On the Convergence of Inexact Predictor-Corrector Methods for Linear Programming

Gregory Dexter [1] **Agniva Chowdhury** [2] **Haim Avron** [3] **Petros Drineas** [1]

Abstract

Interior point methods (IPMs) are a common approach for solving linear programs (LPs) with strong theoretical guarantees and solid empirical performance. The time complexity of these methods is dominated by the cost of solving a linear system of equations at each iteration. In common applications of linear programming, particularly in machine learning and scientific computing, the size of this linear system can become prohibitively large, requiring the use of iterative solvers, which provide an approximate solution to the linear system. However, approximately solving the linear system at each iteration of an IPM invalidates the theoretical guarantees of common IPM analyses. To remedy this, we theoretically and empirically analyze (slightly modified) *predictor-corrector* IPMs when using approximate linear solvers: our approach guarantees that, when certain conditions are satisfied, the number of IPM iterations does not increase *and* that the final solution remains feasible. We also provide practical instantiations of approximate linear solvers that satisfy these conditions for special classes of constraint matrices using randomized linear algebra.

1. Introduction

Linear programming is a ubiquitous problem appearing across applied mathematics and computer science, with extensive applications in both theory and practice. Modern machine learning applications of linear programming include ℓ_1-regularized SVMs (Zhu et al., 2004), basis pursuit (BP) (Yang & Zhang, 2011), sparse inverse covariance matrix estimation (SICE) (Yuan, 2010), the nonnegative matrix factorization (NMF) (Recht et al., 2012), MAP inference (Meshi & Globerson, 2011), compressed sensing (Donoho, 2006), and adversarial deep learning (Wong & Kolter, 2018). The central importance of this problem has resulted in substantial research on provably accurate algorithms for linear programming. At the same time, practically efficient algorithms are critically needed.

The two major families of algorithms used to solve linear programs are simplex methods and interior point methods (IPMs), with combinations of the two (e.g., IPMs used in the early stages and simplex methods used once approximately optimal solutions have been reached) being useful in practice (Wright, 1997). Predictor-corrector methods, a special type of IPMs, have been particularly useful in solving linear programs accurately and are perhaps the most successful example of theoretically provable yet practically efficient approaches for linear programs.

More precisely, consider a linear program (LP) of the following (standard) form. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be the constraint matrix and $\mathbf{x} \in \mathbb{R}^n$ be the free variable:

$$\min \mathbf{c}^T \mathbf{x}, \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{b}, \ \mathbf{x} \geq \mathbf{0}. \quad (1)$$

The associated dual problem is

$$\max \mathbf{b}^T \mathbf{y}, \text{ subject to } \mathbf{A}^T \mathbf{y} + \mathbf{s} = \mathbf{c}, \ \mathbf{s} \geq \mathbf{0}, \quad (2)$$

where $\mathbf{y} \in \mathbb{R}^m$ is the dual variable and $\mathbf{s} \in \mathbb{R}^n$ is the slack variable. The first (weakly) polynomial time algorithm for linear programming is the ellipsoid method, as proved by Khachiyan in 1979 (Khachiyan, 1979). While the ellipsoid method was deemed to be inefficient in practice, it provided inspiration for the first IPM, developed by Karmarkar in 1984 (Karmarkar, 1984). Karmarkar's initial work was followed by an explosion of research on IPMs that led to numerous algorithms with various theory-practice tradeoffs.

Predictor-corrector IPMs achieve nearly optimal theoretical guarantees, while being commonly used in popular linear optimization packages (Schork & Gondzio, 2020; Almeida & Teixeira, 2015). More precisely, predictor-corrector algorithms are primal-dual path-following IPMs. They compute

[1]Department of Computer Science, Purdue University, West Lafayette, IN, USA [2]Computer Science and Mathematics Division, Oak Ridge National Laboratory, TN, USA [3]School of Mathematical Sciences, Tel Aviv University, Tel Aviv, Israel. Correspondence to: Gregory Dexter <gdexter@purdue.edu>, Agniva Chowdhury <chowdhurya@ornl.gov>, Haim Avron <haimav@tauex.tau.ac.il>, Petros Drineas <pdrineas@purdue.edu>.

a sequence of iterates $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$ within the primal-dual polytope of feasible solutions which approach an optimal solution of the LP. Path-following IPMs require that the iterates within the polytope remain near the so-called central path of the polytope, which results in faster convergence. In each iterate, updates are computed by solving the *normal equations*, namely a system of linear equations of the following form:

$$\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\mathbf{y} = \mathbf{p}. \quad (3)$$

In the above equation, $\mathbf{p}, \Delta\mathbf{y} \in \mathbb{R}^m$ are vectors (see eqn. (8) for the exact definition of $\mathbf{p}$) and $\mathbf{D}^2 = \mathbf{X}\mathbf{S}^{-1}$, where $\mathbf{X}$ is the diagonal matrix whose entries are the $\mathbf{x}_i$ and $\mathbf{S}$ is the diagonal matrix whose entries are the $\mathbf{s}_i$. To analyze the computational complexity of predictor-corrector methods, one first computes the number of *outer iterations*, namely the number of iterations in the IPM algorithm required to converge to an approximately optimal solution. Then, one analyzes the time required to compute each of the iterates by solving the linear system of eqn. (3).

Standard approaches analyzing the rate of convergence and the time complexity of predictor-corrector methods (and other IPMs) typically assume that eqn. (3) is solved exactly at each iteration. However, this assumption becomes untenable for large-scale problems and inexact iterative solvers are nearly universally used in practice. The resulting methods are often called *inexact* predictor-corrector IPMs. Theoretically understanding the behavior of inexact linear equation solvers when combined with IPMs is highly non-trivial. Indeed, an early, provably accurate, approach combining inexact solvers with short step IPMs (a different, less practical, class of IPMs) appeared in the work of Daitch & Spielman (2008), which argued that eqn. (3) can be solved in near linear time when the constraint matrix $\mathbf{A}\mathbf{D}^2\mathbf{A}^T$ is symmetric and diagonally dominant. This allowed fast approximate solutions to problems such as generalized maximum flow (Daitch & Spielman, 2008). More recently, the literature survey by Gondzio (2012) highlighted that pairing IPMs with iterative linear solvers is the way forward towards solving large-scale LPs that arise in machine learning applications. See Section 1.2 for a detailed discussion of relevant prior work on LP solvers.

1.1. Our contributions

In this paper, we prove that a (slightly modified) predictor-corrector IPM can tolerate errors in solving the linear system of eqn. (3) at each outer iteration *without* sacrificing the feasibility of the derived solution and *without* increasing the number of outer iterations of predictor-corrector IPMs. Our proposed inexact predictor-corrector IPM (Algorithm 1 in Section 4) starts with a feasible point and converges to an ϵ-optimal exactly *feasible* solution in $\mathcal{O}(\sqrt{n}\log\frac{\mu_0}{\epsilon})$ outer iterations, where μ_0 is the duality measure at the starting point[1], while approximately solving the system of linear equations of eqn. (3).

