3.10, 2.65) 6.30	–0.40	(0.00, 1.00)	(3.10, 2.25) 3.42(S)
		1	(3.10, 2.25) 3.42	3.20	(1.00, 0.00)	(6.30, 2.25) 345.12(F)
		2	(3.10, 2.25) 3.42	–0.80	(0.00, 1.00)	(3.10, 1.45) 1.50(S)

(continued)

Table 8.10 (*continued*)

Iteration k	$\mathbf{x}_k$ $f(\mathbf{x}_k)$	j	$\mathbf{y}_j$ $f(\mathbf{y}_j)$	Δ_j	$\mathbf{d}_j$	$\mathbf{y}_j + \Delta_j \mathbf{d}_j$ $f(\mathbf{y}_j + \Delta_j \mathbf{d}_j)$
		1	(3.10, 1.45) 1.50	–1.60	(1.00, 0.00)	(1.50, 1.45) 2.02(F)
		2	(3.10, 1.45) 1.50	–1.60	(0.00, 1.00)	(3.10, –0.15) 13.02(F)
2	(3.10, 1.45) 1.50	1	(3.10, 1.45) 1.50	0.10	(0.89, –0.45)	(3.19, 1.41) 2.14(F)
		2	(3.10, 1.45) 1.50	0.10	(–0.45, –0.89)	(3.06, 1.36) 1.38(S)
		1	(3.06, 1.36) 1.38	–0.05	(0.89, –0.45)	(3.02, 1.38) 1.15(S)
		2	(3.02, 1.38) 1.15	0.20	(–0.45, –0.89)	(2.93, 1.20) 1.03(S)
		1	(2.93, 1.20) 1.03	–0.10	(0.89, –0.45)	(2.84, 1.25) 0.61(S)
		2	(2.84, 1.25) 0.61	0.40	(–0.45, –0.89)	(2.66, 0.89) 0.96(F)
		1	(2.84, 1.25) 0.61	–0.20	(0.89, –0.45)	(2.66, 1.34) 0.19(S)
		2	(2.66, 1.34) 0.19	–0.20	(–0.45, –0.89)	(2.75, 1.52) 0.40(F)

Proof

From the differentiability of f at $\mathbf{x}$, it follows that

$$f'(\mathbf{x}; \mathbf{d}) = \lim_{\lambda \to 0^+} \frac{f(\mathbf{x} + \lambda \mathbf{d}) - f(\mathbf{x})}{\lambda} = \nabla f(\mathbf{x})^t \mathbf{d}.$$

Thus, the problem reduces to minimizing $\nabla f(\mathbf{x})^t \mathbf{d}$ subject to $\|\mathbf{d}\| \leq 1$. By the Schwartz inequality, for $\|\mathbf{d}\| \leq 1$ we have

$$\nabla f(\mathbf{x})^t \mathbf{d} \geq -\|\nabla f(\mathbf{x})\| \, \|\mathbf{d}\| \geq -\|\nabla f(\mathbf{x})\|,$$

with equality holding throughout if and only if $\mathbf{d} = \bar{\mathbf{d}} \equiv -\nabla f(\mathbf{x}) / \|\nabla f(\mathbf{x})\|$. Thus, $\bar{\mathbf{d}}$ is the optimal solution, and the proof is complete.

Summary of the Steepest Descent Algorithm

Given a point $\mathbf{x}$, the steepest descent algorithm proceeds by performing a line search along the direction $-\nabla f(\mathbf{x})/\|\nabla f(\mathbf{x})\|$ or, equivalently, along the direction $-\nabla f(\mathbf{x})$. A summary of the method is given below.

Initialization Step Let $\varepsilon > 0$ be the termination scalar. Choose a starting point $\mathbf{x}_1$, let $k = 1$, and go to the Main Step.

Main Step

If $\|\nabla f(\mathbf{x}_k)\| < \varepsilon$, stop; otherwise, let $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$, and let λ_k be an optimal solution to the problem to minimize $f(\mathbf{x}_k + \lambda \mathbf{d}_k)$ subject to $\lambda \geq 0$. Let $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k$, replace k by $k + 1$, and repeat the Main Step.

8.6.2 Example

Consider the following problem:

$$\text{Minimize } (x_1 - 2)^4 + (x_1 - 2x_2)^2.$$

We solve this problem using the method of steepest descent, starting with the point (0.00, 3.00). A summary of the computations is given in Table 8.11. After seven iterations, the point $\mathbf{x}_8 = (2.28, 1.15)^t$ is reached. The algorithm is terminated since $\|\nabla f(\mathbf{x}_8)\| = 0.09$ is small. The progress of the method is shown in Figure 8.16. Note that the minimizing point for this problem is (2.00, 1.00).

Convergence of the Steepest Descent Method

Let $\Omega = \{\bar{\mathbf{x}} : \nabla f(\bar{\mathbf{x}}) = \mathbf{0}\}$, and let f be the descent function. The algorithmic map is $\mathbf{A} = \mathbf{MD}$, where $\mathbf{D}(\mathbf{x}) = [\mathbf{x}, \nabla f(\mathbf{x})]$ and $\mathbf{M}$ is the line search map over the closed interval $[0, \infty)$. Assuming that f is continuously differentiable, $\mathbf{D}$ is continuous. Furthermore, $\mathbf{M}$ is closed by Theorem 8.4.1. Therefore, the algorithmic map $\mathbf{A}$ is closed by Corollary 2 to Theorem 7.3.2. Finally, if $\mathbf{x} \notin \Omega$, then $\nabla f(\mathbf{x})^t \mathbf{d} < 0$, where $\mathbf{d} = -\nabla f(\mathbf{x})$. By Theorem 4.1.2, $\mathbf{d}$ is a descent direction, and hence $f(\mathbf{y}) < f(\mathbf{x})$ for $\mathbf{y} \in \mathbf{A}(\mathbf{x})$. Assuming that the sequence generated by the algorithm is contained in a compact set, then by Theorem 7.2.3, the steepest descent algorithm converges to a point with zero gradient.

