15-850: Advanced Algorithms **CMU, Fall 2018**
Lecture #22: Interior-point Methods: an overview October 17, 2018
Lecturer: Anupam Gupta Scribe: Di Wang and Jalani Williams

1 Motivation

In this lecture, we will continue our discussion of polynomial-time algorithms for (constrained) convex optimization. In particular, we'll briefly derive an interior-point algorithm for LP solving, and, in doing so, illustrate the main idea behind Karmarkar's algorithm [Kar84]. For more details, please refer to [GM07] and [Wri97].

At this point, you might be wondering: "What's the point of going into such depth explaining these interior point methods –can't we just use them as a black box to solve LP's in linear time?" And, for the most part, you'd be right: LP solvers are often used as algorithmic black boxes, where most of the labor is done in casting the problem as an LP. However, by looking the inside of the black box, you can at times make notable improvements to the runtime of your algorithm: one example of this is in computing the maximum flow in directed graphs with weight at most U. For more info on that, see [Mąd16].

2 Setting the Scene

We will consider the following LP with *equality* constraints:

$$\begin{aligned} \min\ & c^\intercal x \\ \text{s.t. } & Ax = b \\ & x \geq 0 \end{aligned}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c, x \in \mathbb{R}^n$. We denote an optimal solution by x^*, which is given by

$$x^* := \operatorname*{argmin}_{x\in\mathbb{R}^n : Ax=b, x\geq 0} c^\intercal x.$$

3 Barrier Functions

Instead of solving this LP from first principles, in the interior point method we introduce a new variable η and exchange our constrained linear optimization problem for an unconstrained nonlinear one:

$$\begin{aligned} f_\eta(x) &:= c^\intercal x + \eta\left(\sum_{i=1}^n \log\frac{1}{x_i}\right) \\ x^*_\eta &:= \operatorname*{argmin}_{x\in\mathbb{R}^n : Ax=b} f_\eta(x) \end{aligned}$$

The intuition is that, when x approaches the boundary $x \geq 0$ of the feasible region, the *barrier function* $\sum_{i=1}^n \log\frac{1}{x_i}$ will approach $+\infty$. The new variable η lets us control the distance from x^*_η to the boundary. If η is close to 0, the term $c^\intercal x$ will dominate, and x^*_η should approach x^*. On the other hand, if η is sufficiently large, the barrier function will dominate and x^*_η will be an *interior point*, i.e., be close to the center of the feasible region.

Remark 22.1. In general, if we have *inequality* constraints $Ax \geq b$, we would also take the barrier function $\sum_{i=1}^{m} \log\left(\frac{1}{a_i^\intercal x - b_i}\right)$. By the same reasoning, when x is closed to the boundary of the feasible region, the barrier function will approach $+\infty$. Moreover, if the feasible region is bounded, the barrier function approaches $+\infty$ only as x gets closer to the boundary.

To establish a connection between x_η^* and x^* when η is small enough (and thus the gap $x_\eta^* - x^*$ is small enough), we can apply the same bit representation technique used in the analysis of the ellipsoid algorithm.

Claim 22.2. *The dual of the original LP is given by*

$$\begin{aligned} \max \;\; & b^\intercal y \\ \text{s.t. } & A^\intercal y \leq c \end{aligned}$$

and we can assume both the primal and the dual are strictly *feasible (i.e., both problems are feasible if we replace inequalities with strict ones). Thus $x \in \mathbb{R}^n$ is a minimizer of $f_\eta(x)$ if and only if there exist $y \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$ such that*

$$Ax - b = 0 \tag{22.1}$$

$$A^\intercal y + s = c \tag{22.2}$$

$$\forall i \in [n] : s_i x_i = \eta \tag{22.3}$$

With the claim above, we can derive an upper bound on the *duality gap* as follows:

$$\text{duality gap} = c^\intercal x - b^\intercal y = c^\intercal x - (Ax)^\intercal y = x^\intercal c - x^\intercal (c - s) = x^\intercal s = n \cdot \eta$$

thus if the representation size of the original LP is $L := \langle A \rangle + \langle b \rangle + \langle c \rangle$ and $\eta \leq 2^{-L}$, the duality gap is small enough that we can round x_η^* to obtain an optimal solution x^* to the original LP.

To prove the claim, we will use the method of *Lagrange multipliers* [BV04].

