Trust Region Interior Point Methods: Optimal ℓ_2- and Faster Wide-Neighborhood Path Following

Daniel Dadush
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
dadush@cwi.nl

Haoyuan Ma
University of Bonn
Bonn, Germany
hma@uni-bonn.de

Bento Natura
Columbia University
New York, NY, USA
bn2392@columbia.edu

László A. Végh
University of Bonn
Bonn, Germany
lvegh@uni-bonn.de

Abstract

We present improved running time and iteration complexities of interior point methods for linear programs parametrized by the straight line complexity, i.e., the minimum number of segments of any piecewise linear curve traversing a particular neighborhood of the central path. While the standard measure of progress is the reduction in duality gap, the straight line complexity provides a stronger instance-wise bound, reflecting the combinatorial structure of the problem.

Our first main result focuses on interior point methods that stay in the ℓ_2-neighborhood. We give a much stronger analysis of the trust region interior point method introduced by Lan, Monteiro and Tsuchiya (SIAM J. Optim. 2009), proving that it is approximately instance optimal in this neighborhood. Namely, we show that the iteration complexity of this algorithm is a constant factor of the straight line complexity of the ℓ_2-neighborhood. Further, each iteration can be implemented in current matrix multiplication time.

Our second main result is a wide-neighborhood interior point method whose running time is the wide-neighborhood straight line complexity times current matrix multiplication time, improving in essence a factor n over the algorithm by Allamigeon, Dadush, Loho, Natura, and Végh (SIAM J. Comput. 2025). The algorithm can be seen as a boosted version of the robust interior point methods of Cohen, Lee and Song (JACM 2021) and van den Brand (SODA 2020) that can reduce the gap by a polynomial factor in current matrix multiplication time: our algorithm is also able to traverse any sufficiently straight segment of the central path in current matrix multiplication time, independently of the length of the segment.

A main ingredient in both methods is to solve trust region problems with ℓ_2 and ℓ_∞-constraints, respectively. We develop fast and strongly polynomial algorithms for solving them to high accuracy. In the ℓ_2-setting, this answers an open question by Lan, Monteiro and Tsuchiya.

This is an extended abstract; the full version is available on arXiv.

STOC '26, Salt Lake City, UT, USA

ACM ISBN 979-8-4007-2536-4/2026/06
https://doi.org/10.1145/3798129.3800790

CCS Concepts

• **Theory of computation** → **Linear programming**.

Keywords

linear programming, interior point methods, strongly polynomial, trust region, instance optimality

ACM Reference Format:
Daniel Dadush, Haoyuan Ma, Bento Natura, and László A. Végh. 2026. Trust Region Interior Point Methods: Optimal ℓ_2- and Faster Wide-Neighborhood Path Following. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing (STOC '26), June 22–26, 2026, Salt Lake City, UT, USA*. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3798129.3800790

1 Introduction

Linear programming (LP) is one of the most fundamental problems in optimization and computer science, and the pursuit of efficient LP algorithms has been a major driving force in these disciplines. We write the problem in the standard primal-dual formulation

$$\begin{aligned} \min\ \langle c, x\rangle \qquad & \qquad \max\ \langle b, y\rangle \\ \mathbf{A}x = b \qquad & \qquad \mathbf{A}^\top y + s = c \\ x \geq \mathbf{0}, \qquad & \qquad s \geq \mathbf{0} \end{aligned} \tag{LP}$$

where the input is $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(\mathbf{A}) = m \leq n$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$. We refer to these as primal and dual programs, respectively, and we let $\mathcal{P} := \{x \in \mathbb{R}^n : \mathbf{A}x = b, x \geq \mathbf{0}\}$ and $\mathcal{D} := \{s \in \mathbb{R}^n : \exists y \text{ s.t. } \mathbf{A}^\top y + s = c, s \geq \mathbf{0}\}$ denote the primal and dual feasible regions. *Interior point methods (IPMs)*, first introduced by Karmarkar in 1984 [30], provide a rich family of LP algorithms that are efficient in theory as well as in practice. A popular class is formed by *path-following IPMs*, pioneered by Renegar [52], Megiddo [39], Kojima, Mizuno and Yoshise [32], and Nesterov and Nemirovski [48]. Primal-dual path following methods maintain a pair of strictly feasible primal and dual solutions in every iteration; the main work in each iteration corresponds to solving a linear system of equations. The worst case complexity estimate shows that the optimality gap is reduced by a constant factor in every $O(\sqrt{n})$ iterations.

The iterates are guided by the *central path*: for each value $\mu > 0$, consider the system

$$\begin{aligned} \mathbf{A}x^{\mathrm{cp}}(\mu) = b\,, \quad & x^{\mathrm{cp}}(\mu) \geq \mathbf{0}\,, \\ \mathbf{A}^\top y^{\mathrm{cp}}(\mu) + s^{\mathrm{cp}}(\mu) = c\,, \quad & s^{\mathrm{cp}}(\mu) \geq \mathbf{0}\,, \\ x^{\mathrm{cp}}(\mu)_i s^{\mathrm{cp}}(\mu)_i = \mu \quad & \text{for all } i \in [n]\,. \end{aligned} \tag{CP}$$

This system arises from the optimality characterization of the *logarithmic barrier function* $\min_{x:Ax=b} \langle c, x\rangle - \mu \sum_{i=1}^n \log x_i$. It has a unique solution $(x^{\mathrm{cp}}(\mu), y^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$, and we denote $z^{\mathrm{cp}}(\mu) \coloneqq (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$ and call it the central path point at μ. The duality gap of $z^{\mathrm{cp}}(\mu)$ is $\langle c, x^{\mathrm{cp}}(\mu)\rangle - \langle b, y^{\mathrm{cp}}(\mu)\rangle = \langle x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu)\rangle = n\mu$. The points $\{z^{\mathrm{cp}}(\mu) : \mu > 0\}$ form a smooth algebraic curve in $\mathbb{R}^{2n}$, and for $\mu \searrow 0$, it converges to a pair of primal and dual optimal solutions $(x^\star, s^\star)$ to (LP); for a proof see e.g. [39].

From a practical perspective, according to Gondzio [22], the impressive features of IPM include *"[...] their low-degree polynomial worst-case complexity and an unrivalled ability to deliver optimal solutions in an almost constant number of iterations which depends very little, if at all, on the problem dimension."* According to the documentation of the solver MOSEK [47], the *"interior-point optimizer [...] tends to use between 20 and 100 iterations, almost independently of problem size"*. For the efficient practical performance of the original Karmarkar's algorithm, we refer to [31, 51].

From a theoretical perspective, as a culmination of a long line of research (see Section 1.2), Cohen, Lee, and Song [9] gave a randomized LP algorithm in *current matrix multiplication* time, by amortizing the work of computing subsequent iterates. Let us use ω to denote the matrix multiplication exponent, and α the dual matrix multiplication exponent;[1] for brevity, we denote $\tilde{\omega} \coloneqq \max\left\{\omega, 2 + \frac{1}{6}, 2.5 - \frac{\alpha}{2}\right\}$. For current best values $\omega \le 2.371552$ and $\alpha \ge 0.321334$ [67], we have $\tilde{\omega} = \omega$. Thus, given a pair of solutions near the central path, the algorithm in [9] can reduce the duality gap by a factor ϱ in $n^{\tilde{\omega}+o(1)} \log(\varrho)$ running time. Subsequently, van den Brand [62] gave a deterministic algorithm with the same guarantee, and [29] improved on the second term in the definition of $\tilde{\omega}$ in the randomized setting. We refer to these algorithms as *'robust IPM methods'*; for an accessible tutorial, see [36]. For a rational input with bit-encoding length L, robust IPMs yield $n^{\tilde{\omega}+o(1)}L$ time algorithms for finding exact optimal primal and dual solutions to (LP). These algorithms still use short steps, and the $O(\sqrt{n})$ iteration complexity to reduce the gap by a constant factor remains the best known in the general setting.

However, for a particular instance, much fewer iterations may suffice. As an example, consider the simple LP $\min \langle c, x\rangle$ s.t. $\langle \mathbf{1}_n, x\rangle = 1$, $x \ge 0$, where $c_i = \varepsilon^{i-1}(1-\varepsilon^{n-i})$, $i \in [n]$ for a very small $\varepsilon > 0$. Here, the primal-dual central path comprises n largely linear segments: letting $x_i \coloneqq (\varepsilon^i, \dots, \varepsilon, \mathbf{1}_{n-i})/(n - i + \frac{\varepsilon(1-\varepsilon^i)}{(1-\varepsilon)})$, $s_i \coloneqq c + \varepsilon^i \mathbf{1}_n$ for $i \in \{0, \dots, n-1\}$, and $(x_n, s_n) \coloneqq (e_n, c)$, the i^{th} segment essentially interpolates from (x_{i-1}, s_{i-1}) to (x_i, s_i). Starting from the near-central primal-dual solution (x_0, s_0), the algorithms [9, 62] would require $\tilde{O}(\sqrt{n}\log(1/\varepsilon))$ iterations and a total runtime $n^{\tilde{\omega}+o(1)}\log(1/\varepsilon)$ to reach the gap value ε. Even at this point the solution will be approximately (x_1, s_1), which is very far from the primal-dual optimal solution (e_n, c). Further, while such IPMs converge to an optimal solution, they do not reach an exact one and require a final rounding step when high enough accuracy is reached (in this example, less than ε^n), requiring $\tilde{O}(n^{1.5}\log(1/\varepsilon))$ iterations in total.

Hence, gap reduction may not be the only relevant progress measure for IPMs. The focus of this paper is on IPMs that can achieve *(near) instance optimal guarantees* with much fewer steps by detecting the combinatorial structure of the central path. In the above example, such an IPM would find an exact optimal solution in just $O(n)$ iterations.

Central path neighborhoods. For $z = (x, s) \in \mathcal{P} \times \mathcal{D}$, we define $\overline{\mu}(z) \coloneqq \langle x, s\rangle / n$ and refer to it as the *normalized duality gap* of z. One can define neighborhoods of the central path based on the vector $\frac{xs}{\overline{\mu}(z)} - \mathbf{1}_n$, called the *centrality error*. Throughout, $xs \in \mathbb{R}^n$ denotes the Hadamard product of the two vectors and $\mathbf{1}_n \in \mathbb{R}^n$ is the all ones vector. For $\beta \in (0, 1)$, two commonly used neighborhoods are

$$\mathcal{N}^2(\beta) \coloneqq \left\{z = (x, s) \in \mathcal{P} \times \mathcal{D} \colon \left\| \frac{xs}{\overline{\mu}(z)} - \mathbf{1}_n \right\|_2 \le \beta \right\},$$
$$\mathcal{N}^{-\infty}(\beta) \coloneqq \{z = (x, s) \in \mathcal{P} \times \mathcal{D} \colon xs \ge (1-\beta)\overline{\mu}(z)\mathbf{1}_n\}.$$

$\mathcal{N}^2(\beta)$ is called the ℓ_2*-neighborhood* and $\mathcal{N}^{-\infty}(\beta)$ is called the *wide neighborhood*. Note that $\mathcal{N}^2(\beta) \subseteq \mathcal{N}^{-\infty}(\beta)$, but the latter neighborhood can be significantly larger. We let $\overline{\mathcal{N}}^2(\beta) \coloneqq \mathrm{cl}(\mathcal{N}^2(\beta))$ and $\overline{\mathcal{N}}^{-\infty}(\beta) \coloneqq \mathrm{cl}(\mathcal{N}^{-\infty}(\beta))$ denote the closure of the neighborhoods that includes points $z = (x, s) \in \mathcal{P} \times \mathcal{D}$ with $\overline{\mu}(z) = 0$, i.e., optimal solutions. In general, by a *neighborhood* $\mathcal{N}$ we mean a subset of $\mathcal{P} \times \mathcal{D}$ that contains the central path $\{z^{\mathrm{cp}}(\mu) : \mu > 0\}$, and we let $\overline{\mathcal{N}}$ denote the closure of $\mathcal{N}$.

The ℓ_2-neighborhood is very amenable to theoretical analysis. Many classical IPMs, including the Primal-Dual Predictor-Corrector Method by Mizuno, Todd and Ye [43], stay throughout in the ℓ_2-neighborhood: each step corresponds to a linear segment inside this neighborhood, while the robust IPMs [9, 62] use an intermediate neighborhood that is contained in the wide neighborhood. The wide neighborhood captures the trajectories generated by most IPMs in the literature: it follows from [5] that the trajectories of all algorithms based on a self-concordant barrier function lie in $\mathcal{N}^{-\infty}(1 - 1/(2\nu))$, where ν is the parameter of the barrier.

Straight-line complexity bounds. Allamigeon, Benchimol, Gaubert, and Joswig [3] proposed a systematic way to lower bound the iteration complexity of IPMs. Consider an IPM where each step corresponds to a line segment inside a central path neighborhood $\mathcal{N}$. Then, the following definition provides a natural lower bound on the iteration complexity.

Definition (Straight Line Complexity)**.** Given a central path neighborhood $\mathcal{N}$ and $\mu_0 > \mu_1 \ge 0$, let $\mathrm{SLC}(\mathcal{N}, \mu_1, \mu_0)$ denote the *minimum* number of segments of any piecewise linear curve $\Gamma : [0, 1] \to \overline{\mathcal{N}}$ that satisfies $\overline{\mu}(\Gamma(0)) = \mu_0$ and $\overline{\mu}(\Gamma(1)) = \mu_1$.

