Previous chapters have focused on unconstrained problems where the domain of each design variable is the space of real numbers. Many problems are constrained, which forces design points to satisfy certain conditions. This chapter presents a variety of approaches for transforming problems with constraints into problems without constraints, thereby permitting the use of the optimization algorithms we have already discussed. Analytical methods are also discussed, including the concepts of duality and the necessary conditions for optimality under constrained optimization.

10.1 Constrained Optimization

Recall the core optimization problem equation (1.1):

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad x \in \mathcal{X}
\end{align*}
\]

In unconstrained problems, the feasible set \( \mathcal{X} \) is \( \mathbb{R}^n \). In constrained problems, the feasible set is some subset thereof.

Some constraints are simply upper or lower bounds on the design variables, as we have seen in bracketed line search, in which \( x \) must lie between \( a \) and \( b \). A bracketing constraint \( x \in [a, b] \) can be replaced by two inequality constraints: \( a \leq x \) and \( x \leq b \) as shown in figure 10.1. In multivariate problems, bracketing the input variables forces them to lie within a hyperrectangle as shown in figure 10.2.
Constraints arise naturally when formulating real problems. A hedge fund manager cannot sell more stock than they have, an airplane cannot have wings with zero thickness, and the number of hours you spend per week on your homework cannot exceed 168. We include constraints in such problems to prevent the optimization algorithm from suggesting an infeasible solution.

Applying constraints to a problem can affect the solution, but this need not be the case as shown in figure 10.3.

10.2 Constraint Types

Constraints are not typically specified directly through a known feasible set $\mathcal{X}$. Instead, the feasible set is typically formed from two types of constraints:

1. equality constraints, $h(x) = 0$
2. inequality constraints, $g(x) \leq 0$

© 2019 Massachusetts Institute of Technology, shared under a Creative Commons CC-BY-NC-ND license.
Any optimization problem can be rewritten using these constraints:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) = 0 \text{ for all } i \in \{1, \ldots, \ell\} \\
& \quad g_j(x) \leq 0 \text{ for all } j \in \{1, \ldots, m\}
\end{align*}
\] (10.2)

Of course, constraints can be constructed from a feasible set \( \mathcal{X} \):

\[
h(x) = (x \notin \mathcal{X})
\] (10.3)

where Boolean expressions evaluate to 0 or 1.

We often use equality and inequality functions \((h(x) = 0, g(x) \leq 0)\) to define constraints rather than set membership \((x \in \mathcal{X})\) because the functions can provide information about how far a given point is from being feasible. This information helps drive solution methods toward feasibility.

Equality constraints are sometimes decomposed into two inequality constraints:

\[
h(x) = 0 \iff \begin{cases} h(x) \leq 0 \\ h(x) \geq 0 \end{cases}
\] (10.4)

However, sometimes we want to handle equality constraints separately, as we will discuss later in this chapter.

### 10.3 Transformations to Remove Constraints

In some cases, it may be possible to transform a problem so that constraints can be removed. For example, bound constraints \(a \leq x \leq b\) can be removed by passing \(x\) through a transform (figure 10.4):

\[
x = t_{a,b}(\hat{x}) = \frac{b + a}{2} + \frac{b - a}{2} \left( \frac{2\hat{x}}{1 + \hat{x}^2} \right)
\] (10.5)

Example 10.1 demonstrates this process.

Some equality constraints can be used to solve for \(x_n\) given \(x_1, \ldots, x_{n-1}\). In other words, if we know the first \(n - 1\) components of \(x\), we can use the constraint equation to obtain \(x_n\). In such cases, the optimization problem can be reformulated over \(x_1, \ldots, x_{n-1}\) instead, removing the constraint and removing one design variable. Example 10.2 demonstrates this process.

© 2019 Massachusetts Institute of Technology, shared under a Creative Commons CC-BY-NC-ND license.

2022-05-22 00:25:57-07:00, revision 47fd495, comments to bugs@algorithmsbook.com
Consider the optimization problem
\[
\begin{align*}
\text{minimize} & \quad x \sin(x) \\
\text{subject to} & \quad 2 \leq x \leq 6
\end{align*}
\]

We can transform the problem to remove the constraints:
\[
\begin{align*}
\text{minimize} & \quad t_{2,6}(\hat{x}) \sin(t_{2,6}(\hat{x})) \\
\text{minimize} & \quad \left( 4 + 2 \left( \frac{2\hat{x}}{1 + \hat{x}^2} \right) \right) \sin\left( 4 + 2 \left( \frac{2\hat{x}}{1 + \hat{x}^2} \right) \right)
\end{align*}
\]

We can use the optimization method of our choice to solve the unconstrained problem. In doing so, we find two minima: \( \hat{x} \approx 0.242 \) and \( \hat{x} \approx 4.139 \), both of which have a function value of approximately \(-4.814\).

