CS395T: Continuous Algorithms, Part X
Interior-point methods

Kevin Tian

1 Self-concordance

In Part III, we developed the mirror descent algorithm, which associated a convex set $\mathcal{X}$ with a regularizer $\varphi : \mathcal{X} \to \mathbb{R}$, satisfying certain compatible regularity conditions with the geometry of $\mathcal{X}$. Specifically, in analyzing mirror descent we assumed that $\mathcal{X}$ was bounded with respect to a norm $\|\cdot\|$, and that φ was strongly convex in the same norm. In this lecture, we explore a different way of designing optimization algorithms which cater to the geometry of a constraint set. As we will see, the geometric viewpoint we develop is somewhat more explicitly tied to the function φ, which induces a "local geometry" pointwise in $\mathcal{X}$ through the curvature of its Hessian $\nabla^2\varphi$. The benefit of this characterization is that we have strong control of the local behavior of φ, enabling us to develop algorithms with significantly improved convergence behavior in small neighborhoods. As a payoff of building this viewpoint, in Sections 3 and 4 we introduce *interior-point methods*, a landmark algorithmic framework which has been highly influential in both the theory and practice of linear programming, as well as solving other structured convex programs.[1]

We now make this intuition formal. Throughout, we let $\mathcal{X} \subseteq \mathbb{R}^d$ be an open convex set, and we let $\varphi : \mathcal{X} \to \mathbb{R}$ be a barrier function, which means that $\varphi(x) \to \infty$ as x approaches the boundary of the closure of $\mathcal{X}$. We begin with a one-dimensional definition of *self-concordance*, the key structural assumption on φ which gives us the aforementioned local control.

Definition 1 (Self-concordance in $\mathbb{R}$)**.** *Let $\mathcal{X} \subseteq \mathbb{R}$ be convex and open, and let $\varphi : \mathcal{X} \to \mathbb{R}$ be three-times differentiable and a barrier function. We say φ is a* self-concordant barrier *for $\mathcal{X}$ if*

$$|\varphi'''(x)| \le 2\left(\varphi''(x)\right)^{\frac{3}{2}} \text{ for all } x \in \mathcal{X}.$$

We observe that Definition 1 implies that φ is convex, as $\varphi'' \ge 0$ pointwise on $\mathcal{X}$. To give a motivating example, consider $\varphi(x) := -\log(x)$, on $\mathcal{X} := \mathbb{R}_{>0}$. Clearly φ explodes as $x \to 0$, the boundary of $\mathcal{X}$. Moreover, a simple computation shows

$$|\varphi'''(x)| = \left|\frac{2}{x^3}\right|,\ \varphi''(x) = \frac{1}{x^2}, \tag{1}$$

so indeed φ is a self-concordant barrier for $\mathcal{X}$. For a list of one-dimensional self-concordant barrier functions for various constraint sets, we refer the reader to Lemma 5.38, [LV23]. One crucial aspect of Definition 1 is that it is affine-invariant: if φ is a self-concordant barrier for $\mathcal{X} \subseteq \mathbb{R}$, then for any $a, b \in \mathbb{R}$, $\varphi(a \cdot + b)$ is a self-concordant barrier for $a\mathcal{X} + b := \{ax + b \mid x \in \mathcal{X}\}$.

We now move on to defining a multivariate extension of Definition 1.

Definition 2 (Self-concordance)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ be convex and open, and let $\varphi : \mathcal{X} \to \mathbb{R}$ be three-times differentiable and a barrier function. We say φ is a* self-concordant barrier *for $\mathcal{X}$ if, letting $\mathbf{T}(x) := \nabla^3\varphi(x)$ and $\mathbf{H}(x) := \nabla^2\varphi(x)$ for notational simplicity,*

$$-2\|v\|_{\mathbf{H}(x)}\mathbf{H}(x) \preceq \mathbf{T}(x)[v, \cdot, \cdot] \preceq 2\|v\|_{\mathbf{H}(x)}\mathbf{H}(x) \text{ for all } x \in \mathcal{X},\ v \in \mathbb{R}^d.$$

To demystify Definition 2, we provide the following equivalent, more intuitive definition.

[1] For example, Karmarkar's interior-point method [Kar84] appeared on the cover of the New York Times [Gle84].

Lemma 1. *In the setting of Definition 2, φ is a self-concordant barrier for $\mathcal{X}$ iff for all $v \in \mathbb{R}^d, x \in \mathcal{X}$, the restriction $\varphi_{v,x}(t) = \varphi(x + tv)$ is self-concordant for all $t \in \mathbb{R}$ such that $x + tv \in \mathcal{X}$.*

Proof. If Definition 2 holds, then $\varphi_{v,x}$ is indeed self-concordant, since this is implied by

$$|\mathbf{T}(x+tv)[v,v,v]| \le 2 \|v\|_{\mathbf{H}(x+tv)}^{\frac{3}{2}},$$

which is true by assumption. On the other hand, suppose all the one-dimensional restrictions are self-concordant. This means that for all $x \in \mathcal{X}$, the trilinear form $\widetilde{\mathbf{T}}$ whose action is given by

$$\widetilde{\mathbf{T}}[u,v,w] = \mathbf{T}\left[\mathbf{H}^{-\frac{1}{2}}u, \mathbf{H}^{-\frac{1}{2}}v, \mathbf{H}^{-\frac{1}{2}}w\right], \text{ for } \mathbf{T} := \mathbf{T}(x),\ \mathbf{H} := \mathbf{H}(x),$$

satisfies $\widetilde{\mathbf{T}}[v,v,v] \le 2$ for all $\|v\|_2 \le 1$, since this is equivalent to $\mathbf{T}[v,v,v] \le 2\|v\|_{\mathbf{H}}^{3/2}$. Similarly, the condition in Definition 2 is equivalent to $\widetilde{\mathbf{T}}[u,v,v] \le 2$ for all $\|u\|_2, \|v\|_2 \le 1$. The conclusion follows because for every symmetric trilinear form $\widetilde{\mathbf{T}}$, it is the case that $\widetilde{\mathbf{T}}[u,v,w]$ is maximized over $\|u\|_2, \|v\|_2, \|w\|_2 \le 1$ by some triplet with $u = v = w$ (for a proof, see Section 2.3, [Nem04]). $\square$

One useful aspect of Definition 2 is that it satisfies composition properties.

