1.2 GRADIENT METHODS – CONVERGENCE

We now start our development of computational methods for unconstrained optimization. The conceptual framework of this section is fundamental in nonlinear programming and applies to constrained optimization methods as well.

1.2.1 Descent Directions and Stepsize Rules

As in the case of optimality conditions, the main ideas of unconstrained optimization methods have simple geometrical explanations, but the corresponding convergence analysis is often complex. Thus, for pedagogical reasons, we first discuss informally the methods and their behavior in the present subsection, and we substantiate our conclusions with rigorous analysis in Section 1.2.2.

Consider the problem of unconstrained minimization of a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$. Most of the interesting algorithms for this problem rely on an important idea, called iterative descent that works as follows: We start at some point $x^0$ (an initial guess) and successively generate vectors $x^1, x^2, \ldots$, such that $f$ is decreased at each iteration, that is

$$f(x^{k+1}) < f(x^k), \quad k = 0, 1, \ldots,$$

(cf. Fig. 1.2.1). In doing so, we successively improve our current solution estimate and we hope to decrease $f$ all the way to its minimum. In this section, we introduce a general class algorithms based on iterative descent, and we analyze their convergence to local minima. In Section 1.3 we examine their rate of convergence properties.

Sec. 1.2 Gradient Methods – Convergence

![Figure 1.2.1. Iterative descent for minimizing a function $f$. Each vector in the generated sequence has a lower cost than its predecessor.](image)

Gradient Methods

Given a vector $x \in \mathbb{R}^n$ with $\nabla f(x) \neq 0$, consider the half line of vectors

$$x_\alpha = x - \alpha \nabla f(x), \quad \forall \alpha \geq 0.$$

From the first order Taylor series expansion around $x$ we have

$$f(x_\alpha) = f(x) + \nabla f(x)'(x_\alpha - x) + o(\|x_\alpha - x\|)$$

$$= f(x) - \alpha \|\nabla f(x)\|^2 + o(\alpha \|\nabla f(x)\|),$$

so we can write

$$f(x_\alpha) = f(x) - \alpha \|\nabla f(x)\|^2 + o(\alpha).$$

The term $\alpha \|\nabla f(x)\|^2$ dominates $o(\alpha)$ for $\alpha$ near zero, so for positive but sufficiently small $\alpha$, $f(x_\alpha)$ is smaller than $f(x)$ as illustrated in Fig. 1.2.2.

![Figure 1.2.2. If $\nabla f(x) \neq 0$, there is an interval $(0, \delta)$ of stepsize such that](image)

$$f(x - \alpha \nabla f(x)) < f(x)$$

for all $\alpha \in (0, \delta)$.

Carrying this idea one step further, consider the half line of vectors

$$x_\alpha = x + \alpha d, \quad \forall \alpha \geq 0,$$
Unconstrained Optimization  Chap. 1

where the direction vector \( d \in \mathbb{R}^n \) makes an angle with \( \nabla f(x) \) that is greater than 90 degrees, that is,

\[
\nabla f(x)'d < 0.
\]

Again by Taylor's theorem we have

\[
f(x+\alpha d) = f(x) + \alpha \nabla f(x)'d + o(\alpha).
\]

For \( \alpha \) near zero, the term \( \alpha \nabla f(x)'d \) dominates \( o(\alpha) \) and as a result, for positive but sufficiently small \( \alpha \), \( f(x+\alpha d) \) is smaller than \( f(x) \) as illustrated in Fig. 1.2.3.

![Figure 1.2.3](image)

The preceding observations form the basis for the broad and important class of algorithms

\[
x^{k+1} = x^k + \alpha^k d^k, \quad k = 0, 1, \ldots.
\]

(1.5)

where, if \( \nabla f(x^k) \neq 0 \), the direction \( d^k \) is chosen so that

\[
\nabla f(x^k)'d^k < 0,
\]

(1.6)

and the stepsize \( \alpha^k \) is chosen to be positive. If \( \nabla f(x^k) = 0 \), the method stops, i.e., \( x^{k+1} = x^k \) (equivalently we choose \( d^k = 0 \)). In view of the relation (1.6) of the direction \( d^k \) and the gradient \( \nabla f(x^k) \), we call algorithms of this type gradient methods. [There is no universally accepted name for these algorithms; some authors reserve the name "gradient method" for the special case where \( d^k = -\nabla f(x^k) \).] The majority of the gradient methods that we will consider are also descent algorithms; that is, the stepsize \( \alpha^k \) is selected so that

\[
f(x^k + \alpha^k d^k) < f(x^k), \quad k = 0, 1, \ldots.
\]

(1.7)

Sec. 1.2 Gradient Methods - Convergence

However, there are some exceptions, which will be described shortly.

There is a large variety of possibilities for choosing the direction \( d^k \) and the stepsize \( \alpha^k \) in a gradient method. Indeed there is no single gradient method that can be recommended for all or even most problems. Otherwise said, given any one of the numerous methods and variations thereof that we will discuss, there are interesting types of problems for which this method is well-suited. Our principal analytical aim is to develop a new guiding principles for understanding the performance of broad classes of methods and for appreciating the practical contexts in which their use is most appropriate.

Selecting the Descent Direction

Many gradient methods are specified in the form

\[
x^{k+1} = x^k - \alpha^k D^k \nabla f(x^k),
\]

(1.8)

where \( D^k \) is a positive definite symmetric matrix. Since \( d^k = -D^k \nabla f(x^k) \), the descent condition \( \nabla f(x^k)'d^k < 0 \) is written as

\[
\nabla f(x^k)'D^k \nabla f(x^k) > 0,
\]

and holds thanks to the positive definiteness of \( D^k \).

Here are some examples of choices of the matrix \( D^k \), resulting in methods that are widely used:

Steepest Descent

\[
D^k = I, \quad k = 0, 1, \ldots,
\]

where \( I \) is the \( n \times n \) identity matrix. This is the simplest choice but it often leads to slow convergence, as we will see in Section 1.3. The difficulty is illustrated in Fig. 1.2.4 and motivates the methods of the subsequent examples. The name "steepest descent" is derived from an interesting property of the (normalized) negative gradient direction \( d^k = -\nabla f(x^k) / ||\nabla f(x^k)|| \); among all directions \( d \in \mathbb{R}^n \) that are normalized so that \( ||d|| = 1 \), it is the one that minimizes the slope \( \nabla f(x^k)'d \) of the cost \( f(x^k + \alpha d) \) along the direction \( d \) at \( \alpha = 0 \). Indeed, by the Schwartz inequality (Prop. A.2 in Appendix A), we have for all \( d \) with \( ||d|| = 1 \),

\[
\nabla f(x^k)'d \geq -||\nabla f(x^k)|| \cdot ||d|| = -||\nabla f(x^k)||,
\]

and it is seen that equality is attained above for \( d \) equal to \( -\nabla f(x^k) / ||\nabla f(x^k)|| \).
Figure 1.2.4. Slow convergence of the steepest descent method

\[ x^{k+1} = x^k - \alpha^k \nabla f(x^k) \]

when the equal cost surfaces of \( f \) are "diagonalized." The difficulty is that the gradient direction is almost orthogonal to the direction that leads to the minimum. As a result, the method is zig-zagging without making fast progress.

**Newton's Method**

\[ D^k = \left( \nabla^2 f(x^k) \right)^{-1}, \quad k = 0, 1, \ldots \]

provided \( \nabla^2 f(x^k) \) is positive definite. If \( \nabla^2 f(x^k) \) is not positive definite, some modification is necessary as will be explained in Section 1.4. The idea in Newton's method is to minimize at each iteration the quadratic approximation of \( f \) around the current point \( x^k \) given by

\[ f^k(x) = f(x^k) + \nabla f(x^k)'(x - x^k) + \frac{1}{2}(x - x^k)' \nabla^2 f(x^k)(x - x^k), \]

(see Fig. 1.2.5). By setting the derivative of \( f^k(x) \) to zero,

\[ \nabla f(x^k) + \nabla^2 f(x^k)(x - x^k) = 0, \]

we obtain the next iterate \( x^{k+1} \) as the minimum of \( f^k(x) \):

\[ x^{k+1} = x^k - \left( \nabla^2 f(x^k) \right)^{-1} \nabla f(x^k). \]

This is the pure Newton iteration. It corresponds to the more general iteration

\[ x^{k+1} = x^k - \alpha^k \left( \nabla^2 f(x^k) \right)^{-1} \nabla f(x^k), \]

where the stepsize \( \alpha^k = 1 \). Note that Newton's method finds the global minimum of a positive definite quadratic function in a single iteration (assuming \( \alpha^k = 1 \)). More generally, Newton's method typically converges very fast asymptotically and does not exhibit the zig-zagging behavior of steepest descent, as we will show in Section 1.4. For this reason, many other methods try to emulate Newton's method. Some examples will be given shortly.

