7. Constrained Optimization

7.0 Fundamentals, Definitions and Terminology

The discussion so far has focussed on unconstrained optimization, that is with no restrictive limits being placed on the values to which the variables may be set. In practice, there are often physical constraints which prevent complete optimization of the problem, e.g. dimension limits in certain areas. In mathematical terms the problem becomes:

$$\begin{array}{lll} \text{minimize} & f(\mathbf{x})\,, & \mathbf{x} = (x_1, x_2, \ldots, x_n)^T \\ \text{subject to} & h_i(\mathbf{x}) = 0\,, & i = 1, 2, \ldots, m' \\ & g_i(\mathbf{x}) \le 0\,, & i = m'+1, \ldots, m \end{array} \tag{7.1}$$

Constraints are said to be *active* when they impose a direct restriction on small variations of $\mathbf{x}$. Thus, equality constraints are always active. If $g_i(\mathbf{x}) = 0$, an inequality constraint is active. If $g_i(\mathbf{x}) < 0$, an inequality constraint is *inactive*. An inequality constraint is *violated*, if $g_i(\mathbf{x}) > 0$.

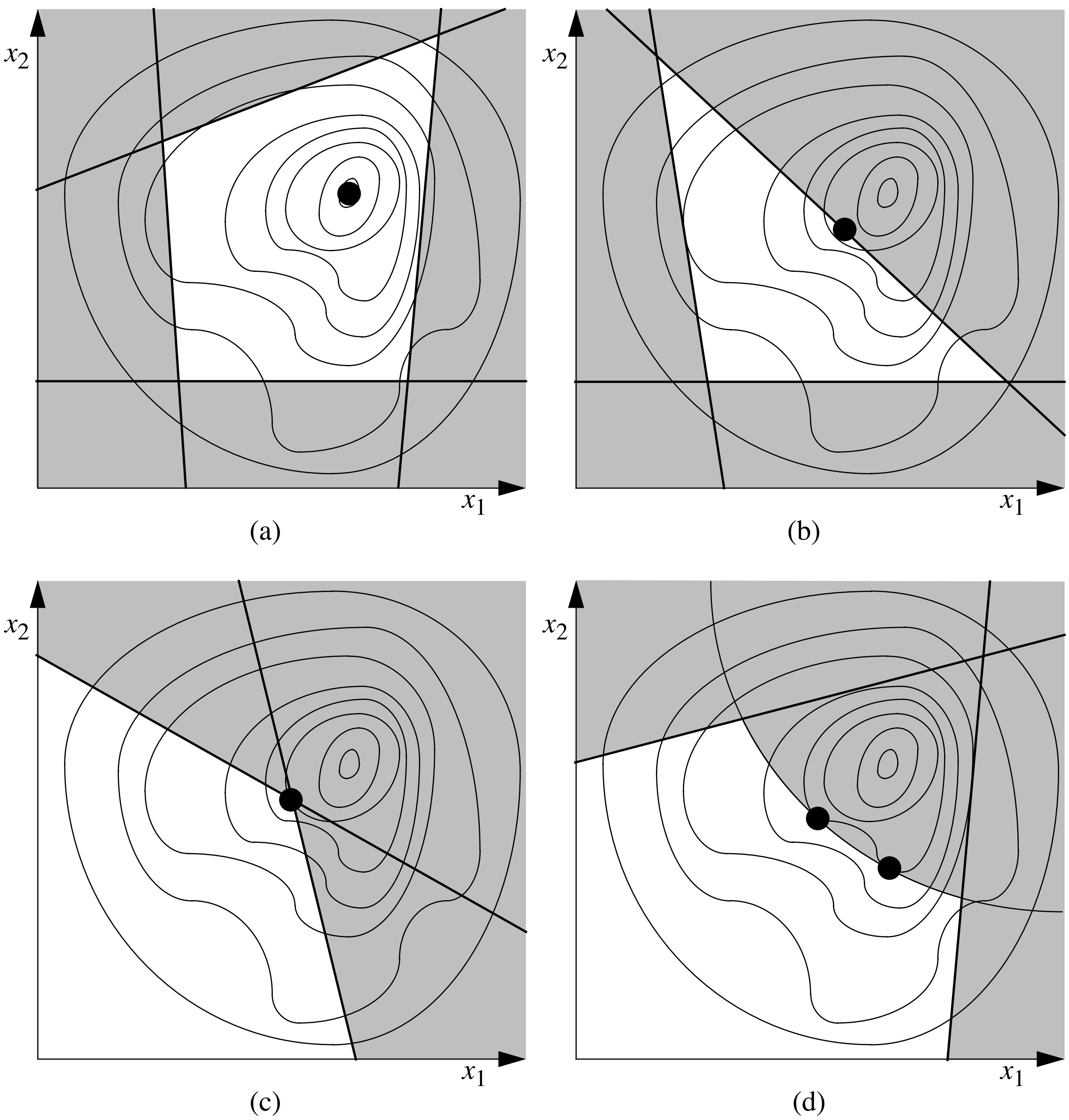
Figure 7.1 illustrates different cases: in (a), none of the constraints are active near the minimum; in (b) and (d), only one of three constraints is active at the minimum; and in (c), two constraints are active at the minimum.

As Figure 7.1 demonstrates, identifying the *feasible region* graphically and superimposing objective function contours on this diagram can help to identify which constraints will be active at the optimum, and, in some cases, it may even be possible to identify where the optimum is by inspection!

Note that if the constraints are independent, it is impossible to have more that n constraints active at once. Also, if it was possible to know the active and inactive constraints at the optimum in advance, then we could ignore the inactive constraints and treat the active ones as equality constraints.

By suitable rearrangement and substitution of variables it is often possible to eliminate equality constraints altogether (by eliminating a variable) and to rearrange inequality constraints into simple limits on a variable. This greatly reduces the chances of a search algorithm getting stuck, as exploratory searches along other variables are then in directions which can't activate constraints.

Figure 7.1: Examples of Minima with Constraints.



