CHAPTER 18

A Path-Following Method

In this chapter, we define an interior-point method for linear programming that is called a path-following method. Recall that for the simplex method we required a two-phase solution procedure. The path-following method is a one-phase method. This means that the method can begin from a point that is neither primal nor dual feasible and it will proceed from there directly to the optimal solution. Hence, we start with an arbitrary choice of strictly positive values for all the primal and dual variables, i.e., $(x, w, y, z) > 0$, and then iteratively update these values as follows:

(1) Estimate an appropriate value for μ (i.e., smaller than the "current" value but not too small).
(2) Compute step directions $(\Delta x, \Delta w, \Delta y, \Delta z)$ pointing approximately at the point $(x_\mu, w_\mu, y_\mu, z_\mu)$ on the central path.
(3) Compute a step length parameter θ such that the new point

$$\begin{aligned} \tilde{x} &= x + \theta \Delta x, & \tilde{y} &= y + \theta \Delta y, \\ \tilde{w} &= w + \theta \Delta w, & \tilde{z} &= z + \theta \Delta z \end{aligned}$$

continues to have strictly positive components.
(4) Replace (x, w, y, z) with the new solution $(\tilde{x}, \tilde{w}, \tilde{y}, \tilde{z})$.

To fully define the path-following method, it suffices to make each of these four steps precise. Since the second step is in some sense the most fundamental, we start by describing that one after which we turn our attention to the others.

1. Computing Step Directions

Our aim is to find $(\Delta x, \Delta w, \Delta y, \Delta z)$ such that the new point $(x + \Delta x, w + \Delta w, y + \Delta y, z + \Delta z)$ lies approximately on the primal–dual central path at the point $(x_\mu, w_\mu, y_\mu, z_\mu)$. Recalling the defining equations for this point on the central path,

$$\begin{aligned} Ax + w &= b \\ A^T y - z &= c \\ XZe &= \mu e \\ YWe &= \mu e, \end{aligned}$$

we see that the new point $(x + \Delta x, w + \Delta w, y + \Delta y, z + \Delta z)$, if it were to lie exactly on the central path at μ, would be defined by

R.J. Vanderbei, *Linear Programming*, International Series in Operations Research & Management Science 196, DOI 10.1007/978-1-4614-7630-6_18,

$$\begin{aligned} A(x+\Delta x) + (w+\Delta w) &= b \\ A^T(y+\Delta y) - (z+\Delta z) &= c \\ (X+\Delta X)(Z+\Delta Z)e &= \mu e \\ (Y+\Delta Y)(W+\Delta W)e &= \mu e. \end{aligned}$$

Thinking of (x, w, y, z) as data and $(\Delta x, \Delta w, \Delta y, \Delta z)$ as unknowns, we rewrite these equations with the unknowns on the left and the data on the right:

$$\begin{aligned} A\Delta x + \Delta w &= b - Ax - w \ =: \rho \\ A^T\Delta y - \Delta z &= c - A^T y + z =: \sigma \\ Z\Delta x + X\Delta z + \Delta X \Delta Z e &= \mu e - XZe \\ W\Delta y + Y\Delta w + \Delta Y \Delta W e &= \mu e - YWe. \end{aligned}$$

Note that we have introduced abbreviated notations, ρ and σ, for the first two right-hand sides. These two vectors represent the *primal infeasibility* and the *dual infeasibility*, respectively.

Now, just as before, these equations form a system of nonlinear equations (this time for the "delta" variables). We want to have a linear system, so at this point we simply drop the nonlinear terms to arrive at the following linear system:

$$A\Delta x + \Delta w = \rho \tag{18.1}$$

$$A^T\Delta y - \Delta z = \sigma \tag{18.2}$$

$$Z\Delta x + X\Delta z = \mu e - XZe \tag{18.3}$$

$$W\Delta y + Y\Delta w = \mu e - YWe. \tag{18.4}$$

This system of equations is a linear system of $2n + 2m$ equations in $2n + 2m$ unknowns. We will show later that this system is nonsingular (under the mild assumption that A has full rank) and therefore that it has a unique solution that defines the step directions for the path-following method. Chapters 19 and 20 are devoted to studying methods for efficiently solving systems of this type.

If the business of dropping the nonlinear "delta" terms strikes you as bold, let us remark that this is the most common approach to solving nonlinear systems of equations. The method is called Newton's method. It is described briefly in the next section.

2. Newton's Method

Given a function

$$F(\xi) = \begin{bmatrix} F_1(\xi) \\ F_2(\xi) \\ \vdots \\ F_N(\xi) \end{bmatrix}, \qquad \xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_N \end{bmatrix},$$

from $\mathbb{R}^N$ into $\mathbb{R}^N$, a common problem is to find a point $\xi^* \in \mathbb{R}^N$ for which $F(\xi^*) = 0$. Such a point is called a *root* of F. Newton's method is an iterative method for solving this problem. One step of the method is defined as follows.