Using this framework, they constructed a family of LPs in n variables and $O(n)$ constraints parametrized by a real parameter $t > 0$, such that for any $\beta > 0$, there exists an LP in the family and $t > 0$ with $\mathrm{SLC}(\mathcal{N}^{-\infty}(\beta), 1, t) \ge 2^n - 1$. This result was strengthened by Allamigeon, Gaubert and Vandame [5], who provided a family of combinatorial cubes (a.k.a. Klee–Minty cubes) having $2n$ constraints and n variables, achieving the same result.

Allamigeon, Dadush, Loho, Natura, and Végh [4] complemented the straight line complexity lower bound by an algorithmic result. They developed a *subspace layered least square (SLLS) IPM* that stays in the ℓ_2-neighborhood, while guaranteeing a number of iterations

[1] That is, any two $n \times n$ matrices can be multiplied in $n^{\omega+o(1)}$ time, and any $n \times n$ and $n \times n^\alpha$ matrices can be multiplied in $n^{2+o(1)}$ time.

that matches the straight line complexity of the wide neighborhood up to a polynomial factor.

Theorem 1.1 ([4]). *Let $\theta \in (0,1)$, $\beta \in (0, 1/6]$, and $\mu_0 > \mu_1 \geq 0$. There exists a path following (SLLS) IPM that starting from any $z \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) \leq \mu_0$, finds a solution $z' \in \overline{\mathcal{N}}^2(\beta)$ with $\overline{\mu}(z') \leq \mu_1$ within $O(n^{1.5}\beta^{-1}\log(n/(\beta(1-\theta)))\,\mathrm{SLC}(\mathcal{N}^{-\infty}(\theta), \mu_1, \mu_0))$ iterations. Every iteration corresponds to a line segment in $\mathcal{N}^2(\beta)$, and can be implemented in strongly polynomial time.*

We emphasize that only the existence of the piecewise linear curve is assumed and the algorithm does not have access to it. The key insight is that a long straight segment in the central path neighborhood enforces a certain 'polarization' of the corresponding part of the central path; see Definition 2.6, and the algorithm exploits this property. The IPM in [4] is the key component of the strongly polynomial algorithm for LPs with two nonzeros per column in [13]; see Section 1.2 for more background on 'combinatorial' IPMs and their applications for strongly polynomial computability.

1.1 Our Contributions

Theorem 1.1 can be interpreted as a form of *approximate instance optimality* in the context of path following. Instance optimality is a very strong concept in beyond worst-case complexity, asserting that an algorithm is (up to a constant factor) on any input better than any other algorithm from a given class; see [54, Chapter 3], and we mention some examples in Section 1.2. Given the straight line complexity lower bound, one can define (approximate) instance optimality of path following IPMs in a neighborhood as follows.

Definition 1.2. Let $\mathcal{N}$ be a neighborhood of the central path, and $\mu_0 > \mu_1 \geq 0$ be given. A primal-dual IPM is *$f(n)$-optimal in $\mathcal{N}$* if it takes at most $f(n)\,\mathrm{SLC}(\mathcal{N}, \mu_1, \mu_0)$ many iterations to arrive from any $z \in \mathcal{N}$ with $\overline{\mu}(z) = \mu_0$ to some $z_1 \in \mathcal{N}$ with $\overline{\mu}(z_1) \leq \mu_1$, and all iterates and the line segments between consecutive iterates lie in $\mathcal{N}$. If $f(n) = O(1)$ as a universal constant then we simply say that the algorithm is *near-optimal in $\mathcal{N}$*.

In these terms, the SLLS IPM in [4] is $O\left(n^{1.5}\log\left(n/(1-\theta)\right)\right)$-optimal in $\mathcal{N} = \mathcal{N}^{-\infty}(\theta)$. In fact, the iterates stay in the narrower $\mathcal{N}^2(\beta)$-neighborhood. We improve on this result in two directions, by exhibiting two algorithms, the *ℓ_2-Trust Region IPM* (TR2-IPM), and the *ℓ_∞-Trust Region IPM* (TRW-IPM).

The ℓ_2-Trust Region IPM. Our first main result shows that the ℓ_2-Trust Region IPM, a natural and relatively simple method first proposed by Lan, Monteiro, and Tsuchiya [33], is near-optimal in the ℓ_2-neighborhood. This is the neighborhood used in many classical IPMs, as well as by the SLLS IPM in [4]. The method uses standard corrector $(\Delta x^{\mathrm{c}}, \Delta s^{\mathrm{c}})$ and affine scaling $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ steps (see Section 2), along with trust region steps $(\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ that can traverse long straight segments.

Theorem 1.3. *For any $\bar{\beta} \in (0, 1/128]$, the algorithm TR2-IPM with the parameter choice $\beta = \bar{\beta}/82$ is near-optimal in $\mathcal{N}^2(\bar{\beta})$. Namely, given any $\mu_0 > \mu_1 \geq 0$ and a starting point $z_0 \in \mathcal{N}^2(\bar{\beta})$ with $\overline{\mu}(z_0) \in [\mu_1, \mu_0]$, the TR2-IPM algorithm finds a solution $z_1 \in \overline{\mathcal{N}}^2(\bar{\beta})$ with $\overline{\mu}(z_1) \leq \mu_1$ in $O\left(\mathrm{SLC}(\mathcal{N}^2(\bar{\beta}), \mu_1, \mu_0)\right)$ many iterations. All iterates and the line segments between consecutive iterates stay in $\overline{\mathcal{N}}^2(\bar{\beta})$.*

Further, every iteration of the algorithm can be implemented in deterministic strongly polynomial running time $O(n^3)$, or by an $\tilde{O}(n^\omega)$ randomized algorithm.

Algorithm 1: TR2-IPM

Input : An instance of (LP) with constraint matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathrm{rk}(\mathbf{A}) = m$, and initial iterate $(x^0, s^0) \in \mathcal{P} \times \mathcal{D}$, $\beta \in (0, 1/12)$, $\mu_1 \geq 0$.

Output : $(x, s) \in \overline{\mathcal{N}}^2(\beta)$ satisfying $\overline{\mu}(x, s) \leq \mu_1$

$(x, s) \leftarrow (x^0, s^0)$;
while $\overline{\mu}(x, s) > \mu_1$ **do**
  while $\left\| \frac{xs}{\overline{\mu}(x,s)} - \mathbf{1} \right\|^2 > \beta^2$ **do**
    Compute corrector step $(\Delta x^{\mathrm{c}}, \Delta s^{\mathrm{c}})$ at (x, s) ;
    $(x, s) \leftarrow (x + \Delta x^{\mathrm{c}}, s + \Delta s^{\mathrm{c}})$;
  Compute affine scaling step $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ at (x, s);
  if $\frac{81}{16}\left\| \frac{\Delta x^{\mathrm{a}} \Delta s^{\mathrm{a}}}{\overline{\mu}(x,s)} \right\|^2 \leq 80^2\beta^2$ **then**
    $B \leftarrow \left\{ i \in [n] : \left| \frac{\Delta x_i^{\mathrm{a}}}{x_i} \right| \leq \left| \frac{\Delta s_i^{\mathrm{a}}}{s_i} \right| \right\}$, $N \leftarrow [n] \setminus B$;
    $\gamma \leftarrow 12\beta$;
    Compute $(\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ at (x, s) by solving $\mathrm{TR}_2(B, N, \gamma)$ with $\delta = \frac{1}{64}$;
    $r \leftarrow \left\| \frac{(x+\Delta x^{\mathrm{TR}})(s+\Delta s^{\mathrm{TR}})}{xs} \right\|$, $\alpha^{\mathrm{TR}} \leftarrow 1/(1 + \frac{r}{36\beta})$;
    $(x, s) \leftarrow (x + \alpha^{\mathrm{TR}}\Delta x^{\mathrm{TR}}, s + \alpha^{\mathrm{TR}}\Delta s^{\mathrm{TR}})$;
  else
    $\alpha^{\mathrm{a}} \leftarrow \sup\{\alpha : (x + \alpha\Delta x^{\mathrm{a}}, s + \alpha\Delta s^{\mathrm{a}}) \in \mathcal{N}^2(82\beta)\}$;
    $(x, s) \leftarrow (x + \alpha^{\mathrm{a}}\Delta x^{\mathrm{a}}, s + \alpha^{\mathrm{a}}\Delta s^{\mathrm{a}})$;
return (x, s);

While the algorithm was already introduced in [33], they gave a weaker convergence analysis, showing that the number of iterations is bounded as $\mathrm{poly}(n, \log \bar{\chi}_{\mathbf{A}}^*)$ where $\bar{\chi}_{\mathbf{A}}^*$ is the optimized version of the Dikin–Stuart–Todd condition number of $\mathbf{A}$; see Section 1.2 for the context and motivation. Theorem 1.3 implies a stronger version of this bound. Moreover, while [33] used trust region steps, their algorithm for computing these steps was weakly polynomial. Our work also resolves their open question on the existence of a strongly polynomial trust region step.

Since each iteration of [4] stays inside $\mathcal{N}^2(\beta)$ for $\beta \in (0, 1/6]$, it follows that the iteration complexity of TR2-IPM is asymptotically at least as good as the SLLS IPM. In fact, [4] identifies the trust-region direction as the 'ideal' movement direction along a polarized segment. It develops the subspace LLS step as a 'good enough' approximation of trust region. The paper states the main reason for not using the trust region step as the lack of a strongly polynomial implementation, which we resolve in this paper.

The proof of the approximate instance optimality in Theorem 1.3 has two main ingredients. The first theorem asserts that the TR2-IPM algorithm terminates in $4\,\mathrm{SLC}(\mathcal{N}^2(\beta), \mu_1, \mu_0)$ iterations, and the iterates and line segments generated are guaranteed to stay in a wider $\mathcal{N}^2(82\beta)$-neighborhood.

Theorem 1.4. *Let $\beta \in (0, 1/128]$ and $\mu_0 > \mu_1 \geq 0$. Starting from any point $z_0 \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z_0) \in [\mu_1, \mu_0]$, the TR2-IPM*

algorithm finds a solution $z_1 \in \overline{\mathcal{N}}^2(\beta)$ with $\overline{\mu}(z_1) \le \mu_1$ in at most $4\,\mathrm{SLC}(\mathcal{N}^2(\beta), \mu_1, \mu_0)$ iterations, and all iterates and the line segments between consecutive iterates stay in $\mathcal{N}^2(82\beta)$.

The second ingredient is the dependence of the straight line complexity $\mathrm{SLC}(\mathcal{N}^2(\beta), \mu_1, \mu_0)$ on the parameter β. By definition, it is monotonically decreasing in β. Our next theorem establishes a stable property of ℓ_2-neighborhood in the sense that decreasing the neighborhood parameter increases the straight line complexity by $O(1)$ times a constant power of the factor of decrease in β. This shows that $\mathrm{SLC}(\mathcal{N}^2(\bar{\beta}/82), \mu_1, \mu_0)$ is only a constant factor of $\mathrm{SLC}(\mathcal{N}^2(\bar{\beta}), \mu_1, \mu_0)$, which gives the final iteration complexity of TR2-IPM in Theorem 1.3. Whether wide neighborhood also admits such a stability property remains an interesting open question.

Theorem 1.5. *There exists a universal constant $C \ge 1$ such that for any $0 \le \beta \le \bar{\beta} \le 1/128$, and $0 \le \mu_1 < \mu_0$,*

$$\mathrm{SLC}(\mathcal{N}^2(\beta), \mu_1, \mu_0) \le O(1) \cdot \left(\frac{\bar{\beta}}{\beta}\right)^C \mathrm{SLC}(\mathcal{N}^2(\bar{\beta}), \mu_1, \mu_0)\,.$$

The ℓ_∞-Trust Region IPM. Our second main result gives a wide neighborhood IPM that significantly improves the running time bound in Theorem 1.1. We present an IPM whose running time is bounded by the wide neighborhood straight line complexity times the current matrix multiplication time.

Theorem 1.6. *For any $\theta \in [1/8, 1)$, given $\mu_0 > \mu_1 \ge 0$ and a starting point $z_0 \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_0) \in [\mu_1, \mu_0]$, the algorithm TRW-IPM finds a solution $z_1 \in \overline{\mathcal{N}}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) \le \mu_1$ in randomized running time $n^{\tilde{\omega}+o(1)} \log\left(\frac{1}{1-\theta}\right) \mathrm{SLC}(\mathcal{N}^{-\infty}(\theta), \mu_1, \mu_0)$, or in deterministic strongly polynomial running time*

$$O\left(\left(n^3 + n^{\tilde{\omega}+o(1)} \log\left(\frac{1}{1-\theta}\right)\right) \mathrm{SLC}(\mathcal{N}^{-\infty}(\theta), \mu_1, \mu_0)\right).$$

The algorithm makes calls to the deterministic Robust IPM by van den Brand [62] as black-box, denoted by $\textsc{PathFollow}(x, s, \varrho)$. The new ingredient is the use of trust region step $(\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ which leads to the main improvement that TRW-IPM can traverse in $n^{\tilde{\omega}+o(1)}$ time an *arbitrarily long* segment between μ and μ' of the central path assuming there exists a straight line segment in the wide neighborhood, instead of $n^{\tilde{\omega}+o(1)} \log(\mu/\mu')$ as guaranteed by the robust IPM [62]. The running time of the SLLS IPM in [4] is independent of $\log(\mu/\mu')$, but requires $\tilde{O}(n^{1.5})$ iterations, for a total runtime of $n^{\tilde{\omega}+1.5+o(1)}$. While the short steps in this algorithm could be replaced by a Robust IPM subroutine, it still requires n subspace layered least square steps, each taking $O(n^3)$ deterministically or $n^{\tilde{\omega}}$ by a randomized algorithm. Hence, Theorem 1.6 improves by a factor n over [4].