**Sec. 1.2 Gradient Methods - Convergence**

![Illustration of the fast convergence rate of Newton's method with a stepsize equal to one. Given \( x^k \), the method obtains \( x^{k+1} \) as the minimum of a quadratic approximation of \( f \) based on a second order Taylor expansion around \( x^k \).][1]

**Diagonally Scaled Steepest Descent**

\[ D^k = \begin{pmatrix} d_{11}^k & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & d_{22}^k & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & d_{n-1,n-1}^k & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & d_{nn}^k & 0 \end{pmatrix}, \quad k = 0, 1, \ldots. \]

where \( d_{ij}^k \) are positive scalars, thus ensuring that \( D^k \) is positive definite. A popular choice, resulting in a method known as a diagonal approximation to Newton's method, is to take \( d_{ii}^k \) to be an approximation to the inverted second partial derivative of \( f \) with respect to \( x_i \), that is,

\[ d_{ii}^k = \left( \frac{\partial^2 f(x_i)}{\partial x_i^2} \right)^{-1}. \]

**Modified Newton's Method**

\[ D^k = \left( \nabla^2 f(x^k) \right)^{-1}, \quad k = 0, 1, \ldots. \]

provided \( \nabla^2 f(x^k) \) is positive definite. This method is the same as Newton's method except that to economize on overhead, the Hessian matrix is not recalculated at each iteration. A related method is obtained when the Hessian is recomputed every \( p > 1 \) iterations.
Discretized Newton's Method

\[ D^k = \left( H(x^k) \right)^{-1}, \quad k = 0, 1, \ldots, \]

where \( H(x^k) \) is a positive definite symmetric approximation of \( \nabla^2 f(x^k) \), formed by using finite difference approximations of the second derivatives, based on first derivatives or values of \( f \).

Gauss-Newton Method

This method is applicable to the problem of minimizing the sum of squares of real valued functions \( g_1, \ldots, g_m \), a problem often encountered in statistical data analysis and in the context of neural network training (see Section 1.5). By denoting \( g = (g_1, \ldots, g_m) \), the problem is written as

\[
\text{minimize} \quad f(x) = \frac{1}{2} \| g(x) \|^2 = \frac{1}{2} \sum_{i=1}^{m} (g_i(x))^2 \\
\text{subject to} \quad x \in \mathbb{R}^n.
\]

We choose

\[ D^k = \left( \nabla g(x^k)^2 \right)^{-1}, \quad k = 0, 1, \ldots, \]

assuming the matrix \( \nabla g(x^k)^2 \) is invertible. The latter matrix is always positive semidefinite, and it is positive definite and hence invertible if and only if the matrix \( \nabla g(x^k)^2 \) has rank \( n \) (Prop. A.29 in Appendix A). Since

\[ \nabla f(x^k) = \nabla g(x^k)^2 g(x^k), \]

the Gauss-Newton method takes the form

\[ x^{k+1} = x^k - \alpha^k \left( \nabla g(x^k)^2 \right)^{-1} \nabla g(x^k)^2 g(x^k). \quad (1.9) \]

We will see in Section 1.5 that the Gauss-Newton method may be viewed as an approximation to Newton's method, particularly when the optimal value of \( \| g(x) \|^2 \) is small.

Other choices of \( D^k \) yield the class of Quasi-Newton methods discussed in Section 1.7. There are also some interesting methods where the direction \( d^k \) is not usually expressed as \( d^k = -D^k \nabla f(x^k) \). Important examples are the conjugate gradient method and the coordinate descent methods, which are discussed in Sections 1.6 and 1.8, respectively.

Stepsize Selection

There are a number of rules for choosing the stepsize \( \alpha^k \) in a gradient method. We list some that are used widely in practice:

Sec. 1.2 Gradient Methods - Convergence

Minimization Rule

Here \( \alpha^k \) is such that the cost function is minimized along the direction \( d^k \), that is, \( \alpha^k \) satisfies

\[ f(x^k + \alpha^k d^k) = \min_{\alpha \geq 0} f(x^k + \alpha d^k). \quad (1.10) \]

Limited Minimization Rule

This is a version of the minimization rule, which is more easily implemented in many cases. A fixed scalar \( s > 0 \) is selected and \( \alpha^k \) is chosen to yield the greatest cost reduction over all stepizes in the interval \( [0, s] \), i.e.,

\[ f(x^k + \alpha^k d^k) = \min_{\alpha \in [0, s]} f(x^k + \alpha d^k). \]

The minimization and limited minimization rules must typically be implemented with the aid of one-dimensional line search algorithms (see Appendix C). In general, minimizing stepsize cannot be computed exactly, and in practice, the line search is stopped once a stepsize \( \alpha^k \) satisfying some termination criterion is obtained. Some stopping criteria are discussed in Exercise 1.2.16.

Successive Stepsize Reduction - Armijo Rule

To avoid the often considerable computation associated with the line minimization rules, it is natural to consider rules based on successive stepsizes reduction. In the simplest rule of this type an initial stepsize \( s \) is chosen, and if the corresponding vector \( x^k + sd^k \) does not yield an improved value of \( f \), that is, \( f(x^k + sd^k) > f(x^k) \), the stepsize is reduced, perhaps repeatedly, by a certain factor, until the value of \( f \) is improved. While this method often works in practice, it is theoretically unsound because the cost improvement obtained at each iteration may not be substantial enough to guarantee convergence to a minimum. This is illustrated in Fig. 1.2.6.

The Armijo rule is essentially the successive reduction rule just described, suitably modified to eliminate the theoretical convergence difficulties shown in Fig. 1.2.6. Here, fixed scalars \( s, \beta, \) and \( \sigma \), with \( 0 < \beta < 1 \), and \( 0 < \sigma \leq 1 \) are chosen, and we set \( \alpha^k = \beta^m s \), where \( m \) is the first nonnegative integer \( m \) for which

\[ f(x^k) - f(x^k + \beta^m s d^k) \geq -\sigma \beta^m s \nabla f(x^k)^2 d^k. \quad (1.11) \]

In other words, the stepsizes \( \beta^m s \), \( m = 0, 1, \ldots \), are tried successively until the above inequality is satisfied for \( m = m_k \). Thus, the cost improvement must not be just positive; it must be sufficiently large as per the test (1.11). Figure 1.2.7 illustrates the rule.
Figure 1.2.6. Example of failure of the successive stepsize reduction rule for the one-dimensional function

\[
f(x) = \begin{cases} 
3(1-x)^2 - 2(1-x) & \text{if } x > 1, \\
3(1+x)^2 - 2(1+x) & \text{if } x < -1, \\
\frac{3}{4} & \text{if } -1 \leq x \leq 1.
\end{cases}
\]

The gradient of \( f \) is given by

\[
\nabla f(x) = \begin{cases} 
\frac{3x}{2} + \frac{1}{2} & \text{if } x > 1, \\
\frac{3x}{2} - \frac{1}{2} & \text{if } x < -1, \\
\frac{3}{2x} & \text{if } -1 \leq x \leq 1.
\end{cases}
\]

It is seen that \( f \) is strictly convex, continuously differentiable, and is minimized at \( x^* = 0 \). Furthermore, for any two scalars \( x \) and \( \tilde{x} \) we have

\[
f(x) < f(\tilde{x}) \quad \text{if and only if} \quad |x| < |\tilde{x}|.
\]

We have for \( x > 1 \)

\[
x - \nabla f(x) = x - \frac{3x}{2} - \frac{1}{2} = \left(1 + \frac{x - 1}{2}\right).
\]

from which it can be verified that \( |x - \nabla f(x)| < |x| \), so that \( f(x - \nabla f(x)) < f(x) \) and \( x - \nabla f(x) < x \). Similarly, for \( x < -1 \), we have \( f(x - \nabla f(x)) < f(x) \) and \( x - \nabla f(x) > x \). Consider now the steepest descent iteration where the step size is successively reduced from an initial step size \( s = 1 \) until descent is obtained. Let the starting point satisfy \( |x^0| > 1 \). From the preceding equations, it follows that \( f(x^k - \nabla f(x^k)) < f(x^k) \) and the step size \( s = 1 \) will be accepted by the method. Thus, the next point is \( x^1 = x^0 - \nabla f(x^0) \), which satisfies \( |x^1| > 1 \). By repeating the preceding argument, we see that the generated sequence \( \{x^k\} \) satisfies \( |x^k| > 1 \) for all \( k \), and cannot converge to the unique stationary point \( x^* = 0 \). In fact, it can be shown that \( \{x^k\} \) will have two limit points, \( z = 1 \) and \( z = -1 \), for every \( x^0 \) with \( |x^0| > 1 \).

