Chapter 13
Penalty and Barrier Methods

Penalty and barrier methods are procedures for approximating constrained optimization problems by unconstrained problems. The approximation is accomplished in the case of penalty methods by adding to the objective function a term that prescribes a high cost for violation of the constraints, and in the case of barrier methods by adding a term that favors points interior to the feasible region over those near the boundary. Associated with these methods is a parameter c or μ that determines the severity of the penalty or barrier and consequently the degree to which the unconstrained problem approximates the original constrained problem. For a problem with n variables and m constraints, penalty and barrier methods work directly in the n-dimensional space of variables, as compared to primal methods that work in $(n - m)$-dimensional space.

There are two fundamental issues associated with the methods of this chapter. The first has to do with how well the unconstrained problem approximates the constrained one. This is essential in examining whether, as the parameter c is increased toward infinity, the solution of the unconstrained problem converges to a solution of the constrained problem. The other issue, most important from a practical viewpoint, is the question of how to solve a given unconstrained problem when its objective function contains a penalty or barrier term. It turns out that as c is increased to yield a good approximating problem, the corresponding structure of the resulting unconstrained problem becomes increasingly unfavorable thereby slowing the convergence rate of many algorithms that might be applied. (Exact penalty functions also have a very unfavorable structure.) It is necessary, then, to devise acceleration procedures that circumvent this slow convergence phenomenon.

Penalty and barrier methods are of great interest to both the practitioner and the theorist. To the practitioner they offer a simple straightforward method for handling constrained problems that can be implemented without sophisticated computer programming and that possess much the same degree of generality as primal methods. The theorist, striving to make this approach practical by overcoming its inherently slow convergence, finds it appropriate to bring into play nearly all aspects

D.G. Luenberger, Y. Ye, *Linear and Nonlinear Programming*, International Series in Operations Research & Management Science 228,
DOI 10.1007/978-3-319-18842-3_13

of optimization theory; including Lagrange multipliers, necessary conditions, and many of the algorithms discussed earlier in this book. The canonical rate of convergence associated with the original constrained problem again asserts its fundamental role by essentially determining the natural *accelerated* rate of convergence for unconstrained penalty or barrier problems.

13.1 Penalty Methods

Consider the problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in S, \end{aligned} \tag{13.1}$$

where f is a continuous function on E^n and S is a constraint set in E^n. In most applications S is defined implicitly by a number of functional constraints, but in this section the more general description in (13.1) can be handled. The idea of a penalty function method is to replace problem (13.1) by an unconstrained problem of the form

$$\text{minimize} \quad f(\mathbf{x}) + cP(\mathbf{x}), \tag{13.2}$$

where c is a positive constant and P is a function on E^n satisfying: (i) P is continuous, (ii) $P(\mathbf{x}) \geqslant 0$ for all $\mathbf{x} \in E^n$, and (iii) $P(\mathbf{x}) = 0$ if and only if $\mathbf{x} \in S$.

Example 1. Suppose S is defined by a number of inequality constraints:

$$S = \{\mathbf{x} : g_i(\mathbf{x}) \leqslant 0, \quad i = 1, 2, \ldots, p\}.$$

A very useful penalty function in this case is

$$P(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^{P} (\max[0, \; g_i(\mathbf{x})])^2.$$

The function $cP(\mathbf{x})$ is illustrated in Fig. 13.1 for the one-dimensional case with $g_1(x) = x - b, \; g_2(x) = a - x$.

For large c it is clear that the minimum point of problem (13.2) will be in a region where P is small. Thus, for increasing c it is expected that the corresponding solution points will approach the feasible region S and, subject to being close, will minimize f. Ideally then, as $c \to \infty$ the solution point of the penalty problem will converge to a solution of the constrained problem.

The Method

The procedure for solving problem (13.1) by the penalty function method is this: Let $\{c_k\}$, $k = 1, 2, \ldots$, be a sequence tending to infinity such that for each k, $c_k \geqslant 0$, $c_{k+1} > c_k$. Define the function

$$q(c, \; \mathbf{x}) = f(\mathbf{x}) + cP(\mathbf{x}). \tag{13.3}$$

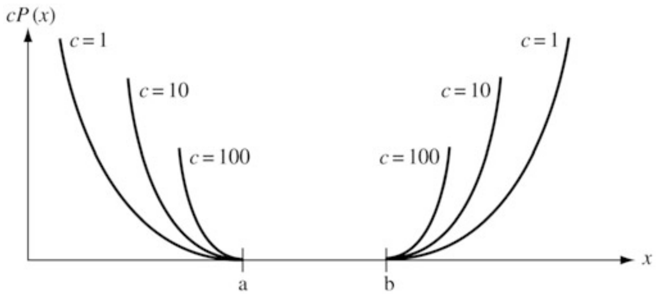

Fig. 13.1 Plot of $cP(x)$

For each k solve the problem

$$\text{minimize} \quad q(c_k, \mathbf{x}), \tag{13.4}$$

obtaining a solution point $\mathbf{x}_k$.

We assume here that, for each k, problem (13.4) has a solution. This will be true, for example, if $q(c, \mathbf{x})$ increases unboundedly as $|\mathbf{x}| \to \infty$. (Also see Exercise 2 to see that it is not necessary to obtain the minimum precisely.)

Convergence

The following lemma gives a set of inequalities that follow directly from the definition of $\mathbf{x}_k$ and the inequality $c_{k+1} > c_k$.

Lemma 1.

$$q(c_k, \mathbf{x}_k) \leqslant q(c_{k+1}, \mathbf{x}_{k+1}) \tag{13.5}$$
$$P(\mathbf{x}_k) \geqslant P(\mathbf{x}_{k+1}) \tag{13.6}$$
$$f(\mathbf{x}_k) \leqslant f(\mathbf{x}_{k+1}). \tag{13.7}$$

Proof.

$$\begin{aligned} q(c_{k+1}, \mathbf{x}_{k+1}) &= f(\mathbf{x}_{k+1}) + c_{k+1}P(\mathbf{x}_{k+1}) \geqslant f(\mathbf{x}_{k+1}) + c_kP(\mathbf{x}_{k+1}) \\ &\geqslant f(\mathbf{x}_k) + c_kP(\mathbf{x}_k) = q(c_k, \mathbf{x}_k), \end{aligned}$$

which proves (13.5).

We also have

$$f(\mathbf{x}_k) + c_kP(\mathbf{x}_k) \leqslant f(\mathbf{x}_{k+1}) + c_kP(\mathbf{x}_{k+1}) \tag{13.8}$$
$$f(\mathbf{x}_{k+1}) + c_{k+1}P(\mathbf{x}_{k+1}) \leqslant f(\mathbf{x}_k) + c_{k+1}P(\mathbf{x}_k). \tag{13.9}$$

Adding (13.8) and (13.9) yields

$$(c_{k+1} - c_k)P(\mathbf{x}_{k+1}) \leqslant (c_{k+1} - c_k)P(\mathbf{x}_k),$$

which proves (13.6).

Also

$$f(\mathbf{x}_{k+1}) + c_k P(\mathbf{x}_{k+1}) \geqslant f(\mathbf{x}_k) + c_k P(\mathbf{x}_k),$$

and hence using (13.6) we obtain (13.7). ∎

Lemma 2. *Let* $\mathbf{x}^*$ *be a solution to problem (13.1). Then for each k*

$$f(\mathbf{x}^*) \geqslant q(c_k, \mathbf{x}_k) \geqslant f(\mathbf{x}_k).$$

Proof.

$$f(\mathbf{x}^*) = f(\mathbf{x}^*) + c_k P(\mathbf{x}^*) \geqslant f(\mathbf{x}_k) + c_k P(\mathbf{x}_k) \geqslant f(\mathbf{x}_k). \blacksquare$$

Global convergence of the penalty method, or more precisely verification that any limit point of the sequence is a solution, follows easily from the two lemmas above.

Theorem. *Let* $\{\mathbf{x}_k\}$ *be a sequence generated by the penalty method. Then, any limit point of the sequence is a solution to (13.1).*

Proof. Suppose the subsequence $\{\mathbf{x}_k\}$, $k \in \mathcal{K}$ is a convergent subsequence of $\{\mathbf{x}_k\}$ having limit $\bar{\mathbf{x}}$. Then by the continuity of f, we have

$$\operatorname*{limit}_{k\in\mathcal{K}} f(\mathbf{x}_k) = f(\bar{\mathbf{x}}). \tag{13.10}$$

Let f^* be the optimal value associated with problem (13.1). Then according to Lemmas 1 and 2, the sequence of values $q(c_k, \mathbf{x}_k)$ is nondecreasing and bounded above by f^*. Thus

$$\operatorname*{limit}_{k\in\mathcal{K}} q(c_k, \mathbf{x}_k) = q^* \leqslant f^*. \tag{13.11}$$

Subtracting (13.10) from (13.11) yields

$$\operatorname*{limit}_{k\in\mathcal{K}} c_k P(\mathbf{x}_k) = q^* - f(\bar{\mathbf{x}}). \tag{13.12}$$

Since $P(\mathbf{x}_k) \geqslant 0$ and $c_k \to \infty$, (13.12) implies

$$\operatorname*{limit}_{k\in\mathcal{K}} P(\mathbf{x}_k) = 0.$$

Using the continuity of P, this implies $P(\bar{\mathbf{x}}) = 0$. We therefore have shown that the limit point $\bar{\mathbf{x}}$ is feasible for (13.1).

To show that $\bar{\mathbf{x}}$ is optimal we note that from Lemma 2, $f(\mathbf{x}_k) \leqslant f^*$ and hence

$$f(\bar{\mathbf{x}}) = \text{limit}_{k\in\mathcal{K}} f(\mathbf{x}_k) \leqslant f^*. \blacksquare$$

13.2 Barrier Methods

Barrier methods are applicable to problems of the form

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in S, \end{aligned} \tag{13.13}$$

where the constraint set S has a nonempty interior that is arbitrarily close to any point of S. Intuitively, what this means is that the set has an interior and it is possible to get to any boundary point by approaching it from the interior. We shall refer to such a set as *robust*. Some examples of robust and nonrobust sets are shown in Fig. 13.2. This kind of set often arises in conjunction with inequality constraints, where S takes the form

$$S = \{\mathbf{x} : g_i(\mathbf{x}) \leqslant 0, \ i = 1, 2, \ \ldots, \ p\}$$

Barrier methods are also termed *interior methods*. They work by establishing a barrier on the boundary of the feasible region that prevents a search procedure from leaving the region. A *barrier function* is a function B defined on the interior of S such that: (i) B is continuous, (ii) $B(\mathbf{x}) \geqslant 0$, (iii) $B(\mathbf{x}) \to \infty$ as $\mathbf{x}$ approaches the boundary of S.