Given any point $\xi \in \mathbb{R}^N$, the goal is to find a *step direction* $\Delta\xi$ for which $F(\xi + \Delta\xi) = 0$. Of course, for a nonlinear F it is not possible to find such a step direction. Hence, it is approximated by the first two terms of its Taylor's series expansion,

$$F(\xi + \Delta\xi) \approx F(\xi) + F'(\xi)\Delta\xi,$$

where

$$F'(\xi) = \begin{bmatrix} \frac{\partial F_1}{\partial \xi_1} & \frac{\partial F_1}{\partial \xi_2} & \cdots & \frac{\partial F_1}{\partial \xi_N} \\ \frac{\partial F_2}{\partial \xi_1} & \frac{\partial F_2}{\partial \xi_2} & \cdots & \frac{\partial F_2}{\partial \xi_N} \\ \vdots & \vdots & & \vdots \\ \frac{\partial F_N}{\partial \xi_1} & \frac{\partial F_N}{\partial \xi_2} & \cdots & \frac{\partial F_N}{\partial \xi_N} \end{bmatrix}.$$

The approximation is linear in $\Delta\xi$. Hence, equating it to zero gives a linear system to solve for the step direction:

$$F'(\xi)\Delta\xi = -F(\xi).$$

Given $\Delta\xi$, Newton's method updates the current solution ξ by replacing it with $\xi + \Delta\xi$. The process continues until the current solution is approximately a root (i.e., $F(\xi) \approx 0$). Simple one-dimensional examples given in every elementary calculus text illustrate that this method works well, when it works, but it can fail if F is not well behaved and the initial point is too far from a solution.

Let's return now to the problem of finding a point on the central path. Letting

$$\xi = \begin{bmatrix} x \\ w \\ y \\ z \end{bmatrix}$$

and

$$F(\xi) = \begin{bmatrix} Ax + w - b \\ A^T y - z - c \\ XZe - \mu e \\ YWe - \mu e \end{bmatrix},$$

we see that the set of equations defining $(x_\mu, w_\mu, y_\mu, z_\mu)$ is a root of F. The matrix of derivatives of F is given by

$$F'(\xi) = \begin{bmatrix} A & I & 0 & 0 \\ 0 & 0 & A^T & -I \\ Z & 0 & 0 & X \\ 0 & Y & W & 0 \end{bmatrix}.$$

Noting that

$$\Delta\xi = \begin{bmatrix} \Delta x \\ \Delta w \\ \Delta y \\ \Delta z \end{bmatrix},$$

it is easy to see that the Newton direction coincides with the direction obtained by solving equations (18.1)–(18.4).

3. Estimating an Appropriate Value for the Barrier Parameter

We need to say how to pick μ. If μ is chosen to be too large, then the sequence could converge to the analytic center of the feasible set, which is not our intention. If, on the other hand, μ is chosen to be too small, then the sequence could stray too far from the central path and the algorithm could *jam* into the boundary of the feasible set at a place that is suboptimal. The trick is to find a reasonable compromise between these two extremes. To do this, we first figure out a value that represents, in some sense, the current value of μ and we then choose something smaller than that, say a fixed fraction of it.

We are given a point (x, w, y, z) that is almost certainly off the central path. If it were on the central path, then there are several formulas by which we could recover the corresponding value of μ. For example, we could just compute $z_j x_j$ for any fixed index j. Or we could compute $y_i w_i$ for any fixed i. Or, perverse as it may seem, we could average all these values:

$$\mu = \frac{z^T x + y^T w}{n + m}. \tag{18.5}$$

This formula gives us exactly the value of μ whenever it is known that (x, w, y, z) lies on the central path. The key point here then is that we will use this formula to produce an estimate for μ even when the current solution (x, w, y, z) does not lie on the central path. Of course, the algorithm needs a value of μ that represents a point closer to optimality than the current solution. Hence, the algorithm takes this "par" value and reduces it by a certain fraction:

$$\mu = \delta \frac{z^T x + y^T w}{n + m},$$

where δ is a number between zero and one. In practice, one finds that setting δ to approximately 1/10 works quite well, but for the sake of discussion we will always leave it as a parameter.

