| | |
|---|---|
| **Mathematical Programming** | **Lecture 27** |
| **OR 630 Fall 2005** | **December 1, 2005** |
| **Notes by An-swol Hu** | |

# 1 Path-Following Methods

Recall that, as long as $\mathcal{F}^0(P)$ and $\mathcal{F}^0(D)$ are non-empty, for each $\mu > 0$ there is a (unique) solution $(x(\mu), y(\mu), s(\mu))$ to

$$\begin{aligned} A^T y + s &= c \\ Ax &= b \\ XSe &= \mu e \end{aligned} \tag{1}$$

for $x > 0$ and $s > 0$, defining the central path. Suppose we have an approximate solution $(x, y, s)$ to this system with $x \in \mathcal{F}^0(P)$, $y \in \mathcal{F}^0(D)$ and $\frac{1}{\mu}XSe \approx e$, where $\mu = \frac{x^T s}{n}$. We want to find $(x+\Delta x, y+\Delta y, s+\Delta s)$ to approximately satisfy (1) with $\mu$ replaced by $\sigma\mu$, $0 \le \sigma \le 1$. ($\sigma = 0$ means we are trying to find the optimal solution, $\sigma = 1$ means we are trying to get a better approximation to $(x(\mu), y(\mu), s(\mu))$, so usually $0 < \sigma < 1$). Then, $(\Delta x, \Delta y, \Delta s)$ satisfy

$$\begin{aligned} A^T \Delta y + \Delta s &= 0 \\ A\Delta x &= 0 \\ S\Delta x + X\Delta s &= \sigma\mu e - XSe \end{aligned}$$

where in the last equation we have dropped the second order term $\Delta X \Delta S e$ to get a linear system.

What we need is to solve this $(2n+m) \times (2n+m)$ structured linear system. We proceed as follows:

1. Express $\Delta s$ in terms of $\Delta y$: $\Delta s = -A^T \Delta y$.

2. Express $\Delta x$ in terms of $\Delta s$, and hence $\Delta y$:

$$\begin{aligned} \Delta x &= \sigma\mu s^{-1} - x - XS^{-1}\Delta s \\ &= \sigma\mu s^{-1} - x + XS^{-1}A^T\Delta y, \end{aligned}$$

   where $s^{-1}$ is the vector composed of $1/s_i$, $i = 1, \ldots, n$.

3. Substitute this into the second set of equations to get: $(AXS^{-1}A^T)\Delta y = b - \sigma\mu A s^{-1}$, where we have used $Ax = b$. $AXS^{-1}A^T$ is a symmetric $m \times m$ positive definite matrix that is possibly sparse. Note also that this is primal-dual path following since we have $XS^{-1}$ in the matrix.

Solve the (Schur complement) equation for $\Delta y$, then $\Delta s$ and $\Delta x$. Then, set $(x_+, y_+, s_+) = (x, y, s) + \alpha(\Delta x, \Delta y, \Delta s)$ for some $\alpha > 0$ and continue. Note: the system in 3. is very much like that used to compute the affine-scaling direction $\bar{d}$: $(AX^2A^T)y = AX^2c$, then $\bar{d} = -X^2c - XA^Ty$.

We need a strategy for choosing $\sigma$ and $\alpha$ at each iteration. These are often based on staying in some neighborhood of the central path:

- $||\frac{1}{\mu}XSe - e||_2 \leq \beta$, the $L_2$-neighborhood;
- $||\frac{1}{\mu}XSe - e||_\infty \leq \beta$, the $L_\infty$-neighborhood;
- $\frac{1}{\mu}XSe - e \geq -(1-\gamma)e \Leftrightarrow XSe \geq \gamma\mu e$, the $L_{-\infty}$-neighborhood;

for some $0 < \beta < 1$ or $0 < \gamma < 1$.

Common strategies for doing this are:

- Let $\sigma = 1 - \frac{\theta}{\sqrt{n}}$ for some fixed $0 < \theta < 1$ and let $\alpha = 1$. Then, if $||\frac{1}{\mu}XSe - e||_2 \leq \beta$, $||\frac{1}{\mu_+}X_+S_+e - e||_2 \leq \beta$. Thus, we stay in a small $L_2$-neighborhood of the central path. This gives an "$O(\sqrt{n}\ln\frac{1}{\epsilon})$" iteration algorithm.
- Choose $0 < \sigma < 1$ independent of $n$ and let $\alpha$ be the largest value in $[0, 1]$ such that $X(\alpha)S(\alpha) \geq \gamma\mu(\alpha)e$ for all $0 \leq \alpha \leq \bar{\alpha}$. It can be shown that $\bar{\alpha} = \Omega(\frac{1}{n})$ and we get an "$O(n\ln\frac{1}{\epsilon})$" iteration algorithm. But, in practice this technique usually works better than the previous one.
- Suppose $||\frac{1}{\mu}XSe - e||_2 \leq \frac{1}{4}$, $\sigma = 0$. Choose the largest $\bar{\alpha}$ such that $||\frac{1}{\mu(\alpha)}X(\alpha)S(\alpha)e - e||_2 \leq \frac{1}{2}$ for $0 \leq \alpha \leq \bar{\alpha}$, and let the result be $(\hat{x}, \hat{y}, \hat{s})$. Now take a second step from this point using $\sigma = 1$ and $\alpha = 1$ to get $(x_+, y_+, s_+)$ with $||\frac{1}{\mu_+}X_+S_+e - e||_2 \leq \frac{1}{4}$. This gives an "$O(\sqrt{n}\ln\frac{1}{\epsilon})$" iteration algorithm, but it also has quadratic convergence.