The solution for the original problem is obtained by passing \( \hat{x} \) through the transform. Both values of \( \hat{x} \) produce \( x = t_{2,6}(\hat{x}) \approx 4.914 \).
Consider the constraint:

\[ h(x) = c_1x_1 + c_2x_2 + \cdots + c_nx_n = 0 \]

We can solve for \( x_n \) using the first \( n - 1 \) variables:

\[ x_n = \frac{1}{c_n}(-c_1x_1 - c_2x_2 - \cdots - c_{n-1}x_{n-1}) \]

We can transform

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h(x) = 0
\end{align*}
\]

into

\[
\min_{x_1, \ldots, x_{n-1}} f\left(\left[x_1, \ldots, x_{n-1}, \frac{1}{c_n}(-c_1x_1 - c_2x_2 - \cdots - c_{n-1}x_{n-1})\right]\right)
\]

### 10.4 Lagrange Multipliers

The method of Lagrange multipliers is used to optimize a function subject to equality constraints. Consider an optimization problem with a single equality constraint:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h(x) = 0
\end{align*}
\]

(10.6)

where \( f \) and \( h \) have continuous partial derivatives. Example 10.3 discusses such a problem.

The method of Lagrange multipliers is used to compute where a contour line of \( f \) is aligned with the contour line of \( h(x) = 0 \). Since the gradient of a function at a point is perpendicular to the contour line of that function through that point, we know the gradient of \( h \) will be perpendicular to the contour line \( h(x) = 0 \). Hence, we need to find where the gradient of \( f \) and the gradient of \( h \) are aligned.

We seek the best \( x \) such that the constraint

\[ h(x) = 0 \]

(10.7)

is satisfied and the gradients are aligned

\[ \nabla f(x) = \lambda \nabla h(x) \]

(10.8)
Consider the minimization problem:

\[
\begin{align*}
\text{minimize} \quad & - \exp \left( - \left( x_1 x_2 - \frac{3}{2} \right)^2 - \left( x_2 - \frac{3}{2} \right)^2 \right) \\
\text{subject to} \quad & x_1 - x_2^2 = 0
\end{align*}
\]

We substitute the constraint \( x_1 = x_2^2 \) into the objective function to obtain an unconstrained objective:

\[
f_{\text{unc}} = - \exp \left( - \left( x_2^3 - \frac{3}{2} \right)^2 - \left( x_2 - \frac{3}{2} \right)^2 \right)
\]

whose derivative is:

\[
\frac{\partial}{\partial x_2} f_{\text{unc}} = 6 \exp \left( - \left( x_2^3 - \frac{3}{2} \right)^2 - \left( x_2 - \frac{3}{2} \right)^2 \right) \left( x_2^5 - \frac{3}{2} x_2^2 + \frac{1}{3} x_2 - \frac{1}{2} \right)
\]

Setting the derivative to zero and solving for \( x_2 \) yields \( x_2 \approx 1.165 \). The solution to the original optimization problem is thus \( x^* \approx [1.358, 1.165] \). The optimum lies where the contour line of \( f \) is aligned with \( h \).

If the point \( x^* \) optimizes \( f \) along \( h \), then its directional derivative at \( x^* \) along \( h \) must be zero. That is, small shifts of \( x^* \) along \( h \) cannot result in an improvement.

The contour lines of \( f \) are lines of constant \( f \). Thus, if a contour line of \( f \) is tangent to \( h \), then the directional derivative of \( h \) at that point, along the direction of the contour \( h(x) = 0 \), must be zero.
for some Lagrange multiplier $\lambda$. We need the scalar $\lambda$ because the magnitudes of the gradients may not be the same.\(^2\)

We can formulate the Lagrangian, which is a function of the design variables and the multiplier

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda h(\mathbf{x})$$

(10.9)

Solving $\nabla \mathcal{L}(\mathbf{x}, \lambda) = 0$ solves equations (10.7) and (10.8). Specifically, $\nabla_x \mathcal{L} = 0$ gives us the condition $\nabla f = \lambda \nabla h$, and $\nabla_\lambda \mathcal{L} = 0$ gives us $h(\mathbf{x}) = 0$. Any solution is considered a critical point. Critical points can be local minima, global minima, or saddle points.\(^3\) Example 10.4 demonstrates this approach.