4. Choosing the Step Length Parameter

The step directions, which were determined using Newton's method, were determined under the assumption that the step length parameter θ would be equal to one (i.e., $\tilde{x} = x + \Delta x$, etc.). But taking such a step might cause the new solution to violate the property that every component of all the primal and the dual variables must remain positive. Hence, we may need to use a smaller value for θ. We need to guarantee, for example, that

$$x_j + \theta \Delta x_j > 0, \qquad j = 1, 2, \ldots, n.$$

initialize $(x, w, y, z) > 0$
while (not optimal) {
$$\rho = b - Ax - w$$
$$\sigma = c - A^T y + z$$
$$\gamma = z^T x + y^T w$$
$$\mu = \delta \frac{\gamma}{n+m}$$
solve:
$$\begin{aligned} A\Delta x + \Delta w &= \rho \\ A^T \Delta y - \Delta z &= \sigma \\ Z\Delta x + X\Delta z &= \mu e - XZe \\ W\Delta y + Y\Delta w &= \mu e - YWe \end{aligned}$$
$$\theta = r \left(\max_{ij} \left\{ -\frac{\Delta x_j}{x_j}, -\frac{\Delta w_i}{w_i}, -\frac{\Delta y_i}{y_i}, -\frac{\Delta z_j}{z_j} \right\} \right)^{-1} \wedge 1$$
$$x \leftarrow x + \theta \Delta x, \qquad w \leftarrow w + \theta \Delta w$$
$$y \leftarrow y + \theta \Delta y, \qquad z \leftarrow z + \theta \Delta z$$
}

FIGURE 18.1. The path-following method.

Moving the Δx_j term to the other side and then dividing through by θ and x_j, both of which are positive, we see that θ must satisfy

$$\frac{1}{\theta} > -\frac{\Delta x_j}{x_j}, \qquad j = 1, 2, \ldots, n.$$

Of course, a similar inequality must be satisfied for the w, y, and z variables too. Putting it all together, the largest value of θ would be given by

$$\frac{1}{\theta} = \max_{ij} \left\{ -\frac{\Delta x_j}{x_j}, -\frac{\Delta w_i}{w_i}, -\frac{\Delta y_i}{y_i}, -\frac{\Delta z_j}{z_j} \right\},$$

where we have abused notation slightly by using the $\max_{ij}$ to denote the maximum of all the ratios in the indicated set. However, this choice of θ will not guarantee strict inequality, so we introduce a parameter r, which is a number close to but strictly less than one, and we set[1]

$$\theta = r \left(\max_{ij} \left\{ -\frac{\Delta x_j}{x_j}, -\frac{\Delta w_i}{w_i}, -\frac{\Delta y_i}{y_i}, -\frac{\Delta z_j}{z_j} \right\} \right)^{-1} \wedge 1. \tag{18.6}$$

This formula may look messy, and no one should actually do it by hand, but it is trivial to program a computer to do it. Such a subroutine will be really fast (requiring only on the order of $2n + 2m$ operations).

A summary of the algorithm is shown in Figure 18.1. In the next section, we investigate whether this algorithm actually converges to an optimal solution.

[1]For compactness, we use the notation $a \wedge b$ to represent the minimum of the two numbers a and b.

5. Convergence Analysis

In this section, we investigate the convergence properties of the path-following algorithm. Recall that the simplex method is a finite algorithm (assuming that steps are taken to guard against cycling). For interior-point methods, the situation is different. Every solution produced has all variables strictly positive. Yet for a solution to be optimal generally requires many variables to vanish. This vanishing can only happen "in the limit." This raises questions, the most fundamental of which are these: does the sequence of solutions produced by the path-following method converge? If so, is the limit optimal? How fast is the convergence? In particular, if we set "optimality tolerances," how many iterations will it take to achieve these tolerances? We will address these questions in this section.

In this section, we will need to measure the size of various vectors. There are many choices. For example, for each $1 \leq p < \infty$, we can define the so-called *p-norm* of a vector x as

$$\|x\|_p = \left(\sum_j |x_j|^p \right)^{\frac{1}{p}}.$$

The limit as p tends to infinity is also well defined, and it simplifies to the so-called *sup-norm*:

$$\|x\|_\infty = \max_j |x_j|.$$

5.1. Measures of Progress. Recall from duality theory that there are three criteria that must be met in order that a primal–dual solution be optimal:

(1) Primal feasibility,
(2) Dual feasibility, and
(3) Complementarity.

For each of these criteria, we introduce a measure of the extent to which they fail to be met.

For the primal feasibility criterion, we use the 1-norm of the primal infeasibility vector

$$\rho = b - Ax - w.$$

For the dual feasibility criterion, we use the 1-norm of the dual infeasibility vector

$$\sigma = c - A^T y + z.$$

For complementarity, we use

$$\gamma = z^T x + y^T w.$$

5.2. Progress in One Iteration. For the analysis in the section, we prefer to modify the algorithm slightly by having it take shorter steps than specified before. Indeed, we let

$$\theta = r\left(\max_{i,j}\left\{\left|\frac{\Delta x_j}{x_j}\right|, \left|\frac{\Delta w_i}{w_i}\right|, \left|\frac{\Delta y_i}{y_i}\right|, \left|\frac{\Delta z_j}{z_j}\right|\right\}\right)^{-1} \wedge 1$$

$$= \frac{r}{\max(\|X^{-1}\Delta x\|_\infty, \ldots, \|Z^{-1}\Delta z\|_\infty)} \wedge 1. \tag{18.7}$$