Example 1. Let g_i, $i = 1, 2, \ \ldots, \ p$ be continuous functions on E^n. Suppose

$$S = \{\mathbf{x} : g_i(\mathbf{x}) \leqslant 0, \ i = 1, 2, \ \ldots, \ p\}.$$

is robust, and suppose the interior of S is the set of $\mathbf{x}$'s where $g_i(\mathbf{x}) < 0$, $i = 1, 2, \ \ldots, \ p$. Then the function

$$B(\mathbf{x}) = -\sum_{i=1}^{p} \frac{1}{g_i(\mathbf{x})},$$

defined on the interior of S, is a barrier function. It is illustrated in one dimension for $g_1 = x - a$, $g_2 = x - b$ in Fig. 13.3.

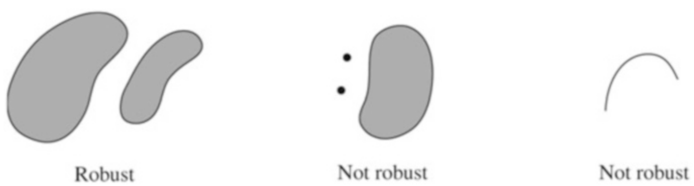

Fig. 13.2 Examples

Example 2. For the same situation as Example 1, we may use the logarithmic utility function

$$B(\mathbf{x}) = -\sum_{i=1}^{p} \log[-g_i(\mathbf{x})].$$

This is the barrier function commonly used in linear programming interior point methods, and it is frequently used more generally as well.

Corresponding to the problem (13.13), consider the approximate problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) + \tfrac{1}{c}B(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in \text{interior of } S, \end{aligned} \tag{13.14}$$

where c is a positive constant.

Alternatively, it is common to formulate the barrier method as

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) + \mu B(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{x} \in \text{interior of } S. \end{aligned} \tag{13.15}$$

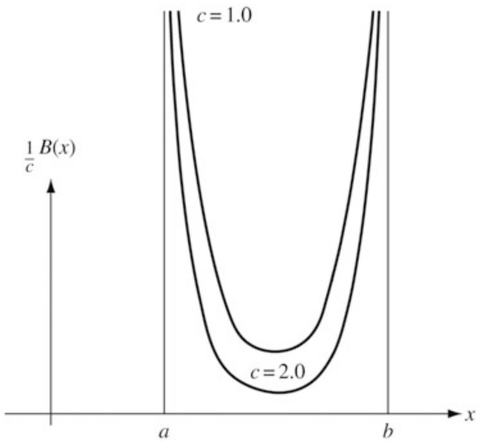

Fig. 13.3 Barrier function

When formulated with c we take c large (going to infinity); while when formulated with μ we take μ small (going to zero). Either way the result is a constrained problem, and indeed the constraint is somewhat more complicated than in the original problem (13.13). The advantage of this problem, however, is that it can be solved by using an unconstrained search technique. To find the solution one starts at an initial interior point and then searches from that point using steepest descent or some other iterative descent method applicable to unconstrained problems. Since the value of the objective function approaches infinity near the boundary of S, the search technique (if carefully implemented) will automatically remain within the interior of S, and the constraint need not be accounted for explicitly. Thus, although problem (13.14) or (13.15) is from a formal viewpoint a constrained problem, from a computational viewpoint it is unconstrained.

The Method

The barrier method is quite analogous to the penalty method. Let $\{c_k\}$ be a sequence tending to infinity such that for each k, $k = 1, 2, \ldots, c_k \geqslant 0, c_{k+1} > c_k$. Define the function

$$r(c, \mathbf{x}) = f(\mathbf{x}) + \frac{1}{c} B(\mathbf{x}).$$

For each k solve the problem

$$\begin{array}{ll} \text{minimize} & r(c_k, \mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \text{interior of} S, \end{array}$$

obtaining the point $\mathbf{x}_k$.

Convergence

Virtually the same convergence properties hold for the barrier method as for the penalty method. We leave to the reader the proof of the following result.

Theorem. *Any limit point of a sequence $\{x_k\}$ generated by the barrier method is a solution to problem (13.13).*

13.3 Properties of Penalty and Barrier Functions

Penalty and barrier methods are applicable to nonlinear programming problems having a very general form of constraint set S. In most situations, however, this set is not given explicitly but is defined implicitly by a number of functional constraints. In these situations, the penalty or barrier function is invariably defined in terms of the constraint functions themselves; and although there are an unlimited number of ways in which this can be done, some important general implications follow from this kind of construction.

For economy of notation we consider problems of the form

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & g_i(\mathbf{x}) \leqslant 0, \quad i = 1, 2, \ldots, p. \end{array} \tag{13.16}$$

For our present purposes, equality constraints are suppressed, at least notationally, by writing each of them as two inequalities. If the problem is to be attacked with a barrier method, then, of course, equality constraints are not present even in an unsuppressed version.

Penalty Functions

A penalty function for a problem expressed in the form (13.16) will most naturally be expressed in terms of the auxiliary constraint functions

$$g_i^+(\mathbf{x}) \equiv \max[0,\ g_i(\mathbf{x})],\ i = 1, 2,\ \ldots,\ p. \tag{13.17}$$

This is because in the interior of the constraint region $P(\mathbf{x}) \equiv 0$ and hence P should be a function only of violated constraints. Denoting by $\mathbf{g}^+(\mathbf{x})$ the p-dimensional vector made up of the $g_i^+(\mathbf{x})$'s, we consider the general class of penalty functions

$$P(\mathbf{x}) = \gamma(\mathbf{g}^+(\mathbf{x})), \tag{13.18}$$

where γ is a continuous function from E^p to the real numbers, defined in such a way that P satisfies the requirements demanded of a penalty function.

Example 1. Set

$$P(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{p} g_i^+(\mathbf{x})^2 = \frac{1}{2}|\mathbf{g}^+(\mathbf{x})|^2,$$

which is without doubt the most popular penalty function. In this case γ is one-half times the identity quadratic form on E^p, that is, $\gamma(\mathbf{y}) = \frac{1}{2}|\mathbf{y}|^2$.

Example 2. By letting

$$\gamma(\mathbf{y}) = \mathbf{y}^T\mathbf{\Gamma}\mathbf{y},$$

where $\mathbf{\Gamma}$ is a symmetric positive definite $p \times p$ matrix, we obtain the penalty function

$$P(\mathbf{x}) = \mathbf{g}^+(\mathbf{x})^T\mathbf{\Gamma}\mathrm{g}^+(\mathbf{x}).$$

Example 3. A general class of penalty functions is

$$P(\mathbf{x}) = \sum_{i=1}^{p}(g_i^+(\mathbf{x}))^{\varepsilon}$$

for some $\varepsilon > 0$.

Lagrange Multipliers

In the penalty method we solve, for various c_k, the unconstrained problem

$$\text{minimize} \quad f(\mathbf{x}) + c_k P(\mathbf{x}). \tag{13.19}$$

Most algorithms require that the objective function has continuous first partial derivatives. Since we shall, as usual, assume that both f and $\mathbf{g} \in C^1$, it is natural to require, then, that the penalty function $P \in C^1$. We define

$$\boldsymbol{\nabla} g_i^+(\mathbf{x}) = \begin{cases} \boldsymbol{\nabla} g_i(\mathbf{x}) \text{ if } g_i(\mathbf{x}) \geqslant 0 \\ \mathbf{0} \qquad\quad \text{if } g_i(\mathbf{x}) < 0 \end{cases}, \tag{13.20}$$

and, of course, $\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x})$ is the $m \times n$ matrix whose rows are the $\boldsymbol{\nabla}\mathbf{g}_i^+$'s. Unfortunately, $\boldsymbol{\nabla}\mathbf{g}^+$ is usually discontinuous at points where $g_i^+(\mathbf{x}) = 0$ for some $i = 1, 2, \ldots, p$, and thus some restrictions must be placed on γ in order to guarantee $P \in C^1$. We assume that $\gamma \in C^1$ and that if $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, $\boldsymbol{\nabla}\gamma(\mathbf{y}) = (\nabla\gamma_1, \nabla\gamma_2, \ldots, \nabla\gamma_n)$, then

$$y_i = 0 \quad \text{implies} \quad \nabla\gamma_i = 0. \tag{13.21}$$

(In Example 3 above, for instance, this condition is satisfied only for $\varepsilon > 1$.) With this assumption, the derivative of $\gamma(\mathbf{g}^+(\mathbf{x}))$ with respect to $\mathbf{x}$ is continuous and can be written as $\boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}))\boldsymbol{\nabla}\mathbf{g}(\mathbf{x})$. In this result $\boldsymbol{\nabla}\mathbf{g}(\mathbf{x})$ legitimately replaces the discontinuous $\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x})$, because it is premultiplied by $\boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}))$. Of course, these considerations are necessary only for inequality constraints. If equality constraints are treated directly, the situation is far simpler.

In view of this assumption, problem (13.19) will have its solution at a point $\mathbf{x}_k$ satisfying

$$\boldsymbol{\nabla} f(\mathbf{x}_k) + c_k \boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}_k))\boldsymbol{\nabla}\mathbf{g}(\mathbf{x}_k) = \mathbf{0},$$

which can be written as

$$\boldsymbol{\nabla} f(\mathbf{x}_k) + \boldsymbol{\lambda}_k^T \boldsymbol{\nabla}\mathbf{g}(\mathbf{x}_k) = \mathbf{0}, \tag{13.22}$$

where

$$\boldsymbol{\lambda}_k^T \equiv c_k \boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}_k)). \tag{13.23}$$

Thus, associated with every c is a Lagrange multiplier vector that is determined after the unconstrained minimization is performed.

If a solution $\mathbf{x}^*$ to the original problem (13.16) is a regular point of the constraints, then there is a unique Lagrange multiplier vector $\boldsymbol{\lambda}^*$ associated with the solution. The result stated below says that $\boldsymbol{\lambda}_k \to \boldsymbol{\lambda}^*$.

Proposition. *Suppose that the penalty function method is applied to problem (13.16) using a penalty function of the form (13.18) with $\gamma \in C^1$ and satisfying (13.21). Corresponding to the sequence $\{\mathbf{x}_k\}$ generated by this method, define $\boldsymbol{\lambda}_k^T = c_k \boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}_k))$. If $\mathbf{x}_k \to \mathbf{x}^*$, a solution to (13.16), and this solution is a regular point, then $\boldsymbol{\lambda}_k \to \boldsymbol{\lambda}^*$, the Lagrange multiplier associated with problem (13.16).*

Proof left to the reader.