We can use the method of Lagrange multipliers to solve the problem in example 10.3. We form the Lagrangian

$$\mathcal{L}(x_1, x_2, \lambda) = -\exp\left(-\left(x_1 x_2 - \frac{3}{2}\right)^2 - \left(x_2 - \frac{3}{2}\right)^2\right) - \lambda (x_1 - x_2^2)$$

and compute the gradient

$$\frac{\partial \mathcal{L}}{\partial x_1} = 2x_2 f(x) \left(\frac{3}{2} - x_1 x_2\right) - \lambda$$

$$\frac{\partial \mathcal{L}}{\partial x_2} = 2\lambda x_2 + f(x) \left(-2x_1 (x_1 x_2 - \frac{3}{2}) - 2(x_2 - \frac{3}{2})\right)$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = x_2^2 - x_1$$

Setting these derivatives to zero and solving yields $x_1 \approx 1.358$, $x_2 \approx 1.165$, and $\lambda \approx 0.170$.

The method of Lagrange multipliers can be extended to multiple equality constraints. Consider a problem with two equality constraints:

$$\begin{align*}
\text{minimize} & \quad f(\mathbf{x}) \\
\text{subject to} & \quad h_1(\mathbf{x}) = 0 \\
& \quad h_2(\mathbf{x}) = 0
\end{align*}$$

(10.10)
We can collapse these constraints into a single constraint. The new constraint is satisfied by exactly the same points as before, so the solution is unchanged.

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_{\text{comb}}(x) = h_1(x)^2 + h_2(x)^2 = 0
\end{align*}
\] (10.11)

We can now apply the method of Lagrange multipliers as we did before. In particular, we compute the gradient condition

\[
\nabla f - \lambda \nabla h_{\text{comb}} = 0 \\
\nabla f - 2\lambda (h_1 \nabla h_1 + h_2 \nabla h_2) = 0
\] (10.12, 10.13)

Our choice for \( h_{\text{comb}} \) was somewhat arbitrary. We could have used

\[
h_{\text{comb}}(x) = h_1(x)^2 + c \cdot h_2(x)^2
\] (10.14)

for some constant \( c > 0 \).

With this more general formulation, we get

\[
0 = \nabla f - \lambda \nabla h_{\text{comb}} \\
= \nabla f - 2\lambda h_1 \nabla h_1 - 2c\lambda h_2 \nabla h_2 \\
= \nabla f - \lambda_1 \nabla h_1 - \lambda_2 \nabla h_2
\] (10.15, 10.16, 10.17)

We can thus define a Lagrangian with \( \ell \) Lagrange multipliers for problems with \( \ell \) equality constraints

\[
\mathcal{L}(x, \lambda) = f(x) - \sum_{i=1}^{\ell} \lambda_i h_i(x) = f(x) - \lambda^T \mathbf{h}(x)
\] (10.18)

### 10.5 Inequality Constraints

Consider a problem with a single inequality constraint:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g(x) \leq 0
\end{align*}
\] (10.19)

We know that if the solution lies at the constraint boundary, then the Lagrange condition holds

\[
\nabla f + \mu \nabla g = 0
\] (10.20)
for some non-negative constant $\mu$. When this occurs, the constraint is considered *active*, and the gradient of the objective function is limited exactly as it was with equality constraints. Figure 10.5 shows an example.

If the solution to the problem does not lie at the constraint boundary, then the constraint is considered *inactive*. Solutions of $f$ will simply lie where the gradient of $f$ is zero, as with unconstrained optimization. In this case, equation (10.20) will hold by setting $\mu$ to zero. Figure 10.6 shows an example.

We could optimize a problem with an inequality constraint by introducing an infinite step penalty for infeasible points:

$$f_{\infty\text{-step}}(x) = \begin{cases} f(x) & \text{if } g(x) \leq 0 \\ \infty & \text{otherwise} \end{cases} \quad (10.21)$$

$$= f(x) + \infty \cdot (g(x) > 0) \quad (10.22)$$

Unfortunately, $f_{\infty\text{-step}}$ is inconvenient to optimize.\(^4\) It is discontinuous and nondifferentiable. Search routines obtain no directional information to steer themselves toward feasibility.

We can instead use a linear penalty $\mu g(x)$, which forms a lower bound on $\infty \cdot (g(x) > 0)$ and penalizes the objective toward feasibility as long as $\mu > 0$. This linear penalty is visualized in figure 10.7.