Sec. 1.2 Gradient Methods - Convergence

Figure 1.2.7. Line search by the Armijo rule. We start with the trial stepsize \( s \) and continue with \( \beta s, \beta^2 s, \ldots \) until the first time that \( \beta^m s \) falls within the set of suitable stepsizes \( \alpha \) satisfying the inequality

\[
f(x^k) - f(x^k + \alpha d^k) \geq -\sigma \alpha \nabla f(x^k)' d^k.
\]

While this set need not be an interval, it will always contain an interval of the form \([0, \delta]\) with \( \delta > 0 \), provided \( \nabla f(x^k)' d^k < 0 \). For this reason the stepsize \( \alpha^k \) chosen by the Armijo rule is well defined and will be found after a finite number of trial evaluations of \( f \) at the points \((x^k + \alpha d^k), (x^k + \beta d^k), \ldots\)

Usually \( \sigma \) is chosen close to zero, for example, \( \sigma \in [10^{-5}, 10^{-1}] \). The reduction factor \( \beta \) is usually chosen from 1/2 to 1/10 depending on the confidence we have on the quality of the initial stepsize \( s \). We can always take \( s = 1 \) and multiply the direction \( d^k \) by a scaling factor. Many methods, such as Newton-like methods, incorporate some type of implicit scaling of the direction \( d^k \), which makes \( s = 1 \) a good stepsize choice (see the discussion on rate of convergence in Section 1.3). If a suitable scaling factor for \( d^k \) is not known, one may use various ad hoc schemes to determine one. For example, a simple possibility is based on quadratic interpolation of the function

\[
g(\alpha) = f(x^k + \alpha d^k),
\]

which is the cost along the direction \( d^k \), viewed as a function of the stepsize \( \alpha \). In this scheme, we select some stepsize \( \tilde{\alpha} \), evaluate \( g(\tilde{\alpha}) \), and perform the quadratic interpolation of \( g \) on the basis of \( g(\alpha) = f(x^k), dg(\alpha)/d\alpha = \nabla f(x^k)' d^k, \) and \( g(\delta) \). If \( \alpha \) minimizes the quadratic interpolation, we replace \( \tilde{\alpha} \) by \( \alpha^k = \tilde{\alpha} d^k \), and we use an initial stepsize \( s = 1 \).
Goldstein Rule

Here, a fixed scalar $\sigma \in (0, 1/2)$ is selected, and $\alpha^k$ is chosen to satisfy
\[
\sigma \leq \frac{f(x^* + \alpha^k d^k) - f(x^*)}{\alpha^k \nabla f(x^*)^T d^k} \leq 1 - \sigma,
\]
(cf. Fig. 1.2.8). It is possible to show that if $f$ is bounded below, there exists an interval of stepsizes $\alpha^k$ for which the relation above is satisfied. There are fairly simple algorithms for finding such a stepsize but we will not go into the details, since in practice the simpler Armijo rule seems to be universally preferred. The Goldstein rule is included here primarily because of its historical significance: it was the first sound proposal for a general-purpose stepsize rule that did not rely on line minimisation, and it embodies the fundamental idea on which the subsequently proposed Armijo rule was based.

![Figure 1.2.8. Illustration of the set of stepsizes that are acceptable in the Goldstein rule.](image)

**Constant Stepsize:**

Here a fixed stepsize $s > 0$ is selected and
\[
\alpha^k = s, \quad k = 0, 1, \ldots
\]

The constant stepsize rule is very simple. However, if the stepsize is too large, divergence will occur, while if the stepsize is too small, the rate of convergence may be very slow. Thus, the constant stepsize rule is useful only for problems where an appropriate constant stepsize value is known or can be determined fairly easily. (A method that attempts to determine automatically an appropriate value of stepsize is given in Exercise 1.2.20, for the case where $f$ is convex.)

**Diminishing Stepsize**

Here the stepsize converges to zero,
\[
\alpha^k \to 0.
\]

This stepsize rule is different than the preceding ones in that it does not guarantee descent at each iteration, although descent becomes more likely as the stepsize diminishes. One difficulty with a diminishing stepsize is that it may become so small that substantial progress cannot be maintained, even when far from a stationary point. For this reason, we require that
\[
\sum_{k=0}^{\infty} \alpha^k = \infty.
\]

The last condition guarantees that $\{x^k\}$ does not converge to a nonstationary point. Indeed, if $x^k \to \bar{x}$, then for any large index $m$ and $n$ ($m > n$), we have
\[
x^m \approx x^n \approx \bar{x}, \quad x^m = x^n - \sum_{i=n}^{m-1} \alpha^i \nabla f(x),
\]
which is a contradiction when $\bar{x}$ is nonstationary and $\sum_{i=n}^{m-1} \alpha^i$ can be made arbitrarily large. Generally, the diminishing stepsize rule has good theoretical convergence properties (see Prop. 1.2.4, and Exercises 1.2.13 and 1.2.14). The associated convergence rate tends to be slow, so this stepsize rule is used primarily in situations where slow convergence is inevitable; for example, in singular problems or when the gradient is calculated with error (see the discussion later in this section).

**Convergence Issues**

Let us now delineate the type of convergence issues that we would like to clarify. We will first discuss informally these issues and we will state and prove the associated convergence results in Section 1.2.2. Given a gradient method, ideally we would like the generated sequence $\{x^k\}$ to converge to a global minimum. Unfortunately, however, this is too much to expect, at least when $f$ is not convex, because of the presence of local minima that are not global. Indeed a gradient method is guided downhill by the form of $f$ near the current iterate, while being oblivious to the global structure of $f$. 
and thus, can easily get attracted to any type of minimum, global or not. Furthermore, if a gradient method starts or lands at any stationary point, including a local maximum, it stops at that point. Thus, the most we can expect from a gradient method is that it converges to a stationary point. Such a point is a global minimum if \( f \) is convex, but this need not be so for nonconvex problems. Thus, it must be recognized that gradient methods can be quite inadequate, particularly if little is known about the location and/or other properties of global minima. For such problems one must either try an often difficult and frustrating process of running a gradient method from multiple starting points, or else resort to a fundamentally different approach.

Generally, depending on the nature of the cost function \( f \), the sequence \( \{x^k\} \) generated by a gradient method need not have a limit point; in fact \( \{x^k\} \) is typically unbounded if \( f \) has no local minima. If, however, we know that the level set \( \{ x \mid f(x) \leq f(x^0) \} \) is bounded, and the stepsize is chosen to enforce descent at each iteration, then the sequence \( \{x^k\} \) must be bounded since it belongs to this level set. It must then have at least one limit point; this is because every bounded sequence has at least one limit point (see Prop. A.5 of Appendix A).

Even if \( \{x^k\} \) is bounded, convergence to a single limit point may not be easy to guarantee. However, it can be shown that local minima, which are isolated stationary points (unique stationary points within some open sphere), tend to attract most types of gradient methods, that is, once a gradient method gets sufficiently close to such a local minimum, it converges to it. This is the subject of a simple and remarkably powerful result, the capture theorem, which is given in the next subsection (Prop. 1.2.5).

Exercise 1.2.13 develops another result of convergence to a single limit for the steepest descent method, under the assumption that \( f \) is convex. Generally, if there is a connected set of multiple global minima, it is theoretically possible for \( \{x^k\} \) to have multiple limit points (see Exercise 1.2.18), but the occurrence of such a phenomenon has never been documented in practice.