7.4 Nonlinear Constrained Optimization

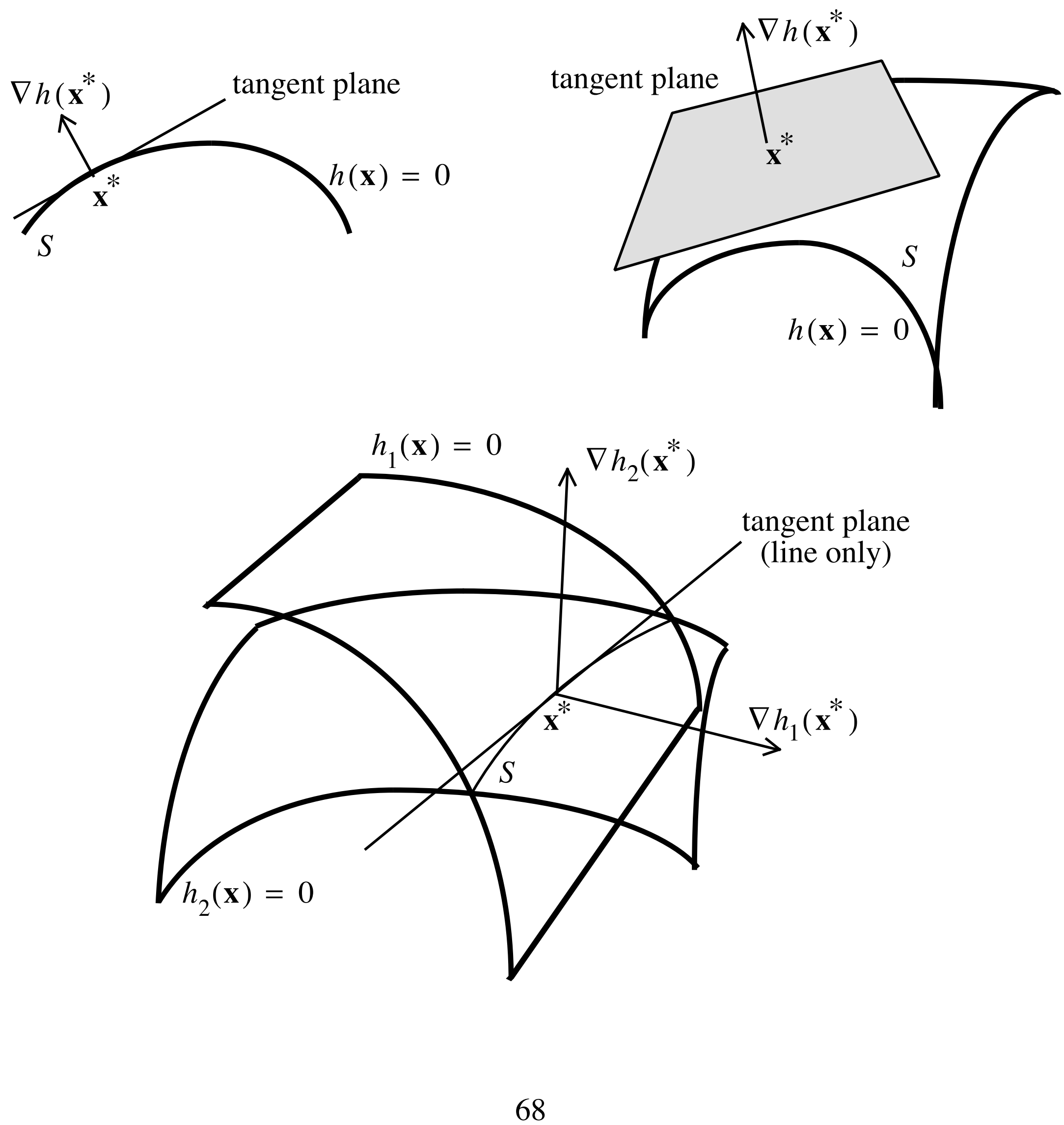
7.4.1 Optimality Conditions

As shown in Section 4.2, for $\mathbf{x}^*$ to be local minimum $\nabla f(\mathbf{x}^*) \cdot \mathbf{d} \geq 0$ for all feasible directions. This means that, if an optimum is to lie on a constraint boundary, the gradient ∇f must be perpendicular to the constraint boundary. If it is not, then there must be a feasible direction along the constraint boundary for which $\nabla f(\mathbf{x}^*) \cdot \mathbf{d} < 0$. (When moving *along* a constraint boundary, if $\mathbf{d}$ is a feasible direction then $-\mathbf{d}$ is also a feasible direction.)

If more than one constraint is active, then the feasible directions at a given point on the constraint boundary lie in the tangent plane of the surface defined by the constraints. Thus, for $\mathbf{x}^*$ to be local minimum, ∇f must be perpendicular to the tangent plane of the surface defined by the constraints. Some examples of tangent planes/lines are shown in Figure 7.16.

An alternative way of stating the same condition is that the gradient of the objective function and the gradients of the active constraints lie in the same plane at any local minimum (or max-

Figure 7.16: Examples of Tangent Planes at $\mathbf{x}^*$.



imum), so that the gradient of the objective function can be written as a linear sum of the gradients of the active constraints:

$$\nabla f = \sum_{i=1}^{m} \alpha_i \nabla h_i = \nabla \mathbf{h}^T \boldsymbol{\alpha} , \tag{7.16}$$

where $\nabla \mathbf{h}$ is the Jacobian matrix:

$$\nabla \mathbf{h}(\mathbf{x}^*) = \begin{bmatrix} \nabla h_1(\mathbf{x}^*)^T \\ \vdots \\ \nabla h_m(\mathbf{x}^*)^T \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_n} \end{bmatrix} \tag{7.17}$$

For simplicity, equation (7.16) has been written in terms of equality constraints $h_i(\mathbf{x})$ only, but, as an active inequality constraint is effectively an equality constraint, there is no loss of generality in so doing.

Equation (7.16) gives us a first-order necessary condition for a local minimum.

7.5 Lagrange and Kuhn-Tucker Multipliers

7.5.1 Lagrange Multipliers

Let the problem be to minimize the objective function $f(\mathbf{x})$ subject to a set of m equality constraints $h_1(\mathbf{x}) = 0, \ldots, h_m(\mathbf{x}) = 0$.

Now define an unconstrained function

$$L(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i h_i(\mathbf{x}) , \tag{7.18}$$

where the λ_i are called *Lagrange multipliers.*

If $\mathbf{x}^*$ is the minimum of *L*, then

$$\frac{\partial L(\mathbf{x}^*, \boldsymbol{\lambda})}{\partial x_1} = \ldots = \frac{\partial L(\mathbf{x}^*, \boldsymbol{\lambda})}{\partial x_n} = 0 \tag{7.19}$$

or

$$\nabla L(\mathbf{x}^*, \boldsymbol{\lambda}) = 0 . \tag{7.20}$$

Thus

$$\nabla f(\mathbf{x}^*) + [\nabla \mathbf{h}(\mathbf{x}^*)]^T \boldsymbol{\lambda} = 0 , \tag{7.21}$$

which can be by rewritten as

$$\nabla f(\mathbf{x}^*) = -[\nabla \mathbf{h}(\mathbf{x}^*)]^T \boldsymbol{\lambda} . \tag{7.22}$$

Hence, the rate of change of the objective function *f* at a local minimum $\mathbf{x}^*$ of *L* is a linear combination of the rates of change of the active constraint equations at $\mathbf{x}^*$, and the Lagrange multipliers are the coefficients of the combinations corresponding to each constraint. That is, at $\mathbf{x}^*$, the gradient of the objective function *f* and the gradients of the constraints lie in the same plane. Thus, a minimum of the unconstrained function *L* is also a minimum of the constrained function *f*. (Comparing equation (7.22) with equation (7.16), it should be clear that they, in effect, impose the same condition on $\mathbf{x}^*$.)