We can use this linear penalty to construct a Lagrangian for inequality constraints

$$L(x, \mu) = f(x) + \mu g(x) \quad (10.23)$$

We can recover $f_{\infty\text{-step}}$ by maximizing with respect to $\mu$:

$$f_{\infty\text{-step}} = \maximize_{\mu \geq 0} L(x, \mu) \quad (10.24)$$

For any infeasible $x$ we get infinity and for any feasible $x$ we get $f(x)$.

The new optimization problem is thus

$$\minimize_x \maximize_{\mu \geq 0} L(x, \mu) \quad (10.25)$$

This reformulation is known as the *primal* problem.
Optimizing the primal problem requires finding critical points $x^*$ such that:

1. $g(x^*) \leq 0$
   The point is feasible.

2. $\mu \geq 0$
   The penalty must point in the right direction. This requirement is sometimes called *dual feasibility*.

3. $\mu g(x^*) = 0$
   A feasible point on the boundary will have $g(x) = 0$, whereas a feasible point with $g(x) < 0$ will have $\mu = 0$ to recover $f(x^*)$ from the Lagrangian.

4. $\nabla f(x^*) + \mu \nabla g(x^*) = 0$
   When the constraint is active, we require that the contour lines of $f$ and $g$ be aligned, which is equivalent to saying that their gradients be aligned. When the constraint is inactive, our optimum will have $\nabla f(x^*) = 0$ and $\mu = 0$.

These four requirements can be generalized to optimization problems with any number of equality and inequality constraints:

$$\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g(x) \leq 0 \\
& \quad h(x) = 0
\end{align*}$$

(10.26)

where each component of $g$ is an inequality constraint and each component of $h$ is an equality constraint. The four conditions are called the *KKT conditions*.

1. **Feasibility**: The constraints are all satisfied.
   $$g(x^*) \leq 0$$
   $$h(x^*) = 0$$
   (10.27) (10.28)

2. **Dual feasibility**: Penalization is toward feasibility.
   $$\mu \geq 0$$
   (10.29)

3. **Complementary slackness**: The Lagrange multipliers takes up the slack, where $\mu_i$ is zero or $g_i(x^*)$ is zero.
   $$\mu \odot g = 0$$
   (10.30)

---

If $u$ and $v$ are vectors of the same length, then we say $u \leq v$ when $u_i \leq v_i$ for all $i$. We define $\geq$, $<$, and $>$ similarly for vectors.

Named after Harold W. Kuhn and Albert W. Tucker who published the conditions in 1951. It was later discovered that William Karush studied these conditions in an unpublished master’s thesis in 1939. A historical prospective is provided by T.H. Kjeldsen, “A Contextualized Historical Analysis of the Kuhn-Tucker Theorem in Nonlinear Programming: The Impact of World War II,” *Historia Mathematica*, vol. 27, no. 4, pp. 331–361, 2000.

The operation $a \odot b$ indicates the element-wise product between vectors $a$ and $b$. 

© 2019 Massachusetts Institute of Technology, shared under a Creative Commons CC-BY-NC-ND license.

2022-05-22 00:25:57-07:00, revision 47fd495, comments to bugs@algorithmsbook.com
4. **Stationarity**: The objective function contour is tangent to each active constraint.\(^8\)

\[
\nabla f(x^*) + \sum_i \mu_i \nabla g_i(x^*) + \sum_j \lambda_j \nabla h_j(x^*) = 0 \tag{10.31}
\]

These four conditions are first-order necessary conditions for optimality and are thus FONCs for problems with smooth constraints. Just as with the FONCs for unconstrained optimization, special care must be taken to ensure that identified critical points are actually local minima.

### 10.6 Duality

In deriving the FONCs for constrained optimization, we also find a more general form for the Lagrangian. This *generalized Lagrangian* is

\[
\mathcal{L}(x, \mu, \lambda) = f(x) + \sum_i \mu_i g_i(x) + \sum_j \lambda_j h_j(x) \tag{10.32}
\]

The *primal form* of the optimization problem is the original optimization problem formulated using the generalized Lagrangian

\[
\begin{align*}
\text{minimize} & \quad \maximize_{\mu \geq 0, \lambda} \mathcal{L}(x, \mu, \lambda) \\
\text{maximize} & \quad \minimize_x \mathcal{L}(x, \mu, \lambda) \tag{10.33}
\end{align*}
\]

The primal problem is identical to the original problem and is just as difficult to optimize.

