Chapter 11

Interior-point methods

11.1 Inequality constrained minimization problems

In this chapter we discuss *interior-point methods* for solving convex optimization problems that include inequality constraints,

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b, \end{array} \tag{11.1}$$

where $f_0, \ldots, f_m : \mathbf{R}^n \to \mathbf{R}$ are convex and twice continuously differentiable, and $A \in \mathbf{R}^{p \times n}$ with $\mathbf{rank}\, A = p < n$. We assume that the problem is solvable, *i.e.*, an optimal $x^\star$ exists. We denote the optimal value $f_0(x^\star)$ as $p^\star$.

We also assume that the problem is strictly feasible, *i.e.*, there exists $x \in \mathcal{D}$ that satisfies $Ax = b$ and $f_i(x) < 0$ for $i = 1, \ldots, m$. This means that Slater's constraint qualification holds, so there exist dual optimal $\lambda^\star \in \mathbf{R}^m$, $\nu^\star \in \mathbf{R}^p$, which together with $x^\star$ satisfy the KKT conditions

$$\begin{array}{rcll} Ax^\star = b, \quad f_i(x^\star) & \leq & 0, & i = 1, \ldots, m \\ \lambda^\star & \succeq & 0 & \\ \nabla f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star \nabla f_i(x^\star) + A^T \nu^\star & = & 0 & \\ \lambda_i^\star f_i(x^\star) & = & 0, & i = 1, \ldots, m. \end{array} \tag{11.2}$$

Interior-point methods solve the problem (11.1) (or the KKT conditions (11.2)) by applying Newton's method to a sequence of equality constrained problems, or to a sequence of modified versions of the KKT conditions. We will concentrate on a particular interior-point algorithm, the *barrier method*, for which we give a proof of convergence and a complexity analysis. We also describe a simple *primal-dual interior-point method* (in §11.7), but do not give an analysis.

We can view interior-point methods as another level in the hierarchy of convex optimization algorithms. Linear equality constrained quadratic problems are the simplest. For these problems the KKT conditions are a set of linear equations, which can be solved analytically. Newton's method is the next level in the hierarchy. We can think of Newton's method as a technique for solving a linear equality constrained optimization problem, with twice differentiable objective, by reducing it to a sequence of linear equality constrained quadratic problems. Interior-point methods form the next level in the hierarchy: They solve an optimization problem with linear equality and inequality constraints by reducing it to a sequence of linear equality constrained problems.

Examples

Many problems are already in the form (11.1), and satisfy the assumption that the objective and constraint functions are twice differentiable. Obvious examples are LPs, QPs, QCQPs, and GPs in convex form; another example is linear inequality constrained entropy maximization,

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g \\ & Ax = b, \end{array}$$

with domain $\mathcal{D} = \mathbf{R}^n_{++}$.

Many other problems do not have the required form (11.1), with twice differentiable objective and constraint functions, but can be reformulated in the required form. We have already seen many examples of this, such as the transformation of an unconstrained convex piecewise-linear minimization problem

$$\text{minimize} \quad \max_{i=1,\ldots,m}(a_i^T x + b_i)$$

(with nondifferentiable objective), to the LP

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & a_i^T x + b_i \leq t, \quad i = 1, \ldots, m \end{array}$$

(which has twice differentiable objective and constraint functions).

Other convex optimization problems, such as SOCPs and SDPs, are not readily recast in the required form, but can be handled by extensions of interior-point methods to problems with generalized inequalities, which we describe in §11.6.

11.2 Logarithmic barrier function and central path

Our goal is to approximately formulate the inequality constrained problem (11.1) as an equality constrained problem to which Newton's method can be applied. Our first step is to rewrite the problem (11.1), making the inequality constraints implicit in the objective:

$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^m I_-(f_i(x)) \\ \text{subject to} & Ax = b, \end{array} \qquad (11.3)$$

where $I_- : \mathbf{R} \to \mathbf{R}$ is the indicator function for the nonpositive reals,

$$I_-(u) = \begin{cases} 0 & u \leq 0 \\ \infty & u > 0. \end{cases}$$

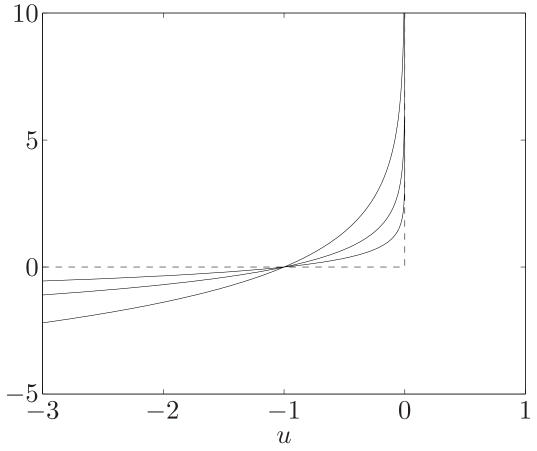

Figure 11.1 The dashed lines show the function $I_-(u)$, and the solid curves show $\widehat{I}_-(u) = -(1/t)\log(-u)$, for $t = 0.5,\ 1,\ 2$. The curve for $t = 2$ gives the best approximation.

The problem (11.3) has no inequality constraints, but its objective function is not (in general) differentiable, so Newton's method cannot be applied.

11.2.1 Logarithmic barrier

The basic idea of the barrier method is to approximate the indicator function I_- by the function

$$\widehat{I}_-(u) = -(1/t)\log(-u), \qquad \mathbf{dom}\,\widehat{I}_- = -\mathbf{R}_{++},$$

where $t > 0$ is a parameter that sets the accuracy of the approximation. Like I_-, the function $\widehat{I}_-$ is convex and nondecreasing, and (by our convention) takes on the value ∞ for $u > 0$. Unlike I_-, however, $\widehat{I}_-$ is differentiable and closed: it increases to ∞ as u increases to 0. Figure 11.1 shows the function I_-, and the approximation $\widehat{I}_-$, for several values of t. As t increases, the approximation becomes more accurate.

Substituting $\widehat{I}_-$ for I_- in (11.3) gives the approximation

$$\begin{array}{ll} \text{minimize} & f_0(x) + \sum_{i=1}^m -(1/t)\log(-f_i(x)) \\ \text{subject to} & Ax = b. \end{array} \tag{11.4}$$

The objective here is convex, since $-(1/t)\log(-u)$ is convex and increasing in u, and differentiable. Assuming an appropriate closedness condition holds, Newton's method can be used to solve it.

The function

$$\phi(x) = -\sum_{i=1}^m \log(-f_i(x)), \tag{11.5}$$

with $\mathbf{dom}\,\phi = \{x \in \mathbf{R}^n \mid f_i(x) < 0,\ i = 1, \ldots, m\}$, is called the *logarithmic barrier* or *log barrier* for the problem (11.1). Its domain is the set of points that satisfy the inequality constraints of (11.1) strictly. No matter what value the positive parameter t has, the logarithmic barrier grows without bound if $f_i(x) \to 0$, for any i.

Of course, the problem (11.4) is only an approximation of the original problem (11.3), so one question that arises immediately is how well a solution of (11.4) approximates a solution of the original problem (11.3). Intuition suggests, and we will soon confirm, that the quality of the approximation improves as the parameter t grows.

On the other hand, when the parameter t is large, the function $f_0 + (1/t)\phi$ is difficult to minimize by Newton's method, since its Hessian varies rapidly near the boundary of the feasible set. We will see that this problem can be circumvented by solving a *sequence* of problems of the form (11.4), increasing the parameter t (and therefore the accuracy of the approximation) at each step, and starting each Newton minimization at the solution of the problem for the previous value of t.

For future reference, we note that the gradient and Hessian of the logarithmic barrier function ϕ are given by

$$\begin{aligned} \nabla\phi(x) &= \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x), \\ \nabla^2\phi(x) &= \sum_{i=1}^m \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla^2 f_i(x) \end{aligned}$$

(see §A.4.2 and §A.4.4).

11.2.2 Central path

We now consider in more detail the minimization problem (11.4). It will simplify notation later on if we multiply the objective by t, and consider the equivalent problem

$$\begin{array}{ll} \text{minimize} & t f_0(x) + \phi(x) \\ \text{subject to} & Ax = b, \end{array} \tag{11.6}$$

which has the same minimizers. We assume for now that the problem (11.6) can be solved via Newton's method, and, in particular, that it has a unique solution for each $t > 0$. (We will discuss this assumption in more detail in §11.3.3.)

For $t > 0$ we define $x^\star(t)$ as the solution of (11.6). The *central path* associated with problem (11.1) is defined as the set of points $x^\star(t)$, $t > 0$, which we call the *central points*. Points on the central path are characterized by the following necessary and sufficient conditions: $x^\star(t)$ is strictly feasible, *i.e.*, satisfies

$$Ax^\star(t) = b, \qquad f_i(x^\star(t)) < 0, \quad i = 1, \ldots, m,$$

and there exists a $\hat\nu \in \mathbf{R}^p$ such that

$$0 = t\nabla f_0(x^\star(t)) + \nabla\phi(x^\star(t)) + A^T\hat\nu$$

$$= \quad t\nabla f_0(x^\star(t)) + \sum_{i=1}^m \frac{1}{-f_i(x^\star(t))}\nabla f_i(x^\star(t)) + A^T\hat{\nu} \qquad (11.7)$$

holds.

Example 11.1 *Inequality form linear programming.* The logarithmic barrier function for an LP in inequality form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b, \end{array} \qquad (11.8)$$

is given by

$$\phi(x) = -\sum_{i=1}^m \log(b_i - a_i^T x), \qquad \mathbf{dom}\,\phi = \{x \mid Ax \prec b\},$$

where $a_1^T, \ldots, a_m^T$ are the rows of A. The gradient and Hessian of the barrier function are

$$\nabla\phi(x) = \sum_{i=1}^m \frac{1}{b_i - a_i^T x} a_i, \qquad \nabla^2\phi(x) = \sum_{i=1}^m \frac{1}{(b_i - a_i^T x)^2} a_i a_i^T,$$

or, more compactly,

$$\nabla\phi(x) = A^T d, \qquad \nabla^2\phi(x) = A^T \mathbf{diag}(d)^2 A,$$

where the elements of $d \in \mathbf{R}^m$ are given by $d_i = 1/(b_i - a_i^T x)$. Since x is strictly feasible, we have $d \succ 0$, so the Hessian of ϕ is nonsingular if and only if A has rank n.

The centrality condition (11.7) is

$$tc + \sum_{i=1}^m \frac{1}{b_i - a_i^T x} a_i = tc + A^T d = 0. \qquad (11.9)$$

We can give a simple geometric interpretation of the centrality condition. At a point $x^\star(t)$ on the central path the gradient $\nabla\phi(x^\star(t))$, which is normal to the level set of ϕ through $x^\star(t)$, must be parallel to $-c$. In other words, the hyperplane $c^T x = c^T x^\star(t)$ is tangent to the level set of ϕ through $x^\star(t)$. Figure 11.2 shows an example with $m = 6$ and $n = 2$.

Dual points from central path

From (11.7) we can derive an important property of the central path: Every central point yields a dual feasible point, and hence a lower bound on the optimal value $p^\star$. More specifically, define

$$\lambda_i^\star(t) = -\frac{1}{tf_i(x^\star(t))}, \quad i = 1, \ldots, m, \qquad \nu^\star(t) = \hat{\nu}/t. \qquad (11.10)$$

We claim that the pair $\lambda^\star(t)$, $\nu^\star(t)$ is dual feasible.

First, it is clear that $\lambda^\star(t) \succ 0$ because $f_i(x^\star(t)) < 0$, $i = 1, \ldots, m$. By expressing the optimality conditions (11.7) as

$$\nabla f_0(x^\star(t)) + \sum_{i=1}^m \lambda_i^\star(t)\nabla f_i(x^\star(t)) + A^T\nu^\star(t) = 0,$$

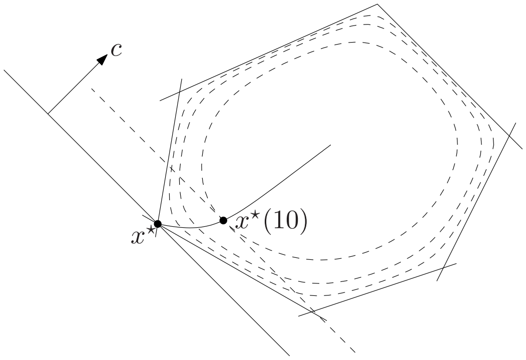

Figure 11.2 Central path for an LP with $n = 2$ and $m = 6$. The dashed curves show three contour lines of the logarithmic barrier function ϕ. The central path converges to the optimal point $x^\star$ as $t \to \infty$. Also shown is the point on the central path with $t = 10$. The optimality condition (11.9) at this point can be verified geometrically: The line $c^T x = c^T x^\star(10)$ is tangent to the contour line of ϕ through $x^\star(10)$.

we see that $x^\star(t)$ minimizes the Lagrangian

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \nu^T(Ax - b),$$

for $\lambda = \lambda^\star(t)$ and $\nu = \nu^\star(t)$, which means that $\lambda^\star(t)$, $\nu^\star(t)$ is a dual feasible pair. Therefore the dual function $g(\lambda^\star(t), \nu^\star(t))$ is finite, and

$$\begin{array}{rcl} g(\lambda^\star(t), \nu^\star(t)) & = & f_0(x^\star(t)) + \sum_{i=1}^m \lambda_i^\star(t) f_i(x^\star(t)) + \nu^\star(t)^T(Ax^\star(t) - b) \\ & = & f_0(x^\star(t)) - m/t. \end{array}$$

In particular, the duality gap associated with $x^\star(t)$ and the dual feasible pair $\lambda^\star(t)$, $\nu^\star(t)$ is simply m/t. As an important consequence, we have

$$f_0(x^\star(t)) - p^\star \le m/t,$$

i.e., $x^\star(t)$ is no more than m/t-suboptimal. This confirms the intuitive idea that $x^\star(t)$ converges to an optimal point as $t \to \infty$.

Example 11.2 *Inequality form linear programming.* The dual of the inequality form LP (11.8) is

$$\begin{array}{ll} \text{maximize} & -b^T\lambda \\ \text{subject to} & A^T\lambda + c = 0 \\ & \lambda \succeq 0. \end{array}$$

From the optimality conditions (11.9), it is clear that

$$\lambda_i^\star(t) = \frac{1}{t(b_i - a_i^T x^\star(t))}, \quad i = 1, \ldots, m,$$

is dual feasible, with dual objective value

$$-b^T\lambda^\star(t) = c^Tx^\star(t) + (Ax^\star(t) - b)^T\lambda^\star(t) = c^Tx^\star(t) - m/t.$$

Interpretation via KKT conditions

We can also interpret the central path conditions (11.7) as a continuous deformation of the KKT optimality conditions (11.2). A point x is equal to $x^\star(t)$ if and only if there exists λ, ν such that

$$\begin{array}{rcll} Ax = b, \quad f_i(x) & \leq & 0, & i = 1,\ldots,m \\ \lambda & \succeq & 0 & \\ \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + A^T\nu & = & 0 & \\ -\lambda_i f_i(x) & = & 1/t, & i = 1,\ldots,m. \end{array} \tag{11.11}$$

The only difference between the KKT conditions (11.2) and the centrality conditions (11.11) is that the complementarity condition $-\lambda_i f_i(x) = 0$ is replaced by the condition $-\lambda_i f_i(x) = 1/t$. In particular, for large t, $x^\star(t)$ and the associated dual point $\lambda^\star(t)$, $\nu^\star(t)$ 'almost' satisfy the KKT optimality conditions for (11.1).

Force field interpretation

We can give a simple mechanics interpretation of the central path in terms of potential forces acting on a particle in the strictly feasible set C. For simplicity we assume that there are no equality constraints.

We associate with each constraint the force

$$F_i(x) = -\nabla\left(-\log(-f_i(x))\right) = \frac{1}{f_i(x)}\nabla f_i(x)$$

acting on the particle when it is at position x. The potential associated with the total force field generated by the constraints is the logarithmic barrier ϕ. As the particle moves toward the boundary of the feasible set, it is strongly repelled by the forces generated by the constraints.

Now we imagine another force acting on the particle, given by

$$F_0(x) = -t\nabla f_0(x),$$

when the particle is at position x. This objective force field acts to pull the particle in the negative gradient direction, *i.e.*, toward smaller f_0. The parameter t scales the objective force, relative to the constraint forces.

The central point $x^\star(t)$ is the point where the constraint forces exactly balance the objective force felt by the particle. As the parameter t increases, the particle is more strongly pulled toward the optimal point, but it is always trapped in C by the barrier potential, which becomes infinite as the particle approaches the boundary.

Example 11.3 *Force field interpretation for inequality form LP.* The force field associated with the ith constraint of the LP (11.8) is

$$F_i(x) = \frac{-a_i}{b_i - a_i^Tx}.$$

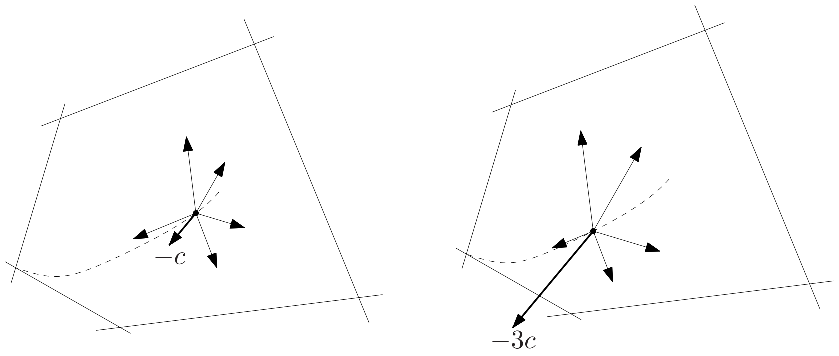

Figure 11.3 *Force field interpretation of central path.* The central path is shown as the dashed curve. The two points $x^\star(1)$ and $x^\star(3)$ are shown as dots in the left and right plots, respectively. The objective force, which is equal to $-c$ and $-3c$, respectively, is shown as a heavy arrow. The other arrows represent the constraint forces, which are given by an inverse-distance law. As the strength of the objective force varies, the equilibrium position of the particle traces out the central path.

This force is in the direction of the inward pointing normal to the constraint plane $\mathcal{H}_i = \{x \mid a_i^T x = b_i\}$, and has magnitude inversely proportional to the distance to $\mathcal{H}_i$, i.e.,

$$\|F_i(x)\|_2 = \frac{\|a_i\|_2}{b_i - a_i^T x} = \frac{1}{\mathbf{dist}(x, \mathcal{H}_i)}.$$

In other words, each constraint hyperplane has an associated repulsive force, given by the inverse distance to the hyperplane.

The term $tc^T x$ is the potential associated with a constant force $-tc$ on the particle. This 'objective force' pushes the particle in the direction of low cost. Thus, $x^\star(t)$ is the equilibrium position of the particle when it is subject to the inverse-distance constraint forces, and the objective force $-tc$. When t is very large, the particle is pushed almost to the optimal point. The strong objective force is balanced by the opposing constraint forces, which are large because we are near the feasible boundary.

Figure 11.3 illustrates this interpretation for a small LP with $n = 2$ and $m = 5$. The lefthand plot shows $x^\star(t)$ for $t = 1$, as well as the constraint forces acting on it, which balance the objective force. The righthand plot shows $x^\star(t)$ and the associated forces for $t = 3$. The larger value of objective force moves the particle closer to the optimal point.

11.3 The barrier method

We have seen that the point $x^\star(t)$ is m/t-suboptimal, and that a certificate of this accuracy is provided by the dual feasible pair $\lambda^\star(t)$, $\nu^\star(t)$. This suggests a very straightforward method for solving the original problem (11.1) with a guaranteed specified accuracy ϵ: We simply take $t = m/\epsilon$ and solve the equality constrained problem

$$\begin{array}{ll}\text{minimize} & (m/\epsilon)f_0(x) + \phi(x)\\ \text{subject to} & Ax = b\end{array}$$

using Newton's method. This method could be called the *unconstrained minimization method*, since it allows us to solve the inequality constrained problem (11.1) to a guaranteed accuracy by solving an unconstrained, or linearly constrained, problem. Although this method can work well for small problems, good starting points, and moderate accuracy (*i.e.*, ϵ not too small), it does not work well in other cases. As a result it is rarely, if ever, used.

11.3.1 The barrier method

A simple extension of the unconstrained minimization method does work well. It is based on solving a sequence of unconstrained (or linearly constrained) minimization problems, using the last point found as the starting point for the next unconstrained minimization problem. In other words, we compute $x^\star(t)$ for a sequence of increasing values of t, until $t \geq m/\epsilon$, which guarantees that we have an ϵ-suboptimal solution of the original problem. When the method was first proposed by Fiacco and McCormick in the 1960s, it was called the *sequential unconstrained minimization technique* (SUMT). Today the method is usually called the *barrier method* or *path-following method*. A simple version of the method is as follows.

Algorithm 11.1 *Barrier method.*

```
given strictly feasible $x$, $t := t^{(0)} > 0$, $\mu > 1$, tolerance $\epsilon > 0$.
repeat
    1. Centering step.
        Compute $x^\star(t)$ by minimizing $tf_0 + \phi$, subject to $Ax = b$, starting at $x$.
    2. Update. $x := x^\star(t)$.
    3. Stopping criterion. quit if $m/t < \epsilon$.
    4. Increase $t$. $t := \mu t$.
```

At each iteration (except the first one) we compute the central point $x^\star(t)$ starting from the previously computed central point, and then increase t by a factor $\mu > 1$. The algorithm can also return $\lambda = \lambda^\star(t)$, and $\nu = \nu^\star(t)$, a dual ϵ-suboptimal point, or certificate for x.

We refer to each execution of step 1 as a *centering step* (since a central point is being computed) or an *outer iteration*, and to the first centering step (the computation of $x^\star(t^{(0)})$) as the *initial centering step*. (Thus the simple algorithm with $t^{(0)} = m/\epsilon$ consists of only the initial centering step.) Although any method for linearly constrained minimization can be used in step 1, we will assume that Newton's method is used. We refer to the Newton iterations or steps executed during the centering step as *inner iterations*. At each inner step, we have a primal feasible point; we have a dual feasible point, however, only at the end of each outer (centering) step.

Accuracy of centering

We should make some comments on the accuracy to which we solve the centering problems. Computing $x^\star(t)$ exactly is not necessary since the central path has no significance beyond the fact that it leads to a solution of the original problem as $t \to \infty$; inexact centering will still yield a sequence of points $x^{(k)}$ that converges to an optimal point. Inexact centering, however, means that the points $\lambda^\star(t)$, $\nu^\star(t)$, computed from (11.10), are not exactly dual feasible. This can be corrected by adding a correction term to the formula (11.10), which yields a dual feasible point provided the computed x is near the central path, *i.e.*, $x^\star(t)$ (see exercise 11.9).

On the other hand, the cost of computing an *extremely accurate* minimizer of $tf_0 + \phi$, as compared to the cost of computing a *good* minimizer of $tf_0 + \phi$, is only marginally more, *i.e.*, a few Newton steps at most. For this reason it is not unreasonable to assume exact centering.

Choice of μ

The choice of the parameter μ involves a trade-off in the number of inner and outer iterations required. If μ is small (*i.e.*, near 1) then at each outer iteration t increases by a small factor. As a result the initial point for the Newton process, *i.e.*, the previous iterate x, is a very good starting point, and the number of Newton steps needed to compute the next iterate is small. Thus for small μ we expect a small number of Newton steps per outer iteration, but of course a large number of outer iterations since each outer iteration reduces the gap by only a small amount. In this case the iterates (and indeed, the iterates of the inner iterations as well) closely follow the central path. This explains the alternate name *path-following method*.

On the other hand if μ is large we have the opposite situation. After each outer iteration t increases a large amount, so the current iterate is probably not a very good approximation of the next iterate. Thus we expect many more inner iterations. This 'aggressive' updating of t results in fewer outer iterations, since the duality gap is reduced by the large factor μ at each outer iteration, but more inner iterations. With μ large, the iterates are widely separated on the central path; the inner iterates veer way off the central path.