Zigzagging of the Steepest Descent Method

The method of steepest descent usually works quite well during early stages of the optimization process, depending on the point of initialization. However, as a stationary point is approached, the method usually behaves poorly, taking small,

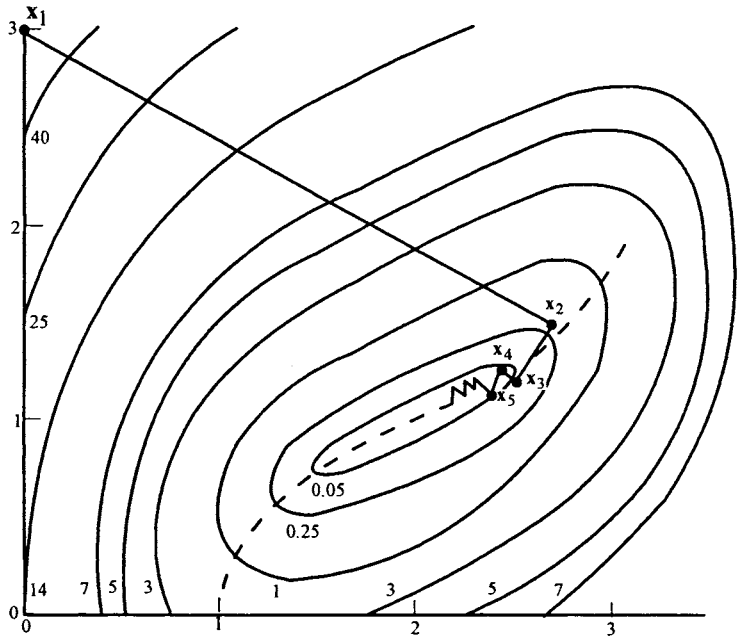

Figure 8.16 Method of steepest descent.

Table 8.11 Summary of Computations for the Method of Steepest Descent

Iteration k	$\mathbf{x}_k$ $f(\mathbf{x}_k)$	$\nabla f(\mathbf{x}_k)$	$\|\nabla f(\mathbf{x}_k)\|$	$\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$	λ_k	$\mathbf{x}_{k+1}$
1	(0.00, 3.00) 52.00	(–44.00, 24.00)	50.12	(44.00, –24.00)	0.062	(2.70, 1.51)
2	(2.70, 1.51) 0.34	(0.73, 1.28)	1.47	(–0.73, –1.28)	0.24	(2.52, 1.20)
3	(2.52, 1.20) 0.09	(0.80, –0.48)	0.93	(–0.80, 0.48)	0.11	(2.43, 1.25)
4	(2.43, 1.25) 0.04	(0.18, 0.28)	0.33	(–0.18, –0.28)	0.31	(2.37, 1.16)
5	(2.37, 1.16) 0.02	(0.30, –0.20)	0.36	(–0.30, 0.20)	0.12	(2.33, 1.18)
6	(2.33, 1.18) 0.01	(0.08, 0.12)	0.14	(–0.08, –0.12)	0.36	(2.30, 1.14)
7	(2.30, 1.14) 0.009	(0.15, –0.08)	0.17	(–0.15, 0.08)	0.13	(2.28, 1.15)
8	(2.28, 1.15) 0.007	(0.05, 0.08)	0.09			

nearly orthogonal steps. This *zigzagging* phenomenon was encountered in Example 8.6.2 and is illustrated in Figure 8.16, in which zigzagging occurs along the valley shown by the dashed lines.

Zigzagging and poor convergence of the steepest descent algorithm at later stages can be explained intuitively by considering the following expression of the function f:

$$f(\mathbf{x}_k + \lambda\mathbf{d}) = f(\mathbf{x}_k) + \lambda\nabla f(\mathbf{x}_k)'\mathbf{d} + \lambda\|\mathbf{d}\|\alpha(\mathbf{x}_k; \lambda\mathbf{d}),$$

where $\alpha(\mathbf{x}_k; \lambda\mathbf{d}) \to 0$ as $\lambda\mathbf{d} \to \mathbf{0}$, and $\mathbf{d}$ is a search direction with $\|\mathbf{d}\| = 1$. If $\mathbf{x}_k$ is close to a stationary point with zero gradient and f is continuously differentiable, then $\|\nabla f(\mathbf{x}_k)\|$ will be small, making the coefficient of λ in the term $\lambda\nabla f(\mathbf{x}_k)'\mathbf{d}$ of a small order of magnitude. Since the steepest descent method employs the linear approximation of f to find a direction of movement, where the term $\lambda\|\mathbf{d}\|\alpha(\mathbf{x}_k; \lambda\mathbf{d})$ is essentially ignored, we should expect that the directions generated at late stages will not be very effective if the latter term contributes significantly to the description of f, even for relatively small values of λ.