The *dual form* of the optimization problem reverses the order of the minimization and maximization in equation (10.33):

\[
\begin{align*}
\maximize_{\mu \geq 0, \lambda} & \quad \minimize_x \mathcal{L}(x, \mu, \lambda) \\
\minimize & \quad \maximize_a \minimize_b f(a, b) \leq \minimize_b \maximize_a f(a, b) \tag{10.35}
\end{align*}
\]

The max-min inequality states that for any function \( f(a, b) \):

The solution to the dual problem is thus a lower bound to the solution of the primal problem. That is, \( d^* \leq p^* \), where \( d^* \) is the dual value and \( p^* \) is the primal value.

The inner minimization in the dual problem is often folded into a *dual function*,

\[
\mathcal{D}(\mu \geq 0, \lambda) = \minimize_x \mathcal{L}(x, \mu, \lambda) \tag{10.36}
\]

\(^8\)Since the sign of \( \lambda \) is not restricted, we can reverse the sign for the equality constraints from the method of Lagrange multipliers.
for notational convenience. The dual function is concave.\footnote{For a detailed overview, see S. Nash and A. Sofer, Linear and Non-linear Programming. McGraw-Hill, 1996.} Gradient ascent on a concave function always converges to the global maximum. Optimizing the dual problem is easy whenever minimizing the Lagrangian with respect to $x$ is easy.

We know that $\max_{\mu \geq 0, \lambda} D(\mu, \lambda) \leq p^*$. It follows that the dual function is always a lower bound on the primal problem (see example 10.5). For any $\mu \geq 0$ and any $\lambda$, we have

$$D(\mu \geq 0, \lambda) \leq p^* \quad (10.37)$$

The difference $p^* - d^*$ between the dual and primal values is called the \textit{duality gap}. In some cases, the dual problem is guaranteed to have the same solution as the original problem—the duality gap is zero.\footnote{Conditions that guarantee a zero duality gap are discussed in S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge University Press, 2004.} In such cases, duality provides an alternative approach for optimizing our problem. Example 10.6 demonstrates this approach.

### 10.7 Penalty Methods

We can use \textit{penalty methods} to convert constrained optimization problems into unconstrained optimization problems by adding penalty terms to the objective function, allowing us to use the methods developed in previous chapters.

Consider a general optimization problem:

$$\begin{align*}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & g(x) \leq 0 \\
& h(x) = 0
\end{align*} \quad (10.38)$$

A simple penalty method counts the number of constraint equations that are violated:

$$p_{\text{count}}(x) = \sum_i (g_i(x) > 0) + \sum_j (h_j(x) \neq 0) \quad (10.39)$$

which results in the unconstrained optimization problem that penalizes infeasibility

$$\begin{align*}
\text{minimize} \quad & f(x) + \rho \cdot p_{\text{count}}(x) \\
\text{subject to} \quad & \text{subject to}
\end{align*} \quad (10.40)$$

where $\rho > 0$ adjusts the penalty magnitude. Figure 10.8 shows an example.
Consider the optimization problem:

\[
\begin{align*}
\text{minimize} & \quad \sin(x) \\
\text{subject to} & \quad x^2 \leq 1
\end{align*}
\]

The generalized Lagrangian is \( L(x, \mu) = \sin(x) + \mu(x^2 - 1) \), making the primal problem:

\[
\begin{align*}
\text{minimize} & \quad \text{maximize} \quad \sin(x) + \mu(x^2 - 1) \\
\text{subject to} & \quad \mu \geq 0
\end{align*}
\]

and the dual problem:

\[
\begin{align*}
\text{maximize} & \quad \text{minimize} \quad \sin(x) + \mu(x^2 - 1) \\
\text{subject to} & \quad \mu \geq 0
\end{align*}
\]

The objective function is plotted in black, with the feasible region traced over in blue. The minimum is at \( x^* = -1 \) with \( p^* \approx -0.841 \). The purple lines are the Lagrangian \( L(x, \mu) \) for different values of \( \mu \), each of which has a minimum lower than \( p^* \).
Consider the problem:

\[
\begin{align*}
\text{minimize} & \quad x_1 + x_2 + x_1 x_2 \\
\text{subject to} & \quad x_1^2 + x_2^2 = 1
\end{align*}
\]

The Lagrangian is \( \mathcal{L}(x_1, x_2, \lambda) = x_1 + x_2 + x_1 x_2 + \lambda(x_1^2 + x_2^2 - 1) \).