Lemma 2. *Let $\mathcal{X}, \mathcal{X}' \subseteq \mathbb{R}^d$ be convex and open, and let φ, ϕ be three-times differentiable and barrier functions for $\mathcal{X}, \mathcal{X}'$ respectively.*

1. *If φ, ϕ are self-concordant barriers for $\mathcal{X}, \mathcal{X}'$ respectively, then $\varphi + \phi$ is a self-concordant barrier for $\mathcal{X} \cap \mathcal{X}'$.*
2. *If φ is a self-concordant barrier for $\mathcal{X}$, for any $\mathbf{A} \in \mathbb{R}^{d\times n}$, $b \in \mathbb{R}^d$, the function $\varphi_{\mathbf{A},b}(y) := \varphi(\mathbf{A}y + b)$ is a self-concordant barrier for $\mathcal{Y} := \{y \in \mathbb{R}^n \mid \mathbf{A}y + b \in \mathcal{X}\}$.*

Proof. To see the first claim, fix $x \in \mathcal{X} \cap \mathcal{X}'$, $v \in \mathbb{R}^d$, and consider the one-dimensional restrictions $\varphi_{v,x}, \phi_{v,x}$ as defined in Lemma 1. We have for all t such that $x + tv \in \mathcal{X} \cap \mathcal{X}'$,

$$\begin{aligned}\left|\varphi'''_{v,x}(t) + \phi'''_{v,x}(t)\right| &\le \left|\varphi'''_{v,x}(t)\right| + \left|\phi'''_{v,x}(t)\right| \\ &\le 2\left(\varphi''_{v,x}(t)\right)^{\frac{3}{2}} + 2\left(\phi''_{v,x}(t)\right)^{\frac{3}{2}} \le 2\left(\varphi''_{v,x}(t) + \phi''_{v,x}(t)\right)^{\frac{3}{2}},\end{aligned}$$

since $(a+b)^{3/2} \ge a^{3/2} + b^{3/2}$ for all $a, b \ge 0$. To see the second claim, again consider a one-dimensional restriction $[\varphi_{\mathbf{A},b}]_{v,x}$, such that

$$[\varphi_{\mathbf{A},b}]_{v,x}(t) = \varphi\left((\mathbf{A}x + b) + t\mathbf{A}v\right).$$

It hence suffices to show that for all v, x such that $\mathbf{A}x + b \in \mathcal{X}$,

$$|\mathbf{T}[\mathbf{A}v, \mathbf{A}v, \mathbf{A}v]| \le 2\|\mathbf{A}v\|_{\mathbf{H}}^{\frac{3}{2}},$$

for $\mathbf{T} := \nabla^3\varphi(\mathbf{A}x+b+t\mathbf{A}v)$, $\mathbf{H} := \nabla^2\varphi(\mathbf{A}x+b+t\mathbf{A}v)$, which follows via Lemma 1 with $v \leftarrow \mathbf{A}v$. $\square$

We will give an application of Lemma 2 to the setting where $\mathcal{X}$ is a polytope $\{x \in \mathbb{R}^d \mid \mathbf{A}x \le b\}$ at the end of the section. In the remainder of the section, unless otherwise specified, we fix an open, convex $\mathcal{X} \subseteq \mathbb{R}^d$, and let φ be a self-concordant barrier for $\mathcal{X}$. We also let $\mathbf{T}(x) := \nabla^3\varphi(x)$ and $\mathbf{H}(x) := \nabla^2\varphi(x)$ for shorthand, and abbreviate the quadratic norm[2] in the Hessian and its dual:

$$\|v\|_x := \|v\|_{\mathbf{H}(x)},\ \|v\|_{x,*} := \|v\|_{\mathbf{H}(x)^\dagger}, \text{ for all } x \in \mathcal{X},\ v \in \mathbb{R}^d.$$

We next develop a crucial implication of Definition 2: in appropriate local neighborhoods, we can capture the behavior of φ using a quadratic model in $\nabla^2\varphi$. To make this more precise, we introduce the following definition of a local neighborhood around a point $x \in \mathcal{X}$ induced by $\|\cdot\|_x$.

Definition 3 (Dikin ellipse)**.** *Let φ be a self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$, and let $r \ge 0$. We define the* Dikin ellipse *at $x \in \mathcal{X}$ by*

$$\mathcal{E}_x(r) := \{y \in \mathcal{X} \mid \|y - x\|_x \le r\}.$$

[2] If $\mathbf{H}(x)$ is degenerate, i.e. it has a kernel, then $\|\cdot\|_{\mathbf{H}(x)}$ is technically only a seminorm, rather than a norm.

We are now ready to state and prove the key local regularity condition imposed by self-concordance.

Lemma 3. *Let φ be a self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$. If $\|y-x\|_x < 1$,*

$$(1 - \|y-x\|_x)^2 \mathbf{H}(x) \preceq \mathbf{H}(y) \preceq \frac{1}{(1-\|y-x\|_x)^2}\mathbf{H}(x).$$

Proof. Let $v := y - x$, $x_t := x + tv$, and $\phi(t) := \|v\|_{x_t}^2$. Then, by self-concordance,

$$|\phi'(t)| = |\mathbf{T}(x_t)[v,v,v]| \le 2\|v\|_{x_t}^3 = 2\phi(t)^{\frac{3}{2}}. \tag{2}$$

Rearranging shows that for $t \in [0,1]$,

$$\left|\frac{\mathrm{d}}{\mathrm{d}t}\frac{1}{\sqrt{\phi(t)}}\right| \le 1 \implies \frac{1}{\sqrt{\phi(t)}} \ge \frac{1}{\sqrt{\phi(0)}} - t \implies \phi(t) \le \frac{\phi(0)}{\left(1 - t\sqrt{\phi(0)}\right)^2}.$$

Next, for any $u \in \mathbb{R}^d$, let $\psi(t) := \|u\|_{x_t}^2$. By a variant of the calculation (2), where we obtain a trilinear form applied to $(v,u,u,)$, we have $|\psi'(t)| \le 2\sqrt{\phi(t)}\psi(t)$, so that

$$\left|\frac{\mathrm{d}}{\mathrm{d}t}\log\psi(t)\right| \le 2\sqrt{\phi(t)} \le \frac{2\sqrt{\phi(0)}}{1-t\sqrt{\phi(0)}}$$
$$\implies \left|\log\frac{\psi(1)}{\psi(0)}\right| \le \int_0^1 \frac{2\sqrt{\phi(0)}}{1-t\sqrt{\phi(0)}}\mathrm{d}t = 2\log\left(\frac{1}{1-\sqrt{\phi(0)}}\right).$$

Rearranging and using $\sqrt{\phi(0)} = \|y-x\|_x$ gives the result, since $\psi(1) = \|u\|_y^2$ and $\psi(0) = \|u\|_x^2$. □

Lemma 3 shows that as long as y is in the interior of $\mathcal{E}_x(1)$, $\mathbf{H}(y)$ approximates $\mathbf{H}(x)$ up to a constant. In other words, the Hessian of φ is multiplicatively stable within Dikin ellipses of radius 1. Because of the barrier property of φ, i.e. $\varphi \to \infty$ when approaching the boundary of $\mathcal{X}$, this also implies that the interiors of all Dikin ellipses $\mathcal{E}_x(1)$ are contained in $\mathcal{X}$.