As we shall learn in the remainder of the chapter, there are some ways to overcome the difficulties of zigzagging by *deflecting the gradient*. Rather than moving along $\mathbf{d} = -\nabla f(\mathbf{x})$, we can move along $\mathbf{d} = -\mathbf{D}\nabla f(\mathbf{x})$ or along $\mathbf{d} = -\nabla f(\mathbf{x}) + \mathbf{g}$, where $\mathbf{D}$ is an appropriate matrix and $\mathbf{g}$ is an appropriate vector. These correction procedures will be discussed in more detail shortly.

Convergence Rate Analysis for the Steepest Descent Algorithm

In this section we give a more formalized analysis of the zigzagging phenomenon and the empirically observed slow convergence rate of the steepest descent algorithm. This analysis will also afford insights into possible ways of alleviating this poor algorithmic performance.

Toward this end, let us begin by considering a bivariate quadratic function $f(x_1, x_2) = (1/2)(x_1^2 + \alpha x_2^2)$, where $\alpha > 1$. Note that the Hessian matrix to this function is $\mathbf{H} = \text{diag}\{1, \alpha\}$, with eigenvalues 1 and α. Let us define the *condition number* of a positive definite matrix to be the ratio of its largest to smallest eigenvalues. Hence, the condition number of $\mathbf{H}$ for our example is α. The contours of f are plotted in Figure 8.17. Observe that as α increases, a phenomenon that is known as *ill-conditioning*, or a *worsening of the condition number* results, whereby the contours become increasingly skewed and the graph of the function becomes increasingly steep in the x_2 direction relative to the x_1 direction.

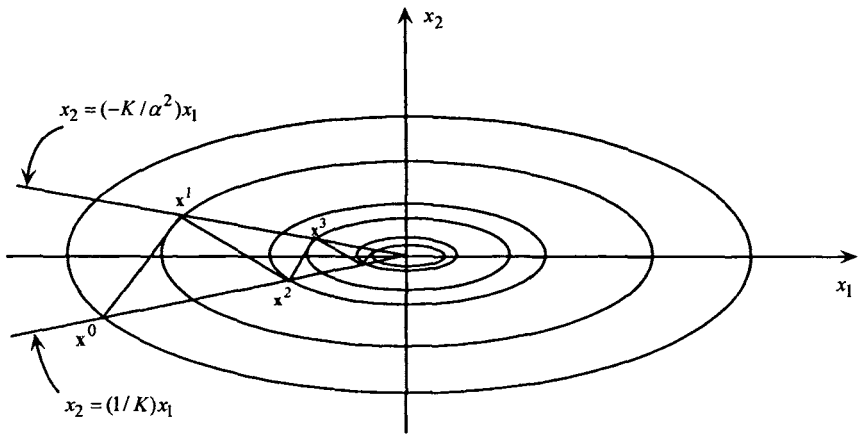

Figure 8.17 Convergence rate analysis of the steepest descent algorithm.

Now, given a starting point $\mathbf{x} = (x_1, x_2)^t$, let us apply an iteration of the steepest descent algorithm to obtain a point $\mathbf{x}_{new} = (x_{1new}, x_{2new})^t$. Note that if $x_1 = 0$ or $x_2 = 0$, the procedure converges to the optimal minimizing solution $\mathbf{x}^* = (0, 0)^t$ in one step. Hence, suppose that $x_1 \neq 0$ and $x_2 \neq 0$. The steepest descent direction is given by $\mathbf{d} = -\nabla f(\mathbf{x}) = -(x_1, \alpha x_2)^t$, resulting in $\mathbf{x}_{new} = \mathbf{x} + \lambda\mathbf{d}$, where λ solves the line search problem to minimize $\theta(\lambda) \equiv f(\mathbf{x} + \lambda\mathbf{d}) = (1/2)[x_1^2(1 - \lambda)^2 + \alpha x_2^2(1 - \alpha\lambda)^2]$ subject to $\lambda \geq 0$. Using simple calculus, we obtain

$$\lambda = \frac{x_1^2 + \alpha^2 x_2^2}{x_1^2 + \alpha^3 x_2^2},$$

so

$$\mathbf{x}_{new} = \left[\frac{\alpha^2 x_1 x_2^2(\alpha - 1)}{x_1^2 + \alpha^3 x_2^2}, \frac{x_1^2 x_2(1 - \alpha)}{x_1^2 + \alpha^3 x_2^2}\right]. \tag{8.13}$$

Observe that $x_{1new}/x_{2new} = -\alpha^2(x_2/x_1)$. Hence, if we begin with a solution $\mathbf{x}^0$ having $x_1^0/x_2^0 = K \neq 0$ and generate a sequence of iterates $\{\mathbf{x}^k\}$, $k = 1, 2,...$, using the steepest descent algorithm, then the sequence of values $\{x_1^k/x_2^k\}$ alternate between the values K and $-\alpha^2/K$ as the sequence $\{\mathbf{x}^k\}$ converges to $\mathbf{x}^* = (0, 0)^t$. For our example this means that the sequence zigzags between the pair of straight lines $x_2 = (1/K)x_1$ and $x_2 = (-K/\alpha^2)x_1$, as shown in Figure

8.17. Note that as the condition number α increases, this zigzagging phenomenon becomes more pronounced. On the other hand, if $\alpha = 1$, then the contours of f are circular, and we obtain $\mathbf{x}^1 = \mathbf{x}^*$ in a single iteration.