Note that the only change has been to replace the negative ratios with the absolute value of the same ratios. Since the maximum of the absolute values can be larger than the maximum of the ratios themselves, this formula produces a smaller value for θ. In this section, let x, y, etc., denote quantities from one iteration of the algorithm, and put a tilde on the same letters to denote the same quantity at the next iteration of the algorithm. Hence,

$$\begin{aligned} \tilde{x} &= x + \theta\Delta x, & \tilde{y} &= y + \theta\Delta y, \\ \tilde{w} &= w + \theta\Delta w, & \tilde{z} &= z + \theta\Delta z. \end{aligned}$$

Now let's compute some of the other quantities. We begin with the primal infeasibility:

$$\begin{aligned} \tilde{\rho} &= b - A\tilde{x} - \tilde{w} \\ &= b - Ax - w - \theta(A\Delta x + \Delta w). \end{aligned}$$

But $b - Ax - w$ equals the primal infeasibility ρ (by definition) and $A\Delta x + \Delta w$ also equals ρ, since this is precisely the first equation in the system that defines the "delta" variables. Hence,

$$\tilde{\rho} = (1-\theta)\rho. \tag{18.8}$$

Similarly,

$$\begin{aligned} \tilde{\sigma} &= c - A^T\tilde{y} + \tilde{z} \\ &= c - A^T y + z - \theta(A\Delta y - \Delta z) \\ &= (1-\theta)\sigma. \end{aligned} \tag{18.9}$$

Since θ is a number between zero and one, it follows that each iteration produces a decrease in both the primal and the dual infeasibility and that this decrease is better the closer θ is to one.

The analysis of the complementarity is a little more complicated (after all, this is the part of the system where the linearization took place):

$$\begin{aligned} \tilde{\gamma} &= \tilde{z}^T\tilde{x} + \tilde{y}^T\tilde{w} \\ &= (z + \theta\Delta z)^T(x + \theta\Delta x) + (y + \theta\Delta y)^T(w + \theta\Delta w) \\ &= z^T x + y^T w \\ &\qquad + \theta(z^T\Delta x + \Delta z^T x + y^T\Delta w + \Delta y^T w) \\ &\qquad + \theta^2(\Delta z^T\Delta x + \Delta y^T\Delta w). \end{aligned}$$

We need to analyze each of the θ terms separately. From (18.3), we see that

$$\begin{aligned} z^T\Delta x + \Delta z^T x &= e^T\left(Z\Delta x + X\Delta z\right) \\ &= e^T\left(\mu e - ZXe\right) \\ &= \mu n - z^T x. \end{aligned}$$

Similarly, from (18.4), we have

$$\begin{aligned} y^T\Delta w + \Delta y^T w &= e^T\left(Y\Delta w + W\Delta y\right) \\ &= e^T\left(\mu e - YWe\right) \\ &= \mu m - y^T w. \end{aligned}$$

Finally, (18.1) and (18.2) imply that

$$\begin{aligned} \Delta z^T\Delta x + \Delta y^T\Delta w &= \left(A^T\Delta y - \sigma\right)^T\Delta x + \Delta y^T\left(\rho - A\Delta x\right) \\ &= \Delta y^T\rho - \sigma^T\Delta x. \end{aligned}$$

Substituting these expressions into the last expression for $\tilde{\gamma}$, we get

$$\begin{aligned} \tilde{\gamma} = z^T x &+ y^T w \\ &+ \theta\left(\mu(n+m) - (z^T x + y^T w)\right) \\ &+ \theta^2\left(\Delta y^T\rho - \sigma^T\Delta x\right). \end{aligned}$$

At this point, we recognize that $z^T x + y^T w = \gamma$ and that $\mu(n+m) = \delta\gamma$. Hence,

$$\tilde{\gamma} = \left(1 - (1-\delta)\theta\right)\gamma + \theta^2\left(\Delta y^T\rho - \sigma^T\Delta x\right).$$

We must now abandon equalities and work with estimates. Our favorite tool for estimation is the following inequality:

$$\begin{aligned} |v^T w| &= |\sum_j v_j w_j| \\ &\le \sum_j |v_j||w_j| \\ &\le (\max_j |v_j|)(\sum_j |w_j|) \\ &= \|v\|_\infty \|w\|_1. \end{aligned}$$

This inequality is the trivial case of *Hölder's inequality*. From Hölder's inequality, we see that

$$|\Delta y^T\rho| \le \|\rho\|_1\|\Delta y\|_\infty \qquad \text{and} \qquad |\sigma^T\Delta x| \le \|\sigma\|_1\|\Delta x\|_\infty.$$

