Linear Programming

Chapter 6.14-7.3

Björn Morén

LINKÖPING UNIVERSITY

LINKÖPING UNIVERSITY

LINKÖPING UNIVERSITY

Problem Formulation

$$
\begin{aligned}
\text{Minimize} \quad & z(x) = cx \\
\text{subject to} \quad & Ax = b \\
& 0 \le x \le u,
\end{aligned}
$$

1. Upper bounds as constraints
2. Consider upper bounds implicitly

Basic Solution

Divide variables if they are basic, at lower bound, or at upper bound (x_B, x_L, x_U)

Dual Problem

$$\begin{aligned}\text{Minimize} \quad & z(x) = cx \\ \text{subject to} \quad & Ax = b \\ & 0 \leq x \leq u,\end{aligned}$$

$$\begin{aligned}\text{Maximize} \quad & v = \pi b + \sum_{j \in J} \mu_j u_j \\ \text{subject to} \quad & \pi A_{.j} - \mu_j \leq c_j, \quad \text{for } j \in J \\ & \pi A_{.j} \leq c_j \quad \text{for } j \in \bar{J} \\ & \pi \text{ unrestricted}, \quad \mu_j \geq 0, \quad \text{for } j \in J,\end{aligned}$$

Optimality Conditions and Dual Feasibility

Reduced costs are defined as $\bar{c}_j = \pi A_{,j}$

Optimality Conditions and Dual Feasibility

Reduced costs are defined as $\bar{c}_j = \pi A_{,j}$

$$\bar{c}_j \begin{cases} \geq 0 & \text{for all } j \text{ s. th. } x_j \in x_L, \\ \leq 0 & \text{for all } j \text{ s. th. } x_j \in x_U. \end{cases}$$

Updated Simplex Method

1. Check optimality conditions
2. Select entering variable
3. Minimum ratio test (and select exiting variable)
4. Do pivot step

Updated Simplex Method

1. **Check optimality conditions**
2. Select entering variable
3. **Minimum ratio test (and select exiting variable)**
4. Do pivot step

Optimality Conditions

$$\begin{aligned} \text{Minimize} \quad & z(x) = cx \\ \text{subject to} \quad & Ax = b \\ & 0 \le x \le u, \end{aligned}$$

Solution is optimal if

1. For all x_j at lower bound, $x_j = l_j$ reduced cost $\bar{c_j} \geq 0$
2. For all x_j at upper bound, $x_j = u_j$ reduced cost $\bar{c_j} \leq 0$

Minimum Ratio

If the entering variable x_s is at lower bound:

1. $x_s = \theta$
2. $x_B = \bar{g} - \theta \bar{A}_{,s}$

$$\theta = \min \left[u_s, \left\{ \frac{u_i - \bar{g}_i}{-\bar{a}_{is}} : i \ \text{ s. th. } \ \bar{a}_{is} < 0 \right\}, \left\{ \frac{\bar{g}_i}{\bar{a}_{is}} : i \ \text{ s. th. } \ \bar{a}_{is} > 0 \right\} \right],$$

Minimum Ratio

If the entering variable x_s is at upper bound:

1. $x_s = -\theta$
2. $x_B = \bar{g} + \theta \bar{A}_{,s}$

$$\theta = \min\left[u_s, \left\{\frac{u_i - \bar{g}_i}{\bar{a}_{is}} : i \text{ s. th. } \bar{a}_{is} > 0\right\}, \left\{\frac{\bar{g}_i}{-\bar{a}_{is}} : i \text{ s. th. } \bar{a}_{is} < 0\right\}\right],$$

Phase I

To find a feasible solution, a standard Phase I method can be used.

LINKÖPING UNIVERSITY

History

Dikin 1967 was first.
Khachiyan 1979, ellipsoid algorithm, solves Linear Programs in polynomial time.
Work by Karmakar 1984 made it popular. Also solves Linear Programs in polynomial time, more effective than the ellipsoid algorithm.

Interior Point

$$Dx \geq d$$

Interior Point

$$Dx \geq d$$

x interior point if

$$Dx > d$$

Relative Interior Point

$$
\begin{aligned}
Ax &= b \\
Dx &\geq d
\end{aligned}
$$

Relative Interior Point

$$
\begin{aligned}
Ax &= b \\
Dx &\geq d
\end{aligned}
$$

x **relative** interior point if

$$
\begin{aligned}
Ax &= b \\
Dx &> d
\end{aligned}
$$

Phase I - examples

$$\begin{aligned} \text{Minimize} \quad & cx \\ \text{subject to} \quad & Ax \geq b \end{aligned}$$

Phase I - examples

$$\begin{aligned} \text{Minimize} \quad & cx \\ \text{subject to} \quad & Ax \geq b \end{aligned}$$

Find a solution from the following Phase I problem

$$\begin{aligned} \text{Minimize} \quad & cx + Mx_{n+1} \\ \text{subject to} \quad & Ax + ex_{n+1} \geq b \\ & x_{n+1} \geq 0 \end{aligned}$$

Phase I - examples

$$
\begin{aligned}
\text{Minimize} \quad & cx \\
\text{subject to} \quad & Ax = b \\
& x \geq 0
\end{aligned}
$$

Phase I - examples

$$
\begin{aligned}
\text{Minimize} \quad & cx \\
\text{subject to} \quad & Ax = b \\
& x \geq 0
\end{aligned}
$$

Find a solution from the following Phase I problem

$$
\begin{aligned}
\text{Minimize} \quad & cx + Mx_{n+1} \\
\text{subject to} \quad & Ax + A_{.n+1}x_{n+1} = b \\
& x,\ x_{n+1} \geq 0.
\end{aligned}
$$

Phase I - examples

$$
\begin{aligned}
\text{Minimize} \quad & cx \\
\text{subject to} \quad & Ax = b \\
& Dx \geq d
\end{aligned}
$$

Phase I - examples

$$\begin{aligned} \text{Minimize} \quad & cx \\ \text{subject to} \quad & Ax = b \\ & Dx \geq d \end{aligned}$$

Find a solution from the following Phase I problem

$$\begin{aligned} \text{Minimize} \quad & cx \\ \text{subject to} \quad & Ax = b \\ & Dx + ex_{n+1} \geq d \end{aligned}$$

Rounding Procedure

$$
\begin{aligned}
\text{Minimize} \quad & cx \\
\text{subject to} \quad & Ax = b \\
& x \geq 0
\end{aligned}
$$

Theorem 1 (7.1). *if x is sufficiently close to the optimal objective function value. An optimal basic feasible solution can be found by using the purification routine.*

General Algorithm

1. Find search direction
   1. Solving approximation problems e.g affine scaling method, Karmarkars projective scaling method
   2. Solving a problem with optimality conditions, nonlinear equations, e.g primal-dual path following IPM

General Algorithm

1. Find search direction
   1. Solving approximation problems e.g affine scaling method, Karmarkars projective scaling method
   2. Solving a problem with optimality conditions, nonlinear equations, e.g primal-dual path following IPM
2. Determine step length

Observations

- Number of iterations grow very slowly with problem size, almost constant
- Time per iteration increases with problem size
- Extra steps required to get a BFS (purification routine)
- Useful on large scale problems
- Simplex method is good to start with for understanding

Question?

www.liu.se

LINKÖPING UNIVERSITY