To study the rate of convergence, let us examine the rate at which $\{f(\mathbf{x}^k)\}$ converges to the value zero. From (8.13) it is easily verified that

$$\frac{f(\mathbf{x}^{k+1})}{f(\mathbf{x}^k)} = \frac{K_k^2\alpha(\alpha-1)^2}{(K_k^2+\alpha^3)(K_k^2+\alpha)}, \qquad \text{where } K_k \equiv \frac{x_1^k}{x_2^k}. \tag{8.14}$$

Indeed, the expression in (8.14) can be seen to be maximized when $K_k^2 = \alpha^2$ (see Exercise 8.19), so that we obtain

$$\frac{f(\mathbf{x}^{k+1})}{f(\mathbf{x}^k)} \leq \frac{(\alpha-1)^2}{(\alpha+1)^2}. \tag{8.15}$$

Note from (8.15) that $\{f(\mathbf{x}^k)\} \to 0$ at a geometric or linear rate bounded by the ratio $(\alpha-1)^2/(\alpha+1)^2 < 1$. In fact, if we initialize the process with $x_1^0/x_2^0 = K = \alpha$, then, since $K_k^2 = (x_1^k/x_2^k)^2 = \alpha^2$ from above (see Figure 8.17), we get from (8.14) that the convergence ratio $f(\mathbf{x}^{k+1})/f(\mathbf{x}^k)$ is precisely $(\alpha-1)^2/(\alpha+1)^2$. Hence, as α approaches infinity, this ratio approaches 1 from below, and the rate of convergence becomes increasingly slower.

The foregoing analysis can be extended to a general quadratic function $f(\mathbf{x}) = \mathbf{c}^t\mathbf{x} + (1/2)\mathbf{x}^t\mathbf{H}\mathbf{x}$, where $\mathbf{H}$ is an $n \times n$, symmetric, positive definite matrix. The unique minimizer $\mathbf{x}^*$ for this function is given by the solution to the system $\mathbf{H}\mathbf{x}^* = -\mathbf{c}$ obtained by setting $\nabla f(\mathbf{x}^*) = \mathbf{0}$. Also, given an iterate $\mathbf{x}_k$, the optimal step length λ and the revised iterate $\mathbf{x}_{k+1}$ are given by the following generalization of (8.13), where $\mathbf{g}_k \equiv \nabla f(\mathbf{x}_k) = \mathbf{c} + \mathbf{H}\mathbf{x}_k$:

$$\lambda = \frac{\mathbf{g}_k^t\mathbf{g}_k}{\mathbf{g}_k^t\mathbf{H}\mathbf{g}_k} \qquad \text{and} \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - \lambda\mathbf{g}_k. \tag{8.16}$$

Now, to evaluate the rate of convergence, let us employ a convenient measure for convergence given by the following *error function*:

$$e(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^t\mathbf{H}(\mathbf{x}-\mathbf{x}^*) = f(\mathbf{x}) + \frac{1}{2}\mathbf{x}^{*t}\mathbf{H}\mathbf{x}^*, \tag{8.17}$$

where we have used the fact that $\mathbf{H}\mathbf{x}^* = -\mathbf{c}$. Note that $e(\mathbf{x})$ differs from $f(\mathbf{x})$ by only a constant and equals zero if and only if $\mathbf{x} = \mathbf{x}^*$. In fact, it can be shown, analogous to (8.15), that (see Exercise 8.21)

$$e(\mathbf{x}_{k+1}) = \left[1 - \frac{(\mathbf{g}_k^t \mathbf{g}_k)^2}{(\mathbf{g}_k^t \mathbf{H} \mathbf{g}_k)(\mathbf{g}_k^t \mathbf{H}^{-1} \mathbf{g}_k)}\right] e(\mathbf{x}_k) \leq \frac{(\alpha - 1)^2}{(\alpha + 1)^2} e(\mathbf{x}_k), \qquad (8.18)$$

where α is the condition number of $\mathbf{H}$. Hence, $\{e(\mathbf{x}_k)\} \to 0$ at a linear or geometric convergence rate bounded above by $(\alpha - 1)^2/(\alpha + 1)^2$; so, as before, we can expect the convergence to become increasingly slower as α increases, depending on the initial solution $\mathbf{x}_0$.

For continuously twice differentiable nonquadratic functions f: $R^n \to R$, a similar result is known to hold. In such a case, if $\mathbf{x}^*$ is a local minimum to which a sequence $\{\mathbf{x}_k\}$ generated by the steepest descent algorithm converges, and if $\mathbf{H}(\mathbf{x}^*)$ is positive definite with a condition number α, then the corresponding sequence of objective values $\{f(\mathbf{x}_k)\}$ can be shown to converge linearly to the value $f(\mathbf{x}^*)$ at a rate bounded above by $(\alpha - 1)^2/(\alpha + 1)^2$.

Convergence Analysis of the Steepest Descent Algorithm Using Armijo's Inexact Line Search

In Section 8.3 we introduced Armijo's rule for selecting an acceptable, inexact step length during a line search process. It is instructive to observe how such a criterion still guarantees algorithmic convergence. Below, we present a convergence analysis for an inexact steepest descent algorithm applied to a function f: $R^n \to R$ whose gradient function $\nabla f(\mathbf{x})$ is *Lipschitz continuous with constant* $G > 0$ on $S(\mathbf{x}_0) \equiv \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ for some given $\mathbf{x}_0 \in R^n$. That is, we have $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq G\|\mathbf{x} - \mathbf{y}\|$ for all $\mathbf{x}$, $\mathbf{y} \in S(\mathbf{x}_0)$. For example, if the Hessian of f at any point has a norm bounded above by a constant G on $\text{conv}S(\mathbf{x}_0)$ (see Appendix A for the norm of a matrix), then such a function has Lipschitz continuous gradients. This follows from the mean value theorem, noting that for any $\mathbf{x} \neq \mathbf{y} \in S(\mathbf{x}_0)$, $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| = \|H(\hat{\mathbf{x}})(\mathbf{x} - \mathbf{y})\| \leq G\|\mathbf{x} - \mathbf{y}\|$.