Equation (7.21) is therefore a first-order necessary condition for a local minimum. As in the unconstrained case, a second-order condition is needed to ensure that $\mathbf{x}^*$ is a local minimum.

Differentiating equation (7.21) gives

$$\nabla^2 L(\mathbf{x}^*) = \nabla^2 f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \nabla^2 h_i(\mathbf{x}^*)\,, \qquad (7.23)$$

where ∇^2 is the Hessian operator.

A sufficient second-order condition for $\mathbf{x}^*$ to be a local minimum is that $\nabla^2 L(\mathbf{x}^*)$ be positive definite on the tangent plane *M* at $\mathbf{x}^*$, that is

$$\mathbf{y}^T \nabla^2 L(\mathbf{x}^*)\,\mathbf{y} > 0 \quad \forall\ \mathbf{y} \in M\,. \qquad (7.24)$$

The problem is now to solve the following equations:

$$\begin{aligned} \nabla f(\mathbf{x}^*) + [\nabla \mathbf{h}(\mathbf{x}^*)]^T \boldsymbol{\lambda} &= 0 \quad (n \text{ equations}) \\ \mathbf{h}(\mathbf{x}^*) &= 0 \quad (m \text{ equations}) \end{aligned} \qquad (7.25)$$

So we have a system of $m + n$ equations and $m + n$ unknowns (m Lagrange multipliers λ_i and n control variables x_i).

See Chapter 3 of Gill, Murray and Wright for the proofs of the above statements and a more detailed discussion.

7.5.2 Example: Lagrange Multipliers

Minimize $f(\mathbf{x}) = x_1 x_2^2$ subject to $h(\mathbf{x}) = x_1^2 + x_2^2 - 2 = 0$

$$L = x_1 x_2^2 + \lambda(x_1^2 + x_2^2 - 2)$$

$$\frac{\partial L}{\partial x_1} = x_2^2 + 2x_1\lambda = 0 \qquad (1)$$

$$\frac{\partial L}{\partial x_2} = 2x_1x_2 + 2x_2\lambda = 0 \qquad (2)$$

$$x_1^2 + x_2^2 - 2 = 0 \qquad (3)$$

$$\nabla^2 L = \begin{bmatrix} 2\lambda & 2x_2 \\ 2x_2 & 2x_1 + 2\lambda \end{bmatrix}$$

(2) $\Rightarrow x_2 = 0$ or $x_1 = -\lambda$
(a) (b)

(a) if $x_2 = 0$ (1) $\Rightarrow x_1 = 0$ violates (3)
or $\lambda = 0$

if $x_2 = 0, \lambda = 0$ (3) $\Rightarrow x_1 = \pm\sqrt{2}$

For $x_2 = 0, \lambda = 0$

$$\nabla^2 L = \begin{bmatrix} 0 & 0 \\ 0 & 2x_1 \end{bmatrix}$$

At $x_1 = -\sqrt{2}$

$$y = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ or } \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

$$y^T \nabla^2 L y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & -2\sqrt{2} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$= -2\sqrt{2}$

$\therefore$ not PD

$\therefore$ not a minimum

actually a maximum

For $x_1 = \sqrt{2}$, $y = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ or $\begin{bmatrix} 0 \\ -1 \end{bmatrix}$

$y^T \nabla^2 L y = 2\sqrt{2}$ $\therefore$ a minimum

(b) If $x_1 = -\lambda$

(1) $\Rightarrow x_2^2 = -2x_1\lambda = 2\lambda^2$

$\therefore$ (3) $\Rightarrow \lambda^2 + 2\lambda^2 = 2 \quad \Rightarrow \lambda = \pm\sqrt{\frac{2}{3}}$

$\therefore \quad x_1 = \mp\sqrt{\frac{2}{3}} \qquad x_2 = \pm\sqrt{\frac{4}{3}}$

Note problem is symmetric in x_2

For $\lambda = \sqrt{\frac{2}{3}} \qquad \underline{x} = \left[-\sqrt{\frac{2}{3}}, \pm\sqrt{\frac{4}{3}}\right]$

For $\underline{x} = \left[-\sqrt{\frac{2}{3}}, -\sqrt{\frac{4}{3}}\right] \qquad \underline{y} = \pm\left[-\sqrt{\frac{4}{3}}, \sqrt{\frac{2}{3}}\right]$

$$\nabla^2 L = \begin{bmatrix} 1.63 & -2.31 \\ -2.31 & 0 \end{bmatrix}$$

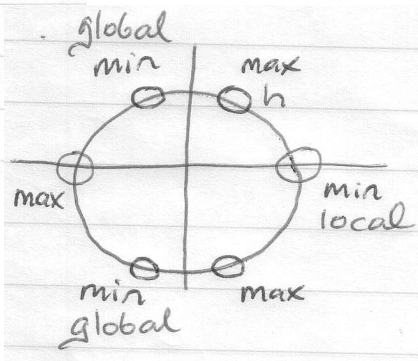
$y^T \nabla^2 L y = 6.53$ +ve $\therefore$ PD