We apply the method of Lagrange multipliers:

\[
\begin{align*}
\frac{\partial \mathcal{L}}{\partial x_1} &= 1 + x_2 + 2\lambda x_1 = 0 \\
\frac{\partial \mathcal{L}}{\partial x_2} &= 1 + x_1 + 2\lambda x_2 = 0 \\
\frac{\partial \mathcal{L}}{\partial \lambda} &= x_1^2 + x_2^2 - 1 = 0
\end{align*}
\]

Solving yields four potential solutions, and thus four critical points:

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( \lambda )</th>
<th>( x_1 + x_2 + x_1 x_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>0</td>
<td>1/2</td>
<td>-1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>1/2</td>
<td>-1</td>
</tr>
<tr>
<td>( \frac{\sqrt{2} + 1}{\sqrt{2} + 2} )</td>
<td>( \frac{\sqrt{2} + 1}{\sqrt{2} + 2} )</td>
<td>( \frac{1}{2} \left( -1 - \sqrt{2} \right) )</td>
<td>( \frac{1}{2} + \sqrt{2} \approx 1.914 )</td>
</tr>
<tr>
<td>( \frac{\sqrt{2} - 1}{\sqrt{2} - 2} )</td>
<td>( \frac{\sqrt{2} - 1}{\sqrt{2} - 2} )</td>
<td>( \frac{1}{2} \left( -1 + \sqrt{2} \right) )</td>
<td>( \frac{1}{2} - \sqrt{2} \approx -0.914 )</td>
</tr>
</tbody>
</table>

We find that the two optimal solutions are \([-1, 0]\) and \([0, -1]\).

The dual function has the form

\[
D(\lambda) = \min_{x_1, x_2} x_1 + x_2 + x_1 x_2 + \lambda(x_1^2 + x_2^2 - 1)
\]

The dual function is unbounded below when \( \lambda \) is less than \( 1/2 \) (consider \( x_1 \to \infty \) and \( x_2 \to -\infty \)). Setting the gradient to \( 0 \) and solving yields \( x_2 = -1 - 2\lambda x_1 \) and \( x_1 = (2\lambda - 1)/(1 - 4\lambda^2) \) for \( \lambda \neq \pm 1/2 \). When \( \lambda = 1/2 \), \( x_1 = -1 - x_2 \) and \( D(1/2) = -1 \). Substituting these into the dual function yields:

\[
D(\lambda) = \begin{cases} 
-\lambda - \frac{1}{2\lambda + 1} & \lambda \geq \frac{1}{2} \\
-\infty & \text{otherwise}
\end{cases}
\]

The dual problem maximize\( \lambda \ D(\lambda) \) is maximized at \( \lambda = 1/2 \).
Penalty methods start with an initial point $x$ and a small value for $\rho$. The unconstrained optimization problem equation (10.40) is then solved. The resulting design point is then used as the starting point for another optimization with an increased penalty. We continue with this procedure until the resulting point is feasible, or a maximum number of iterations has been reached. Algorithm 10.1 provides an implementation.

```python
function penalty_method(f, p, x, k_max; ρ=1, γ=2)
    for k in 1 : k_max
        x = minimize(x -> f(x) + ρ*p(x), x)
        ρ *= γ
        if p(x) == 0
            return x
    return x
end
```

This penalty will preserve the problem solution for large values of $\rho$, but it introduces a sharp discontinuity. Points not inside the feasible set lack gradient information to guide the search towards feasibility.

We can use quadratic penalties to produce a smooth objective function (figure 10.9):

$$p_{\text{quadratic}}(x) = \sum_i \max(g_i(x), 0)^2 + \sum_j h_j(x)^2$$  \hspace{1cm} (10.41)

Quadratic penalties close to the constraint boundary are very small and may require $\rho$ to approach infinity before the solution ceases to violate the constraints.

It is also possible to mix a count and a quadratic penalty function (figure 10.10):

$$p_{\text{mixed}}(x) = ρ_1 p_{\text{count}}(x) + ρ_2 p_{\text{quadratic}}(x)$$  \hspace{1cm} (10.42)

Such a penalty mixture provides a clear boundary between the feasible region and the infeasible region while providing gradient information to the solver.

Figure 10.11 shows the progress of the penalty function as $\rho$ is increased. Quadratic penalty functions cannot ensure feasibility as discussed in example 10.7.
Consider the problem

\[
\begin{align*}
\text{minimize} & \quad x \\
\text{subject to} & \quad x \geq 5
\end{align*}
\]

using a quadratic penalty function.