Example 4. For $P(\mathbf{x}) = \frac{1}{2}|\mathbf{g}^+(\mathbf{x})|^2$ we have $\boldsymbol{\lambda}_k = c_k\mathbf{g}^+(\mathbf{x}_k)$.

As a final observation we note that in general if $\mathbf{x}_k \to \mathbf{x}^*$, then since $\boldsymbol{\lambda}_k = c_k\boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}_k))^T \to \boldsymbol{\lambda}^*$, the sequence $\mathbf{x}_k$ approaches $\mathbf{x}^*$ from outside the constraint region. Indeed, as $\mathbf{x}_k$ approaches $\mathbf{x}^*$ all constraints that are active at $\mathbf{x}^*$ and have positive Lagrange multipliers will be violated at $\mathbf{x}_k$ because the corresponding components of $\boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}_k))$ are positive. Thus, if we assume that the active constraints are nondegenerate (all Lagrange multipliers are strictly positive), every active constraint will be approached from the outside.

The Hessian Matrix

Since the penalty function method must, for various (large) values of c, solve the unconstrained problem

$$\text{minimize} \quad f(\mathbf{x}) + cP(\mathbf{x}), \tag{13.24}$$

it is important, in order to evaluate the difficulty of such a problem, to determine the eigenvalue structure of the Hessian of this modified objective function. We show here that the structure becomes increasingly unfavorable as c increases.

Although in this section we require that the function $P \in C^1$, we do not require that $P \in C^2$. In particular, the most popular penalty function $P(\mathbf{x}) = \frac{1}{2}|\mathbf{g}^+(\mathbf{x})|^2$, illustrated in Fig. 13.1 for one component, has a discontinuity in its second derivative at any point where a component of $\mathbf{g}$ is zero. At first this might appear to be a serious drawback, since it means the Hessian is discontinuous at the boundary of the constraint region—right where, in general, the solution is expected to lie. However, as pointed out above, the penalty method generates points that approach a boundary solution from outside the constraint region. Thus, except for some possible chance occurrences, the sequence will, as $\mathbf{x}_k \to \mathbf{x}^*$, be at points where the Hessian is well-defined. Furthermore, in iteratively solving the unconstrained problem (13.24) with a fixed c_k, a sequence will be generated that converges to $\mathbf{x}_k$ which is (for most values of k) a point where the Hessian is well-defined, and hence the standard type of analysis will be applicable to the tail of such a sequence.

Defining $q(c, \mathbf{x}) = f(\mathbf{x}) + c\gamma(\mathbf{g}^+(\mathbf{x}))$ we have for the Hessian, $\mathbf{Q}$, of q (with respect to $\mathbf{x}$)

$$\mathbf{Q}(c, \mathbf{x}) = \mathbf{F}(\mathbf{x}) + c\boldsymbol{\nabla}\gamma(\mathbf{g}^+(\mathbf{x}))\mathbf{G}(\mathbf{x}) + c\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x})^T\boldsymbol{\Gamma}(\mathbf{g}^+(\mathbf{x}))\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}),$$

where $\mathbf{F}$, $\mathbf{G}$, and $\boldsymbol{\Gamma}$ are, respectively, the Hessians of f, $\mathbf{g}$, and γ. For a fixed c_k we use the definition of $\boldsymbol{\lambda}_k$ given by (13.23) and introduce the rather natural definition

$$\mathbf{L}_k(\mathbf{x}_k) = \mathbf{F}(\mathbf{x}_k) + \boldsymbol{\lambda}_k^T\mathbf{G}(\mathbf{x}_k), \tag{13.25}$$

which is the Hessian of the corresponding Lagrangian. Then we have

$$\mathbf{Q}(c_k, \mathbf{x}_k) = \mathbf{L}_k(\mathbf{x}_k) + c_k\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k)^T\boldsymbol{\Gamma}(\mathbf{g}^+(\mathbf{x}_k))\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k), \tag{13.26}$$

which is the desired expression.

The first term on the right side of (13.26) converges to the Hessian of the Lagrangian of the original constrained problem as $\mathbf{x}_k \to \mathbf{x}^*$, and hence has a limit that is independent of c_k. The second term is a matrix having rank equal to the rank of the active constraints and having a magnitude tending to infinity. (See Exercise 7.)

Example 5. For $P(\mathbf{x}) = \frac{1}{2}|\mathbf{g}^+(\mathbf{x})|^2$ we have

$$\boldsymbol{\Gamma}(\mathbf{g}^+(\mathbf{x}_k)) = \begin{bmatrix} e_1 & 0 & \cdots 0 \\ 0 & e_2 & 0 \\ 0 & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 0 & \cdots 0 & e_p \end{bmatrix},$$

where

$$e_i = \begin{cases} 1 & \text{if } g_i(\mathbf{x}_k) > 0 \\ 0 & \text{if } g_i(\mathbf{x}_k) < 0 \\ \text{undefined} & \text{if } g_i(\mathbf{x}_k) = 0 \end{cases}$$

Thus

$$c_k \boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k)^T(\mathbf{g}^+(\mathbf{x}_k))\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k) = c_k \boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k)^T \boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}_k),$$

which is c_k times a matrix that approaches $\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}^*)^T\boldsymbol{\nabla}\mathbf{g}^+(\mathbf{x}^*)$. This matrix has rank equal to the rank of the active constraints at $\mathbf{x}^*$ [refer to (13.20)].

Assuming that there are r active constraints at the solution $\mathbf{x}^*$, then for well-behaved γ, the Hessian matrix $\mathbf{Q}(c_k,\ \mathbf{x}_k)$ has r eigenvalues that tend to infinity as $c_k \to \infty$, arising from the second term on the right side of (13.26). There will be $n - r$ other eigenvalues that, although varying with c_k, tend to finite limits. These limits turn out to be, as is perhaps not too surprising at this point, the eigenvalues of $\mathbf{L}(\mathbf{x}^*)$ restricted to the tangent subspace M of the active constraints. The proof of this requires some further analysis.

Lemma 1. *Let* $\mathbf{A}(c)$ *be a symmetric matrix written in partitioned form*

$$\boldsymbol{A}(c) = \begin{bmatrix} \boldsymbol{A}_1(c) & \boldsymbol{A}_2(c) \\ \boldsymbol{A}_2^T(c) & \boldsymbol{A}_3(c) \end{bmatrix}, \tag{13.27}$$

where $\boldsymbol{A}_1(c)$ *tends to a positive definite matrix* $\mathbf{A}_1$, $\boldsymbol{A}_2(c)$ *tends to a finite matrix, and* $\mathbf{A}_3(c)$ *is a positive definite matrix tending to infinity with c (that is, for any* $s > 0$, $\boldsymbol{A}_3(c) \to s\mathrm{I}$ *is positive definite for sufficiently large c). Then*

$$\boldsymbol{A}^{-1}(c) \to \begin{bmatrix} \boldsymbol{A}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{13.28}$$

as $c \to \infty$.

Proof. We have the identity

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_2^T & \mathbf{A}_3 \end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A}_1 - \mathbf{A}_2\mathbf{A}_3^{-1}\mathbf{A}_2^T)^{-1} & -(\mathbf{A}_1 - \mathbf{A}_2\mathbf{A}_3^{-1}\mathbf{A}_2^T)\mathbf{A}_2\mathbf{A}_3^{-1} \\ -\mathbf{A}_3^{-1}\mathbf{A}_2^T(\mathbf{A}_1 - \mathbf{A}_2\mathbf{A}_3^{-1}\mathbf{A}_2^T)^{-1} & (\mathbf{A}_3 - \mathbf{A}_2^T\mathbf{A}_1^{-1}\mathbf{A}_2)^{-1} \end{bmatrix}. \tag{13.29}$$

Using the fact that $\mathbf{A}_3^{-1}(c) \to \mathbf{0}$ gives the result. ∎

To apply this result to the Hessian matrix (13.26) we associate $\mathbf{A}$ with $\mathbf{Q}(c_k, \mathbf{x}_k)$ and let the partition of $\mathbf{A}$ correspond to the partition of the space E^n into the subspace M and the subspace N that is orthogonal to M; that is, N is the subspace spanned by the gradients of the active constraints. In this partition, $\mathbf{L}_M$, the restriction of $\mathbf{L}$ to M, corresponds to the matrix $\mathbf{A}_1$.

We leave the details of the required continuity arguments to the reader. The important conclusion is that if $\mathbf{x}^*$ is a solution to (13.16), is a regular point, and has exactly r active constraints none of which are degenerate, then the Hessian matrices $\mathbf{Q}(c_k, \mathbf{x}_k)$ of a penalty function of form (13.18) have r eigenvalues tending to infinity as $c_k \to \infty$, and $n - r$ eigenvalues tending to the eigenvalues of $\mathbf{L}_M$.

This explicit characterization of the structure of penalty function Hessians is of great importance in the remainder of the chapter. The fundamental point is that virtually any choice of penalty function (within the class considered) leads both to an ill-conditioned Hessian and to consideration of the ubiquitous Hessian of the Lagrangian restricted to M.

Barrier Functions

Essentially the same story holds for barrier function. If we consider for Problem (13.16) barrier functions of the form

$$B(\mathbf{x}) = \eta(\mathbf{g}(\mathbf{x})), \tag{13.30}$$

then Lagrange multipliers and ill-conditioned Hessians are again inevitable. Rather than parallel the earlier analysis of penalty functions, we illustrate the conclusions with two examples.