This trade-off in the choice of μ is confirmed both in practice and, as we will see, in theory. In practice, small values of μ (*i.e.*, near one) result in many outer iterations, with just a few Newton steps for each outer iteration. For μ in a fairly large range, from around 3 to 100 or so, the two effects nearly cancel, so the total number of Newton steps remains approximately constant. This means that the choice of μ is not particularly critical; values from around 10 to 20 or so seem to work well. When the parameter μ is chosen to give the best worst-case bound on the total number of Newton steps required, values of μ near one are used.

Choice of $t^{(0)}$

Another important issue is the choice of initial value of t. Here the trade-off is simple: If $t^{(0)}$ is chosen too large, the first outer iteration will require too many iterations. If $t^{(0)}$ is chosen too small, the algorithm will require extra outer iterations, and possibly too many inner iterations in the first centering step.

Since $m/t^{(0)}$ is the duality gap that will result from the first centering step, one reasonable choice is to choose $t^{(0)}$ so that $m/t^{(0)}$ is approximately of the same order as $f_0(x^{(0)}) - p^\star$, or μ times this amount. For example, if a dual feasible point λ, ν is known, with duality gap $\eta = f_0(x^{(0)}) - g(\lambda, \nu)$, then we can take $t^{(0)} = m/\eta$. Thus, in the first outer iteration we simply compute a pair with the same duality gap as the initial primal and dual feasible points.

Another possibility is suggested by the central path condition (11.7). We can interpret

$$\inf_{\nu} \left\| t\nabla f_0(x^{(0)}) + \nabla\phi(x^{(0)}) + A^T\nu \right\|_2 \tag{11.12}$$

as a measure for the deviation of $x^{(0)}$ from the point $x^\star(t)$, and choose for $t^{(0)}$ the value that minimizes (11.12). (This value of t and ν can be found by solving a least-squares problem.)

A variation on this approach uses an affine-invariant measure of deviation between x and $x^\star(t)$ in place of the Euclidean norm. We choose t and ν that minimize

$$\alpha(t,\nu) = \left(t\nabla f_0(x^{(0)}) + \nabla\phi(x^{(0)}) + A^T\nu\right)^T H_0^{-1} \left(t\nabla f_0(x^{(0)}) + \nabla\phi(x^{(0)}) + A^T\nu\right),$$

where

$$H_0 = t\nabla^2 f_0(x^{(0)}) + \nabla^2\phi(x^{(0)}).$$

(It can be shown that $\inf_\nu \alpha(t,\nu)$ is the square of the Newton decrement of $tf_0 + \phi$ at $x^{(0)}$.) Since α is a quadratic-over-linear function of ν and t, it is convex.

Infeasible start Newton method

In one variation on the barrier method, an infeasible start Newton method (described in §10.3) is used for the centering steps. Thus, the barrier method is initialized with a point $x^{(0)}$ that satisfies $x^{(0)} \in \textbf{dom}\, f_0$ and $f_i(x^{(0)}) < 0$, $i = 1, \ldots, m$, but not necessarily $Ax^{(0)} = b$. Assuming the problem is strictly feasible, a full Newton step is taken at some point during the first centering step, and thereafter, the iterates are all primal feasible, and the algorithm coincides with the (standard) barrier method.

11.3.2 Examples

Linear programming in inequality form

Our first example is a small LP in inequality form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \end{array}$$

with $A \in \mathbf{R}^{100\times 50}$. The data were generated randomly, in such a way that the problem is strictly primal and dual feasible, with optimal value $p^\star = 1$.

The initial point $x^{(0)}$ is on the central path, with a duality gap of 100. The barrier method is used to solve the problem, and terminated when the duality gap is less than 10^{-6}. The centering problems are solved by Newton's method with backtracking, using parameters $\alpha = 0.01$, $\beta = 0.5$. The stopping criterion for

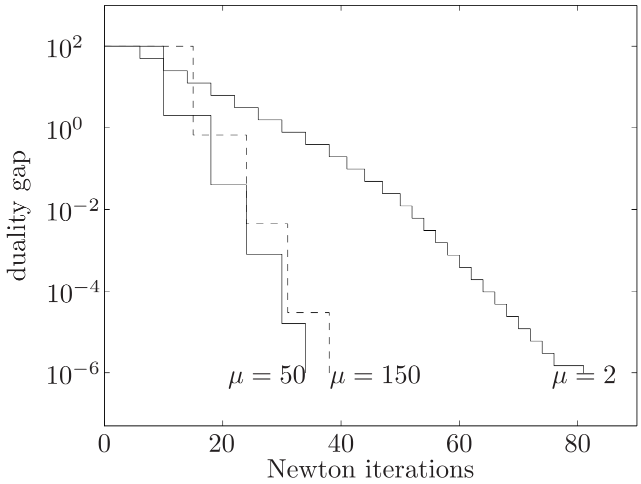

Figure 11.4 Progress of barrier method for a small LP, showing duality gap versus cumulative number of Newton steps. Three plots are shown, corresponding to three values of the parameter μ: 2, 50, and 150. In each case, we have approximately linear convergence of duality gap.

Newton's method is $\lambda(x)^2/2 \leq 10^{-5}$, where $\lambda(x)$ is the Newton decrement of the function $tc^T x + \phi(x)$.

The progress of the barrier method, for three values of the parameter μ, is shown in figure 11.4. The vertical axis shows the duality gap on a log scale. The horizontal axis shows the cumulative total number of inner iterations, *i.e.*, Newton steps, which is the natural measure of computational effort. Each of the plots has a staircase shape, with each stair associated with one outer iteration. The width of each stair tread (*i.e.*, horizontal portion) is the number of Newton steps required for that outer iteration. The height of each stair riser (*i.e.*, the vertical portion) is exactly equal to (a factor of) μ, since the duality gap is reduced by the factor μ at the end of each outer iteration.

The plots illustrate several typical features of the barrier method. First of all, the method works very well, with approximately linear convergence of the duality gap. This is a consequence of the approximately constant number of Newton steps required to re-center, for each value of μ. For $\mu = 50$ and $\mu = 150$, the barrier method solves the problem with a total number of Newton steps between 35 and 40.

The plots in figure 11.4 clearly show the trade-off in the choice of μ. For $\mu = 2$, the treads are short; the number of Newton steps required to re-center is around 2 or 3. But the risers are also short, since the duality gap reduction per outer iteration is only a factor of 2. At the other extreme, when $\mu = 150$, the treads are longer, typically around 7 Newton steps, but the risers are also much larger, since the duality gap is reduced by the factor 150 in each outer iteration.

The trade-off in choice of μ is further examined in figure 11.5. We use the barrier method to solve the LP, terminating when the duality gap is smaller than 10^{-3}, for 25 values of μ between 1.2 and 200. The plot shows the total number of Newton steps required to solve the problem, as a function of the parameter μ.

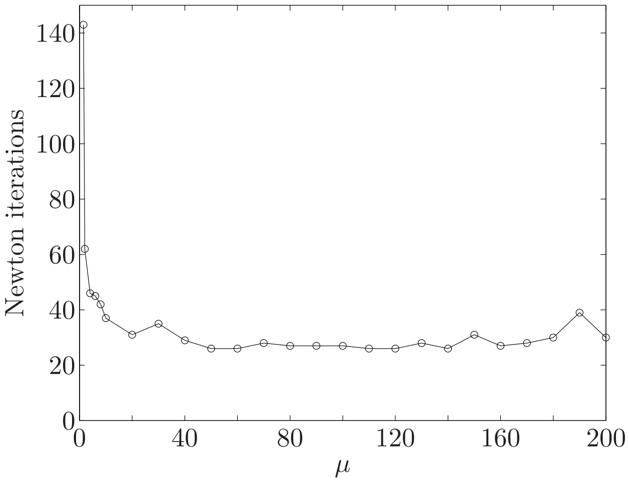

Figure 11.5 Trade-off in the choice of the parameter μ, for a small LP. The vertical axis shows the total number of Newton steps required to reduce the duality gap from 100 to 10^{-3}, and the horizontal axis shows μ. The plot shows the barrier method works well for values of μ larger than around 3, but is otherwise not sensitive to the value of μ.

This plot shows that the barrier method performs very well for a wide range of values of μ, from around 3 to 200. As our intuition suggests, the total number of Newton steps rises when μ is too small, due to the larger number of outer iterations required. One interesting observation is that the total number of Newton steps does not vary much for values of μ larger than around 3. Thus, as μ increases over this range, the decrease in the number of outer iterations is offset by an increase in the number of Newton steps per outer iteration. For even larger values of μ, the performance of the barrier method becomes less predictable (*i.e.*, more dependent on the particular problem instance). Since the performance does not improve with larger values of μ, a good choice is in the range 10 – 100.

Geometric programming

We consider a geometric program in convex form,

$$\begin{array}{ll} \text{minimize} & \log\left(\sum_{k=1}^{K_0} \exp(a_{0k}^T x + b_{0k})\right) \\ \text{subject to} & \log\left(\sum_{k=1}^{K_i} \exp(a_{ik}^T x + b_{ik})\right) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

with variable $x \in \mathbf{R}^n$, and associated logarithmic barrier

$$\phi(x) = -\sum_{i=1}^{m} \log\left(-\log\sum_{k=1}^{K_i} \exp(a_{ik}^T x + b_{ik})\right).$$

The problem instance we consider has $n = 50$ variables and $m = 100$ inequalities (like the small LP considered above). The objective and constraint functions all

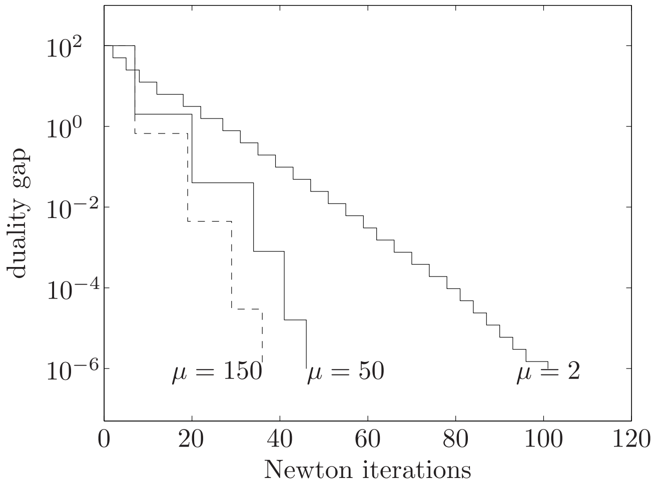

Figure 11.6 Progress of barrier method for a small GP, showing duality gap versus cumulative number of Newton steps. Again we have approximately linear convergence of duality gap.

have $K_i = 5$ terms. The problem instance was generated randomly, in such a way that it is strictly primal and dual feasible, with optimal value one.

We start with a point $x^{(0)}$ on the central path, with a duality gap of 100. The barrier method is used to solve the problem, with parameters $\mu = 2$, $\mu = 50$, and $\mu = 150$, and terminated when the duality gap is less than 10^{-6}. The centering problems are solved using Newton's method, with the same parameter values as in the LP example, *i.e.*, $\alpha = 0.01$, $\beta = 0.5$, and stopping criterion $\lambda(x)^2/2 \leq 10^{-5}$.

Figure 11.6 shows the duality gap versus cumulative number of Newton steps. This plot is very similar to the plot for LP, shown in figure 11.4. In particular, we see an approximately constant number of Newton steps required per centering step, and therefore approximately linear convergence of the duality gap.

The variation of the total number of Newton steps required to solve the problem, versus the parameter μ, is very similar to that in the LP example. For this GP, the total number of Newton steps required to reduce the duality gap below 10^{-3} is around 30 (ranging from around 20 to 40 or so) for values of μ between 10 and 200. So here, too, a good choice of μ is in the range $10 - 100$.

A family of standard form LPs

In the examples above we examined the progress of the barrier method, in terms of duality gap versus cumulative number of Newton steps, for a randomly generated instance of an LP and a GP, with similar dimensions. The results for the two examples are remarkably similar; each shows approximately linear convergence of duality gap with the number of Newton steps. We also examined the variation in performance with the parameter μ, and found essentially the same results in the two cases. For μ above around 10, the barrier method performs very well, requiring around 30 Newton steps to bring the duality gap down from 10^2 to 10^{-6}. In both cases, the choice of μ hardly affects the total number of Newton steps required (provided μ is larger than 10 or so).

In this section we examine the performance of the barrier method as a function of the problem dimensions. We consider LPs in standard form,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \succeq 0 \end{array}$$

with $A \in \mathbf{R}^{m \times n}$, and explore the total number of Newton steps required as a function of the number of variables n and number of equality constraints m, for a family of randomly generated problem instances. We take $n = 2m$, *i.e.*, twice as many variables as constraints.

The problems were generated as follows. The elements of A are independent and identically distributed, with zero mean, unit variance normal distribution $\mathcal{N}(0,1)$. We take $b = Ax^{(0)}$ where the elements of $x^{(0)}$ are independent, and uniformly distributed in $[0,1]$. This ensures that the problem is strictly primal feasible, since $x^{(0)} \succ 0$ is feasible. To construct the cost vector c, we first compute a vector $z \in \mathbf{R}^m$ with elements distributed according to $\mathcal{N}(0,1)$ and a vector $s \in \mathbf{R}^n$ with elements from a uniform distribution on $[0,1]$. We then take $c = A^T z + s$. This guarantees that the problem is strictly dual feasible, since $A^T z \prec c$.

The algorithm parameters we use are $\mu = 100$, and the same parameters for the centering steps in the examples above: backtracking parameters $\alpha = 0.01$, $\beta = 0.5$, and stopping criterion $\lambda(x)^2/2 \leq 10^{-5}$. The initial point is on the central path with $t^{(0)} = 1$ (*i.e.*, gap n). The algorithm is terminated when the initial duality gap is reduced by a factor 10^4, *i.e.*, after completing two outer iterations.

Figure 11.7 shows the duality gap versus iteration number for three problem instances, with dimensions $m = 50$, $m = 500$, and $m = 1000$. The plots look very much like the others, with approximately linear convergence of the duality gap. The plots show a small increase in the number of Newton steps required as the problem size grows from 50 constraints (100 variables) to 1000 constraints (2000 variables).

To examine the effect of problem size on the number of Newton steps required, we generate 100 problem instances for each of 20 values of m, ranging from $m = 10$ to $m = 1000$. We solve each of these 2000 problems using the barrier method, noting the number of Newton steps required. The results are summarized in figure 11.8, which shows the mean and standard deviation in the number of Newton steps, for each value of m. The first comment we make is that the standard deviation is around 2 iterations, and appears to be approximately independent of problem size. Since the average number of steps required is near 25, this means that the number of Newton steps required varies only around ±10%.

The plot shows that the number of Newton steps required grows only slightly, from around 21 to around 27, as the problem dimensions increase by a factor of 100. This behavior is typical for the barrier method in general: The number of Newton steps required grows very slowly with problem dimensions, and is almost always around a few tens. Of course, the computational effort to carry out one Newton step grows with the problem dimensions.

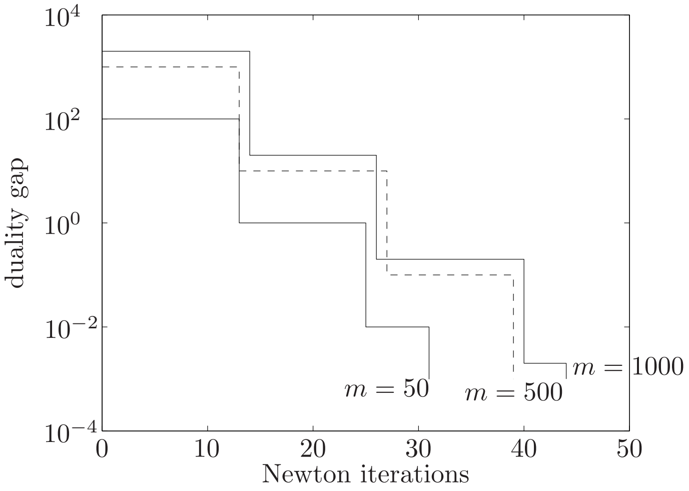

Figure 11.7 Progress of barrier method for three randomly generated standard form LPs of different dimensions, showing duality gap versus cumulative number of Newton steps. The number of variables in each problem is $n = 2m$. Here too we see approximately linear convergence of the duality gap, with a slight increase in the number of Newton steps required for the larger problems.

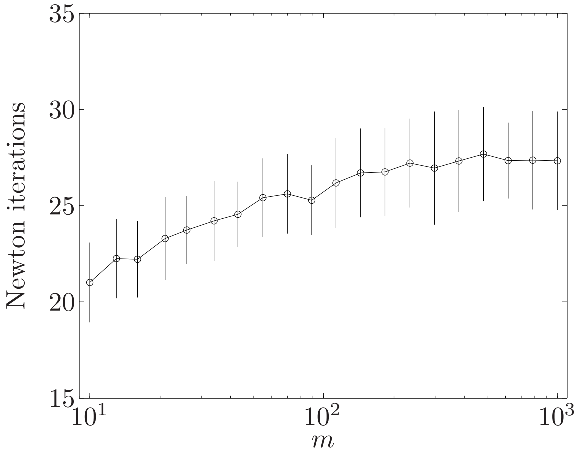

Figure 11.8 Average number of Newton steps required to solve 100 randomly generated LPs of different dimensions, with $n = 2m$. Error bars show standard deviation, around the average value, for each value of m. The growth in the number of Newton steps required, as the problem dimensions range over a 100 : 1 ratio, is very small.

11.3.3 Convergence analysis

Convergence analysis for the barrier method is straightforward. Assuming that $tf_0 + \phi$ can be minimized by Newton's method for $t = t^{(0)}$, $\mu t^{(0)}$, $\mu^2 t^{(0)}, \ldots$, the duality gap after the initial centering step, and k additional centering steps, is $m/(\mu^k t^{(0)})$. Therefore the desired accuracy ϵ is achieved after *exactly*

$$\left\lceil \frac{\log(m/(\epsilon t^{(0)}))}{\log \mu} \right\rceil \tag{11.13}$$

centering steps, plus the initial centering step.

It follows that the barrier method works provided the centering problem (11.6) is solvable by Newton's method, for $t \geq t^{(0)}$. For the standard Newton method, it suffices that for $t \geq t^{(0)}$, the function $tf_0 + \phi$ satisfies the conditions given in §10.2.4, page 529: its initial sublevel set is closed, the associated inverse KKT matrix is bounded, and the Hessian satisfies a Lipschitz condition. (Another set of sufficient conditions, based on self-concordance, will be discussed in detail in §11.5.) If the infeasible start Newton method is used for centering, then the conditions listed in §10.3.3, page 536, are sufficient to guarantee convergence.

Assuming that $f_0, \ldots, f_m$ are closed, a simple modification of the original problem ensures that these conditions hold. By adding a constraint of the form $\|x\|_2^2 \leq R^2$ to the problem, it follows that $tf_0 + \phi$ is strongly convex, for every $t \geq 0$; in particular convergence of Newton's method, for the centering steps, is guaranteed. (See exercise 11.4.)

While this analysis shows that the barrier method does converge, under reasonable assumptions, it does not address a basic question: As the parameter t increases, do the centering problems become more difficult (and therefore take more and more iterations)? Numerical evidence suggests that for a wide variety of problems, this is not the case; the centering problems appear to require a nearly constant number of Newton steps to solve, even as t increases. We will see (in §11.5) that this issue can be resolved, for problems that satisfy certain self-concordance conditions.

11.3.4 Newton step for modified KKT equations

In the barrier method, the Newton step $\Delta x_{\rm nt}$, and associated dual variable are given by the linear equations

$$\left[\begin{array}{cc} t\nabla^2 f_0(x) + \nabla^2\phi(x) & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} \Delta x_{\rm nt} \\ \nu_{\rm nt} \end{array}\right] = -\left[\begin{array}{c} t\nabla f_0(x) + \nabla\phi(x) \\ 0 \end{array}\right]. \tag{11.14}$$

In this section we show how these Newton steps for the centering problem can be interpreted as Newton steps for directly solving the modified KKT equations

$$\begin{array}{rcl} \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + A^T\nu & = & 0 \\ -\lambda_i f_i(x) & = & 1/t, \quad i = 1, \ldots, m \\ Ax & = & b \end{array} \tag{11.15}$$

in a particular way.

To solve the modified KKT equations (11.15), which is a set of $n+p+m$ nonlinear equations in the $n+p+m$ variables x, ν, and λ, we first eliminate the variables λ_i, using $\lambda_i = -1/(tf_i(x))$. This yields

$$\nabla f_0(x) + \sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla f_i(x) + A^T\nu = 0, \qquad Ax = b, \tag{11.16}$$

which is a set of $n+p$ equations in the $n+p$ variables x and ν.

To find the Newton step for solving the set of nonlinear equations (11.16), we form the Taylor approximation for the nonlinear term occurring in the first equation. For v small, we have the Taylor approximation

$$\begin{array}{l} \nabla f_0(x+v) + \displaystyle\sum_{i=1}^m \frac{1}{-tf_i(x+v)} \nabla f_i(x+v) \\ \quad \approx \quad \nabla f_0(x) + \displaystyle\sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla f_i(x) + \nabla^2 f_0(x)v \\ \qquad + \displaystyle\sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla^2 f_i(x)v + \sum_{i=1}^m \frac{1}{tf_i(x)^2} \nabla f_i(x)\nabla f_i(x)^T v. \end{array}$$

The Newton step is obtained by replacing the nonlinear term in equation (11.16) by this Taylor approximation, which yields the linear equations

$$Hv + A^T\nu = -g, \qquad Av = 0, \tag{11.17}$$

where

$$\begin{array}{rcl} H & = & \nabla^2 f_0(x) + \displaystyle\sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla^2 f_i(x) + \sum_{i=1}^m \frac{1}{tf_i(x)^2} \nabla f_i(x)\nabla f_i(x)^T \\ g & = & \nabla f_0(x) + \displaystyle\sum_{i=1}^m \frac{1}{-tf_i(x)} \nabla f_i(x). \end{array}$$

Now we observe that

$$H = \nabla^2 f_0(x) + (1/t)\nabla^2\phi(x), \qquad g = \nabla f_0(x) + (1/t)\nabla\phi(x),$$

so, from (11.14), the Newton steps $\Delta x_{\rm nt}$ and $\nu_{\rm nt}$ in the barrier method centering step satisfy

$$tH\Delta x_{\rm nt} + A^T\nu_{\rm nt} = -tg, \qquad A\Delta x_{\rm nt} = 0.$$

Comparing this with (11.17) shows that

$$v = \Delta x_{\rm nt}, \qquad \nu = (1/t)\nu_{\rm nt}.$$

This shows that the Newton step for the centering problem (11.6) can be interpreted, after scaling the dual variable, as the Newton step for solving the modified KKT equations (11.16).

In this approach, we first eliminated the variable λ from the modified KKT equations, and then applied Newton's method to solve the resulting set of equations. Another variation on this approach is to directly apply Newton's method to the modified KKT equations, without first eliminating λ. This method yields the so-called *primal-dual search directions*, discussed in §11.7.

11.4 Feasibility and phase I methods

The barrier method requires a strictly feasible starting point $x^{(0)}$. When such a point is not known, the barrier method is preceded by a preliminary stage, called *phase I*, in which a strictly feasible point is computed (or the constraints are found to be infeasible). The strictly feasible point found during phase I is then used as the starting point for the barrier method, which is called the *phase II* stage. In this section we describe several phase I methods.

11.4.1 Basic phase I method

We consider a set of inequalities and equalities in the variables $x \in \mathbf{R}^n$,

$$f_i(x) \leq 0, \quad i = 1, \ldots, m, \qquad Ax = b, \tag{11.18}$$

where $f_i : \mathbf{R}^n \to \mathbf{R}$ are convex, with continuous second derivatives. We assume that we are given a point $x^{(0)} \in \mathbf{dom}\, f_1 \cap \cdots \cap \mathbf{dom}\, f_m$, with $Ax^{(0)} = b$.