The procedure we analyze is the often-used variant of Armijo's rule described in Section 8.3 with parameters $0 < \varepsilon < 1$, $\alpha = 2$, and a fixed-step-length parameter $\bar{\lambda}$, wherein either $\bar{\lambda}$ itself is chosen, if acceptable, or is sequentially halved until an acceptable step length results. This procedure is embodied in the following result.

8.6.3 Theorem

Let f: $R^n \to R$ be such that its gradient $\nabla f(\mathbf{x})$ is Lipschitz continuous with constant $G > 0$ on $S(\mathbf{x}_0) = \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ for some given $\mathbf{x}_0 \in R^n$. Pick some fixed-step-length parameter $\bar{\lambda} > 0$, and let $0 < \varepsilon < 1$. Given any iterate

$\mathbf{x}_k$, define the search direction $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$, and consider Armijo's function $\hat{\theta}(\lambda) = \theta(0) + \lambda\varepsilon\theta'(0)$, $\lambda \geq 0$, where $\theta(\lambda) = f(\mathbf{x}_k + \lambda\mathbf{d}_k)$, $\lambda \geq 0$, is the line search function. If $\mathbf{d}_k = 0$, then stop. Otherwise, find the smallest integer $t \geq 0$ for which $\theta(\bar{\lambda}/2^t) \leq \hat{\theta}(\bar{\lambda}/2^t)$ and define the next iterate as $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k\mathbf{d}_k$, where $\lambda_k \equiv \bar{\lambda}/2^t$. Now suppose that starting with some iterate $\mathbf{x}_0$, this procedure produces a sequence of iterates $\mathbf{x}_0$, $\mathbf{x}_1$, $\mathbf{x}_2$,... . Then either the procedure terminates finitely with $\nabla f(\mathbf{x}_k) = \mathbf{0}$ for some K, or else an infinite sequence $\{\mathbf{x}_k\}$ is generated such that the corresponding sequence $\{\nabla f(\mathbf{x}_k)\} \to \mathbf{0}$.

Proof

The case of finite termination is clear. Hence, suppose that an infinite sequence $\{\mathbf{x}_k\}$ is generated. Note that the Armijo criterion $\theta(\bar{\lambda}/2^t) \leq \hat{\theta}(\bar{\lambda}/2^t)$ is equivalent to $\theta(\bar{\lambda}/2^t) \equiv f(\mathbf{x}_{k+1}) \leq \hat{\theta}(\bar{\lambda}/2^t) = \theta(0) + (\bar{\lambda}\varepsilon/2^t)\nabla f(\mathbf{x}_k)^t\mathbf{d}_k = f(\mathbf{x}_k) - (\bar{\lambda}\varepsilon/2^t)\|\nabla f(\mathbf{x}_k)\|^2$. Hence, $t \geq 0$ is the smallest integer for which

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) \leq \frac{-\bar{\lambda}\varepsilon}{2^t}\|\nabla f(\mathbf{x}_k)\|^2. \tag{8.19}$$

Now, using the mean value theorem, we have, for some strict convex combination $\tilde{\mathbf{x}}$ of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, that

$$\begin{aligned} f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) &= \lambda_k\mathbf{d}_k^t\nabla f(\tilde{\mathbf{x}}) \\ &= -\lambda_k\nabla f(\mathbf{x}_k)^t[\nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_k) + \nabla f(\tilde{\mathbf{x}})] \\ &= -\lambda_k\|\nabla f(\mathbf{x}_k)\|^2 + \lambda_k\nabla f(\mathbf{x}_k)^t[\nabla f(\mathbf{x}_k) - \nabla f(\tilde{\mathbf{x}})] \\ &\leq -\lambda_k\|\nabla f(\mathbf{x}_k)\|^2 + \lambda_k\|\nabla f(\mathbf{x}_k)\|\|\nabla f(\mathbf{x}_k) - \nabla f(\tilde{\mathbf{x}})\|. \end{aligned}$$

But by the Lipschitz continuity of ∇f, noting from (8.19) that the descent nature of the algorithm guarantees that $\mathbf{x}_k \in S(\mathbf{x}_0)$ for all k, we have $\|\nabla f(\mathbf{x}_k) - \nabla f(\tilde{\mathbf{x}})\| \leq G\|\mathbf{x}_k - \tilde{\mathbf{x}}\| \leq G\|\mathbf{x}_k - \mathbf{x}_{k+1}\| = G\lambda_k\|\nabla f(\mathbf{x}_k)\|$. Substituting this above, we obtain

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) \leq -\lambda_k\|\nabla f(\mathbf{x}_k)\|^2(1 - \lambda_k G) = \frac{-\bar{\lambda}}{2^t}\|\nabla f(\mathbf{x}_k)\|^2\left(1 - \frac{\bar{\lambda}G}{2^t}\right). \tag{8.20}$$