Example 1. Define

$$B(\mathbf{x}) = \sum_{i=1}^{p} -\frac{1}{g_i(\mathbf{x})}. \tag{13.31}$$

The barrier objective

$$r(c_k, \mathbf{x}) = f(\mathbf{x}) - \frac{1}{c_k}\sum_{i=1}^{p}\frac{1}{g_i(\mathbf{x})}$$

has its minimum at a point $\mathbf{x}_k$ satisfying

$$\nabla f(\mathbf{x}_k) + \frac{1}{c_k}\sum_{i=1}^{p}\frac{1}{g_i(\mathbf{x}_k)^2}\nabla g_i(\mathbf{x}_k) = \mathbf{0}. \tag{13.32}$$

Thus, we define $\boldsymbol{\lambda}_k$ to be the vector having ith component $\frac{1}{c_k} \cdot \frac{1}{g_i(\mathbf{x}_k)^2}$. Then (13.32) can be written as

$$\nabla f(\mathbf{x}_k) + \boldsymbol{\lambda}_k^T \nabla \mathbf{g}(\mathbf{x}_k) = \mathbf{0}.$$

Again, assuming $\mathbf{x}_k \to \mathbf{x}^*$, the solution of (13.16), we can show that $\boldsymbol{\lambda}_k \to \boldsymbol{\lambda}^*$, the Lagrange multiplier vector associated with the solution. This implies that if g_i is an active constraint,

$$\frac{1}{c_k g_i(\mathbf{x}_k)^2} \to \lambda_i^* < \infty. \tag{13.33}$$

Next, evaluating the Hessian $\mathbf{R}(c_k,\ \mathbf{x}_k)$ of $r(c_k,\ \mathbf{x}_k)$, we have

$$\begin{aligned}\mathbf{R}(c_k,\ \mathbf{x}_k) &= \mathbf{F}(\mathbf{x}_k) + \frac{1}{c_k}\sum_{i=1}^{p}\frac{1}{g_i(\mathbf{x}_k)^2}\mathbf{G}_i(\mathbf{x}_k) - \frac{1}{c_k}\sum_{i=1}^{p}\frac{2}{g_i(\mathbf{x}_k)^3}\nabla g_i(\mathbf{x}_k)^T\nabla g_i(\mathbf{x}_k)\\ &= \mathbf{L}(\mathbf{x}_k) - \frac{1}{c_k}\sum_{i=1}^{p}\frac{2}{g_i(\mathbf{x}_k)^3}\nabla g_i(\mathbf{x}_k)^T\nabla g_i(\mathbf{x}_k).\end{aligned}$$

As $c_k \to \infty$ we have

$$\frac{-1}{c_k g_i(\mathbf{x}_k)^3} \to \begin{cases}\infty \text{ if} & g_i \text{ is active at } \mathbf{x}^*\\ 0 \ \text{ if} & g_i \text{ is inactive at } \mathbf{x}^*\end{cases}$$

so that we may write, from (13.33),

$$\mathbf{R}(c_k,\ \mathbf{x}_k) \to \mathbf{L}(\mathbf{x}_k) + \sum_{i\in 1} -\frac{\lambda_{i,k}}{g_i(\mathbf{x}_k)}\nabla g_i(\mathbf{x}_k)^T\nabla g_i(\mathbf{x}_k),$$

where I is the set of indices corresponding to active constraints. Thus the Hessian of the barrier objective function has exactly the same structure as that of penalty objective functions.

Example 2. Let us use the logarithmic barrier function

$$B(\mathbf{x}) = -\sum_{i=1}^{p}\log[-g_i(\mathbf{x})]$$

In this case we will define the barrier objective in terms of μ as

$$r(\mu,\ \mathbf{x}) = f(\mathbf{x}) - \mu\sum_{i=1}^{p}\log[-g_i(\mathbf{x})]$$

The minimum point $\mathbf{x}_\mu$ satisfies

$$\mathbf{0} = \nabla f(\mathbf{x}_\mu) + \mu\sum_{i=1}^{p}\frac{-1}{g_i(\mathbf{x}_\mu)}\nabla g_i(\mathbf{x}_\mu). \tag{13.34}$$

Defining

$$\lambda_{\mu,i} = \mu\frac{-1}{g_i(\mathbf{x}_\mu)}$$

(13.34) can be written as

$$\nabla f(\mathbf{x}_\mu) + \boldsymbol{\lambda}_\mu^T \nabla \mathbf{g}(\mathbf{x}_\mu) = \mathbf{0}.$$

Further we expect that $\boldsymbol{\lambda}_\mu \to \boldsymbol{\lambda}^*$ as $\mu \to 0$.

The Hessian of $r(\mu,\ \mathbf{x})$ is

$$R(\mu,\ \mathbf{x}_\mu) = \mathbf{F}(\mathbf{x}_\mu) + \sum_{i=1}^{p} \lambda_{i,\mu} \mathbf{G}_i(\mathbf{x}_\mu) + \sum_{i=1}^{p} -\frac{\lambda_{i,\mu}}{g_I(\mathbf{x}_\mu)} \nabla g_i(\mathbf{x}_\mu)^T \nabla g_i(\mathbf{x}_\mu).$$

Hence, for small μ it has the same structure as that found in Example 1.

The Central Path

The definition of the central path associated with linear programs is easily extended to general nonlinear programs. For example, consider the problem

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) = \mathbf{0},\ \mathbf{g}(\mathbf{x}) \leq \mathbf{0}. \end{array}$$

We assume that $\mathring{\mathcal{F}} = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0},\ \mathbf{g}(\mathbf{x}) < \mathbf{0}\} \neq \phi$. Then we use the logarithmic barrier function to define the problems

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) - \mu \sum_{i=1}^{p} \log[-g_i(\mathbf{x})] \\ \text{subject to} & \mathbf{h}(\mathbf{x}) = \mathbf{0}. \end{array}$$

The solution $\mathbf{x}_\mu$ parameterized by $\mu \to 0$ is called the central path; see Chap. 5.

The necessary conditions for the problem can be written as

$$\begin{aligned} \nabla f(\mathbf{x}_\mu) + \mathbf{s}^T \nabla \mathbf{g}(\mathbf{x}_\mu) + \mathbf{y}^T \nabla \mathbf{h}(\mathbf{x}_\mu) &= \mathbf{0} \\ \mathbf{h}(\mathbf{x}_\mu) &= \mathbf{0}. \\ s_i g_i(\mathbf{x}_\mu) &= -\mu;\ i = 1, 2,\ \ldots,\ p \end{aligned}$$

where $\mathbf{y}$ is the Lagrange multiplier vector for the constraint $\mathbf{h}(\mathbf{x}_\mu) = \mathbf{0}$. Then, the Newton method can be applied to solving the condition system as μ is gradually reduced to 0, that is, following the path.

Geometric Interpretation: The Primal Function

There is a geometric construction that provides a simple interpretation of penalty functions. The basis of the construction itself is also useful in other areas of optimization, especially duality theory, as explained in the next chapter.

Let us again consider the problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{13.35}$$

where $\mathbf{h}(\mathbf{x}) \in E^m$. We assume that the solution point $\mathbf{x}^*$ of (13.35) is a regular point and that the second-order sufficiency conditions are satisfied. Corresponding to this problem we introduce the following definition:

Definition. Corresponding to the constrained minimization problem (13.35), the *primal function* ω is defined on E^m in a neighborhood of $\mathbf{0}$ to be

$$\omega(\mathbf{y}) = \min\{f(\mathbf{x}) : \mathbf{h}(\mathbf{x}) = \mathbf{y}\}. \tag{13.36}$$

The primal function gives the optimal value of the objective for various values of the right-hand side. In particular $\omega(0)$ gives the value of the original problem.

Strictly speaking the minimum in the definition (13.36) must be specified as a local minimum, in a neighborhood of $\mathbf{x}^*$. The existence of $\omega(\mathbf{y})$ then follows directly from the Sensitivity Theorem in Sect. 11.7. Furthermore, from that theorem it follows that $\nabla\omega(\mathbf{0}) = -\boldsymbol{\lambda}^{*T}$.

Now consider the penalty problem and note the following relations:

$$\begin{aligned} \min\{f(\mathbf{x}) + \frac{1}{2}c|\mathbf{h}(\mathbf{x})|^2\} &= \min_{\mathbf{x},\mathbf{y}}\{f(\mathbf{x}) + \frac{1}{2}c|\mathbf{y}|^2 : \mathbf{h}(\mathbf{x}) = \mathbf{y}\} \\ &= \min_{\mathbf{y}}\{\omega(\mathbf{y}) + \frac{1}{2}c|\mathbf{y}|^2\}. \end{aligned} \tag{13.37}$$

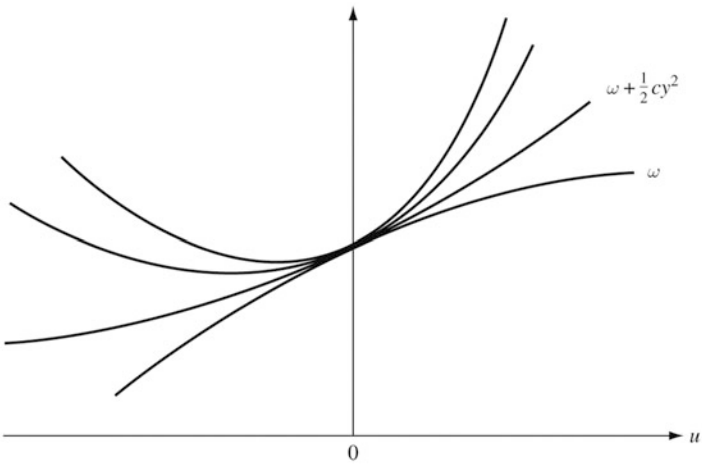

Fig. 13.4 The primal function

This is illustrated in Fig. 13.4 for the case where y is one-dimensional. The primal function is the lowest curve in the figure. Its value at $y = 0$ is the value of the original constrained problem. Above the primal function are the curves $\omega(y) + \frac{1}{2}cy^2$ for various values of c. The value of the penalty problem is shown by (13.37) to be the minimum point of this curve. For large values of c this curve becomes convex near 0 even if $\omega(y)$ is not convex. Viewed in this way, the penalty functions can be thought of as *convexifying* the primal.

Also, as c increases, the associated minimum point moves toward 0. However, it is never zero for finite c. Furthermore, in general, the criterion for $\mathbf{u}$ to be optimal for the penalty problem is that the gradient of $\omega(\mathbf{y}) + \frac{1}{2}c\mathbf{y}^2$ equals zero. This yields $\nabla\omega(\mathbf{y}) + c\mathbf{y}^T = \mathbf{0}$. Using $\nabla\omega(\mathbf{y}) = -\boldsymbol{\lambda}^T$ and $\mathbf{y} = \mathbf{h}(\mathbf{x}_c)$, where now $\mathbf{x}_c$ denotes the minimum point of the penalty problem, gives $\boldsymbol{\lambda} = c\mathbf{h}(\mathbf{x}_c)$, which is the same as (13.23).

13.4 Newton's Method and Penalty Functions

In the next few sections we address the problem of efficiently solving the unconstrained problems associated with a penalty or barrier method. The main difficulty is the extremely unfavorable eigenvalue structure that, as explained in Sect. 13.3, always accompanies unconstrained problems derived in this way. Certainly straightforward application of the method of steepest descent is out of the question!

One method for avoiding slow convergence for these problems is to apply Newton's method (or one of its variations), since the order two convergence of Newton's method is unaffected by the poor eigenvalue structure. In applying the method, however, special care must be devoted to the manner by which the Hessian is inverted, since it is ill-conditioned. Nevertheless, if second-order information is easily available, Newton's method offers an extremely attractive and effective method for solving modest size penalty or barrier optimization problems. When such information is not readily available, or if data handling and storage requirements of Newton's method are excessive, attention naturally focuses on first-order methods.