We also note that the running time bound in Theorem 1.6 directly improves on the bounds of the Robust IPM. In fact, by choosing $\theta = 1 - 1/\mathrm{poly}(n)$, the straight line complexity of any segment between μ and μ' with $\mu/\mu' = \mathrm{poly}(n)$ is $O(1)$. This follows from Lemma 2.2 which shows that any segment of multiplicative length $\gamma \in [0, 1]$ can be traversed by a straight line in $\mathcal{N}^{-\infty}(1 - \gamma(1-\theta))$.

While the above result gives the strong theoretical guarantee that any straight segment of the wide neighborhood can be traversed essentially in current matrix multiplication time, the underlying Robust IPM used in each large iteration still performs

Algorithm 2: TRW-IPM

Input : An instance of (LP) with constraint matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathrm{rk}(\mathbf{A}) = m$, $\theta \in (0,1)$, an initial iterate $(x^0, s^0) \in \mathcal{N}^{-\infty}(\theta)$, and $\mu_1 \ge 0$.

Output : $(x, s) \in \overline{\mathcal{N}}^{-\infty}(\theta)$ satisfying $\overline{\mu}(x, s) \le \mu_1$

$(x, s) \leftarrow (x^0, s^0)$;

$\varrho \leftarrow \frac{(1-\theta)^9}{256n^9}$;

while $\overline{\mu}(x, s) > \mu_1$ **do**

  $(x', s') \leftarrow \textsc{PathFollow}(x, s, \varrho)$;

  $B \leftarrow \left\{i \in [n] : x'_i \ge \frac{(1-\theta)^4}{4n^4} x_i\right\}$, $N \leftarrow [n] \setminus B$;

  $l \leftarrow \frac{(1-\theta)^3}{16n^3}$, $u \leftarrow \frac{n}{1-\theta}$;

  Compute $(\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ at (x, s) by solving $\mathrm{TR}_\infty(B, N, \ell, u)$ with $\delta = \frac{(1-\theta)^4}{16n^4}$;

  $\varepsilon \leftarrow \max\left\{\left\|\mathbf{1}_N + \frac{\Delta x^{\mathrm{TR}}_N}{x_N}\right\|_\infty, \left\|\mathbf{1}_B + \frac{\Delta s^{\mathrm{TR}}_B}{s_B}\right\|_\infty\right\}$;

  $\alpha^{\mathrm{TR}} \leftarrow 1/(1 + 2\varepsilon)$;

  if $\overline{\mu}(x + \alpha^{\mathrm{TR}}\Delta x^{\mathrm{TR}}, s + \alpha^{\mathrm{TR}}\Delta s^{\mathrm{TR}}) \le \varrho\overline{\mu}(x, s)$ **then**

    $(x, s) \leftarrow (x + \alpha^{\mathrm{TR}}\Delta x^{\mathrm{TR}}, s + \alpha^{\mathrm{TR}}\Delta s^{\mathrm{TR}})$;

  else

    $(x, s) \leftarrow \textsc{PathFollow}(x, s, \varrho)$;

return (x, s);

$O(\sqrt{n}\log(n/(1-\theta)))$ short steps. The computational cost of these steps is amortized by using implicit updates and efficient data structures. In contrast, TR2-IPM directly bounds the number of iterations of an IPM, and achieves essentially the best possible complexity in the ℓ_2-neighborhood. The iteration complexity of TR2-IPM might be more explanatory and consistent with the practical performance of IPMs, which are known to take a very small number of iterations [22, 47], each computed by solving a linear system.

Both TR2-IPM and TRW-IPM use strongly polynomial singular value decomposition (SVD). The difference between the deterministic and randomized running times arises from this: while [17] yields a randomized $\tilde{O}(n^\omega)$ subroutine for SVD in strongly polynomial time, it would require $O(n^3)$ deterministic running time. All other steps and subroutines of TR2-IPM and TRW-IPM are deterministic.

Comparison to the Subspace Layered Least Squares IPM. We note that [4] also includes a stronger, amortized version of Theorem 1.1. A more fine-grained view on straight line complexity (SLC) is to define a separate bound for each variable i. The SLC in Theorem 1.1 is between the largest coordinate-wise SLC and the sum of these quantities. Theorem 1.4 in their paper shows that the number of SLLS steps can be bounded by the sum of coordinate-wise SLCs. Our bound in Theorem 1.6 can be shown to be at least as good as this, and can be a factor n better if the SLC is closer to the maximum coordinate-wise SLC.

Solving the Trust Region Problems. The key ingredient of both TR2-IPM and TRW-IPM is to use trust region steps as long steps which can traverse over arbitrary long segments of the central path. Such steps are computed by solving an ℓ_2-trust region problem (see $\mathrm{TR}_2(B, N, \gamma)$) for TR2-IPM and an ℓ_∞-trust region problem

(see $\mathrm{TR}_\infty(B, N, \ell, u)$) for TRW-IPM. Both problems can be easily transformed to the following form. Let $\mathbf{B} \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(\mathbf{B}) = m$, $b \in \mathbb{R}^m$ and $I \cup J = [n]$ be a partition of the index set:

$$\begin{array}{rl} \min \|y_J\|_2 & \\ \|y_I\|_2 \le 1, & \text{(TR-2)} \\ \mathbf{B}y = b, & \end{array} \qquad \begin{array}{rl} \min \|y_J\|_\infty & \\ \|y_I\|_\infty \le 1, & \text{(TR-max)} \\ \mathbf{B}y = b, & \end{array}$$

While the above two problems look similar, there are fundamental differences between them. Besides our IPM trust region steps, they also arise in a broad range of other contexts; for example, as the classical ridge regression model in statistics [26]. In the context of optimization, solving trust region problems arises as direction finding subroutines in trust region methods [11]. Compared to ℓ_2-trust region, ℓ_∞-trust region problems are less understood and considered more challenging.

One cannot expect an exact optimal solution to (TR-2): the optimal solution may not even be rational for rational input.[2] While the optimal solution to (TR-max) is rational, solving it exactly in strongly polynomial time can be shown to imply a strongly polynomial LP algorithm. We define the notions of approximate solutions as follows; this slightly differs in the two cases.

Definition 1.7. Let $\delta \in (0, 1)$. We say that $y \in \mathbb{R}^n$ is a *δ-feasible solution* to (TR-2) if $\|y_I\| \le 1 + \delta$ and $\mathbf{B}y = b$, and *δ-feasible solution* to (TR-max), if $\|y_I\|_\infty \le 1 + \delta$ and $\mathbf{B}y = b$.

Let $\mathrm{OPT}_2 = \mathrm{OPT}_2(\mathbf{B}, b, I, J)$ and $\mathrm{OPT}_M = \mathrm{OPT}_M(\mathbf{B}, b, I, J)$ denote the optimum values of (TR-2) and (TR-max), respectively. We say that $y \in \mathbb{R}^n$ is a *δ-optimal solution* to (TR-2) if it is δ-feasible and satisfies $\|y_J\| \le \mathrm{OPT}_2$. We say that $y \in \mathbb{R}^n$ is a *δ-optimal solution* to (TR-max), if it is δ-feasible and satisfies $\|y_J\|_\infty \le (1 + \delta)\mathrm{OPT}_M$.

The δ-optimality in the ℓ_2-setting is much stronger: we can achieve OPT_2 at a slight violation of the trust region constraint; whereas in the ℓ_∞-setting, the objective value is also slightly suboptimal. Getting the stronger guarantee of achieving OPT_M by at most δ violation on I in strongly polynomial time would again imply solving LP in strongly polynomial time, already in the case $I = \emptyset$. For TR2-IPM, setting $\delta = 1/64$ already suffices, while for TRW-IPM, we need to set $\delta = (1 - \theta)^4/(16n^4)$.

Lan, Monteiro, and Tsuchiya [33] gave a *weakly polynomial* algorithm for solving (TR-2) based on bisection scheme, and left it as an open question to find a strongly polynomial algorithm, i.e. where the number of arithmetic operations is $\mathrm{poly}(n)$. The strongly polynomial solvability of (TR-2) was an important issue as the goal in [33] was to develop an algorithm whose running time is polynomial in n and the logarithm of condition number of $\mathbf{A}$, but independent of b and c. The affine scaling step as well as other IPM step directions, including the layered least square steps in [64] can be obtained by solving linear systems. We resolve this open question in the affirmative.

Theorem 1.8. *For $\delta \in (0, 1)$, there exists an algorithm that finds a δ-optimal solution to* (TR-2) *or certifies infeasibility in strongly polynomial time. The number of arithmetic operations is dominated by $O(\log n + \log\log(|I|/\delta))$ many linear system solves of size m, plus the time to get a $O\left(2^{2n}\right)$ multiplicative approximation of the eigenvalues of a positive semidefinite matrix of size at most n.*

Further, the algorithm can be implemented in randomized running time $O\left(n^{\tilde{\omega}}(\log n + \log\log(|I|/\delta))\right)$ or in deterministic running time $O\left(n^3 + n^{\tilde{\omega}}(\log n + \log\log(|I|/\delta))\right)$.

Moreover, we give a strongly polynomial algorithm for approximately solving the significantly more challenging ℓ_∞-trust region problem, so that every step taken in TRW-IPM can be computed in current matrix multiplication time.

Theorem 1.9. *For $\delta \in (0, 1)$, there exists an algorithm that finds a δ-optimal solution to* (TR-max) *or certifies infeasibility in strongly polynomial time. Further, the algorithm can be implemented in randomized running time $n^{\tilde{\omega}+o(1)}\log(1/\delta)$ or deterministic running time $n^{\tilde{\omega}+o(1)}\log(1/\delta) + O(n^3)$.*

We refer to Section 4 for an overview of both algorithms and the full version for their formal description.

1.2 Related Work

Several IPMs use barrier functions different from the logarithmic barrier function. For example, a path-following IPM based on the volumetric barrier function was introduced by Vaidya [61] for its desirable iteration complexity, despite the higher operation cost per iteration. Central path with weight modification was first used by Madry [38] to solve unit capacity maximum flow and minimum cut problems. Lee and Sidford [34, 35] designed a path-following IPM based on Lewis weights that achieved both $\tilde{O}(\sqrt{\mathrm{rk}(\mathbf{A})}L)$ iteration complexity and $\tilde{O}(1)$ linear system solves per iteration. All these barrier functions belong to a large class called self-concordant barrier functions, for which the notion of central path exists, and so does a theoretical universal barrier function [48]. As mentioned earlier, it follows from [5] that the trajectories of all algorithms based on reasonable self-concordant barrier functions also lie in $\mathcal{N}^{-\infty}(1 - \theta)$ for $\theta = 1/\mathrm{poly}(n)$, and therefore the straight line complexity bounds are applicable. On the other hand, it is currently unknown whether any analogous trust region step for these barrier functions exists.

Despite the empirical evidence suggesting that IPMs in very large neighborhoods of the central path are much more efficient in practice (see [10, 21, 28, 41, 42, 57]), already establishing theoretical running time bounds for wide neighborhood IPMs that match with those of ℓ_2-neighborhood IPMs has been regarded as a genuine challenge; see [53]. Mizuno–Todd–Ye predictor corrector algorithm in the wide neighborhood gives an $O(nL)$ iteration bound [43]. Hung and Ye [27] reduced this iteration bound to $O(n^{\frac{n+1}{2n}}L)$ with high-order correction. To further improve on this, Ai [1] gave a new wide neighborhood IPM achieving $O(\sqrt{n}L)$ bound. Peng, Terlaky and Zhao [50] also proposed an $O(\sqrt{n}\log(n)L)$ algorithm based on self-regular functions. Later, Ai and Zhang [2] devised a wide neighborhood IPM that allows large update with also an $O(\sqrt{n}L)$ iteration bound. Following this long line of research, Theorem 1.6 in this paper can be seen as improving the theoretical running time bound of wide neighborhood IPM closer towards its actual efficient practical performance.

The SLLS IPM algorithm in [4] builds on a long line of work on *combinatorial interior point methods*. In a seminal 1996 paper,

[2] E.g., let $\mathbf{B} = (1, 1, 1)$, $b = (2)$, $I = \{1, 2\}$, $J = \{3\}$.