**Limit Points of Gradient Methods**

We now address the question of whether each limit point of a sequence \( \{x^k\} \) generated by a gradient method is a stationary point. From the first order Taylor expansion

\[
 f(x^{k+1}) = f(x^k) + \alpha^k \nabla f(x^k)^T d^k + o(\alpha^k),
\]

we see that if the slope of \( f \) at \( x^k \) along the direction \( d^k \), which is \( \nabla f(x^k)^T d^k \), has "substantial" magnitude, the rate of progress of the method will also tend to be substantial. If on the other hand, the directions \( d^k \) tend to become asymptotically orthogonal to the gradient direction,

\[
 \frac{\nabla f(x^k)^T d^k}{\|\nabla f(x^k)\| \|d^k\|} \to 0,
\]

as \( x^k \) approaches a nonstationary point, there is a chance that the method will get "stuck" near that point. To ensure that this does not happen, we consider rather technical conditions on the directions \( d^k \), which are either naturally satisfied or can be easily enforced in most algorithms of interest.

One such condition for the case where

\[
 d^k = -D^k \nabla f(x^k),
\]

is to assume that the eigenvalues of the positive definite symmetric matrix \( D^k \) are bounded above and bounded away from zero, that is, for some positive scalars \( c_1 \) and \( c_2 \), we have

\[
 c_1 \|z\|^2 \leq z^TD^kz \leq c_2 \|z\|^2, \quad \forall \ z \in \mathbb{R}^n, \ k = 0, 1, \ldots \quad (1.12)
\]

It can be seen then that

\[
 |\nabla f(x^k)^T d^k| = |\nabla f(x^k)^T D^k \nabla f(x^k)| \geq c_1 \|\nabla f(x^k)\|^2
\]

and

\[
 \|d^k\|^2 = |\nabla f(x^k)^T D^k \nabla f(x^k)| \leq c_2 \|\nabla f(x^k)\|^2,
\]

where we have used the fact that, from Eq. (1.12), \( c_2 \) is no less than the largest eigenvalue of \( D^k \), and that the eigenvalues of \( (D^k)^2 \) are equal to the squares of the corresponding eigenvalues of \( D^k \) (Props. A.18 and A.13 in Appendix A). Thus, as long as \( \nabla f(x^k) \) does not tend to zero, \( \nabla f(x^k)^T d^k \) and \( d^k \) cannot become asymptotically orthogonal.

We now introduce another "nonorthogonality" type of condition, which is more general than the "bounded eigenvalues" condition (1.12). Let us consider the sequence \( \{x^k, d^k\} \) generated by a given gradient method. We say that the direction sequence \( \{d^k\} \) is gradient related to \( \{x^k\} \) if the following property can be shown:

For any subsequence \( \{x^k\}_{k \in K} \) that converges to a nonstationary point, the corresponding subsequence \( \{d^k\}_{k \in K} \) is bounded and satisfies

\[
 \limsup_{k \to \infty} \frac{\nabla f(x^k)^T d^k}{d^k} < 0. \quad (1.13)
\]

In particular, if \( \{d^k\} \) is gradient related, it follows that if a subsequence \( \{\nabla f(x^k)\}_{k \in K} \) tends to a nonzero vector, the corresponding subsequence of directions \( d^k \) is bounded and does not tend to be orthogonal to \( \nabla f(x^k) \). Roughly, this means that \( d^k \) does not become "too small" or "too large" relative to \( \nabla f(x^k) \), and that the angle between \( d^k \) and \( \nabla f(x^k) \) does not get "too close" to 90 degrees.

We can often guarantee a priori that \( \{d^k\} \) is gradient related. In particular, if \( d^k = -D^k \nabla f(x^k) \) and the eigenvalues of \( D^k \) are bounded as in the "bounded eigenvalues" condition (1.12), it can be seen that \( \{d^k\} \) is
Gradient related, provided \( x^k \) is nonstationary for all \( k \) (if \( x^k \) is stationary for some \( k \), the issue of convergence in effect does not arise). Two other examples of conditions that, if satisfied for some scalars \( c_1 > 0, c_2 > 0, p_1 \geq 0, p_2 \geq 0 \), and all \( k \), guarantee that \( \{d^k\} \) is gradient related are

\[
\begin{align*}
  (a) & \quad c_1 \|\nabla f(x^k)\|^{p_1} \leq -\nabla f(x^k)^T d^k, \\
  & \quad \|d^k\| \leq c_2 \|\nabla f(x^k)\|^{p_2}.
\end{align*}
\]

(b) \[d^k = -D^k \nabla f(x^k),\]

with \( D^k \) a positive definite symmetric matrix satisfying

\[c_1 \|\nabla f(x^k)\|^{p_1} \|z\|^2 \leq z^T D^k z \leq c_2 \|\nabla f(x^k)\|^{p_2} \|z\|^2, \quad \forall z \in \mathbb{R}^n.
\]

This condition generalizes the "bounded eigenvalues" condition (1.12), which is obtained for \( p_1 = p_2 = 0 \).

An important convergence result is that if \( \{d^k\} \) is gradient related and the minimization rule, or the limited minimization rule, or the Armijo rule is used, then all limit points of \( \{x^k\} \) are stationary. This is shown in Prop. 1.2.1, given in the next subsection. Proposition 1.2.2 provides a similar result for the Goldstein rule.

When a constant stepsize is used, convergence can be proved assuming that the stepsize is sufficiently small and that \( f \) satisfies some further conditions (cf. Prop. 1.2.3). Under the same conditions, convergence can also be proved for a diminishing stepsize.

There is a common line of proof for these convergence results. The main idea is that the cost function is improved at each iteration and that, based on our assumptions, the improvement is "substantial" near a nonstationary point, i.e., it is bounded away from zero. We then argue that the algorithm cannot approach a nonstationary point, since in this case the total cost improvement would accumulate to infinity.

**Termination of Gradient Methods**

Generally, gradient methods are not finitely convergent, so it is necessary to have criteria for terminating the iterations with some assurance that we are reasonably close to at least a local minimum. A typical approach is to stop the computation when the norm of the gradient becomes sufficiently small, that is, when a point \( x^k \) is obtained with

\[\|\nabla f(x^k)\| \leq \epsilon,\]

where \( \epsilon \) is a small positive scalar. Unfortunately, it is not known a priori how small one should take \( \epsilon \) in order to guarantee that the final point \( x^k \) is a "good" approximation to a stationary point. The appropriate value of \( \epsilon \) depends on how the problem is scaled. In particular, if \( f \) is multiplied by some scalar, the appropriate value of \( \epsilon \) is also multiplied by the same scalar. It is possible to correct this difficulty by replacing the criterion \( \|\nabla f(x^k)\| \leq \epsilon \) with

\[
\frac{\|\nabla f(x^k)\|}{\|\nabla f(x^k)\|} \leq \epsilon.
\]

Still, however, the gradient norm \( \|\nabla f(x^k)\| \) depends on all the components of the gradient, and depending on how the optimization variables are scaled, the preceding termination criterion may not work well. In particular, some components of the gradient may be much smaller than others, thus requiring a smaller value of \( \epsilon \) than the other components.

Assuming that the direction \( d^k \) captures the relative scaling of the optimization variables, it may be appropriate to terminate computation when the norm of the direction \( d^k \) becomes sufficiently small, that is,

\[\|d^k\| \leq \epsilon.
\]

Still the appropriate value of \( \epsilon \) may not be easy to guess, and it may be necessary to experiment prior to settling on a reasonable termination criterion for a given problem. Sometimes, other problem-dependent criteria are used, in addition to or in place of \( \|\nabla f(x^k)\| \leq \epsilon \) and \( \|d^k\| \leq \epsilon \).

When \( \nabla^2 f(x) \) is positive definite, the condition \( \|\nabla f(x^k)\| \leq \epsilon \) yields bounds on the distance from local minimum. In particular, if \( x^* \) is a local minimum of \( f \) and there exists \( m > 0 \) such that for all \( z \) in a sphere \( S \) centered at \( x^* \) we have

\[m\|z\|^2 \leq z^T \nabla^2 f(x) z, \quad \forall z \in \mathbb{R}^n,
\]

then every \( x \in S \) satisfying \( \|\nabla f(x)\| \leq \epsilon \) also satisfies

\[\|x - x^*\| \leq \frac{\epsilon}{m}, \quad f(x) - f(x^*) \leq \frac{\epsilon^2}{m},
\]

(see Exercise 1.2.10).