Consequently, from (8.20), we know that (8.19) will hold true when t is increased to no larger an integer value than is necessary to make $1 - (\bar{\lambda}G/2^t) \geq$

ε, for then (8.20) will imply (8.19). But this means that $1 - (\bar{\lambda}G/2^{t-1}) < \varepsilon$; that is, $\bar{\lambda}\varepsilon/2^t > \varepsilon(1-\varepsilon)/2G$. Substituting this in (8.19), we get

$$f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k) < \frac{-\varepsilon(1-\varepsilon)}{2G}\|\nabla f(\mathbf{x}_k)\|^2.$$

Hence, noting that $\{f(\mathbf{x}_k)\}$ is a monotone decreasing sequence and so has a limit, taking limits as $t \to \infty$, we get

$$0 \le \frac{-\varepsilon(1-\varepsilon)}{2G} \lim_{k\to\infty} \|\nabla f(\mathbf{x}_k)\|^2,$$

which implies that $\{\nabla f(\mathbf{x}_k)\} \to 0$. This completes the proof.

Method of Newton

In Section 8.2 we discussed Newton's method for minimizing a function of a single variable. The method of Newton is a procedure that deflects the steepest descent direction by premultiplying it by the inverse of the Hessian matrix. This operation is motivated by finding a suitable direction for the quadratic approximation to the function rather than by finding a linear approximation to the function, as in the gradient search. To motivate the procedure, consider the following approximation q at a given point $\mathbf{x}_k$:

$$q(\mathbf{x}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^t(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^t \mathbf{H}(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k),$$

where $\mathbf{H}(\mathbf{x}_k)$ is the Hessian matrix of f at $\mathbf{x}_k$. A necessary condition for a minimum of the quadratic approximation q is that $\nabla q(\mathbf{x}) = \mathbf{0}$, or $\nabla f(\mathbf{x}_k) + \mathbf{H}(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) = \mathbf{0}$. Assuming that the inverse of $\mathbf{H}(\mathbf{x}_k)$ exists, the successor point $\mathbf{x}_{k+1}$ is given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k). \tag{8.21}$$

Equation (8.21) gives the recursive form of the points generated by Newton's method for the multidimensional case. Assuming that $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$, that $\mathbf{H}(\bar{\mathbf{x}})$ is positive definite at a local minimum $\bar{\mathbf{x}}$, and that f is continuously twice differentiable, it follows that $\mathbf{H}(\mathbf{x}_k)$ is positive definite at points close to $\bar{\mathbf{x}}$, and hence the successor point $\mathbf{x}_{k+1}$ is well defined.

It is interesting to note that Newton's method can be interpreted as a *steepest descent algorithm with affine scaling.* Specifically, given a point $\mathbf{x}_k$ at iteration k, suppose that $\mathbf{H}(\mathbf{x}_k)$ is positive definite and that we have a Cholesky factorization (see Appendix A.2) of its inverse given by $\mathbf{H}(\mathbf{x}_k)^{-1} = \mathbf{L}\mathbf{L}^t$, where $\mathbf{L}$ is a lower triangular matrix with positive diagonal elements. Now, consider

the affine scaling transformation $\mathbf{x} = \mathbf{L}\mathbf{y}$. This transforms the function $f(\mathbf{x})$ to the function $F(\mathbf{y}) \equiv f[\mathbf{L}\mathbf{y}]$, and the current point in the $\mathbf{y}$ space is $\mathbf{y}_k = \mathbf{L}^{-1}\mathbf{x}_k$. Hence, we have $\nabla F(\mathbf{y}_k) = \mathbf{L}^t \nabla f[\mathbf{L}\mathbf{y}_k] = \mathbf{L}^t \nabla f(\mathbf{x}_k)$. A unit step size along the negative gradient direction in the $\mathbf{y}$ space will then take us to the point $\mathbf{y}_{k+1} = \mathbf{y}_k - \mathbf{L}^t \nabla f(\mathbf{x}_k)$. Translating this to the corresponding movement in the $\mathbf{x}$ space by premultiplying throughout by $\mathbf{L}$ produces precisely Equation (8.21) and hence yields a steepest descent interpretation of Newton's method. Observe that this comment alludes to the benefits of using an appropriate scaling transformation. Indeed, if the function f was quadratic in the above analysis, then a unit step along the steepest descent direction in the transformed space would be an optimal step along that direction, which would moreover take us directly to the optimal solution in one iteration starting from any given solution.

We also comment here that (8.21) can be viewed as an application of the *Newton–Raphson method* to the solution of the system of equations $\nabla f(\mathbf{x}) = \mathbf{0}$. Given a well-determined system of nonlinear equations, each iteration of the Newton–Raphson method adopts a first-order Taylor series approximation to this equation system at the current iterate and solves the resulting linear system to determine the next iterate. Applying this to the system $\nabla f(\mathbf{x}) = \mathbf{0}$ at an iterate $\mathbf{x}_k$, the first-order approximation to $\nabla f(\mathbf{x})$ is given by $\nabla f(\mathbf{x}_k) + \mathbf{H}(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$. Setting this equal to zero and solving produces the solution $\mathbf{x} = \mathbf{x}_{k+1}$ as given by (8.21).