Vavasis and Ye [64] addressed the problem that affine scaling is not always aggressive enough to quickly traverse "long and straight" segments of the central path. They introduced the *layered least squares (LLS)* predictor step, which uses a combinatorial layering procedure to compute the predictor direction. Unlike most IPMs, this algorithm terminates with an exact optimal solution without any additional rounding step. The total number of iterations can be bounded by $O(n^{3.5}\log \bar{\chi}_{\mathbf{A}})$, where $\bar{\chi}_{\mathbf{A}}$ is the Dikin–Stuart–Todd condition number of the matrix. This result has remarkable implications in the context of strongly polynomial solvability of Linear Programming. This fundamental open question asks for exactly solving (LP) in $\text{poly}(m,n)$ basic arithmetic operations. Tardos [58] gave an algorithm with $\text{poly}(n, \log \Delta_{\mathbf{A}})$ running time, assuming $\mathbf{A}$ has integer entries and $\Delta_{\mathbf{A}}$ is the largest sub-determinant. This yields strongly polynomial algorithms for 'combinatorial LPs', such as network flow or multi-commodity flow problems, or packing and covering problems with small integer coefficients. The LLS IPM strengthens Tardos's result, since $\bar{\chi}_{\mathbf{A}}$ can be bounded by $\Delta_{\mathbf{A}}$; at the same time, this is a purely geometric condition number that does not require integrality and can be much smaller.

Improved LLS methods were given in [40, 44]. Monteiro and Tsuchiya [45] majorized the iteration bound of Mizuno–Todd–Ye predictor-corrector method by that of LLS IPMs. They show that the number of affine scaling steps to reduce the duality gap by a constant factor is asymptotically a $\beta^{-1/2}$-factor of a certain Sonnevend integral as $\beta \to 0$, which is bounded by $\text{poly}(n, \log \bar{\chi}^*_{\mathbf{A}})$, where $\bar{\chi}^*_{\mathbf{A}} := \inf\{\bar{\chi}_{\mathbf{AD}} : \mathbf{D} \succ \mathbf{0} \text{ diagonal }\}$ i.e., the best achievable value of $\bar{\chi}_{\mathbf{A}}$ under column scaling. However, in contrast to many standard IPMs, the aforementioned LLS algorithms were not invariant under rescaling the columns of $\mathbf{A}$. Hence, the above works left the open question of finding a *scaling invariant LLS* algorithm; the running time dependence would thus improve to $\text{poly}(n, \log \bar{\chi}^*_{\mathbf{A}})$. The Trust Region IPM in the focus of our current work was introduced by Lan, Monteiro, and Tsuchiya [33] to tackle this challenge. The Trust Region IPM is scaling invariant, and they showed an $O(n^{3.5}\log \bar{\chi}^*_{\mathbf{A}})$ iteration bound. As mentioned above, the steps themselves were not strongly polynomial. A scaling invariant LLS IPM method with iteration bound $O(n^{2.5}\log(n)\log \bar{\chi}^*_{\mathbf{A}})$ was given by Dadush, Huiberts, Natura and Végh [12]; a key insight is to relate the condition number $\bar{\chi}_{\mathbf{A}}$ to the *circuit imbalance measure*, which is the ratio between the largest and smallest absolute value entries of minimal linear dependencies in $\ker(\mathbf{A})$; see [19].

The SLLS IPM—and consequently TR2-IPM—subsumes the above strongly polynomial results as it is approximately optimal with respect to the straight-line complexity lower bound. It was the key ingredient in the result by Dadush, Koh, Natura, Olver, and Végh [13] that gave the first strongly polynomial algorithm for LPs with two nonzero entries per column. The crux of the result is a strongly polynomial bound on the straight line complexity of such LPs. The TR2-IPM is also applicable to this problem and may ultimately lead to simpler algorithms.

Before [3, 5], the behavior of IPM trajectories in the neighborhood of Klee–Minty cubes has been studied in different contexts. A lower bound $\Omega(n^{1/3})$ has been established for the primal-dual affine scaling IPM and its variants [59, 60]. Very recently, Vladu [65] gave an IPM with a matching iteration bound $O(n^{1/3})$ for a class of M-matrix-based quadratic programs. Deza, Nematollahi, Peyghami and Terlaky [15], and Deza, Nematollahi and Terlaky [16] proved further lower bounds for the number of iterations needed to decrease the gap by a constant factor. In terms of the above framework, an example of a Klee-Minty cube was given in [16] for which $\text{SLC}(\mathcal{N}^2(\beta), 1, 2) = \Omega(\sqrt{n/\log^5(n)})$.

The concept of *instance optimality* was first introduced by Fagin, Lotem, and Naor [20] in the context of database aggregation. Other examples include various sorting algorithms, see [54, Chapter 3] and references within. A recent breakthrough by Haeupler, Hladík, Rozhoň, Tarjan, and Tětek [24] showed that Dijkstra's algorithm with an appropriate heap data structure is universally optimal, i.e., optimal for graphs with a fixed topology. Subsequently, they showed in [25] that bidirectional Dijkstra is instance optimal if outputting the order of vertices by distances is required; see [18] for a recent breakthrough without such a requirement. We note that our notion of instance optimality is restricted to a particular class of algorithms; on the other hand, we consider the very general linear programming problem.

2 From Central Path Geometry to Trust Region

Affine Scaling and Corrector Directions. The search directions commonly used in IPMs at $z = (x, s)$ are obtained as the solution $(\Delta x, \Delta s) \in \ker(\mathbf{A}) \times \text{Im}(\mathbf{A}^\top)$ to the following linear system for some $t \in [0, 1]$:

$$\mathbf{A}\Delta x = \mathbf{0} \tag{1}$$

$$\mathbf{A}^\top \Delta y + \Delta s = \mathbf{0} \tag{2}$$

$$s\Delta x + x\Delta s = t\overline{\mu}(z)\mathbf{1} - xs \tag{3}$$

$t = 1$ is used for *corrector steps*, which gives the *centrality direction* denoted as $\Delta z^{\text{c}} = (\Delta x^{\text{c}}, \Delta s^{\text{c}})$ throughout. $t = 0$ is used for *predictor steps*, which gives the *affine scaling direction* denoted as $\Delta z^{\text{a}} = (\Delta x^{\text{a}}, \Delta s^{\text{a}})$ throughout. The following proposition summarizes the well-known properties of predictor and corrector steps.

Proposition 2.1. *For $\beta \in (0, 1)$, let $z = (x, s) \in \mathcal{N}^2(\beta)$.*

(i) Let Δz^{c} be the centrality direction at z. Then $\overline{\mu}(z + \Delta z^{\text{c}}) = \overline{\mu}(z)$ and $z + \Delta z^{\text{c}} \in \mathcal{N}^2(\frac{\sqrt{2}\beta^2}{4(1-\beta)})$.

(ii) Let $\alpha \in [0, 1]$ and Δz^{a} be the affine scaling direction at z, then $\overline{\mu}(z + \alpha\Delta z^{\text{a}}) = (1 - \alpha)\overline{\mu}(z)$. Moreover, if $z + \alpha\Delta z^{\text{a}} \in \mathcal{N}^2(\beta)$, then $z + \alpha'\Delta z^{\text{a}} \in \mathcal{N}^2(\beta)$ for any $\alpha' \in [0, \alpha]$.

Straight line deviation. For any two points $z, z_1 \in \mathcal{P} \times \mathcal{D}$ with $\overline{\mu}(z_1) \le \overline{\mu}(z)$, we use $[z_1, z]$ to denote the straight line between z_1 and z in $\mathbb{R}^{2n}$, i.e.,

$$[z_1, z] := \{(1 - \lambda)z + \lambda z_1 : \lambda \in [0, 1]\} .$$

The wide neighborhood is big enough to contain $[z_1, z]$ for any $z, z_1 \in \mathcal{N}^{-\infty}(\theta)$. In particular, taking z and z_1 to be any subsequent iterates in short step IPMs in e.g. [9, 43, 62], $[z_1, z]$ counts as 1 straight line complexity in $\mathcal{N}^{-\infty}\left(1 - \frac{\overline{\mu}(z_1)}{\overline{\mu}(z)}(1 - \theta)\right)$.

Lemma 2.2. *For $\theta \in [0, 1)$, let $z = (x, s) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z) = \mu$ and $z_1 = (x_1, s_1) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) = \nu\mu$ where $\nu \in [0, 1]$. Then, $[z_1, z] \subseteq \mathcal{N}^{-\infty}(1 - \nu(1 - \theta))$. If further $z = z^{\text{cp}}(\mu)$ and $z_1 = z^{\text{cp}}(\mu_1)$, then $[z_1, z] \subseteq \mathcal{N}^{-\infty}(1 - \nu)$.*

For points in the narrower ℓ_2-neighborhood, we first have the following proximity result on the distance of both primal and dual variables to the corresponding central path point.

Lemma 2.3 (Gonzaga [23]). *For $\beta \in (0,1)$, let $z = (x,s) \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu$. Then,*

$$\left\| \left(\frac{x - x^{\mathrm{cp}}(\mu)}{x}, \frac{s - s^{\mathrm{cp}}(\mu)}{s} \right) \right\| \le \frac{\beta}{1-\beta}. \tag{4}$$

We further introduce the following measure on how much the entire straight line $[z_1, z]$ deviates from the central path.

Definition 2.4 (Straight line deviation). Given any two points $z = (x,s), z_1 = (x_1, s_1) \in \mathcal{P} \times \mathcal{D}$ with $\overline{\mu}(z) = \mu_0$ and $\overline{\mu}(z_1) = \mu_1 \le \mu_0$, we define the deviation of $[z_1, z]$ from the central path to be

$$\kappa(z_1, z) := \frac{\|(x_1 - x)(s_1 - s)\|}{\left(\sqrt{\mu_0} + \sqrt{\mu_1}\right)^2},$$

and we simply let $\kappa(\mu_1, \mu_0) := \kappa(z^{\mathrm{cp}}(\mu_0), z^{\mathrm{cp}}(\mu_1))$.

The following lemma confirms the geometric interpretation of κ which states that $[z_1, z]$ stays close to the central path iff $\kappa(z_1, z)$ is small. Further, $\kappa(\mu_1, \mu_0)$ is the maximum ℓ_2-centrality error attained by any point on $[z^{\mathrm{cp}}(\mu_1), z^{\mathrm{cp}}(\mu_0)]$.

Lemma 2.5. *For $\beta \in (0,1)$, let $z = (x,s), z_1 = (x_1, s_1) \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu_0$ and $\overline{\mu}(z_1) = \mu_1 \le \mu_0$.*

(i) If $[z_1, z] \subseteq \mathcal{N}^2(\beta)$, then $\kappa(z_1, z) \le 2\beta$.
(ii) Conversely, if $\kappa(z_1, z) \le \beta$, then $[z_1, z] \subseteq \mathcal{N}^2(\beta)$.
(iii) It holds that

$$\kappa(\mu_1, \mu_0) = \sup \left\{ \left\| \frac{\bar{x}\bar{s}}{\overline{\mu}(\bar{x}, \bar{s})} - \mathbf{1}_n \right\| : (\bar{x}, \bar{s}) \in [z^{\mathrm{cp}}(\mu_1), z^{\mathrm{cp}}(\mu_0)] \right\}.$$

Polarization. Both of our algorithms can be understood through the lens of *polarization*. This was formally introduced in [4], but also used implicitly in previous work such as [33, 64]. We assume the existence of a piecewise linear curve of T segments that stays entirely in $\mathcal{N}^{-\infty}(\theta)$. It could be taken as the curve attaining the straight line complexity lower bound, or the trajectory generated by any path following IPMs. Throughout, we focus on a single straight line segment $[z_1, z] \subseteq \mathcal{N}^{-\infty}(\theta)$ between $z = (x,s) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z) = \mu_0$ and $z_1 = (x_1, s_1) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) = \mu_1 < \mu_0$. It was shown in [4] that the existence of such a segment guarantees a partition (B, N) of the variable set $[n]$ such that for $i \in N$, $x_i^{\mathrm{cp}}(\mu)$ scales down approximately linearly with $\mu \in [\mu_1, \mu_0]$, whereas for $i \in B$, $x_i^{\mathrm{cp}}(\mu)$ changes only in a bounded way; the duals $s_i^{\mathrm{cp}}(\mu)$ exhibit the analogous behavior with B and N swapped. This is captured by the following definition.

Definition 2.6 (Central Path Polarization). For $\gamma \in (0,1]$ and $0 \le \mu_1 < \mu_0$, we say that the central path is *γ-polarized* on $[\mu_1, \mu_0]$ with partition (B, N) if for any $\mu \in [\mu_1, \mu_0]$,

$$\gamma x_i^{\mathrm{cp}}(\mu_0) \le x_i^{\mathrm{cp}}(\mu) \le n x_i^{\mathrm{cp}}(\mu_0) \quad \forall i \in B,$$
$$\frac{\mu}{n\mu_0} x_i^{\mathrm{cp}}(\mu_0) \le x_i^{\mathrm{cp}}(\mu) \le \frac{\mu}{\gamma\mu_0} x_i^{\mathrm{cp}}(\mu_0) \quad \forall i \in N.$$

We generalize this notion to any straight line with two endpoints in the wide neighborhood of the central path.