We introduce one additional parameterization of self-concordant functions, which is important in applications because it gives us control of how the minimizers of self-concordant barriers change under multiplicative perturbations. In particular, in Sections 2 and 3 we will use the norm of the gradient as a potential measuring the optimality of iterates for minimizing self-concordant barrier functions. The following definition bounds the worst-case gradient norm over a set.

Definition 4 (ν-self-concordance)**.** *In the setting of Definition 2, for $\nu \ge 0$, we say φ is a* ν-self-concordant barrier *for $\mathcal{X}$ if it is a self-concordant barrier for $\mathcal{X}$, and*

$$(\nabla\varphi(x))(\nabla\varphi(x))^\top \preceq \nu\mathbf{H}(x) \text{ for all } x \in \mathcal{X}.$$

An equivalent characterization to Definition 4 when $\mathbf{H}(x)$ is full rank is simply that

$$\|\nabla\varphi(x)\|_{x,*} \le \sqrt{\nu} \text{ for all } x \in \mathcal{X}.$$

When $\mathbf{H}(x)$ is degenerate, ν-self-concordance imposes both the above condition and that $\nabla\varphi(x) \in \mathrm{Span}(\mathbf{H}(x))$ pointwise. Namely, while self-concordance shows that $\mathcal{X}$ is locally "large" when viewed through the local norm induced by $\mathbf{H}(x)$ (since it contains the interior of $\mathcal{E}_x(1)$), i.e. it lower bounds the size of $\mathcal{X}$ in this local norm, ν-self-concordance further upper bounds $\mathcal{X}$ in the direction of $\nabla\varphi(x)$. This property was called "explosure" in [Nem04], which we now formalize.

Lemma 4. *Let φ be a ν-self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$. If $v \in \mathbb{R}^d$, $x \in \mathcal{X}$ satisfy $\langle\nabla\varphi(x), v\rangle > 0$, then for all $t \ge \frac{\nu}{\langle\nabla\varphi(x),v\rangle}$, we have $x + tv \notin \mathcal{X}$.*

Proof. For all $t \ge 0$, define $x_t := x + tv$, $\phi(t) := \langle\nabla\varphi(x_t), v\rangle$, and $\phi'(t) = \mathbf{H}(x_t)[v,v]$. We observe that ν-self-concordance implies that $\phi(t)^2 \le \nu\phi'(t)$, so that

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(-\frac{1}{\phi(t)}\right) \ge \frac{1}{\nu} \implies -\frac{1}{\phi(t)} \ge -\frac{1}{\phi(0)} + \frac{t}{\nu} \implies \phi(t) \ge \frac{\nu\phi(0)}{\nu - t\phi(0)}.$$

Now, if $x_t \in \mathcal{X}$, $\phi(t)$ is bounded, so the above inequality implies $\nu - t\phi(0) > 0$, and rearranging yields the conclusion, by taking the contrapositive of the desired claim and using $\phi(0) = \langle\nabla\varphi(x), v\rangle$. □

Lemma 4 shows that if v is a vector with unit size when measured in the direction of $\nabla\varphi(x)$, then $x+\nu v$ is no longer in the set $\mathcal{X}$. In other words, there is a limit to how much we can perturb points by their gradients $\nabla\varphi(x)$ while not leaving $\mathcal{X}$, and ν-self-concordance parameterizes the maximum perturbation. We next give an analogous composition property of Definition 4, extending Lemma 2.

Lemma 5. *Let $\mathcal{X}, \mathcal{X}' \subseteq \mathbb{R}^d$ be convex and open, and let φ, ϕ be three-times differentiable and barrier functions for $\mathcal{X}, \mathcal{X}'$ respectively.*

1. *If φ, ϕ are ν-self-concordant and μ-self-concordant barriers for $\mathcal{X}, \mathcal{X}'$ respectively, then $\varphi + \phi$ is a $(\nu+\mu)$-self-concordant barrier for $\mathcal{X} \cap \mathcal{X}'$.*
2. *If φ is a ν-self-concordant barrier for $\mathcal{X}$, for any $\mathbf{A} \in \mathbb{R}^{d\times n}$, $b \in \mathbb{R}^d$, the function $\varphi_{\mathbf{A},b}(y) := \varphi(\mathbf{A}y+b)$ is a ν-self-concordant barrier for $\mathcal{Y} := \{y \in \mathbb{R}^n \mid \mathbf{A}y + b \in \mathcal{X}\}$.*

Proof. We proved all the properties required in Lemma 2, except for the gradient norm bound required by Definition 4, which we prove presently. For the first claim, for any $u \in \mathbb{R}^d$,

$$\begin{aligned}
|\langle \nabla\varphi(x) + \nabla\phi(x), u\rangle| &\le |\langle \nabla\varphi(x), u\rangle| + |\langle \nabla\phi(x), u\rangle| \\
&\le \sqrt{\nu \cdot \nabla^2\varphi(x)[u,u]} + \sqrt{\mu \cdot \nabla^2\phi(x)[u,u]} \\
&\le \sqrt{\nu+\mu} \cdot \sqrt{\nabla^2\varphi(x)[u,u] + \nabla^2\phi(x)[u,u]},
\end{aligned}$$

where the last line used the Cauchy-Schwarz inequality. Squaring then gives the desired $(\nu+\mu)$-self-concordance of $\varphi+\phi$. For the second claim, ν-self-concordance of $\varphi_{\mathbf{A},b}$ is equivalent to

$$(\mathbf{A}\nabla\varphi(x))\,(\mathbf{A}\nabla\varphi(x))^\top \preceq \nu \mathbf{A}\mathbf{H}(x)\mathbf{A}^\top,$$

which follows from ν-self-concordance of φ. □

To give a motivating example tying together this section, and to introduce some calculations which will be used in Section 4, we finally give a self-concordant barrier over a polytope.

