

Lecture 14

Primal-dual interior-point methods

- primal-dual path-following
- Mehrotra's corrector step
- computing the search directions

14-1

Central path and complementary slackness

$$\begin{aligned}
 s + Ax - b &= 0 \\
 A^T z + c &= 0 \\
 z_i s_i &= 1/t, \quad i = 1, \dots, m \\
 z &\geq 0, \quad s \geq 0
 \end{aligned}$$

- continuous deformation of optimality conditions
- defines central path: solution is $x = x^*(t)$, $s = b - Ax^*(t)$, $z_i = 1/ts_i$
- $m + n$ linear and m nonlinear equations in the variables $s \in \mathbf{R}^m$, $x \in \mathbf{R}^n$, $z \in \mathbf{R}^m$

Interpretation of barrier method

apply Newton's method to

$$s + Ax - b = 0, \quad A^T z + c = 0, \quad z_i - 1/(ts_i) = 0, \quad i = 1, \dots, m$$

i.e., linearize around current x, z, s :

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & X^{-1}/t \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -(Ax + s - b) \\ -(A^T z + c) \\ \mathbf{1}/t - Xz \end{bmatrix}$$

where $X = \mathbf{diag}(s)$

solution (for $s + Ax - b = 0, A^T z + c = 0$):

- determine Δx from $A^T X^{-2} A \Delta x = -tc - A^T X^{-1} \mathbf{1}$ *i.e.*, Δx is the Newton direction used in barrier method
- substitute to obtain $\Delta s, \Delta z$

Primal-dual path-following methods

- modifications to the barrier method:
 - different linearization of central path
 - update both x and z after each Newton step
 - allow infeasible iterates
 - very aggressive step size selection (99% or 99.9% of step to the boundary)
 - update t after each Newton step (hence distinction between outer & inner iteration disappears)
 - linear or polynomial approximation to the central path
- limited theory, fewer convergence results
- work better in practice (faster and more reliable)

Primal-dual linearization

apply Newton's method to

$$\begin{aligned} s + Ax - b &= 0 \\ A^T z + c &= 0 \\ z_i s_i - 1/t &= 0, \quad i = 1, \dots, m \end{aligned}$$

i.e., linearize around s, x, z :

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta x \\ \Delta s \end{bmatrix} = \begin{bmatrix} -(Ax + s - b) \\ -(A^T z + c) \\ \mathbf{1}/t - Xz \end{bmatrix}$$

where $X = \mathbf{diag}(s)$, $Z = \mathbf{diag}(z)$

- iterates can be infeasible: $b - Ax \neq s$, $A^T z + c \neq 0$
- we assume $s > 0$, $z > 0$

computing $\Delta x, \Delta z, \Delta s$

1. compute Δx from

$$A^T X^{-1} Z A \Delta x = A^T z - A^T X^{-1} \mathbf{1}/t - r_z - A^T X^{-1} Z r_x$$

where $r_x = Ax + s - b$, $r_z = A^T z + c$

2. $\Delta s = -r_x - A \Delta x$

3. $\Delta z = X^{-1} \mathbf{1}/t - z - X^{-1} Z \Delta s$

the most expensive step is step 1

Affine scaling direction

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{aff}} \\ \Delta x^{\text{aff}} \\ \Delta s^{\text{aff}} \end{bmatrix} = \begin{bmatrix} -(Ax + s - b) \\ -(A^T z + c) \\ -Xz \end{bmatrix}$$

where $X = \mathbf{diag}(s)$, $Z = \mathbf{diag}(z)$

- limit of Newton direction for $t \rightarrow \infty$
- Newton step for

$$s + Ax - b = 0$$

$$A^T z + c = 0$$

$$z_i s_i = 0, \quad i = 1, \dots, m$$

i.e., the primal-dual optimality conditions

Centering direction

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{cent}} \\ \Delta x^{\text{cent}} \\ \Delta s^{\text{cent}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mathbf{1} \end{bmatrix}$$

where $X = \mathbf{diag}(s)$, $Z = \mathbf{diag}(z)$

- limit of Newton direction for $t \rightarrow 0$
- search direction is weighted sum of centering direction and affine scaling direction

$$\Delta x = (1/t)\Delta x^{\text{cent}} + \Delta x^{\text{aff}}$$

$$\Delta z = (1/t)\Delta z^{\text{cent}} + \Delta z^{\text{aff}}$$

$$\Delta s = (1/t)\Delta s^{\text{cent}} + \Delta s^{\text{aff}}$$

- in practice:
 - compute affine scaling direction first
 - choose t
 - compute centering direction and add to affine scaling direction