Definition 2.7 (Wide Neighborhood Polarization). For $\gamma \in (0,1]$ and $z = (x,s), z_1 = (x_1, s_1) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) \le \overline{\mu}(z)$, we say the straight line $[z_1, z]$ is γ-polarized if there exists a partition $B \cup N = [n]$ such that for all $z(\mu) \in [z_1, z]$,

$$x_i(\mu) \ge \gamma \cdot x_i \quad \forall i \in B \quad \text{and} \quad s_i(\mu) \ge \gamma \cdot s_i \quad \forall i \in N.$$

The following lemma asserts that if a straight line stays throughout in $\mathcal{N}^{-\infty}(\theta)$, then it is $\frac{(1-\theta)^3}{16n^3}$-polarized for some partition (B, N).

Lemma 2.8. *For $\theta \in [0,1)$, let $z = (x,s), z_1 = (x_1, s_1) \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) < \overline{\mu}(z)$. If the straight line $[z_1, z] \subseteq \mathcal{N}^{-\infty}(\theta)$, then there exists a partition $B \cup N = [n]$ such that $[z_1, z]$ is $\frac{(1-\theta)^3}{16n^3}$-polarized with partition (B, N).*

2.1 Wide Neighborhood Trust Region Step

From here, we can model the polarization in $\mathcal{N}^{-\infty}(\theta)$ by an ℓ_∞-trust region problem ($\mathrm{TR}_\infty(B, N, \ell, u)$). It follows from Lemma 2.8 that by setting $\ell = (1-\theta)^3/(16n^3)$ and $u = n/(1-\theta)$, the 'ideal direction' $\Delta z^{\mathrm{id}} = (\Delta x^{\mathrm{id}}, \Delta s^{\mathrm{id}}) := (x_1 - x, s_1 - s)$ is feasible to the problem.

$$\begin{aligned} \min \ & \left\| \mathbf{1}_N + \frac{\Delta x_N}{x_N} \right\|_\infty \\ \text{s.t. } & \ell \mathbf{1} \le \mathbf{1}_B + \frac{\Delta x_B}{x_B} \le u\mathbf{1} \\ & \mathbf{A}\Delta x = \mathbf{0} \end{aligned} \qquad \begin{aligned} \min \ & \left\| \mathbf{1}_B + \frac{\Delta s_B}{s_B} \right\|_\infty \\ \text{s.t. } & \ell \mathbf{1} \le \mathbf{1}_N + \frac{\Delta s_N}{s_N} \le u\mathbf{1} \\ & \mathbf{A}^\top \Delta y + \Delta s = \mathbf{0} \end{aligned} \tag{$\mathrm{TR}_\infty(B, N, \ell, u)$}$$

We define the trust region direction $\Delta z^{\mathrm{TR}} := (\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ to be the optimal solution to the above problem. The primal trust region step Δx^{TR} is trying to decrease the coordinates in N by the maximum possible amount while only allowing a limited multiplicative change in the coordinates in B. Since the ideal direction provides a sufficiently good solution, with a suitable step length $\alpha^{\mathrm{TR}} \in (0,1]$, a trust region step reaches near the end of the current polarized segment while staying in $\mathcal{N}^{-\infty}(\theta')$ for a larger θ' value.

Lemma 2.9. *For $\theta \in (0,1]$, suppose there exist some $z \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z) = \mu_0$ and $z_1 \in \mathcal{N}^{-\infty}(\theta)$ with $\overline{\mu}(z_1) = \mu_1 \le \frac{(1-\theta)^3}{64n^3}\mu_0$ such that $[z_1, z] \subseteq \mathcal{N}^{-\infty}(\theta)$. Let $\Delta z^{\mathrm{TR}} = (\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ be the trust region direction at z. Then for $\alpha^{\mathrm{TR}} = \frac{1}{1+2\varepsilon} \in [0,1]$ where*

$$\varepsilon := \max \left\{ \left\| \mathbf{1}_N + \frac{\Delta x_N^{\mathrm{TR}}}{x_N} \right\|_\infty, \left\| \mathbf{1}_B + \frac{\Delta s_B^{\mathrm{TR}}}{s_B} \right\|_\infty \right\},$$

$\overline{\mu}(z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}) \le \frac{32n^4}{(1-\theta)^4}\mu_1$ and $z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}} \in \mathcal{N}^{-\infty}\left(1 - \frac{(1-\theta)^5}{72n^4}\right)$.

In TRW-IPM, we use the subroutine PathFollow to decrease the duality gap by a factor $\mathrm{poly}(n/(1-\theta))$ in $n^{\tilde{\omega}+o(1)}$ time by implementing the deterministic Robust IPM. After each call to this subroutine, we guess the partition (B, N), by observing the changes in the variables x_i and s_i during PathFollow. If we are currently on a long polarized segment, there is a simple way to guess the polarized partition (B, N): B is formed by the variables where the multiplicative change in x_i is smaller than the change in s_i; see full version for the formal statement and its proof. We then use a second subroutine (discussed in Section 4) to find $(\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ by approximately solving $\mathrm{TR}_\infty(B, N, \ell, u)$ in $n^{\tilde{\omega}+o(1)}$ time. With the

above step length α^{TR}, a trust region step $z+\alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}$ is guaranteed to take us close to the end of this segment.

2.2 ℓ_2-Polarization and Trust Region

If we now assume that the piecewise linear curve of T segments stays inside the ℓ_2-neighborhood, then we can get stronger ℓ_2-bounds on the change of primal and dual variables on each segment. To show the $O(1)$ instance optimality of TR2-IPM, we need a more refined analysis by separately considering whether each segment is long or short.

Long straight line segment. We say $[z_1, z] \subseteq \mathcal{N}^2(\beta)$ is long if $\mu_1 \le \mu_0/4$. Let $(\Delta x^{\mathrm{id}}, \Delta s^{\mathrm{id}}) := (x_1 - x, s_1 - s)$ be the ideal direction. By Lemma 2.5, $[z_1, z]$ has a bounded deviation as $\kappa(z_1, z) \le 2\beta$. This together with the *residual equation*

$$\left(1+\frac{\Delta x}{x}\right)\left(1+\frac{\Delta s}{s}\right) = \frac{x_1 s_1}{xs} \approx \frac{\mu_1}{\mu_0}\cdot \mathbf{1}_n < \mathbf{1}_n/4 \tag{5}$$

ensures the existence of a *polarized partition* $B \cup N = [n]$ such that by going from (x, s) to (x_1, s_1), the primal variables in N must scale down massively in ℓ_2-norm while those in B barely change, and for the dual variables the roles of B and N are swapped.

Lemma 2.10. *For $\beta \in (0, 1/128]$, suppose there exist some $z \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu_0$ and $z_1 \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z_1) = \mu_1 \le \mu_0/4$ such that $[z_1, z] \subseteq \mathcal{N}^2(\beta)$. Then there exists a partition $B \cup N = [n]$ such that*

(i)

$$\max\left\{\left\|\frac{\Delta x^{\mathrm{id}}_B}{x_B}\right\|, \left\|\frac{\Delta s^{\mathrm{id}}_N}{s_N}\right\|\right\} \le 12\beta(1-\mu_1/\mu_0), \tag{6}$$

(ii)

$$\max\left\{\left\|\mathbf{1}_N + \frac{\Delta x^{\mathrm{id}}_N}{(1-\mu_1/\mu_0)x_N}\right\|, \left\|\mathbf{1}_B + \frac{\Delta s^{\mathrm{id}}_B}{(1-\mu_1/\mu_0)s_B}\right\|\right\} \le 16\beta\mu_1/\mu_0\,. \tag{7}$$

It follows from Lemma 2.10 (i) that $(\Delta x^{\mathrm{id}}, \Delta s^{\mathrm{id}})$ is feasible to the following ℓ_2-trust region problem ($\mathrm{TR}_2(B, N, \gamma)$) with $\gamma = 12\beta$.

$$\begin{aligned} \min \left\|\frac{x_N + \Delta x_N}{x_N}\right\| \quad & \qquad \min \left\|\frac{s_B + \Delta s_B}{s_B}\right\| \\ \text{s.t.} \left\|\frac{\Delta x_B}{x_B}\right\| \le \gamma \quad & \qquad \text{s.t.} \left\|\frac{\Delta s_N}{s_N}\right\| \le \gamma \qquad (\mathrm{TR}_2(B,N,\gamma)) \\ \mathbf{A}\Delta x = \mathbf{0} \quad & \qquad \mathbf{A}^\top \Delta y + \Delta s = \mathbf{0} \end{aligned}$$

Let $\Delta z^{\mathrm{TR}} = (\Delta x^{\mathrm{TR}}, \Delta s^{\mathrm{TR}})$ denote the optimal solution. By definition, Δx^{TR} achieves a maximal multiplicative decrease on the coordinates in N under the condition that the coordinates in B change barely with both measured in ℓ_2-norm. Since the multiplicative decrease attained by $(\Delta x^{\mathrm{id}}, \Delta s^{\mathrm{id}})$ is large enough (Lemma 2.10 (ii)), a trust region step at z is guaranteed to reach the end of the segment $[z_1, z]$ while staying in a larger $\mathcal{N}^2(O(\beta))$ neighborhood. The threshold at $1/4$ was chosen to obtain a suitable constant in $\mathcal{N}^2(O(\beta))$.

Lemma 2.11. *For $\beta \in (0, 1/128]$, suppose there exist some $z \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu_0$ and $z_1 \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z_1) = \mu_1 \le \mu_0/4$ such that $[z_1, z] \subseteq \mathcal{N}^2(\beta)$. Let Δz^{TR} be the trust region direction at z, then there exists a step-length $\alpha^{\mathrm{TR}} \in (0, 1]$ so that $\overline{\mu}(z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}) \le \mu_1$ and $z + \alpha\Delta z^{\mathrm{TR}} \in \mathcal{N}^2(82\beta)$ for all $\alpha \in [0, \alpha^{\mathrm{TR}}]$.*

Short straight line segment. We say $[z_1, z] \subseteq \mathcal{N}^2(\beta)$ is short if $\mu_1 \ge \mu_0/4$. A standard affine scaling step $z + \alpha\Delta z^{\mathrm{a}}$ reduces the duality gap by exactly a factor of $1-\alpha$. Zhao and Stoer [70] and Zhao [69] proved that the number of affine scaling steps required to traverse along a central path segment $[\mu_1, \mu_0]$ in $\mathcal{N}^2(\beta)$ can be upper bounded by a *Sonnevend curvature integral* [56] on the interval $[\mu_1, \mu_0]$ up to a factor of $O(1/\sqrt{\beta})$. Furthermore, when the curvature is uniformly lower bounded by a constant, they showed that the number of steps is lower bounded by the same integral up to a factor of $O(\sqrt{\beta})$. Intuitively, this last result can be interpreted as saying that affine scaling step is essentially optimal when only short steps are possible to reduce duality gap while staying in an ℓ_2-neighborhood, which is analogous to the setting that the segment $[z_1, z] \subseteq \mathcal{N}^2(\beta)$ is short. We derive an explicit bound in the following lemma which asserts that an affine scaling step is indeed optimal in $\mathcal{N}^2(O(\beta))$ to traverse any short segment $[z_1, z]$.

Lemma 2.12. *For $\beta \in (0, 1/128]$, suppose there exist some $z \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu_0$ and $z_1 \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z_1) = \mu_1 \ge \mu_0/4$ such that $[z_1, z] \subseteq \mathcal{N}^2(\beta)$. Let Δz^{a} be the affine scaling direction at z, then there exists a step-length $\alpha^{\mathrm{a}} \in [0, 1]$ so that $\overline{\mu}(z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}}) \le \mu_1$ and $z + \alpha\Delta z^{\mathrm{a}} \in \mathcal{N}^2(82\beta)$ for all $\alpha \in [0, \alpha^{\mathrm{a}}]$.*

Let $\alpha^* := 1 - \mu_1/\mu_0$ be the *ideal step length* so that $\overline{\mu}(z + \alpha^*\Delta z^{\mathrm{a}}) = \mu_1$. The next result shows that the difference between $\alpha^*\Delta z^{\mathrm{a}}$ and Δz^{id} can be upper bounded by the deviation of $[z_1, z]$:

Lemma 2.13. *For $\beta \in (0, 1)$, let $z = (x, s), z_1 = (x_1, s_1) \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z) = \mu_0$ and $\overline{\mu}(z_1) = \mu_1 \le \mu_0$. Let $\alpha^* := 1 - \mu_1/\mu_0 \in [0, 1]$, $\Delta z^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ be the affine scaling direction at z, and $\Delta z^{\mathrm{id}} = (\Delta x^{\mathrm{id}}, \Delta s^{\mathrm{id}}) := (x_1 - x, s_1 - s)$. Then,*

$$\left\|\left(\frac{\Delta x^{\mathrm{id}} - \alpha^*\Delta x^{\mathrm{a}}}{x}, \frac{\Delta s^{\mathrm{id}} - \alpha^*\Delta s^{\mathrm{a}}}{s}\right)\right\| \le \frac{4\kappa(z_1, z) + 2\beta\mu_1/\mu_0}{1-\beta}\,.$$

The assumption that $[z_1, z] \subseteq \mathcal{N}^2(\beta)$ guarantees $\kappa(z_1, z) \le 2\beta$ (Lemma 2.5). Then, $z + \alpha^*\Delta z^{\mathrm{a}} \in \mathcal{N}^2(O(\beta))$ follows from $z + \Delta z^{\mathrm{id}} \in \mathcal{N}^2(O(\beta))$ as two steps resemble each other by the above result, and Lemma 2.12 follows for a suitable $\alpha^{\mathrm{a}} \in [\alpha^*, 1]$. We point out that this approach does not work for long straight line segments, because in that case there is no lower bound on μ_1 in terms of μ_0. As α^* becomes very close to 1, $z + \alpha^*\Delta z^{\mathrm{a}}$ resembles taking a full affine scaling step, and hence $z + \alpha^*\Delta z^{\mathrm{a}}$ would not stay in $\mathcal{N}^2(C\beta)$ for a fixed constant $C > 0$.