Lemma 6. *Let $\mathbf{A} \in \mathbb{R}^{n\times d}$ have rows $\{a_i\}_{i\in[n]}$, let $b \in \mathbb{R}^n$, and define the polytope*

$$\mathcal{X} := \{x \mid \mathbf{A}x < b\}.$$

Then φ is an n-self-concordant barrier for $\mathcal{X}$, where

$$\varphi(x) := -\sum_{i\in[n]} \log\left(b_i - a_i^\top x\right). \tag{3}$$

Proof. To establish that Definition 2 holds for this choice of φ, we first observe that letting $\mathcal{Y} := \mathbb{R}^n_{>0}$, applying the first property in Lemma 2 shows that $-\sum_{i\in[n]}\log(y_i)$ is a self-concordant barrier for $\mathcal{Y}$, because $-\log(y)$ is a self-concordant barrier for $\mathbb{R}_{>0}$ as shown by (1). Therefore, applying the second property in Lemma 2 shows that φ is a self-concordant barrier for $\mathcal{X}$, under the affine transformation $y = b - \mathbf{A}x$ which sends $\mathcal{X}$ to $\mathcal{Y}$. We can establish the bound $\nu = m$ in Definition 4 using a similar argument, where we apply Lemma 5 in place of Lemma 2.

We instead present a more direct calculation, to introduce some useful computations specialized to (3). Letting φ be as in (3) and $x \in \mathcal{X}$, we derive[3]

$$\begin{gathered}
\nabla\varphi(x) = \mathbf{A}^\top\mathbf{S}_x^{-1}\mathbb{1}_n,\ \nabla^2\varphi(x) = \mathbf{A}^\top\mathbf{S}_x^{-2}\mathbf{A}, \\
\text{where } s_x := b - \mathbf{A}x,\ \mathbf{S} := \mathbf{diag}\,(s_x)\,.
\end{gathered} \tag{4}$$

Then, we have the desired n-self-concordance from the following calculation: for any $v \in \mathbb{R}^d$,

$$(\langle v, \nabla\varphi(x)\rangle)^2 = \left\langle \mathbb{1}_n, \mathbf{S}_x^{-1}\mathbf{A}v\right\rangle^2 \le n\left\|\mathbf{S}_x^{-1}\mathbf{A}v\right\|_2^2 = n\nabla^2\varphi(x)[v,v].$$

□

[3]The definition of the variable s_x in (4) corresponds to the *slack variables* on $x \in \mathcal{X}$, i.e. $s_x = b - \mathbf{A}x$ captures how close the constraints $\mathbf{A}x \le b$ are to being violated.

2 Newton's method

In this section, we analyze an algorithm for optimizing self-concordant barriers φ when initialized at a point which is already close to the minimizer. Intuitively, this is possible in a Dikin ellipse centered at the minimizer of φ, because Lemma 3 shows that in this local region the Hessian of φ is multiplicatively stable, and hence can be used as a preconditioner for gradient descent in this region (Section 1, Part VIII), where the function is well-approximated by a quadratic.

We demonstrate this phenomenon formally by analyzing gradient descent in an appropriate norm. Specifically, consider a step of *Newton's method*, where we take a step

$$x' \leftarrow \text{argmin}_x \left\{ \varphi(x) + \langle \nabla\varphi(x), x' - x\rangle + \frac{1}{2\eta} \|x' - x\|_{\nabla^2\varphi(x)}^2 \right\},$$

where we make the tautological assumption that x' stays in $\mathcal{E}_x(1)$ for now, such that $\nabla^2\varphi(x') \approx \nabla^2\varphi(x)$ so the approximation above is a reasonable model for how $\varphi(x')$ behaves, by a second-order Taylor expansion. By directly computing the minimizer, this suggests using

$$x' \leftarrow x - \eta \left(\nabla^2\varphi(x)\right)^\dagger \nabla\varphi(x) \tag{5}$$

as our update. Now to fulfill our earlier assumption that $x' \in \mathcal{E}_x(1)$, we need $\|x' - x\|_x^2$ to be small, which by a direct computation (up to rescaling by η factors) is:

$$\|\nabla\varphi(x)\|_{x,*}^2 = \left(\nabla^2\varphi(x)\right)^\dagger [\nabla\varphi(x), \nabla\varphi(x)]\,. \tag{6}$$

We call the quantity in (6) the *Newton decrement*, and it often serves as a potential function for analyzing the progress made by Newton's method (5). We first show the aforementioned intuition regarding stability of $\nabla^2\varphi$ along (5) holds true, if the initial Newton decrement is small.

Lemma 7. *Let φ be a self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$. If $x \in \mathcal{X}$ has $\|\nabla\varphi(x)\|_{x,*} \le \Delta$, then letting x' be defined as in (5), if $\eta\Delta < 1$,*

$$(1 - \eta\Delta)^2\, \mathbf{H}(x) \preceq \mathbf{H}(x_\lambda) \preceq \frac{1}{(1 - \eta\Delta)^2}\mathbf{H}(x),$$
$$\textit{where } x_\lambda := (1-\lambda)x + \lambda x' \textit{ for all } \lambda \in [0,1].$$

Proof. This immediately follows from Lemma 7 with $y \leftarrow x_\lambda$, using

$$\|x_\lambda - x\|_x \le \eta \|\nabla\varphi(x)\|_{x,*} \le \eta\Delta.$$

□

Note that for constant η, Lemma 7 shows that the condition number of $\mathbf{H}(x_\lambda)$ relative to $\mathbf{H}(x)$, for x_λ on the line along the Newton's method update, is $1 + O(\Delta)$. This suggests that in principle (e.g. as Lemma 1, Part VIII would suggest), we should be able to decrease our current distance to the optimizer by a multiplicative $O(\Delta)$ factor, which improves when Δ is already small. We show this is indeed the case for the Newton decrement, in the following more general analysis.

Lemma 8. *Let $\varphi : \mathcal{X} \to \mathbb{R}$ be twice-differentiable, where $\mathcal{X} \subseteq \mathbb{R}^d$ is convex. Let $\mathbf{H} \in \mathbb{S}_{\succeq \mathbf{0}}^{d\times d}$ and $x \in \mathbb{R}^d$, and suppose $\nabla\varphi(y) \in \mathrm{Span}(\mathbf{H})$ for all $y \in \mathcal{X}$. Defining $x_\lambda := x - \lambda\eta\mathbf{H}^\dagger\nabla\varphi(x)$ for all $\lambda \in [0,1]$ and some $\eta > 0$, suppose that for $0 < \mu \le L$, we have*

$$x_\lambda \in \mathcal{X},\ \mu\mathbf{H} \preceq \nabla^2\varphi(x_\lambda) \preceq L\mathbf{H},\ \textit{for all } \lambda \in [0,1]. \tag{7}$$

Then, for $x' := x - \eta\mathbf{H}^\dagger\nabla\varphi(x)$,

$$\|\nabla\varphi(x')\|_{\mathbf{H}^\dagger} \le \max\left(|1 - \eta\mu|, |1 - \eta L|\right) \|\nabla\varphi(x)\|_{\mathbf{H}^\dagger}$$