A simple modified Newton's method can often be quite effective for some penalty problems. For example, consider the problem having only equality constraints

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{13.38}$$

with $\mathbf{x} \in E^n$, $\mathbf{h}(\mathbf{x}) \in E^m$, $m < n$. Applying the standard quadratic penalty method we solve instead the unconstrained problem

$$\text{minimize } f(\mathbf{x}) + \tfrac{1}{2}c|\mathbf{h}(\mathbf{x})|^2 \tag{13.39}$$

for some large c. Calling the penalty objective function $q(\mathbf{x})$ we consider the iterative process

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k[\mathbf{I} + c\nabla\mathbf{h}(\mathbf{x}_k)^T\nabla\mathbf{h}(\mathbf{x}_k)]^{-1}\nabla q(\mathbf{x}_k)^T, \tag{13.40}$$

where α_k is chosen to minimize $q(\mathbf{x}_{k+1})$. The matrix $\mathbf{I} + c\nabla\mathbf{h}(\mathbf{x}_k)^T\nabla\mathbf{h}(\mathbf{x}_k)$ is positive definite and although quite ill-conditioned it can be inverted efficiently (see Exercise 11).

According to the Modified Newton Method Theorem (Sect. 10.1) the rate of convergence of this method is determined by the eigenvalues of the matrix

$$[\mathbf{I} + c\nabla\mathbf{h}(\mathbf{x}_k)^T\nabla\mathbf{h}(\mathbf{x}_k)]^{-1}\mathbf{Q}(\mathbf{x}_k), \tag{13.41}$$

where $\mathbf{Q}(\mathbf{x}_k)$ is the Hessian of q at $\mathbf{x}_k$. In view of (13.26), as $c \to \infty$ the matrix (13.41) will have m eigenvalues that approach unity, while the remaining $n - m$ eigenvalues approach the eigenvalues of $\mathbf{L}_M$ evaluated at the solution $\mathbf{x}^*$ of (13.38). Thus, if the smallest and largest eigenvalues of $\mathbf{L}_M$, a and A, are located such that the interval $[a, A]$ contains unity, the convergence ratio of this modified Newton's method will be equal (in the limit of $c \to \infty$) to the canonical ratio $[(A-a)/(A+a)]^2$ for problem (13.38).

If the eigenvalues of $\mathbf{L}_M$ are not spread below and above unity, the convergence rate will be slowed. If a point in the interval containing the eigenvalues of $\mathbf{L}_M$ is known, a scalar factor can be introduced so that the canonical rate is achieved, but such information is often not easily available.

Inequalities

If there are inequality as well as equality constraints in the problem, the analogous procedure can be applied to the associated penalty objective function. The unusual feature of this case is that corresponding to an inequality constraint $g_i(\mathbf{x}) \leqslant 0$, the term $\nabla g_i^+(\mathbf{x})^T\nabla g_i^+(\mathbf{x})$ used in the iteration matrix will suddenly appear if the constraint is violated. Thus the iteration matrix is discontinuous with respect to $\mathbf{x}$, and as the method progresses its nature changes according to which constraints are violated. This discontinuity does not, however, imply that the method is subject to jamming, since the result of Exercise 4, Chap. 10 is applicable to this method.

13.5 Conjugate Gradients and Penalty Methods

The partial conjugate gradient method proposed and analyzed in Sect. 9.5 is ideally suited to penalty or barrier problems having only a few active constraints. If there are m active constraints, then taking cycles of $m + 1$ conjugate gradient steps will yield a rate of convergence that is independent of the penalty constant c. For example, consider the problem having only equality constraints:

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{13.42}$$

where $\mathbf{x} \in E^n$, $\mathbf{h}(\mathbf{x}) \in E^m$, $m < n$. Applying the standard quadratic penalty method, we solve instead the unconstrained problem

$$\text{minimize } f(\mathbf{x}) + \tfrac{1}{2}c|\mathbf{h}(\mathbf{x})|^2 \tag{13.43}$$

for some large c. The objective function of this problem has a Hessian matrix that has m eigenvalues that are of the order c in magnitude, while the remaining $n - m$ eigenvalues are close to the eigenvalues of the matrix $\mathbf{L}_M$, corresponding to problem (13.42). Thus, letting $\mathbf{x}_{k+1}$ be determined from $\mathbf{x}_k$ by taking $m + 1$ steps of a (non-quadratic) conjugate gradient method, and assuming $\mathbf{x}_k \to \overline{\mathbf{x}}$, a solution to (13.43), the sequence $\{f(\mathbf{x}_k)\}$ converges linearly to $f(\overline{\mathbf{x}})$ with a convergence ratio equal to approximately

$$\left(\frac{A - a}{A + a}\right)^2 \tag{13.44}$$

where a and A are, respectively, the smallest and largest eigenvalues of $\mathbf{L}_M(\overline{\mathbf{x}})$.

This is an extremely effective technique when m is relatively small. The programming logic required is only slightly greater than that of steepest descent, and the time per iteration is only about $m + 1$ times as great as for steepest descent. The method can be used for problems having inequality constraints as well but it is advisable to change the cycle length, depending on the number of constraints active at the end of the previous cycle.

Example 3.

$$\begin{aligned}
\text{minimize } & f(x_1, x_2, \ldots, x_{10}) = \sum_{k=1}^{10} kx_k^2 \\
\text{subject to } & 1.5x_1 + x_2 + x_3 + 0.5x_4 + 0.5x_5 = 5.5 \\
& 2.0x_6 - 0.5x_7 - 0.5x_8 + x_9 - x_{10} = 2.0 \\
& x_1 + x_3 + x_5 + x_7 + x_9 = 10 \\
& x_2 + x_4 + x_6 + x_8 + x_{10} = 15.
\end{aligned}$$

This problem was treated by the penalty function approach, and the resulting composite function was then solved for various values of c by using various cycle lengths of a conjugate gradient algorithm. In Table 13.1 p is the number of conjugate gradient steps in a cycle. Thus, $p = 1$ corresponds to ordinary steepest descent; $p = 5$ corresponds, by the theory of Sect. 9.5, to the smallest value of p for which the rate of convergence is independent of c; and $p = 10$ is the standard conjugate gradient method. Note that for $p < 5$ the convergence rate does indeed depend on c, while it is more or less constant for $p \geqslant 5$. The value of c's selected are not artificially large, since for $c = 200$ the constraints are satisfied only to within 0.5 % of their right-hand sides. For problems with nonlinear constraints the results will most likely be somewhat less favorable, since the predicted convergence rate would apply only to the tail of the sequence.

Table 13.1 Results for example 3

	p (steps per cycle)	Number of cycles to convergence	No. of steps	Value of modified objective
$c = 20$	1	90	90	388.565
	3	8	24	388.563
	5	3	15	388.563
	7	3	21	388.563
$c = 200$	1	230[a]	230	488.607
	3	21	63	487.446
	5	4	20	487.438
	7	2	14	487.433
$c = 2000$	1	260[a]	260	525.238
	3	45[a]	135	503.550
	5	3	15	500.910
	7	3	21	500.882

[a] Program not run to convergence due to excessive time

13.6 Normalization of Penalty Functions

There is a good deal of freedom in the selection of penalty or barrier functions that can be exploited to accelerate convergence. We propose here a simple normalization procedure that together with a two-step cycle of conjugate gradients yields the canonical rate of convergence. Again for simplicity we illustrate the technique for the penalty method applied to the problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{13.45}$$

as in Sects. 13.4 and 13.5, but the idea is easily extended to other penalty or barrier situations.

Corresponding to (13.45) we consider the family of quadratic penalty functions

$$P(\mathbf{x}) = \frac{1}{2}\mathbf{h}(\mathbf{x})^T \boldsymbol{\Gamma} \mathbf{h}(\mathbf{x}), \tag{13.46}$$

where $\boldsymbol{\Gamma}$ is a symmetric positive definite $m \times m$ matrix. We ask what the best choice of $\boldsymbol{\Gamma}$ might be.

Letting

$$q(c, \mathbf{x}) = f(\mathbf{x}) + cP(\mathbf{x}), \tag{13.47}$$

the Hessian of q turns out to be, using (13.26),

$$\mathbf{Q}(c, \mathbf{x}_k) = \mathbf{L}(\mathbf{x}_k) + c\nabla\mathbf{h}(\mathbf{x}_k)^T \boldsymbol{\Gamma} \nabla\mathbf{h}(\mathbf{x}_k). \tag{13.48}$$

The m large eigenvalues are due to the second term on the right. The observation we make is that although the m large eigenvalues are all proportional to c, they are not necessarily all equal. Indeed, for very large c these eigenvalues are determined almost exclusively by the second term, and are therefore c times the nonzero eigenvalues of the matrix $\mathbf{\nabla h}(\mathbf{x}_k)^T\mathbf{\Gamma}\mathbf{\nabla h}(\mathbf{x}_k)$. We would like to select $\mathbf{\Gamma}$ so that these eigenvalues are not spread out but are nearly equal to one another. An ideal choice for the kth iteration would be

$$\mathbf{\Gamma} = [\mathbf{\nabla h}(\mathbf{x}_k)\mathbf{\nabla h}(\mathbf{x}_k)^T]^{-1}, \tag{13.49}$$

since then all nonzero eigenvalues would be exactly equal. However, we do not allow to change at each step, and therefore compromise by setting

$$\mathbf{\Gamma} = [\mathbf{\nabla h}(\mathbf{x}_0)\mathbf{\nabla h}(\mathbf{x}_0)^T]^{-1}, \tag{13.50}$$

where $\mathbf{x}_0$ is the initial point of the iteration.

Using this penalty function, the corresponding eigenvalue structure will at any point look approximately like that shown in Fig. 13.5. The eigenvalues are bunched into two separate groups. As c is increased the smaller eigenvalues move into the interval $[a, A]$ where a and A are, as usual, the smallest and largest eigenvalues of $\mathbf{L}_M$ at the solution to (13.45). The larger eigenvalues move forward to the right and spread further apart.

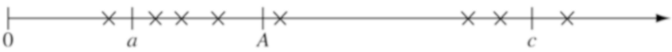

Fig. 13.5 Eigenvalue distributions

Using the result of Exercise 11, Chap. 9, we see that if $\mathbf{x}_{k+1}$ is determined from $\mathbf{x}_k$ by two conjugate gradient steps, the rate of convergence will be linear at a ratio determined by the widest of the two eigenvalue groups. If our normalization is sufficiently accurate, the large-valued group will have the lesser width. In that case convergence of this scheme is approximately that of the canonical rate for the original problem. Thus, by proper normalization it is possible to obtain the canonical rate of convergence for only about twice the time per iteration as required by steepest descent.