Heuristic for selecting t

- compute affine scaling direction
- compute primal and dual step lengths to the boundary along the affine scaling direction

$$\alpha_x = \max\{\alpha \in [0, 1] \mid s + \alpha \Delta s^{\text{aff}} \geq 0\}$$

$$\alpha_z = \max\{\alpha \in [0, 1] \mid z + \alpha \Delta z^{\text{aff}} \geq 0\}$$

- compute

$$\sigma = \left(\frac{(s + \alpha_x \Delta s^{\text{aff}})^T (z + \alpha_z \Delta z^{\text{aff}})}{s^T z} \right)^3$$

small σ means affine scaling directions are good search directions (significant reduction in $s^T z$)

- use $t = m/(\sigma s^T z)$ *i.e.*, search direction will be the Newton direction towards the central point with duality gap $\sigma s^T z$

a *heuristic*, based on extensive experiments

Mehrotra's corrector step

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{cor}} \\ \Delta x^{\text{cor}} \\ \Delta s^{\text{cor}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\Delta X^{\text{aff}} \Delta z^{\text{aff}} \end{bmatrix}$$

- higher-order correction to the affine scaling direction:

$$(s_i + \Delta s_i^{\text{aff}} + \Delta s_i^{\text{cor}})(z_i + \Delta z_i^{\text{aff}} + \Delta z_i^{\text{cor}}) \approx 0$$

- computation can be combined with centering step, *i.e.*, use

$$\Delta x = \Delta x^{\text{cc}} + \Delta x^{\text{aff}}, \quad \Delta z = \Delta z^{\text{cc}} + \Delta z^{\text{aff}}, \quad \Delta s = \Delta s^{\text{cc}} + \Delta s^{\text{aff}}$$

where

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{cc}} \\ \Delta x^{\text{cc}} \\ \Delta s^{\text{cc}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mathbf{1}/t - \Delta X^{\text{aff}} \Delta z^{\text{aff}} \end{bmatrix}$$

Step size selection

- determine step to the boundary

$$\alpha_x = \max\{\alpha \geq 0 \mid s + \alpha \Delta s \geq 0\}$$

$$\alpha_z = \max\{\alpha \geq 0 \mid z + \alpha \Delta z \geq 0\}$$

- update x, s, z

$$x := x + \min\{1, 0.99\alpha_x\} \Delta x$$

$$s := s + \min\{1, 0.99\alpha_x\} \Delta s$$

$$z := z + \min\{1, 0.99\alpha_z\} \Delta z$$

Algorithm

choose starting points x, z, s with $s > 0, z > 0$

1. evaluate stopping criteria

- primal feasibility: $\|Ax + s - b\| \leq \epsilon_1(1 + \|b\|)$
- dual feasibility: $\|A^T z + c\| \leq \epsilon_2(1 + \|c\|)$
- maximum absolute error: $c^T x + b^T z \leq \epsilon_3$
- maximum relative error:

$$\begin{aligned} c^T x + b^T z &\leq \epsilon_4 |b^T z| && \text{if } -b^T z > 0 \\ c^T x + b^T z &\leq \epsilon_4 |c^T x| && \text{if } c^T x < 0 \end{aligned}$$

2. compute affine scaling direction ($X = \text{diag}(s), Z = \text{diag}(z)$)

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{aff}} \\ \Delta x^{\text{aff}} \\ \Delta s^{\text{aff}} \end{bmatrix} = \begin{bmatrix} -(Ax + s - b) \\ -(A^T z + c) \\ -Xz \end{bmatrix}$$