Approximately solving the linear system of eqn. (3) is problematic for two reasons: first, it invalidates known analyses of classical predictor-corrector IPMs and, second, it results in infeasible iterates, even when the IPM starts from a feasible point. To address these issues in theory and in practice, we introduce an error-adjustment vector, similar to the work of Monteiro & O'Neal (2003); Chowdhury et al. (2020). The error adjustment vector is our only modification to the classical predictor-corrector IPM and it allows us to return provably accurate, *feasible* solutions, without any increase in the outer iteration complexity of predictor-corrector IPMs.

More precisely, let $\Delta\tilde{\mathbf{y}}$ be an approximate solution for the linear system of eqn. (3) and let $\mathbf{v} \in \mathbb{R}^n$ be an *error adjustment vector* (more on this vector $\mathbf{v}$ later). Let these two vectors satisfy the following conditions:

$$\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} = \mathbf{p} + \mathbf{A}\mathbf{S}^{-1}\mathbf{v} \quad \text{and} \quad \|\mathbf{v}\|_2 < \mathcal{O}(\epsilon). \quad (4)$$

In words, the above conditions simply state that the approximate solution $\Delta\tilde{\mathbf{y}}$ is an exact solution to a slightly modified system of normal equations, where the vector $\mathbf{p}$ has been replaced by the vector $\mathbf{p} + \mathbf{A}\mathbf{S}^{-1}\mathbf{v}$. The two norm of the vector $\mathbf{v}$ must be relatively small, namely less than $\mathcal{O}(\epsilon)$, where ϵ is the target accuracy of the (overall) IPM solver. We emphasize that the error-adjustment vector $\mathbf{v} \in \mathbb{R}^n$ is *user-controlled*, as long as it satisfies the above conditions. Then, these conditions are *sufficient* to guarantee that our predictor-corrector IPM (see Algorithm 1 in Section 4) converges to a solution $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with a duality measure $\mu^* < \epsilon$ in $\mathcal{O}(\sqrt{n}\log\frac{\mu_0}{\epsilon})$ outer iterations. Importantly, the final solution is *exactly* (and not approximately) feasible.

A few additional remarks are necessary to better understand our results for the error-adjusted inexact predictor-corrector IPM. First, we note that our method achieves the best-known outer iteration complexity for predictor-corrector IPMs. Second, the error tolerance of the approximate linear equation solver does not directly depend on n and is constant if ϵ is constant. Third, there are many potential constructions of $\mathbf{v}$ which fulfill the above guarantees. This raises the problem of finding efficient constructions of $\mathbf{v}$ for an inexact linear solver, as this has significant impact on the efficiency of the method.

To address the last point, we exhibit efficient, practical methods to compute $\Delta\tilde{\mathbf{y}}$ and $\mathbf{v}$ by adapting the preconditioned conjugate gradient (PCG) algorithm for constraint matrices $\mathbf{A}$ that are short-and-fat, tall-and-thin, or even have exact low-rank (see Section 4.1 and Appendix D for details).

---

[1]The *duality measure* $\mu = \frac{1}{n}\mathbf{x}^T\mathbf{s}$ quantifies how close a primal-dual point $(\mathbf{x}, \mathbf{y}, \mathbf{s})$ is to optimality at a certain iteration.

More precisely, we show that using PCG, we can compute $\Delta\tilde{\mathbf{y}}$ and $\mathbf{v}$ in $\mathcal{O}\left(\log\frac{n\mu}{\epsilon}\right)$ inner iterations, where each inner iteration is simply a matrix-vector product. It is notable that this inner iteration complexity *does not depend* on the spectrum or the condition number of the input matrix $\mathbf{A}\mathbf{D}^2\mathbf{A}^T$. This is particularly important since the condition number of this matrix changes over iterations and might increase significantly as the outer iterations of the predictor-corrector IPM approach the optimal solution.

Our second contribution in this paper is a novel analysis of the classical, inexact predictor-corrector IPMs in the special setting where the final solution is allowed to be only *approximately feasible*. More precisely, assume that $\Delta\tilde{\mathbf{y}}$ is an approximate solution to the linear system of eqn. (3) that satisfies the following two conditions:

$$\|\Delta\tilde{\mathbf{y}} - (\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{A}\mathbf{D}^2\mathbf{A}^T} \leq \delta \qquad (5)$$

$$\text{and} \quad \|\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2 \leq \delta. \qquad (6)$$

Here δ is the error tolerance of the solver and we note that the first condition guarantees that the exact and the approximate solutions are close with respect to the energy norm, while the second condition guarantees that the two-norm of the residual error of the solver is small. (See Section 2 for notation.) We provide a novel analysis of the standard predictor-corrector method described in Wright (1997) when approximate solvers that satisfy the above conditions are used. Theorem 1 proves that these conditions suffice in order to prove that the standard predictor-corrector algorithm (see Algorithm 3 in Section 5) converges to a solution $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with duality measure μ^* such that $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 < \epsilon$ and $\mu^* < \epsilon$ in $\mathcal{O}(\sqrt{n}\log\frac{\mu_0}{\epsilon})$ outer iterations. Assuming that μ_0 and ϵ are constant, the accuracy parameter δ is set to $\Theta\left(1/\sqrt{n}\right)$ at all iterations of the algorithm.

A few remarks are necessary to better understand the second result. First, the outer iteration complexity is equivalent to the "optimal" $\mathcal{O}(\sqrt{n}\log\frac{\mu_0}{\epsilon})$ iteration complexity of the exact predictor-corrector IPM methods and exhibits linear convergence in the accuracy parameter ϵ. Second, the accuracy parameter δ for the approximate solver is, generally, proportional to $1/\sqrt{n}$, which is similar to the condition of Daitch & Spielman (2008). It is worth noting that our proof collapses if the error bound exceeds this threshold, and an interesting open problem is whether this condition is necessary for predictor-corrector IPMs. Third, the final solution vector is only approximately (and not exactly) feasible, satisfying $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 < \epsilon$ for the accuracy parameter[2] ϵ. Fourth, for the same family of matrices as in our

---

[2] For notational simplicity, we use the same accuracy parameter ϵ for both the duality measure and the approximate feasibility of the final solution vector. Our analysis can be easily extended to use different accuracy parameters.

previous approach (tall-and-thin, short-and-fat, or exact low-rank $k \ll \min\{m, n\}$), we can again show that by using PCG solvers, we can efficiently compute an approximate solution $\Delta\tilde{\mathbf{y}}$ in $\mathcal{O}\left(\log\frac{\sigma_{\max}(\mathbf{A}\mathbf{D})\,n\mu}{\delta}\right)$ iterations of the preconditioned solver, where $\sigma_{\max}(\mathbf{A}\mathbf{D})$ is the largest singular value of the matrix $\mathbf{A}\mathbf{D}$. We emphasize that, unlike our previous approach that uses the error-adjustment vector $\mathbf{v}$, the convergence of the standard inexact predictor-corrector IPMs depends logarithmically on properties of the input matrix $\mathbf{A}\mathbf{D}$ at each iteration.