Hence,

$$\tilde{\gamma} \le \left(1 - (1-\delta)\theta\right)\gamma + \theta\left(\|\rho\|_1\|\theta\Delta y\|_\infty + \|\sigma\|_1\|\theta\Delta x\|_\infty\right).$$

Next, we use the specific choice of step length θ to get a bound on $\|\theta\Delta y\|_\infty$ and $\|\theta\Delta x\|_\infty$. Indeed, (18.7) implies that

$$\theta \le \frac{r}{\|X^{-1}\Delta x\|_\infty} \le \frac{x_j}{|\Delta x_j|} \qquad \text{for all } j.$$

Hence,

$$\|\theta\Delta x\|_\infty \le \|x\|_\infty.$$

Similarly,

$$\|\theta\Delta y\|_\infty \le \|y\|_\infty.$$

If we now assume that, along the sequence of points x and y visited by the algorithm, $\|x\|_\infty$ and $\|y\|_\infty$ are bounded by a large real number M, then we can estimate the new complementarity as

$$\tilde{\gamma} \le (1-(1-\delta)\theta)\,\gamma + M\|\rho\|_1 + M\|\sigma\|_1. \tag{18.10}$$

5.3. Stopping Rule. Let $\epsilon > 0$ be a small positive tolerance, and let $M < \infty$ be a large finite tolerance. If $\|x\|_\infty$ gets larger than M, then we stop and declare the problem primal unbounded. If $\|y\|_\infty$ gets larger than M, then we stop and declare the problem dual unbounded. Finally, if $\|\rho\|_1 < \epsilon$, $\|\sigma\|_1 < \epsilon$, and $\gamma < \epsilon$, then we stop and declare the current solution to be optimal (at least within this small tolerance).

Since γ is a measure of complementarity and complementarity is related to the duality gap, one would expect that a small value of γ should translate into a small duality gap. This turns out to be true. Indeed, from the definitions of γ, σ, and ρ, we can write

$$\begin{aligned}
\gamma &= z^T x + y^T w \\
&= (\sigma + A^T y - c)^T x + y^T(b - Ax - \rho) \\
&= b^T y - c^T x + \sigma^T x - \rho^T y.
\end{aligned}$$

At this point, we use Hölder's inequality to bound some of these terms to get an estimate on the duality gap:

$$\begin{aligned}
|b^T y - c^T x| &\le \gamma + |\sigma^T x| + |y^T \rho| \\
&\le \gamma + \|\sigma\|_1 \|x\|_\infty + \|\rho\|_1 \|y\|_\infty.
\end{aligned}$$

Now, if γ, $\|\sigma\|_1$, and $\|\rho\|_1$ are all small (and $\|x\|_\infty$ and $\|y\|_\infty$ are not too big), then the duality gap will be small. This estimate shows that one shouldn't expect the duality gap to get small until the primal and the dual are very nearly feasible. Actual implementations confirm this expectation.

5.4. Progress Over Several Iterations. Now let $\rho^{(k)}$, $\sigma^{(k)}$, $\gamma^{(k)}$, $\theta^{(k)}$, etc., denote the values of these quantities at the kth iteration. We have the following result about the overall performance of the algorithm:

THEOREM 18.1. *Suppose there is a real number $t > 0$, a real number $M < \infty$, and an integer K such that for all $k \le K$,*

$$\begin{aligned}
\theta^{(k)} &\ge t, \\
\|x^{(k)}\|_\infty &\le M, \\
\|y^{(k)}\|_\infty &\le M.
\end{aligned}$$

Then there exists a constant $\bar{M} < \infty$ such that

$$\begin{aligned}\|\rho^{(k)}\|_1 &\leq (1-t)^k \|\rho^{(0)}\|_1, \\ \|\sigma^{(k)}\|_1 &\leq (1-t)^k \|\sigma^{(0)}\|_1, \\ \gamma^{(k)} &\leq (1-\tilde{t})^k \bar{M},\end{aligned}$$

for all $k \leq K$ where

$$\tilde{t} = t(1-\delta).$$

PROOF. From (18.8) and the bound on $\theta^{(k)}$, it follows that

$$\|\rho^{(k)}\|_1 \leq (1-t)\|\rho^{(k-1)}\|_1 \leq \cdots \leq (1-t)^k \|\rho^{(0)}\|_1.$$

Similarly, from (18.9), it follows that

$$\|\sigma^{(k)}\|_1 \leq (1-t)\|\sigma^{(k-1)}\|_1 \leq \cdots \leq (1-t)^k \|\sigma^{(0)}\|_1.$$