Proof. By the fundamental theorem of calculus,

$$\nabla\varphi(x') - \nabla\varphi(x) = \int_0^1 \nabla^2\varphi(x_\lambda)(x_1 - x_0)\mathrm{d}\lambda = -\eta \int_0^1 \nabla^2\varphi(x_\lambda)\mathbf{H}^\dagger\nabla\varphi(x)\mathrm{d}\lambda.$$

Next, by rearranging the above display and left-multiplying by $\mathbf{H}^{\frac{\dagger}{2}}$, we have

$$\mathbf{H}^{\frac{\dagger}{2}}\nabla\varphi(x') = \left(\mathbf{I}_d - \eta\int_0^1 \mathbf{H}^{\frac{\dagger}{2}}\nabla^2\varphi(x_\lambda)\mathbf{H}^{\frac{\dagger}{2}}\mathrm{d}\lambda\right)\mathbf{H}^{\frac{\dagger}{2}}\nabla\varphi(x).$$

We further have, by using (7) and letting $\mathbf{\Pi}_{\mathrm{Span}(\mathbf{H})}$ be the projection matrix onto $\mathrm{Span}(\mathbf{H})$,

$$\eta\mu\mathbf{\Pi}_{\mathrm{Span}(\mathbf{H})} \preceq \eta\int_0^1 \mathbf{H}^{\frac{\dagger}{2}}\nabla^2\varphi(x_\lambda)\mathbf{H}^{\frac{\dagger}{2}}\mathrm{d}\lambda \preceq \eta L\mathbf{\Pi}_{\mathrm{Span}(\mathbf{H})}$$
$$\implies \left\|\mathbf{\Pi}_{\mathrm{Span}(\mathbf{H})} - \eta\int_0^1 \mathbf{H}^{\frac{\dagger}{2}}\nabla^2\varphi(x_\lambda)\mathbf{H}^{\frac{\dagger}{2}}\mathrm{d}\lambda\right\|_{\mathrm{op}} \le \max\left(|1-\eta\mu|, |1-\eta L|\right).$$

The conclusion follows by combining the above two displays and taking ℓ_2 norms, where we use $\nabla\varphi(x) \in \mathrm{Span}(\mathbf{H})$, and $\|\mathbf{H}^{\frac{\dagger}{2}}\nabla\varphi(x)\|_2 = \|\nabla\varphi(x)\|_{x,*}$, $\|\mathbf{H}^{\frac{\dagger}{2}}\nabla\varphi(x')\|_2 = \|\nabla\varphi(x')\|_{x,*}$. $\square$

By combining Lemmas 7 and 8, we finally give our analysis of Newton's method (6).

Proposition 1. *Let φ be a ν-self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$ and for $\nu > 0$. If $x \in \mathcal{X}$ has $\|\nabla\varphi(x)\|_{x,*} \le \Delta < \frac{1}{5}$, then letting x' be defined as in (5) with $\eta = 1$, we have*

$$\|\nabla\varphi(x')\|_{x',*} \le 4\Delta^2.$$

Proof. Let $\mathbf{H} \leftarrow \nabla^2\varphi(x)$ and $\eta \leftarrow 1$ in the context of Lemma 8, which is valid because ν-self-concordance implies $\nabla\varphi(x) \in \mathrm{Span}(\mathbf{H}(x))$, and $\mathrm{Span}(\mathbf{H}(\cdot))$ does not change within the Dikin ellipse $\mathcal{E}_x(1)$ due to Lemma 3. Lemma 7 then shows that (7) holds with $L = (1-\Delta)^{-2}$ and $\mu = (1-\Delta)^2$. Therefore, Lemma 3 implies

$$\|\nabla\varphi(x')\|_{x,*} \le \frac{2\Delta - \Delta^2}{(1-\Delta)^2}\|\nabla\varphi(x)\|_{x,*}.$$

The claim follows from $\mathbf{H}(x')^\dagger \preceq (1-\Delta)^{-2}\mathbf{H}(x)$ and $(2\Delta-\Delta^2)(1-\Delta)^{-3} \le 4\Delta$ for $\Delta \in (0, \frac{1}{5})$. $\square$

Proposition 1 shows that the rate of improvement of the Newton decrement potential, when iterating (5) for self-concordant φ, is proportional to the initial Newton decrement $\|\nabla\varphi(x)\|_{x,*}$ itself, if it was already small. This rate of convergence is sometimes called *quadratic* by the continuous optimization community, similar to the discussion of linear convergence rates in Section 4, Part II.

Remark 1. *Proposition 1 gives one way to use stability of the Hessian of a convex function to bound the progress of Newton's method (5). Beyond the multiplicative stability afforded by self-concordance via Lemma 7, another common assumption is that the Hessian of our objective is Lipschitz (analogous to Lipschitzness of the gradient, i.e. Definition 3, Part II), which is an additive stability assumption. A natural algorithm called the* cubic-regularized Newton's method*, which builds upon our smooth gradient descent intuition from Section 3, Part III, and uses a third-order regularization of a second-order Taylor expansion, was analyzed in [NP06, Nes08]. These results were subsequently generalized to Lipschitzness of the p^{th}-order derivative [Nes21], and accelerated to obtain near-optimal dependence on the iteration count in the convergence rate [MS13, GDG+19]. Finally, all extraneous logarithmic factors were removed in the recent works [CHJ+22, KG22]. For a more detailed exposition on these and related results, we refer the reader to Chapter 11, [Sid23].*

3 Interior-point methods

In this section, we give a simple algorithmic framework for minimizing linear functions $c^\top x$ over an open convex set $\mathcal{X} \subseteq \mathbb{R}^d$, assuming that $\mathcal{X}$ admits a self-concordant barrier φ. Our strategy is to consider a family of regularized objectives and their minimizers,[4]

$$\varphi_{c,t}(x) := \frac{1}{t}c^\top x + \varphi(x),\ x^\star_{c,t} := \mathrm{argmin}_{x\in\mathcal{X}}\left\{\varphi_{c,t}(x)\right\},\ \text{for } t \ge 0. \tag{8}$$

[4] Although $\mathcal{X}$ is open, because φ is a barrier for $\mathcal{X}$, we can restrict our domain of consideration to a closed subset of $\mathcal{X}$, so the minimizer exists from compactness.

Note that as $t \to 0$, $x^\star_{c,t}$ approaches the desired minimizer of $c^\top x$ over $\mathcal{X}$. On the other hand, as $t \to \infty$, $x^\star_{c,t}$ approaches the minimizer of φ. The set $\{x^\star_{c,t}\}_{t \geq 0}$ is then an interpolation between these two extremes, and is called the *central path*, see e.g. [Gon92] for more on this perspective.