3.1 Some Background on Lagrange Multipliers

Theorem 22.3. *Let $f : U \to \mathbb{R}$ and $g_1, \cdots, g_m : U \to \mathbb{R}$ be C^1 functions (i.e., functions with continuous derivatives) where $U \subseteq \mathbb{R}^n$ is an open set. If x is a local optimum of the following optimization problem*

$$\begin{aligned} \min \;\; & f(x) \\ \text{s.t. } & \forall i \in [m] : g_i(x) = 0 \\ & x \in U \end{aligned}$$

then there exists $y^* \in \mathbb{R}^m$ such that $\nabla f(x^*) = \sum_{i=1}^{m} y_i^* \cdot \nabla g_i(x^*)$. Under certain conditions, this property is sufficient for a feasible solution to be optimal.

The strict feasibility of the primal and dual of the original LP that we assume in Claim 22.2 implies that the property in Thm. 22.3 is sufficient. Moreover, the LP is a convex optimization problem, local optima for it are global optima. Knowing this, we can directly apply the method of Lagrange multipliers.

Proof of Claim. 22.2.

$$
\begin{aligned}
& x \in \mathbb{R}^n \text{ is a minimizer of } f_\eta(x) \\
\iff & \exists y \in \mathbb{R}^m : Ax = b \wedge \nabla f_\eta(x) = \sum_{i=1}^m y_i \cdot \nabla(a_i^\intercal x - b_i) \\
\iff & \exists y \in \mathbb{R}^m : Ax - b = 0 \wedge c - \eta \cdot \left(\frac{1}{x_1}, \cdots, \frac{1}{x_n}\right) = \sum_{i=1}^m y_i \cdot a_i \\
\iff & \exists y \in \mathbb{R}^m, s \in \mathbb{R}^n : Ax - b = 0 \wedge A^\intercal y + s = c \wedge \forall i \in [n] : s_i = \frac{\eta}{x_i}.
\end{aligned}
$$

□

4 An Interior-Point Algorithm

In essence, the interior-point method has the following scheme:

1. Pick a sufficiently large η_0 and a starting point $x^{(0)}$ that is a minimizer of $f_{\eta_0}(x)$.
2. At step t, solve the corresponding problem with a smaller $\eta_{t+1} \coloneqq \eta_t \cdot (1 - \epsilon)$ and move to the corresponding minimizer $x^{(t+1)}$.
3. Repeat until η is small enough that we can apply Claim 22.2 to round x^*_η to obtain an optimal solution x^*.

With rounding handled, the least obvious step is to determine how to obtain $x^{(t+1)}$ from $x^{(t)}$ at step t. The difficulty comes from the nonlinearity of the system given by Eqns. (22.1), (22.2), and (22.3)r; we cannot simply use Claim 22.2 and linear system solving to compute $x^{(t+1)}$ directly. To get around this, we attempt to *approximate* the solution successively with a suitable linear system.

Remark 22.4. The general idea at work here is to somehow approximately follow the *central path* of x's that one would obtain by lowering η to 0 in infinitesimally small steps (shown in Figure 22.1). In general, interior point algorithms that make use of this idea are referred to as *path following* algorithms. Though we mainly follow [Kar84], this is far from the only one: check out [Ren88] and [NT08] if you're interested in seeing more!

Suppose at step t our solution $(x^{(t)}, y^{(t)}, s^{(t)})$ satisfies the following

$$Ax^{(t)} = b \tag{22.4}$$

$$A^\intercal y^{(t)} + s^{(t)} = c \tag{22.5}$$

$$\sqrt{\sum_{i=1}^n \left(s_i^{(t)} x_i^{(t)} - \eta_t\right)^2} \le 0.4\eta_t \tag{22.6}$$

(Note that we slightly weaken the condition $\forall i \in [n] : s_i^{(t)} x_i^{(t)} = \eta_t$ to Eqn. (22.6).)

At step t, we update our values in the following way:

1. Set $\eta_{t+1} \coloneqq \eta_t \cdot \left(1 - \frac{0.4}{\sqrt{n}}\right)$.

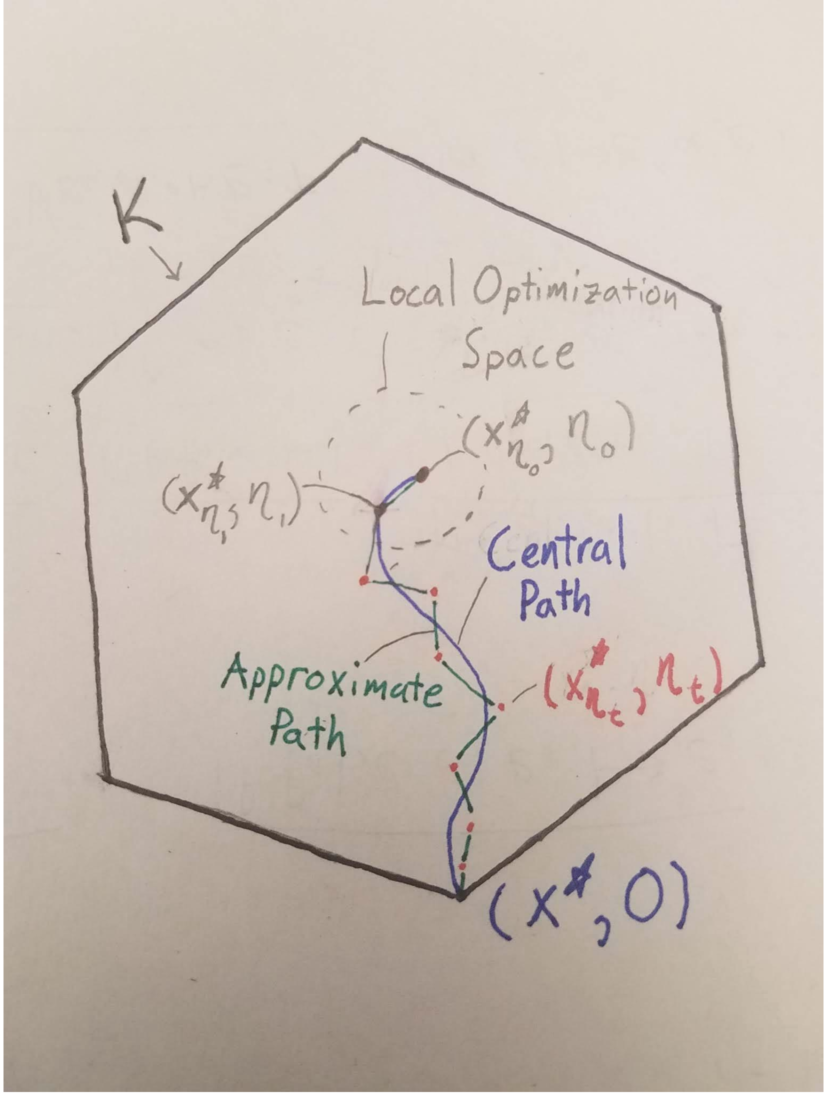

Figure 22.1: A visualization of a path following algorithm.

2. Solve the following *linear* system to find $\Delta x, \Delta y, \Delta s$:

$$A\Delta x = 0 \tag{22.7}$$

$$A^\intercal \Delta y + \Delta s = 0 \tag{22.8}$$

$$\forall i \in [n] : s_i^{(t)} x_i^{(t)} + s_i^{(t)} \Delta x_i + x_i^{(t)} \Delta s_i = \eta_{t+1} \tag{22.9}$$

3. Set $x^{(t+1)} := x^{(t)} + \Delta x$, $y^{(t+1)} := y^{(t)} + \Delta y$, $s^{(t+1)} := s^{(t)} + \Delta s$.

The following claims establish the efficacy and efficiency of the algorithm.

Claim 22.5. *For every step t, we will have $\left(x^{(t+1)}\right)^\intercal s^{(t+1)} = \left\langle x^{(t+1)}, s^{(t+1)} \right\rangle = n \cdot \eta_{t+1}$. This property enables us to use Claim 22.2 to reason about the duality gap.*

Proof.

$$\begin{aligned}
\left\langle x^{(t+1)}, s^{(t+1)} \right\rangle &= \left\langle x^{(t)} + \Delta x, s^{(t)} + \Delta s \right\rangle \\
&= \sum_{i=1}^{n} \left(s_i^{(t)} x_i^{(t)} + s_i^{(t)} \Delta x_i + x_i^{(t)} \Delta s_i + \Delta s_i \Delta x_i \right) \\
&= n \cdot \eta_{t+1} + \sum_{i=1}^{n} \Delta s_i \Delta x_i \\
&= n \cdot \eta_{t+1} + (\Delta s)^\intercal \Delta x \\
&= n \cdot \eta_{t+1} - \left(A^\intercal \Delta y\right)^\intercal \Delta x \\
&= n \cdot \eta_{t+1} - (\Delta y)^\intercal A \Delta x \\
&= n \cdot \eta_{t+1}.
\end{aligned}$$

□

Claim 22.6. *The given algorithm maintains the invariants given by Eqns. (22.4), (22.5), and (22.6) for every step t.*

Proof.