There are, of course, numerous variations of this method that can be used in practice. $\mathbf{\Gamma}$ can, for example, be allowed to vary at each step, or it can be occasionally updated.

Example. The example problem presented in the previous section was also solved by the normalization method presented above. The results for various values of c and for cycle lengths of one, two, and three are presented in Table 13.2. (All runs were initiated from the zero vector.)

Table 13.2 Results for example 3

	p (steps per cycle)	Number of cycles to convergence	No. of steps	Value of modified objective
$c = 10$	1	28	28	251.2657
	2	9	18	251.2657
	3	5	15	251.2657
$c = 100$	1	153	153	379.5955
	2	13	26	379.5955
	3	11	33	379.5955
$c = 1000$	1	261[a]	261	402.0903
	2	14	28	400.1687
	3	13	39	400.1687

[a] Program not run to convergence due to excessive time

13.7 Penalty Functions and Gradient Projection

The penalty function method can be combined with the idea of the gradient projection method to yield an attractive general purpose procedure for solving constrained optimization problems. The proposed combination method can be viewed either as a way of accelerating the rate of convergence of the penalty function method by eliminating the effect of the large eigenvalues, or as a technique for efficiently handling the delicate and usually cumbersome requirement in the gradient projection method that each point be feasible. The combined method converges at the canonical rate (the same as does the gradient projection method), is globally convergent (unlike the gradient projection method), and avoids much of the computational difficulty associated with staying feasible.

Underlying Concept

The basic theoretical result that motivates the development of this algorithm is the Combined Steepest Descent and Newton's Method Theorem of Sect. 10.7. The idea is to apply this combined method to a penalty problem. For simplicity we first consider the equality constrained problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0}, \end{aligned} \tag{13.51}$$

where $\mathbf{x} \in E^n$, $\mathbf{h}(\mathbf{x}) \in E^m$. The associated unconstrained penalty problem that we consider is

$$\text{minimize} \quad q(\mathbf{x}), \tag{13.52}$$

where

$$q(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{2}c|\mathbf{h}(\mathbf{x})|^2.$$

At any point $\mathbf{x}_k$ let $M(\mathbf{x}_k)$ be the subspace tangent to the surface $S_k = \{\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{h}(\mathbf{x}_k)\}$. This is a slight extension of the tangent subspaces that we have considered before, since $M(\mathbf{x}_k)$ is defined even for points that are not feasible. If the sequence $\{\mathbf{x}_k\}$ converges to a solution $\mathbf{x}_c$ of problem (13.52), then we expect that $M(\mathbf{x}_k)$ will in some sense converge to $M(\mathbf{x}_c)$. The orthogonal complement of $M(\mathbf{x}_k)$ is the space generated by the gradients of the constraint functions evaluated at $\mathbf{x}_k$. Let us denote this space by $N(\mathbf{x}_k)$. The idea of the algorithm is to take N as the subspace over which Newton's method is applied, and M as the space over which the gradient method is applied. A cycle of the algorithm would be as follows:

1. Given $\mathbf{x}_k$, apply one step of Newton's method over, the subspace $N(\mathbf{x}_k)$ to obtain a point $\mathbf{w}_k$ of the form

$$\begin{aligned} \mathbf{w}_k &= \mathbf{x}_k + \nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{u}_k \\ \mathbf{u}_k &\in E^m. \end{aligned}$$

2. From $\mathbf{w}_k$, take an ordinary steepest descent step to obtain $\mathbf{x}_{k+1}$.

Of course, we must show how Step 1 can be easily executed, and this is done below, but first, without drawing out the details, let us examine the general structure of this algorithm.

The process is illustrated in Fig. 13.6. The first step is analogous to the step in the gradient projection method that returns to the feasible surface; except that here the criterion is reduction of the objective function rather than satisfaction of constraints. To interpret the second step, suppose for the moment that the original problem (13.51) has a quadratic objective and linear constraints; so that, consequently, the penalty problem (13.52) has a quadratic objective and $N(\mathbf{x})$, $M(\mathbf{x})$ and $\nabla \mathbf{h}(\mathbf{x})$ are independent of $\mathbf{x}$. In that case the first (Newton) step would exactly minimize q with respect to N, so that the gradient of q at $\mathbf{w}_k$ would be orthogonal to N; that is, the gradient would lie in the subspace M. Furthermore, since $\nabla q(\mathbf{w}_k) = \nabla f(\mathbf{w}_k) + c\mathbf{h}(\mathbf{w}_k)\nabla \mathbf{h}(\mathbf{w}_k)$, we see that $\nabla q(\mathbf{w}_k)$ would in that case be equal to the projection of the gradient of f onto M. Hence, the second step is, in the quadratic case exactly, and in the general case approximately, a move in the direction of the projected negative gradient of the original objective function.

The convergence properties of such a scheme are easily predicted from the theorem on the Combined Steepest Descent and Newton's Method, in Sect. 10.7, and our analysis of the structure of the Hessian of the penalty objective function given by (13.26). As $\mathbf{x}_k \to \mathbf{x}_c$ the rate will be determined by the ratio of largest to smallest eigenvalues of the Hessian restricted to $M(\mathbf{x}_c)$.

This leads, however, by what was shown in Sect. 12.3, to approximately the canonical rate for problem (13.51). Thus this combined method will yield again the canonical rate as $c \to \infty$.

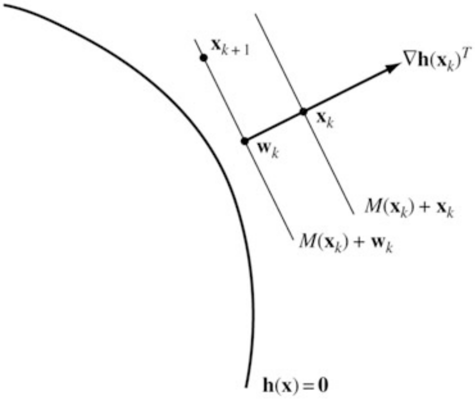

Fig. 13.6 Illustration of the method

Implementing the First Step

To implement the first step of the algorithm suggested above it is necessary to show how a Newton step can be taken in the subspace $N(\mathbf{x}_k)$. We show that, again for large values of c, this can be accomplished easily.

At the point $\mathbf{x}_k$ the function b, defined by

$$b(\mathbf{u}) = q(\mathbf{x}_k + \nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{u}) \tag{13.53}$$

for $\mathbf{u} \in E^m$, measures the variations in q with respect to displacements in $N(\mathbf{x}_k)$. We shall, for simplicity, assume that at each point, $\mathbf{x}_k$, $\nabla \mathbf{h}(\mathbf{x}_k)$ has rank m. We can immediately calculate the gradient with respect to $\mathbf{u}$,

$$\nabla b(\mathbf{u}) = \nabla q(\mathbf{x}_k + \nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{u}) \nabla \mathbf{h}(\mathbf{x}_k)^T, \tag{13.54}$$

and the $m \times n$ Hessian with respect to $\mathbf{u}$ at $\mathbf{u} = \mathbf{0}$,

$$\mathbf{B} = \nabla \mathbf{h}(\mathbf{x}_k) \mathbf{Q}(\mathbf{x}_k) \nabla \mathbf{h}(\mathbf{x}_k)^T. \tag{13.55}$$

where $\mathbf{Q}$ is the $n \times n$ Hessian of q with respect to $\mathbf{x}$. From (13.26) we have that at $\mathbf{x}_k$

$$\mathbf{Q}(\mathbf{x}_k) = \mathbf{L}_k(\mathbf{x}_k) + c \nabla \mathbf{h}(\mathbf{x}_k)^T \nabla \mathbf{h}(\mathbf{x}_k). \tag{13.56}$$

And given $\mathbf{B}$, the direction for the Newton step in N would be

$$\begin{aligned} \mathbf{d}_k &= -\nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{B}^{-1} \nabla c(\mathbf{0})^T \\ &= -\nabla \mathbf{h}(\mathbf{x}_k)^T \mathbf{B}^{-1} \nabla \mathbf{h}(\mathbf{x}_k) \nabla q(\mathbf{x}_k)^T. \end{aligned} \tag{13.57}$$

It is clear from (13.55) and (13.56) that exact evaluation of the Newton step requires knowledge of $\mathbf{L}(\mathbf{x}_k)$ which usually is costly to obtain. For large values of c, however, $\mathbf{B}$ can be approximated by

$$\mathbf{B} \simeq c[\nabla \mathbf{h}(\mathbf{x}_k) \nabla \mathbf{h}(\mathbf{x}_k)^T]^2, \tag{13.58}$$

and hence a good approximation to the Newton direction is

$$\mathbf{d}_k = -\frac{1}{c}\nabla\mathbf{h}(\mathbf{x}_k)^T[\nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T]^{-2}\nabla\mathbf{h}(\mathbf{x}_k)\nabla q(\mathbf{x}_k)^T. \tag{13.59}$$

Thus a suitable implementation of one cycle of the algorithm is:

1. Calculate

$$\mathbf{d}_k = -\frac{1}{c}\nabla\mathbf{h}(\mathbf{x}_k)^T[\nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T]^{-2}\nabla\mathbf{h}(\mathbf{x}_k)\nabla q(\mathbf{x}_k)^T.$$

2. Find β_k to minimize $q(\mathbf{x}_k + \beta\mathbf{d}_k)$ (using $\beta_k = 1$ as an initial search point), and set $\mathbf{w}_k = \mathbf{x}_k + \beta_k\mathbf{d}_k$.
3. Calculate $\mathbf{p}_k = -\nabla q(\mathbf{w}_k)^T$.
4. Find α_k to minimize $q(\mathbf{w}_k + \alpha\mathbf{p}_k)$, and set $\mathbf{x}_{k+1} = \mathbf{w}_k + \alpha_k\mathbf{p}_k$.

It is interesting to compare the Newton step of this version of the algorithm with the step for returning to the feasible region used in the ordinary gradient projection method. We have

$$\nabla q(\mathbf{x}_k)^T = \nabla f(\mathbf{x}_k)^T + c\nabla\mathbf{h}(\mathbf{x}_k)^T\mathbf{h}(\mathbf{x}_k). \tag{13.60}$$

If we neglect $\nabla f(\mathbf{x}_k)^T$ on the right (as would be valid if we are a long distance from the constraint boundary) then the vector $\mathbf{d}_k$ reduces to

$$\mathbf{d}_k = -\nabla\mathbf{h}(\mathbf{x}_k)^T[\nabla\mathbf{h}(\mathbf{x}_k)\nabla\mathbf{h}(\mathbf{x}_k)^T]^{-1}\mathbf{h}(\mathbf{x}_k),$$

which is precisely the first estimate used to return to the boundary in the gradient projection method. The scheme developed in this section can therefore be regarded as one which corrects this estimate by accounting for the variation in f.