The key idea of path following interior-point methods is to alternate steps which track the central path closely via Newton updates, and steps which decrement the t parameter to advance along the central path. Specifically, we implement this by taking a point x which is close to $x^\star_{c,t}$ for a given value of t (as measured by its Newton decrement), decreasing t (which potentially increases the Newton decrement due to the changed objective), and then taking a step of Newton's method to improve the Newton decrement. As we will see, the maximum amount we can change t by per iteration is closely related to the ν parameter in Definition 4, as shown in the following.

Theorem 1 (Path following interior-point method). *Let φ be a ν-self-concordant barrier for convex, open $\mathcal{X} \subseteq \mathbb{R}^d$ where $\nu \geq 1$.*[5] *Suppose that we have $x_0 \in \mathcal{X}$, $t_0 \in \mathbb{R}_{>0}$ such that $\frac{1}{t_0}\|c\|_{x_0,*} + \|\nabla\varphi(x_0)\|_{x_0,*} \leq \frac{1}{16}$. Then for any $\epsilon > 0$, we can compute a point $\hat{x}$ such that*

$$\|c + \epsilon\nabla\varphi(\hat{x})\|_{\hat{x}} \leq \epsilon,$$

using T evaluations to $\nabla\varphi$ and $\nabla^2\varphi$, and T linear system solves in $d \times d$ matrices, for

$$T = O\left(\sqrt{\nu}\log\frac{t_0}{\epsilon}\right).$$

Proof. We define a sequence $\{x_k, t_k\}_{k\geq 0}$, where we iteratively define $t_{k+1} \leftarrow (1 - \frac{1}{48\sqrt{\nu}})t_k$ until we reach $t_k = \epsilon$. In each iteration $k+1$, we then take one Newton step from x_k using the self-concordant barrier $\varphi_{c,t_{k+1}}$ to obtain x_{k+1}, maintaining the invariant that in all iterations k,

$$\left\|\nabla\varphi_{c,t_k}(x_k)\right\|_{x_k,*} \leq \frac{1}{16}.$$

This invariant holds at initialization, i.e. $k = 0$, by assumption. We proceed by induction: supposing the above invariant holds in iteration k, by the triangle inequality,

$$\begin{aligned}
\left\|\nabla\varphi_{c,t_{k+1}}(x_k)\right\|_{x_k,*} &= \left\|\frac{1}{t_{k+1}}c + \nabla\varphi(x_k)\right\|_{x_k,*} \\
&\leq \frac{1}{1 - \frac{1}{48\sqrt{\nu}}}\left\|\frac{1}{t_k}c + \nabla\varphi(x_k)\right\|_{x_k,*} + \frac{\frac{1}{48\sqrt{\nu}}}{1 - \frac{1}{48\sqrt{\nu}}}\left\|\nabla\varphi(x_k)\right\|_{x_k,*} \\
&\leq \frac{1}{1 - \frac{1}{48\sqrt{\nu}}} \cdot \frac{1}{16} + \frac{1}{48 - \frac{1}{\sqrt{\nu}}} \leq \frac{1}{8}.
\end{aligned}$$

Therefore, taking one Newton iteration as in Proposition 1 to define x_{k+1} yields the invariant in iteration $k+1$. The conclusion follows because we only need $O(\sqrt{\nu}\log\frac{t_0}{\epsilon})$ iterations until $t_k = \epsilon$, and then we can multiply the definition of the Newton decrement by ϵ to obtain the claim. $\square$

We comment on the initialization and termination conditions (i.e. choices of x_0, t_0, ϵ) in the special case of linear programming in the following Section 4, but note that in many applications, $\frac{t_0}{\epsilon}$ is polynomially-bounded in relevant problem parameters, and a corresponding feasible x_0 can be found in polynomial time. Moreover, at least three different barrier functions in the literature [NN94, Hil14, Fox15, BE19] have been constructed for arbitrary open convex sets $\mathcal{X} \subseteq \mathbb{R}^d$ achieving $\nu = O(d)$.[6] In principle, this shows that linear optimization over any convex set in $\mathbb{R}^d$ can be performed in $\approx \sqrt{d}$ iterations, each assuming access to the gradient and Hessian of an appropriate barrier function. In fact, because optimization of arbitrary (potentially nonlinear) functions f can be phrased as linear optimization over the epigraph $\mathrm{epi}(f)$, i.e. minimize t such that $(x, t) \in \mathrm{epi}(f)$, this implies a similar result for nonlinear convex optimization. Of course, we have not discussed the computational complexity of accessing these $O(d)$-self concordant barriers, which is not known to be performable exactly in polynomial time for any of the aforementioned constructions. Nonetheless, we describe in Section 4 how these ideas and more have led to a revolution in the theoretical complexity of linear programming over the last decade.

[5] This assumption is without loss of generality, since if $\nu < 1$, φ is also a 1-self-concordant barrier, and running this proof with $\nu \leftarrow 1$ instead uses only a logarithmic number of iterations.

[6] Two of these have since been improved to satisfy $\nu = d$, with no additional constant [LY21, Che23]. This is optimal and the hypercube is a tight instance, as shown in Proposition 2.3.6, [NN94].

Remark 2. *The path following interior-point method described in Theorem 1 is sometimes called a short step interior-point method, because it chooses a small update to the t parameter in each iteration to maintain approximate centrality (i.e. a Newton decrement of constant size). There are other frameworks for designing interior-point methods which can perform better in practice, however, by exploiting the fact that the worst-case analysis of how approximate centrality measures change need not apply in every iteration, so one can sometimes make more progress by using a more adaptive algorithm. Examples include long step, predictor-corrector, and potential reduction interior-point methods. For more on these frameworks, we refer the reader to [NN94, Ye98].*

4 Linear programming

In this section, we provide a short discussion on how the ideas in Section 3 have improved the runtime of solving linear programs in a constraint matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$, with $n \geq d$, over the past decade. We note that all algorithms in this section run in weakly-polynomial time, i.e. they incur logarithmic dependence on a final precision parameter, rather than returning exact solutions. It is a well-known open problem (see e.g. Smale's 9^{th} problem [Sma98]) to determine whether general linear programming is solvable in strongly-polynomial time. For an exposition on the state-of-the-art exact solvers and the parameters their runtimes depend on, we defer to [DNV20]. For the remainder of the section, we focus primarily on the following linear program:

$$\min_{\substack{x\in\mathbb{R}^d\\ \mathbf{A}x\leq b}} c^\top x. \tag{9}$$

We refer to the optimal argument in (9) by $x^\star$.