Our goal is to find a strictly feasible solution of these inequalities and equalities, or determine that none exists. To do this we form the following optimization problem:

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \ldots, m \\ & Ax = b \end{array} \tag{11.19}$$

in the variables $x \in \mathbf{R}^n$, $s \in \mathbf{R}$. The variable s can be interpreted as a bound on the maximum infeasibility of the inequalities; the goal is to drive the maximum infeasibility below zero.

This problem is always strictly feasible, since we can choose $x^{(0)}$ as starting point for x, and for s, we can choose any number larger than $\max_{i=1,\ldots,m} f_i(x^{(0)})$. We can therefore apply the barrier method to solve the problem (11.19), which is called the *phase I optimization problem* associated with the inequality and equality system (11.19).

We can distinguish three cases depending on the sign of the optimal value $\bar{p}^\star$ of (11.19).

1. If $\bar{p}^\star < 0$, then (11.18) has a strictly feasible solution. Moreover if (x, s) is feasible for (11.19) with $s < 0$, then x satisfies $f_i(x) < 0$. This means we do not need to solve the optimization problem (11.19) with high accuracy; we can terminate when $s < 0$.

2. If $\bar{p}^\star > 0$, then (11.18) is infeasible. As in case 1, we do not need to solve the phase I optimization problem (11.19) to high accuracy; we can terminate when a dual feasible point is found with positive dual objective (which proves that $\bar{p}^\star > 0$). In this case, we can construct the alternative that proves (11.18) is infeasible from the dual feasible point.

3. If $\bar{p}^\star = 0$ and the minimum is attained at $x^\star$ and $s^\star = 0$, then the set of inequalities is feasible, but not strictly feasible. If $\bar{p}^\star = 0$ and the minimum is not attained, then the inequalities are infeasible.

In practice it is impossible to determine exactly that $\bar{p}^\star = 0$. Instead, an optimization algorithm applied to (11.19) will terminate with the conclusion that $|\bar{p}^\star| < \epsilon$ for some small, positive ϵ. This allows us to conclude that the inequalities $f_i(x) \leq -\epsilon$ are infeasible, while the inequalities $f_i(x) \leq \epsilon$ are feasible.

Sum of infeasibilities

There are many variations on the basic phase I method just described. One method is based on minimizing the sum of the infeasibilities, instead of the maximum infeasibility. We form the problem

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T s \\ \text{subject to} & f_i(x) \leq s_i, \quad i = 1, \ldots, m \\ & Ax = b \\ & s \succeq 0. \end{array} \tag{11.20}$$

For fixed x, the optimal value of s_i is $\max\{f_i(x), 0\}$, so in this problem we are minimizing the sum of the infeasibilities. The optimal value of (11.20) is zero and achieved if and only if the original set of equalities and inequalities is feasible.

This sum of infeasibilities phase I method has a very interesting property when the system of equalities and inequalities (11.19) is infeasible. In this case, the optimal point for the phase I problem (11.20) often violates only a small number, say r, of the inequalities. Therefore, we have computed a point that satisfies many $(m - r)$ of the inequalities, *i.e.*, we have identified a large subset of inequalities that is feasible. In this case, the dual variables associated with the strictly satisfied inequalities are zero, so we have also proved infeasibility of a subset of the inequalities. This is more informative than finding that the m inequalities, together, are mutually infeasible. (This phenomenon is closely related to ℓ_1-norm regularization, or basis pursuit, used to find sparse approximate solutions; see §6.1.2 and §6.5.4).

Example 11.4 *Comparison of phase I methods.* We apply two phase I methods to an infeasible set of inequalities $Ax \preceq b$ with dimensions $m = 100$, $n = 50$. The first method is the basic phase I method

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & Ax \preceq b + \mathbf{1}s, \end{array}$$

which minimizes the maximum infeasibility. The second method minimizes the sum of the infeasibilities, *i.e.*, solves the LP

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T s \\ \text{subject to} & Ax \preceq b + s \\ & s \succeq 0. \end{array}$$

Figure 11.9 shows the distributions of the infeasibilities $b_i - a_i^T x$ for these two values of x, denoted x_{max} and x_{sum}, respectively. The point x_{max} satisfies 39 of the 100 inequalities, whereas the point x_{sum} satisfies 79 of the inequalities.

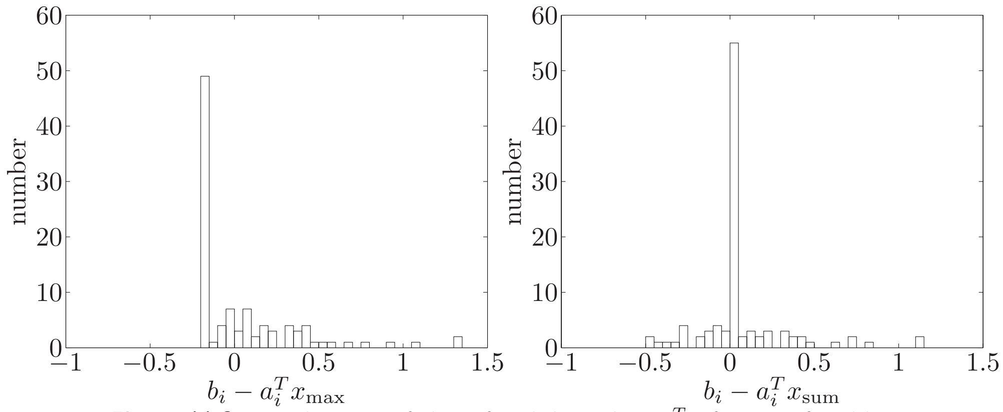

Figure 11.9 Distributions of the infeasibilities $b_i - a_i^T x$ for an infeasible set of 100 inequalities $a_i^T x \leq b_i$, with 50 variables. The vector $x_{\max}$ used in the left plot was obtained by the basic phase I algorithm. It satisfies 39 of the 100 inequalities. In the right plot the vector $x_{\rm sum}$ was obtained by minimizing the sum of the infeasibilities. This vector satisfies 79 of the 100 inequalities.

Termination near the phase II central path

A simple variation on the basic phase I method, using the barrier method, has the property that (when the equalities and inequalities are strictly feasible) the central path for the phase I problem intersects the central path for the original optimization problem (11.1).

We assume a point $x^{(0)} \in \mathcal{D} = \textbf{dom}\, f_0 \cap \textbf{dom}\, f_1 \cap \cdots \cap \textbf{dom}\, f_m$, with $Ax^{(0)} = b$ is given. We form the phase I optimization problem

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \ldots, m \\ & f_0(x) \leq M \\ & Ax = b, \end{array} \qquad (11.21)$$

where M is a constant chosen to be larger than $\max\{f_0(x^{(0)}), p^\star\}$.

We assume now that the original problem (11.1) is strictly feasible, so the optimal value $\bar{p}^\star$ of (11.21) is negative. The central path of (11.21) is characterized by

$$\sum_{i=1}^m \frac{1}{s - f_i(x)} = \bar{t}, \qquad \frac{1}{M - f_0(x)} \nabla f_0(x) + \sum_{i=1}^m \frac{1}{s - f_i(x)} \nabla f_i(x) + A^T \nu = 0,$$

where $\bar{t}$ is the parameter. If (x, s) is on the central path and $s = 0$, then x and ν satisfy

$$t \nabla f_0(x) + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x) + A^T \nu = 0$$

for $t = 1/(M - f_0(x))$. This means that x is on the central path for the original optimization problem (11.1), with associated duality gap

$$m(M - f_0(x)) \leq m(M - p^\star). \tag{11.22}$$

11.4.2 Phase I via infeasible start Newton method

We can also carry out the phase I stage using an infeasible start Newton method, applied to a modified version of the original problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & Ax = b. \end{array}$$

We first express the problem in the (obviously equivalent) form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \dots, m \\ & Ax = b, \quad s = 0, \end{array}$$

with the additional variable $s \in \mathbf{R}$. To start the barrier method, we use an infeasible start Newton method to solve

$$\begin{array}{ll} \text{minimize} & t^{(0)} f_0(x) - \sum_{i=1}^m \log(s - f_i(x)) \\ \text{subject to} & Ax = b, \quad s = 0. \end{array}$$

This can be initialized with any $x \in \mathcal{D}$, and any $s > \max_i f_i(x)$. Provided the problem is strictly feasible, the infeasible start Newton method will eventually take an undamped step, and thereafter we will have $s = 0$, *i.e.*, x strictly feasible.

The same trick can be applied if a point in $\mathcal{D}$, the common domain of the functions, is not known. We simply apply the infeasible start Newton method to the problem

$$\begin{array}{ll} \text{minimize} & t^{(0)} f_0(x + z_0) - \sum_{i=1}^m \log(s - f_i(x + z_i)) \\ \text{subject to} & Ax = b, \quad s = 0, \quad z_0 = 0, \quad \dots, \quad z_m = 0 \end{array}$$

with variables x, $z_0, \dots, z_m$, and $s \in \mathbf{R}$. We initialize z_i so that $x + z_i \in \mathbf{dom}\, f_i$.

The main disadvantage of this approach to the phase I problem is that there is no good stopping criterion when the problem is infeasible; the residual simply fails to converge to zero.

11.4.3 Examples

We consider a family of linear feasibility problems,

$$Ax \preceq b(\gamma)$$

where $A \in \mathbf{R}^{50 \times 20}$ and $b(\gamma) = b + \gamma \Delta b$. The problem data are chosen so that the inequalities are strictly feasible for $\gamma > 0$ and infeasible for $\gamma < 0$. For $\gamma = 0$ the problem is feasible but not strictly feasible.

Figure 11.10 shows the total number of Newton steps required to find a strictly feasible point, or a certificate of infeasibility, for 40 values of γ in $[-1, 1]$. We use the basic phase I method of §11.4.1, *i.e.*, for each value of γ, we form the LP

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & Ax \preceq b(\gamma) + s\mathbf{1}. \end{array}$$

The barrier method is used with $\mu = 10$, and starting point $x = 0$, $s = -\min_i b_i(\gamma) + 1$. The method terminates when a point (x, s) with $s < 0$ is found, or a feasible solution z of the dual problem

$$\begin{array}{ll} \text{maximize} & -b(\gamma)^T z \\ \text{subject to} & A^T z = 0 \\ & \mathbf{1}^T z = 1 \\ & z \succeq 0 \end{array}$$

is found with $-b(\gamma)^T z > 0$.

The plot shows that when the inequalities are feasible, with some margin, it takes around 25 Newton steps to produce a strictly feasible point. Conversely, when the inequalities are infeasible, again with some margin, it takes around 35 steps to produce a certificate proving infeasibility. The phase I effort increases as the set of inequalities approaches the boundary between feasible and infeasible, *i.e.*, γ near zero. When γ is very near zero, so the inequalities are very near the boundary between feasible and infeasible, the number of steps grows substantially. Figure 11.11 shows the total number of Newton steps required for values of γ near zero. The plots show an approximately logarithmic increase in the number of steps required to detect feasibility, or prove infeasibility, for problems very near the boundary between feasible and infeasible.

This example is typical: The cost of solving a set of convex inequalities and linear equalities using the barrier method is modest, and approximately constant, as long as the problem is not very close to the boundary between feasibility and infeasibility. When the problem is very close to the boundary, the number of Newton steps required to find a strictly feasible point or produce a certificate of infeasibility grows. When the problem is *exactly* on the boundary between strictly feasible and infeasible, for example, feasible but not strictly feasible, the cost becomes infinite.

Feasibility using infeasible start Newton method

We also solve the same set of feasibility problems using the infeasible start Newton method, applied to the problem

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^m \log s_i \\ \text{subject to} & Ax + s = b(\gamma). \end{array}$$

We use backtracking parameters $\alpha = 0.01$, $\beta = 0.9$, and initialize with $x^{(0)} = 0$, $s^{(0)} = \mathbf{1}$, $\nu^{(0)} = 0$. We consider only feasible problems (*i.e.*, $\gamma > 0$) and terminate once a feasible point is found. (We do not consider infeasible problems, since in that case the residual simply converges to a positive number.) Figure 11.12 shows the number of Newton steps required to find a feasible point, as a function of γ.

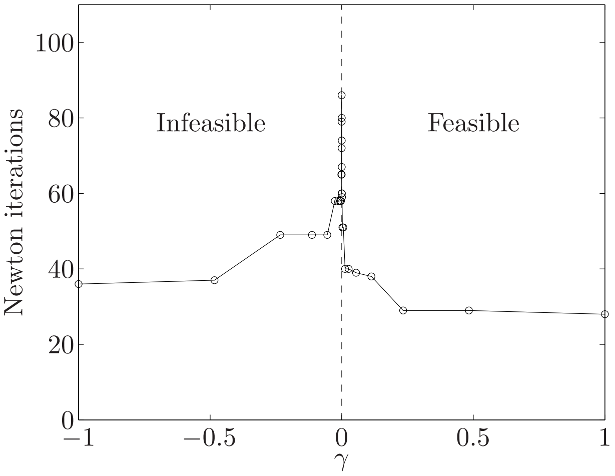

Figure 11.10 Number of Newton iterations required to detect feasibility or infeasibility of a set of linear inequalities $Ax \preceq b + \gamma\Delta b$ parametrized by $\gamma \in \mathbf{R}$. The inequalities are strictly feasible for $\gamma > 0$, and infeasible for $\gamma < 0$. For γ larger than around 0.2, about 30 steps are required to compute a strictly feasible point; for γ less than -0.5 or so, it takes around 35 steps to produce a certificate proving infeasibility. For values of γ in between, and especially near zero, more Newton steps are required to determine feasibility.

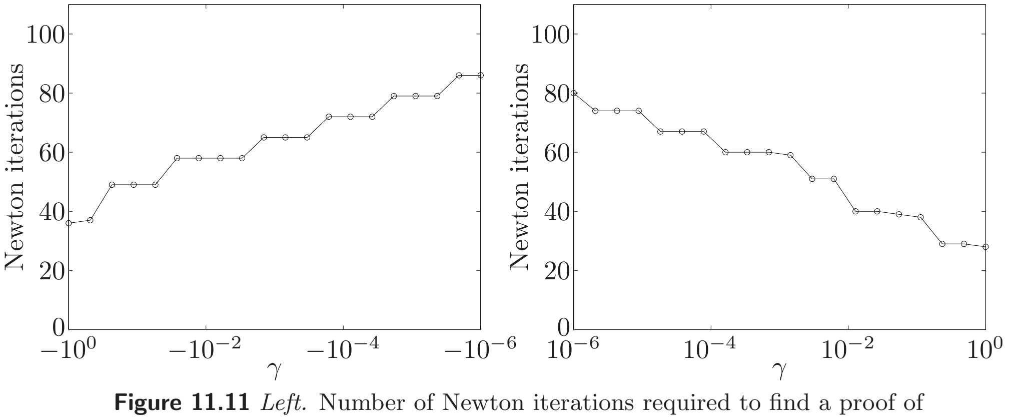

Figure 11.11 *Left.* Number of Newton iterations required to find a proof of infeasibility versus γ, for γ small and negative. *Right.* Number of Newton iterations required to find a strictly feasible point versus γ, for γ small and positive.

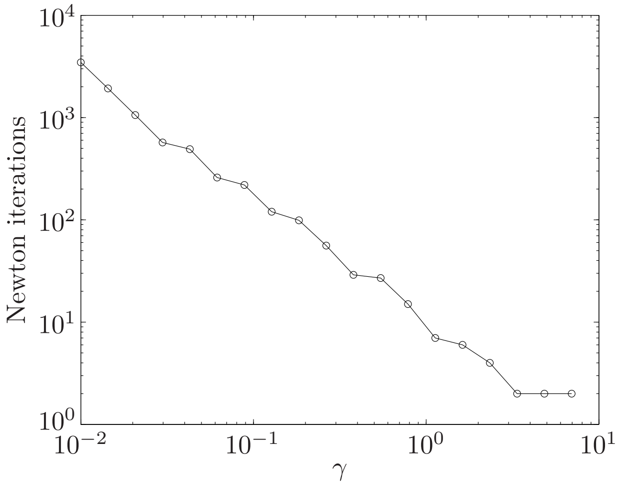

Figure 11.12 Number of iterations required to find a feasible point for a set of linear inequalities $Ax \preceq b + \gamma\Delta b$ parametrized by $\gamma \in \mathbf{R}$. The infeasible start Newton method is used, and terminated when a feasible point is found. For $\gamma = 10$, the starting point $x^{(0)} = 0$ happened to be feasible (0 iterations).

The plot shows that for γ larger than 0.3 or so, it takes fewer than 20 Newton steps to find a feasible point. In these cases the method is more efficient than a phase I method, which takes a total of around 30 Newton steps. For smaller values of γ, the number of Newton steps required grows dramatically, approximately as $1/\gamma$. For $\gamma = 0.01$, the infeasible start Newton method requires several thousand iterations to produce a feasible point. In this region the phase I approach is far more efficient, requiring only 40 iterations or so.

These results are quite typical. The infeasible start Newton method works very well provided the inequalities are feasible, and not very close to the boundary between feasible and infeasible. But when the feasible set is just barely nonempty (as is the case in this example with small γ), a phase I method is far better. Another advantage of the phase I method is that it gracefully handles the infeasible case; the infeasible start Newton method, in contrast, simply fails to converge.

11.5 Complexity analysis via self-concordance

Using the complexity analysis of Newton's method for self-concordant functions (§9.6.4, page 503, and §10.2.4, page 531), we can give a complexity analysis of the barrier method. The analysis applies to many common problems, and leads to several interesting conclusions: It gives a rigorous bound on the total number of Newton steps required to solve a problem using the barrier method, and it justifies our observation that the centering problems do not become more difficult as t increases.

11.5.1 Self-concordance assumption

We make two assumptions.

- The function $tf_0 + \phi$ is closed and self-concordant for all $t \geq t^{(0)}$.
- The sublevel sets of (11.1) are bounded.

The second assumption implies that the centering problem has bounded sublevel sets (see exercise 11.3), and, therefore, the centering problem is solvable. The bounded sublevel set assumption also implies that the Hessian of $tf_0 + \phi$ is positive definite everywhere (see exercise 11.14). While the self-concordance assumption restricts the complexity analysis to a particular class of problems, it is important to emphasize that the barrier method works well in general, whether or not the self-concordance assumption holds.

The self-concordance assumption holds for a variety of problems, including all linear and quadratic problems. If the functions f_i are linear or quadratic, then

$$tf_0 - \sum_{i=1}^{m} \log(-f_i)$$

is self-concordant for all values of $t \geq 0$ (see §9.6). The complexity analysis given below therefore applies to LPs, QPs, and QCQPs.

In other cases, it is possible to reformulate the problem so the assumption of self-concordance holds. As an example, consider the linear inequality constrained entropy maximization problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g \\ & Ax = b. \end{array}$$

The function

$$tf_0(x) + \phi(x) = t\sum_{i=1}^{n} x_i \log x_i - \sum_{i=1}^{m} \log(g_i - f_i^T x),$$

where $f_1^T, \ldots, f_m^T$ are the rows of F, is not closed (unless $Fx \preceq g$ implies $x \succeq 0$), or self-concordant. We can, however, add the redundant inequality constraints $x \succeq 0$ to obtain the equivalent problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^n x_i \log x_i \\ \text{subject to} & Fx \preceq g \\ & Ax = b \\ & x \succeq 0. \end{array} \qquad (11.23)$$

For this problem we have

$$tf_0(x) + \phi(x) = t\sum_{i=1}^{n} x_i \log x_i - \sum_{i=1}^{n} \log x_i - \sum_{i=1}^{m} \log(g_i - f_i^T x),$$

which *is* self-concordant and closed, for any $t \geq 0$. (The function $ty \log y - \log y$ is self-concordant on $\mathbf{R}_{++}$, for all $t \geq 0$; see exercise 11.13.) The complexity analysis therefore applies to the reformulated linear inequality constrained entropy maximization problem (11.23).

As a more exotic example, consider the GP

$$\begin{array}{ll} \text{minimize} & f_0(x) = \log\left(\sum_{k=1}^{K_0} \exp(a_{0k}^T x + b_{0k})\right) \\ \text{subject to} & \log\left(\sum_{k=1}^{K_i} \exp(a_{ik}^T x + b_{ik})\right) \leq 0, \quad i = 1, \dots, m. \end{array}$$

It is not clear whether or not the function

$$tf_0(x) + \phi(x) = t \log\left(\sum_{k=1}^{K_0} \exp(a_{0k}^T x + b_{0k})\right) - \sum_{i=1}^{m} \log\left(-\log \sum_{k=1}^{K_i} \exp(a_{ik}^T x + b_{ik})\right)$$

is self-concordant, so although the barrier method works, the complexity analysis of this section need not hold.

We can, however, reformulate the GP in a form that definitely satisfies the self-concordance assumption. For each (monomial) term $\exp(a_{ik}^T x + b_{ik})$ we introduce a new variable y_{ik} that serves as an upper bound,

$$\exp(a_{ik}^T x + b_{ik}) \leq y_{ik}.$$

Using these new variables we can express the GP in the form

$$\begin{array}{ll} \text{minimize} & \sum_{k=1}^{K_0} y_{0k} \\ \text{subject to} & \sum_{k=1}^{K_i} y_{ik} \leq 1, \quad i = 1, \dots, m \\ & a_{ik}^T x + b_{ik} - \log y_{ik} \leq 0, \quad i = 0, \dots, m, \quad k = 1, \dots, K_i \\ & y_{ik} \geq 0, \quad i = 0, \dots, m, \quad k = 1, \dots, K_i. \end{array}$$

The associated logarithmic barrier is

$$\sum_{i=0}^{m} \sum_{k=1}^{K_i} \left(-\log y_{ik} - \log(\log y_{ik} - a_{ik}^T x - b_{ik})\right) - \sum_{i=1}^{m} \log\left(1 - \sum_{k=1}^{K_i} y_{ik}\right),$$

which is closed and self-concordant (example 9.8, page 500). Since the objective is linear, it follows that $tf_0 + \phi$ is closed and self-concordant for any t.

11.5.2 Newton iterations per centering step

The complexity theory of Newton's method for self-concordant functions, developed in §9.6.4 (page 503) and §10.2.4 (page 531), shows that the number of Newton iterations required to minimize a closed strictly convex self-concordant function f is bounded above by

$$\frac{f(x) - p^\star}{\gamma} + c. \tag{11.24}$$

Here x is the starting point for Newton's method, and $p^\star = \inf_x f(x)$ is the optimal value. The constant γ depends only on the backtracking parameters α and β, and is given by

$$\frac{1}{\gamma} = \frac{20-8\alpha}{\alpha\beta(1-2\alpha)^2}.$$

The constant c depends only on the tolerance $\epsilon_{\rm nt}$,

$$c = \log_2 \log_2(1/\epsilon_{\rm nt}),$$

and can reasonably be approximated as $c = 6$. The expression (11.24) is a quite conservative bound on the number of Newton steps required, but our interest in this section is only to establish a complexity bound, concentrating on how it increases with problem size and algorithm parameters.

In this section we use this result to derive a bound on the number of Newton steps required for one outer iteration of the barrier method, *i.e.*, for computing $x^\star(\mu t)$, starting from $x^\star(t)$. To lighten the notation we use x to denote $x^\star(t)$, the current iterate, and we use x^+ to denote $x^\star(\mu t)$, the next iterate. We use λ and ν to denote $\lambda^\star(t)$ and $\nu^\star(t)$, respectively.