Our two contributions exhibit a trade-off between algorithmic simplicity and theoretical guarantees. On one hand, the standard predictor-corrector IPM can be used without modifications with an iterative linear solver, but will not return a feasible solution and will need higher solver accuracy that depends on the largest singular value of $\mathbf{A}\mathbf{D}$. Alternatively, the predictor-corrector method can be slightly modified to use an error-adjustment vector with the added benefits of obtaining an exactly feasible solution and removing dependence of the inner iteration complexity on the largest singular value of $\mathbf{A}\mathbf{D}$.

We conclude by noting that our proof techniques are flexible and can be extended to analyze long-step and short-step IPMs, which are, however, less interesting in practice.

1.2. Related Work

Due to the central importance of linear programming in computer science, there exists a large body of work on LPs and IPMs specifically. We refer the reader to the 2012 survey of Gondzio (2012) for more information on the broader state of IPMs, as we focus on literature that is most closely related to our work. Recall that our main focus in this paper is a theoretical analysis of the outer iteration complexity of inexact predictor-corrector IPMs with and without a correction vector that guarantees an exactly feasible solution.

Since the 1950s, there has been continual effort in the theoretical computer science community to develop new LP solvers with improved worst-case asymptotic time complexity. Presently, there is no single fastest LP solver over all typical regimes of LPs. The work of Lee & Sidford (2019) provides an IPM which requires $\widetilde{\mathcal{O}}(\sqrt{\text{rank}(\mathbf{A})}\log\frac{1}{\epsilon})$ outer iterations and $\widetilde{\mathcal{O}}(1)$ linear system solves at each outer iterations. The recent works of Cohen et al. (2021); Song & Yu (2021) have a total time complexity of $\mathcal{O}^*(n^\omega\log\frac{n}{\epsilon})$ where $\omega \sim 2.37$ is the current best-known exponent of matrix multiplication. The work of van den Brand et al. (2020) provides the theoretically fastest solver when $\mathbf{A}$ is tall and dense, in which case the outer iteration complexity is $\widetilde{\mathcal{O}}(\sqrt{n})$ and the total time complexity is $\widetilde{\mathcal{O}}(mn + n^3)$. All three of these works provide short-step IPMs, which have been found to converge slowly in practice. Addition-

ally, these works leverage techniques such as fast matrix multiplication and inverse maintenance, which, due to numerical instability and large constant factors, are generally ineffective in practice. Our algorithms do not depend on any of these techniques. We instead focus on the predictor-corrector method, which is highly effective in practice, yet still has strong theoretical guarantees, with an outer iteration complexity of $\mathcal{O}(\sqrt{n} \log \frac{\mu_0}{\epsilon})$ and one linear system solve per outer iteration. Our method can be fully implemented in less than 150 lines of code using well-established numerical techniques such as preconditioned conjugate gradient descent, as we show in Appendix D. The time complexity of this inexact linear system solver is $\widetilde{\mathcal{O}}(\text{nnz}(\mathbf{A}) + k^3)$, where k is the rank of $\mathbf{A}$.

In our work, we analyze the prototypical predictor-corrector algorithm described in (Wright, 1997). One variant of this method, given by Mehrotra (1992), is considered the industry-standard approach to solving LPs and is perhaps the most common IPM used in linear programming packages (Schork & Gondzio, 2020; Almeida & Teixeira, 2015). We do note that Mehotra's algorithm (unlike the standard predictor-corrector IPMs) does not come with provable accuracy guarantees. Development of new predictor-corrector variants along with theoretical analyses is ongoing. Examples include the variant of Mehrotra's predictor-corrector IPM given by Salahi et al. (2008) or the analysis by Almeida & Teixeira (2015) of a predictor-corrector method specifically suited for LPs arising in transportation problems (Bastos & Paixão, 1993). Other recent works includes the paper by Schork & Gondzio (2020) on empirically evaluating Mehrotra's algorithm when using preconditioned conjugate gradient descent to solve the normal equations at each step. The work of Yang & Yamashita (2018) provides an infeasible predictor-corrector method with $\mathcal{O}(n \log \frac{1}{\epsilon})$ outer iteration complexity and empirically demonstrates its competitiveness with existing methods. The importance of predictor-corrector methods motivates us to develop a better theoretical understanding of their convergence properties when using inexact linear solvers.

Multiple works have analyzed the impact of using inexact linear solvers within IPM algorithms, with early examples being (Bellavia, 1998) and (Mizuno & Jarre, 1999). One method which is relevant to our work is that of Monteiro & O'Neal (2003) which guaranteed the convergence of a long-step IPM by correcting the error of the inexact solver using a correction vector $\mathbf{v}$ as we describe in eqn. (4). This idea was further developed by Chowdhury et al. (2020), which introduced a more efficient construction of $\mathbf{v}$. Another example of such works is Daitch & Spielman (2008), which gives a short-step IPM alongside an inexact Laplacian system solver to solve the generalized max-flow problem. However, the analysis of their inexact short-step IPM does not seem to be directly applicable to solving general LPs

where the constraint matrix is not Laplacian. We improve over these prior works by analyzing the inexact predictor-corrector method using two different approaches and we make minimal assumptions of the linear system solver and LP.

We further note that recently various first-order methods (with proper enhancements) are also being explored to identify high-quality solutions to large-scale LPs quickly (Basu et al., 2020; Lin et al., 2021; Applegate et al., 2021). However, most of these endeavors are based on the combinations of existing heuristics and do not come with theoretical guarantees.

2. Background

2.1. Notation

For any natural number n, let $[n] = \{1, 2, ...n\}$. Bold capital letters denote matrices (e.g, $\mathbf{A}$); bold lower case letters denote vectors (e.g., $\mathbf{x}$); and the i-th element of vector $\mathbf{x}$ is written as $\mathbf{x}_i$. Let $\mathbf{I}_n$ denote the $n \times n$ identity matrix; let $\mathbf{1}_m$ and $\mathbf{0}_m$ denote length m vectors of all ones and zeroes respectively. We define the norm of a vector $\|\mathbf{x}\|_p$ to be its well-known ℓ_p norm and the norm of a matrix $\|\mathbf{A}\|_p$ to be the induced ℓ_p norm, i.e. $\|\mathbf{A}\|_p = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{A}\mathbf{x}\|_p$. We also use the *energy norm* $\|\mathbf{x}\|_{\mathbf{M}} = \sqrt{\mathbf{x}^T \mathbf{M} \mathbf{x}}$, where $\mathbf{x}$ is a vector and $\mathbf{M}$ is a symmetric positive definite matrix. We denote the Hadamard (element-wise) product of two vectors $\mathbf{u}, \mathbf{v}$ as $\mathbf{u} \circ \mathbf{v}$. Finally, we denote the Moore-Penrose pseudoinverse of a matrix $\mathbf{A}$ as $\mathbf{A}^\dagger$.