Overall, TR2-IPM is a predictor-corrector method that uses two types of predictor steps. Starting from an iterate $z \in \mathcal{N}^2(\beta)$, the algorithm first computes the affine scaling direction Δz^{a} at z. Then it checks whether the maximal step length α^{a} that $z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}} \in \mathcal{N}^2(O(\beta))$ is at least a constant (the standard analysis in [43] yields $\alpha^{\mathrm{a}} = \Omega(1/\sqrt{n})$). If $\alpha^{\mathrm{a}} \ge 1/4$, then the algorithm computes the trust region direction Δz^{TR} at z along with a suitable step length α^{TR} and implements a trust region step $z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}$. Otherwise, the algorithm takes an affine scaling step $z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}}$. In both cases, at most 3 corrector steps are taken afterwards to move back to $\mathcal{N}^2(\beta)$. Then, Lemma 2.11 and Lemma 2.12 together imply the iteration bound for TR2-IPM in Theorem 1.4.

The trust region steps are computed by solving $\mathrm{TR}_2(B, N, \gamma)$ (see Section 4). This requires a partition (B, N), for which we use the *associated partition* with $B = \{i \in [n] : |\Delta x^{\mathrm{a}}_i / x_i| \le |\Delta s^{\mathrm{a}}_i / s_i|\}$. If the

affine scaling step length α^{a} is sufficiently long, then the associated partition (B, N) coincides with the true polarized partition; see the full version for a formal statement and its proof.

A canonical and near optimal IPM. Path following IPMs in the ℓ_2-neighborhood are prevalent and core to the IPM family, see e.g. [12, 43, 44, 64]. We now argue that TR2-IPM can be seen as a canonical and near optimal algorithm for path following in the ℓ_2-neighborhood.

Consider two subsequent iterates generated by any path following IPMs denoted by $z, z^+ \in \mathcal{N}^2(\beta)$ with gap values $\mu_0 = \overline{\mu}(z) > \mu_1 = \overline{\mu}(z^+)$, and the direction $\Delta z \coloneqq z^+ - z$. As shown in Section 2.2, if $[z^+, z]$ is long, i.e., $\mu_1/\mu_0 \le 1/4$, then Δz is a feasible solution to $\mathrm{TR}_2(B, N, \gamma)$ (Lemma 2.10), and a trust region step $z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}$ gives $\overline{\mu}(z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}) \le \mu_1$ and $[z + \alpha^{\mathrm{TR}}\Delta z^{\mathrm{TR}}, z] \subseteq \mathcal{N}^2(O(\beta))$ (Lemma 2.11). If $[z^+, z]$ is short, i.e., $\mu_1/\mu_0 > 1/4$, then Δz is close to the affine scaling step with ideal step length at z (Lemma 2.13), and an affine scaling step $z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}}$ gives $\overline{\mu}(z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}}) \le \mu_1$ and $[z + \alpha^{\mathrm{a}}\Delta z^{\mathrm{a}}, z] \subseteq \mathcal{N}^2(O(\beta))$ (Lemma 2.12).

Hence, one could replace each step of an ℓ_2-path following IPM by an affine scaling or trust region step to make at least the same progress in optimality gap while staying in a wider $\mathcal{N}^2(O(\beta))$ neighborhood; one can then return to $\mathcal{N}^2(\beta)$ in at most 3 corrector steps. Consequently, TR2-IPM makes the same progress as any other ℓ_2-path following algorithm with at most a constant increase in the number of corrector steps. It is also worth noting that the primal affine scaling step coincides with the trust region step for $N = [n]$ and $B = \emptyset$, and the analogue holds for the dual affine scaling step with B and N swapped. Hence, both steps can be seen as a trust region step that arises from $(\mathrm{TR}_2(B, N, \gamma))$.

3 Path Complexity in ℓ_2-Neighborhood

We now outline the proof of Theorem 1.5. For any two points $z, z_1 \in \mathcal{N}^2(\beta)$ with $\overline{\mu}(z_1) < \overline{\mu}(z)$, the centrality error of points on $[z_1, z]$ depends on the deviation $\kappa(z_1, z)$ of the whole line. In particular, when z_1, z are both on the central path, it turns out that the maximum ℓ_2-centrality error attained by any point on $[z_1, z]$ is exactly $\kappa(z_1, z)$; see Lemma 2.5. Motivated by this result, we introduce the following notion.

Definition 3.1. We let $\overline{\mathrm{SLC}}(\kappa, \mu_1, \mu_0)$ denote the minimum number of segments of any piecewise linear curve traversing from $z^{\mathrm{cp}}(\mu_0)$ to $z^{\mathrm{cp}}(\mu_1)$ such that all the breakpoints are on the central path and the deviation of each segment is at most the parameter κ.

$\overline{\mathrm{SLC}}(\kappa, \mu_1, \mu_0)$ considers a subclass of piecewise linear curves $\Gamma : [0, 1] \to \mathcal{N}^2(\beta)$ with $\overline{\mu}(\Gamma(0)) = \mu_0$ and $\overline{\mu}(\Gamma(1)) = \mu_1$, and so the inequality $\mathrm{SLC}(\mathcal{N}^2(\beta), \mu_1, \mu_0) \le \overline{\mathrm{SLC}}(\beta, \mu_1, \mu_0)$ holds by definition. The reverse direction of the inequality also holds up to a constant factor of the curvature parameter. This is a consequence of the path stability of any straight line in the ℓ_2-neighborhood, namely if $[z_1, z_0] \subseteq \mathcal{N}^2(\beta)$, then mapping the endpoints to the central path points, we have $[z^{\mathrm{cp}}(\overline{\mu}(z_1)), z^{\mathrm{cp}}(\overline{\mu}(z_0))] \subseteq \mathcal{N}^2(8\beta)$; see the full version for its formal proof. Knowing how to relate path complexity to straight line complexity, it suffices to prove the following result for Theorem 1.5.

Theorem 3.2. *There exists a universal constant $C \ge 0$ such that for any $0 < \beta_1 \le \beta_2 \le 1/128$ and $0 \le \mu_1 < \mu_0$,*

$$\overline{\mathrm{SLC}}\,(\beta_1, \mu_1, \mu_0) = O(1) \cdot \left(\frac{\beta_2}{\beta_1}\right)^C \overline{\mathrm{SLC}}\,(\beta_2, \mu_1, \mu_0)\ . \tag{8}$$

We focus on a single straight line segment $[z^{\mathrm{cp}}(\mu_1), z^{\mathrm{cp}}(\mu_0)]$ on any piecewise linear curve with breakpoints on the central path and consider differently when this segment is either long or short. We outline the strategy in both cases and refer to the full version for formal statements and proofs.

Short segments. The aforementioned relationship between the Sonnevend integral of the central path and the number of iterations of standard path-following methods as in [45, 69, 70] is related to the statement in Theorem 3.2 when only short steps to $\mu_1 > \mu_0/4$ are possible. However, these results do not study precisely the same notion of $\overline{\mathrm{SLC}}$ as we do. Furthermore, the result by Monteiro and Tsuchiya [45] only applies in the limit as $\beta \to 0$, where they showed that the number of iterations of standard affine scaling algorithms converges to $\beta^{-1/2}$ times the curvature integral, which thus suggests an exponent $C = 1/2$ for $\mu_1 = \Omega(\mu_0)$. We show the next result with a self-contained proof based on Lemma 2.3, then the proof of Theorem 3.2 for short segments follows from μ_1/μ_0 being lower bounded. This approach does not require the curvature integral and holds for all small constants β.

Lemma 3.3. *There exists a universal constant $C_1 > 0$ such that for any $0 < \bar{\kappa} \le \kappa \le 1/128$ and $0 \le \eta \le \mu$,*

$$\overline{\mathrm{SLC}}(\bar{\kappa}, \eta, \mu) \le 2\left(1 + \log\left(\frac{\mu}{\eta}\right)\right)\left(\frac{\kappa}{\bar{\kappa}}\right)^{C_1} \cdot \overline{\mathrm{SLC}}(\kappa, \eta, \mu)\ . \tag{9}$$

Long segments. The analysis for long segments, i.e., $\mu_1 \ll \mu_0$, uses different techniques, as there is no upper bound on μ/η in Lemma 3.3. Intuitively, the analysis for short segments no longer applies, because the denominator in $\kappa(\mu_1, \mu)$ becomes arbitrarily small compared to μ_0 for $\mu \in (\mu_1, \mu_0)$ with $\mu \ll \mu_0$, so the deviation could increase rapidly along a segment. As discussed in Section 2.2, long segments exhibit ℓ_2-polarization, namely the variables can be partitioned into two sets B and N such that $x(\mu)_B/x(\mu_0)_B \approx \mathbf{1}_B$ and $x(\mu)_N/x(\mu_0)_N \approx \mu/\mu_0 \cdot \mathbf{1}_N$ for $\mu \in (\mu_1, \mu_0)$, and the analogue holds for $s(\mu)/s(\mu_0)$ with B and N swapped. Therefore, for $\mu \in (\mu_1, \mu_0)$ with $\mu \ll \mu_0$, $\kappa(\mu, \mu_0)$ can be approximated by the term

$$\begin{aligned}\kappa(\mu, \mu_0) &= \frac{\|(x^{\mathrm{cp}}(\mu) - x^{\mathrm{cp}}(\mu_0))(s^{\mathrm{cp}}(\mu) - s^{\mathrm{cp}}(\mu_0))\|}{\left(\sqrt{\mu_0} + \sqrt{\mu}\right)^2} \\ &\approx \left\|\left(\frac{x^{\mathrm{cp}}(\mu)_B}{x^{\mathrm{cp}}(\mu_0)_B} - \mathbf{1}_B, \frac{s^{\mathrm{cp}}(\mu)_N}{s^{\mathrm{cp}}(\mu_0)_N} - \mathbf{1}_N\right)\right\| \eqqcolon \|\phi(\mu, \mu_0)\|\,.\end{aligned}$$

As such, we must investigate how $\|\phi(\cdot, \mu_0)\|$ evolves over $[\mu_1, \mu_0]$. Geometrically, we can consider the trajectory $\phi(\cdot, \mu_0) \subseteq \mathbb{R}^n$. For μ_0, the corresponding vector is $\phi(\mu_0, \mu_0) = (\mathbf{0}_B, \mathbf{0}_N)$, i.e., the origin. By the approximation above, for a fixed neighborhood width β, we can traverse $\phi(\cdot, \mu_0)$ up to a parameter μ for which $\|\phi(\mu, \mu_0)\| \approx \beta$. The goal is to show that for parameter 2β and its corresponding value η such that $\|\phi(\eta, \mu_0)\| \approx 2\beta$, the distance between $\phi(\eta, \mu_0)$ and $\phi(\mu, \mu_0)$ is smaller than the distance between $\phi(\eta, \mu_0)$ and the origin. A priori, this is unclear, as for example $\phi(\cdot, \mu_0)$ may have sharp turns and follow a spiral trajectory. However, we show in the full version that this cannot happen and for *well-separated* $\mu_1 \ll \mu \ll \mu_0$ the

approximate equality $\|\phi(\mu_1,\mu_0)\|_2^2 \approx \|\phi(\mu_1,\mu)\|_2^2+\|\phi(\mu,\mu_0)\|_2^2$. This gives the following result, and the proof of Theorem 3.2 for long segments follows from iteratively applying it.

Lemma 3.4. *There exists a universal constant C_2 such that for any $0 \le \eta \le 2^{-16}\mu_0$ with $\kappa \le 1/128$, we have*

$$\overline{\mathrm{SLC}}\left(\sqrt{4/7}\kappa,\eta,\mu_0\right) \le C_2 \cdot \overline{\mathrm{SLC}}\left(\kappa,\eta,\mu_0\right) . \tag{10}$$

4 Solving the Trust Region Problems

We now discuss the algorithms for solving (TR-2) and (TR-max), starting with some necessary tools.

Approximate Singular Value Decomposition. Approximating eigenvalues is of great interest and has been extensively studied in different communities. Various algorithms with different approaches, computational models and notions of approximation were proposed to tackle this problem, see e.g. [6–8, 49]. In particular, the randomized algorithm in [7] achieves diagonalization of general matrices with high probability in nearly matrix multiplication time. For general non-Hermitian matrices, it is difficult to design a globally convergent and numerically stable algorithm to approximate the eigenvalues and eigenvectors; see [14]. In contrast, fast and stable eigenvalue algorithms that globally converge are well-known for Hermitian matrices (e.g. [66]) with further speed-up possible, for example, $\tilde{O}(n^\omega)$ running time if only the largest eigenvalue is needed [37].