The self-concordance assumption implies that

$$\frac{\mu t f_0(x) + \phi(x) - \mu t f_0(x^+) - \phi(x^+)}{\gamma} + c \qquad (11.25)$$

is an upper bound on the number of Newton steps required to compute $x^+ = x^\star(\mu t)$, starting at $x = x^\star(t)$. Unfortunately we do not know x^+, and hence the upper bound (11.25), until we actually compute x^+, *i.e.*, carry out the Newton algorithm (whereupon we know the *exact* number of Newton steps required to compute $x^\star(\mu t)$, which defeats the purpose). We can, however, derive an upper bound on (11.25), as follows:

$$\begin{array}{ll} \multicolumn{2}{l}{\mu t f_0(x) + \phi(x) - \mu t f_0(x^+) - \phi(x^+)} \\ = & \mu t f_0(x) - \mu t f_0(x^+) + \displaystyle\sum_{i=1}^m \log(-\mu t \lambda_i f_i(x^+)) - m \log \mu \\ \leq & \mu t f_0(x) - \mu t f_0(x^+) - \mu t \displaystyle\sum_{i=1}^m \lambda_i f_i(x^+) - m - m\log\mu \\ = & \mu t f_0(x) - \mu t \left(f_0(x^+) + \displaystyle\sum_{i=1}^m \lambda_i f_i(x^+) + \nu^T(Ax^+ - b) \right) - m - m \log \mu \\ \leq & \mu t f_0(x) - \mu t g(\lambda, \nu) - m - m \log \mu \\ = & m(\mu - 1 - \log \mu). \end{array}$$

This chain of equalities and inequalities needs some explanation. To obtain the second line from the first, we use $\lambda_i = -1/(t f_i(x))$. In the first inequality we use the fact that $\log a \leq a - 1$ for $a > 0$. To obtain the fourth line from the third, we use $Ax^+ = b$, so the extra term $\nu^T(Ax^+ - b)$ is zero. The second inequality follows

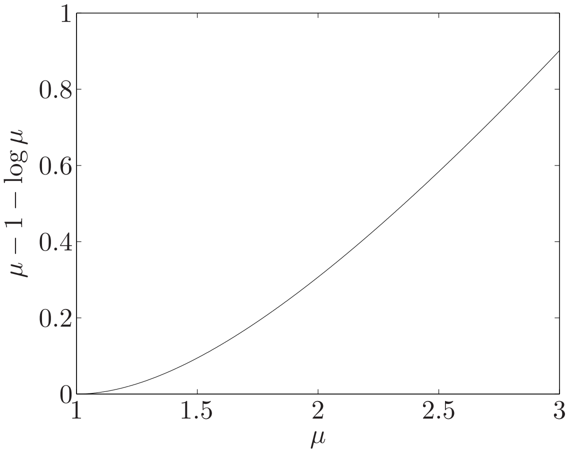

Figure 11.13 The function $\mu - 1 - \log \mu$, versus μ. The number of Newton steps required for one outer iteration of the barrier method is bounded by $(m/\gamma)(\mu - 1 - \log \mu) + c$.

from the definition of the dual function:

$$\begin{aligned} g(\lambda, \nu) &= \inf_z \left(f_0(z) + \sum_{i=1}^m \lambda_i f_i(z) + \nu^T(Az - b) \right) \\ &\leq f_0(x^+) + \sum_{i=1}^m \lambda_i f_i(x^+) + \nu^T(Ax^+ - b). \end{aligned}$$

The last line follows from $g(\lambda, \nu) = f_0(x) - m/t$.

The conclusion is that

$$\frac{m(\mu - 1 - \log \mu)}{\gamma} + c \qquad (11.26)$$

is an upper bound on (11.25), and therefore an upper bound on the number of Newton steps required for one outer iteration of the barrier method. The function $\mu - 1 - \log \mu$ is shown in figure 11.13. For small μ it is approximately quadratic; for large μ it grows approximately linearly. This fits with our intuition that for μ near one, the number of Newton steps required to center is small, whereas for large μ, it could well grow.

The bound (11.26) shows that the number of Newton steps required in each centering step is bounded by a quantity that depends mostly on μ, the factor by which t is updated in each outer step of the barrier method, and m, the number of inequality constraints in the problem. It also depends, weakly, on the parameters α and β used in the line search for the inner iterations, and in a very weak way on the tolerance used to terminate the inner iterations. It is interesting to note that the bound does not depend on n, the dimension of the variable, or p, the number of equality constraints, or the particular values of the problem data, *i.e.*, the objective and constraint functions (provided the self-concordance assumption in §11.5.1 holds). Finally, we note that it does not depend on t; in particular, as $t \to \infty$, a uniform bound on the number of Newton steps per outer iteration holds.

11.5.3 Total number of Newton iterations

We can now give an upper bound on the total number of Newton steps in the barrier method, not counting the initial centering step (which we will analyze later, as part of phase I). We multiply (11.26), which bounds the number of Newton steps per outer iteration, by (11.13), the number of outer steps required, to obtain

$$N = \left\lceil \frac{\log(m/(t^{(0)}\epsilon))}{\log \mu} \right\rceil \left(\frac{m(\mu - 1 - \log \mu)}{\gamma} + c \right), \tag{11.27}$$

an upper bound on the total number of Newton steps required. This formula shows that when the self-concordance assumption holds, we can bound the number of Newton steps required by the barrier method, for any value of $\mu > 1$.

If we fix μ and m, the bound N is proportional to $\log(m/(t^{(0)}\epsilon))$, which is the log of the ratio of the initial duality gap $m/t^{(0)}$ to the final duality gap ϵ, *i.e.*, the log of the required duality gap reduction. We can therefore say that the barrier method converges at least linearly, since the number of steps required to reach a given precision grows logarithmically with the inverse of the precision.

If μ, and the required duality gap reduction factor, are fixed, the bound N grows linearly with m, the number of inequalities. The bound N is independent of the other problem dimensions n and p, and the particular problem data or functions. We will see below that by a particular choice of μ, that depends on m, we can obtain a bound on the number of Newton steps that grows only as $\sqrt{m}$, instead of as m.

Finally, we analyze the bound N as a function of the algorithm parameter μ. As μ approaches one, the first term in N grows large, and therefore so does N. This is consistent with our intuition and observation that for μ near one, the number of outer iterations is very large. As μ becomes large, the bound N grows approximately as $\mu/\log \mu$, this time because the bound on the number of Newton iterations required per outer iteration grows. This, too, is consistent with our observations. As a result, the bound N has a minimum value as a function of μ.

The variation of the bound with the parameter μ is illustrated in figure 11.14, which shows the bound (11.27) versus μ for the values

$$c = 6, \qquad \gamma = 1/375, \qquad m/(t^{(0)}\epsilon) = 10^5, \qquad m = 100.$$

The bound is qualitatively consistent with intuition, and our observations: it grows very large as μ approaches one, and increases, more slowly, as μ becomes large. The bound N has a minimum at $\mu \approx 1.02$, which gives a bound on the total number of Newton iterations around 8000. The complexity analysis of Newton's method is conservative, but the basic trade-off in the choice of μ is reflected in the plot. (In practice, far larger values of μ, from around 2 to 100, work very well, and require a total number of Newton iterations on the order of a few tens.)

Choosing μ as a function of m

When μ (and the required duality gap reduction) is fixed, the bound (11.27) grows linearly with m, the number of inequalities. It turns out we can obtain a better

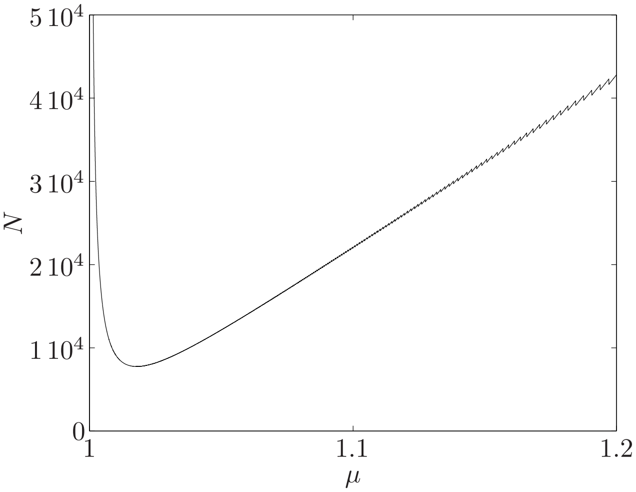

Figure 11.14 The upper bound N on the total number of Newton iterations, given by equation (11.27), for $c = 6$, $\gamma = 1/375$, $m = 100$, and a duality gap reduction factor $m/(t^{(0)}\epsilon) = 10^5$, versus the barrier algorithm parameter μ.

exponent for m by making μ a function of m. Suppose we choose

$$\mu = 1 + 1/\sqrt{m}. \tag{11.28}$$

Then we can bound the second term in (11.27) as

$$\begin{array}{rcl} \mu - 1 - \log \mu & = & 1/\sqrt{m} - \log(1 + 1/\sqrt{m}) \\ & \leq & 1/\sqrt{m} - 1/\sqrt{m} + 1/(2m) \\ & = & 1/(2m) \end{array}$$

(using $-\log(1+a) \leq -a + a^2/2$ for $a \geq 0$). Using concavity of the logarithm, we also have

$$\log \mu = \log(1 + 1/\sqrt{m}) \geq (\log 2)/\sqrt{m}.$$

Using these inequalities we can bound the total number of Newton steps by

$$\begin{array}{rcl} N & \leq & \left\lceil \dfrac{\log(m/(t^{(0)}\epsilon))}{\log \mu} \right\rceil \left(\dfrac{m(\mu - 1 - \log \mu)}{\gamma} + c \right) \\ & \leq & \left\lceil \sqrt{m} \dfrac{\log(m/(t^{(0)}\epsilon))}{\log 2} \right\rceil \left(\dfrac{1}{2\gamma} + c \right) \\ & = & \left\lceil \sqrt{m} \log_2(m/(t^{(0)}\epsilon)) \right\rceil \left(\dfrac{1}{2\gamma} + c \right) \\ & \leq & c_1 + c_2\sqrt{m}, \end{array} \tag{11.29}$$

where

$$c_1 = \frac{1}{2\gamma} + c, \qquad c_2 = \log_2(m/(t^{(0)}\epsilon)) \left(\frac{1}{2\gamma} + c \right).$$

Here c_1 depends (and only weakly) on algorithm parameters for the centering Newton steps, and c_2 depends on these and the required duality gap reduction. Note that the term $\log_2(m/(t^{(0)}\epsilon))$ is exactly the number of bits of required duality gap reduction.

For fixed duality gap reduction, the bound (11.29) grows as $\sqrt{m}$, whereas the bound N in (11.27) grows like m, if the parameter μ is held constant. For this reason the barrier method, with parameter value (11.28), is said to be an *order* $\sqrt{m}$ method.

In practice, we would not use the value $\mu = 1 + 1/\sqrt{m}$, which is far too small, or even decrease μ as a function of m. Our only interest in this value of μ is that it (approximately) minimizes our (very conservative) upper bound on the number of Newton steps, and yields an overall estimate that grows as $\sqrt{m}$, instead of m.

11.5.4 Feasibility problems

In this section we analyze the complexity of a (minor) variation on the basic phase I method described in §11.4.1, used to solve a set of convex inequalities,

$$f_1(x) \le 0, \quad \ldots, \quad f_m(x) \le 0, \tag{11.30}$$

where $f_1, \ldots, f_m$ are convex, with continuous second derivatives. (We will consider equality constraints later.) We assume that the phase I problem

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \le s, \quad i = 1, \ldots, m \end{array} \tag{11.31}$$

satisfies the conditions in §11.5.1. In particular we assume that the feasible set of the inequalities (11.30) (which of course can be empty) is contained in a Euclidean ball of radius R:

$$\{x \mid f_i(x) \le 0,\ i = 1, \ldots, m\} \subseteq \{x \mid \|x\|_2 \le R\}.$$

We can interpret R as a prior bound on the norm of any point in the feasible set of the inequalities. This assumption implies that the sublevel sets of the phase I problem are bounded. Without loss of generality, we will start the phase I method at the point $x = 0$. We define $F = \max_i f_i(0)$, which is the maximum constraint violation, assumed to be positive (since otherwise $x = 0$ satisfies the inequalities (11.30)).

We define $\bar{p}^\star$ as the optimal value of the phase I optimization problem (11.31). The sign of $\bar{p}^\star$ determines whether or not the set of inequalities (11.30) is feasible. The magnitude of $\bar{p}^\star$ also has a meaning. If $\bar{p}^\star$ is positive and large (say, near F, the largest value it can have) it means that the set of inequalities is quite infeasible, in the sense that for each x, at least one of the inequalities is substantially violated (by at least $\bar{p}^\star$). On the other hand, if $\bar{p}^\star$ is negative and large, it means that the set of inequalities is quite feasible, in the sense that there is not only an x for which $f_i(x)$ are all nonpositive, but in fact there is an x for which $f_i(x)$ are all quite negative (no more than $\bar{p}^\star$). Thus, the magnitude $|\bar{p}^\star|$ is a measure of how clearly the set of inequalities is feasible or infeasible, and therefore related to the difficulty of determining feasibility of the inequalities (11.30). In particular, if $|\bar{p}^\star|$ is small, it means the problem is near the boundary between feasibility and infeasibility.

To determine feasibility of the inequalities, we use a variation on the basic phase I problem (11.31). We add a redundant linear inequality $a^T x \leq 1$, to obtain

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \leq s, \quad i = 1, \ldots, m \\ & a^T x \leq 1. \end{array} \tag{11.32}$$

We will specify a later. Our choice will satisfy $\|a\|_2 \leq 1/R$, so $\|x\|_2 \leq R$ implies $a^T x \leq 1$, *i.e.*, the extra constraint is redundant.

We will choose a and s_0 so that $x = 0$, $s = s_0$ is on the central path of the problem (11.32), with a parameter value $t^{(0)}$, *i.e.*, they minimize

$$t^{(0)} s - \sum_{i=1}^m \log(s - f_i(x)) - \log(1 - a^T x).$$

Setting to zero the derivative with respect to s, we get

$$t^{(0)} = \sum_{i=1}^m \frac{1}{s_0 - f_i(0)}. \tag{11.33}$$

Setting to zero the gradient with respect to x yields

$$a = -\sum_{i=1}^m \frac{1}{s_0 - f_i(0)} \nabla f_i(0). \tag{11.34}$$

So it remains only to pick the parameter s_0; once we have chosen s_0, the vector a is given by (11.34), and the parameter $t^{(0)}$ is given by (11.33). Since $x = 0$ and $s = s_0$ must be strictly feasible for the phase I problem (11.32), we must choose $s_0 > F$.

We must also pick s_0 to make sure that $\|a\|_2 \leq 1/R$. From (11.34), we have

$$\|a\|_2 \leq \sum_{i=1}^m \frac{1}{s_0 - f_i(0)} \|\nabla f_i(0)\| \leq \frac{mG}{s_0 - F},$$

where $G = \max_i \|\nabla f_i(0)\|_2$. Therefore we can take $s_0 = mGR + F$, which ensures $\|a\|_2 \leq 1/R$, so the extra linear inequality is redundant.

Using (11.33), we have

$$t^{(0)} = \sum_{i=1}^m \frac{1}{mGR + F - f_i(0)} \geq \frac{1}{mGR},$$

since $F = \max_i f_i(0)$. Thus $x = 0$, $s = s_0$ are on the central path for the phase I problem (11.32), with initial duality gap

$$\frac{m+1}{t^{(0)}} \leq (m+1)mGR.$$

To solve the original inequalities (11.30) we need to determine the sign of $\bar{p}^\star$. We can stop when either the primal objective value of (11.32) is negative, or the dual objective value is positive. One of these two cases must occur when the duality gap for (11.32) is less than $|\bar{p}^\star|$.

We use the barrier method to solve (11.32), starting from a central point with duality gap no more than $(m+1)mGR$, and terminating when (or before) the duality gap is less than $|\bar{p}^\star|$. Using the results of the previous section, this requires no more than

$$\left\lceil \sqrt{m+1}\log_2 \frac{m(m+1)GR}{|\bar{p}^\star|} \right\rceil \left(\frac{1}{2\gamma} + c \right) \tag{11.35}$$

Newton steps. (Here we take $\mu = 1 + 1/\sqrt{m+1}$, which gives a better complexity exponent for m than a fixed value of μ.)

The bound (11.35) grows only slightly faster than $\sqrt{m}$, and depends weakly on the algorithm parameters used in the centering steps. It is approximately proportional to $\log_2((GR)/|\bar{p}^\star|)$, which can be interpreted as a measure of how difficult the particular feasibility problem is, or how close it is to the boundary between feasibility and infeasibility.

Feasibility problems with equality constraints

We can apply the same analysis to feasibility problems that include equality constraints, by eliminating the equality constraints. This does not affect the self-concordance of the problem, but it does mean that G and R refer to the reduced, or eliminated, problem.

11.5.5 Combined phase I/phase II complexity

In this section we give an end-to-end complexity analysis for solving the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

using (a variation on) the barrier method. First we solve the phase I problem

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \le s, \quad i = 1, \dots, m \\ & f_0(x) \le M \\ & Ax = b \\ & a^T x \le 1, \end{array}$$

which we assume satisfies the self-concordance and bounded sublevel set assumptions of §11.5.1. Here we have added two redundant inequalities to the basic phase I problem. The constraint $f_0(x) \le M$ is added to guarantee that the phase I central path intersects the central path for phase II, as described in section §11.4.1 (see (11.21)). The number M is a prior bound on the optimal value of the problem. The second added constraint is the linear inequality $a^T x \le 1$, where a is chosen as described in §11.5.4. We use the barrier method to solve this problem, with $\mu = 1 + 1/\sqrt{m+2}$, and the starting points $x = 0$, $s = s_0$ given in §11.5.4.

To either find a strictly feasible point, or determine the problem is infeasible, requires no more than

$$N_{\mathrm{I}} = \left\lceil \sqrt{m+2} \log_2 \frac{(m+1)(m+2)GR}{|\bar{p}^\star|} \right\rceil \left(\frac{1}{2\gamma} + c \right) \tag{11.36}$$

Newton steps, where G and R are as given in §11.5.4. If the problem is infeasible we are done; if it is feasible, then we find a point in phase I, associated with $s = 0$, that lies on the central path of the phase II problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \\ & a^T x \leq 1. \end{array}$$

The associated initial duality gap of this initial point is no more than $(m+1)(M - p^*)$ (see (11.22)). We assume the phase II problem also satisfies the the self-concordance and bounded sublevel set assumptions in §11.5.1.

We now proceed to phase II, again using the barrier method. We must reduce the duality gap from its initial value, which is no more than $(m+1)(M - p^*)$, to some tolerance $\epsilon > 0$. This takes at most

$$N_{\mathrm{II}} = \left\lceil \sqrt{m+1} \log_2 \frac{(m+1)(M - p^\star)}{\epsilon} \right\rceil \left(\frac{1}{2\gamma} + c \right) \tag{11.37}$$

Newton steps.

The total number of Newton steps is therefore no more than $N_{\mathrm{I}} + N_{\mathrm{II}}$. This bound grows with the number of inequalities m approximately as $\sqrt{m}$, and includes two terms that depend on the particular problem instance,

$$\log_2 \frac{GR}{|\bar{p}^\star|}, \qquad \log_2 \frac{M - p^\star}{\epsilon}.$$

11.5.6 Summary

The complexity analysis given in this section is mostly of theoretical interest. In particular, we remind the reader that the choice $\mu = 1 + 1/\sqrt{m}$, discussed in this section, would be a very poor one to use in practice; its only advantage is that it results in a bound that grows like $\sqrt{m}$ instead of m. Likewise, we do not recommend adding the redundant inequality $a^T x \leq 1$ in practice.

The actual bounds obtained from the analysis given here are far higher than the numbers of iterations actually observed. Even the order in the bound appears to be conservative. The best bounds on the number of Newton steps grow like $\sqrt{m}$, whereas practical experience suggests that the number of Newton steps hardly grows at all with m (or any other parameter, in fact).

Still, it is comforting to know that when the self-concordance condition holds, we can give a uniform bound on the number of Newton steps required in each centering step of the barrier method. An obvious potential pitfall of the barrier method is the possibility that as t grows, the associated centering problems might become more difficult, requiring more Newton steps. While practical experience suggests that this is not the case, the uniform bound bolsters our confidence that it cannot happen.

Finally, we mention that it is not yet clear whether or not there is a practical advantage to formulating a problem so that the self-concordance condition holds. All we can say is that when the self-concordance condition holds, the barrier method will work well in practice, and we can give a worst case complexity bound.

11.6 Problems with generalized inequalities

In this section we show how the barrier method can be extended to problems with generalized inequalities. We consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \preceq_{K_i} 0, \quad i=1,\ldots,m \\ & Ax=b, \end{array} \tag{11.38}$$

where $f_0 : \mathbf{R}^n \to \mathbf{R}$ is convex, $f_i : \mathbf{R}^n \to \mathbf{R}^{k_i}$, $i=1,\ldots,k$, are K_i-convex, and $K_i \subseteq \mathbf{R}^{k_i}$ are proper cones. As in §11.1, we assume that the functions f_i are twice continuously differentiable, that $A \in \mathbf{R}^{p\times n}$ with $\mathbf{rank}\, A = p$, and that the problem is solvable.

The KKT conditions for problem (11.38) are

$$\begin{array}{rcll} Ax^\star & = & b & \\ f_i(x^\star) & \preceq_{K_i} & 0, & i=1,\ldots,m \\ \lambda_i^\star & \succeq_{K_i^*} & 0, & i=1,\ldots,m \\ \nabla f_0(x^\star) + \sum_{i=1}^m Df_i(x^\star)^T\lambda_i^\star + A^T\nu^\star & = & 0 & \\ \lambda_i^{\star T} f_i(x^\star) & = & 0, & i=1,\ldots,m, \end{array} \tag{11.39}$$

where $Df_i(x^\star) \in \mathbf{R}^{k_i\times n}$ is the derivative of f_i at $x^\star$. We will assume that problem (11.38) is strictly feasible, so the KKT conditions are necessary and sufficient conditions for optimality of $x^\star$.

The development of the method is parallel to the case with scalar constraints. Once we develop a generalization of the logarithm function that applies to general proper cones, we can define a logarithmic barrier function for the problem (11.38). From that point on, the development is essentially the same as in the scalar case. In particular, the central path, barrier method, and complexity analysis are very similar.

11.6.1 Logarithmic barrier and central path

Generalized logarithm for a proper cone

We first define the analog of the logarithm, $\log x$, for a proper cone $K \subseteq \mathbf{R}^q$. We say that $\psi : \mathbf{R}^q \to \mathbf{R}$ is a *generalized logarithm* for K if

- ψ is concave, closed, twice continuously differentiable, $\mathbf{dom}\,\psi = \mathbf{int}\,K$, and $\nabla^2\psi(y) \prec 0$ for $y \in \mathbf{int}\,K$.
- There is a constant $\theta > 0$ such that for all $y \succ_K 0$, and all $s > 0$,

  $$\psi(sy) = \psi(y) + \theta \log s.$$

  In other words, ψ behaves like a logarithm along any ray in the cone K.

We call the constant θ the *degree* of ψ (since $\exp\psi$ is a homogeneous function of degree θ). Note that a generalized logarithm is only defined up to an additive constant; if ψ is a generalized logarithm for K, then so is $\psi + a$, where $a \in \mathbf{R}$. The ordinary logarithm is, of course, a generalized logarithm for $\mathbf{R}_+$.

We will use the following two properties, which are satisfied by any generalized logarithm: If $y \succ_K 0$, then

$$\nabla\psi(y) \succ_{K^*} 0, \tag{11.40}$$

which implies ψ is K-increasing (see §3.6.1), and

$$y^T\nabla\psi(y) = \theta.$$

The first property is proved in exercise 11.15. The second property follows immediately from differentiating $\psi(sy) = \psi(y) + \theta\log s$ with respect to s.

Example 11.5 *Nonnegative orthant.* The function $\psi(x) = \sum_{i=1}^n \log x_i$ is a generalized logarithm for $K = \mathbf{R}^n_+$, with degree n. For $x \succ 0$,

$$\nabla\psi(x) = (1/x_1, \ldots, 1/x_n),$$

so $\nabla\psi(x) \succ 0$, and $x^T\nabla\psi(x) = n$.

