Advanced Mathematical Programming
IE417

Lecture 21

Dr. Ted Ralphs
Reading for this lecture

- Sections 9.4-9.5
Barrier Methods

- So far, we have talked about *exterior penalty methods*.
- Now, we move on to *interior penalty methods* or *interior point methods*.
- The idea is similar, except now we start with a feasible point and impose a steep penalty for approaching the boundary.
- Previously, we let the penalty multiplier go to infinity. Now, we will let the penalty itself go to infinity.
- For reasons which will be obvious, these methods only work with inequality constraints.
Barrier Functions

• A barrier function is $B(x) = \sum_{i=1}^{m} \phi(g_i(x))$ where
  - $\phi$ is a continuous function of one variable,
  - $\phi(y) \geq 0$ if $y < 0$,
  - $\lim_{y \to 0^+} \phi(y) = \infty$,

• **Example**: $\phi(y) = -1/y$, $\phi(y) = \log(\min\{1, -y\})$

• Consider $\theta(\mu) = \inf\{f(x) + \mu B(x) : x \in X\}$

• What happens if we solve

\[
\begin{align*}
\min \theta(\mu) \\
\text{s.t. } \mu \geq 0
\end{align*}
\]
Performance of Barrier Methods

• If $f, g_i$ and $B$ are continuous, $X$ is a closed, nonempty compact set, then
  – For each $\mu > 0$, there exists an $x_\mu$ that minimizes $\theta(\mu)$.
  – $f(x_\mu)$ and $\theta(\mu)$ are nondecreasing functions of $\mu$.
  – $B(x_\mu)$ is a nonincreasing function of $\mu$.

• Under a few additional assumptions on the location of the optimal solution
  – $\min\{f(x) : g(x) \leq 0\} = \lim_{\mu \to 0^+} \theta(\mu) = \inf_{\mu > 0} \theta(\mu)$
  – All limit points of $x_\mu$ are optimal.
Implementing Barrier Methods

- **Initialization:** Choose termination scalar \( \epsilon > 0 \), an initial point \( x_1 \) with \( g(x_1) < 0 \), an initial penalty parameter \( \mu_1 > 0 \), and a scalar \( \beta \in (0, 1) \). Set \( k = 1 \).

- **Loop**
  - Minimize \( f(x) + \mu_k B(x) \) subject to \( g(x) < 0, x \in X \) to obtain \( x_{k+1} \).
  - If \( \mu_k B(x_{k+1}) < \epsilon \), then STOP. Otherwise, let \( \mu_{k+1} = \beta \mu_k \), replace \( k \) by \( k + 1 \) and iterate.

- Again notice the relationship to solving the Lagrangian dual.
Computational Issues

• As before, we can recover the Lagrange multipliers at optimality by computing \((\mu_\mu)_i \equiv \mu \phi'(g_i(x_\mu))\).

• Note that the search must start with a point \(x\) such that \(g(x) < 0\).

• As before, serious ill-conditioning can occur for small values of the barrier multiplier.

• We must make explicit checks to make sure we remain in the feasible region if fixed step-lengths are used.
Barrier Method for LP

• Consider the linear program

\[
\begin{align*}
    \text{min} & \quad c^T x \\
    \text{s.t.} & \quad Ax = b \\
    & \quad x \geq 0
\end{align*}
\]

• Optimality conditions are that \( \exists (x^*, u^*, v^*) \), such that

\[
\begin{align*}
    Ax^* & = b \quad x^* \geq 0 \\
    A^T v^* + u^* & = c \quad u^* \geq 0 \\
    u^T x^* & = 0
\end{align*}
\]
**Barrier Algorithm**

- Consider handling the non-negativity constraints with a barrier function $- \sum \ln(x_i)$.

- Then we solve the following barrier problem

$$\min\{c^T x - \mu \sum \ln(x_i) : Ax = b\}$$

- Optimality conditions are now

$$Ax^* = b$$
$$A^T v^* + u^* = c$$
$$u^* = \mu X^{-1} e$$
Comments on Barrier for LP

- Note that these are the old optimality conditions with the estimate of the KKT multipliers we discussed earlier.

- There exists a unique solution $x_\mu$ to these conditions.

- Further, the triple $w_\mu = (x_\mu, u_\mu, v_\mu)$ converges to the primal-dual optimal solution.

- We now have $u^T x = c^T x - b^T v = n_\mu$. This is the usual LP duality gap.

- Hence, the duality gap goes to zero as $\mu$ goes to zero, as expected.
Implementing Barrier for LP

• Start with a chosen $\mu_1 > 0$ and a corresponding $w_1 = (x_1, u_1, v_1)$ “sufficiently close” to $w_{\mu_1}$.

• Update $\mu_1$ to $\mu_2 = \beta \mu_1$ for some $0 < \beta < 1$.

• Use a Newton step to update $w_1$ to $w_2$

• *Sufficiently close* means

  \[
  \begin{align*}
  Ax^* &= b \\
  A^T u^* + u^* &= c \\
  u^T x &= n \mu \\
  \|X u^* - \mu e\| &\leq \Theta \mu \quad 0 \leq \Theta < 0.5
  \end{align*}
  \]
Solving Nonlinear Systems

- Solving a system of nonlinear equations is essentially equivalent to solving a NLP.
- Optimality conditions are just systems of non-linear equations that we are trying to solve.
- We can use a Newton method to solve a system of equations.
- Approximate the nonlinear system as a linear system, solve it, and iterate.
- We can use such a Newton method to implement the Barrier algorithm for LP.