$\therefore$ a minimum

Same for $\underline{x} = \left[-\sqrt{\frac{2}{3}}, \sqrt{\frac{4}{3}}\right]$

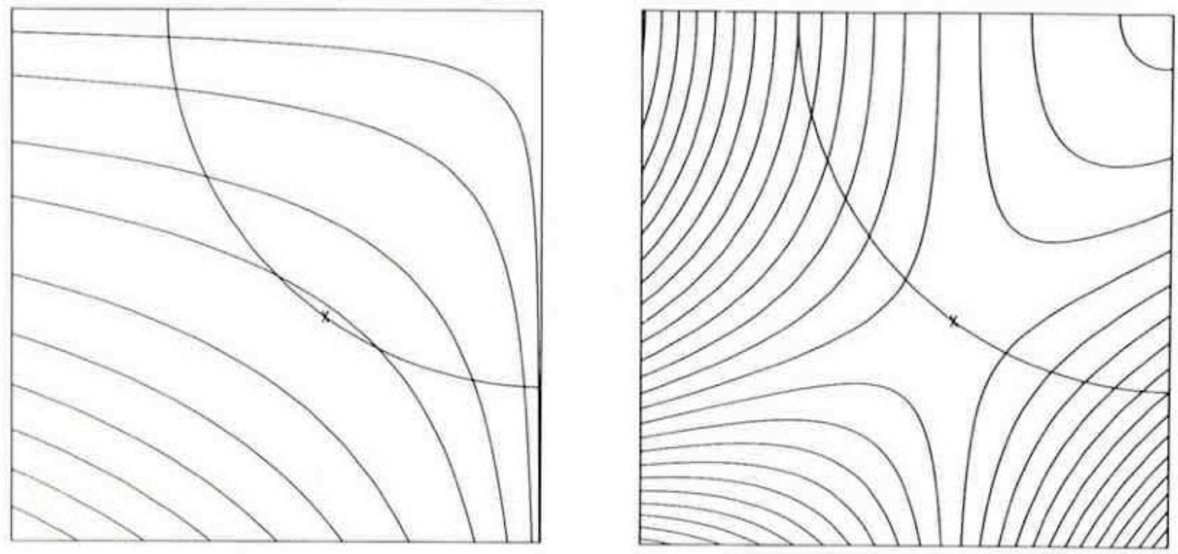
For $\lambda = -\sqrt{\frac{2}{3}}$, $\underline{x} = \left[\sqrt{\frac{2}{3}}, \pm\sqrt{\frac{4}{3}}\right]$

$y^T \nabla^2 L y = -6.53$ $\therefore$ maximum



Example: Lagrange Multipliers (continued)

Figure 7.17: Contours of $f(\mathbf{x})$ and $L(\mathbf{x})$.



The left-hand diagram depicts the contours of $f(\mathbf{x}) = x_1 x_2^2$. The contour line corresponding to the constraint $h(\mathbf{x}) = x_1^2 + x_2^2 - 2 = 0$ is superimposed. The right-hand diagram depicts the contours of the Lagrangian.

7.5.3 Inequality Constraints

As already mentioned in Section 7.3.4, one way of dealing with inequality constraints is to transform them into equality constraints by adding an extra (non-negative) variable, called a *slack variable*, and an extra constraint. That is, since an inequality constraint is defined as $g_i(\mathbf{x}) \le 0$, a positive slack variable s_i transforms the constraint to $\hat{g}_i(\mathbf{x}) = g_i(\mathbf{x}) + s_i = 0$. For this new constraint, the Lagrange multiplier must be greater than or equal to zero. This method is described in more detail in Siddall.

Alternatively, inequality constraints can be dealt with using the method of *Kuhn-Tucker multipliers*.

7.5.4 Kuhn-Tucker Multipliers

Let the problem be to minimize the objective function $f(\mathbf{x})$ subject to a set of m equality constraints $h_1(\mathbf{x}) = 0, \ldots, h_m(\mathbf{x}) = 0$ and p inequality constraints $g_1(\mathbf{x}) \le 0, \ldots, g_p(\mathbf{x}) \le 0$.

Define an unconstrained function

$$L(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i h_i(\mathbf{x}) + \sum_{i=1}^{p} \mu_i g_i(\mathbf{x}) . \tag{7.26}$$

If $\mathbf{x}^*$ is a minimum of $L(\mathbf{x})$, then

$$\nabla f(\mathbf{x}^*) + [\nabla \mathbf{h}(\mathbf{x}^*)]^T \lambda + [\nabla \mathbf{g}(\mathbf{x}^*)]^T \mu = 0 \tag{7.27}$$

and

$$\mu_i g_i(\mathbf{x}) = 0 \quad \forall \quad i = 1, \ldots, p \tag{7.28}$$

where $\mu \geq 0$ are the *Kuhn-Tucker multipliers.*

Equations (7.27) and (7.28) are called the *Kuhn-Tucker conditions.* These are first-order necessary conditions for a minimum.

Note that equation (7.28) is readily satisfied for active constraints ($g_i(\mathbf{x}) = 0$), but for inactive constraints ($g_i(\mathbf{x}) < 0$), it implies that $\mu_i = 0$.

We can see that $\mathbf{x}^*$ satisfying the Kuhn-Tucker conditions is a candidate for a minimum as follows.

For $\varepsilon > 0$ consider a small change in $\mathbf{x}^*$ in a feasible direction $\mathbf{y}$

$$f(\mathbf{x}^* + \varepsilon\mathbf{y}) \approx f(\mathbf{x}^*) + [\nabla f(\mathbf{x}^*)]^T \varepsilon\mathbf{y}$$

$$\therefore \quad \frac{f(\mathbf{x}^* + \varepsilon\mathbf{y}) - f(\mathbf{x}^*)}{\varepsilon} \approx [\nabla f(\mathbf{x}^*)]^T \mathbf{y} = -\lambda^T \nabla \mathbf{h}(\mathbf{x}^*)\mathbf{y} - \mu^T \nabla \mathbf{g}(\mathbf{x}^*)\mathbf{y} \geq 0 \,, \tag{7.29}$$

using the fact that, from equation (7.27),

$$[\nabla f(\mathbf{x}^*)]^T = -\lambda^T \nabla \mathbf{h}(\mathbf{x}^*) - \mu^T \nabla \mathbf{g}(\mathbf{x}^*) \,.$$

- As $\mathbf{x}^*$ is a minimum $\nabla \mathbf{h}(\mathbf{x}^*)\,\mathbf{y} = 0$ (All feasible directions lie in the tangent plane defined by the equality constraints, and, thus, are perpendicular to the gradients of these constraints.)
- If $g_i(\mathbf{x})$ is not active at $\mathbf{x}^*$, then $\mu_i = 0$
- If $g_i(\mathbf{x})$ remains active, then $\nabla g_i(\mathbf{x}^*)\,\mathbf{y} = 0$ — $g_i(\mathbf{x})$ is effectively an equality constraint
- If $g_i(\mathbf{x})$ becomes inactive, then $g_i(\mathbf{x})$ must decrease, so $\nabla \mathbf{g}(\mathbf{x}^*)\,\mathbf{y}$ must be < 0, and, as $\mu_i > 0$, $-\mu^T \nabla \mathbf{g}(\mathbf{x}^*)\,\mathbf{y} > 0$

Hence the change in f is ≥ 0 and hence $\mathbf{x}^*$ is a candidate for a minimum.