Initialization and termination. In the linear programming setting, where φ is defined as in (3) (and we follow the notation in (4), (8)), we first observe that it suffices to approximate $x^\star_{c,t}$ where t is not too small in order to obtain an approximately-optimal solution. Specifically, suppose we wish for Δ additive suboptimality to (9). We claim it suffices to approximate $x^\star_{c,t}$ for $t = \frac{\Delta}{2n}$. To see this, for the stated value of t, the optimality condition on $x^\star_{c,t}$ shows that

$$\frac{c}{t} = -\nabla\varphi(x^\star_{c,t}) = -\mathbf{A}^\top \mathbf{S}^{-1}_{x^\star_{c,t}} \mathbb{1}_n$$
$$\implies c^\top\left(x^\star_{c,t} - x^\star\right) = t\left\langle \mathbf{S}^{-1}_{x^\star_{c,t}}\mathbb{1}_n, \mathbf{A}(x^\star - x^\star_{c,t})\right\rangle = t\left\langle \mathbf{S}^{-1}_{x^\star_{c,t}}\mathbb{1}_n, s_{x^\star_{c,t}} - s_{x^\star}\right\rangle \leq tn \leq \frac{\Delta}{2},$$

where we used the calculation (4), and the fact that all slack variables are nonnegative in the first inequality. Moreover, for the above value of t, it suffices to obtain an x whose Newton decrement is bounded by a polynomial in Δ, t, and n, via directly bounding the suboptimality gap using local strong convexity and smoothness (this is described in more detail in Chapter 12, [Sid23]).

Regarding initialization, in many combinatorial linear programs one can find an initial feasible point $\hat{x}_0$; more generally this can be achieved by reparameterizing the problem with additional dimensions in a way which does not significantly affect the objective value. Next, the key observation is that $\hat{x}_0$ is the optimal point on the central path for a different cost function, $\hat{c} := -\nabla\varphi(\hat{x}_0)$, because

$$\hat{x}_0 = \text{argmin}_{\substack{x\in\mathbb{R}^d\\ \mathbf{A}x\leq b}} \left\{\langle \hat{c}, x\rangle + \varphi(x_0)\right\}$$

by the first-order optimality condition. Therefore, we can run the interior-point method in Theorem 1 in reverse starting from $t = 1$, until t is a sufficiently large value that we can switch the cost function from $\hat{c}$ to c while negligibly affecting the centrality parameter (Newton decrement).

Altogether, putting together Theorem 1 with the strategies described above gives an algorithm which requires $\approx \sqrt{n}$ iterations, due to the self-concordance parameter bound in Lemma 6. Moreover, each iteration of the interior-point method computes slack variables $\mathbf{S}_x$ and performs a single Newton step using the gradient and Hessian calculations in (4). The naïve cost of implementing each iteration is dominated by computing the Hessian, which requires $\approx nd^{\omega-1}$ time.

Reducing the iteration count. The first recent progress made towards significantly improving the framework discussed previously was achieved by [LS14] (and subsequently simplified and improved in [LS19]), who showed that there is an efficiently computable barrier for polytopes with self-concordance parameter $\approx d$, up to logarithmic factors. Recall from the discussion following Theorem 1 that such a result was known to be achievable for arbitrary convex sets, but the corresponding barrier may not be efficiently computable. The [LS14] barrier was based on an ℓ_p generalization of the leverage scores introduced in Section 3, Part VIII, which are used to reweight the constraints in the uniformly-weighted barrier in Lemma 6. These reweighting coefficients can be interpreted as local importance scores on the constraints, and are based on ℓ_p generalizations of leverage scores called Lewis weights [Lew78], which were introduced to the theoretical computer science community by [CP15]. Importantly, the runtime of each iteration of the [LS14] interior-point method is also dominated by computation and inversion of an appropriate Hessian matrix. This result generically improved the iteration complexity of linear programming from $\approx \sqrt{n}$ to $\approx \sqrt{d}$, which can be a significant savings when there are many more constraints than variables.

Reducing the cost per iteration. The next sequence of developments in improving the runtime of linear programming was based on an idea from [Vai89], who noticed that because all of the Newton's method steps were based on closely-related matrices of the form $\mathbf{A}^\top \mathbf{S}_k^{-2} \mathbf{A}$ for a diagonal slack matrix $\mathbf{S}_k$ computed in each iteration, as long as $\mathbf{S}_k$ does not multiplicatively change significantly over the course of an iteration, the previous inverse can still be used as a preconditioner in the next iteration. Moreover, even if a few slack variables change by a large amount in a given iteration, low-rank update formulas such as the Sherman-Morrison-Woodbury identity (a generalization of Eq. (16), Part VIII) can be used to speed up the cost of recomputing a new inverse matrix. The key technical difficulty in executing this plan is to carefully trade off how often these large slack moves occur, how to detect the large changes, and how often the inverse is partially recomputed. These ideas were refined in [LS15, CLS19], derandomized in [vdB20], and slightly improved in [JSWZ21] in certain regimes of the matrix multiplication exponent. As a result of these efforts, we now know how to solve linear programs in time $\approx \max(n, d)^\omega$ for any value of $\omega > 2 + \frac{1}{18}$, matching the cost of square matrix multiplication up to polylogarithmic overhead.

The frontier. The current state-of-the-art linear programming solver in theory, for general linear programs with $n \gg d$, is due to [vdBLL$^+$21] (improving upon [vdBLSS20]), which achieved a runtime of $\approx nd + d^{2.5}$. To see why this runtime is surprising, the mere cost of computing a single matrix-vector product through $\mathbf{A}$ (let alone matrix-matrix products or matrix inversions) in each of $\sqrt{d}$ iterations is at least $nd^{1.5}$, which is already larger than the stated runtime if implemented naïvely. This runtime bottleneck was sidestepped by using a careful combination of data structures, including a heavy-hitters data structure which detects the aforementioned large coordinate moves requiring a partial recomputation of an inverse matrix for Newton's method. Finally, we mention that similar techniques have since been used to speed up the state-of-the-art in related problems solvable using interior-point methods, such as structured empirical risk minimization [LSZ19] and semidefinite programming [JKL$^+$20, HJS$^+$22]. For a detailed exposition on these and other advanced techniques in algebraic methods for algorithm design, see [vdB22].

Source material

Portions of this lecture are based on reference material in [NN94, LV23, Sid23], as well as the author's own experience working in the field.

References

[BE19] Sébastien Bubeck and Ronen Eldan. The entropic barrier: Exponential families, log-concave geometry, and self-concordance. *Math. Oper. Res.*, 44(1):264–276, 2019.