2.2. Background

Interior point methods using an exact linear solver iteratively converge towards a primal-dual solution $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$, which optimally solves the primal and dual LPs of eqns. (1, 2). The direction of each iterative step is determined by solving the so-called *normal equations*:

$$\mathbf{A}\mathbf{D}^2\mathbf{A}^T \Delta \mathbf{y} = -\sigma\mu \mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x}, \quad (7a)$$

$$\Delta \mathbf{s} = -\mathbf{A}^T \Delta \mathbf{y}, \quad (7b)$$

$$\Delta \mathbf{x} = -\mathbf{x} + \sigma\mu \mathbf{S}^{-1}\mathbf{1}_m - \mathbf{D}^2 \Delta \mathbf{s}. \quad (7c)$$

In the above, $\sigma \in [0, 1]$ is the *centering parameter*, which controls the tradeoff between progressing towards the optimal solution and staying near the central path. Let $\mathbf{p}$ be equal to the right-hand-side of eqn. (7a), i.e.,

$$\mathbf{p} = -\sigma\mu \mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x}. \quad (8)$$

Path-following IPM algorithms ensure that the iterates remain sufficiently far from the boundary of the convex polytope representing the feasible set of the primal and dual LPs and near the central path. In this paper, we use the ℓ_2

neighborhood $\mathcal{N}_2(\theta)$ defined as follows.

$$\mathcal{N}_2(\theta) = \big\{(\mathbf{x},\mathbf{y},\mathbf{s}) \in \mathbb{R}^{2n+m} :$$
$$\|\mathbf{x}\circ\mathbf{s} - \mu\mathbf{1}_n\|_2 \leq \theta\mu, \ (\mathbf{x},\mathbf{s}) > 0\big\}. \quad (9)$$

The step size of the outer iterations will need to be dynamically determined to ensure that the iterates remain in the appropriate neighborhood. The following notation compactly describes the next iterate after a step of size α:

$$\mathbf{x}(\alpha) = \mathbf{x} + \alpha\Delta\mathbf{x}, \ \mathbf{y}(\alpha) = \mathbf{y} + \alpha\Delta\mathbf{y}, \ \mathbf{s}(\alpha) = \mathbf{s} + \alpha\Delta\mathbf{s},$$
$$\text{and } \mu(\alpha) = (\mathbf{x} + \alpha\Delta\mathbf{x})^T(\mathbf{s} + \alpha\Delta\mathbf{s})/n.$$

The following identities hold for the exact steps determined by eqn. (7) above; see (Wright, 1997) for details:

$$\Delta\mathbf{x}^T\Delta\mathbf{s} = 0, \quad (10)$$
$$\mathbf{s}^T\Delta\mathbf{x} + \mathbf{x}^T\Delta\mathbf{s} = -n\mu + n\sigma\mu, \quad (11)$$
$$\mu(\alpha) = (1 - \alpha + \alpha\sigma)\mu. \quad (12)$$

We will also use the following bound on $\|\Delta\mathbf{x}\circ\Delta\mathbf{s}\|_2$ to argue that the iterative steps remain in the neighborhood defined by eqn. (9); see Lemma 5.4 in (Wright, 1997) for a proof:

$$(\mathbf{x},\mathbf{y},\mathbf{s}) \in \mathcal{N}_2(\theta) \Rightarrow \|\Delta\mathbf{x}\circ\Delta\mathbf{s}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu. \quad (13)$$

3. Overview of our approach and proofs

Inexactly solving the normal equations when determining the step direction in predictor-corrector IPMs adds new difficulties to the convergence analysis such methods. Identities that are critical in analyzing the the exact methods, such as $\Delta\mathbf{x}^T\Delta\mathbf{s} = 0$, no longer hold. Another source of difficulty is that, even when starting from a feasible initial point, the iterates will become infeasible due to the error incurred by the solvers. We can handle this infeasibility in two different ways. First, we can assume bounds on the maximum error of the solver at each step, which would guarantee that the final solution is ϵ-feasible, i.e. $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 \leq \epsilon$. The second approach, previously introduced in (Monteiro & O'Neal, 2003), is to adjust the error in each step to ensure that the next iterate is feasible. We analyze both approaches in our work.

The following equation block designates the step of inexactly solving the normal equations, where the vector $\mathbf{f}$ is the error residual incurred by solving for $\Delta\tilde{\mathbf{y}}$ using an inexact linear solver:

$$\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} = -\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x} - \mathbf{f}, \quad (14a)$$
$$\Delta\tilde{\mathbf{s}} = -\mathbf{A}^T\Delta\tilde{\mathbf{y}}, \quad (14b)$$
$$\Delta\tilde{\mathbf{x}} = -\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{D}^2\Delta\tilde{\mathbf{s}}. \quad (14c)$$

The next equation block deals with the case of an error-adjusted approximate step. The idea behind this error-adjustment is the construction of a vector $\mathbf{v}$ with small norm such that the error vector $\mathbf{f}$ is exactly equal to $-\mathbf{A}\mathbf{S}^{-1}\mathbf{v}$. We then correct the primal step $\Delta\tilde{\mathbf{x}}$ by subtracting $\mathbf{S}^{-1}\mathbf{v}$, which guarantees that $\mathbf{A}\Delta\tilde{\mathbf{x}}$ is equal to zero:

$$\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} = -\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x} + \mathbf{A}\mathbf{S}^{-1}\mathbf{v}, \quad (15a)$$
$$\Delta\tilde{\mathbf{s}} = -\mathbf{A}^T\Delta\tilde{\mathbf{y}}, \quad (15b)$$
$$\Delta\tilde{\mathbf{x}} = -\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{D}^2\Delta\tilde{\mathbf{s}} - \mathbf{S}^{-1}\mathbf{v}. \quad (15c)$$

Note that both the inexact and error-adjusted normal equations maintain dual feasibility of the iterate when starting from any dual feasible starting point, i.e., $\mathbf{A}^T\mathbf{y} + \mathbf{s} = \mathbf{c}$.

In order to derive our theoretical bounds, we first analyzed the *uncorrected* inexact predictor-corrector IPM, which is the original predictor-corrector IPM using an approximate solver denoted Solve. We present an overview of this analysis in Section 5 (see Algorithm 3). In the interest of space, we delegate detailed analysis of this algorithm to the Appendix (see Appendix B). The uncorrected inexact predictor-corrector IPM takes two steps (a predictor step and a corrector step) in each outer iteration. Starting from a point in $\mathcal{N}_2(0.25)$, the algorithm takes a *predictor step* with centering parameter $\sigma = 0$ and a dynamically chosen step size α, such that the iterate remains in $\mathcal{N}_2(0.5)$. The algorithm then takes a *corrector step* with centering parameter $\sigma = 1$ and step size $\alpha = 1$, which returns the iterate back to $\mathcal{N}_2(0.25)$. The predictor step results in a multiplicative decrease in the duality measure, and the corrector step sets up the next predictor step, while only resulting in a slight additive increase in the duality measure. Our main result for Algorithm 3 in Section 5 is given by the following theorem.