In the absence of positive definiteness conditions on \( \nabla^2 f(x) \), it may be very difficult to infer the proximity of the current iterate to the optimal solution set by just using the gradient norm. We will return to this point when we will discuss singular local minima in the next section.

**Spacer Steps**

Often, optimization problems are solved with complex descent algorithms in which the rule used to determine the next point may depend on several previous points or on the iteration index \( k \). Some of the conjugate direction algorithms discussed in Section 1.6 are of this type. Other algorithms
Gradient Methods with Random and Nonrandom Errors*

Frequently in optimization problems the gradient $\nabla f(x^k)$ is not computed exactly. Instead, one has available

$$g^k = \nabla f(x^k) + e^k,$$

where $e^k$ is an uncontrollable error vector. There are several potential sources of error: roundoff error, and discretization error due to finite difference approximations to the gradient are two possibilities, but there are others that will be discussed in more detail in Section 1.5. Let us for concreteness focus on the steepest descent method with errors,

$$x^{k+1} = x^k - \alpha^k g^k,$$

and let us consider several qualitatively different cases:

(a) $e^k$ is small relative to the gradient, that is,

$$\|e^k\| < \|\nabla f(x^k)\|, \quad \forall k.$$

Then, assuming $\nabla f(x^k) \neq 0$, $-g^k$ is a direction of cost improvement, that is, $\nabla f(x^k) g^k > 0$. This is illustrated in Fig. 1.2.9, and is verified by the calculation

$$\nabla f(x^k) g^k = \|\nabla f(x^k)\|^2 + \nabla f(x^k) e^k$$

$$\geq \|\nabla f(x^k)\|^2 - \|\nabla f(x^k)\| \|e^k\|$$

$$= \|\nabla f(x^k)\|^2 (\|\nabla f(x^k)\| - \|e^k\|)$$

$$> 0.$$  \hspace{1cm} (1.14)

In this case convergence results that are analogous to Props. 1.2.3 and 1.2.4 can be shown.

(b) $\{e^k\}$ is bounded, that is,

$$\|e^k\| \leq \delta, \quad \forall k,$$

where $\delta$ is some scalar. Then by the preceding calculation (1.14), the method operates like a descent method within the region

$$\{x | \|\nabla f(x)\| > \delta\}.$$

In the complementary region where $\|\nabla f(x)\| \leq \delta$, the method can behave quite unpredictably. For example, if the errors $e^k$ are constant, say $e^k \equiv e$, then since $g^k = \nabla f(x^k) + e$, the method will essentially be trying to minimize $f(x) + e^T x$ and will typically converge to a point $\bar{x}$ with $\nabla f(\bar{x}) = -e$. If the errors $e^k$ vary substantially, the method will tend to oscillate within the region where $\|\nabla f(x)\| \leq \delta$ (see Exercise 1.2.17 and also Exercise 1.3.4 in the next section). The precise behavior will depend on the precise nature of the errors, and on whether a constant or a diminishing stepsize is used (see also the following cases).

(c) $\{e^k\}$ is proportional to the stepsize, that is,

$$\|e^k\| \leq \alpha^k q, \quad \forall k,$$

where $q$ is some scalar. If the stepsize is constant, we come under case (b), while if the stepsize is diminishing, the behavior described in case (b) applies, but with $\delta \rightarrow 0$, so the method will tend to converge to a stationary point of $f$. Important situations where the condition $\|e^k\| \leq \alpha^k q$ holds will be encountered in Section 1.5 (see Prop. 1.5.1 of that section). A more general condition under which similar behavior occurs is

$$\|e^k\| \leq \alpha^k (q + p \|\nabla f(x^k)\|). \quad \forall k.$$

where $q$ and $p$ are some scalars. Generally, under this condition and with a diminishing stepsize, the convergence behavior is similar to
the case where there are no errors (see the following Prop. 1.2.4 and Exercise 1.2.21).

(d) \{e^k\} are independent zero mean random vectors with finite variance. An important special case where such errors arise is when \( f \) is of the form
\[
    f(x) = E_w \{F(x, w)\},
\]
where \( F : \mathbb{R}^{n+m} \to \mathbb{R} \) is some function, \( w \) is a random vector in \( \mathbb{R}^m \), and \( E_w \{ \} \) denotes expected value. Under very mild assumptions it can be shown that if \( F \) is continuously differentiable, the same is true of \( f \) and furthermore,
\[
    \nabla f(x) = E_w \{ \nabla_x F(x, w) \}.
\]

Often an approximation \( g^k \) to \( \nabla f(x^k) \) is computed by simulation or by using a limited number of samples of \( \nabla F(x, w) \), with potentially substantial error resulting. In an extreme case, we have
\[
    g^k = \nabla_x F(x^k, w^k),
\]
where \( w^k \) is a single sample value corresponding to \( x^k \). Then the error
\[
    e^k = \nabla_x F(x^k, w^k) - \nabla f(x^k) = \nabla_x F(x^k, w^k) - E_w \{ \nabla_x F(x^k, w) \}
\]
need not diminish with \( ||\nabla f(x^k)|| \), but has zero mean, and under appropriate conditions, its effects are “averaged out.” What is happening here is that the descent condition \( \nabla f(x^k) g^k > 0 \) holds on the average at nonstationary points \( x^k \). It is still possible that for some sample values of \( e^k \), the direction \( g^k \) is “bad”, but with a diminishing stepsize, the occasional use of a bad direction cannot deteriorate the cost enough for the method to oscillate, given that on the average the method uses “good” directions. The detailed analysis of gradient methods with random errors is beyond the scope of this text. We refer to the literature (see e.g. [BeTh89], [KuC78], [KaY97], [LPW92], [PH96], [PoT73], [PoB78], [TBA86], [Be96]). Let us mention one representative convergence result, due to Bertsekas and Tsi- skidis [BeTo01], which parallels the following Prop. 1.2.4 that deals with a gradient method without errors: if in the iteration
\[
    x^{k+1} = x^k - \alpha^k \nabla_x F(x^k, w^k)
\]
the random variables \( w^0, w^1, \ldots \) are independent and have finite variance, the stepsize is diminishing and satisfies
\[
    \alpha^k \to 0, \quad \sum_{k=0}^{\infty} \alpha^k = \infty, \quad \sum_{k=0}^{\infty} (\alpha^k)^2 < \infty,
\]
and some additional technical conditions (of the type given in Prop. 1.2.4) hold, then with probability one, we either have \( f(x^k) \to -\infty \) or else \( \nabla f(x^k) \to 0 \).

Sec. 1.2 Gradient Methods Convergence

The Role of Convergence Analysis

The following subsection gives a number of mathematical propositions relating to the convergence properties of gradient methods. The meaning of these propositions is usually quite intuitive but their statement often requires complicated mathematical assumptions. Furthermore, their proof often involves tedious \( \epsilon-\delta \) arguments, so at first sight students may wonder whether “we really have to go through all this.”

When Euclid was faced with a similar question from king Ptolemy of Alexandria, he replied that “there is no royal road to geometry.” In our case, however, the answer is not so simple because we are not dealing with a pure subject such as geometry that may be developed without regard for its practical application. In the eyes of most people, the value of an analysis or algorithm in nonlinear programming is judged primarily by its practical impact in solving various types of problems. It is therefore important to give some thought to the interface between convergence analysis and its practical application. To this end it is useful to consider two extreme viewpoints; most workers in the field find themselves somewhere between the two.

In the first viewpoint, convergence analysis is considered primarily a mathematical subject. The properties of an algorithm are quantified to the extent possible through mathematical statements. General and broadly applicable assertions, and simple and elegant proofs are at a premium here. The rationale is that simple statements and proofs are more readily understood, and general statements apply not only to the problems at hand but also to other problems that are likely to appear in the future. On the negative side, one may remark that simplicity is not always compatible with relevance, and broad applicability is often achieved through assumptions that are hard to verify or appreciate.