3. compute steps to the boundary

$$\alpha_x = \max\{\alpha \in [0, 1] \mid s + \alpha \Delta s^{\text{aff}} \geq 0\}$$

$$\alpha_z = \max\{\alpha \in [0, 1] \mid z + \alpha \Delta z^{\text{aff}} \geq 0\}$$

4. compute centering-corrector steps

$$\begin{bmatrix} 0 & A & I \\ A^T & 0 & 0 \\ X & 0 & Z \end{bmatrix} \begin{bmatrix} \Delta z^{\text{cc}} \\ \Delta x^{\text{cc}} \\ \Delta s^{\text{cc}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sigma \frac{s^T z}{m} \mathbf{1} - \Delta X^{\text{aff}} \Delta z^{\text{aff}} \end{bmatrix}$$

where $\Delta X^{\text{aff}} = \text{diag}(\Delta s^{\text{aff}})$, and

$$\sigma = \left(\frac{(s + \alpha_x \Delta s^{\text{aff}})^T (z + \alpha_z \Delta z^{\text{aff}})}{s^T z} \right)^3$$

5. compute search directions

$$\Delta x = \Delta x^{\text{aff}} + \Delta x^{\text{cc}}, \quad \Delta s = \Delta s^{\text{aff}} + \Delta s^{\text{cc}}, \quad \Delta z = \Delta z^{\text{aff}} + \Delta z^{\text{cc}}$$

6. determine step sizes and update

$$\alpha_x = \max\{\alpha \geq 0 \mid s + \alpha \Delta s \geq 0\}$$

$$\alpha_z = \max\{\alpha \geq 0 \mid z + \alpha \Delta z \geq 0\}$$

$$x := x + \min\{1, 0.99\alpha_x\} \Delta x$$

$$s := s + \min\{1, 0.99\alpha_x\} \Delta s$$

$$z := z + \min\{1, 0.99\alpha_z\} \Delta z$$

go to step 1

Computing the search direction

most expensive part of one iteration: solve two sets of equations

$$A^T X^{-1} Z A \Delta x^{\text{aff}} = r_1, \quad A^T X^{-1} Z A \Delta x^{\text{cc}} = r_2$$

for some r_1, r_2

two methods

- sparse Cholesky factorization: used in all general-purpose solvers
- conjugate gradients: used for extremely large LPs, or LPs with special structure

Cholesky factorization

if $B = B^T \in \mathbf{R}^{n \times n}$ is positive definite, then it can be written as

$$B = LL^T$$

L lower triangular with $l_{ii} > 0$

- L is called the Cholesky factor of B
- costs $O(n^3)$ if B is dense

application: solve $Bx = d$ with $B = LL^T$

- solve $Ly = d$ (forward substitution)
- solve $L^T x = y$ (backward substitution)

Sparse Cholesky factorization

solve $Bx = d$ with B positive definite and sparse

1. *reordering* of rows and columns of B to increase sparsity of L
2. *symbolic factorization*: based on sparsity pattern of B , determine sparsity pattern of L
3. *numerical factorization*: determine L
4. *forward and backward substitution*: compute x

only steps 3,4 depend on the numerical values of B ; only step 4 depends on the right hand side; most expensive steps: 2,3

in Mehrotra's method with sparse LP: $B = A^T X^{-1} Z A$

- do steps 1,2 once, at the beginning of the algorithm ($A^T X^{-1} Z A$ has same sparsity pattern as $A^T A$)
- do step 3 once per iteration, step 4 twice

Conjugate gradients

solve $Bx = d$ with $B = B^T \in \mathbf{R}^{n \times n}$ positive definite

- iterative method
- requires n evaluations of Bx (in theory)
- faster if evaluation of Bx is cheap (*e.g.*, B is sparse, Toeplitz, . . .)
- much cheaper in memory than Cholesky factorization
- less accurate and robust (requires preconditioning)

in Mehrotra's method:

$$B = A^T X^{-1} Z A$$

evaluations Bx are cheap if evaluations Ax and $A^T y$ are cheap (*e.g.*, A is sparse)