8.6.4 Example

Consider the following problem:

$$\text{Minimize } (x_1 - 2)^4 + (x_1 - 2x_2)^2.$$

The summary of the computations using Newton's method is given in Table 8.12. At each iteration, $\mathbf{x}_{k+1}$ is given by $\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}(\mathbf{x}_k)^{-1} \nabla f(\mathbf{x}_k)$. After six iterations, the point $\mathbf{x}_7 = (1.83,\ 0.91)^t$ is reached. At this point, $\|\nabla f(\mathbf{x}_7)\| = 0.04$, and the procedure is terminated. The points generated by the method are shown in Figure 8.18.

In Example 8.6.4 the value of the objective function decreased at each iteration. However, this will not generally be the case, so f cannot be used as a descent function. Theorem 8.6.5 indicates that Newton's method indeed converges, provided that we start from a point close enough to an optimal point.

Order-Two Convergence of the Method of Newton

In general, the points generated by the method of Newton may not converge. The reason for this is that $\mathbf{H}(\mathbf{x}_k)$ may be singular, so that $\mathbf{x}_{k+1}$ is not

Table 8.12 Summary of Computations for the Method of Newton

Iteration k	$\mathbf{x}_k$ $f(\mathbf{x}_k)$	$\nabla f(\mathbf{x}_k)$	$\mathbf{H}(\mathbf{x}_k)$	$\mathbf{H}(\mathbf{x}_k)^{-1}$	$-\mathbf{H}(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)$	$\mathbf{x}_{k+1}$
1	(0.00, 3.00) 52.00	(–44.0, 24.0)	$\begin{bmatrix} 50.0 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{384}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 50.0 \end{bmatrix}$	(0.67, –2.67)	(0.67, 0.33)
2	(0.67, 0.33) 3.13	(–9.39, –0.04)	$\begin{bmatrix} 23.23 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{169.84}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 23.23 \end{bmatrix}$	(0.44, 0.23)	(1.11, 0.56)
3	(1.11, 0.56) 0.63	(–2.84, –0.04)	$\begin{bmatrix} 11.50 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{76}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 11.50 \end{bmatrix}$	(0.30, 0.14)	(1.41, 0.70)
4	(1.41, 0.70) 0.12	(–0.80, –0.04)	$\begin{bmatrix} 6.18 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{33.44}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 6.18 \end{bmatrix}$	(0.20, 0.10)	(1.61, 0.80)
5	(1.61, 0.80) 0.02	(–0.22, –0.04)	$\begin{bmatrix} 3.83 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{14.64}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 3.83 \end{bmatrix}$	(0.13, 0.07)	(1.74, 0.87)
6	(1.74, 0.87) 0.005	(–0.07, 0.00)	$\begin{bmatrix} 2.81 & -4.0 \\ -4.0 & 8.0 \end{bmatrix}$	$\frac{1}{6.48}\begin{bmatrix} 8.0 & 4.0 \\ 4.0 & 2.81 \end{bmatrix}$	(0.09, 0.04)	(1.83, 0.91)
7	(1.83, 0.91) 0.0009	(0.0003, –0.04)				

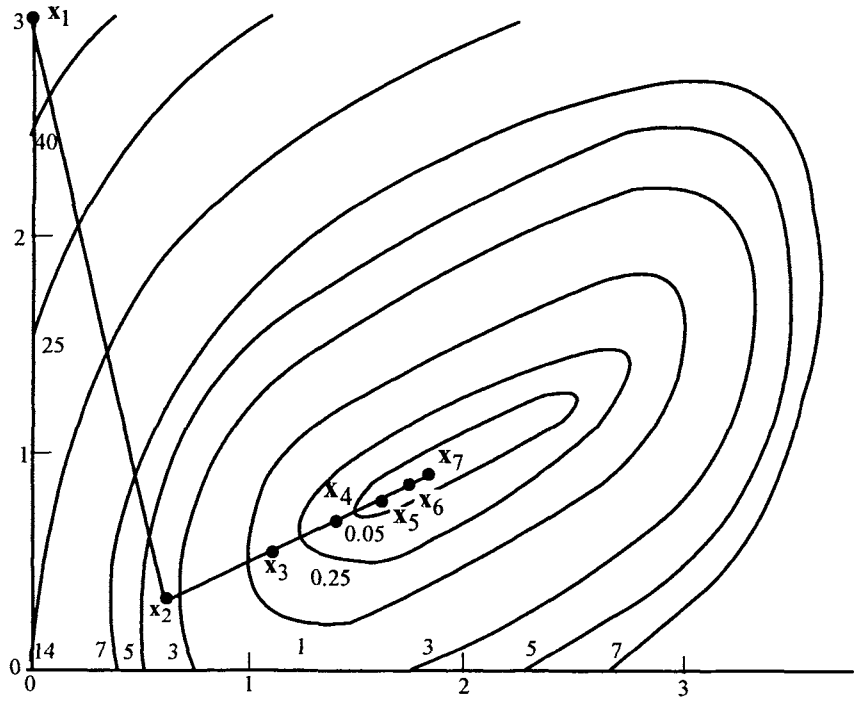

Figure 8.18 Method of Newton.

well defined. Even if $\mathbf{H}(\mathbf{x}_k)^{-1}$ exists, $f(\mathbf{x}_{k+1})$ is not necessarily less than $f(\mathbf{x}_k)$. However, if the starting point is close enough to a point $\bar{\mathbf{x}}$ such that $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$ and $\mathbf{H}(\bar{\mathbf{x}})$ is of full rank, then the method of Newton is well defined and converges to $\bar{\mathbf{x}}$. This is proved in Theorem 8.6.5 by showing that all the assumptions of Theorem 7.2.3 hold true, where the descent function α is given by $\alpha(\mathbf{x}) = \|\mathbf{x} - \bar{\mathbf{x}}\|$.