Example 11.6 *Second-order cone.* The function

$$\psi(x) = \log\left(x_{n+1}^2 - \sum_{i=1}^n x_i^2\right)$$

is a generalized logarithm for the second-order cone

$$K = \left\{ x \in \mathbf{R}^{n+1} \;\middle|\; \left(\sum_{i=1}^n x_i^2\right)^{1/2} \le x_{n+1} \right\},$$

with degree 2. The gradient of ψ at a point $x \in \mathbf{int}\, K$ is given by

$$\begin{aligned}
\frac{\partial \psi(x)}{\partial x_j} &= \frac{-2x_j}{\left(x_{n+1}^2 - \sum_{i=1}^n x_i^2\right)}, \qquad j = 1, \ldots, n \\
\frac{\partial \psi(x)}{\partial x_{n+1}} &= \frac{2x_{n+1}}{\left(x_{n+1}^2 - \sum_{i=1}^n x_i^2\right)}.
\end{aligned}$$

The identities $\nabla \psi(x) \in \mathbf{int}\, K^* = \mathbf{int}\, K$ and $x^T \nabla \psi(x) = 2$ are easily verified.

Example 11.7 *Positive semidefinite cone.* The function $\psi(X) = \log\det X$ is a generalized logarithm for the cone $\mathbf{S}_+^p$. The degree is p, since

$$\log\det(sX) = \log\det X + p \log s$$

for $s > 0$. The gradient of ψ at a point $X \in \mathbf{S}_{++}^p$ is equal to

$$\nabla \psi(X) = X^{-1}.$$

Thus, we have $\nabla \psi(X) = X^{-1} \succ 0$, and the inner product of X and $\nabla \psi(X)$ is equal to $\mathbf{tr}(XX^{-1}) = p$.

Logarithmic barrier functions for generalized inequalities

Returning to problem (11.38), let $\psi_1, \ldots, \psi_m$ be generalized logarithms for the cones $K_1, \ldots, K_m$, respectively, with degrees $\theta_1, \ldots, \theta_m$. We define the *logarithmic barrier function* for problem (11.38) as

$$\phi(x) = -\sum_{i=1}^m \psi_i(-f_i(x)), \qquad \mathbf{dom}\, \phi = \{x \mid f_i(x) \prec_{K_i} 0,\ i = 1, \ldots, m\}.$$

Convexity of ϕ follows from the fact that the functions ψ_i are K_i-increasing, and the functions f_i are K_i-convex (see the composition rule of §3.6.2).

The central path

The next step is to define the central path for problem (11.38). We define the central point $x^\star(t)$, for $t \geq 0$, as the minimizer of $tf_0 + \phi$, subject to $Ax = b$, *i.e.*, as the solution of

$$\begin{array}{ll}
\text{minimize} & tf_0(x) - \sum_{i=1}^m \psi_i(-f_i(x)) \\
\text{subject to} & Ax = b
\end{array}$$

(assuming the minimizer exists, and is unique). Central points are characterized by the optimality condition

$$\begin{aligned}
& t\nabla f_0(x) + \nabla \phi(x) + A^T \nu \\
& \quad = t\nabla f_0(x) + \sum_{i=1}^m Df_i(x)^T \nabla \psi_i(-f_i(x)) + A^T \nu = 0,
\end{aligned} \tag{11.41}$$

for some $\nu \in \mathbf{R}^p$, where $Df_i(x)$ is the derivative of f_i at x.

Dual points on central path

As in the scalar case, points on the central path give dual feasible points for the problem (11.38). For $i = 1, \ldots, m$, define

$$\lambda_i^\star(t) = \frac{1}{t}\nabla\psi_i(-f_i(x^\star(t))), \tag{11.42}$$

and let $\nu^\star(t) = \nu/t$, where ν is the optimal dual variable in (11.41). We will show that $\lambda_1^\star(t), \ldots, \lambda_m^\star(t)$, together with $\nu^\star(t)$, are dual feasible for the original problem (11.38).

First, $\lambda_i^\star(t) \succ_{K_i^*} 0$, by the monotonicity property (11.40) of generalized logarithms. Second, it follows from (11.41) that the Lagrangian

$$L(x, \lambda^\star(t), \nu^\star(t)) = f_0(x) + \sum_{i=1}^m \lambda_i^\star(t)^T f_i(x) + \nu^\star(t)^T(Ax - b)$$

is minimized over x by $x = x^\star(t)$. The dual function g evaluated at $(\lambda^\star(t), \nu^\star(t))$ is therefore equal to

$$\begin{aligned} g(\lambda^\star(t), \nu^\star(t)) &= f_0(x^\star(t)) + \sum_{i=1}^m \lambda_i^\star(t)^T f_i(x^\star(t)) + \nu^\star(t)^T(Ax^\star(t) - b) \\ &= f_0(x^\star(t)) + (1/t)\sum_{i=1}^m \nabla\psi_i(-f_i(x^\star(t)))^T f_i(x^\star(t)) \\ &= f_0(x^\star(t)) - (1/t)\sum_{i=1}^m \theta_i, \end{aligned}$$

where θ_i is the degree of ψ_i. In the last line, we use the fact that $y^T\nabla\psi_i(y) = \theta_i$ for $y \succ_{K_i} 0$, and therefore

$$\lambda_i^\star(t)^T f_i(x^\star(t)) = -\theta_i/t, \quad i = 1, \ldots, m. \tag{11.43}$$

Thus, if we define

$$\overline{\theta} = \sum_{i=1}^m \theta_i,$$

then the primal feasible point $x^\star(t)$ and the dual feasible point $(\lambda^\star(t), \nu^\star(t))$ have duality gap $\overline{\theta}/t$. This is just like the scalar case, except that $\overline{\theta}$, the sum of the degrees of the generalized logarithms for the cones, appears in place of m, the number of inequalities.

Example 11.8 *Second-order cone programming.* We consider an SOCP with variable $x \in \mathbf{R}^n$:

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m, \end{array} \tag{11.44}$$

where $A_i \in \mathbf{R}^{n_i \times n}$. As we have seen in example 11.6, the function

$$\psi(y) = \log\left(y_{p+1}^2 - \sum_{i=1}^p y_i^2\right)$$

is a generalized logarithm for the second-order cone in $\mathbf{R}^{p+1}$, with degree 2. The corresponding logarithmic barrier function for (11.44) is

$$\phi(x) = -\sum_{i=1}^m \log((c_i^T x + d_i)^2 - \|A_i x + b_i\|_2^2), \tag{11.45}$$

with $\mathbf{dom}\,\phi = \{x \mid \|A_i x + b_i\|_2 < c_i^T x + d_i,\ i = 1, \ldots, m\}$. The optimality condition on the central path is $tf + \nabla\phi(x^\star(t)) = 0$, where

$$\nabla\phi(x) = -2\sum_{i=1}^m \frac{1}{(c_i^T x + d_i)^2 - \|A_i x + b_i\|_2^2}\left((c_i^T x + d_i)c_i - A_i^T(A_i x + b_i)\right).$$

It follows that the point

$$z_i^\star(t) = -\frac{2}{t\alpha_i}(A_i x^\star(t) + b_i), \qquad w_i^\star(t) = \frac{2}{t\alpha_i}(c_i^T x^\star(t) + d_i), \qquad i = 1, \ldots, m,$$

where $\alpha_i = (c_i^T x^\star(t) + d_i)^2 - \|A_i x^\star(t) + b_i\|_2^2$, is strictly feasible in the dual problem

$$\begin{array}{ll} \text{maximize} & -\sum_{i=1}^m (b_i^T z_i + d_i w_i) \\ \text{subject to} & \sum_{i=1}^m (A_i^T z_i + c_i w_i) = f \\ & \|z_i\|_2 \le w_i, \quad i = 1, \ldots, m. \end{array}$$

The duality gap associated with $x^\star(t)$ and $(z^\star(t), w^\star(t))$ is

$$\sum_{i=1}^m \left((A_i x^\star(t) + b_i)^T z_i^\star(t) + (c_i^T x^\star(t) + d_i) w_i^\star(t)\right) = \frac{2m}{t},$$

which agrees with the general formula $\overline{\theta}/t$, since $\theta_i = 2$.

Example 11.9 *Semidefinite programming in inequality form.* We consider the SDP with variable $x \in \mathbf{R}^n$,

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & F(x) = x_1 F_1 + \cdots + x_n F_n + G \preceq 0, \end{array}$$

where $G, F_1, \ldots, F_n \in \mathbf{S}^p$. The dual problem is

$$\begin{array}{ll} \text{maximize} & \mathbf{tr}(GZ) \\ \text{subject to} & \mathbf{tr}(F_i Z) + c_i = 0, \quad i = 1, \ldots, n \\ & Z \succeq 0. \end{array}$$

Using the generalized logarithm $\log\det X$ for the positive semidefinite cone $\mathbf{S}_+^p$, we have the barrier function (for the primal problem)

$$\phi(x) = \log\det(-F(x)^{-1})$$

with $\mathbf{dom}\,\phi = \{x \mid F(x) \prec 0\}$. For strictly feasible x, the gradient of ϕ is equal to

$$\frac{\partial\phi(x)}{\partial x_i} = \mathbf{tr}(-F(x)^{-1}F_i), \quad i = 1, \ldots, n,$$

which gives us the optimality conditions that characterize central points:

$$tc_i + \mathbf{tr}(-F(x^\star(t))^{-1}F_i) = 0, \quad i = 1, \ldots, n.$$

Hence the matrix

$$Z^\star(t) = \frac{1}{t}\left(-F(x^\star(t))\right)^{-1}$$

is strictly dual feasible, and the duality gap associated with $x^\star(t)$ and $Z^\star(t)$ is p/t.

11.6.2 Barrier method

We have seen that the key properties of the central path generalize to problems with generalized inequalities.

- Computing a point on the central path involves minimizing a twice differentiable convex function subject to equality constraints (which can be done using Newton's method).
- With the central point $x^\star(t)$ we can associate a dual feasible point $(\lambda^\star(t), \nu^\star(t))$ with associated duality gap $\overline{\theta}/t$. In particular, $x^\star(t)$ is no more than $\overline{\theta}/t$-suboptimal.

This means we can apply the barrier method, exactly as described in §11.3, to the problem (11.38). The number of outer iterations, or centering steps, required to compute a central point with duality gap ϵ starting at $x^\star(t^{(0)})$ is equal to

$$\left\lceil \frac{\log(\overline{\theta}/(t^{(0)}\epsilon))}{\log \mu} \right\rceil,$$

plus one initial centering step. The only difference between this result and the associated one for the scalar case is that $\overline{\theta}$ takes the place of m.

Phase I and feasibility problems

The phase I methods described in §11.4 are readily extended to problems with generalized inequalities. Let $e_i \succ_{K_i} 0$ be some given, K_i-positive vectors, for $i = 1, \ldots, m$. To determine feasibility of the equalities and generalized inequalities

$$f_1(x) \preceq_{K_1} 0, \quad \ldots, \quad f_L(x) \preceq_{K_m} 0, \qquad Ax = b,$$

we solve the problem

$$\begin{array}{ll} \text{minimize} & s \\ \text{subject to} & f_i(x) \preceq_{K_i} se_i, \quad i = 1, \ldots, m \\ & Ax = b, \end{array}$$

with variables x and $s \in \mathbf{R}$. The optimal value $\bar{p}^\star$ determines the feasibility of the equalities and generalized inequalities, exactly as in the case of ordinary inequalities. When $\bar{p}^\star$ is positive, any dual feasible point with positive objective gives an alternative that proves the set of equalities and generalized inequalities is infeasible (see page 270).

11.6.3 Examples

A small SOCP

We solve an SOCP

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \leq c_i^T x + d_i, \quad i = 1, \ldots, m, \end{array}$$

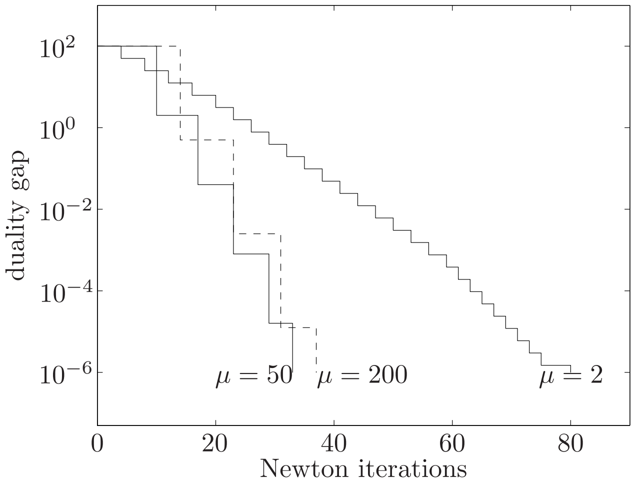

Figure 11.15 Progress of barrier method for an SOCP, showing duality gap versus cumulative number of Newton steps.

with $x \in \mathbf{R}^{50}$, $m = 50$, and $A_i \in \mathbf{R}^{5\times 50}$. The problem instance was randomly generated, in such a way that the problem is strictly primal and dual feasible, and has optimal value $p^\star = 1$. We start with a point $x^{(0)}$ on the central path, with a duality gap of 100.

The barrier method is used to solve the problem, using the barrier function

$$\phi(x) = -\sum_{i=1}^{m} \log\left((c_i^T x + d_i)^2 - \|A_i x + b_i\|_2^2\right).$$

The centering problems are solved using Newton's method, with the same algorithm parameters as in the examples of §11.3.2: backtracking parameters $\alpha = 0.01$, $\beta = 0.5$, and a stopping criterion $\lambda(x)^2/2 \leq 10^{-5}$.

Figure 11.15 shows the duality gap versus cumulative number of Newton steps. The plot is very similar to those for linear and geometric programming, shown in figures 11.4 and 11.6, respectively. We see an approximately constant number of Newton steps required per centering step, and therefore approximately linear convergence of the duality gap. For this example, too, the choice of μ has little effect on the total number of Newton steps, provided μ is at least 10 or so. As in the examples for linear and geometric programming, a reasonable choice of μ is in the range 10 – 100, which results in a total number of Newton steps around 30 (see figure 11.16).

A small SDP

Our next example is an SDP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \sum_{i=1}^n x_i F_i + G \preceq 0 \end{array} \qquad (11.46)$$

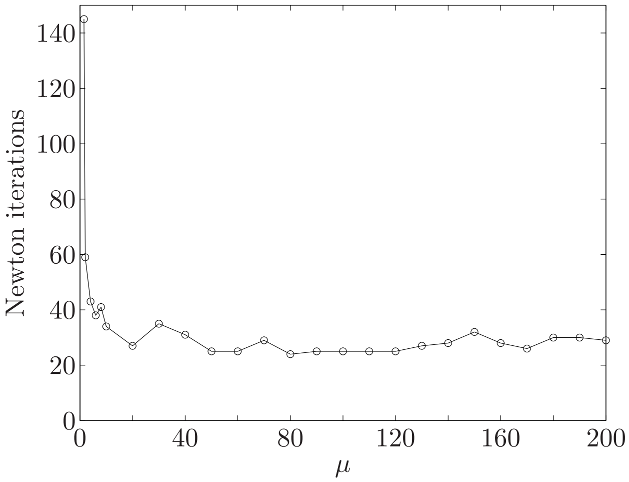

Figure 11.16 Trade-off in the choice of the parameter μ, for a small SOCP. The vertical axis shows the total number of Newton steps required to reduce the duality gap from 100 to 10^{-3}, and the horizontal axis shows μ.

with variable $x \in \mathbf{R}^{100}$, and $F_i \in \mathbf{S}^{100}$, $G \in \mathbf{S}^{100}$. The problem instance was generated randomly, in such a way that the problem is strictly primal and dual feasible, with $p^\star = 1$. The initial point is on the central path, with a duality gap of 100.

We apply the barrier method with logarithmic barrier function

$$\phi(x) = -\log\det\left(-\sum_{i=1}^{n} x_i F_i - G\right).$$

The progress of the barrier method for three values of μ is shown in figure 11.17. Note the similarity with the plots for linear, geometric, and second-order cone programming, shown in figures 11.4, 11.6, and 11.15. As in the other examples, the parameter μ has only a small effect on the efficiency, provided it is not too small. The number of Newton steps required to reduce the duality gap by a factor 10^5, versus μ, is shown in figure 11.18.

A family of SDPs

In this section we examine the performance of the barrier method as a function of the problem dimensions. We consider a family of SDPs of the form

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T x \\ \text{subject to} & A + \mathbf{diag}(x) \succeq 0, \end{array} \qquad (11.47)$$

with variable $x \in \mathbf{R}^n$, and parameter $A \in \mathbf{S}^n$. The matrices A are generated as follows. For $i \geq j$, the coefficients A_{ij} are generated from independent $\mathcal{N}(0,1)$ distributions. For $i < j$, we set $A_{ij} = A_{ji}$, so $A \in \mathbf{S}^n$. We then scale A so that its (spectral) norm is one.

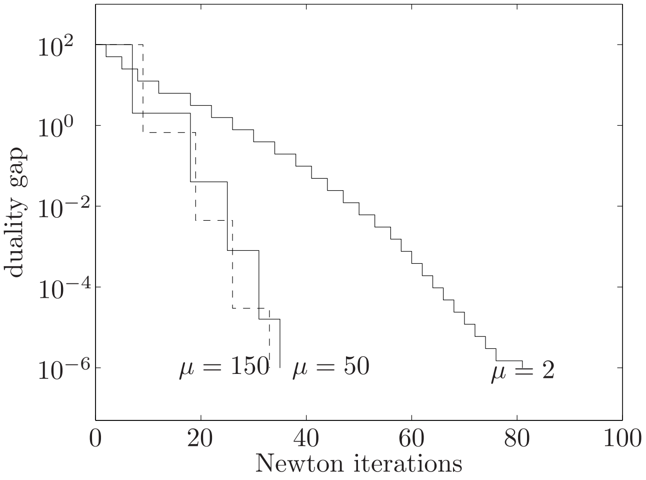

Figure 11.17 Progress of barrier method for a small SDP, showing duality gap versus cumulative number of Newton steps. Three plots are shown, corresponding to three values of the parameter μ: 2, 50, and 150.

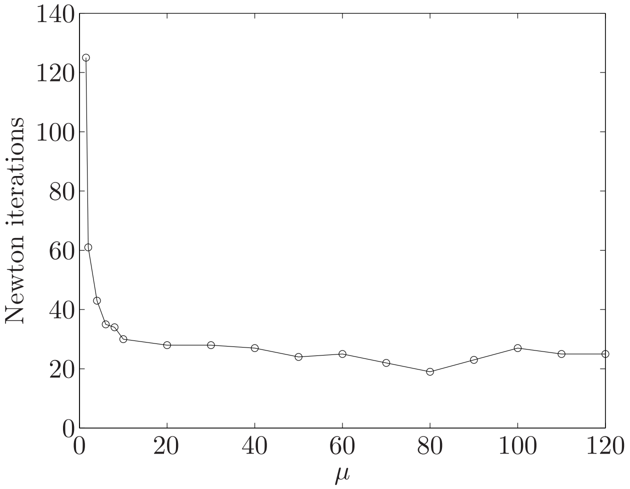

Figure 11.18 Trade-off in the choice of the parameter μ, for a small SDP. The vertical axis shows the total number of Newton steps required to reduce the duality gap from 100 to 10^{-3}, and the horizontal axis shows μ.

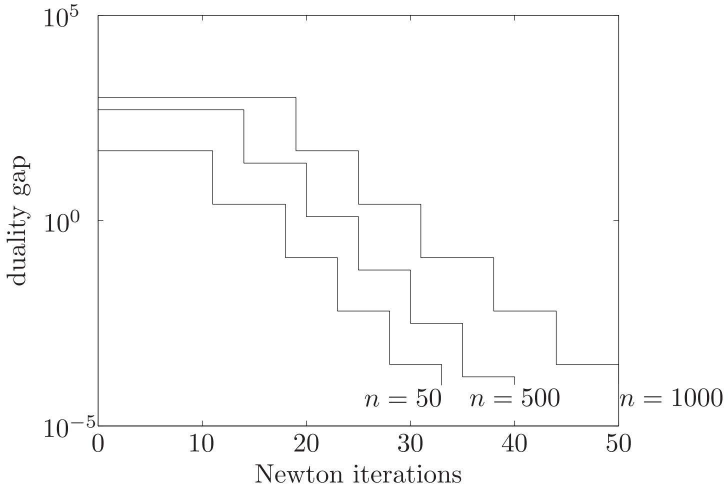

Figure 11.19 Progress of barrier method for three randomly generated SDPs of the form (11.47), with different dimensions. The plot shows duality gap versus cumulative number of Newton steps. The number of variables in each problem is n.

The algorithm parameters are $\mu = 20$, and the same parameters for the centering steps as in the examples above: backtracking parameters $\alpha = 0.01$, $\beta = 0.5$, and stopping criterion $\lambda(x)^2/2 \leq 10^{-5}$. The initial point is on the central path with $t^{(0)} = 1$ (*i.e.*, gap n). The algorithm is terminated when the initial duality gap is reduced by a factor 8000, *i.e.*, after completing three outer iterations.

Figure 11.19 shows the duality gap versus iteration number for three problem instances, with dimensions $n = 50$, $n = 500$, and $n = 1000$. The plots look very much like the others, and very much like the ones for LPs.

To examine the effect of problem size on the number of Newton steps required, we generate 100 problem instances for each of 20 values of n, ranging from $n = 10$ to $n = 1000$. We solve each of these 2000 problems using the barrier method, noting the number of Newton steps required. The results are summarized in figure 11.20, which shows the mean and standard deviation in the number of Newton steps, for each value of n. The plot looks very much like the one for LPs, shown in figure 11.8. In particular, the number of Newton steps required grows very slowly, from around 20 to 26 iterations, as the problem dimensions increase by a factor of 100.

11.6.4 Complexity analysis via self-concordance

In this section we extend the complexity analysis of the barrier method for problems with ordinary inequalities (given in §11.5), to problems with generalized inequalities. We have already seen that the number of outer iterations is given by

$$\left\lceil \frac{\log(\overline{\theta}/t^{(0)}\epsilon)}{\log \mu} \right\rceil,$$

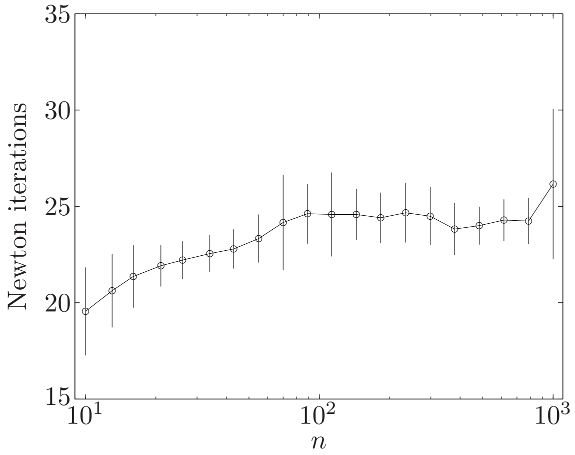

Figure 11.20 Average number of Newton steps required to solve 100 randomly generated SDPs (11.47) for each of 20 values of n, the problem size. Error bars show standard deviation, around the average value, for each value of n. The growth in the average number of Newton steps required, as the problem dimensions range over a 100 : 1 ratio, is very small.

plus one initial centering step. It remains to bound the number of Newton steps required in each centering step, which we will do using the complexity theory of Newton's method for self-concordant functions. For simplicity, we will exclude the cost of the initial centering.

We make the same assumptions as in §11.5: The function $tf_0 + \phi$ is closed and self-concordant for all $t \geq t^{(0)}$, and the sublevel sets of (11.38) are bounded.

Example 11.10 *Second-order cone programming.* The function

$$-\psi(x) = -\log\left(x_{p+1}^2 - \sum_{i=1}^{p} x_i^2\right),$$

is self-concordant (see example 9.8), so the logarithmic barrier function (11.45) satisfies the closedness and self-concordance assumption for the SOCP (11.44).

Example 11.11 *Semidefinite programming.* The self-concordance assumption holds for general semidefinite programs, using $\log\det X$ as generalized logarithm for the positive semidefinite cone. For example, for the standard form SDP

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(CX) \\ \text{subject to} & \mathbf{tr}(A_iX) = b_i, \quad i = 1, \dots, p \\ & X \succeq 0, \end{array}$$

with variable $X \in \mathbf{S}^n$, the function $t^{(0)}\,\mathbf{tr}(CX) - \log\det X$ is self-concordant (and closed), for any $t^{(0)} \geq 0$.

