Primal–Dual Interior Point Algorithm (Path–Following Algorithm) for Linear Programming

This Hypercard stack was prepared by:
Dennis L. Bricker,
Dept. of Industrial Engineering,
University of Iowa,
Iowa City, Iowa 52242
e-mail: dbricker@icaen.uiowa.edu

author

Consider the primal/dual pair of LPs:

Primal

Minimize $c^t x$
subject to $Ax = b$
$x \geq 0$

Dual

Maximize $y b$
subject to $yA \leq c^t$

i.e.,

Maximize $b^t y$
subject to $A^t y \leq c$

Convert dual constraints to equalities:

Primal

Minimize $c^t x$
subject to $Ax = b$
$x \geq 0$

Dual

Maximize $b^t y$
subject to $A^t y + z = c^t$
$z \geq 0$

Use barrier functions to relax the non-negativity conditions:

Primal

Minimize $c x - \mu \sum_{j=1}^{n} \ln(x_j)$
subject to
$A x = b$

as $x \longrightarrow 0$, $-\mu \ln(x) \longrightarrow \infty$

Dual

Maximize $b^t y + \mu \sum_{j=1}^{n} \ln(z_j)$
subject to
$A^t y + z = c^t$

Use Lagrange multipliers to relax the equality constraints:

Lagrangian Functions

$$L_P(x, y) = c^t x - \mu \sum_{j=1}^{n} \ln(x_j) + y^t(A x - b)$$

$$L_D(x,y,z) = b^t y + \mu \sum_{j=1}^{n} \ln(x_j) - x^t (A^t y + z - c)$$

The optimality conditions may be written

$$\frac{\partial L_P(x,y)}{\partial x} = 0, \quad \frac{\partial L_P(x,y)}{\partial y} = 0$$

and

$$\frac{\partial L_D(x,y,z)}{\partial x} = 0, \quad \frac{\partial L_D(x,y,z)}{\partial y} = 0, \quad \frac{\partial L_D(x,y,z)}{\partial z} = 0$$

These reduce to the following
optimality conditions

$A x = b$
$A^t y + z = c$
} *linear equations*

$x_j z_j = \mu$, $j=1,2, \ldots n$ ← *nonlinear equations*

To solve the nonlinear system of equations, we might use the ***Newton–Raphson*** method:

Given an initial approximate solution (x^0, y^0, z^0): an improved approximate solution is given by

$$\begin{cases} x^1 = x^0 + \delta_x \\ y^1 = y^0 + \delta_y \\ z^1 = z^0 + \delta_z \end{cases}$$

where δ_x, δ_y, and δ_z are found by solving a linear system.

Notation

$$\mathbf{X} = \text{diag}\{x_1, x_2, \ldots x_n\}$$

$$\mathbf{Z} = \text{diag}\{z_1, z_2, \ldots z_n\}$$

$$e = [1, 1, \ldots. 1]$$

Then the constraints

$$x_j z_j = \mu, \; j=1,2, \ldots n$$

may be written

$$\mathbf{X}\mathbf{Z}e = \mu e$$

We wish to solve the *nonlinear* system

$$\begin{cases} Ax - b = 0 \\ A^t y + z - c = 0 \\ \mathbf{X}\mathbf{Z}e - \mu e = 0 \end{cases}$$

Newton-Raphson Method: given (x^o, y^o, z^o), solve the *linear* system

$$\begin{cases} A\,\delta_x = -\left[Ax^0 - b\right] \\ A^t\,\delta_y + \delta_z = -\left[A^t y^0 + z^0 - c\right] \\ Z\,\delta_x + \mathbf{X}\,\delta_z = -\left[\mathbf{X}\mathbf{Z}e - \mu e\right] \end{cases}$$

That is, solve

Jacobian matrix →

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^t & \mathbf{I} \\ Z & 0 & \mathbf{X} \end{bmatrix} \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix} = \begin{bmatrix} d_P \\ -d_D \\ \mu e - \mathbf{X}\mathbf{Z}e \end{bmatrix}$$

where $d_P = b - Ax^0$ ← *primal infeasibility*

$d_D = A^t y^0 + z^0 - c$ ← *dual infeasibility*

and then compute the improved approximation

$$\begin{cases} x^1 = x^0 + \delta_x \\ y^1 = y^0 + \delta_y \\ z^1 = z^0 + \delta_z \end{cases}$$

Solving the linear system:

$$\delta_x = Z^{-1}\left[\mu e - \mathbf{X}Ze - \mathbf{X}\,\delta_z\right]$$

$$\delta_z = -d_D - A^t\,\delta_y$$

$$\Longrightarrow \left[A Z^{-1}\mathbf{X} A^t\right]\delta_y = b - \mu A Z^{-1}e - AZ^{-1}\mathbf{X}\,d_D$$

or $$\delta_y = \left[A Z^{-1}\mathbf{X} A^t\right]^{-1}\left(b - \mu A Z^{-1}e - AZ^{-1}\mathbf{X}d_D\right)$$

Computing

$$\delta_y = \left[A Z^{-1}\mathbf{X} A^t\right]^{-1}\left(b - \mu A Z^{-1}e - AZ^{-1}\mathbf{X}d_D\right)$$

by using matrix inversion is computationally costly for large problems...
other methods for solving the linear system for δ_y are preferred.

After computing the step $(\delta_x, \delta_y, \delta_z)$,

$$\begin{cases} x^1 = x^0 + \delta_x \\ y^1 = y^0 + \delta_y \\ z^1 = z^0 + \delta_z \end{cases}$$

An alternative would be to go (almost) as far as possible in the x direction and the (y,z) direction:

$$\begin{cases} x^1 = x^0 + \alpha_P\,\delta_x \\ y^1 = y^0 + \alpha_D\,\delta_y \\ z^1 = z^0 + \alpha_D\,\delta_z \end{cases}$$

for stepsizes α_P and α_D, respectively.

$$\alpha_P = \tau \min_j \left\{\frac{-x_j^0}{\delta_{xj}} : \delta_{xj} < 0\right\}$$

$$\alpha_D = \tau \min_j \left\{\frac{-z_j^0}{\delta_{zj}} : \delta_{zj} < 0\right\}$$

for $0 < \tau < 1$ *e.g., $\tau = 0{,}995$*
($\tau = 1$ will result in one of the x and z variables reaching zero!)

Generally, only one Newton-Raphson step is used, so that the nonlinear system is only approximately solved.

This completes one iteration. As $\mu \longrightarrow 0$, the values of x,y, and z will converge to the optimal primal and dual solutions.
The path followed by (x,y,z) is referred to as the ***central path*** and the algorithm as a ***path-following*** algorithm.

Reduction of μ:

$$\mu = \frac{c^t x^1 - b^t y^1}{\theta(n)}$$

suggested value of parameter θ:

$$\theta(n) = \begin{cases} n^2 & \text{if } n \leq 5{,}000 \\ n\sqrt{n} & \text{if } n > 5{,}000 \end{cases}$$

Termination criterion:

$$\frac{c^t x^k - b^t y^k}{1 + |b^t y^k|} < \varepsilon$$

The number of iterations required is rather insensitive to the size n of the problem, and is usually between 20 and 80 for most problems.