8.6.5 Theorem

Let f: $R^n \to R$ be continuously twice differentiable. Consider Newton's algorithm defined by the map $\mathbf{A}(\mathbf{x}) = \mathbf{x} - \mathbf{H}(\mathbf{x})^{-1}\nabla f(\mathbf{x})$. Let $\bar{\mathbf{x}}$ be such that $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$ and $\mathbf{H}(\bar{\mathbf{x}})^{-1}$ exists. Let the starting point $\mathbf{x}_1$ be sufficiently close to $\bar{\mathbf{x}}$ so that this proximity implies that there exist k_1, $k_2 > 0$ with $k_1 k_2 \|\mathbf{x}_1 - \bar{\mathbf{x}}\| < 1$ such that

1. $\left\|\mathbf{H}(\bar{\mathbf{x}})^{-1}\right\|^{\dagger} \le k_1$

and by the Taylor series expansion of ∇f,

2. $\left\|\nabla f(\bar{\mathbf{x}}) - \nabla f(\mathbf{x}) - \mathbf{H}(\mathbf{x})(\bar{\mathbf{x}} - \mathbf{x})\right\| \le k_2 \left\|\bar{\mathbf{x}} - \mathbf{x}\right\|^2$

for each $\mathbf{x}$ satisfying $\left\|\mathbf{x} - \bar{\mathbf{x}}\right\| \le \left\|\mathbf{x}_1 - \bar{\mathbf{x}}\right\|$. Then the algorithm converges superlinearly to $\bar{\mathbf{x}}$ with at least an order-two or quadratic rate of convergence.

Proof

Let the solution set $\Omega = \{\bar{\mathbf{x}}\}$ and let $X = \{\mathbf{x} : \left\|\mathbf{x} - \bar{\mathbf{x}}\right\| \le \left\|\mathbf{x}_1 - \bar{\mathbf{x}}\right\|\}$. We prove convergence by using Theorem 7.2.3. Note that X is compact and that the map $\mathbf{A}$ given via (8.21) is closed on X. We now show that $\alpha(\mathbf{x}) = \left\|\mathbf{x} - \bar{\mathbf{x}}\right\|$ is indeed a descent function. Let $\mathbf{x} \in X$, and suppose that $\mathbf{x} \neq \bar{\mathbf{x}}$. Let $\mathbf{y} \in \mathbf{A}(\mathbf{x})$. Then, by the definition of $\mathbf{A}$ and since $\nabla f(\bar{\mathbf{x}}) = \mathbf{0}$, we get

$$\begin{aligned}\mathbf{y} - \bar{\mathbf{x}} &= (\mathbf{x} - \bar{\mathbf{x}}) - \mathbf{H}(\mathbf{x})^{-1}[\nabla f(\mathbf{x}) - \nabla f(\bar{\mathbf{x}})] \\ &= \mathbf{H}(\mathbf{x})^{-1}[\nabla f(\bar{\mathbf{x}}) - \nabla f(\mathbf{x}) - \mathbf{H}(\mathbf{x})(\bar{\mathbf{x}} - \mathbf{x})].\end{aligned}$$

Noting 1 and 2, it then follows that

$$\begin{aligned}\left\|\mathbf{y} - \bar{\mathbf{x}}\right\| &= \left\|\mathbf{H}(\mathbf{x})^{-1}[\nabla f(\bar{\mathbf{x}}) - \nabla f(\mathbf{x}) - \mathbf{H}(\mathbf{x})(\bar{\mathbf{x}} - \mathbf{x})]\right\| \\ &\le \left\|\mathbf{H}(\mathbf{x})^{-1}\right\| \left\|\nabla f(\bar{\mathbf{x}}) - \nabla f(\mathbf{x}) - \mathbf{H}(\mathbf{x})(\bar{\mathbf{x}} - \mathbf{x})\right\| \\ &\le k_1 k_2 \left\|\mathbf{x} - \bar{\mathbf{x}}\right\|^2 \le k_1 k_2 \left\|\mathbf{x}_1 - \bar{\mathbf{x}}\right\| \left\|\mathbf{x} - \bar{\mathbf{x}}\right\| \\ &< \left\|\mathbf{x} - \bar{\mathbf{x}}\right\|.\end{aligned}$$

This shows that α is indeed a descent function. By the corollary to Theorem 7.2.3, we have convergence to $\bar{\mathbf{x}}$. Moreover, for any iterate $\mathbf{x}_k \in X$, the new iterate $\mathbf{y} = \mathbf{x}_{k+1}$ produced by the algorithm satisfies $\left\|\mathbf{x}_{k+1} - \bar{\mathbf{x}}\right\| \le k_1 k_2 \left\|\mathbf{x}_k - \bar{\mathbf{x}}\right\|^2$ from above. Since $\{\mathbf{x}_k\} \to \bar{\mathbf{x}}$, we have at least an order-two rate of convergence.

8.7 Modification of Newton's Method: Levenberg–Marquardt and Trust Region Methods

In Theorem 8.6.5 we have seen that if Newton's method is initialized close enough to a local minimum $\bar{\mathbf{x}}$ with a positive definite Hessian $\mathbf{H}(\bar{\mathbf{x}})$, then it converges quadratically to this solution. In general, we have observed that the

---

† See Appendix A.1 for the norm of a matrix.