An important advantage of the present method is that it is not necessary to carry out the search in detail. If $\beta = 1$ yields an improved value for the penalty objective, no further search is required. If not, one need search only until some improvement is obtained. At worst, if this search is poorly performed, the method degenerates to steepest descent. When one finally gets close to the solution, however, $\beta = 1$ is bound to yield an improvement and terminal convergence will progress at nearly the canonical rate.

Inequality Constraints

The procedure is conceptually the same for problems with inequality constraints. The only difference is that at the beginning of each cycle the subspace $M(\mathbf{x}_k)$ is calculated on the basis of those constraints that are either active or violated at $\mathbf{x}_k$, the others being ignored. The resulting technique is a descent algorithm in that the penalty objective function decreases at each cycle; it is globally convergent because of the pure gradient step taken at the end of each cycle; its rate of convergence approaches the canonical rate for the original constrained problem as $c \to \infty$; and there are no feasibility tolerances or subroutine iterations required.

*13.8 *Exact Penalty Functions

It is possible to construct penalty functions that are exact in the sense that the solution of the penalty problem yields the exact solution to the original problem for a finite value of the penalty parameter. With these functions it is not necessary to solve an infinite sequence of penalty problems to obtain the correct solution. However, a new difficulty introduced by these penalty functions is that they are non-differentiable.

For the general constrained problem

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{h}(\mathbf{x}) = \mathbf{0},\ \mathbf{g}(\mathbf{x}) \leqslant \mathbf{0}, \end{array} \tag{13.61}$$

consider the absolute-value penalty function

$$P(\mathbf{x}) = \sum_{i=1}^{m} |h_i(\mathbf{x})| + \sum_{j=1}^{p} \max(0,\ g_j(\mathbf{x})). \tag{13.62}$$

The penalty problem is then, as usual,

$$\text{minimize } f(\mathbf{x}) + cP(\mathbf{x}) \tag{13.63}$$

for some positive constant c. We investigate the properties of the absolute-value penalty function through an example and then generalize the results.

Example 1. Consider the simple quadratic problem

$$\begin{array}{ll} \text{minimize} & 2x^2 + 2xy + y^2 - 2y \\ \text{subject to} & x = 0. \end{array} \tag{13.64}$$

It is easy to solve this problem directly by substituting $x = 0$ into the objective. This leads immediately to $x = 0,\ y = 1$.

If a standard quadratic penalty function is used, we minimize the objective

$$2x^2 + 2xy + y^2 - 2y + \frac{1}{2}cx^2 \tag{13.65}$$

for $c > 0$. The solution again can be easily found and is $x = -2/(2+c)$, $y = 1 - 2/(2+c)$. This solution approaches the true solution as $c \to \infty$, as predicted by the general theory. However, for any finite c the solution is inexact.

Now let us use the absolute-value penalty function. We minimize the function

$$2x^2 + 2xy + y^2 - 2y + c|x|. \tag{13.66}$$

We rewrite (13.66) as

$$\begin{aligned} 2x^2 + 2xy + y^2 - 2y + c|x| \\ &= 2x^2 + 2xy + c|x| + (y-1)^2 - 1 \\ &= 2x^2 + 2x + c|x| + (y-1)^2 + 2x(y-1) - 1 \\ &= x^2 + (2x + c|x|) + (y-1+x)^2 - 1. \end{aligned} \tag{13.67}$$

All terms (except the $-l$) are nonnegative if $c > 2$. Therefore, the minimum value of this expression is -1, which is achieved (uniquely) by $x = 0,\ y = 1$. Therefore, for $c > 2$ the minimum point of the penalty problem is the correct solution to the original problem (13.64).

We let the reader verify that $\lambda = -2$ for this example. The fact that $c > |\lambda|$ is required for the solution to be exact is an illustration of a general result given by the following theorem.

Exact Penalty Theorem. *Suppose that the point* $\mathbf{x}^*$ *satisfies the second-order sufficiency conditions for a local minimum of the constrained problem* (13.61). Let $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ be the corresponding Lagrange multipliers. Then for $c > \max\{|\lambda_i|,\ \mu_j : i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, p\}$, $\mathbf{x}^*$ is also a local minimum of the absolute-value penalty objective (13.62).

Proof. For simplicity we assume that there are equality constraints only. Define the primal function

$$\omega(\mathbf{z}) = \min_{\mathbf{x}}\{f(\mathbf{x}) : h_i(\mathbf{x}) = z_i \text{ for } i = 1, 2, \ldots, m\}. \tag{13.68}$$

The primal function was introduced in Sect. 12.3. Under our assumption the function exists in a neighborhood of $\mathbf{x}^*$ and is continuously differentiable, with $\nabla\omega(\mathbf{0}) = -\boldsymbol{\lambda}^T$.

Now define

$$\omega_c(\mathbf{z}) = \omega(\mathbf{z}) + c\sum_{i=1}^{m}|z_i|.$$

Then we have

$$\begin{aligned} \min_{\mathbf{x}}\{f(\mathbf{x}) + c\sum_{i=1}^{m}|h_i(\mathbf{x})|\} &= \min_{\mathbf{x},\mathbf{u}}\{f(\mathbf{x}) + c\sum_{i=1}^{m}|z_i| : \mathbf{h}(\mathbf{x}) = \mathbf{z}\} \\ &= \min_{\mathbf{u}}\{p(\mathbf{z}) + c\sum_{i=1}^{m}|z_i|\} \\ &= \min_{\mathbf{u}} p_c(\mathbf{z}). \end{aligned}$$

By the Mean Value Theorem,

$$\omega(\mathbf{z}) = \omega(\mathbf{0}) + \nabla\omega(\alpha\mathbf{z})\mathbf{z}$$

for some $\alpha,\ 0 \leqslant \alpha \leqslant 1$. Therefore,

$$\omega_c(\mathbf{z}) = \omega(\mathbf{0}) + \nabla\omega(\alpha\mathbf{z})\mathbf{z} + c\sum_{i=1}^{m}|z_i|. \tag{13.69}$$

We know that $\nabla\omega(\mathbf{z})$ is continuous at $\mathbf{0}$, and thus given $\varepsilon > 0$ there is a neighborhood of $\mathbf{0}$ such that $|\nabla\omega(\mathbf{z})_i| < |\lambda_i| + \varepsilon$. Thus

$$\nabla\omega(\alpha\mathbf{z})\mathbf{z} = \sum_{i=1}^{m} \nabla\omega(\alpha\mathbf{z})_i z_i \geqslant -\{\max_i |\nabla\omega(\alpha\mathbf{z})_i|\} \sum_{i=1}^{m} |z_i|$$
$$\geqslant -\{\max_i(|\lambda_i| + \varepsilon)\} \sum_{i=1}^{m} |z_i|.$$

Using this in (13.69), we obtain

$$\omega_c(\mathbf{z}) \geqslant p(\mathbf{0}) + (c - \varepsilon - \max|\lambda_i|) \sum_{i=1}^{m} |z_i|.$$

For $c > \varepsilon + \max|\lambda_i|$ it follows that $\omega_c(\mathbf{z})$ is minimized at $\mathbf{z} = \mathbf{0}$. Since ε was arbitrary, the result holds for $c > \max|\lambda_i|$.

This result is easily extended to include inequality constraints. (See Exercise 16.)

∎

It is possible to develop a geometric interpretation of the absolute-value penalty function analogous to the interpretation for ordinary penalty functions given in Fig. 13.4. Figure 13.7 corresponds to a problem for a single constraint. The smooth curve represents the primal function of the problem. Its value at 0 is the value of the original problem, and its slope at 0 is $-\lambda$. The function $\omega_c(z)$ is obtained by adding $c|z|$ to the primal function, and this function has a discontinuous derivative at $z = 0$. It is clear that for $c > |\lambda|$, this composite function has a minimum at exactly $z = 0$, corresponding to the correct solution.

There are other exact penalty functions but, like the absolute-value penalty function, most are nondifferentiable at the solution. Such penalty functions are for this reason difficult to use directly; special descent algorithms for nondifferentiable objective functions have been developed, but they can be cumbersome. Furthermore, although these penalty functions are exact for a large enough c, it is not known at the outset what magnitude is sufficient. In practice a progression of c's must often be used. Because of these difficulties, the major use of exact penalty functions in nonlinear programming is as *merit functions–measuring* the progress of descent but not entering into the determination of the direction of movement. This idea is discussed in Chap. 15.

13.9 Summary

Penalty methods approximate a constrained problem by an unconstrained problem that assigns high cost to points that are far from the feasible region. As the approximation is made more exact (by letting the parameter c tend to infinity) the solution of the unconstrained penalty problem approaches the solution to the

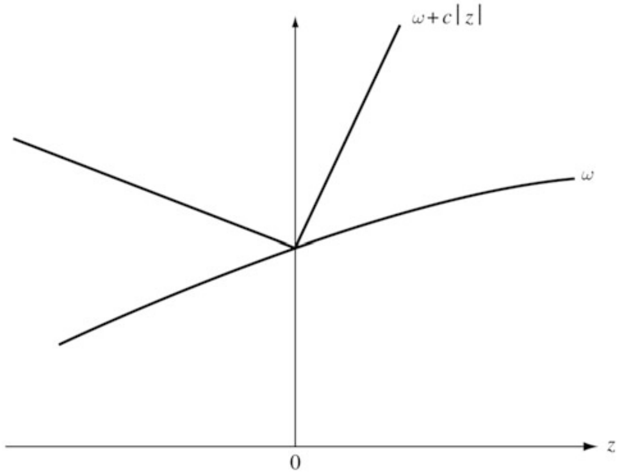

Fig. 13.7 Geometric interpretation of absolute-value penalty function

original constrained problem from outside the active constraints. Barrier methods, on the other hand, approximate a constrained problem by an (essentially) unconstrained problem that assigns high cost to being near the boundary of the feasible region, but unlike penalty methods, these methods are applicable only to problems having a robust feasible region. As the approximation is made more exact, the solution of the unconstrained barrier problem approaches the solution to the original constrained problem from inside the feasible region.

The objective functions of all penalty and barrier methods of the form $P(\mathbf{x}) = \gamma(h(\mathbf{x}))$, $B(\mathbf{x}) = \eta(g(\mathbf{x}))$ are ill-conditioned. If they are differentiable, then as $c \to \infty$ the Hessian (at the solution) is equal to the sum of $\mathbf{L}$, the Hessian of the Lagrangian associated with the original constrained problem, and a matrix of rank r that tends to infinity (where r is the number of active constraints). This is a fundamental property of these methods.