Differentiating equation (7.27) gives

$$\nabla^2 L(\mathbf{x}^*) = \nabla^2 f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \nabla^2 h_i(\mathbf{x}^*) + \sum_{i=1}^{p} \mu_i \nabla^2 g_i(\mathbf{x}^*) \,, \tag{7.30}$$

where ∇^2 is the Hessian operator.

A *sufficient second-order condition* for $\mathbf{x}^*$ to be a local minimum is that $\nabla^2 L(\mathbf{x}^*)$ be positive definite on the tangent plane M of the active constraints at $\mathbf{x}^*$, i.e.

$$\mathbf{y}^T \nabla^2 L(\mathbf{x}^*)\,\mathbf{y} > 0 \quad \forall\ \mathbf{y} \in M\ . \tag{7.31}$$

If there are no equality constraints, i.e. $m = 0$, then the first-order conditions

$$\nabla f(\mathbf{x}^*) + [\nabla \mathbf{g}(\mathbf{x}^*)]^T \mu = 0 \tag{7.32}$$

and

$$\mu_i \geq 0\ \forall\ \ i = 1, \ldots, p \tag{7.33}$$

are sufficient to identify a local minimum $\mathbf{x}^*$.

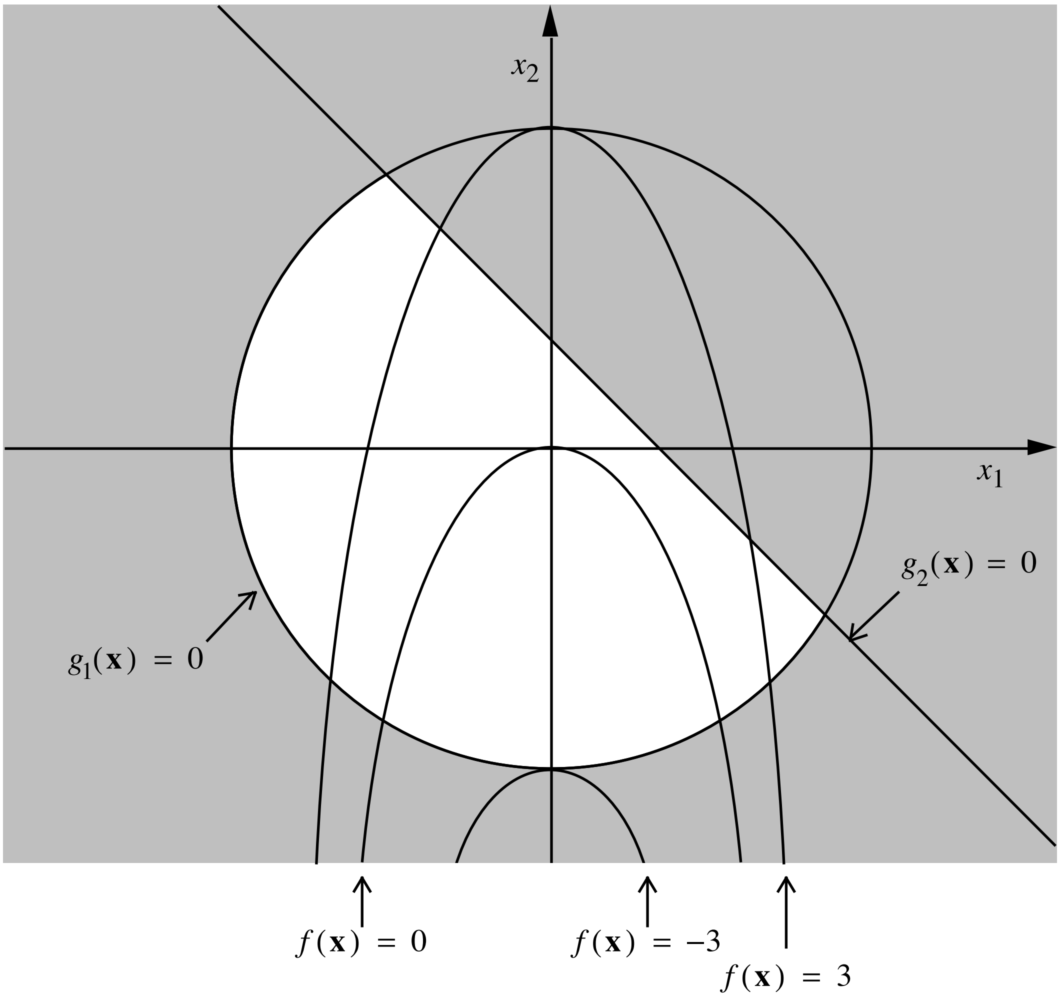
The problem is now to solve the following equations:

$$\begin{aligned} \nabla f(\mathbf{x}^*) + [\nabla \mathbf{h}(\mathbf{x}^*)]^T \lambda\ + [\nabla \mathbf{g}(\mathbf{x}^*)]^T \boldsymbol{\mu} &= 0 \quad (n \text{ equations}) \\ \mathbf{h}(\mathbf{x}^*) &= 0 \quad (m \text{ equations}) \\ \forall\ i = 1, \ldots, p, \quad \mu_i g_i(\mathbf{x}) &= 0 \quad (p \text{ equations}) \end{aligned} \tag{7.34}$$

So we have a system of $m + n + p$ equations and $m + n + p$ unknowns. These equations can be solved by systematically (or otherwise) checking for active inequality constraints, as illustrated in the example which follows.

7.5.5 Example: Kuhn-Tucker Multipliers

Minimize $f(\mathbf{x}) = x_1^2 + x_2$ subject to $g_1(\mathbf{x}) = x_1^2 + x_2^2 - 9 \le 0$ and $g_2(\mathbf{x}) = x_1 + x_2 - 1 \le 0$



$$L = x_1^2 + x_2 + \mu_1(x_1^2 + x_2^2 - 9) + \mu_2(x_1 + x_2 - 1)$$

$$2x_1 + \mu_1 2x_1 + \mu_2 = 0 \quad (1)$$

$$1 + \mu_1 2x_2 + \mu_2 = 0 \quad (2)$$

$$\mu_1(x_1^2 + x_2^2 - 9) = 0 \quad (3)$$

$$\mu_2(x_1 + x_2 - 1) = 0 \quad (4)$$

1. $\mu_1 = 0 \qquad \mu_2 = 0$

(2) $\Rightarrow 1 = 0 \quad \therefore$ impossible

2. $\mu_1 = 0 \qquad \mu_2 > 0$

(2) $\Rightarrow \mu_2 = -1 \quad \therefore$ not a minimum

$\therefore$ stop $\Rightarrow$ $x_1 = 0.5$, $x_2 = 0.5$

Example: Kuhn-Tucker Multipliers (continued)