- To show Eqn. (22.4):

$$Ax^{(t+1)} = A\left(x^{(t)} + \Delta x\right) = Ax^{(t)} + A\Delta x = b + 0 = b.$$

- To show Eqn. (22.5):

$$A^\intercal y^{(t+1)} + s^{(t+1)} = A^\intercal \left(y^{(t)} + \Delta y\right) + \left(s^{(t)} + \Delta s\right) = \left(A^\intercal y^{(t)} + s^{(t)}\right) + (A^\intercal \Delta y + \Delta s) = c + 0 = c.$$

- To show Eqn. (22.6):

$$\begin{aligned}
\sum_{i=1}^{n} \left(s_i^{(t+1)} x_i^{(t+1)} - \eta_{t+1} \right)^2 &= \sum_{i=1}^{n} \left(s_i^{(t)} x_i^{(t)} + s_i^{(t)} \Delta x_i + x_i^{(t)} \Delta s_i + \Delta s_i \Delta x_i - \eta_{t+1} \right)^2 \\
&= \sum_{i=1}^{n} \left(\Delta s_i \Delta x_i\right)^2.
\end{aligned}$$

It suffices to show that $\sqrt{\sum_{i=1}^n (\Delta s_i \Delta x_i)^2} \leq 0.4\eta_{t+1}$. Set $x := x^{(t)}$ and $s := s^{(t)}$. Define $D := \operatorname{diag}\left(\sqrt{\frac{x_i}{s_i}}\right)$. Using the inequality

$$\sqrt{\sum_{i=1}^n (a_i b_i)^2} \leq \frac{1}{2^{3/2}} \sum_{i=1}^n (a_i + b_i)^2,$$

we have

$$\begin{aligned}
\sqrt{\sum_{i=1}^n (\Delta s_i \Delta x_i)^2} &= \sqrt{\sum_{i=1}^n \left((D\Delta s)_i (D^{-1}\Delta x)_i\right)^2} \\
&\leq \frac{1}{2^{3/2}} \sum_{i=1}^n \left((D\Delta s)_i + (D^{-1}\Delta x)_i\right)^2 \\
&= \frac{1}{2^{3/2}} \sum_{i=1}^n \left(\frac{x_i}{s_i} \cdot (\Delta s_i)^2 + \frac{s_i}{x_i} \cdot (\Delta x_i)^2\right) \qquad [(\Delta s)^\intercal \Delta x = 0 \text{ by Claim 22.5}] \\
&= \frac{1}{2^{2/3}} \sum_{i=1}^n \frac{(x_i \Delta s_i)^2 + (s_i \Delta x_i)^2}{s_i x_i} \\
&\leq \frac{1}{2^{2/3}} \frac{\sum_{i=1}^n (x_i \Delta s_i + s_i \Delta x_i)^2}{\min_{i\in[n]} s_i x_i} \\
&= \frac{1}{2^{2/3}} \frac{\sum_{i=1}^n (\eta_{t+1} - s_i x_i)^2}{\min_{i\in[n]} s_i x_i}.
\end{aligned}$$

Observing that $\min_{i\in[n]} s_i x_i$ is maximized if $s_i x_i = s_j x_j$ for all $i, j \in [n]$, we consider a vector that maximizes $\min_{i\in[n]} s_i x_i$. By the invariants for the step $t-1$, i.e., the previous step, we have

$$\begin{aligned}
&\sqrt{\sum_{i=1}^n (s_i x_i - \eta_t)^2} \leq 0.4\eta_t \\
\implies\quad &\sqrt{n} \cdot |\min_{i\in[n]} s_i x_i - \eta_t| \leq 0.4\eta_t \\
\implies\quad &\min_{i\in[n]} s_i x_i \geq \left(1 - \frac{0.4}{\sqrt{n}}\right)\eta_t.
\end{aligned}$$

Let $\sigma := \left(1 - \frac{0.4}{\sqrt{n}}\right)$. Then

$$\begin{aligned}
\sum_{i=1}^n (\eta_{t+1} - s_i x_i)^2 &= \sum_{i=1}^n (\sigma\eta_t - s_i x_i)^2 \\
&= \sum_{i=1}^n \left((s_i x_i - \eta_t) + (1-\sigma)\eta_t\right)^2 \\
&= \sum_{i=1}^n (s_i x_i - \eta_t)^2 + \sum_{i=1}^n ((1-\sigma)\eta_t)^2 + 2\sum_{i=1}^n (s_i x_i - \eta_t)(1-\sigma)\eta_t.
\end{aligned}$$