The second viewpoint largely rejects the role of mathematical analysis. The rationale here is that the validity and the properties of an algorithm for a given class of problems must be verified through practical experimentation anyway, so if an algorithm looks promising on intuitive grounds, why bother with a convergence analysis. Furthermore, there are a number of important practical questions that are hard to address analytically, such as roundoff error, multiple local minima, and a variety of finite termination and approximation issues. The main criticism of this viewpoint is that mathematical analysis often reveals (and explains) fundamental flaws of algorithms that experimentation may miss. These flaws often point the way to better algorithms or modified algorithms that are tailored to the type of practical problem at hand. Similarly, analysis may be more effective than experimentation in delineating the types of problems for which particular algorithms are well-suited.

Our own mathematical approach is tempered by practical concerns, but we note that the balance between theory and practice in nonlinear
programming is particularly delicate, subjective, and problem dependent. Aside from the fact that the mathematical proofs themselves often provide valuable insight into algorithms, here are some of our reasons for insisting on a rigorous convergence analysis:

(a) We want to delineate the range of applicability of various methods. In particular, we want to know for what type of cost function (once or twice differentiable, convex or nonconvex, with singular or nonsingular minima) each algorithm is best suited. If the cost function violates the assumptions under which a given algorithm can be proved to converge, it is reasonable to suspect that the algorithm is unsuitable for this cost function.

(b) We want to understand the qualitative behavior of various methods. For example, we want to know whether convergence of the method depends on the availability of a good starting point, whether the iterates $x^k$ or just the function values $f(x^k)$ are guaranteed to converge, etc. This information may supplement and/or guide the computational experimentation.

(c) We want to provide guidelines for choosing a few algorithms for further experimentation out of the often bewildering array of candidate algorithms that are applicable for the solution of a given type of problem. One of the principal means for this is the rate of convergence analysis to be given in Section 1.3. Note here that while an algorithm may provably converge, in practice it may be entirely inappropriate for a given problem because it converges very slowly. Experience has shown that without a good understanding of the rate of convergence properties of algorithms it may be difficult to exclude bad candidates from consideration without costly experimentation.

At the same time one should be aware of some of the limitations of the mathematical results that we will provide. For example, some of the assumptions under which an algorithm will be proved convergent may be hard to verify for a given type of problem. Furthermore, our convergence rate analysis of Section 1.3 is largely asymptotic; that is, it applies near the eventual limit of the generated sequence. It is possible, that an algorithm has a good asymptotic rate of convergence but it works poorly in practice for a given type of problem because it is very slow in its initial phase.

There is still another viewpoint, which is worth addressing because it is often adopted by the casual user of nonlinear programming algorithms. This user is interested in a particular application of nonlinear programming in his/her special field, and is counting on an existing code or package to solve the problem (several such packages are commercially or publicly available). Since the package will do most of the work, the user may hope that a superficial acquaintance with the properties of the algorithms underlying the package will suffice. This hope is sometimes realized but unfortunately

in many cases it is not. There are a number of reasons for this. First, there are many packages implementing a lot of different methods, and to choose the right package, one needs to have insight into the suitability of different methods for the special features of the application at hand. Second, to use a package one must often know how to suitably formulate the problem, how to set various parameters (e.g., termination criteria, stepsize parameters, etc.), and how to interpret the results of the computation (particularly when things don’t work out as hoped initially, which is often the case). For this, one needs considerable insight into the inner workings of the algorithm underlying the package. Finally, for a challenging practical optimization problem (e.g., one of large dimension), it may be essential to exploit its special structure, and packages often do not have this capability. As a result the user may have to modify the package or write an altogether new code that is tailored to the application at hand. Both of these require an intimate understanding of the convergence properties and other characteristics of the relevant nonlinear programming algorithms.

1.2.2 Convergence Results

We now provide an analysis of the convergence behavior of gradient methods. The following proposition is the main convergence result.

**Proposition 1.2.1:** (Stationarity of Limit Points for Gradient Methods) Let \( \{x^k\} \) be a sequence generated by a gradient method \( x^{k+1} = x^k + \alpha^k d^k \), and assume that \( \{d^k\} \) is gradient related [cf. Eq. (1.13)] and \( \alpha^k \) is chosen by the minimization rule, or the limited minimization rule, or the Armijo rule. Then every limit point of \( \{x^k\} \) is a stationary point.

**Proof:** Consider the Armijo rule, and to arrive at a contradiction, assume that \( \bar{x} \) is a limit point of \( \{x^k\} \) with \( \nabla f(\bar{x}) \neq 0 \). Note that since \( \{f(x^k)\} \) is monotonically nonincreasing, \( \{f(x^k)\} \) either converges to a finite value or diverges to \(-\infty\). Since \( f \) is continuous, \( f(\bar{x}) \) is a limit point of \( \{f(x^k)\} \), so it follows that the entire sequence \( \{f(x^k)\} \) converges to \( f(\bar{x}) \). Hence,

\[
f(x^k) - f(x^{k+1}) = 0.
\]

By the definition of the Armijo rule, we have

\[
f(x^k) - f(x^{k+1}) \geq -\sigma \alpha^k \nabla f(x^k)^T d^k.
\]

Hence, \( \alpha^k \nabla f(x^k)^T d^k \to 0 \). Let \( \{d^k\}_c \) be a subsequence converging to \( \bar{d} \). Since \( \{d^k\} \) is gradient related, we have

\[
\limsup_{k\to\infty} \nabla f(x^k)^T d^k < 0,
\]
and therefore
\[ \{\alpha^k\}_k \rightarrow 0. \]

Hence, by the definition of the Armijo rule, we must have for some index \( k \geq 0 \)
\[ f(x^k) - f(x^k + (\alpha^k/\beta)d^k) < -\sigma(\alpha^k/\beta)\nabla f(x^k)'d^k, \quad \forall k \in \mathbb{K}, \ k \geq \bar{k}, \tag{1.17} \]
that is, the initial stepsize \( s \) will be reduced at least once for all \( k \in \mathbb{K}, \ k \geq \bar{k} \). Since \( \{d^k\}_k \) is gradient related, \( \{d^k\}_k \) is bounded, and it follows that there exists a subsequence \( \{\bar{d}^k\}_k \) of \( \{d^k\}_k \) such that
\[ \{\bar{d}^k\}_k \rightarrow \bar{d}, \]
where \( \bar{d} \) is some vector. From Eq. (1.17), we have
\[ \frac{f(x^k) - f(x^k + \bar{\alpha}^k d^k)}{\bar{\alpha}^k} < -\sigma \nabla f(x^k)'d^k, \quad \forall k \in \mathbb{K}, \ k \geq \bar{k}, \tag{1.18} \]
where \( \bar{\alpha}^k = \alpha^k/\beta \). By using the mean value theorem, this relation is written as
\[ -\nabla f(x^k)'\bar{d} < -\sigma \nabla f(x^k)'d^k, \quad \forall k \in \mathbb{K}, \ k \geq \bar{k}, \]
where \( \bar{\alpha}^k \) is a scalar in the interval \([0, \bar{\alpha}^k]\). Taking limits in the above equation we obtain
\[ -\nabla f(x)'\bar{d} \leq -\sigma \nabla f(x)'d \]
or
\[ 0 \leq (1 - \sigma) \nabla f(x)'\bar{d}. \tag{1.19} \]
Since \( \sigma < 1 \), it follows that
\[ 0 \leq \nabla f(x)'d, \tag{1.19} \]
which contradicts the assumption that \( \{d^k\} \) is gradient related. This proves the result for the Armijo rule.

Consider now the minimization rule, and let \( \{x^k\}_k \) converge to \( \bar{x} \) with \( \nabla f(\bar{x}) \neq 0 \). Again we have that \( \{f(x^k)\} \) decreases monotonically to \( f(\bar{x}) \). Let \( \bar{x}^{k+1} \) be the point generated from \( x^k \) via the Armijo rule, and let \( \bar{\alpha}^k \) be the corresponding stepsize. We have
\[ f(x^k) - f(x^{k+1}) \geq f(x^k) - f(\bar{x}^{k+1}) \geq -\bar{\alpha}^k \nabla f(x^k)'d^k. \]

By repeating the arguments of the earlier proof following Eq. (1.16), replacing \( \alpha^k \) by \( \bar{\alpha}^k \), we can obtain a contradiction. In particular, we have
\[ \{\bar{\alpha}^k\}_k \rightarrow 0, \]
and by the definition of the Armijo rule, we have for some index \( k \geq 0 \)
\[ f(x^k) - f(x^k + (\bar{\alpha}^k/\beta)d^k) < -\sigma(\bar{\alpha}^k/\beta)\nabla f(x^k)'d^k, \quad \forall k \in \mathbb{K}, \ k \geq \bar{k}, \tag{1.17} \]
[cf. Eq. (1.17)]. Proceeding as earlier, we obtain Eqs. (1.18) and (1.19) (with \( \alpha^k = \bar{\alpha}^k/\beta \)), and a contradiction of Eq. (1.19).