We will see that, exactly as in the scalar case, we have

$$\mu t f_0(x^\star(t)) + \phi(x^\star(t)) - \mu t f_0(x^\star(\mu t)) - \phi(x^\star(\mu t)) \leq \overline{\theta}(\mu - 1 - \log \mu). \qquad (11.48)$$

Therefore when the self-concordance and bounded sublevel set conditions hold, the number of Newton steps per centering step is no more than

$$\frac{\overline{\theta}(\mu - 1 - \log \mu)}{\gamma} + c,$$

exactly as in the barrier method for problems with ordinary inequalities. Once we establish the basic bound (11.48), the complexity analysis for problems with generalized inequalities is identical to the analysis for problems with ordinary inequalities, with one exception: $\overline{\theta}$ is the sum of the degrees of the cones, instead of the number of inequalities.

Generalized logarithm for dual cone

We will use conjugates to prove the bound (11.48). Let ψ be a generalized logarithm for the proper cone K, with degree θ. The conjugate of the (convex) function $-\psi$ is

$$(-\psi)^*(v) = \sup_u \left(v^T u + \psi(u)\right).$$

This function is convex, and has domain $-K^* = \{v \mid v \prec_{K^*} 0\}$. Define $\overline{\psi}$ by

$$\overline{\psi}(v) = -(-\psi)^*(-v) = \inf_u \left(v^T u - \psi(u)\right), \qquad \mathbf{dom}\,\overline{\psi} = \mathbf{int}\,K^*. \qquad (11.49)$$

The function $\overline{\psi}$ is concave, and in fact is a generalized logarithm for the dual cone K^*, with the same parameter θ (see exercise 11.17). We call $\overline{\psi}$ the *dual logarithm* associated with the generalized logarithm ψ.

From (11.49) we obtain the inequality

$$\overline{\psi}(v) + \psi(u) \leq u^T v, \qquad (11.50)$$

which holds for any $u \succ_K 0$, $v \succ_{K^*} 0$, with equality holding if and only $\nabla\psi(u) = v$ (or equivalently, $\nabla\overline{\psi}(v) = u$). (This inequality is just a variation on Young's inequality, for concave functions.)

Example 11.12 *Second-order cone.* The second-order cone has generalized logarithm $\psi(x) = \log(x_{p+1}^2 - \sum_{i=1}^p x_i^2)$, with $\mathbf{dom}\,\psi = \{x \in \mathbf{R}^{p+1} \mid x_{p+1} > (\sum_{i=1}^p x_i^2)^{1/2}\}$. The associated dual logarithm is

$$\overline{\psi}(y) = \log\left(y_{p+1}^2 - \sum_{i=1}^p y_i^2\right) + 2 - \log 4,$$

with $\mathbf{dom}\,\psi = \{y \in \mathbf{R}^{p+1} \mid y_{p+1} > (\sum_{i=1}^p y_i^2)^{1/2}\}$ (see exercise 3.36). Except for a constant, it is the same as the original generalized logarithm for the second-order cone.

Example 11.13 *Positive semidefinite cone.* The dual logarithm associated with $\psi(X) = \log\det X$, with $\textbf{dom}\,\psi = \mathbf{S}^p_{++}$, is

$$\overline{\psi}(Y) = \log\det Y + p,$$

with domain $\textbf{dom}\,\psi^* = \mathbf{S}^p_{++}$ (see example 3.23). Again, it is the same generalized logarithm, except for a constant.

Derivation of the basic bound

To simplify notation, we denote $x^\star(t)$ as x, $x^\star(\mu t)$ as x^+, $\lambda_i^\star(t)$ as λ_i, and $\nu^\star(t)$ as ν. From $t\lambda_i = \nabla\psi_i(-f_i(x))$ (in (11.42)) and property (11.43), we conclude that

$$\psi_i(-f_i(x)) + \overline{\psi}_i(t\lambda_i) = -t\lambda_i^T f_i(x) = \theta_i, \tag{11.51}$$

i.e., the inequality (11.50) holds with equality for the pair $u = -f_i(x)$ and $v = t\lambda_i$. The same inequality for the pair $u = -f_i(x^+)$, $v = \mu t\lambda_i$ gives

$$\psi_i(-f_i(x^+)) + \overline{\psi}_i(\mu t\lambda_i) \le -\mu t\lambda_i^T f_i(x^+),$$

which becomes, using logarithmic homogeneity of $\overline{\psi}_i$,

$$\psi_i(-f_i(x^+)) + \overline{\psi}_i(t\lambda_i) + \theta_i \log\mu \le -\mu t\lambda_i^T f_i(x^+).$$

Subtracting the equality (11.51) from this inequality, we get

$$-\psi_i(-f_i(x)) + \psi_i(-f_i(x^+)) + \theta_i\log\mu \le -\theta_i - \mu t\lambda_i^T f_i(x^+),$$

and summing over i yields

$$\phi(x) - \phi(x^+) + \overline{\theta}\log\mu \le -\overline{\theta} - \mu t\sum_{i=1}^m \lambda_i^T f_i(x^+). \tag{11.52}$$

We also have, from the definition of the dual function,

$$\begin{aligned} f_0(x) - \overline{\theta}/t &= g(\lambda,\nu) \\ &\le f_0(x^+) + \sum_{i=1}^m \lambda_i^T f_i(x^+) + \nu^T(Ax^+ - b) \\ &= f_0(x^+) + \sum_{i=1}^m \lambda_i^T f_i(x^+). \end{aligned}$$

Multiplying this inequality by μt and adding to the inequality (11.52), we get

$$\phi(x) - \phi(x^+) + \overline{\theta}\log\mu + \mu t f_0(x) - \mu\overline{\theta} \le \mu t f_0(x^+) - \overline{\theta},$$

which when re-arranged gives

$$\mu t f_0(x) + \phi(x) - \mu t f_0(x^+) - \phi(x^+) \le \overline{\theta}(\mu - 1 - \log\mu),$$

the desired inequality (11.48).

11.7 Primal-dual interior-point methods

In this section we describe a basic primal-dual interior-point method. Primal-dual interior-point methods are very similar to the barrier method, with some differences.

- There is only one loop or iteration, *i.e.*, there is no distinction between inner and outer iterations as in the barrier method. At each iteration, both the primal and dual variables are updated.
- The search directions in a primal-dual interior-point method are obtained from Newton's method, applied to modified KKT equations (*i.e.*, the optimality conditions for the logarithmic barrier centering problem). The primal-dual search directions are similar to, but not quite the same as, the search directions that arise in the barrier method.
- In a primal-dual interior-point method, the primal and dual iterates are *not* necessarily feasible.

Primal-dual interior-point methods are often more efficient than the barrier method, especially when high accuracy is required, since they can exhibit better than linear convergence. For several basic problem classes, such as linear, quadratic, second-order cone, geometric, and semidefinite programming, customized primal-dual methods outperform the barrier method. For general nonlinear convex optimization problems, primal-dual interior-point methods are still a topic of active research, but show great promise. Another advantage of primal-dual algorithms over the barrier method is that they can work when the problem is feasible, but not strictly feasible (although we will not pursue this).

In this section we present a basic primal-dual method for (11.1), without convergence analysis. We refer the reader to the references for a more thorough treatment of primal-dual methods and their convergence analysis.

11.7.1 Primal-dual search direction

As in the barrier method, we start with the modified KKT conditions (11.15), expressed as $r_t(x,\lambda,\nu)=0$, where we define

$$r_t(x,\lambda,\nu) = \left[\begin{array}{c} \nabla f_0(x) + Df(x)^T\lambda + A^T\nu \\ -\,\mathbf{diag}(\lambda)f(x) - (1/t)\mathbf{1} \\ Ax-b \end{array}\right], \tag{11.53}$$

and $t>0$. Here $f:\mathbf{R}^n \to \mathbf{R}^m$ and its derivative matrix Df are given by

$$f(x) = \left[\begin{array}{c} f_1(x) \\ \vdots \\ f_m(x) \end{array}\right], \qquad Df(x) = \left[\begin{array}{c} \nabla f_1(x)^T \\ \vdots \\ \nabla f_m(x)^T \end{array}\right].$$

If x, λ, ν satisfy $r_t(x,\lambda,\nu)=0$ (and $f_i(x)<0$), then $x=x^\star(t)$, $\lambda=\lambda^\star(t)$, and $\nu=\nu^\star(t)$. In particular, x is primal feasible, and λ, ν are dual feasible, with duality gap m/t. The first block component of r_t,

$$r_{\rm dual} = \nabla f_0(x) + Df(x)^T\lambda + A^T\nu,$$

is called the *dual residual*, and the last block component, $r_{\rm pri} = Ax - b$, is called the *primal residual*. The middle block,

$$r_{\rm cent} = -\,\mathbf{diag}(\lambda)f(x) - (1/t)\mathbf{1},$$

is the *centrality residual*, *i.e.*, the residual for the modified complementarity condition.

Now consider the Newton step for solving the nonlinear equations $r_t(x,\lambda,\nu) = 0$, for fixed t (without first eliminating λ, as in §11.3.4), at a point (x,λ,ν) that satisifes $f(x) \prec 0$, $\lambda \succ 0$. We will denote the current point and Newton step as

$$y = (x,\lambda,\nu), \qquad \Delta y = (\Delta x, \Delta\lambda, \Delta\nu),$$

respectively. The Newton step is characterized by the linear equations

$$r_t(y + \Delta y) \approx r_t(y) + Dr_t(y)\Delta y = 0,$$

i.e., $\Delta y = -Dr_t(y)^{-1}r_t(y)$. In terms of x, λ, and ν, we have

$$\left[\begin{array}{ccc} \nabla^2 f_0(x) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(x) & Df(x)^T & A^T \\ -\,\mathbf{diag}(\lambda)Df(x) & -\,\mathbf{diag}(f(x)) & 0 \\ A & 0 & 0 \end{array}\right] \left[\begin{array}{c} \Delta x \\ \Delta\lambda \\ \Delta\nu \end{array}\right] = -\left[\begin{array}{c} r_{\rm dual} \\ r_{\rm cent} \\ r_{\rm pri} \end{array}\right]. \tag{11.54}$$

The *primal-dual search direction* $\Delta y_{\rm pd} = (\Delta x_{\rm pd}, \Delta\lambda_{\rm pd}, \Delta\nu_{\rm pd})$ is defined as the solution of (11.54).

The primal and dual search directions are coupled, both through the coefficient matrix and the residuals. For example, the primal search direction $\Delta x_{\rm pd}$ depends on the current value of the dual variables λ and ν, as well as x. We note also that if x satisfies $Ax = b$, *i.e.*, the primal feasibility residual $r_{\rm pri}$ is zero, then we have $A\Delta x_{\rm pd} = 0$, so $\Delta x_{\rm pd}$ defines a (primal) feasible direction: for any s, $x + s\Delta x_{\rm pd}$ will satisfy $A(x + s\Delta x_{\rm pd}) = b$.

Comparison with barrier method search directions

The primal-dual search directions are closely related to the search directions used in the barrier method, but not quite the same. We start with the linear equations (11.54) that define the primal-dual search directions. We eliminate the variable $\Delta\lambda_{\rm pd}$, using

$$\Delta\lambda_{\rm pd} = -\,\mathbf{diag}(f(x))^{-1}\,\mathbf{diag}(\lambda)Df(x)\Delta x_{\rm pd} + \mathbf{diag}(f(x))^{-1}r_{\rm cent},$$

which comes from the second block of equations. Substituting this into the first block of equations gives

$$\begin{aligned} \left[\begin{array}{cc} H_{\rm pd} & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} \Delta x_{\rm pd} \\ \Delta\nu_{\rm pd} \end{array}\right] &= -\left[\begin{array}{c} r_{\rm dual} + Df(x)^T\,\mathbf{diag}(f(x))^{-1}r_{\rm cent} \\ r_{\rm pri} \end{array}\right] \\ &= -\left[\begin{array}{c} \nabla f_0(x) + (1/t)\sum_{i=1}^m \frac{1}{-f_i(x)}\nabla f_i(x) + A^T\nu \\ r_{\rm pri} \end{array}\right], \end{aligned} \tag{11.55}$$

where

$$H_{\mathrm{pd}} = \nabla^2 f_0(x) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(x) + \sum_{i=1}^m \frac{\lambda_i}{-f_i(x)} \nabla f_i(x) \nabla f_i(x)^T. \qquad (11.56)$$

We can compare (11.55) to the equation (11.14), which defines the Newton step for the centering problem in the barrier method with parameter t. This equation can be written as

$$\begin{aligned} \left[\begin{array}{cc} H_{\mathrm{bar}} & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} \Delta x_{\mathrm{bar}} \\ \nu_{\mathrm{bar}} \end{array}\right] & \\ = \; & -\left[\begin{array}{c} t\nabla f_0(x) + \nabla \phi(x) \\ r_{\mathrm{pri}} \end{array}\right] \\ = \; & -\left[\begin{array}{c} t\nabla f_0(x) + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x) \\ r_{\mathrm{pri}} \end{array}\right], \end{aligned} \qquad (11.57)$$

where

$$H_{\mathrm{bar}} = t\nabla^2 f_0(x) + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla^2 f_i(x) + \sum_{i=1}^m \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T. \qquad (11.58)$$

(Here we give the general expression for the infeasible Newton step; if the current x is feasible, *i.e.*, $r_{\mathrm{pri}} = 0$, then Δx_{bar} coincides with the feasible Newton step Δx_{nt} defined in (11.14).)

Our first observation is that the two systems of equations (11.55) and (11.57) are very similar. The coefficient matrices in (11.55) and (11.57) have the same structure; indeed, the matrices H_{pd} and H_{bar} are both positive linear combinations of the matrices

$$\nabla^2 f_0(x), \qquad \nabla^2 f_1(x), \ldots, \nabla^2 f_m(x), \qquad \nabla f_1(x) \nabla f_1(x)^T, \ldots, \nabla f_m(x) \nabla f_m(x)^T.$$

This means that the same method can be used to compute the primal-dual search directions and the barrier method Newton step.

We can say more about the relation between the primal-dual equations (11.55) and the barrier method equations (11.57). Suppose we divide the first block of equation (11.57) by t, and define the variable $\Delta \nu_{\mathrm{bar}} = (1/t)\nu_{\mathrm{bar}} - \nu$ (where ν is arbitrary). Then we obtain

$$\left[\begin{array}{cc} (1/t) H_{\mathrm{bar}} & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} \Delta x_{\mathrm{bar}} \\ \Delta \nu_{\mathrm{bar}} \end{array}\right] = -\left[\begin{array}{c} \nabla f_0(x) + (1/t) \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla f_i(x) + A^T \nu \\ r_{\mathrm{pri}} \end{array}\right].$$

In this form, the righthand side is identical to the righthand side of the primal-dual equations (evaluated at the same x, λ, and ν). The coefficient matrices differ only in the $1,1$ block:

$$\begin{aligned} H_{\mathrm{pd}} &= \nabla^2 f_0(x) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(x) + \sum_{i=1}^m \frac{\lambda_i}{-f_i(x)} \nabla f_i(x) \nabla f_i(x)^T, \\ (1/t) H_{\mathrm{bar}} &= \nabla^2 f_0(x) + \sum_{i=1}^m \frac{1}{-t f_i(x)} \nabla^2 f_i(x) + \sum_{i=1}^m \frac{1}{t f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T. \end{aligned}$$

When x and λ satisfy $-f_i(x)\lambda_i = 1/t$, the coefficient matrices, and therefore also the search directions, coincide.

11.7.2 The surrogate duality gap

In the primal-dual interior-point method the iterates $x^{(k)}$, $\lambda^{(k)}$, and $\nu^{(k)}$ are not necessarily feasible, except in the limit as the algorithm converges. This means that we cannot easily evaluate a duality gap $\eta^{(k)}$ associated with step k of the algorithm, as we do in (the outer steps of) the barrier method. Instead we define the *surrogate duality gap*, for any x that satisfies $f(x) \prec 0$ and $\lambda \succeq 0$, as

$$\hat{\eta}(x, \lambda) = -f(x)^T \lambda. \qquad (11.59)$$

The surrogate gap $\hat{\eta}$ would be the duality gap, if x were primal feasible and λ, ν were dual feasible, *i.e.*, if $r_{\text{pri}} = 0$ and $r_{\text{dual}} = 0$. Note that the value of the parameter t that corresponds to the surrogate duality gap $\hat{\eta}$ is $m/\hat{\eta}$.

11.7.3 Primal-dual interior-point method

We can now describe the basic primal-dual interior-point algorithm.

Algorithm 11.2 *Primal-dual interior-point method.*

```
given $x$ that satisfies $f_1(x) < 0, \ldots, f_m(x) < 0$, $\lambda \succ 0$, $\mu > 1$, $\epsilon_{\text{feas}} > 0$, $\epsilon > 0$.
repeat
    1. Determine t. Set $t := \mu m / \hat{\eta}$.
    2. Compute primal-dual search direction $\Delta y_{\text{pd}}$.
    3. Line search and update.
        Determine step length $s > 0$ and set $y := y + s\Delta y_{\text{pd}}$.
until $\|r_{\text{pri}}\|_2 \le \epsilon_{\text{feas}}$, $\|r_{\text{dual}}\|_2 \le \epsilon_{\text{feas}}$, and $\hat{\eta} \le \epsilon$.
```

In step 1, the parameter t is set to a factor μ times $m/\hat{\eta}$, which is the value of t associated with the current surrogate duality gap $\hat{\eta}$. If x, λ, and ν were central, with parameter t (and therefore with duality gap m/t), then in step 1 we would increase t by the factor μ, which is exactly the update used in the barrier method. Values of the parameter μ on the order of 10 appear to work well.

The primal-dual interior-point algorithm terminates when x is primal feasible and λ, ν are dual feasible (within the tolerance ϵ_{feas}) and the surrogate gap is smaller than the tolerance ϵ. Since the primal-dual interior-point method often has faster than linear convergence, it is common to choose ϵ_{feas} and ϵ small.

Line search

The line search in the primal-dual interior point method is a standard backtracking line search, based on the norm of the residual, and modified to ensure that $\lambda \succ 0$ and $f(x) \prec 0$. We denote the current iterate as x, λ, and ν, and the next iterate as x^+, λ^+, and ν^+, *i.e.*,

$$x^+ = x + s\Delta x_{\text{pd}}, \qquad \lambda^+ = \lambda + s\Delta \lambda_{\text{pd}}, \qquad \nu^+ = \nu + s\Delta \nu_{\text{pd}}.$$

The residual, evaluated at y^+, will be denoted r^+.

We first compute the largest positive step length, not exceeding one, that gives $\lambda^+ \succeq 0$, *i.e.*,

$$\begin{array}{rcl} s^{\max} & = & \sup\{s \in [0,1] \mid \lambda + s\Delta\lambda \succeq 0\} \\ & = & \min\left\{1,\ \min\{-\lambda_i/\Delta\lambda_i \mid \Delta\lambda_i < 0\}\right\}. \end{array}$$

We start the backtracking with $s = 0.99 s^{\max}$, and multiply s by $\beta \in (0,1)$ until we have $f(x^+) \prec 0$. We continue multiplying s by β until we have

$$\|r_t(x^+, \lambda^+, \nu^+)\|_2 \leq (1 - \alpha s)\|r_t(x, \lambda, \nu)\|_2.$$

Common choices for the backtracking parameters α and β are the same as those for Newton's method: α is typically chosen in the range 0.01 to 0.1, and β is typically chosen in the range 0.3 to 0.8.

One iteration of the primal-dual interior-point algorithm is the same as one step of the infeasible Newton method, applied to solving $r_t(x, \lambda, \nu) = 0$, but modified to ensure $\lambda \succ 0$ and $f(x) \prec 0$ (or, equivalently, with $\mathbf{dom}\, r_t$ restricted to $\lambda \succ 0$ and $f(x) \prec 0$). The same arguments used in the proof of convergence of the infeasible start Newton method show that the line search for the primal-dual method always terminates in a finite number of steps.

11.7.4 Examples

We illustrate the performance of the primal-dual interior-point method for the same problems considered in §11.3.2. The only difference is that instead of starting with a point on the central path, as in §11.3.2, we start the primal-dual interior-point method at a randomly generated $x^{(0)}$, that satisfies $f(x) \prec 0$, and take $\lambda_i^{(0)} = -1/f_i(x^{(0)})$, so the initial value of the surrogate gap is $\hat{\eta} = 100$. The parameter values we use for the primal-dual interior-point method are

$$\mu = 10, \qquad \beta = 0.5, \qquad \epsilon = 10^{-8}, \qquad \alpha = 0.01.$$

Small LP and GP

We first consider the small LP used in §11.3.2, with $m = 100$ inequalities and $n = 50$ variables. Figure 11.21 shows the progress of the primal-dual interior-point method. Two plots are shown: the surrogate gap $\hat{\eta}$, and the norm of the primal and dual residuals,

$$r_{\mathrm{feas}} = \left(\|r_{\mathrm{pri}}\|_2^2 + \|r_{\mathrm{dual}}\|_2^2\right)^{1/2},$$

versus iteration number. (The initial point is primal feasible, so the plot shows the norm of the dual feasibility residual.) The plots show that the residual converges to zero rapidly, and becomes zero to numerical precision in 24 iterations. The surrogate gap also converges rapidly. Compared to the barrier method, the primal-dual interior-point method is faster, especially when high accuracy is required.

Figure 11.22 shows the progress of the primal-dual interior-point method on the GP considered in §11.3.2. The convergence is similar to the LP example.

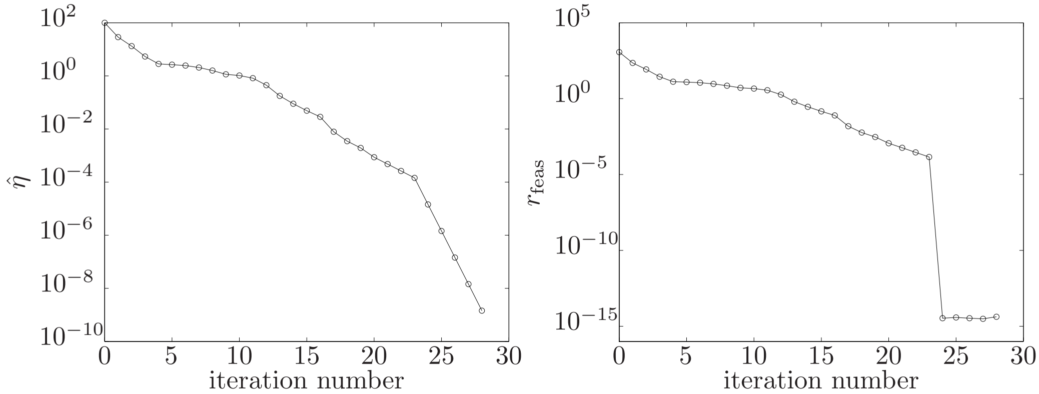

Figure 11.21 Progress of the primal-dual interior-point method for an LP, showing surrogate duality gap $\hat{\eta}$ and the norm of the primal and dual residuals, versus iteration number. The residual converges rapidly to zero within 24 iterations; the surrogate gap also converges to a very small number in about 28 iterations. The primal-dual interior-point method converges faster than the barrier method, especially if high accuracy is required.