On the other hand, by Claim 22.5 we have

$$\sum_{i=1}^n (s_i x_i - \eta_t) = \sum_{i=1}^n s_i x_i - n \cdot \eta_t = 0.$$

Thus

$$\begin{aligned}\sum_{i=1}^{n}(\eta_{t+1}-s_ix_i)^2 &= \sum_{i=1}^{n}(s_ix_i-\eta_t)^2+\sum_{i=1}^{n}((1-\sigma)\eta_t)^2\\ &\le (0.4\eta_t)^2+n(1-\sigma)^2\eta_t^2\\ &= 0.4^2\eta_t^2+n\cdot\frac{0.4^2}{n}\cdot\eta_t^2\\ &= 2(0.4)^2\eta_t^2\end{aligned}$$

and therefore

$$\begin{aligned}\sqrt{\sum_{i=1}^{n}(\Delta s_i\Delta x_i)^2} &\le \frac{1}{2^{2/3}}\frac{\sum_{i=1}^{n}(\eta_{t+1}-s_ix_i)^2}{\min_{i\in[n]} s_ix_i}\\ &\le \frac{1}{2^{3/2}}\frac{2(0.4)^2\eta_t^2}{\sigma\eta_t}\\ &= \frac{0.4}{\sqrt{2\sigma^2}}0.4\eta_{t+1}.\end{aligned}$$

By the fact that $\frac{0.4}{\sqrt{2\sigma^2}}\le\frac{0.4}{\sqrt{2(0.6)^2}}\le 1$, we conclude that $\sqrt{\sum_{i=1}^{n}(\Delta s_i\Delta x_i)^2}\le 0.4\eta_{t+1}$.

□

Claim 22.7. *Recall that we define the representation size of the original LP as $L := \langle A\rangle+\langle b\rangle+\langle c\rangle$. With η_0 bounded by 2^L, after $T=\Theta(\sqrt{n}\cdot L)$ steps, we will have $\eta_T\le 2^{-L}$. Together with Claims 22.2 and 22.5, we are able to round $x^{(T)}$ to an optimal solution to the original LP after T steps.*

Proof. Set $T := 10\sqrt{n}L$.

$$\eta_T=\left(1-\frac{0.4}{\sqrt{n}}\right)^T\eta_0\le\left(1-\frac{0.4}{\sqrt{n}}\right)^{10\sqrt{n}L}\cdot 2^L\le\exp\left(-4L\right)\cdot 2^L\le 2^{-L}.$$

□

Because solving a linear system can be done in polynomial time, and we only need to solve a polynomial number of linear systems to solve this LP, the interior-point algorithm above solves LP's in polynomial time. In other words, we prove the following theorem.

Theorem 22.8. *Given an LP $\{\min c^\intercal x : Ax=b, x\ge 0\}$ with representation size $L := \langle A\rangle+\langle b\rangle+\langle c\rangle$ and an initial feasible (x^0_η,η_0) pair, the interior-point algorithm proposed above can find an optimal solution x^* in time* $\operatorname{poly}(L,n)$.

Remark 22.9. The first step in the algorithm that (picking an initial $(x^{(0)},\eta_0)$ pair is in fact non-trivial. One possible approach is to run the interior-point algorithm "in reverse". The idea is that we can start with a vertex of the feasible region by solving the linear system $Ax=b$, and then successively *increase* η through a similar mechanism as the interior-point algorithm until the value of η is sufficiently large to begin the algorithm.

Acknowledgments

These lecture notes were scribed by Di Wang and Jalani Williams, based on previous scribe notes of Guru Guruganesh and Nicholas Sieger.

References

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. 3

[GM07] B. Gärtner and J. Matoušek. *Understanding and Using Linear Programming*. Universitext. Springer, Berlin, 2007. 1

[Kar84] N. Karmarkar. A New Polynomial-Time Algorithm for Linear Programming. *Combinatorica*, 4(4):373–395, 1984. 1, 22.4

[Mąd16] A. Mądry. Computing Maximum Flow with Augmenting Electrical Flows. *FOCS*, 2016. 1

[NT08] A. S. Nemirovski and M. J. Todd. Interior-point methods for optimization. *Acta Numerica*, pages 191–234, 2008. 22.4

[Ren88] J. Renegar. A polynomial-time algorithm, based on Newton's method, for linear programming. *Mathematical Programming*, 40:59–93, 1988. 22.4

[Wri97] S. J. Wright. *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997. 1