As usual, $\gamma^{(k)}$ is harder to estimate. From (18.10) and the previous two estimates, we see that

$$\begin{aligned}\gamma^{(k)} &\leq (1-t(1-\delta))\,\gamma^{(k-1)} \\ &\qquad + M(1-t)^{k-1}\left(\|\rho^{(0)}\|_1 + \|\sigma^{(0)}\|_1\right) \\ &= (1-\tilde{t})\gamma^{(k-1)} + \tilde{M}(1-t)^{k-1}, \qquad (18.11)\end{aligned}$$

where $\tilde{M} = M\left(\|\rho^{(0)}\|_1 + \|\sigma^{(0)}\|_1\right)$. Since an analogous inequality relates $\gamma^{(k-1)}$ to $\gamma^{(k-2)}$, we can substitute this analogous inequality into (18.11) to get

$$\begin{aligned}\gamma^{(k)} &\leq (1-\tilde{t})\left[(1-\tilde{t})\gamma^{(k-2)} + \tilde{M}(1-t)^{k-2}\right] + \tilde{M}(1-t)^{k-1} \\ &= (1-\tilde{t})^2\gamma^{(k-2)} + \tilde{M}(1-t)^{k-1}\left[\frac{1-\tilde{t}}{1-t} + 1\right].\end{aligned}$$

Continuing in this manner, we see that

$$\begin{aligned}\gamma^{(k)} &\leq (1-\tilde{t})^2\left[(1-\tilde{t})\gamma^{(k-3)} + \tilde{M}(1-t)^{k-3}\right] \\ &\qquad + \tilde{M}(1-t)^{k-1}\left[\frac{1-\tilde{t}}{1-t} + 1\right] \\ &= (1-\tilde{t})^3\gamma^{(k-3)} + \tilde{M}(1-t)^{k-1}\left[\left(\frac{1-\tilde{t}}{1-t}\right)^2 + \frac{1-\tilde{t}}{1-t} + 1\right] \\ &\leq \cdots \leq \\ &\leq (1-\tilde{t})^k\gamma^{(0)} + \tilde{M}(1-t)^{k-1}\left[\left(\frac{1-\tilde{t}}{1-t}\right)^{k-1} + \cdots + \frac{1-\tilde{t}}{1-t} + 1\right].\end{aligned}$$

Now we sum the bracketed partial sum of a geometric series to get

$$(1-t)^{k-1}\left[\left(\frac{1-\tilde{t}}{1-t}\right)^{k-1}+\cdots+\frac{1-\tilde{t}}{1-t}+1\right]=(1-t)^{k-1}\frac{1-\left(\frac{1-\tilde{t}}{1-t}\right)^{k}}{1-\frac{1-\tilde{t}}{1-t}}$$
$$=\frac{(1-\tilde{t})^k-(1-t)^k}{t-\tilde{t}}.$$

Recalling that $\tilde{t}=t(1-\delta)$ and dropping the second term in the numerator, we get

$$\frac{(1-\tilde{t})^k-(1-t)^k}{t-\tilde{t}}\leq\frac{(1-\tilde{t})^k}{\delta t}.$$

Putting this all together, we see that

$$\gamma^{(k)}\leq(1-\tilde{t})^k\left(\gamma^{(0)}+\frac{\tilde{M}}{\delta t}\right).$$

Denoting the parenthesized expression by $\bar{M}$ completes the proof. □

Theorem 18.1 is only a partial convergence result because it depends on the assumption that the step lengths remain bounded away from zero. To show that the step lengths do indeed have this property requires that the algorithm be modified and that the starting point be carefully selected. The details are rather technical and hence omitted (see the Notes at the end of the chapter for references).

Also, before we leave this topic, note that the primal and dual infeasibilities go down by a factor of $1-t$ at each iteration, whereas the duality gap goes down by a smaller amount $1-\tilde{t}$. The fact that the duality gap converges more slowly that the infeasibilities is also readily observed in practice.

Exercises

18.1 Starting from $(x,w,y,z)=(e,e,e,e)$, and using $\delta=1/10$, and $r=9/10$, compute (x,w,y,z) after one step of the path-following method for the problem given in
(a) Exercise 2.3.
(b) Exercise 2.4.
(c) Exercise 2.5.
(d) Exercise 2.10.

18.2 Let $\{(x_\mu,w_\mu,y_\mu,z_\mu):\mu\geq 0\}$ denote the central trajectory. Show that

$$\lim_{\mu\to\infty} b^Ty_\mu-c^Tx_\mu=\infty.$$

Hint: look at (18.5).

18.3 Consider a linear programming problem whose feasible region is bounded and has nonempty interior. Use the result of Exercise 18.2 to show that the dual problem's feasible set is unbounded.