3. $\mu_2 = 0 \quad \mu_1 > 0 \quad 2x_1 + \mu_1 2x_1 = 0$

① $\Rightarrow \mu_1 = -1$ $\therefore$ not a minimum

or $x_1 = 0$

③ $\Rightarrow x_2^2 = 9 \Rightarrow x_2 = \pm 3$

② $\Rightarrow \mu_1 = -\dfrac{1}{2x_2}$

$x_2 = 3 \Rightarrow \mu_1 = \dfrac{-1}{6}$ not a min. also violates g_2

$x_2 = -3 \Rightarrow \mu_1 = \dfrac{1}{6}$ $\underline{\underline{\text{a min!}}}$

4. $\mu_1 > 0 \quad \mu_2 > 0$

③ $\Rightarrow x_1^2 + x_2^2 - 9 = 0$

④ $\Rightarrow x_1 + x_2 - 1 = 0$

$\therefore (1-x_2)^2 + x_2^2 - 9 = 0$

$\therefore 1 - 2x_2 + x_2^2 + x_2^2 - 9 = 0$

$\therefore x_2^2 - x_2 - 4 = 0$

$\therefore x_2 = \dfrac{1 \pm \sqrt{1+16}}{2} = \dfrac{1 \pm \sqrt{17}}{2} = 2.56$ or -1.56

$\therefore x_1 = 1 - x_2 = -1.56$ or 2.56

① - ② $\Rightarrow 2x_1 - 1 + \mu_1 2(x_1 - x_2) = 0$

$\therefore \mu_1 = \dfrac{2x_1 - 1}{2(x_2 - x_1)}$

For $x_1 = -1.56 \quad x_2 = 2.56 \quad \mu_1 = -0.5$

$x_1 = 2.56 \quad x_2 = -1.56 \quad \mu_1 = -0.5$

$\therefore$ not a minimum

7.5.6 Sensitivity of Constraints

It is often desirable to have information about the *cost of the constraints* in a problem or the *sensitivity* of the solution to variations in the constraints. The Lagrange and Kuhn-Tucker multipliers associated with the constrained problem can provide this information.

Equality Constraints

Consider the family of problems

$$\text{Minimize } f(\mathbf{x}) \text{ subject to } \mathbf{h}(\mathbf{x}) = \mathbf{c} \tag{7.35}$$

and suppose that there are local solutions $\mathbf{x}(\mathbf{c})$ varying smoothly with $\mathbf{c}$, with $\mathbf{x}(0) = \mathbf{x}^*$. Then it can be shown that

$$\nabla_{\mathbf{c}} f(\mathbf{x}(\mathbf{c}))\Big|_{\mathbf{c}=0} = -\lambda \tag{7.36}$$

where λ is the Lagrange multiplier vector associated with $\mathbf{x}^*$.

Inequality & Equality Constraints

Consider now the family of problems

$$\text{Minimize } f(\mathbf{x}) \text{ subject to } \mathbf{h}(\mathbf{x}) = \mathbf{c} \text{ and } \mathbf{g}(\mathbf{x}) \le \mathbf{d} \tag{7.37}$$

and suppose that there are local solutions $\mathbf{x}(\mathbf{c}, \mathbf{d})$ varying smoothly with $\mathbf{c}$ and $\mathbf{d}$, with $\mathbf{x}(0, 0) = \mathbf{x}^*$, and $\mu_i > 0$ for all active constraints, then it can be shown that

$$\begin{aligned} \nabla_{\mathbf{c}} f(\mathbf{x}(\mathbf{c}, \mathbf{d}))\Big|_{\mathbf{c},\mathbf{d}=0,0} &= -\boldsymbol{\lambda} \\ \nabla_{\mathbf{d}} f(\mathbf{x}(\mathbf{c}, \mathbf{d}))\Big|_{\mathbf{c},\mathbf{d}=0,0} &= -\mu \end{aligned} \tag{7.38}$$

where μ is the Kuhn-Tucker multiplier vector associated with $\mathbf{x}^*$.

Thus Lagrange and Kuhn-Tucker multipliers provide a relative measure of the sensitivity of $f(\mathbf{x}^*)$ to changes in the constraints. For example, if in a problem we have $\lambda_1 = 10^3$ and $\lambda_2 = 10^{-3}$, then changes in c_1 will tend to have a much larger effect on the value of f than changes in c_2.

7.6 Penalty Functions

Another approach for constrained problems is to use a *penalty function* which is added to the objective function, and 'penalises' points which lie outside the feasible region.

The advantage of this approach is that it essentially turns a constrained problem into an unconstrained one. The disadvantage is that a mild penalty function can still allow the search to leave the feasible region, while a severe penalty function can create awkward 'valleys' where a search algorithm can get stuck, or produce ill-conditioned gradients and Hessians for curve-fitting methods. (Ill-conditioning means that small perturbations in the input can lead to enormous changes in the output.)

The commonest penalty function is

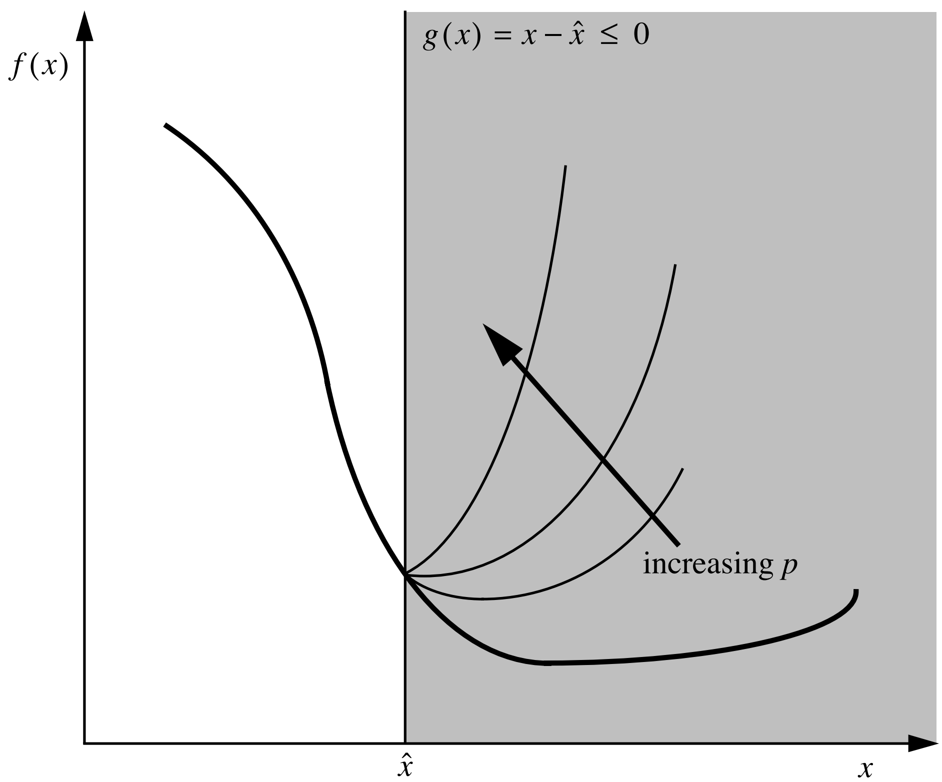
$$q(\mathbf{x}, p) = f(\mathbf{x}) + p\sum_{i=1}^{m}(\max[0,\ g_i(\mathbf{x})])^2\ , \tag{7.39}$$