Our trust region algorithms require a multiplicative approximation of the singular values in strongly polynomial time; the requirements are captured by the following definition. This resembles a spectral approximation with the additional requirement of an orthogonal basis whose basis vectors correspond approximately to singular vectors of $\mathbf{A}$ and whose basis vector norms correspond approximately to the singular values of $\mathbf{A}$.

Definition 4.1 (Partial SVD)**.** Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $m \le n$. A matrix $\mathbf{U} \in \mathbb{R}^{\mathrm{rk}(\mathbf{A})\times n}$ is called a partial ϱ-SVD of $\mathbf{A}$ for some parameter $\varrho \ge 1$ if $\mathbf{U}$ has orthogonal rows and for all $x \in \mathbb{R}^n$ we have that $\varrho^{-1}\|\mathbf{U}x\|_2 \le \|\mathbf{A}x\|_2 \le \varrho\|\mathbf{U}x\|_2$.

In the following theorem, the randomized bound follows by Diakonikolas, Tzamos, and Kane [17], and the deterministic algorithm is described in the full version.

Theorem 4.2 (Strongly polynomial partial SVD). *There exists an $\tilde{O}(n^3 \max\{1,\log(1/\varepsilon)\})$ time deterministic strongly polynomial algorithm and an $\tilde{O}(n^\omega \max\{1,\log(1/\varepsilon)\})$ time randomized strongly polynomial algorithm that, given $m\times n$ matrix $\mathbf{A}$ with $m \le n$ and $\varepsilon > 0$, computes a partial $(1+\varepsilon)$-SVD $\mathbf{U}$ of $\mathbf{A}$.*

Fast Approximate LP solvers. Next, we leverage the deterministic Robust IPM [63] to approximately solve LP with ℓ_∞-norm constraint.

Definition 4.3 (Approximate ℓ_∞-regression solver)**.** An approximate ℓ_∞ regression solver is an algorithm that, given a matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$, a vector $b \in \mathrm{Im}(\mathbf{B})$, and a parameter $\delta > 0$, either

(i) returns a vector $y \in \mathbb{R}^n$ with $\mathbf{B}y = b$ and $\|y\|_\infty \le 1+\delta$, or
(ii) returns a vector $z \in \mathbb{R}^m$ with $\|\mathbf{B}^\top z\|_1 < \langle b, z\rangle$, certifying that no vector y with $\mathbf{B}y = b$ and $\|y\|_\infty \le 1$ exists.

Note that the second outcome is a Farkas certificate. In particular, for any y with $\mathbf{B}y = b$, we have that $\langle b,z\rangle = y^\top\mathbf{B}^\top z \le \|y\|_\infty\|\mathbf{B}^\top z\|_1 < \|y\|_\infty\langle b,z\rangle$, showing $\|y\|_\infty > 1$.

Theorem 4.4. *There exists an approximate ℓ_∞-regression solver satisfying Definition 4.3 that runs in time $O\left(n^{\tilde{\omega}+o(1)}\log\left(\frac{1}{\delta}\right)\right)$.*

We give a brief sketch of the proof. We first compute the minimum-norm point $\bar{y} = \mathbf{B}^\dagger b$. If $\|\bar{y}\|_2 > \sqrt{n}$, we may conclude infeasibility as this would imply that all feasible solutions have ℓ_∞-norm greater than 1, and find a Farkas certificate as in *(ii)*. Otherwise, we can use $\bar{y}$ to initialize the IPM for the following LP: $\min \alpha$ s.t. $\mathbf{B}y = b$, $\|y\|_\infty \le 1+\alpha$ with $\alpha = O(n)$.

4.1 Solving the ℓ_2-Trust Region Problem

By Lagrangian duality, (TR-2) can be formulated as a parametric search problem with parameter λ. Let

$$y(\lambda) := \operatorname*{arg\,min}_{\mathbf{B}y=b}\left(\|y_J\|^2 + \lambda\|y_I\|^2\right) \text{ and } \psi(\lambda) := \|y_I(\lambda)\|^2 . \tag{11}$$

We can first resolve the two extreme cases $\lambda = 0$ and ∞ which give certificate of (in)feasibility by solving the related layered least square problems, which can be done efficiently as shown in [40, 64]. Assuming feasibility, for the root $\psi(\lambda^*) = 1$, $y(\lambda^*)$ is the optimal solution to (TR-2). We also note that for λ with $1 \le \psi(\lambda) < 1+\delta$, $y(\lambda)$ is a δ-optimal solution to (TR-2). Hence, finding a δ-optimal solution to (TR-2) is equivalent to a root finding problem of $\psi(\lambda)-1$ on $\mathbb{R}_{>\mathbf{0}}$. One can find the right value of λ using e.g., binary search. However, the possible value range is a priori the entire $\mathbb{R}_{\ge\mathbf{0}}$, though one can obtain upper and lower bounds for λ^* using the binary encoding length of a rational input $(\mathbf{B}, b)$. This is essentially the weakly polynomial algorithm given in [33].

Our approach first uses a standard strategy in the literature, see e.g. [11, 46], which is to use the eigenvalue decomposition of an associated matrix to express ψ as a univariate analytic function:

Lemma 4.5. *Let $\psi : \mathbb{R}_{>\mathbf{0}} \to \mathbb{R}_{>\mathbf{0}}$ be defined as in (11), $(\mu_k)_{k=1}^{|J|}$ be the eigenvalues of $\mathbf{B}_J^\top(\mathbf{B}\mathbf{B}^\top)^{-1}\mathbf{B}_J$, and $\beta_k = \frac{1-\mu_k}{\mu_k}$ for $\mu_k > 0$. Then for some $w \in \mathbb{R}^m$,*

$$\psi(\lambda) = \sum_{k:0<\mu_k<1} \frac{\beta_k w_k^2}{\mu_k(\beta_k+\lambda)^2} + \sum_{k:\mu_k=0} w_k^2 .$$

In particular, ψ is strictly convex and strictly decreasing on $\mathbb{R}_{\ge\mathbf{0}}$. For any $0 < \lambda_1 < \lambda_2$, $\lambda_1^2\psi(\lambda_1) < \lambda_2^2\psi(\lambda_2)$.

Then our algorithm follows a similar idea as the hybrid Newton's method by Ye [68]. At a high level, Ye's algorithm proceeds as follows: given an initial long interval where the root $\psi(\lambda^*) = 1$ lies, it is first partitioned into short intervals, such that if the root λ^* is contained in any one of these short intervals which would be found by binary search, then its lower endpoint satisfies Smale's criterion [55] using the structure of ψ and the narrowness of the interval. Hence, subsequent Newton's method applied to this interval exhibits quadratic convergence. Since the total number of the short intervals depends on the width of the initial long interval, this gives a weakly polynomial algorithm that can solve a general trust region problem to a high accuracy.

To move toward a strongly polynomial algorithm for our (TR-2), we take the initial long interval to be $\mathbb{R}_{>0}$. We now highlight our procedure to shorten this infinitely long interval by making use of β_k which we identify as the *'critical points'* of ψ. Assume that $\lambda \in [\lambda^-, \lambda^+]$ such that for each critical point, either $\beta_i \ll \lambda^-$ or $\beta_i \gg \lambda^+$. Then, $\psi(\lambda) \approx p + q/\lambda^2$ on $[\lambda^-, \lambda^+]$ for some $p, q \geq 0$.

We show that the critical points of ψ coincide with the positive eigenvalues of an associated matrix, for which multiplicative $O(n2^{2n})$-approximations that suffice our purpose can be computed in strongly polynomial time. Our algorithm first approximates two adjacent critical points $\lambda_1 < \lambda_2$ such that $\psi(\lambda_1) > 1 \geq \psi(\lambda_2)$. If λ_2/λ_1 is $O(n2^{2n})/\mathrm{poly}(\delta)$ bounded, where the $O(n2^{2n})$ factor takes into account the approximation accuracy of the critical points from the eigenvalue computation, then we apply binary search followed by Newton's method to find a value λ with $1 \leq \psi(\lambda) < 1+\delta$. The use of binary search is to provide a good initial guess for the subsequent Newton's method to ensure quadratic convergence. Otherwise, we use the above structural property of ψ to narrow down the search interval to $\lambda_1 \leq \hat{\lambda}_1 \leq \hat{\lambda}_2 \leq \lambda_2$ with $\psi(\hat{\lambda}_1) > 1 \geq \psi(\hat{\lambda}_2)$ and $\hat{\lambda}_2/\hat{\lambda}_1$ is now $\mathrm{poly}(n/\delta)$ bounded. We then proceed with searching this interval as in the previous case. We refer to the full version for all these details.

4.2 Solving the ℓ_∞-Trust Region Problem

We now turn to (TR-max). While for (TR-2), feasibility can be decided exactly by computing a projection, in the ℓ_∞-setting already the feasibility question $\mathbf{B}y = b$, $\|y_I\|_\infty \leq 1$ corresponds to a general LP problem. We can however use the Robust IPM (Theorem 4.4) for approximate feasibility: either conclude infeasibility, or find a $y \in \mathbb{R}^n$ such that $\mathbf{B}y = b$, $\|y_I\|_\infty \leq 1 + \delta$ in $n^{\tilde{\omega}+o(1)} \log(1/\delta)$ time. Our main result in this context is that the same asymptotic running time bound applies for solving (TR-max) in the sense of Definition 1.7.

Similarly, for any fixed value λ, we can either find a feasible solution to

$$\mathbf{B}y = b,\ \|y_I\|_\infty \leq 1 + \delta,\ \|y_J\|_\infty \leq (1 + \delta)\lambda\,, \tag{12}$$

in $n^{\tilde{\omega}+o(1)} \log(1/\delta)$ or show that $\mathbf{B}y = b$, $\|y_I\|_\infty \leq 1$, $\|y_J\|_\infty \leq \lambda$ is infeasible. Thus, finding a δ-optimal solution to (TR-max) can be reduced to a parametric search over the objective value λ. Similarly to the ℓ_2-case, the range of possible λ values is weakly polynomial. To remove the dependence on the bit-complexity of $(\mathbf{B}, b)$, we again rely on an approximate singular value decomposition to narrow down the range of the optimal λ and exploit a simpler behavior between two approximate singular values:

Let us briefly sketch the main ideas. A core object of interest is the following linear function that maps $v \in \mathbb{R}^J$ such that $\mathbf{B}_J v \in \mathrm{Im}(\mathbf{B}_I)$ to the vector $\arg\min_{w \in \mathbb{R}^I : \mathbf{B}_J v + \mathbf{B}_I w = \mathbf{0}} \|w\|$. This is the lifting map $\mathbf{L}$ that maps a vector in the J-coordinates to the 'cheapest' way to lift it to a vector in $\ker(\mathbf{B})$. The *singular values* of this lifting map are the critical values of interest.

Besides the critical values, we also require an approximate partial singular value decomposition (see Definition 4.1) $\mathbf{U}$ such that $\|\mathbf{U}x\|$ approximates $\|\mathbf{L}x\|$ up to a multiplicative factor $\varrho = 2^{O(n)}$.

Using binary search, we narrow the search interval to $[\lambda^-, \lambda^+]$ such that each critical value is either much smaller than λ^- or much larger than λ^+, and $\lambda^- \leq \lambda^\star$, whereas for parameter λ^+ and error $\delta/4$ the approximate ℓ_∞-regression solver returns a feasible solution $\bar{y}$ to (12). If λ^+/λ^- is polynomially bounded, the algorithm finds the right λ value using binary search.

If $\lambda^- \ll \lambda^+$, then we can show that $\ker(\mathbf{B}) = V_{\lambda^-} \oplus V_{\lambda^+}$ for two orthogonal subspaces V_{λ^-} and V_{λ^+} such that for all $v \in V_{\lambda^-}$ we have that $\|v_J\|_2 \gg \lambda^- \|v_I\|_2$ and for all $v \in V_{\lambda^+}$ we have $\|v_J\|_2 \ll \lambda^+ \|v_I\|_2$. Intuitively, the subspace V_{λ^-} corresponds to the 'cheap' part of $\ker(\mathbf{B})$ that can be lifted with small $\|y_I\|_\infty$ cost, while the subspace V_{λ^+} corresponds to the 'expensive' part of $\ker(\mathbf{B})$ that can only be lifted with large $\|y_I\|_\infty$ cost. The approximate singular value decomposition allows us to find such a decomposition of $\ker(\mathbf{B})$ in strongly polynomial time.

We can now remove the components of $\bar{y}_J$ in the coordinate projection of V_{λ^-} onto J without increasing $\|\bar{y}_I\|_\infty$ by much, while potentially significantly reducing $\|\bar{y}_J\|_\infty$. On the other hand, we cannot remove any directions of $\bar{y}_J$ in the coordinate projection of V_{λ^+} onto J without increasing $\|\bar{y}_I\|_\infty$ by a large amount. Hence, we can show that the optimal $\lambda^\star$ must be at least $\hat{\lambda}/n$, where $\hat{\lambda}$ is the value of $\|\bar{y}_J\|_\infty$ after removing the components in V_{λ^-}. We can then apply binary search to find a δ-optimal solution in the interval $[\hat{\lambda}/n, \hat{\lambda}]$.