18.4 *Scale invariance*. Consider a linear program and its dual:

$$(P) \quad \begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax + w = b \\ & x, w \geq 0 \end{array} \qquad (D) \quad \begin{array}{ll} \min & b^T y \\ \text{s.t.} & A^T y - z = c \\ & y, z \geq 0. \end{array}$$

Let R and S be two given diagonal matrices having positive entries along their diagonals. Consider the *scaled* reformulation of the original problem and its dual:

$$(\bar{P}) \quad \begin{array}{ll} \max & (Sc)^T \bar{x} \\ \text{s.t.} & RAS\bar{x} + \bar{w} = Rb \\ & \bar{x}, \bar{w} \geq 0 \end{array} \qquad (\bar{D}) \quad \begin{array}{ll} \min & (Rb)^T \bar{y} \\ \text{s.t.} & SA^T R\bar{y} - \bar{z} = Sc \\ & \bar{y}, \bar{z} \geq 0. \end{array}$$

Let (x^k, w^k, y^k, z^k) denote the sequence of solutions generated by the primal–dual interior-point method applied to (P)–(D). Similarly, let $(\bar{x}^k, \bar{w}^k, \bar{y}^k, \bar{z}^k)$ denote the sequence of solutions generated by the primal–dual interior-point method applied to $(\bar{P})$–$(\bar{D})$. Suppose that we have the following relations among the starting points:

$$\bar{x}^0 = S^{-1} x^0, \quad \bar{w}^0 = R w^0, \quad \bar{y}^0 = R^{-1} y^0, \quad \bar{z}^0 = S z^0.$$

Show that these relations then persist. That is, for each $k \geq 1$,

$$\bar{x}^k = S^{-1} x^k, \quad \bar{w}^k = R w^k, \quad \bar{y}^k = R^{-1} y^k, \quad \bar{z}^k = S z^k.$$

18.5 *Homotopy method*. Let $\bar{x}$, $\bar{y}$, $\bar{z}$, and $\bar{w}$ be given componentwise positive "initial" values for x, y, z, and w, respectively. Let t be a parameter between 0 and 1. Consider the following nonlinear system:

$$\begin{aligned} Ax + w &= tb + (1-t)(A\bar{x} + \bar{w}) \\ A^T y - z &= tc + (1-t)(A^T \bar{y} - \bar{z}) \\ XZe &= (1-t)\bar{X}\bar{Z}e \\ YWe &= (1-t)\bar{Y}\bar{W}e \\ x, y, z, w &> 0. \end{aligned} \tag{18.12}$$

(a) Use Exercise 17.3 to show that this nonlinear system has a unique solution for each $0 \leq t < 1$. Denote it by $(x(t), y(t), z(t), w(t))$.

(b) Show that $(x(0), y(0), z(0), w(0)) = (\bar{x}, \bar{y}, \bar{z}, \bar{w})$.

(c) Assuming that the limit

$$(x(1), y(1), z(1), w(1)) = \lim_{t \to 1} (x(t), y(t), z(t), w(t))$$

exists, show that it solves the standard-form linear programming problem.

(d) The family of solutions $(x(t), y(t), z(t), w(t))$, $0 \leq t < 1$, describes a curve in "primal–dual" space. Show that the tangent to this curve at $t = 0$ coincides with the path-following step direction at $(\bar{x}, \bar{y}, \bar{z}, \bar{w})$ computed with $\mu = 0$; that is,

$$\left(\frac{dx}{dt}(0), \frac{dy}{dt}(0), \frac{dz}{dt}(0), \frac{dw}{dt}(0)\right) = (\Delta x, \Delta y, \Delta z, \Delta w),$$

where $(\Delta x, \Delta y, \Delta z, \Delta w)$ is the solution to (18.1)–(18.4).

18.6 *Higher-order methods.* The previous exercise shows that the path-following step direction can be thought of as the direction one gets by approximating a homotopy path with its tangent line:

$$x(t) \approx x(0) + \frac{dx}{dt}(0)t.$$

By using more terms of the Taylor's series expansion, one can get a better approximation:

$$x(t) \approx x(0) + \frac{dx}{dt}(0)t + \frac{1}{2}\frac{d^2x}{dt^2}(0)t^2 + \cdots + \frac{1}{k!}\frac{d^kx}{dt^k}(0)t^k.$$

(a) Differentiating the equations in (18.12) twice, derive a linear system for $(d^2x/dt^2(0), d^2y/dt^2(0), d^2z/dt^2(0), d^2w/dt^2(0))$.

(b) Can the same technique be applied to derive linear systems for the higher-order derivatives?

18.7 *Linear Complementarity Problem.* Given a $k \times k$ matrix M and a k-vector q, a vector x is said to solve the linear complementarity problem if

$$\begin{aligned} -Mx + z &= q \\ XZe &= 0 \\ x, z &\geq 0 \end{aligned}$$

(note that the first equation can be taken as the definition of z).