Effective exploitation of differentiable penalty and barrier functions requires that schemes be devised that eliminate the effect of the associated large eigenvalues. For this purpose the three general principles developed in earlier chapters, The Partial Conjugate Gradient Method, The Modified Newton Method, and The Combination of Steepest Descent and Newton's Method, when creatively applied, all yield methods that converge at approximately the canonical rate associated with the original constrained problem.

It is necessary to add a point of qualification with respect to some of the algorithms introduced in this chapter, lest it be inferred that they are offered as panaceas for the general programming problem. As has been repeatedly emphasized, the ideal study of convergence is a careful blend of analysis, good sense, and experimentation. The rate of convergence does not always tell the whole story, although it is often a major component of it. Although some of the algorithms presented in this chapter asymptotically achieve the canonical rate of convergence (at least approximately), for large c the points may have to be quite close to the solution before this rate characterizes the process. In other words, for large c the process may converge slowly in its initial phase, and, to obtain a truly representative analysis, one must look beyond the first-order convergence properties of these methods. For this reason many people find Newton's method attractive, although the work at each step can be substantial.

13.10 Exercises

1. Show that if $q(c, \mathbf{x})$ is continuous (with respect to $\mathbf{x}$) and $q(c, \mathbf{x}) \to \infty$ as $|\mathbf{x}| \to \infty$, then $q(c, \mathbf{x})$ has a minimum.
2. Suppose problem (13.1), with f continuous, is approximated by the penalty problem (13.2), and let $\{c_k\}$ be an increasing sequence of positive constants tending to infinity. Define $q(c, \mathbf{x}) = f(\mathbf{x}) + cP(\mathbf{x})$, and fix $\varepsilon > 0$. For each k let $\mathbf{x}_k$ be determined satisfying

$$q(c_k, \mathbf{x}_k) \leqslant [\min_{\mathbf{x}} q(c_k, \mathbf{x})] + \varepsilon.$$

Show that if $\mathbf{x}^*$ is a solution to (13.1), any limit point, $\overline{\mathbf{x}}$, of the sequence $\{\mathbf{x}_k\}$ is feasible and satisfies $f(\overline{\mathbf{x}}) \leqslant f(\mathbf{x}^*) + \varepsilon$.
3. Construct an example problem and a penalty function such that, as $c \to \infty$, the solution to the penalty problem diverges to infinity.
4. *Combined penalty and barrier method*. Consider a problem of the form

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in S \cap T \end{array}$$

and suppose P is a penalty function for S and B is a barrier function for T. Define

$$d(c, \mathbf{x}) = f(\mathbf{x}) + cP(\mathbf{x}) + \frac{1}{c}B(\mathbf{x}).$$

Let $\{c_k\}$ be a sequence $c_k \to \infty$, and for $k = 1, 2, \ldots$ let $\mathbf{x}_k$ be a solution to

$$\text{minimize } d(c_k, \mathbf{x})$$

subject to $\mathbf{x} \in$ interior of T. Assume all functions are continuous, T is compact (and robust), the original problem has a solution $\mathbf{x}^*$, and that $S \cap$ [interior of T] is not empty. Show that

(a) $\underset{k \in \infty}{\text{limit}}\, d(c_k, \mathbf{x}_k) = f(\mathbf{x}^*)$.
(b) $\underset{k \in \infty}{\text{limit}}\, c_k P(\mathbf{x}_k) = 0$.
(c) $\underset{k \in \infty}{\text{limit}}\, \frac{1}{c_k} B(\mathbf{x}_k) = 0$.

5. Prove the Theorem at the end of Sect. 13.2.
6. Find the central path for the problem of minimizing x^2 subject to $x \geqslant 0$.
7. Consider a penalty function for the equality constraints

$$\mathbf{h}(\mathbf{x}) = \mathbf{0}, \ \mathbf{h}(\mathbf{x}) \in E^m,$$

having the form

$$P(\mathbf{x}) = \gamma(\mathbf{h}(\mathbf{x})) = \sum_{i=1}^{m} w(h_i(\mathbf{x})),$$

where w is a function whose derivative w' is analytic and has a zero of order $s \geqslant 1$ at zero.

(a) Show that corresponding to (13.26) we have

$$\mathbf{Q}(c_k, \ \mathbf{x}_k) = L_k(\mathbf{x}_k) + c_k \sum_{i=1}^{m} \{w''(h_i(\mathbf{x}_k))\} \nabla h_i(\mathbf{x}_k)^T \nabla h_i(\mathbf{x}_k).$$

(b) Show that as $c_k \to \infty$, m eigenvalues of $\mathbf{Q}(c_k, \ \mathbf{x}_k)$ have magnitude on the order of $(c_k)^{1/s}$.

8. Corresponding to the problem

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

consider the sequence of unconstrained problems

$$\text{minimize} \quad f(\mathbf{x}) + [g^+(\mathbf{x}) + 1]^k - 1,$$

and suppose $\mathbf{x}_k$ is the solution to the kth problem.

(a) Find an appropriate definition of a Lagrange multiplier λ_k to associate with $\mathbf{x}_k$.
(b) Find the limiting form of the Hessian of the associated objective function, and determine how fast the largest eigenvalues tend to infinity.

9. Repeat Exercise 8 for the sequence of unconstrained problems

$$\text{minimize} \quad f(\mathbf{x}) + [(g(\mathbf{x}) + 1)^+]^k.$$

10. *Morrison 's method*. Suppose the problem

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned} \tag{13.70}$$

has solution $\mathbf{x}^*$. Let M be an optimistic estimate of $f(\mathbf{x}^*)$, that is, $M \leqslant f(\mathbf{x}^*)$. Define $v(M, \ \mathbf{x}) = [f(\mathbf{x}) - M]^2 + |\mathbf{h}(\mathbf{x})|^2$ and define the unconstrained problem

$$\text{minimize} \quad v(M, \ \mathbf{x}). \tag{13.71}$$

Given $M_k \leqslant f(\mathbf{x}^*)$, a solution $\mathbf{x}_{M_k}$ to the corresponding problem (13.71) is found, then M_k is updated through

$$M_{k+1} = M_k + [v(M_k,\ \mathbf{x}_{M_k})]^{1/2} \tag{13.72}$$

and the process repeated.

(a) Show that if $M = f(\mathbf{x}^*)$, a solution to (13.71) is a solution to (13.70).
(b) Show that if $\mathbf{x}_M$ is a solution to (13.71), then $f(\mathbf{x}_M) \leqslant f(\mathbf{x}^*)$.
(c) Show that if $M_k \leqslant f(\mathbf{x}^*)$ then M_{k+1} determined by (13.72) satisfies $M_{k+1} \leqslant f(\mathbf{x}^*)$.
(d) Show that $M_k \to f(\mathbf{x}^*)$.
(e) Find the Hessian of $v(M,\ \mathbf{x})$ (with respect to $\mathbf{x}^*$). Show that, to within a scale factor, it is identical to that associated with the standard penalty function method.

11. Let A be an $m \times n$ matrix of rank m. Prove the matrix identity

$$[\mathbf{I} + \mathbf{A}^T\mathbf{A}]^{-1} = \mathbf{I} - \mathbf{A}^T\ [\mathbf{I} + \mathbf{A}\mathbf{A}^T]^{-1}\mathbf{A}$$

and discuss how it can be used in conjunction with the method of Sect. 13.4.
12. Show that in the limit of large c, a single cycle of the normalization method of Sect. 13.6 is exactly the same as a single cycle of the combined penalty function and gradient projection method of Sect. 13.7.
13. Suppose that at some step k of the combined penalty function and gradient projection method, the $m \times n$ matrix $\nabla \mathbf{h}(\mathbf{x}_k)$ is not of rank m. Show how the method can be continued by temporarily executing the Newton step over a subspace of dimension less than m.
14. For a problem with equality constraints, show that in the combined penalty function and gradient projection method the second step (the steepest descent step) can be replaced by a step in the direction of the negative projected gradient (projected onto M_k) without destroying the global convergence property and without changing the rate of convergence.
15. Develop a method that is analogous to that of Sect. 13.7, but which is a combination of penalty functions and the reduced gradient method. Establish that the rate of convergence of the method is identical to that of the reduced gradient method.
16. Extend the result of the Exact Penalty Theorem of Sect. 13.8 to inequalities. Write $g_j(\mathbf{x}) \leqslant 0$ in the form of an equality as $g_j(\mathbf{x}) + y_j^2 = 0$ and show that the original theorem applies.
17. Develop a result analogous to that of the Exact Penalty Theorem of Sect. 13.8 for the penalty function

$$P(\mathbf{x}) = \max\{0,\ g_i(\mathbf{x}),\ g_2(\mathbf{x}),\ \ldots,\ g_p(\mathbf{x}),\ |h_i(\mathbf{x})|,\ |h_2(\mathbf{x})|,\ \ldots,\ |h_m(\mathbf{x})|\}.$$

18. Solve the problem

$$\begin{aligned} &\text{minimize} && x^2 + xy + y^2 - 2y \\ &\text{subject to} && x + y = 2 \end{aligned}$$

three ways analytically

(a) with the necessary conditions.
(b) with a quadratic penalty function.
(c) with an exact penalty function.

References

13.1 The penalty approach to constrained optimization is generally attributed to Courant [C8]. For more details than presented here, see Butler and Martin [B26] or Zangwill [Z1].

13.2 The barrier method is due to Carroll [C1], but was developed and popularized by Fiacco and McCormick [F4] who proved the general effectiveness of the method. Also see Frisch [F19].

13.3 It has long been known that penalty problems are solved slowly by steepest descent, and the difficulty has been traced to the ill-conditioning of the Hessian. The explicit characterization given here is a generalization of that in Luenberger [L10]. For the geometric interpretation, see Luenberger [L8]. The central path for nonlinear programming was analyzed by Nesterov and Nemirovskii [N2], Jarre [J2] and den Hertog [H6].

13.5 Most previous successful implementations of penalty or barrier methods have employed Newton's method to solve the unconstrained problems and thereby have largely avoided the effects of the ill-conditioned Hessian. See Fiacco and McCormick [F4] for some suggestions. The technique at the end of the section is new.

13.6 This method was first presented in Luenberger [L13].

13.8 See Luenberger [L10], for further analysis of this method.

13.9 The fact that the absolute-value penalty function is exact was discovered by Zangwill [Z1]. The fact that $c > |\lambda|$ is sufficient for exactness was pointed out by Luenberger [L12]. Line search methods have been developed for nonsmooth functions. See Lemarechal and Mifflin [L3].

13.10 For analysis along the lines of Exercise 7, see Lootsma [L7]. For the functions suggested in Exercises 8 and 9, see Levitin and Polyak [L5]. For the method of Exercise 10, see Morrison [M8].