The unconstrained objective function is

\[
f(x) = x + \rho \max(5 - x, 0)^2
\]

The minimum of the unconstrained objective function is

\[
x^* = 5 - \frac{1}{2\rho}
\]

While the minimum of the constrained optimization problem is clearly \(x = 5\), the minimum of the penalized optimization problem merely approaches \(x = 5\), requiring an infinite penalty to achieve feasibility.
10.8 Augmented Lagrange Method

The augmented Lagrange method\textsuperscript{11} is an adaptation of the penalty method specifically for equality constraints. Unlike the penalty method, where $\rho$ must sometimes approach infinity before a feasible solution is found, the augmented Lagrange method will work with smaller values of $\rho$. It uses both a quadratic and a linear penalty for each constraint.

For an optimization problem with equality constraints $h(x) = 0$, the penalty function is:

$$p_{\text{Lagrange}}(x) = \frac{1}{2} \rho \sum_i (h_i(x))^2 - \sum_i \lambda_i h_i(x) \quad (10.43)$$

where $\lambda$ converges toward the Lagrange multiplier.

In addition to increasing $\rho$ with each iteration, the linear penalty vector is updated according to:

$$\lambda^{(k+1)} = \lambda^{(k)} - \rho h(x) \quad (10.44)$$

Algorithm 10.2 provides an implementation.

```
function augmented_lagrange_method(f, h, x, k_max; ρ=1, γ=2)
    λ = zeros(length(h(x)))
    for k in 1 : k_max
        p = x -> ρ/2*sum(h(x).^2) - λ*h(x)
        x = minimize(x -> f(x) + p(x), x)
        λ = ρ*h(x)
        ρ *= γ
    end
    return x
end
```

Algorithm 10.2. The augmented Lagrange method for objective function $f$, equality constraint function $h$, initial point $x$, number of iterations $k_{\text{max}}$, initial penalty scalar $\rho > 0$, and penalty multiplier $\gamma > 1$. The function $\text{minimize}$ should be replaced with the minimization method of your choice.

10.9 Interior Point Methods

Interior point methods (algorithm 10.3), sometimes referred to as barrier methods, are optimization methods that ensure that the search points always remain feasible.\textsuperscript{12} Interior point methods use a barrier function that approaches infinity as one approaches a constraint boundary. This barrier function, $p_{\text{barrier}}(x)$, must satisfy several properties:

1. $p_{\text{barrier}}(x)$ is continuous
2. \( p_{\text{barrier}}(x) \) is nonnegative \((p_{\text{barrier}}(x) \geq 0)\) in the feasible region

3. \( p_{\text{barrier}}(x) \) approaches infinity as \( x \) approaches any constraint boundary

Some examples of barrier functions are:

**Inverse Barrier:**

\[
p_{\text{barrier}}(x) = -\sum_i \frac{1}{g_i(x)}
\]  
(10.45)

**Log Barrier:**

\[
p_{\text{barrier}}(x) = -\sum_i \begin{cases} 
\log(-g_i(x)) & \text{if } g_i(x) \geq -1 \\
0 & \text{otherwise}
\end{cases}
\]  
(10.46)

A problem with inequality constraints can be transformed into an unconstrained optimization problem

\[
\text{minimize } f(x) + \frac{1}{\rho} p_{\text{barrier}}(x)
\]  
(10.47)

When \( \rho \) is increased, the penalty for approaching the boundary decreases (figure 10.12).

Special care must be taken such that line searches do not leave the feasible region. Line searches \( f(x + \alpha d) \) are constrained to the interval \( \alpha = [0, \alpha_u] \), where \( \alpha_u \) is the step to the nearest boundary. In practice, \( \alpha_u \) is chosen such that \( x + \alpha d \) is just inside the boundary to avoid the boundary singularity.

Like the penalty method, the interior point method begins with a low value for \( \rho \) and slowly increases it until convergence. The interior point method is typically terminated when the difference between subsequent points is less than a certain threshold. Figure 10.13 shows an example of the effect of incrementally increasing \( \rho \).