[Che23] Sinho Chewi. The entropic barrier is n-self-concordant. pages 209–222, 2023.

[CHJ+22] Yair Carmon, Danielle Hausler, Arun Jambulapati, Yujia Jin, and Aaron Sidford. Optimal and adaptive monteiro-svaiter acceleration. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, 2022.

[CLS19] Michael B. Cohen, Yin Tat Lee, and Zhao Song. Solving linear programs in the current matrix multiplication time. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*, pages 938–942. ACM, 2019.

[CP15] Michael B. Cohen and Richard Peng. l_p row sampling by lewis weights. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015*, pages 183–192. ACM, 2015.

[DNV20] Daniel Dadush, Bento Natura, and László A. Végh. Revisiting tardos's framework for linear programming: Faster exact solutions using approximate solvers. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020*, pages 931–942. IEEE, 2020.

[Fox15] Daniel F. J. Fox. A schwarz lemma for kähler affine metrics and the canonical potential of a proper convex cone. *Annali di Matematica Pura ed Applicata*, 194:1–42, 2015.

[GDG+19] Alexander V. Gasnikov, Pavel E. Dvurechensky, Eduard Gorbunov, Evgeniya A. Vorontsova, Daniil Selikhanovych, César A. Uribe, Bo Jiang, Haoyue Wang, Shuzhong Zhang, Sébastien Bubeck, Qijia Jiang, Yin Tat Lee, Yuanzhi Li, and Aaron Sidford. Near optimal methods for minimizing convex functions with lipschitz p-th derivatives. In *Conference on Learning Theory, COLT 2019*, volume 99 of *Proceedings of Machine Learning Research*, pages 1392–1393. PMLR, 2019.

[Gle84] James Gleick. Breakthrough in problem solving. *The New York Times*, 1984.

[Gon92] Clóvis C. Gonzaga. Path-following methods for linear programming. *SIAM Rev.*, 34(2):167–224, 1992.

[Hil14] Roland Hildebrand. Canonical barriers on convex cones. *Math. Oper. Res.*, 39(3):841–850, 2014.

[HJS+22] Baihe Huang, Shunhua Jiang, Zhao Song, Runzhou Tao, and Ruizhe Zhang. Solving SDP faster: A robust IPM framework and efficient implementation. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022*, pages 233–244. IEEE, 2022.

[JKL+20] Haotian Jiang, Tarun Kathuria, Yin Tat Lee, Swati Padmanabhan, and Zhao Song. A faster interior point method for semidefinite programming. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020*, pages 910–918. IEEE, 2020.

[JSWZ21] Shunhua Jiang, Zhao Song, Omri Weinstein, and Hengjie Zhang. A faster algorithm for solving general lps. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, 2021*, pages 823–832. ACM, 2021.

[Kar84] Narendra Karmarkar. A new polynomial-time algorithm for linear programming. *Comb.*, 4(4):373–396, 1984.

[KG22] Dmitry Kovalev and Alexander V. Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*, 2022.

[Lew78] D. Lewis. Finite dimensional subspaces of ℓ_p. *Studia Mathematica*, 63(2):207–212, 1978.

[LS14] Yin Tat Lee and Aaron Sidford. Path finding methods for linear programming: Solving linear programs in $\widetilde{O}(\sqrt{\text{rank}})$ iterations and faster algorithms for maximum flow. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014*, pages 424–433. IEEE Computer Society, 2014.

[LS15] Yin Tat Lee and Aaron Sidford. Efficient inverse maintenance and faster algorithms for linear programming. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015*, pages 230–249. IEEE Computer Society, 2015.

[LS19] Yin Tat Lee and Aaron Sidford. Solving linear programs with sqrt(rank) linear system solves. *CoRR*, abs/1910.08033, 2019.

[LSZ19] Yin Tat Lee, Zhao Song, and Qiuyi Zhang. Solving empirical risk minimization in the current matrix multiplication time. In *Conference on Learning Theory, COLT 2019*, volume 99, pages 2140–2157. PMLR, 2019.

[LV23] Yin Tat Lee and Santosh Vempala. *Techniques in Optimization and Sampling.* 2023.

[LY21] Yin Tat Lee and Man-Chung Yue. Universal barrier is n-self-concordant. *Math. Oper. Res.*, 46(3):1129–1148, 2021.

[MS13] Renato D. C. Monteiro and Benar Fux Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM J. Optim.*, 23(2):1092–1125, 2013.

[Nem04] Arkadi Nemirovski. *Lecture Notes on Interior Point Polynomial Methods in Convex Programming.* 2004.

[Nes08] Yurii E. Nesterov. Accelerating the cubic regularization of newton's method on convex problems. *Math. Program.*, 112(1):159–181, 2008.

[Nes21] Yurii E. Nesterov. Implementable tensor methods in unconstrained convex optimization. *Math. Program.*, 186(1):157–183, 2021.

[NN94] Yurii Nesterov and Arkadi Nemirovski. *Interior-Point Polynomial Algorithms in Convex Programming.* Society for Industrial and Applied Mathematics, 1994.

[NP06] Yurii E. Nesterov and Boris T. Polyak. Cubic regularization of newton method and its global performance. *Math. Program.*, 108(1):177–205, 2006.

[Sid23] Aaron Sidford. *Optimization Algorithms.* 2023.

[Sma98] Stephen Smale. Mathematical problems for the next century. *The Mathematical Intelligencer*, 20:7–15, 1998.

[Vai89] Pravin M. Vaidya. Speeding-up linear programming using fast matrix multiplication (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, 1989*, pages 332–337. IEEE Computer Society, 1989.

[vdB20] Jan van den Brand. A deterministic linear program solver in current matrix multiplication time. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020*, pages 259–278. SIAM, 2020.

[vdB22] Jan van den Brand. *Dynamic Algebraic Algorithms Lecture Notes.* 2022.

[vdBLL+21] Jan van den Brand, Yin Tat Lee, Yang P. Liu, Thatchaphol Saranurak, Aaron Sidford, Zhao Song, and Di Wang. Minimum cost flows, mdps, and ℓ_1-regression in nearly linear time for dense instances. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, 2021*, pages 859–869. ACM, 2021.

[vdBLSS20] Jan van den Brand, Yin Tat Lee, Aaron Sidford, and Zhao Song. Solving tall dense linear programs in nearly linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020*, pages 775–788. ACM, 2020.

[Ye98] Yinyu Ye. *Interior point algorithms - theory and analysis.* Wiley-Interscience series in discrete mathematics and optimization. Wiley, 1998.