where m is the number of constraints and p is called the *penalty parameter*.

Note that equality constraints can be incorporated into the function by writing them as two inequalities:

$$h_i(\mathbf{x}) = 0 \ \Rightarrow \ \begin{array}{l} h_i(\mathbf{x}) \le 0 \\ -h_i(\mathbf{x}) \le 0 \end{array}$$

Figure 7.18: An Example of a Penalty Function.



A typical approach is to minimize $q(\mathbf{x}, p)$ repeatedly for increasingly large values of the penalty parameter, using the final point of each search as the starting point of the next until a feasible optimum is obtained. This approach hopefully avoids both the pitfalls mentioned above.

If the initial value of the penalty parameter is too large, then even a robust unconstrained algorithm will typically experience great difficulty in finding the minimum.

Method:

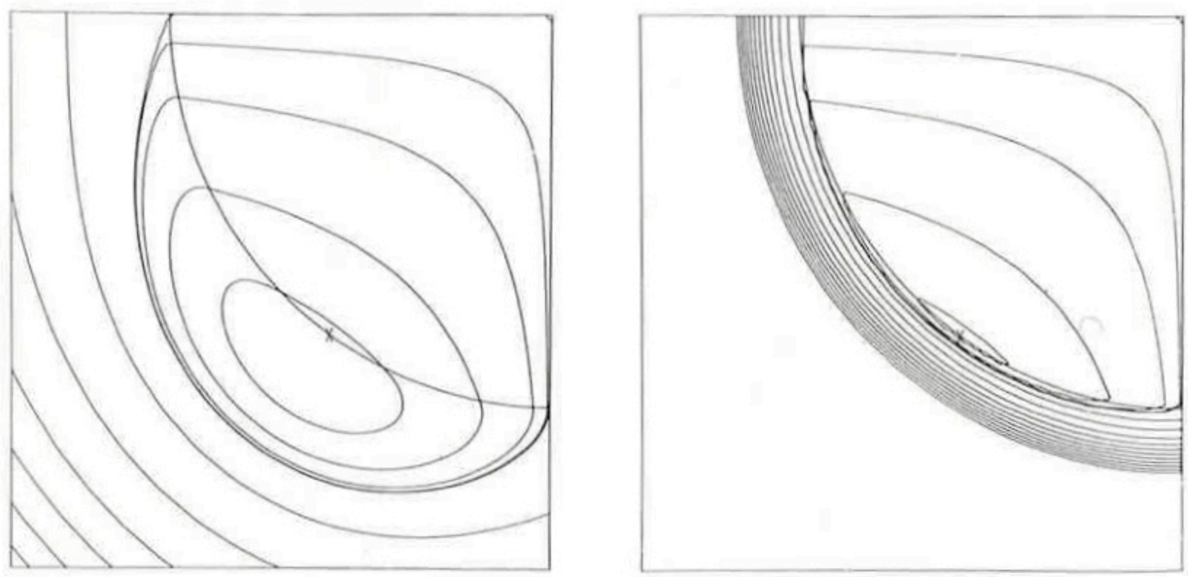
1. Select an initial (small) value for p.
2. Minimize $q(\mathbf{x}, p)$ using a suitable unconstrained optimization algorithm.
3. Increase p and repeat the minimization starting with the final point of 2 until convergence.

Since penalty functions can generate infeasible iterates, they are not appropriate for problems in which feasibility must be maintained. A class of *feasible-point* methods, in which only feasible iterates are generated, is called *barrier function* methods.

7.6.1 Example: Penalty Functions

Minimize $f(\mathbf{x}) = x_1 x_2^2$ subject to $g(\mathbf{x}) = x_1^2 + x_2^2 - 2 \le 0$

Figure 7.19: Contours of $q(\mathbf{x}, p)$.



The two diagrams depict the contours of $q(\mathbf{x}, p)$ for $p = 1$ (left-hand diagram) and $p = 100$ (right-hand diagram).

7.7 Barrier Functions

In order to preserve feasibility a 'barrier' preventing iterates from becoming infeasible is created by adding to the objective function a weighted sum of continuous functions with a positive singularity at the boundary of the feasible region. (This is in contrast to a penalty function which merely adds a penalty for infeasibility).

A barrier function is infinite along the edge of the feasible region. This is fine provided the region is not *disjoint*, that is, any part of the boundary can be approached from within the region. It also means that a barrier function cannot be applied to an equality constraint.