References

[1] Wenbao Ai. 2004. Neighborhood-following algorithms for linear programming. *Science in China Series A: Mathematics* 47, 6 (2004), 812–820.

[2] Wenbao Ai and Shuzhong Zhang. 2005. An $O(\sqrt{n}L)$ Iteration Primal-dual Path-following Method, Based on Wide Neighborhoods and Large Updates, for Monotone LCP. *SIAM Journal on Optimization* 16, 2 (2005), 400–417.

[3] Xavier Allamigeon, Pascal Benchimol, Stéphane Gaubert, and Michael Joswig. 2018. Log-barrier interior point methods are not strongly polynomial. *SIAM Journal on Applied Algebra and Geometry* 2, 1 (2018), 140–178.

[4] Xavier Allamigeon, Daniel Dadush, Georg Loho, Bento Natura, and László A Végh. 2025. Interior point methods are not worse than Simplex. *SIAM J. Comput.* (2025), (available online).

[5] Xavier Allamigeon, Stéphane Gaubert, and Nicolas Vandame. 2022. No self-concordant barrier interior point method is strongly polynomial. In *Proceedings of the 54th Annual ACM Symposium on Theory of Computing (STOC)*. 515–528.

[6] Diego Armentano, Carlos Beltrán, Peter Bürgisser, Felipe Cucker, and Michael Shub. 2018. A stable, polynomial-time algorithm for the eigenpair problem. *Journal of the European Mathematical Society* 20, 6 (2018), 1375–1437.

[7] Jess Banks, Jorge Garza-Vargas, Archit Kulkarni, and Nikhil Srivastava. 2023. Pseudospectral shattering, the sign function, and diagonalization in nearly matrix multiplication time. *Foundations of Computational Mathematics* 23, 6 (2023), 1959–2047.

[8] Jin-Yi Cai. 1994. Computing Jordan normal forms exactly for commuting matrices in polynomial time. *International Journal of Foundations of Computer Science* 5, 03n04 (1994), 293–302.

[9] Michael B Cohen, Yin Tat Lee, and Zhao Song. 2021. Solving linear programs in the current matrix multiplication time. *Journal of the ACM (JACM)* 68, 1 (2021), 1–39.

[10] Marco Colombo and Jacek Gondzio. 2008. Further development of multiple centrality correctors for interior point methods. *Computational Optimization and Applications* 41, 3 (2008), 277–305.

[11] Andrew R Conn, Nicholas IM Gould, and Philippe L Toint. 2000. *Trust region methods*. SIAM.

[12] Daniel Dadush, Sophie Huiberts, Bento Natura, and László A Végh. 2023. A scaling-invariant algorithm for linear programming whose running time depends only on the constraint matrix. *Mathematical Programming* (2023), 1–72.

[13] Daniel Dadush, Zhuan Khye Koh, Bento Natura, Neil Olver, and László A Végh. 2024. A strongly polynomial algorithm for linear programs with at most two nonzero entries per row or column. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. 1561–1572.

[14] James W Demmel. 1997. *Applied numerical linear algebra*. SIAM.

[15] Antoine Deza, Eissa Nematollahi, Reza Peyghami, and Tamás Terlaky. 2006. The central path visits all the vertices of the Klee–Minty cube. *Optimisation Methods and Software* 21, 5 (2006), 851–865.

[16] Antoine Deza, Eissa Nematollahi, and Tamás Terlaky. 2008. How good are interior point methods? Klee–Minty cubes tighten iteration-complexity bounds. *Mathematical Programming* 113, 1 (2008), 1–14.

[17] Ilias Diakonikolas, Christos Tzamos, and Daniel M Kane. 2023. A strongly polynomial algorithm for approximate forster transforms and its application to halfspace learning. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*. 1741–1754.

[18] Ran Duan, Jiayi Mao, Xiao Mao, Xinkai Shu, and Longhui Yin. 2025. Breaking the sorting barrier for directed single-source shortest paths. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*. 36–44.

[19] Farbod Ekbatani, Bento Natura, and László A Végh. 2022. Circuit imbalance measures and linear programming. *Surveys in combinatorics* (2022), 64–114.

[20] Ronald Fagin, Amnon Lotem, and Moni Naor. 2001. Optimal aggregation algorithms for middleware. In *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. 102–113.

[21] Jacek Gondzio. 1996. Multiple centrality corrections in a primal-dual method for linear programming. *Computational optimization and applications* 6, 2 (1996), 137–156.

[22] Jacek Gondzio. 2012. Interior point methods 25 years later. *European Journal of Operational Research* 218, 3 (2012), 587–601.

[23] Clovis C Gonzaga. 1992. Path-following methods for linear programming. *SIAM review* 34, 2 (1992), 167–224.

[24] Bernhard Haeupler, Richard Hladík, Václav Rozhoň, Robert E Tarjan, and Jakub Tetěk. 2024. Universal optimality of Dijkstra via beyond-worst-case heaps. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2099–2130.

[25] Bernhard Haeupler, Richard Hladík, Václav Rozhoň, Robert E Tarjan, and Jakub Tětek. 2025. Bidirectional Dijkstra's Algorithm is Instance-Optimal. In *2025 Symposium on Simplicity in Algorithms (SOSA)*. SIAM, 202–215.

[26] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. The elements of statistical learning: data mining, inference, and prediction.

[27] Pi-Fang Hung and Yinyu Ye. 1996. An Asymptotical O(nL)-Iteration Path-Following Linear Programming Algorithm That Uses Wide Neighborhoods. *SIAM Journal on Optimization* 6, 3 (1996), 570–586.

[28] Florian Jarre and Martin Wechs. 1999. Extending Mehrotra's corrector for linear programs. *Advanced Modeling and Optimization* 1, 2 (1999), 38–60.

[29] Shunhua Jiang, Zhao Song, Omri Weinstein, and Hengjie Zhang. 2021. A faster algorithm for solving general LPs. In *53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. ACM, 823–832.

[30] Narendra Karmarkar. 1984. A new polynomial-time algorithm for linear programming. In *Proceedings of the 16th Annual ACM Symposium on Theory of Computing (STOC)*. 302–311.

[31] NK Karmarkar and KG Ramakrishnan. 1991. Computational results of an interior point algorithm for large scale linear programming. *Mathematical Programming* 52, 1 (1991), 555–586.

[32] Masakazu Kojima, Shinji Mizuno, and Akiko Yoshise. 1989. A primal-dual interior point algorithm for linear programming. In *Progress in Mathematical Programming: Interior-Point and Related Methods*, Nimrod Megiddo (Ed.). Springer.

[33] Guanghui Lan, Renato DC Monteiro, and Takashi Tsuchiya. 2009. A polynomial predictor-corrector trust-region algorithm for linear programming. *SIAM Journal on Optimization* 19, 4 (2009), 1918–1946.

[34] Yin Tat Lee and Aaron Sidford. 2014. Path finding methods for linear programming: Solving linear programs in o (vrank) iterations and faster algorithms for maximum flow. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 424–433.

[35] Yin Tat Lee and Aaron Sidford. 2019. Solving linear programs with sqrt (rank) linear system solves. *arXiv preprint arXiv:1910.08033* (2019).

[36] Yin Tat Lee and Santosh S Vempala. 2021. Tutorial on the robust interior point method. *arXiv preprint arXiv:2108.04734* (2021).

[37] Anand Louis and Santosh S Vempala. 2016. Accelerated Newton iteration for roots of black box polynomials. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 732–740.

[38] Aleksander Madry. 2013. Navigating central path with electrical flows: From flows to matchings, and back. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. IEEE, 253–262.

[39] Nimrod Megiddo. 1989. Pathways to the optimal set in linear programming. In *Progress in mathematical programming: Interior-point and related methods*. Springer, 131–158.

[40] Nimrod Megiddo, Shinji Mizuno, and Takashi Tsuchiya. 1998. A modified layered-step interior-point algorithm for linear programming. *Mathematical Programming* 82, 3 (1998), 339–355.

[41] Sanjay Mehrotra. 1992. On the implementation of a primal-dual interior point method. *SIAM Journal on optimization* 2, 4 (1992), 575–601.

[42] Sanjay Mehrotra and Zhifeng Li. 2005. Convergence Conditions and Krylov Subspace–Based Corrections for Primal-Dual Interior-Point Method. *SIAM Journal on Optimization* 15, 3 (2005), 635–653.

[43] Shinji Mizuno, Michael J. Todd, and Yinyu Ye. 1993. On Adaptive-Step Primal-Dual Interior-Point Algorithms for Linear Programming. *Mathematics of Operations Research* 18, 4 (Nov. 1993), 964–981.

[44] Renato DC Monteiro and Takashi Tsuchiya. 2003. A variant of the Vavasis–Ye layered-step interior-point algorithm for linear programming. *SIAM Journal on Optimization* 13, 4 (2003), 1054–1079.

[45] Renato D.C. Monteiro and Takashi Tsuchiya. 2008. A strong bound on the integral of the central path curvature and its relationship with the iteration-complexity of primal-dual path-following LP algorithms. *Mathematical Programming* 115, 1 (2008), 105–149.

[46] Jorge J Moré and Danny C Sorensen. 1983. Computing a trust region step. *SIAM Journal on scientific and statistical computing* 4, 3 (1983), 553–572.

[47] MOSEK. 2025. MOSEK Optimization Toolbox for MATLAB. https://docs.mosek.com/latest/toolbox.pdf. Release 11.0.13.

[48] Yurii Nesterov and Arkadii Nemirovskii. 1994. *Interior-point polynomial algorithms in convex programming*. SIAM.

[49] Victor Y Pan and Zhao Q Chen. 1999. The complexity of the matrix eigenproblem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. 507–516.

[50] Jiming Peng, Tamás Terlaky, and Yunbin Zhao. 2005. A Predictor-Corrector Algorithm for Linear Optimization Based on a Specific Self-Regular Proximity Function. *SIAM Journal on Optimization* 15, 4 (Jan. 2005), 1105–1127.

[51] MJD Powell. 1993. On the number of iterations of Karmarkar's algorithm for linear programming. *Mathematical Programming* 62, 1 (1993), 153–197.

[52] James Renegar. 1988. A polynomial-time algorithm, based on Newton's method, for linear programming. *Mathematical Programming* 40, 1-3 (1988), 59–93.

[53] James Renegar. 2001. *A mathematical view of interior-point methods in convex optimization*. SIAM.

[54] Tim Roughgarden. 2021. *Beyond the worst-case analysis of algorithms*. Cambridge University Press.

[55] Steve Smale. 1986. Newton's method estimates from data at one point. In *The merging of disciplines: New directions in pure, applied, and computational mathematics*. Springer, 185–196.

[56] Gy Sonnevend, J Stoer, and G Zhao. 1989. On the complexity of following the central path of linear programs by linear extrapolation. *Methods of Operations Research* 63 (1989), 19–31.

[57] Mathieu Tanneau, Miguel F Anjos, and Andrea Lodi. 2021. Design and implementation of a modular interior-point solver for linear optimization. *Mathematical Programming Computation* 13, 3 (2021), 509–551.

[58] Éva Tardos. 1986. A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research* (1986), 250–256.

[59] Michael J Todd. 1993. A lower bound on the number of iterations of an interior-point algorithm for linear programming. In *Numerical Analysis 1993*. CRC Press, 237–259.

[60] Michael J Todd and Yinyu Ye. 1996. A lower bound on the number of iterations of long-step primal-dual linear programming algorithms. *Annals of Operations Research* 62, 1 (1996), 233–252.

[61] Pravin M Vaidya. 1996. A new algorithm for minimizing convex functions over convex sets. *Mathematical programming* 73, 3 (1996), 291–341.

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

[63] Jan van den Brand. 2021. *Unifying Matrix Data Structures: Simplifying and Speeding up Iterative Algorithms*. Society for Industrial and Applied Mathematics, 1–13.

[64] Stephen A Vavasis and Yinyu Ye. 1996. A primal-dual interior point method whose running time depends only on the constraint matrix. *Mathematical Programming* 74, 1 (1996), 79–120.

[65] Adrian Vladu. 2025. Breaking the Barrier of Self-Concordant Barriers: Faster Interior Point Methods for M-Matrices. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing (STOC)*. 2213–2224.

[66] James Hardy Wilkinson. 1968. Global convergene of tridiagonal QR algorithm with origin shifts. *Linear Algebra Appl.* 1, 3 (1968), 409–420.

[67] Virginia Vassilevska Williams, Yinzhan Xu, Zixuan Xu, and Renfei Zhou. 2024. New bounds for matrix multiplication: from alpha to omega. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 3792–3835.

[68] Yinyu Ye. 1992. A new complexity result on minimization of a quadratic function with a sphere constraint. In *Recent Advances in Global Optimization*. Princeton University Press, 19–31.

[69] Gongyun Zhao. 1996. On the relationship between the curvature integral and the complexity of path-following methods in linear programming. *SIAM Journal on Optimization* 6, 1 (1996), 57–73.

[70] Gongyun Zhao and Josef Stoer. 1993. Estimating the complexity of a class of path-following methods for solving linear programs by curvature integrals. *Applied Mathematics and Optimization* 27, 1 (1993), 85–103.

Received 2025-11-04; accepted 2026-02-01