# 2 Initialization

1. Use artificial variables and constraints: the primal problem $(\hat{P})$

$$\begin{array}{rcl} \min \quad c^Tx + M_1x_{n+1} & & \\ Ax + (b - Ae)x_{n+1} & = & b \\ (e-c)^Tx + x_{n+2} & = & M_2 \\ \hat{x} = (x; x_{n+1}; x_{n+2}) & \geq & 0 \end{array}$$

and the dual problem $(\hat{D})$

$$\begin{array}{rcl} \max \quad b^Ty + M_2y_{m+1} & & \\ A^Ty + (e-c)y_{m+1} + s & = & c \\ (b - Ae)^Ty + s_{n+1} & = & M_1 \\ y_{m+1} + s_{n+2} & = & 0 \\ \hat{s} = (s; s_{n+1}; s_{n+2}) & \geq & 0. \end{array}$$

$(\hat{P})$ and $(\hat{D})$ have strictly feasible solutions if $M_1, M_2$ are large enough:

$$\begin{aligned} \hat{x} &= (e; 1; M_2 - (e-c)^T e) > 0, \\ \hat{y} &= (0; -1), \\ \hat{s} &= (e; M_1; 1) > 0. \end{aligned}$$

2. Infeasible-interior-point methods: The "infeasible" means that $Ax \neq b$ and $A^T y + s \neq c$ are possible and the "interior" means that $x, s > 0$. We can start with, say, $x = s = e$ and $y = 0$. Then, when we seek $(\Delta x, \Delta y, \Delta s)$, just compensate for the infeasible $x, (y, s)$:

$$\begin{aligned} A^T \Delta y + \Delta s &= c - A^T y - s \\ A \Delta x &= b - Ax \\ S \Delta x + X \Delta s &= \sigma \mu e - XSe. \end{aligned}$$

   Proceed as before. This works well in practice, but the theory is much more complicated.

# 3 Extensions of Interior-Point Methods (IPM's)

IPM's have been extended to many non-linear, but convex, programming problems, e.g. SDP (semi-definite programming).

$$\begin{aligned} \min \quad & C \cdot X \\ A_i \cdot X &= b_i \qquad i = 1, \ldots, m \\ X &\succeq 0 \qquad \text{(symmetric, positive semi-definite)} \end{aligned}$$

where $U \cdot V = \operatorname{tr}(U^T V) = \sum_i \sum_j u_{ij} v_{ij}$. The dual is

$$\begin{aligned} \max \quad & b^T y \\ \sum_i y_i A_i + S &= C \\ S &\succeq 0 \end{aligned}$$

This problem has a logarithmic barrier function $-\ln(\det(X))$ defined on symmetric, positive definite matrices. The central path is defined by dual-primal feasibility and $XS = \mu I$, but this last equation is not easy to linearize satisfactorily.

# 4 Summary and Overview

- Linear Programs (LP): An important class of optimization problems arising in a wide variety of resource allocation, production planning, and data-fitting applications.
- Powerful Duality Theory: Short certificate of optimality, sensitivity analysis, certificate of near optimality.

- Geometry: Polyhedral (extreme points, extreme directions), nice barrier function.
- Algorithms:
  - Simplex Method: practically efficient, theoretically bad, gives optimal basis, useful for sensitivity analysis and for re-optimization.
  - Ellipsoid Method: practically inefficient, theoretically good, nice implications in combinatorial optimization.
  - Interior-Point Methods: practically efficient and theoretically good, give approximate dual solution but not a basis, not good for re-optimization.
- Future Courses:
  - OR631: integer programming (Trotter), complexity of convex programming (Todd)
  - OR632: non-linear programming (Lewis or Todd)
  - OR634: combinatorial optimization (Bland)
  - OR635: interior-point methods (Renegar or Todd)
  - OR639: convex analysis (Lewis)
  - CS621: matrix computations
  - CS622: non-linear equations and optimization.