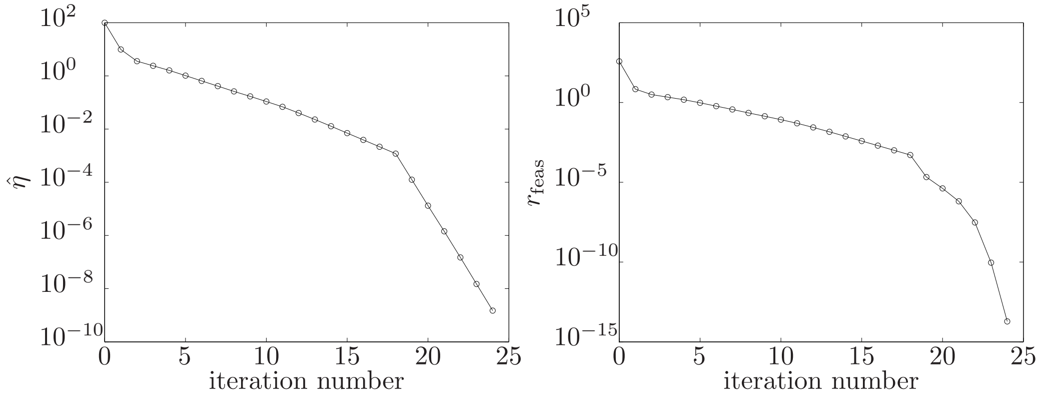

Figure 11.22 Progress of primal-dual interior-point method for a GP, showing surrogate duality gap $\hat{\eta}$ and the norm of the primal and dual residuals versus iteration number.

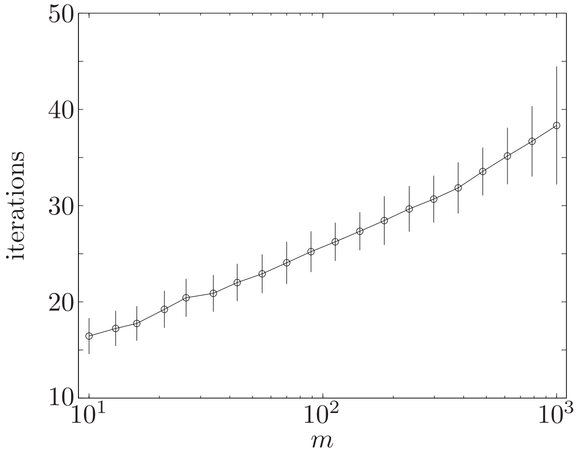

Figure 11.23 Number of iterations required to solve randomly generated standard LPs of different dimensions, with $n = 2m$. Error bars show standard deviation, around the average value, for 100 instances of each dimension. The growth in the number of iterations required, as the problem dimensions range over a 100 : 1 ratio, is approximately logarithmic.

A family of LPs

Here we examine the performance of the primal-dual method as a function of the problem dimensions, for the same family of standard form LPs considered in §11.3.2. We use the primal-dual interior-point method to solve the same 2000 instances, which consist of 100 instances for each value of m. The primal-dual algorithm is started at $x^{(0)} = \mathbf{1}$, $\lambda^{(0)} = \mathbf{1}$, $\nu^{(0)} = 0$, and terminated using tolerance $\epsilon = 10^{-8}$. Figure 11.23 shows the average, and standard deviation, of the number of iterations required versus m. The number of iterations ranges from 15 to 35, and grows approximately as the logarithm of m. Comparing with the results for the barrier method shown in figure 11.8, we see that the number of iterations in the primal-dual method is only slightly higher, despite the fact that we start at infeasible starting points, and solve the problem to a much higher accuracy.

11.8 Implementation

The main effort in the barrier method is computing the Newton step for the centering problem, which consists of solving sets of linear equations of the form

$$\begin{bmatrix} H & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x_{\mathrm{nt}} \\ \nu_{\mathrm{nt}} \end{bmatrix} = - \begin{bmatrix} g \\ 0 \end{bmatrix}, \qquad (11.60)$$

where

$$H = t\nabla^2 f_0(x) + \sum_{i=1}^m \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T + \sum_{i=1}^m \frac{1}{-f_i(x)} \nabla^2 f_i(x)$$

$$g = t\nabla f_0(x) + \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla f_i(x).$$

The Newton equations for the primal-dual method have exactly the same structure, so our observations in this section apply to the primal-dual method as well.

The coefficient matrix of (11.60) has KKT structure, so all of the discussion in §9.7 and §10.4 applies here. In particular, the equations can be solved by elimination, and structure such as sparsity or diagonal plus low rank can be exploited. Let us give some generic examples in which the special structure of the KKT equations can be exploited to compute the Newton step more efficiently.

Sparse problems

If the original problem is sparse, which means that the objective and every constraint function each depend on only a modest number of variables, then the gradients and Hessian matrices of the objective and constraint functions are all sparse, as is the coefficient matrix A. Provided m is not too big, the matrix H is then likely to be sparse, so a sparse matrix method can be used to compute the Newton step. The method will likely work well if there are a few relatively dense rows and columns in the KKT matrix, which would occur, for example, if there were a few equality constraints involving a large number of variables.

Separable objective and a few linear inequality constraints

Suppose the objective function is separable, and there are only a relatively small number of linear equality and inequality constraints. Then $\nabla^2 f_0(x)$ is diagonal, and the terms $\nabla^2 f_i(x)$ vanish, so the matrix H is diagonal plus low rank. Since H is easily inverted, we can solve the KKT equations efficiently. The same method can be applied whenever $\nabla^2 f_0(x)$ is easily inverted, *e.g.*, banded, sparse, or block diagonal.

11.8.1 Standard form linear programming

We first discuss the implementation of the barrier method for the standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \succeq 0, \end{array}$$

with $A \in \mathbf{R}^{m\times n}$. The Newton equations for the centering problem

$$\begin{array}{ll} \text{minimize} & tc^T x - \sum_{i=1}^n \log x_i \\ \text{subject to} & Ax = b \end{array}$$

are given by

$$\left[\begin{array}{cc} \mathbf{diag}(x)^{-2} & A^T \\ A & 0 \end{array}\right] \left[\begin{array}{c} \Delta x_{\rm nt} \\ \nu_{\rm nt} \end{array}\right] = \left[\begin{array}{c} -tc + \mathbf{diag}(x)^{-1}\mathbf{1} \\ 0 \end{array}\right].$$

These equations are usually solved by block elimination of $\Delta x_{\rm nt}$. From the first equation,

$$\begin{array}{rcl}\Delta x_{\rm nt} & = & \mathbf{diag}(x)^2(-tc+\mathbf{diag}(x)^{-1}\mathbf{1}-A^T\nu_{\rm nt}) \\ & = & -t\,\mathbf{diag}(x)^2c+x-\mathbf{diag}(x)^2A^T\nu_{\rm nt}.\end{array}$$

Substituting in the second equation yields

$$A\,\mathbf{diag}(x)^2A^T\nu_{\rm nt}=-tA\,\mathbf{diag}(x)^2c+b.$$

The coefficient matrix is positive definite since by assumption $\mathbf{rank}\,A=m$. Moreover if A is sparse, then usually $A\,\mathbf{diag}(x)^2A^T$ is sparse, so a sparse Cholesky factorization can be used.

11.8.2 ℓ_1-norm approximation

Consider the ℓ_1-norm approximation problem

$$\text{minimize} \quad \|Ax-b\|_1$$

with $A\in\mathbf{R}^{m\times n}$. We will discuss the implementation assuming m and n are large, and A is structured, *e.g.*, sparse, and compare it with the cost of the corresponding least-squares problem

$$\text{minimize} \quad \|Ax-b\|_2^2 \ .$$

We start by expressing the ℓ_1-norm approximation problem as an LP by introducing auxiliary variables $y\in\mathbf{R}^m$:

$$\begin{array}{ll}\text{minimize} & \mathbf{1}^Ty \\ \text{subject to} & \left[\begin{array}{cc} A & -I \\ -A & -I\end{array}\right]\left[\begin{array}{c} x \\ y\end{array}\right]\preceq\left[\begin{array}{c} b \\ -b\end{array}\right].\end{array}$$

The Newton equation for the centering problem is

$$\left[\begin{array}{cc} A^T & -A^T \\ -I & -I\end{array}\right]\left[\begin{array}{cc} D_1 & 0 \\ 0 & D_2\end{array}\right]\left[\begin{array}{cc} A & -I \\ -A & -I\end{array}\right]\left[\begin{array}{c} \Delta x_{\rm nt} \\ \Delta y_{\rm nt}\end{array}\right]=-\left[\begin{array}{c} A^Tg_1 \\ g_2\end{array}\right]$$

where

$$D_1=\mathbf{diag}(b-Ax+y)^{-2}, \qquad D_2=\mathbf{diag}(-b+Ax+y)^{-2}$$

and

$$\begin{array}{rcl} g_1 & = & \mathbf{diag}(b-Ax+y)^{-1}\mathbf{1}-\mathbf{diag}(-b+Ax+y)^{-1}\mathbf{1} \\ g_2 & = & t\mathbf{1}-\mathbf{diag}(b-Ax+y)^{-1}\mathbf{1}-\mathbf{diag}(-b+Ax+y)^{-1}\mathbf{1}.\end{array}$$

If we multiply out the lefthand side, this can be simplified as

$$\left[\begin{array}{cc} A^T(D_1+D_2)A & -A^T(D_1-D_2) \\ -(D_1-D_2)A & D_1+D_2\end{array}\right]\left[\begin{array}{c} \Delta x_{\rm nt} \\ \Delta y_{\rm nt}\end{array}\right]=-\left[\begin{array}{c} A^Tg_1 \\ g_2\end{array}\right].$$

Applying block elimination to $\Delta y_{\rm nt}$, we can reduce this to

$$A^T D A \Delta x_{\rm nt} = -A^T g \tag{11.61}$$

where

$$D = 4D_1D_2(D_1+D_2)^{-1} = 2(\mathbf{diag}(y)^2 + \mathbf{diag}(b-Ax)^2)^{-1}$$

and

$$g = g_1 + (D_1 - D_2)(D_1+D_2)^{-1}g_2.$$

After solving for $\Delta x_{\rm nt}$, we obtain $\Delta y_{\rm nt}$ from

$$\Delta y_{\rm nt} = (D_1+D_2)^{-1}(-g_2 + (D_1-D_2)A\Delta x_{\rm nt}).$$

It is interesting to note that (11.61) are the normal equations of a weighted least-squares problem

$$\mbox{minimize} \quad \|D^{1/2}(A\Delta x + D^{-1}g)\|_2.$$

In other words, the cost of solving the ℓ_1-norm approximation problem is the cost of solving a relatively small number of weighted least-squares problems with the same matrix A, and weights that change at each iteration. If A has structure that allows us to solve the least-squares problem fast (for example, by exploiting sparsity), then we can solve (11.61) fast.

11.8.3 Semidefinite programming in inequality form

We consider the SDP

$$\begin{array}{ll} \mbox{minimize} & c^T x \\ \mbox{subject to} & \sum_{i=1}^n x_i F_i + G \preceq 0, \end{array}$$

with variable $x \in \mathbf{R}^n$, and parameters $F_1, \ldots, F_n, G \in \mathbf{S}^p$. The associated centering problem, using the log-determinant barrier function, is

$$\mbox{minimize} \quad tc^T x - \log\det(-\textstyle\sum_{i=1}^n x_i F_i - G).$$

The Newton step $\Delta x_{\rm nt}$ is found from $H\Delta x_{\rm nt} = -g$, where the Hessian and gradient are given by

$$\begin{array}{rcll} H_{ij} & = & \mathbf{tr}(S^{-1}F_iS^{-1}F_j), & i,\ j = 1, \ldots, n \\ g_i & = & tc_i + \mathbf{tr}(S^{-1}F_i), & i = 1, \ldots, n, \end{array}$$

where $S = -\sum_{i=1}^n x_i F_i - G$. One standard approach is to form H (and g), and then solve the Newton equation via Cholesky factorization.

We first consider the unstructured case, *i.e.*, we assume all matrices are dense. We will also just keep track of the order in the flop count, with respect to the problem dimensions n and p. We first form S, which costs order np^2 flops. We then compute the matrices $S^{-1}F_i$, for each i, via Cholesky factorization of S, and then back substitution with the columns of F_i (or forming S^{-1} and multiplying by F_i). This cost is order p^3 for each i, so the total cost is order np^3. Finally, we form H_{ij} as the inner product of the matrices $S^{-1}F_i$ and $S^{-1}F_j$, which costs order p^2 flops. Since we do this for $n(n+1)/2$ such pairs, the cost is order n^2p^2. Solving for the Newton direction costs order n^3. The dominating order is thus $\max\{np^3, n^2p^2, n^3\}$.

It is not possible, in general, to exploit sparsity in the matrices F_i and G, since H is often dense, even when F_i and G are sparse. One exception is when F_i and G have a common block diagonal structure, in which case all the operations described above can be carried out block by block.

It is often possible to exploit (common) sparsity in F_i and G to form the (dense) Hessian H more efficiently. If we can find an ordering that results in S having a reasonably sparse Cholesky factor, then we can compute the matrices $S^{-1}F_i$ efficiently, and form H_{ij} far more efficiently.

One interesting example that arises frequently is an SDP with matrix inequality

$$\mathbf{diag}(x) \preceq B.$$

This corresponds to $F_i = E_{ii}$, where E_{ii} is the matrix with i, i entry one and all others zero. In this case, the matrix H can be found very efficiently:

$$H_{ij} = (S^{-1})_{ij}^2,$$

where $S = B - \mathbf{diag}(x)$. The cost of forming H is thus the cost of forming S^{-1}, which is at most (*i.e.*, when no other structure is exploited) order n^3.

11.8.4 Network rate optimization

We consider a variation on the optimal network flow problem described in §10.4.3 (page 550), which is sometimes called the *network rate optimization problem*. The network is described as a directed graph with L arcs or links. Goods, or packets of information, travel on the network, passing through the links. The network supports n *flows*, with (nonnegative) *rates* $x_1, \ldots, x_n$, which are the optimization variables. Each flow moves along a fixed, or pre-determined, *path* (or *route*) in the network, from a source node to a destination node. Each link can support multiple flows passing through it. The *total traffic* on a link is the sum of the flow rates of the flows that travel over the link. Each link has a positive *capacity*, which is the maximum total traffic it can handle.

We can describe these link capacity limits using the *flow-link incidence matrix* $A \in \mathbf{R}^{L\times n}$, defined as

$$A_{ij} = \begin{cases} 1 & \text{flow } j \text{ passes through link } i \\ 0 & \text{otherwise.} \end{cases}$$

The total traffic on link i is then given by $(Ax)_i$, so the link capacity constraints can be expressed as $Ax \preceq c$, where c_i is the capacity of link i. Usually each path passes through only a small fraction of the total number of links, so the matrix A is sparse.

In the network rate problem the paths are fixed (and encoded in the matrix A, which is a problem parameter); the variables are the flow rates x_i. The objective is to choose the flow rates to maximize a separable utility function U, given by

$$U(x) = U_1(x_1) + \cdots + U_n(x_n).$$

We assume that each U_i (and hence, U) is concave and nondecreasing. We can think of $U_i(x_i)$ as the income derived from supporting the ith flow at rate x_i; $U(x)$ is then the total income associated with the flows. The network rate optimization problem is then

$$\begin{array}{ll} \text{maximize} & U(x) \\ \text{subject to} & Ax \preceq c, \quad x \succeq 0, \end{array} \tag{11.62}$$

which is a convex optimization problem.

Let us apply the barrier method to solve this problem. At each step we must minimize a function of the form

$$-tU(x) - \sum_{i=1}^{L} \log(c - Ax)_i - \sum_{j=1}^{n} \log x_j,$$

using Newton's method. The Newton step $\Delta x_{\rm nt}$ is found by solving the linear equations

$$(D_0 + A^T D_1 A + D_2)\Delta x_{\rm nt} = -g,$$

where

$$\begin{array}{rcl} D_0 & = & -t\,\mathbf{diag}(U_1''(x), \ldots, U_n''(x)) \\ D_1 & = & \mathbf{diag}(1/(c - Ax)_1^2, \ldots, 1/(c - Ax)_L^2) \\ D_2 & = & \mathbf{diag}(1/x_1^2, \ldots, 1/x_n^2) \end{array}$$

are diagonal matrices, and $g \in \mathbf{R}^n$. We can describe the sparsity structure of this $n \times n$ coefficient matrix precisely:

$$(D_0 + A^T D_1 A + D_2)_{ij} \neq 0$$

if and only if flow i and flow j share a link. If the paths are relatively short, and each link has relatively few paths passing through it, then this matrix is sparse, so a sparse Cholesky factorization can be used. We can also solve the Newton system efficiently when some, but not too many, of the rows and columns are relatively dense. This occurs when a few of the flows intersect with a large number of the other flows, which might occur if a few flows are relatively long.

We can also use the matrix inversion lemma to compute the Newton step by solving a system with $L \times L$ coefficient matrix, with form

$$(D_1^{-1} + A(D_0 + D_2)^{-1}A^T)y = -A(D_0 + D_2)^{-1}g,$$

and then computing

$$\Delta x_{\rm nt} = -(D_0 + D_2)^{-1}(g + A^T y).$$

Here too we can precisely describe the sparsity pattern:

$$(D_1^{-1} + A(D_0 + D_2)^{-1}A^T)_{ij} \neq 0$$

if and only if there is a path that passes through link i and link j. If most paths are short, this matrix is sparse. This matrix will be sparse, with a few dense rows and columns, if there are a few bottlenecks, *i.e.*, a few links over which many flows travel.

Bibliography

The early history of the barrier method is described in detail by Fiacco and McCormick [FM90, §1.2]. The method was a popular algorithm for convex optimization in the 1960s, along with closely related techniques such as the method of centers (Liêũ and Huard [LH66]; see also exercise 11.11), and penalty (or exterior-point) methods [FM90, §4]. Interest declined in the 1970s amid concerns about the ill-conditioning of the Newton equations of the centering problem (11.6) for high values of t.

The barrier method regained popularity in the 1980s, after Gill, Murray, Saunders, Tomlin, and Wright [GMS$^+$86] pointed out the close connections with Karmarkar's polynomial-time projective algorithm for linear programming [Kar84]. The focus of research throughout the 1980s remained on linear (and to a lesser extent, quadratic) programming, resulting in different variations of the basic interior-point methods, and improved worst-case complexity results (see Gonzaga [Gon92]). Primal-dual methods emerged as the algorithms of choice for practical implementations (see Mehrotra [Meh92], Lustig, Marsten, and Shanno [LMS94], Wright [Wri97]).

In their 1994 book, Nesterov and Nemirovski extended the complexity theory of linear programming interior-point methods to nonlinear convex optimization problems, using the convergence theory of Newton's method for self-concordant functions. They also developed interior-point methods for problems with generalized inequalities, and discussed ways of reformulating problems to satisfy the self-concordance assumption. The geometric programming reformulation on page 587, for example, is from [NN94, §6.3.1].

As mentioned on page 585, the complexity analysis shows that, contrary to what one might expect, the centering problems in the barrier method do not become more difficult as t increases, at least not in exact arithmetic. Practical experience, supported by theoretical results (Forsgren, Gill, and Wright [FGW02, §4.3.2], Nocedal and Wright [NW99, page 525]), also indicates that the effects of ill-conditioning on the computed solution of the Newton system are more benign than thought earlier.

Recent research on interior-point methods has concentrated on extending the primal-dual methods for linear programming, which converge faster and reach higher accuracies than (primal) barrier methods, to nonlinear convex problems. One popular approach, along the lines of the simple primal-dual method of §11.7, is based on linearizing modified KKT equations for a convex optimization problem in standard form, *i.e.*, problem (11.1). More sophisticated algorithms of this type differ from algorithm 11.2 in the strategy used to select t (which is crucial to achieve superlinear asymptotic convergence), and the line search. We refer to Wright [Wri97, chapter 8], Ralph and Wright [RW97], den Hertog [dH93], Terlaky [Ter96], and the survey by Forsgren, Gill, and Wright [FGW02, §5] for details and references.

Other authors adopt the cone programming framework as starting point for extending primal-dual interior-point methods for linear programming to convex optimization (see for example, Nesterov and Todd [NT98]). This approach has resulted in efficient and accurate primal-dual methods for semidefinite and second-order programming (see the surveys by Todd [Tod01] and Alizadeh and Goldfarb [AG03]).

As for linear programming, primal-dual methods for semidefinite programming are usually described as variations of Newton's method applied to modified KKT equations. Unlike in linear programming, however, the linearization can be carried out in many different ways, which lead to different search directions and algorithms; see Helmberg, Rendl, Vanderbei, and Wolkowicz [HRVW96], Kojima, Shindo, and Harah [KSH97], Monteiro [Mon97], Nesterov and Todd [NT98], Zhang [Zha98], Alizadeh, Haeberly, and Overton [AHO98], and Todd, Toh, and Tütüncü [TTT98].

Great progress has also been made in the area of initialization and infeasibility detection. Homogeneous self-dual formulations provide an elegant and efficient alternative to the classical two-phase approach of §11.4; see Ye, Todd, and Mizuno [YTM94], Xu, Hung, and Ye [XHY96], Andersen and Ye [AY98] and Luo, Sturm, and Zhang [LSZ00] for details.

The primal-dual interior-point methods for semidefinite and second-order cone programming have been implemented in a number of software packages, including SeDuMi [Stu99], SDPT3 [TTT02], SDPA [FKN98], CSDP [Bor02], and DSDP [BY02], A user-friendly interface to several of these codes is provided by YALMIP [Löf04].

The following books document the recent developments in this rapidly advancing field in greater detail: Vanderbei [Van96], Wright [Wri97], Roos, Terlaky, and Vial [RTV97] Ye [Ye97], Wolkowicz, Saigal, and Vandenberghe [WSV00], Ben-Tal and Nemirovski, [BTN01], Renegar [Ren01], and Peng, Roos, and Terlaky [PRT02].

Exercises

The barrier method

11.1 *Barrier method example.* Consider the simple problem

$$\begin{array}{ll} \text{minimize} & x^2+1 \\ \text{subject to} & 2 \leq x \leq 4, \end{array}$$

which has feasible set $[2,4]$, and optimal point $x^\star = 2$. Plot f_0, and $tf_0 + \phi$, for several values of $t > 0$, versus x. Label $x^\star(t)$.

11.2 What happens if the barrier method is applied to the LP

$$\begin{array}{ll} \text{minimize} & x_2 \\ \text{subject to} & x_1 \leq x_2, \quad 0 \leq x_2, \end{array}$$

with variable $x \in \mathbf{R}^2$?

11.3 *Boundedness of centering problem.* Suppose the sublevel sets of (11.1),

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i=1,\ldots,m \\ & Ax = b, \end{array}$$

are bounded. Show that the sublevel sets of the associated centering problem,

$$\begin{array}{ll} \text{minimize} & tf_0(x) + \phi(x) \\ \text{subject to} & Ax = b, \end{array}$$

are bounded.

11.4 *Adding a norm bound to ensure strong convexity of the centering problem.* Suppose we add the constraint $x^Tx \leq R^2$ to the problem (11.1):

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i=1,\ldots,m \\ & Ax = b \\ & x^Tx \leq R^2. \end{array}$$

Let $\tilde{\phi}$ denote the logarithmic barrier function for this modified problem. Find $a > 0$ for which $\nabla^2(tf_0(x) + \tilde{\phi}(x)) \succeq aI$ holds, for all feasible x.

11.5 *Barrier method for second-order cone programming.* Consider the SOCP (without equality constraints, for simplicity)

$$\begin{array}{ll} \text{minimize} & f^Tx \\ \text{subject to} & \|A_ix + b_i\|_2 \leq c_i^Tx + d_i, \quad i=1,\ldots,m. \end{array} \tag{11.63}$$

The constraint functions in this problem are not differentiable (since the Euclidean norm $\|u\|_2$ is not differentiable at $u = 0$) so the (standard) barrier method cannot be applied. In §11.6, we saw that this SOCP can be solved by an extension of the barrier method that handles generalized inequalities. (See example 11.8, page 599, and page 601.) In this exercise, we show how the standard barrier method (with scalar constraint functions) can be used to solve the SOCP.