The line of argument just used establishes that any stepsize rule that gives a larger reduction in cost at each iteration than the Armijo rule inherits its convergence properties. This also proves the proposition for the limited minimization rule. Q.E.D.

The following proposition can be shown similar to Prop. 1.2.1. Its proof is left for the reader.

**Proposition 1.2.2:** The conclusions of Prop. 1.2.1 hold if \( \{d^k\} \) is gradient related and \( \alpha^k \) is chosen by the Goldstein rule.

The next proposition establishes, among other things, convergence for the case of a constant stepsize. The idea is that if the rate of growth of the gradient of \( f \) is limited (i.e., the curvature of \( f \) is limited), then one can construct a quadratic function that majorizes \( f \); see Fig. 1.2.10. Given \( x^k \) and \( d^k \), an appropriate constant stepsize \( \alpha^k \) can then be obtained within an interval around the scalar \( \alpha^k \) that minimizes this quadratic function along the direction \( d^k \).

**Proposition 1.2.3:** (Convergence for a Constant Stepsize) Let \( \{x^k\} \) be a sequence generated by a gradient method \( x^{k+1} = x^k + \alpha^k d^k \), where \( \{d^k\} \) is gradient related. Assume that for some constant \( L > 0 \), we have
\[ \|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathbb{R}^n, \tag{1.20} \]
and that for all \( k \) we have \( \alpha^k \neq 0 \) and
\[ \epsilon \leq \alpha^k \leq (2 - \epsilon)\alpha^k, \tag{1.21} \]
where
\[ \frac{\alpha^k}{\|d^k\|^2} = \frac{\|\nabla f(x^k)\|d^k}{L\|d^k\|^2}. \]
and \( \epsilon \) is a fixed positive scalar. Then every limit point of \( \{x^k\} \) is a stationary point of \( f \).
Sec. 1.2 Gradient Methods - Convergence

Proof: By using the descent lemma (Prop. A.24 of Appendix A), we obtain

\[ f(x^k + \alpha^k d^k) - f(x^k) \leq \alpha^k \nabla f(x^k)' d^k + \frac{1}{2} \alpha^k L \|d^k\|^2 \]

\[ = \alpha^k \left( \frac{1}{2} \alpha^k L \|d^k\|^2 - \|\nabla f(x^k)' d^k\| \right). \tag{1.24} \]

The right-hand side of Eq. (1.21) yields

\[ \frac{1}{2} \alpha^k L \|d^k\|^2 - \|\nabla f(x^k)' d^k\| \leq -\frac{1}{2} \epsilon (\|\nabla f(x^k)' d^k\|). \]

Using this relation together with the condition \( \alpha^k \geq \epsilon \) in the inequality (1.24), we obtain the following bound on the cost improvement obtained at iteration \( k \):

\[ f(x^k) - f(x^k + \alpha^k d^k) \geq \frac{\epsilon}{2} \|\nabla f(x^k)' d^k\|. \]

Now if a subsequence \( \{x^k\}_k \) converges to a nonstationary point \( \bar{x} \), we must have, as in the proof of Prop. 1.2.1, \( f(x^k) - f(x^{k+1}) \rightarrow 0 \), and the preceding relation implies that \( \|\nabla f(x^k)' d^k\| \rightarrow 0 \). This contradicts the assumption that \( \{d^k\} \) is gradient related. Hence, every limit point of \( \{x^k\} \) is stationary. Q.E.D.

A condition of the form

\[ \|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|, \quad \forall x, y \in \mathbb{R}^n, \]

(cf. Eq. (1.20)) is called a Lipschitz continuity condition on \( \nabla f \). Exercise 1.2.3 provides an example showing that this condition is essential for the validity of Prop. 1.2.3. The Lipschitz condition requires roughly that the "curvature" of \( f \) is no more than \( L \) in all directions. In particular, it is possible to show that this condition is satisfied for some \( L > 0 \), if \( f \) is twice differentiable and the Hessian \( \nabla^2 f \) is bounded over \( \mathbb{R}^n \). Unfortunately, however, it is generally difficult to obtain an estimate of \( L \), so in most cases the range of step sizes that guarantee convergence [cf. Eq. (1.21) or (1.23)] is unknown, and experimentation may be necessary to obtain an appropriate range of step sizes (a method that attempts to determine an appropriate value of step size automatically is given in Exercise 1.2.20). Even worse, many types of cost function \( f \), though twice differentiable have Hessian \( \nabla^2 f \) that is unbounded over \( \mathbb{R}^n \) [this is so for any function \( f(x) \) that grows faster than a quadratic as \( x \to \infty \), such as \( f(x) = \|x\|^2 \)].

Fortunately, the Lipschitz condition can be significantly weakened, as shown in Exercise 1.2.5. In particular, it is sufficient that it holds for all \( x, y \) in the set \( \{x \mid f(x) < f(x^0)\} \), in which case, however, the range of step sizes that guarantee convergence depends on the starting point \( x^0 \).

The Lipschitz continuity condition also essentially guarantees convergence for a diminishing step size, as shown by the following proposition.
Proposition 1.2.4: (Convergence for a Diminishing Stepsize)

Let \( \{x^k\} \) be a sequence generated by a gradient method \( x^{k+1} = x^k + \alpha^k d^k \). Assume that for some constant \( L > 0 \), we have
\[
\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|, \quad \forall x, y \in \mathbb{R}^n, \quad (1.25)
\]
and that there exist positive scalars \( c_1, c_2 \) such that for all \( k \) we have
\[
c_1 \|\nabla f(x^k)\|^2 \leq -\nabla f(x^k)^T d^k, \quad \|d^k\|^2 \leq c_2 \|\nabla f(x^k)\|^2. \quad (1.26)
\]
Suppose also that
\[
\alpha^k \to 0, \quad \sum_{k=0}^{\infty} \alpha^k = \infty.
\]
Then either \( f(x^k) \to -\infty \) or else \( \{f(x^k)\} \) converges to a finite value and \( \nabla f(x^k) \to 0 \). Furthermore, every limit point of \( \{x^k\} \) is a stationary point of \( f \).

Proof: Combining Eqs. (1.24) and (1.26), we have
\[
f(x^{k+1}) \leq f(x^k) + \alpha^k \left( \frac{1}{2} \alpha^k L \|d^k\|^2 - \|\nabla f(x^k)^T d^k\| \right) \\
\leq f(x^k) - \alpha^k (c_1 - \frac{1}{2} \alpha^k c_2 L) \|\nabla f(x^k)\|^2.
\]
(1.27)

Since the linear term in \( \alpha^k \) dominates the quadratic term in \( \alpha^k \) for sufficiently small \( \alpha^k \), and \( \alpha^k \to 0 \), we have for some positive constant \( c \) and all \( k \) greater than some index \( \tilde{k} \),
\[
f(x^{k+1}) \leq f(x^k) - \alpha^k c \|\nabla f(x^k)\|^2.
\]
From this relation, we see that for \( k \geq \tilde{k} \), \( \{f(x^k)\} \) is monotonically decreasing, so either \( f(x^k) \to -\infty \) or \( \{f(x^k)\} \) converges to a finite value. In the latter case, by adding Eq. (1.27) over all \( k \geq \tilde{k} \), we obtain
\[
c \sum_{k=\tilde{k}}^{\infty} \alpha^k \|\nabla f(x^k)\|^2 \leq f(x^\tilde{k}) - \lim_{k \to \infty} f(x^k) < \infty.
\]
(1.28)

Sec. 1.2 Gradient Methods - Convergence

Let \( \{m_j\} \) and \( \{n_j\} \) be sequences of indices such that
\[
m_j < n_j < m_{j+1}.
\]
(1.29)
\[
\frac{\epsilon}{3} \leq f(x^k) \quad \text{for} \quad m_j \leq k < n_j.
\]
(1.30)
\[
\|f(x^k)\| \leq \frac{\epsilon}{3} \quad \text{for} \quad n_j \leq k < m_{j+1}.
\]