Theorem 1. *Let $\epsilon > 0$ be a tolerance parameter and $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0) \in \mathcal{N}_2(0.25)$ with duality measure μ_0 be a feasible starting point. Then, Algorithm 3 (see Section 5) converges to a dual-feasible point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with duality measure μ^* such that $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 < \epsilon$ and $\mu^* < 2\epsilon$ in $\mathcal{O}(\sqrt{n}\log\frac{\mu_0}{\epsilon})$ outer iterations, where the approximate linear solver Solve is called twice in each outer iteration with error tolerance parameter $\delta_{\epsilon,n} = \min\{\frac{\sqrt{\epsilon}}{2^6}, \frac{\epsilon C_0}{2\sqrt{n}\log\mu_0/\epsilon}\}$ and the constant C_0 is defined in Lemma B.4.*

Next, in Section 4, we present and analyze *our main contribution*, the error-adjusted predictor-corrector method (Algorithm 1), which uses a linear solver[3] $\text{Solve}^{\mathbf{v}}$ to compute an approximate solution $\Delta\tilde{\mathbf{y}}$ and an error-adjustment vector $\mathbf{v}$ that satisfy eqns. (15). The main result of Section 4 is the following theorem.

---

[3] $(\Delta\tilde{\mathbf{y}}, \mathbf{v}) = \text{Solve}^{\mathbf{v}}(\mathbf{A}, \mathbf{p}, \delta)$ takes three inputs: the input matrix $\mathbf{A}$, the response vector $\mathbf{p}$, and the target accuracy (or tolerance) δ and returns an approximate solution $\Delta\tilde{\mathbf{y}}$ and the error-adjustment vector $\mathbf{v}$.

Theorem 2. *Let $\epsilon > 0$ be a tolerance parameter and $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0) \in \mathcal{N}_2(0.25)$ with duality measure μ_0 be a feasible starting point. Then, Algorithm 1 converges to a primal-dual feasible point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with duality measure μ^* such that $\mu^* < 2\epsilon$ in $\mathcal{O}(\sqrt{n} \log \frac{\mu_0}{\epsilon})$ outer iterations, where* Solve^v *is called twice in each outer iteration with error tolerance $\epsilon/2^7$.*

The convergence analysis of both inexact predictor-corrector algorithms shares the same overall structure, which we now outline. First, we upper bound $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$, a technical result that will be needed in upcoming steps. Second, we derive a bound for the left-hand side of the $\mathcal{N}_2$ neighborhood condition ($\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2$) after step size α. This bound depends on $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$ and the error of the linear solver. Third, we find a value for the step size α that depends on $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$, which keeps the next iterate in the appropriate neighborhood, $\mathcal{N}_2(0.5)$. Fourth, we lower bound the step size α using the upper bound on $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$. Fifth, we use the lower bound on α from the previous step to lower bound the multiplicative decrease in the duality measure after the predictor step. Finally, we prove that the corrector step with step size $\alpha = 1$ returns the iterate to $\mathcal{N}_2(0.25)$ by using the inequality from the second step and then bound the resulting additive increase in the duality measure.

For both predictor-corrector algorithms, the above structure provides a guaranteed decrease in the duality measure over a single step of the form $\tilde{\mu}_1 \leq \left(1 - \frac{C_0}{\sqrt{n}}\right)\mu_0 + C_1\frac{\delta_{\epsilon,n}}{n}$, where $C_0 \in (0,1)$, $C_1 \in [0, C_0/\sqrt{n}]$, and $\delta_{\epsilon,n} > 0$ is the tolerance parameter for the corresponding linear solver. We can use this relation to conclude (using standard arguments) that each algorithm converges to a point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with duality measure $\mu^* < 2\epsilon$. We note that the proof of the inexact predictor-corrector IPM without using error-adjustment is simpler, partly because the duality measure during its inexact predictor-corrector step is always higher than the duality measure during the exact step. This is not the case for the inexact predictor-corrector IPM with error-adjustment, which needs extra care in bounding the duality gap decrease in each iteration.

In Section 4.1 (see also Appendix D), we show how the approximate linear solver Solve^v can be efficiently instantiated when the constraint matrix $\mathbf{A}$ has exact low rank (which includes as special cases tall-and-thin and short-and-fat matrices), by using a preconditioned conjugate gradient (PCG) method.

4. Error-adjusted Inexact Predictor-Corrector IPMs

In this section, we introduce an algorithm that we will call error-adjusted inexact predictor-corrector IPM (Algorithm 1). This algorithm uses an inexact linear solver

Solve^v, which returns an approximate solution $\Delta\tilde{\mathbf{y}}$ and a correction vector $\mathbf{v}$ that satisfies the conditions of eqn. (4). This correction vector guarantees that the final solution will be exactly feasible and, as discussed in Section 1, Algorithm 1 can tolerate larger errors for the inexact solver. We follow the proof sketch of Section 3 to prove convergence guarantees and time complexity for Algorithm 1. We will assume that the matrix $\mathbf{A}$ has full row rank, i.e., $rank(\mathbf{A}) = m \leq n$; see Appendix D.3 for extensions alleviating this constraint.

Algorithm 1 Error-adjusted Inexact Predictor-corrector

Input: $A \in \mathbb{R}^{m \times n}$, initial feasible point $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{s}^0) \in \mathcal{N}_2(0.25)$; IPM tolerance $\epsilon > 0$.

Initialize: $k \leftarrow 0$;

while $\mu_k > 2\epsilon$

Predictor Step ($\sigma = 0$):
(a) Compute $(\Delta\tilde{\mathbf{y}}, \mathbf{v}) =$ Solve^v$(\mathbf{A}\mathbf{D}^2\mathbf{A}^T, \mathbf{A}\mathbf{x}, \epsilon/2^7)$.

(b) Compute $\Delta\tilde{\mathbf{x}}$ and $\Delta\tilde{\mathbf{s}}$ via eqn. (15).

(c) Set $\alpha = \min\left\{1/2, \ (\mu/16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2)^{1/2}\right\}$

(d) Compute $(\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) + \alpha(\Delta\tilde{\mathbf{x}}_k, \Delta\tilde{\mathbf{y}}_k, \Delta\tilde{\mathbf{s}}_k)$.

Corrector Step ($\alpha = 1, \sigma = 1$):
(e) Compute $(\Delta\tilde{\mathbf{y}}, \mathbf{v}) =$ Solve^v$(\mathbf{A}\mathbf{D}^2\mathbf{A}^T, -\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x}, \epsilon/2^7)$.

(f) Compute $\Delta\tilde{\mathbf{x}}$ and $\Delta\tilde{\mathbf{s}}$ via eqn. (15).

(g) Compute $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \mathbf{s}_{k+1}) = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) + (\Delta\tilde{\mathbf{x}}_k, \Delta\tilde{\mathbf{y}}_k, \Delta\tilde{\mathbf{s}}_k)$.