A common form of barrier function is the *inverse barrier function*

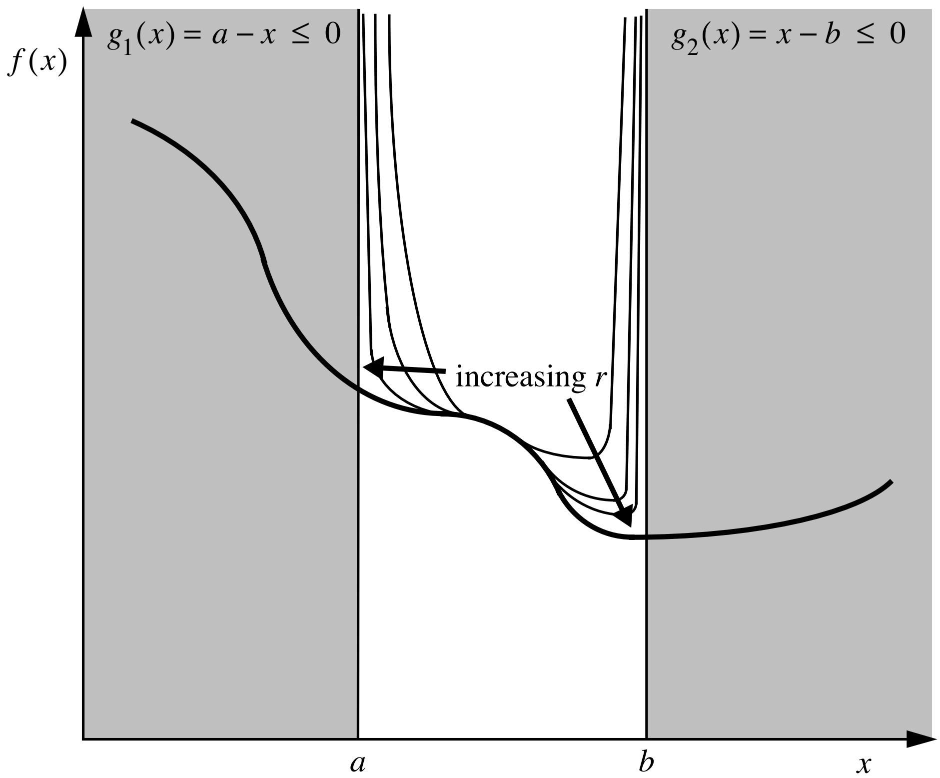
$$b(\mathbf{x}, r) = f(\mathbf{x}) - \frac{1}{r}\sum_{i=1}^{m} \frac{1}{g_i(\mathbf{x})}\ , \tag{7.40}$$

where r is termed the *barrier parameter.*

Another common form is the *logarithmic barrier function*

$$b(\mathbf{x}, r) = f(\mathbf{x}) - \frac{1}{r}\sum_{i=1}^{m} \ln\left[-g_i(\mathbf{x})\right]\ . \tag{7.41}$$

Figure 7.20: An Example of a Barrier Function with Two Constraints.



These barrier functions always yields infinity on the boundary, but conform increasingly closely to the original function within the feasible region for increasing values of the barrier parameter r. Again, it is necessary to increase the barrier parameter gradually, and care must be taken to use small step sizes when r is large, or the search algorithm might leap clear over the boundary in “a single bound”.

Method:

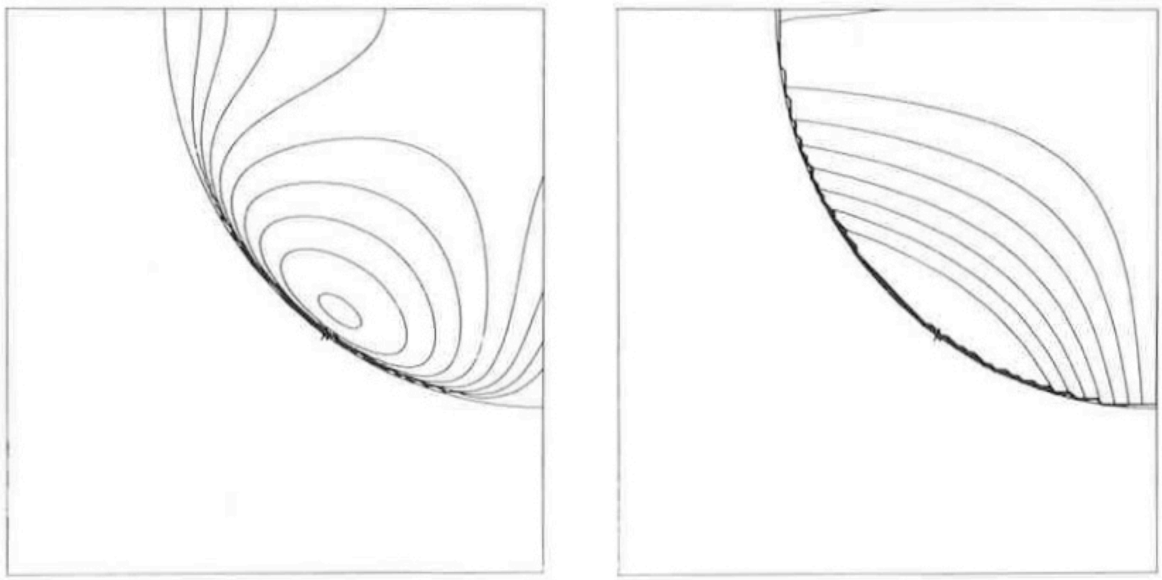
1. Select an initial (small) value for r and a starting point in the feasible region.
2. Minimize $b(\mathbf{x}, r)$ using a suitable unconstrained optimization algorithm.
3. Increase r and repeat the minimization starting with the final point of 2 until convergence.

One of the practical advantages of the barrier function method is that the optimization process will yield a feasible answer even if it is terminated prematurely.

7.7.1 Example: Barrier Functions

Minimize $f(\mathbf{x}) = x_1 x_2^2$ subject to $g(\mathbf{x}) = x_1^2 + x_2^2 - 2 \le 0$

Figure 7.21: Contours of $b(\mathbf{x}, r)$.



The two diagrams represent the contours of the *logarithmic* barrier function $b(\mathbf{x}, r)$ for $r = 5$ (left-hand diagram) and $r = 1000$ (right-hand diagram).

The selection of an appropriate unconstrained optimization method to use in conjunction with penalty and barrier functions depends a great deal on the type of problem to be solved.

Table 7.1: Summary Comparison of Penalty vs Barrier Functions.

Penalty Functions	Barrier Functions
Handles equalities or inequalities	Handles inequalities only
Arbitrary starting point	Starting point in feasible region
Can iterate through infeasible region	Only feasible iterates
Can get stuck in infeasible region	Always yields feasible answer

7.8 Other Constrained Minimization Methods

There are other common constrained optimization methods which will not be discussed in this course but are worth a brief mention. Two of these are the techniques of quadratic programming and augmented Lagrangian.

Quadratic programming is based on directly solving the Lagrange first-order necessary conditions for quadratic functions. It is also the basis for some more general nonlinear programming algorithms. A good discussion on quadratic programming can be found in Luenberger, Chapter 14.

An effective general class of nonlinear programming methods is the class of *augmented Lagrangian* methods. These methods can be viewed as a combination of the penalty function and Lagrange multiplier methods. The two concepts work together to eliminate many of the disadvantages associated with either method alone.

The augmented Lagrangian for the equality constrained problem

$$\text{Minimize } f(\mathbf{x}) \text{ subject to } \mathbf{h}(\mathbf{x}) = 0 \tag{7.42}$$

is the function

$$L(\mathbf{x},\boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T\mathbf{h}(\mathbf{x}) + \tfrac{1}{2}c|\mathbf{h}(\mathbf{x})|^2 \tag{7.43}$$

for some positive constant c.