Let also \( \tilde{j} \) be sufficiently large so that
\[
\sum_{k=m_j}^{\infty} \alpha^k \|\nabla f(x^k)\|^2 \leq \frac{\epsilon^2}{9L\sqrt{c_2}}. \quad (1.31)
\]

For any \( j \geq \tilde{j} \) and any \( m \) with \( m_j \leq m \leq n_j - 1 \), we have
\[
\|\nabla f(x^{m_j}) - \nabla f(x^m)\| \leq \sum_{k=m_j}^{m-1} \|\nabla f(x^{k+1}) - \nabla f(x^k)\|
\]
\[
\leq L \sum_{k=m_j}^{n_j-1} \|x^{k+1} - x^k\|
\]
\[
= L \sum_{k=m_j}^{n_j-1} \alpha^k \|d^k\|
\]
\[
\leq L \sqrt{c_2} \sum_{k=m_j}^{n_j-1} \alpha^k \|\nabla f(x^k)\|
\]
\[
\leq \frac{3L \sqrt{c_2}}{\epsilon} \sum_{k=m_j}^{n_j-1} \alpha^k \|\nabla f(x^k)\|^2
\]
\[
\leq \frac{3L \sqrt{c_2}}{\epsilon} \frac{\epsilon^2}{9L \sqrt{c_2}}
\]
\[
= \frac{\epsilon}{3},
\]
where the last two inequalities follow using Eqs. (1.29) and (1.31). Thus
\[
\|\nabla f(x^m)\| \leq \|\nabla f(x^{n_j})\| + \frac{\epsilon}{3} \leq \frac{2\epsilon}{3}, \quad \forall j \geq \tilde{j}, m_j \leq m \leq n_j - 1.
\]

Thus, using also Eq. (1.30), we have for all \( m \geq m_j \)
\[
\|\nabla f(x^m)\| \leq \frac{2\epsilon}{3}.
\]
This contradicts Eq. (1.28), implying that $\lim_{k\to\infty} \nabla f(x^k) = 0$.

Finally, if $\bar{x}$ is a limit point of $x^k$, then $f(x^k)$ converges to the finite value $f(\bar{x})$. Thus we have $\nabla f(x^k) \rightarrow 0$, implying that $\nabla f(\bar{x}) = 0$. Q.E.D.

Under the assumptions of the preceding proposition, descent is not guaranteed in the initial iterations. However, if the step sizes are all sufficiently small [e.g., they satisfy the right-hand side inequality of Eq. (1.23)], descent is guaranteed at all iterations. In this case, it is sufficient that the Lipschitz condition $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ holds for all $x, y$ in the set $\{z \mid f(x) \leq f(z)\}$ (see Exercise 1.2.5); otherwise the Lipschitz condition must hold over a set larger than $\{z \mid f(z) \leq f(x^0)\}$ to guarantee convergence (see Exercise 1.2.15).

The following proposition explains to some extent why sequences generated by gradient methods tend to practice to have unique limit points. It essentially states that local minima which are "isolated" tend to attract gradient methods; once the method gets close enough to such a minimum it remains close and converges to it.

**Proposition 1.2.5: (Capture Theorem)** Let $f$ be continuously differentiable and let $\{x^k\}$ be a sequence satisfying $f(x^{k+1}) \leq f(x^k)$ for all $k$ and generated by a gradient method $x^{k+1} = x^k + \alpha^k d^k$, which is convergent in the sense that every limit point of sequences that it generates is a stationary point of $f$. Assume that there exist scalars $s > 0$ and $c > 0$ such that for all $k$ there holds

$$\alpha^k \leq s, \quad \|d^k\| \leq c\|\nabla f(x^k)\|.$$

Let $x^*$ be a local minimum of $f$, which is the only stationary point of $f$ within some open set. Then there exists an open set $S$ containing $x^*$ such that if $x^k \in S$ for some $k \geq 0$, then $x^k \in S$ for all $k \geq k$ and $\{x^k\} \rightarrow x^*$. Furthermore, given any scalar $\bar{c} > 0$, the set $S$ can be chosen so that $\|x - x^*\| < \bar{c}$ for all $x \in S$.

**Note:** The conditions $f(x^{k+1}) \leq f(x^k)$ and $\alpha^k \leq s$ are satisfied for the Armijo rule and the limited minimization rule. They are also satisfied for a constant and a diminishing stepsize under conditions that guarantee descent at each iteration (see the proofs of Props. 1.2.3 and 1.2.4). The condition $\|d^k\| \leq c\|\nabla f(x^k)\|$ is satisfied if $d^k = -D^k \nabla f(x^k)$ with the eigenvalues of $D^k$ bounded from above.

**Proof:** Suppose that $\rho > 0$ is such that

$$f(x^*) < f(x), \quad \forall x \text{ with } \|x - x^*\| \leq \rho.$$

Define for $t \in [0, \rho]$

$$\phi(t) = \min_{(x^*) \leq \|z - x^*\| \leq \rho} f(z) - f(x^*),$$

and note that $\phi$ is a monotonically nondecreasing function of $t$, and that $\phi(t) > 0$ for all $t \in (0, \rho]$. Given any $e \in (0, \rho]$, let $r \in (0, e]$ be such that

$$\|x - x^*\| < r \Rightarrow \|x - x^*\| + se\|\nabla f(x)\| < e.$$

Consider the open set

$$S = \{x \mid \|x - x^*\| < e, f(x) < f(x^*) + \phi(e)\}.$$

We claim that if $x^k \in S$ for some $k$, then $x^{k+1} \in S$.

Indeed if $x^k \in S$, from the definition of $\phi$ and $S$ we have

$$\phi(\|x^k - x^*\|) \leq f(x^k) - f(x^*) < \phi(r).$$

Since $\phi$ is monotonically nondecreasing, the above relation implies that $\|x^k - x^*\| < r$, so that by Eq. (1.32),

$$\|x^k - x^*\| + se\|\nabla f(x^k)\| < e.$$

We also have by using the hypotheses $\alpha^k \leq s$ and $\|d^k\| \leq c\|\nabla f(x^k)\|$

$$\|x^{k+1} - x^*\| \leq \|x^k - x^*\| + \alpha^kd^k \leq \|x^k - x^*\| + se\|\nabla f(x^k)\|,$$

so from the last two relations it follows that $\|x^{k+1} - x^*\| < e$. Since $f(x^{k+1}) < f(x^k)$, we also obtain $f(x^{k+1}) - f(x^*) < \phi(r)$, so we conclude that $x^{k+1} \in S$.

By using induction it follows that if $x^k \in S$ for some $k$, we have $x^k \in S$ for all $k \geq k$. Let $S$ be the closure of $S$. Since $S$ is compact, the sequence $\{x^k\}$ will have at least one limit point, which by assumption must be a stationary point of $f$. Now the only stationary point of $f$ within $S$ is the point $x^*$ (since we have $\|x - x^*\| \leq \rho$ for all $x \in S$). Hence $x^k \rightarrow x^*$. Finally given any $\bar{e} > 0$, we can choose $e \leq \bar{e}$ in which case we have $\|x - x^*\| < \bar{e}$ for all $x \in S$. Q.E.D.

Finally, we state a result that deals with the convergence of algorithms involving a combination of different methods. It shows that for convergence it is enough to insert, perhaps irregularly but infinitely often, an iteration of a convergent gradient algorithm, provided that the other iterations do not degrade the value of the cost function. The proof is similar to the one of Prop. 1.2.1, and is left for the reader.
Proposition 1.2.8: (Convergence for Spacer Steps) Consider a sequence \( \{x^k\} \) such that
\[
f(x^{k+1}) \leq f(x^k), \quad k = 0, 1, \ldots
\]
Assume that there exists an infinite set \( \mathcal{K} \) of integers for which
\[
x^{k+1} = x^k + \alpha^k d^k, \quad \forall k \in \mathcal{K},
\]
where \( \{d^k\}_{\mathcal{K}} \) is gradient related and \( \alpha^k \) is chosen by the minimization rule, or the limited minimization rule, or the Armijo rule. Then every limit point of the subsequence \( \{x^k\}_{\mathcal{K}} \) is a stationary point.