(h) $k \leftarrow k + 1$.
end while

We proceed by expressing the difference of the exact vs. approximate solutions, using eqn. (7) vs. eqn. (15):

$$\Delta\mathbf{y} - \Delta\tilde{\mathbf{y}} = -(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}, \tag{16}$$

$$\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}} = -\mathbf{A}^T(\Delta\mathbf{y} - \Delta\tilde{\mathbf{y}})$$
$$= \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}, \tag{17}$$

$$\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}} = -\mathbf{D}^2(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}}) + \mathbf{S}^{-1}\mathbf{v}$$
$$= -\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \mathbf{S}^{-1}\mathbf{v}. \tag{18}$$

We prove that Algorithm 1 converges to a point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$, such that $\mu^* < 2\epsilon$, $\mathbf{A}\mathbf{x}^* = \mathbf{b}$, and $\mathbf{A}^T\mathbf{y}^* + \mathbf{s}^* = \mathbf{c}$ in $\mathcal{O}(\sqrt{n} \log \mu_0/\epsilon)$ outer iterations. First, we start with a technical result to bound $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$. (All proofs are delegated to Appendix C.)

Lemma 4.1. *Let $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$ and let $(\Delta\tilde{\mathbf{x}}, \Delta\tilde{\mathbf{y}}, \Delta\tilde{\mathbf{s}})$ be the step calculated from the inexact normal equations*

without error-adjustment (see eqn. (15)). Then,

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu$$
$$+ 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2$$
$$+ 2\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2^2.$$

This inequality represents a key technical contribution of our results. The proofs of Monteiro & O'Neal (2003); Chowdhury et al. (2020) on the convergence of a long-step IPM cannot be readily extended to the predictor-corrector method, as the predictor-corrector algorithm requires finer control over deviations from the central path due to using the ℓ_2-neighborhood. However, this more restrictive neighborhood allows it to achieve better outer iteration complexity. Observe that in the corrector step, when $\sigma = 1$, our bound does not directly scale with n in this case, in contrast to Lemma 16 in (Chowdhury et al., 2020) and Lemma 3.7 in (Monteiro & O'Neal, 2003).

We can use the previous inequality to bound the deviation of the iterate from the central path after a step of size α.

Lemma 4.2. *If $\alpha \in [0, 1]$, then*

$$\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2$$
$$+ \alpha^2\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + 2\alpha\|\mathbf{v}\|_2.$$

Given the previous bound, we can then derive a step size α which guarantees that the iterate remains in $\mathcal{N}_2(0.5)$ after the predictor step.

Lemma 4.3. *If $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.25)$, $\alpha = \min\left\{1/2, (\mu/16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2)^{1/2}\right\}$, and $\|\mathbf{v}\|_2 \leq \mu/32$, then the predictor step $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$.*

We then show that the predictor step with step size α as given in the above lemma guarantees a multiplicative decrease in the duality gap. Recall that $\sigma = 0$ in the predictor step when solving the normal equations.

Lemma 4.4. *If $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.25)$, $\alpha = \min\left\{1/2, (\mu/16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2)^{1/2}\right\}$, and $\|\mathbf{v}\|_2 \leq \mu/32$, then the predictor step $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha))$ remains in $\mathcal{N}_2(0.5)$ and there exists a constant $C_0 \in (0, 1)$ such that,*

$$\frac{\tilde{\mu}(\alpha)}{\mu} \leq 1 - \frac{C_0}{\sqrt{n}}.$$

After the previous lemma, we have shown that the predictor step results in a multiplicative decrease in the duality gap, while keeping the next iterate in the neighborhood $\mathcal{N}_2(0.5)$. We then show that the corrector step returns the iterate to the $\mathcal{N}_2(0.25)$ neighborhood, while increasing the duality gap by a small additive amount.

Lemma 4.5. *Let $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.5)$ and $\|\mathbf{v}\|_2 \leq \mu/2^7$. Then, the corrector step $(\tilde{\mathbf{x}}(1), \tilde{\mathbf{y}}(1), \tilde{\mathbf{s}}(1)) \in \mathcal{N}_2(0.25)$ and $|\tilde{\mu}(1) - \mu| \leq \frac{1}{\sqrt{n}}\|\mathbf{v}\|_2$.*

Overall, the structure of the proof approach for the inexact predictor-corrector method (Section 5) is similar to the proof structure shown here. However, proving the individual lemmas for the error-adjusted algorithm requires slightly more work, since the correction step adds an additional adjustment to the iterates in each step, which must be accounted for.

In comparison to the proof of the standard predictor-corrector method found in Wright (1997), the general idea and organization of the lemmas are shared, however, accounting for the error in each step can lead to unwieldy and complicated formulas. An important part of our proof that partially alleviates this problem is to generalize the inequalities used in Wright (1997) to depend smoothly on the error of the inexact linear system solve, so we recover the statement of each lemma when the error is zero. This allows us to appropriately set the linear system solver precision to give up small factors in the tightness of our result in each lemma, which can then be offset by increasing the precision of the linear system solver. By doing so, we ensure that the added proof complexity for the inexact predictor-corrector method is locally resolved in each lemma. As a result, our proofs are conceptually simpler than prior work proving the convergence of inexact IPMs, such as Chowdhury et al. (2020) and Daitch & Spielman (2008).

4.1. Implementing `Solve`$^{\mathbf{v}}$

We demonstrate that `Solve`$^{\mathbf{v}}$ can be effectively implemented using a preconditioned conjugate gradient (PCG) method that also constructs the correction vector $\mathbf{v}$ that satisfies the conditions of eqn. (4). By employing a randomized preconditioner, the resulting PCG method guarantees an exponential decrease in the energy norm of the residual. Here we sketch our approach, and we provide details in Appendix D.

Let $\mathbf{AD} = \mathbf{U\Sigma V}^T$ be the thin SVD representation with $\mathbf{V} \in \mathbb{R}^{m \times n}$ and $\mathbf{W} \in \mathbb{R}^{n \times w}$ be an *oblivious sparse sketching matrix* which satisfies, for some accuracy parameter $\zeta \in (0, 1/2)$:

$$\|\mathbf{VWW}^T\mathbf{V}^T - \mathbf{I}_m\|_2 \leq \frac{\zeta}{2}, \tag{19}$$

with probability at least $1 - \eta$. The work of Cohen et al. (2016) shows how to construct such a matrix $\mathbf{W}$ fulfilling this guarantee with sketch size $w = \mathcal{O}(m/\zeta^2 \cdot \log m/\eta)$ and $\mathcal{O}(1/\zeta \cdot \log m/\eta)$ non-zero entries per row. Next, we use the above sketching matrix to define
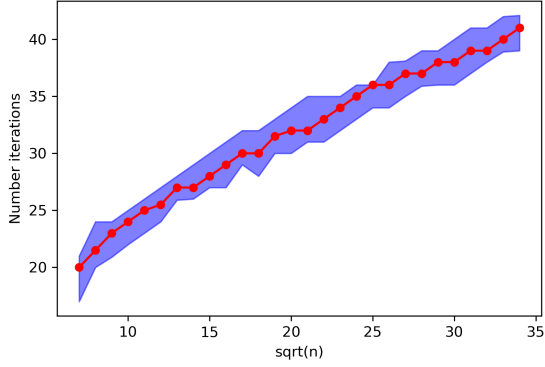
$$\mathbf{Q} = \mathbf{ADWW}^T\mathbf{DA}^T.$$

Figure 1. This graph demonstrates the linear relationship between the number of iterations and $\sqrt{n}$, as predicted by Theorem 2. The line shows the median number of iterations and the intervals designate the 10% and 90% quantiles out of 60 repetitions. Other parameters are $m = 20$; $\epsilon = 0.1$; and solver tolerance 0.001.