We first reformulate the SOCP as

$$\begin{array}{ll} \text{minimize} & f^Tx \\ \text{subject to} & \|A_ix + b_i\|_2^2/(c_i^Tx + d_i) \leq c_i^Tx + d_i, \quad i=1,\ldots,m \\ & c_i^Tx + d_i \geq 0, \quad i=1,\ldots,m. \end{array} \tag{11.64}$$

The constraint function

$$f_i(x) = \frac{\|A_i x + b_i\|_2^2}{c_i^T x + d_i} - c_i^T x - d_i$$

is the composition of a quadratic-over-linear function with an affine function, and is twice differentiable (and convex), provided we define its domain as $\mathbf{dom}\, f_i = \{x \mid c_i^T x + d_i > 0\}$. Note that the two problems (11.63) and (11.64) are not exactly equivalent. If $c_i^T x^\star + d_i = 0$ for some i, where $x^\star$ is the optimal solution of the SOCP (11.63), then the reformulated problem (11.64) is not solvable; $x^\star$ is not in its domain. Nevertheless we will see that the barrier method, applied to (11.64), produces arbitrarily accurate suboptimal solutions of (11.64), and hence also for (11.63).

(a) Form the log barrier ϕ for the problem (11.64). Compare it to the log barrier that arises when the SOCP (11.63) is solved using the barrier method for generalized inequalities (in §11.6).

(b) Show that if $tf^T x + \phi(x)$ is minimized, the minimizer $x^\star(t)$ is $2m/t$-suboptimal for the problem (11.63). It follows that the standard barrier method, applied to the reformulated problem (11.64), solves the SOCP (11.63), in the sense of producing arbitrarily accurate suboptimal solutions. This is the case even though the optimal point $x^\star$ need not be in the domain of the reformulated problem (11.64).

11.6 *General barriers.* The log barrier is based on the approximation $-(1/t)\log(-u)$ of the indicator function $\widehat{I}_-(u)$ (see §11.2.1, page 563). We can also construct barriers from other approximations, which in turn yield generalizations of the central path and barrier method. Let $h : \mathbf{R} \to \mathbf{R}$ be a twice differentiable, closed, increasing convex function, with $\mathbf{dom}\, h = -\mathbf{R}_{++}$. (This implies $h(u) \to \infty$ as $u \to 0$.) One such function is $h(u) = -\log(-u)$; another example is $h(u) = -1/u$ (for $u < 0$).

Now consider the optimization problem (without equality constraints, for simplicity)

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

where f_i are twice differentiable. We define the *h-barrier* for this problem as

$$\phi_h(x) = \sum_{i=1}^m h(f_i(x)),$$

with domain $\{x \mid f_i(x) < 0,\ i = 1, \ldots, m\}$. When $h(u) = -\log(-u)$, this is the usual logarithmic barrier; when $h(u) = -1/u$, ϕ_h is called the *inverse barrier*. We define the *h-central path* as

$$x^\star(t) = \mathop{\mathrm{argmin}}\, tf_0(x) + \phi_h(x),$$

where $t > 0$ is a parameter. (We assume that for each t, the minimizer exists and is unique.)

(a) Explain why $tf_0(x) + \phi_h(x)$ is convex in x, for each $t > 0$.

(b) Show how to construct a dual feasible λ from $x^\star(t)$. Find the associated duality gap.

(c) For what functions h does the duality gap found in part (b) depend only on t and m (and no other problem data)?

11.7 *Tangent to central path.* This problem concerns $dx^\star(t)/dt$, which gives the tangent to the central path at the point $x^\star(t)$. For simplicity, we consider a problem without equality constraints; the results readily generalize to problems with equality constraints.

(a) Find an explicit expression for $dx^\star(t)/dt$. *Hint.* Differentiate the centrality equations (11.7) with respect to t.

(b) Show that $f_0(x^\star(t))$ decreases as t increases. Thus, the objective value in the barrier method decreases, as the parameter t is increased. (We already know that the duality gap, which is m/t, decreases as t increases.)

11.8 *Predictor-corrector method for centering problems.* In the standard barrier method, $x^\star(\mu t)$ is computed using Newton's method, starting from the initial point $x^\star(t)$. One alternative that has been proposed is to make an approximation or prediction $\widehat{x}$ of $x^\star(\mu t)$, and then start the Newton method for computing $x^\star(\mu t)$ from $\widehat{x}$. The idea is that this should reduce the number of Newton steps, since $\widehat{x}$ is (presumably) a better initial point than $x^\star(t)$. This method of centering is called a *predictor-corrector method*, since it first makes a *prediction* of what $x^\star(\mu t)$ is, then *corrects* the prediction using Newton's method.

The most widely used predictor is the first-order predictor, based on the tangent to the central path, explored in exercise 11.7. This predictor is given by

$$\widehat{x} = x^\star(t) + \frac{dx^\star(t)}{dt}(\mu t - t).$$

Derive an expression for the first-order predictor $\widehat{x}$. Compare it to the Newton update obtained, *i.e.*, $x^\star(t) + \Delta x_{\mathrm{nt}}$, where Δx_{nt} is the Newton step for $\mu t f_0(x) + \phi(x)$, at $x^\star(t)$. What can you say when the objective f_0 is linear? (For simplicity, you can consider a problem without equality constraints.)

11.9 *Dual feasible points near the central path.* Consider the problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m, \end{array}$$

with variable $x \in \mathbf{R}^n$. We assume the functions f_i are convex and twice differentiable. (We assume for simplicity there are no equality constraints.) Recall (from §11.2.2, page 565) that $\lambda_i = -1/(t f_i(x^\star(t)))$, $i = 1, \ldots, m$, is dual feasible, and in fact, $x^\star(t)$ minimizes $L(x, \lambda)$. This allows us to evaluate the dual function for λ, which turns out to be $g(\lambda) = f_0(x^\star(t)) - m/t$. In particular, we conclude that $x^\star(t)$ is m/t-suboptimal.

In this problem we consider what happens when a point x is close to $x^\star(t)$, but not quite centered. (This would occur if the centering steps were terminated early, or not carried out to full accuracy.) In this case, of course, we cannot claim that $\lambda_i = -1/(t f_i(x))$, $i = 1, \ldots, m$, is dual feasible, or that x is m/t-suboptimal. However, it turns out that a slightly more complicated formula does yield a dual feasible point, provided x is close enough to centered.

Let Δx_{nt} be the Newton step at x of the centering problem

$$\text{minimize} \quad t f_0(x) - \textstyle\sum_{i=1}^m \log(-f_i(x)).$$

A formula that often gives a dual feasible point when Δx_{nt} is small (*i.e.*, for x nearly centered) is

$$\lambda_i = \frac{1}{-t f_i(x)} \left(1 + \frac{\nabla f_i(x)^T \Delta x_{\mathrm{nt}}}{-f_i(x)} \right), \quad i = 1, \ldots, m.$$

In this case, the vector x *does not* minimize $L(x, \lambda)$, so there is no general formula for the dual function value $g(\lambda)$ associated with λ. (If we have an analytical expression for the dual objective, however, we can simply evaluate $g(\lambda)$.)

Verify that for a QCQP

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m, \end{array}$$

the formula for λ yields a dual feasible point (*i.e.*, $\lambda \succeq 0$ and $L(x, \lambda)$ is bounded below) when Δx_{nt} is sufficiently small.

Hint. Define

$$x_0 = x + \Delta x_{\rm nt}, \qquad x_i = x - \frac{1}{t\lambda_i f_i(x)}\Delta x_{\rm nt}, \quad i = 1, \ldots, m.$$

Show that

$$\nabla f_0(x_0) + \sum_{i=1}^m \lambda_i \nabla f_i(x_i) = 0.$$

Now use $f_i(z) \geq f_i(x_i) + \nabla f_i(x_i)^T(z - x_i)$, $i = 0, \ldots, m$, to derive a lower bound on $L(z, \lambda)$.

11.10 *Another parametrization of the central path.* We consider the problem (11.1), with central path $x^\star(t)$ for $t > 0$, defined as the solution of

$$\begin{array}{ll} \mbox{minimize} & tf_0(x) - \sum_{i=1}^m \log(-f_i(x)) \\ \mbox{subject to} & Ax = b. \end{array}$$

In this problem we explore another parametrization of the central path.

For $u > p^\star$, let $z^\star(u)$ denote the solution of

$$\begin{array}{ll} \mbox{minimize} & -\log(u - f_0(x)) - \sum_{i=1}^m \log(-f_i(x)) \\ \mbox{subject to} & Ax = b. \end{array}$$

Show that the curve defined by $z^\star(u)$, for $u > p^\star$, is the central path. (In other words, for each $u > p^\star$, there is a $t > 0$ for which $x^\star(t) = z^\star(u)$, and conversely, for each $t > 0$, there is an $u > p^\star$ for which $z^\star(u) = x^\star(t)$).

11.11 *Method of analytic centers.* In this problem we consider a variation on the barrier method, based on the parametrization of the central path described in exercise 11.10. For simplicity, we consider a problem with no equality constraints,

$$\begin{array}{ll} \mbox{minimize} & f_0(x) \\ \mbox{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m. \end{array}$$

The method of analytic centers starts with any strictly feasible initial point $x^{(0)}$, and any $u^{(0)} > f_0(x^{(0)})$. We then set

$$u^{(1)} = \theta u^{(0)} + (1 - \theta) f_0(x^{(0)}),$$

where $\theta \in (0, 1)$ is an algorithm parameter (usually chosen small), and then compute the next iterate as

$$x^{(1)} = z^\star(u^{(1)})$$

(using Newton's method, starting from $x^{(0)}$). Here $z^\star(s)$ denotes the minimizer of

$$-\log(s - f_0(x)) - \sum_{i=1}^m \log(-f_i(x)),$$

which we assume exists and is unique. This process is then repeated.

The point $z^\star(s)$ is the *analytic center* of the inequalities

$$f_0(x) \leq s, \quad f_1(x) \leq 0, \ldots, f_m(x) \leq 0,$$

hence the algorithm name.

Show that the method of centers works, *i.e.*, $x^{(k)}$ converges to an optimal point. Find a stopping criterion that guarantees that x is ϵ-suboptimal, where $\epsilon > 0$.

Hint. The points $x^{(k)}$ are on the central path; see exercise 11.10. Use this to show that

$$u^+ - p^\star \leq \frac{m + \theta}{m + 1}(u - p^\star),$$

where u and u^+ are the values of u on consecutive iterations.

11.12 *Barrier method for convex-concave games.* We consider a convex-concave game with inequality constraints,

$$\begin{array}{lll} \text{minimize}_w \text{ maximize}_z & f_0(w,z) & \\ \text{subject to} & f_i(w) \leq 0, & i=1,\ldots,m \\ & \tilde{f}_i(z) \leq 0, & i=1,\ldots,\tilde{m}. \end{array}$$

Here $w \in \mathbf{R}^n$ is the variable associated with minimizing the objective, and $z \in \mathbf{R}^{\tilde{n}}$ is the variable associated with maximizing the objective. The constraint functions f_i and $\tilde{f}_i$ are convex and differentiable, and the objective function f_0 is differentiable and convex-concave, *i.e.*, convex in w, for each z, and concave in z, for each w. We assume for simplicity that $\mathbf{dom}\, f_0 = \mathbf{R}^n \times \mathbf{R}^{\tilde{n}}$.

A *solution* or *saddle-point* for the game is a pair $w^\star$, $z^\star$, for which

$$f_0(w^\star, z) \leq f_0(w^\star, z^\star) \leq f_0(w, z^\star)$$

holds for every feasible w and z. (For background on convex-concave games and functions, see §5.4.3, §10.3.4 and exercises 3.14, 5.24, 5.25, 10.10, and 10.13.) In this exercise we show how to solve this game using an extension of the barrier method, and the infeasible start Newton method (see §10.3).

(a) Let $t > 0$. Explain why the function

$$tf_0(w,z) - \sum_{i=1}^{m} \log(-f_i(w)) + \sum_{i=1}^{\tilde{m}} \log(-\tilde{f}_i(z))$$

is convex-concave in (w,z). We will assume that it has a unique saddle-point, $(w^\star(t), z^\star(t))$, which can be found using the infeasible start Newton method.

(b) As in the barrier method for solving a convex optimization problem, we can derive a simple bound on the suboptimality of $(w^\star(t), z^\star(t))$, which depends only on the problem dimensions, and decreases to zero as t increases. Let W and Z denote the feasible sets for w and z,

$$W = \{w \mid f_i(w) \leq 0,\ i=1,\ldots,m\}, \qquad Z = \{z \mid \tilde{f}_i(z) \leq 0,\ i=1,\ldots,\tilde{m}\}.$$

Show that

$$\begin{array}{rcl} f_0(w^\star(t), z^\star(t)) & \leq & \inf_{w \in W} f_0(w, z^\star(t)) + \dfrac{m}{t}, \\ f_0(w^\star(t), z^\star(t)) & \geq & \sup_{z \in Z} f_0(w^\star(t), z) - \dfrac{\tilde{m}}{t}, \end{array}$$

and therefore

$$\sup_{z \in Z} f_0(w^\star(t), z) - \inf_{w \in W} f_0(w, z^\star(t)) \leq \frac{m + \tilde{m}}{t}.$$

Self-concordance and complexity analysis

11.13 *Self-concordance and negative entropy.*

(a) Show that the negative entropy function $x \log x$ (on $\mathbf{R}_{++}$) is *not* self-concordant.

(b) Show that for any $t > 0$, $tx \log x - \log x$ is self-concordant (on $\mathbf{R}_{++}$).

11.14 *Self-concordance and the centering problem.* Let ϕ be the logarithmic barrier function of problem (11.1). Suppose that the sublevel sets of (11.1) are bounded, and that $tf_0 + \phi$ is closed and self-concordant. Show that $t\nabla^2 f_0(x) + \nabla^2 \phi(x) \succ 0$, for all $x \in \mathbf{dom}\, \phi$. *Hint.* See exercises 9.17 and 11.3.

Barrier method for generalized inequalities

11.15 *Generalized logarithm is K-increasing.* Let ψ be a generalized logarithm for the proper cone K. Suppose $y \succ_K 0$.

(a) Show that $\nabla\psi(y) \succeq_{K^*} 0$, *i.e.*, that ψ is K-nondecreasing. *Hint.* If $\nabla\psi(y) \not\succeq_{K^*} 0$, then there is some $w \succ_K 0$ for which $w^T\nabla\psi(y) \leq 0$. Use the inequality $\psi(sw) \leq \psi(y) + \nabla\psi(y)^T(sw - y)$, with $s > 0$.

(b) Now show that $\nabla\psi(y) \succ_{K^*} 0$, *i.e.*, that ψ is K-increasing. *Hint.* Show that $\nabla^2\psi(y) \prec 0$, $\nabla\psi(y) \succeq_{K^*} 0$ imply $\nabla\psi(y) \succ_{K^*} 0$.

11.16 [NN94, page 41] *Properties of a generalized logarithm.* Let ψ be a generalized logarithm for the proper cone K, with degree θ. Prove that the following properties hold at any $y \succ_K 0$.

(a) $\nabla\psi(sy) = \nabla\psi(y)/s$ for all $s > 0$.

(b) $\nabla\psi(y) = -\nabla^2\psi(y)y$.

(c) $y^T\nabla\psi^2(y)y = -\theta$.

(d) $\nabla\psi(y)^T\nabla^2\psi(y)^{-1}\nabla\psi(y) = -\theta$.

11.17 *Dual generalized logarithm.* Let ψ be a generalized logarithm for the proper cone K, with degree θ. Show that the dual generalized logarithm $\overline{\psi}$, defined in (11.49), satisfies

$$\overline{\psi}(sv) = \psi(v) + \theta\log s,$$

for $v \succ_{K^*} 0$, $s > 0$.

11.18 Is the function

$$\psi(y) = \log\left(y_{n+1} - \frac{\sum_{i=1}^n y_i^2}{y_{n+1}}\right),$$

with $\mathbf{dom}\,\psi = \{y \in \mathbf{R}^{n+1} \mid y_{n+1} > \sum_{i=1}^n y_i^2\}$, a generalized logarithm for the second-order cone in $\mathbf{R}^{n+1}$?

Implementation

11.19 *Yet another method for computing the Newton step.* Show that the Newton step for the barrier method, which is given by the solution of the linear equations (11.14), can be found by solving a *larger* set of linear equations with coefficient matrix

$$\left[\begin{array}{ccc} t\nabla^2 f_0(x) + \sum_i \frac{1}{-f_i(x)}\nabla^2 f_i(x) & Df(x)^T & A^T \\ Df(x) & -\mathbf{diag}(f(x))^2 & 0 \\ A & 0 & 0 \end{array}\right]$$

where $f(x) = (f_1(x), \ldots, f_m(x))$.

For what types of problem structure might solving this larger system be interesting?

11.20 *Network rate optimization via the dual problem.* In this problem we examine a dual method for solving the network rate optimization problem of §11.8.4. To simplify the presentation we assume that the utility functions U_i are strictly concave, with $\mathbf{dom}\,U_i = \mathbf{R}_{++}$, and that they satisfy $U_i'(x_i) \to \infty$ as $x_i \to 0$ and $U_i'(x_i) \to 0$ as $x_i \to \infty$.

(a) Express the dual problem of (11.62) in terms of the conjugate utility functions $V_i = (-U_i)^*$, defined as

$$V_i(\lambda) = \sup_{x>0}(\lambda x + U_i(x)).$$

Show that $\mathbf{dom}\,V_i = -\mathbf{R}_{++}$, and that for each $\lambda < 0$ there is a unique x with $U_i'(x) = -\lambda$.

(b) Describe a barrier method for the dual problem. Compare the complexity per iteration with the complexity of the method in §11.8.4. Distinguish the same two cases as in §11.8.4 (A^TA is sparse and AA^T is sparse).

Numerical experiments

11.21 *Log-Chebyshev approximation with bounds.* We consider an approximation problem: find $x \in \mathbf{R}^n$, that satisfies the variable bounds $l \preceq x \preceq u$, and yields $Ax \approx b$, where $b \in \mathbf{R}^m$. You can assume that $l \prec u$, and $b \succ 0$ (for reasons we explain below). We let a_i^T denote the ith row of the matrix A.

We judge the approximation $Ax \approx b$ by the *maximum fractional deviation*, which is

$$\max_{i=1,\ldots,n} \max\{(a_i^T x)/b_i, b_i/(a_i^T x)\} = \max_{i=1,\ldots,n} \frac{\max\{a_i^T x, b_i\}}{\min\{a_i^T x, b_i\}},$$

when $Ax \succ 0$; we define the maximum fractional deviation as ∞ if $Ax \not\succ 0$.

The problem of minimizing the maximum fractional deviation is called the *fractional Chebyshev approximation problem*, or the *logarithmic Chebyshev approximation problem*, since it is equivalent to minimizing the objective

$$\max_{i=1,\ldots,n} |\log a_i^T x - \log b_i|.$$

(See also exercise 6.3, part (c).)

(a) Formulate the fractional Chebyshev approximation problem (with variable bounds) as a convex optimization problem with twice differentiable objective and constraint functions.

(b) Implement a barrier method that solves the fractional Chebyshev approximation problem. You can assume an initial point $x^{(0)}$, satisfying $l \prec x^{(0)} \prec u$, $Ax^{(0)} \succ 0$, is known.

11.22 *Maximum volume rectangle inside a polyhedron.* Consider the problem described in exercise 8.16, *i.e.*, finding the maximum volume rectangle $\mathcal{R} = \{x \mid l \preceq x \preceq u\}$ that lies in a polyhedron described by a set of linear inequalities, $\mathcal{P} = \{x \mid Ax \preceq b\}$. Implement a barrier method for solving this problem. You can assume that $b \succ 0$, which means that for small $l \prec 0$ and $u \succ 0$, the rectangle $\mathcal{R}$ lies inside $\mathcal{P}$.

Test your implementation on several simple examples. Find the maximum volume rectangle that lies in the polyhedron defined by

$$A = \begin{bmatrix} 0 & -1 \\ 2 & -4 \\ 2 & 1 \\ -4 & 4 \\ -4 & 0 \end{bmatrix}, \qquad b = \mathbf{1}.$$

Plot this polyhedron, and the maximum volume rectangle that lies inside it.

11.23 *SDP bounds and heuristics for the two-way partitioning problem.* In this exercise we consider the two-way partitioning problem (5.7), described on page 219, and also in exercise 5.39:

$$\begin{array}{ll} \text{minimize} & x^T W x \\ \text{subject to} & x_i^2 = 1, \quad i = 1, \ldots, n, \end{array} \qquad (11.65)$$

with variable $x \in \mathbf{R}^n$. We assume, without loss of generality, that $W \in \mathbf{S}^n$ satisfies $W_{ii} = 0$. We denote the optimal value of the partitioning problem as $p^\star$, and $x^\star$ will denote an optimal partition. (Note that $-x^\star$ is also an optimal partition.)

The Lagrange dual of the two-way partitioning problem (11.65) is given by the SDP

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0, \end{array} \qquad (11.66)$$

with variable $\nu \in \mathbf{R}^n$. The dual of this SDP is

$$\begin{array}{ll} \text{minimize} & \mathbf{tr}(WX) \\ \text{subject to} & X \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n, \end{array} \tag{11.67}$$

with variable $X \in \mathbf{S}^n$. (This SDP can be interpreted as a relaxation of the two-way partitioning problem (11.65); see exercise 5.39.) The optimal values of these two SDPs are equal, and give a lower bound, which we denote $d^\star$, on the optimal value $p^\star$. Let $\nu^\star$ and $X^\star$ denote optimal points for the two SDPs.

(a) Implement a barrier method that solves the SDP (11.66) and its dual (11.67), given the weight matrix W. Explain how you obtain nearly optimal ν and X, give formulas for any Hessians and gradients that your method requires, and explain how you compute the Newton step. Test your implementation on some small problem instances, comparing the bound you find with the optimal value (which can be found by checking the objective value of all 2^n partitions). Try your implementation on a randomly chosen problem instance large enough that you cannot find the optimal partition by exhaustive search (*e.g.*, $n = 100$).

(b) *A heuristic for partitioning.* In exercise 5.39, you found that if $X^\star$ has rank one, then it must have the form $X^\star = x^\star(x^\star)^T$, where $x^\star$ is optimal for the two-way partitioning problem. This suggests the following simple heuristic for finding a good partition (if not the best): solve the SDPs above, to find $X^\star$ (and the bound $d^\star$). Let v denote an eigenvector of $X^\star$ associated with its largest eigenvalue, and let $\hat{x} = \mathbf{sign}(v)$. The vector $\hat{x}$ is our guess for a good partition.

Try this heuristic on some small problem instances, and the large problem instance you used in part (a). Compare the objective value of your heuristic partition, $\hat{x}^T W \hat{x}$, with the lower bound $d^\star$.

(c) *A randomized method.* Another heuristic technique for finding a good partition, given the solution $X^\star$ of the SDP (11.67), is based on *randomization*. The method is simple: we generate independent samples $x^{(1)}, \ldots, x^{(K)}$ from a normal distribution on $\mathbf{R}^n$, with zero mean and covariance $X^\star$. For each sample we consider the heuristic approximate solution $\hat{x}^{(k)} = \mathbf{sign}(x^{(k)})$. We then take the best among these, *i.e.*, the one with lowest cost. Try out this procedure on some small problem instances, and the large problem instance you considered in part (a).

(d) *A greedy heuristic refinement.* Suppose you are given a partition x, *i.e.*, $x_i \in \{-1, 1\}$, $i = 1, \ldots, n$. How does the objective value change if we move element i from one set to the other, *i.e.*, change x_i to $-x_i$? Now consider the following simple greedy algorithm: given a starting partition x, move the element that gives the largest reduction in the objective. Repeat this procedure until no reduction in objective can be obtained by moving an element from one set to the other.

Try this heuristic on some problem instances, including the large one, starting from various initial partitions, including $x = \mathbf{1}$, the heuristic approximate solution found in part (b), and the randomly generated approximate solutions found in part (c). How much does this greedy refinement improve your approximate solutions from parts (b) and (c)?

11.24 *Barrier and primal-dual interior-point methods for quadratic programming.* Implement a barrier method, and a primal-dual method, for solving the QP (without equality constraints, for simplicity)

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x \\ \text{subject to} & Ax \preceq b, \end{array}$$

with $A \in \mathbf{R}^{m \times n}$. You can assume a strictly feasible initial point is given. Test your codes on several examples. For the barrier method, plot the duality gap versus Newton steps. For the primal-dual interior-point method, plot the surrogate duality gap and the norm of the dual residual versus iteration number.