(a) Show that the optimality conditions for linear programming can be expressed as a linear complementarity problem with

$$M = \begin{bmatrix} 0 & -A \\ A^T & 0 \end{bmatrix}.$$

(b) The path-following method introduced in this chapter can be extended to cover linear complementarity problems. The main step in the derivation is to replace the complementarity condition $XZe = 0$ with a μ-complementarity condition $XZe = \mu e$ and then to use Newton's method to derive step directions Δx and Δz. Carry out this procedure and indicate the system of equations that define Δx and Δz.

(c) Give conditions under which the system derived above is guaranteed to have a unique solution.

(d) Write down the steps of the path-following method for the linear complementarity problem.

(e) Study the convergence of this algorithm by adapting the analysis given in Section 18.5.

18.8 Consider again the L^1-regression problem:

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

Complete the following steps to derive the step direction vector Δx associated with the primal–dual affine-scaling method for solving this problem.

(a) Show that the L^1-regression problem is equivalent to the following linear programming problem:

$$\begin{array}{lll} & \text{minimize} & e^T(t_+ + t_-) \\ (18.13) & \text{subject to} & Ax + t_+ - t_- = b \\ & & \qquad t_+, t_- \geq 0. \end{array}$$

(b) Write down the dual of (18.13).

(c) Add slack and/or surplus variables as necessary to reformulate the dual so that all inequalities are simple nonnegativities of variables.

(d) Identify all primal–dual pairs of complementary variables.

(e) Write down the nonlinear system of equations consisting of: (1) the primal equality constraints, (2) the dual equality constraints, (3) all complementarity conditions (using $\mu = 0$ since we are looking for an affine-scaling algorithm).

(f) Apply Newton's method to the nonlinear system to obtain a linear system for step directions for all of the primal and dual variables.

(g) We may assume without loss of generality that both the initial primal solution and the initial dual solution are feasible. Explain why.

(h) The linear system derived above is a 6×6 block matrix system. But it is easy to solve most of it by hand. First eliminate those step directions associated with the nonnegative variables to arrive at a 2×2 block matrix system.

(i) Next, solve the 2×2 system. Give an explicit formula for Δx.

(j) How does this primal–dual affine-scaling algorithm compare with the iteratively reweighted least squares algorithm defined in Section 12.5?

18.9 (a) Let ξ_j, $j = 1, 2, \ldots,$ denote a sequence of real numbers between zero and one. Show that $\prod_j (1 - \xi_j) = 0$ if $\sum_j \xi_j = \infty$.

(b) Use the result of part a to prove the following convergence result: if the sequences $\|x^{(k)}\|_\infty$, $k = 1, 2, \ldots,$ and $\|y^{(k)}\|_\infty$, $k = 1, 2, \ldots,$ are bounded and $\sum_k \theta^{(k)} = \infty$, then

$$\lim_{k \to \infty} \|\rho^{(k)}\|_1 = 0$$

$$\lim_{k \to \infty} \|\sigma^{(k)}\|_1 = 0$$

$$\lim_{k \to \infty} \gamma^{(k)} = 0.$$

Notes

The path-following algorithm introduced in this chapter has its origins in a paper by Kojima et al. (1989). Their paper assumed an initial feasible solution and therefore was a true interior-point method. The method given in this chapter does not assume the initial solution is feasible—it is a one-phase algorithm. The simple yet beautiful idea of modifying the Kojima–Mizuno–Yoshise primal–dual algorithm to make it into a one-phase algorithm is due to Lustig (1990).

Of the thousands of papers on interior-point methods that have appeared in the last decade, the majority have included convergence proofs for some version of an interior-point method. Here, we only mention a few of the important papers. The first polynomial-time algorithm for linear programming was discovered by Khachian (1979). Khachian's algorithm is fundamentally different from any algorithm presented in this book. Paradoxically, it proved in practice to be inferior to the simplex method. N.K. Karmarkar's pathbreaking paper (Karmarkar 1984) contained a detailed convergence analysis. His claims, based on preliminary testing, that his algorithm is uniformly substantially faster than the simplex method sparked a revolution in linear programming. Unfortunately, his claims proved to be exaggerated, but nonetheless interior-point methods have been shown to be competitive with the simplex method and usually superior on very large problems. The convergence proof for a primal–dual interior-point method was given by Kojima et al. (1989). Shortly thereafter, Monteiro and Adler (1989) improved on the convergence analysis. Two recent survey papers, Todd (1995) and Anstreicher (1996), give nice overviews of the current state of the art. Also, a soon-to-be-published book by Wright (1996) should prove to be a valuable reference to the reader wishing more information on convergence properties of these algorithms.

The homotopy method outlined in Exercise 18.5 is described in Nazareth (1986, 1996). Higher-order path-following methods are described (differently) in Carpenter et al. (1993).