We note that $\mathbf{Q}$ does not need to be explicitly constructed, since we will only use the inverse of its square root $\mathbf{Q}^{-1/2}$ (see Algorithm 2). More specifically, since $\mathbf{W}$ has $\log m/\eta$ non-zero entries per row and $\mathbf{D}$ is a diagonal matrix, $\mathbf{ADW}$ can be computed in $\mathcal{O}(\mathsf{nnz}(\mathbf{A}) \cdot \log m/\eta)$ time. Then, computing $\mathbf{Q}^{-1/2}$ via the SVD of $\mathbf{ADW}$ takes $\mathcal{O}(m^3 \log m/\eta)$ time. The overall time complexity to compute $\mathbf{Q}^{-1/2}$ is $\mathcal{O}(\mathsf{nnz}(\mathbf{A}) \cdot \log m/\eta + m^3 \log m/\eta)$.

Next, we prove that the vector $\tilde{\mathbf{z}}^t$ returned by PCG (Algorithm 2) fulfills the following inequality with probability at least $1 - \eta$:

$$\|\mathbf{Q}^{-1/2}(\mathbf{AD}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\tilde{\mathbf{z}}^t - \mathbf{Q}^{-1/2}\mathbf{p}\|_2 \leq \zeta^t\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2,$$

for the aforementioned error-parameter ζ.

Algorithm 2 Preconditioned Conjugate Gradient (Algorithm 1 in (Chowdhury et al., 2020))

Input: $\mathbf{AD} \in \mathbb{R}^{m \times n}$ with $m \ll n$, $\mathbf{p} \in \mathbb{R}^m$, failure probability η, iteration count t;

1. Compute $\mathbf{ADW}$ and its SVD, where $\mathbf{W} \in \mathbb{R}^{n \times w}$ fulfills eqn. (33) with $r = m$. Let $\mathbf{U}_{\mathbf{Q}} \in \mathbb{R}^{m \times m}$ be the matrix of its left singular vectors and let $\boldsymbol{\Sigma}_{\mathbf{Q}}^{1/2} \in \mathbb{R}^{m \times m}$ be the matrix of its singular values;

2. Compute $\mathbf{Q}^{-1/2} = \mathbf{U}_{\mathbf{Q}}\boldsymbol{\Sigma}_{\mathbf{Q}}^{-1/2}\mathbf{U}_{\mathbf{Q}}^{\mathsf{T}}$;

3. Initialize $\tilde{\mathbf{z}}^0 \leftarrow \mathbf{0}_m$ and run standard CG on $\mathbf{Q}^{-1/2}\mathbf{AD}^2\mathbf{A}^T\mathbf{Q}^{-1/2}\tilde{\mathbf{z}} = \mathbf{Q}^{-1/2}\mathbf{p}$ for t iterations;

Output: return $\Delta\tilde{\mathbf{y}} = \mathbf{Q}^{-1/2}\tilde{\mathbf{z}}^t$;

Given $\mathbf{W}$ (the sketching matrix used to construct the preconditioner), we proceed to construct the error-adjustment
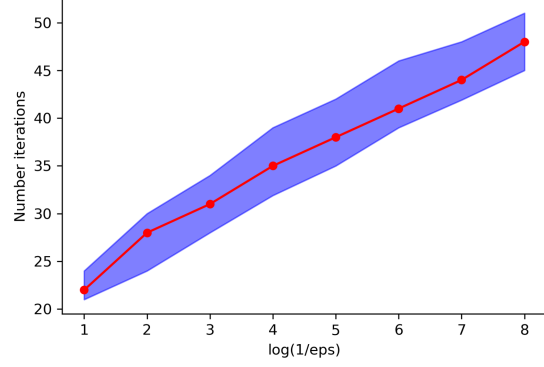

Figure 2. This graph demonstrates the linear relationship between the number of iterations and $\log(1/\epsilon)$, as predicted by Theorem 2. The line shows the median number of iterations and the confidence intervals designate the 10% and 90% quantiles out of 60 repetitions. Other parameters are $m = 30$; $n = 70$; and solver tolerance equal to ϵ.

vector $\mathbf{v}$ as follows:

$$\mathbf{v} = (\mathbf{XS})^{1/2}\mathbf{W}(\mathbf{ADW})^{\dagger}(\mathbf{AD}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}),$$

where $\Delta\tilde{\mathbf{y}} = \tilde{\mathbf{z}}^t$ after $t = \mathcal{O}\left(\log \frac{n\mu}{\epsilon}\right)$ iterations. The additional time needed to compute the error-adjustment vector $\mathbf{v}$ is negligible, since it only adds matrix vector products, using quantities that have already been computed and are available to the algorithm. Combining randomized preconditioning, conjugate gradients, and our proposed construction of the error-adjustment vector $\mathbf{v}$ is theoretically and practically efficient for short-and-fat, tall-and-thin, and exact low-rank matrices. It can also take advantage of any sparsity in the input matrix, since both our preconditioner construction and conjugate gradient methods leverage input sparsity.

4.2. Empirical validation of Theorem 2

We experimentally validate the statement of Theorem 2 on synthetic data. Specifically, we observe the predicted linear relationship between the number of iterations vs. $\sqrt{n}$ and $\log(1/\epsilon)$, when the precision of $\mathtt{Solve}^{\mathbf{v}}$ is set to $\mathcal{O}(\epsilon)$. To generate the synthetic LPs, we sample a constraint matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, the initial primal variable $\mathbf{x}_0 \in \mathbb{R}^n$, and the initial dual variable $\mathbf{y}_0 \in \mathbb{R}^m$, where each entry is sampled uniformly over an appropriate interval. From these points, we can choose an initial slack variable such that $\|\mathbf{x}_0 \circ \mathbf{s}_0 - \mu_0\mathbf{1}_n\|_2 < (0.25)\mu_0$. The initial primal-dual point along with the constraint matrix completely describes the LP, since the initial point is assumed to be primal-dual feasible. We implement $\mathtt{Solve}^{\mathbf{v}}$ to find the primal-dual step and correction vector $\mathbf{v}$ by uniformly sampling a vector $\mathbf{v} \in \mathbb{R}^n$ fulfilling $\|\mathbf{v}\|_2 = \mathcal{O}(\epsilon)$ and then solving for the corresponding step in eqn. (15a) exactly; see Figures 1 and 2. We also test our preconditioned gradient descent method for

finding the primal-dual step, and error-adjustment vector $\mathbf{v}$ in Appendix E and we find that the performance of our approach is comparable to the exact method shown here in terms of outer iteration number. Additionally, we evaluate the real running time of Algorithm 1 versus the same predictor-corrector IPM using an exact linear system solve (see Appendix E.4)

5. Inexact Predictor-Corrector IPM

We next briefly introduce the inexact predictor-corrector IPM (without error-adjustment) and describe how the proof of Theorem 1 compares to the approach described in Section 4. This algorithm calls an inexact linear system solver `Solve` which returns a solution with guarantees on the energy-norm and ℓ_2-norm of the residual (see eqn. (5)). Further details are provided in Appendix B.