The interior point method requires a feasible point from which to start the search. One convenient method for finding a feasible point is to optimize the quadratic penalty function

\[
\text{minimize } p_{\text{quadratic}}(x)
\]  
(10.48)
Algorithm 10.3. The interior point method for objective function $f$, barrier function $p$, initial point $x$, initial penalty $\rho > 0$, penalty multiplier $\gamma > 1$, and stopping tolerance $\epsilon > 0$.

function interior_point_method($f$, $p$, $x$; $\rho=1$, $\gamma=2$, $\epsilon=0.001$)
    delta = Inf
    while delta > $\epsilon$
        $x'$ = minimize($x \rightarrow f(x) + p(x)/\rho$, $x$)
        delta = norm($x'$ - $x$)
        $x = x'$
        $\rho$ *= $\gamma$
    end
    return $x$
end

Figure 10.13. The interior point method with the inverse barrier applied to the flower function, appendix B.4, and the constraint $x_1^2 + x_2^2 \geq 2$.
10.10 Summary

- Constraints are requirements on the design points that a solution must satisfy.
- Some constraints can be transformed or substituted into the problem to result in an unconstrained optimization problem.
- Analytical methods using Lagrange multipliers yield the generalized Lagrangian and the necessary conditions for optimality under constraints.
- A constrained optimization problem has a dual problem formulation that is easier to solve and whose solution is a lower bound of the solution to the original problem.
- Penalty methods penalize infeasible solutions and often provide gradient information to the optimizer to guide infeasible points toward feasibility.
- Interior point methods maintain feasibility but use barrier functions to avoid leaving the feasible set.

10.11 Exercises

Exercise 10.1. Solve

\[
\begin{align*}
\text{minimize} & \quad x \\
\text{subject to} & \quad x \geq 0 \\
\end{align*}
\]  \hspace{1cm} (10.49)

using the quadratic penalty method with \(\rho > 0\). Solve the problem in closed form.

Exercise 10.2. Solve the problem above using the count penalty method with \(\rho > 1\) and compare it to the quadratic penalty method.

Exercise 10.3. Suppose that you are solving a constrained problem with the penalty method. You notice that the iterates remain infeasible and you decide to stop the algorithm. What can you do to be more successful in your next attempt?

Exercise 10.4. Consider a simple univariate minimization problem where you minimize a function \(f(x)\) subject to \(x \geq 0\). Assume that you know that the constraint is active, that is, \(x^* = 0\) where \(x^*\) is the minimizer and \(f'(x^*) > 0\) from the optimality conditions. Show that solving the same problem with the penalty method

\[
f(x) + (\min(x,0))^2
\]  \hspace{1cm} (10.50)
yields an infeasible solution with respect to the original problem.

**Exercise 10.5.** What is the advantage of the augmented Lagrange method compared to the quadratic penalty method?

**Exercise 10.6.** When would you use the barrier method in place of the penalty method?

**Exercise 10.7.** Give an example of a smooth optimization problem, such that, for any penalty parameter \( \rho > 0 \), there exists a starting point \( x^{(1)} \) for which the steepest descent method diverges.

**Exercise 10.8.** Suppose you have an optimization problem

\[
\begin{align*}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & h(x) = 0 \\
& g(x) \leq 0
\end{align*}
\] (10.51)

but do not have an initial feasible design. How would you find a feasible point with respect to the constraints, provided that one exists?

**Exercise 10.9.** Solve the constrained optimization problem

\[
\begin{align*}
\text{minimize} \quad & \sin \left( \frac{4}{x} \right) \\
\text{subject to} \quad & x \in [1, 10]
\end{align*}
\] (10.52)

using both the transform \( x = t_{a,b}(\tilde{x}) \) and a sigmoid transform for constraint bounds \( x \in [a, b] \):

\[
x = s(\tilde{x}) = a + \frac{b-a}{1+e^{-\tilde{x}}}
\] (10.53)

Why is the \( t \) transform better than the sigmoid transform?

**Exercise 10.10.** Give an example of a quadratic objective function involving two design variables where the addition of a linear constraint results in a different optimum.

**Exercise 10.11.** Suppose we want to minimize \( x_1^3 + x_2^2 + x_3 \) subject to the constraint that \( x_1 + 2x_2 + 3x_3 = 6 \). How might we transform this into an unconstrained problem with the same minimizer?
Exercise 10.12. Suppose we want to minimize \(-x_1 - 2x_2\) subject to the constraints \(ax_1 + x_2 \leq 5\) and \(x_1, x_2 \geq 0\). If \(a\) is a bounded constant, what range of values of \(a\) will result in an infinite number of optimal solutions?

Exercise 10.13. Consider using a penalty method to optimize

\[
\begin{align*}
\text{minimize} &\quad 1 - x^2 \\
\text{subject to} &\quad |x| \leq 2
\end{align*}
\]

(10.54)

Optimization with the penalty method typically involves running several optimizations with increasing penalty weights. Impatient engineers may wish to optimize once using a very large penalty weight. Explain what issues are encountered for both the count penalty method and the quadratic penalty method.