Algorithm 3 Inexact Predictor-Corrector without correction

Input: $\mathbf{A} \in \mathbb{R}^{m \times n}$, initial feasible point $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{s}^0) \in \mathcal{N}_2(0.25)$; IPM tolerance $\epsilon > 0$, linear solver tolerance $\delta_{\epsilon,n} > 0$.

Initialize: $k \leftarrow 0$;

while $\mu_k > 2\epsilon$

Predictor Step $(\sigma = 0)$:

(a) Compute $\Delta\tilde{\mathbf{y}} = \texttt{Solve}(\mathbf{AD}^2\mathbf{A}^T, \mathbf{Ax}, \delta_{\epsilon,n})$.

(b) Compute $\Delta\tilde{\mathbf{x}}$ and $\Delta\tilde{\mathbf{s}}$ using eqn. (14).

(c) Set $\alpha = \min\left\{\frac{1}{2}, \ \left(\mu/16(\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2)\right)^{1/2}\right\}$

(d) Compute $(\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) + \alpha(\Delta\tilde{\mathbf{x}}_k, \Delta\tilde{\mathbf{y}}_k, \Delta\tilde{\mathbf{s}}_k)$.

Corrector Step $(\alpha = 1, \sigma = 1)$:

(e) Compute $\Delta\tilde{\mathbf{y}} = \texttt{Solve}(\mathbf{AD}^2\mathbf{A}^T, -\mu\mathbf{AS}^{-1}\mathbf{1}_n + \mathbf{Ax}, \delta_{\epsilon,n})$.

(f) Compute $\Delta\tilde{\mathbf{x}}$ and $\Delta\tilde{\mathbf{s}}$ using eqn. (14).

(g) Compute $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}, \mathbf{s}_{k+1}) = (\mathbf{x}_k, \mathbf{y}_k, \mathbf{s}_k) + (\Delta\tilde{\mathbf{x}}_k, \Delta\tilde{\mathbf{y}}_k, \Delta\tilde{\mathbf{s}}_k)$.

(h) $k \leftarrow k + 1$.

end while

We again proceed by expressing the difference of the exact vs. approximate solutions, using eqn. (7) vs. eqn. (14):

$$\Delta\mathbf{y} - \Delta\tilde{\mathbf{y}} = (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}, \qquad (20)$$

$$\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}} = -\mathbf{A}^T(\Delta\mathbf{y} - \Delta\tilde{\mathbf{y}})$$
$$= -\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}, \qquad (21)$$

$$\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}} = -\mathbf{D}^2(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}})$$
$$= \mathbf{D}^2\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}. \qquad (22)$$

Proving the convergence of Algorithm 3 follows essentially

the same high-level structure as for the error-adjusted algorithm described in Section 4, where now $\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2$ effectively measures the the error of the IPM steps and duality measure decrease due to the inexact solve in place of $\|\mathbf{v}\|_2$. By ensuring that the energy norm of the residual

$$\|\Delta\tilde{\mathbf{y}} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{AD}^2\mathbf{A}^T}$$

is sufficiently small, we ensure that, equivalently, $\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2$ is sufficiently small, allowing us to prove that the inexact solver converges to a point with duality measure less than ϵ, with only a constant factor increase in the number of outer iterations. The proofs of the lemmas for the predictor-corrector without error-adjustment are slightly simpler due to the fact that we do not need to account for the correction of the primal step (compare eqn. (22) to eqn. (18)). Therefore, we provide the proof for Theorem 1 first in the appendix, since it served as a starting point for our work.

We note that while guaranteeing that the energy-norm of the residual is small is sufficient for a fast decrease in the duality measure of the iterates, this does not necessarily imply that the iterates remain approximately feasible. As described in Appendix B, the maximum change in the primal infeasibility at each iteration is exactly $\|\mathbf{f}\|_2$, which is the ℓ_2-norm of the residual of the inexactly solved normal equations. Lemma D.1 shows that both the energy-norm and the ℓ_2-norm guarantee of `Solve` can be guaranteed using Algorithm 2.

6. Conclusions

We present and analyze an inexact predictor-corrector IPM algorithm that uses preconditioned inexact solvers to accelerate each iteration of the IPM, without increasing the number of iterations or sacrificing the feasibility of the returned solution. In future work, it is of interest to extend this framework to design fast and scalable algorithms for more general convex problems, such as semidefinite programming.

References

Almeida, R. and Teixeira, A. On the convergence of a predictor-corrector variant algorithm. *Top*, 23(2):401–418, 2015.

Applegate, D., Diaz, M. D., Hinder, O., Lu, H., Lubin, M., O'Donoghue, B., and Schudy, W. Practical large-scale linear programming using primal-dual hybrid gradient. In *Advances in Neural Information Processing Systems*, 2021.

Avron, H. and Toledo, S. Randomized algorithms for estimating the trace of an implicit symmetric positive semidefinite matrix. *Journal of the ACM (JACM)*, 58(2):1–34, 2011.

Avron, H., Maymounkov, P., and Toledo, S. Blendenpik: Supercharging LAPACK's least-squares solver. *SIAM Journal on Scientific Computing*, 32(3):1217–1236, 2010. ISSN 10648275. doi: 10.1137/090767911. URL http://epubs.siam.org/doi/abs/10.1137/090767911.

Barrett, R., Berry, M., Chan, T. F., Demmel, J., Donato, J., Dongarra, J., Eijkhout, V., Pozo, R., Romine, C., and Van der Vorst, H. *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.

Bastos, F. and Paixão, J. Interior-point approaches to the transportation and assignment problems on microcomputers. *Investigação Operacional*, 13(1):3–15, 1993.

Basu, K., Ghoting, A., Mazumder, R., and Pan, Y. ECLIPSE: An extreme-scale linear program solver for web-applications. In *International Conference on Machine Learning*, pp. 704–714, 2020.

Bellavia, S. Inexact interior-point method. *Journal of Optimization Theory and Applications*, 96(1):109–121, 1998.

Chowdhury, A., London, P., Avron, H., and Drineas, P. Faster randomized infeasible interior point methods for tall/wide linear programs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8704–8715, 2020.

Cohen, M. B., Nelson, J., and Woodruff, D. P. Optimal approximate matrix product in terms of stable rank. In *43rd International Colloquium on Automata, Languages, and Programming*, pp. 11:1–11:14, 2016.

Cohen, M. B., Lee, Y. T., and Song, Z. Solving linear programs in the current matrix multiplication time. *Journal of the ACM (JACM)*, 68(1):1–39, 2021.

Dahiya, Y., Konomis, D., and Woodruff, D. P. An empirical evaluation of sketching for numerical linear algebra. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1292–1300, 2018. doi: 10.1145/3219819.3220098.

Daitch, S. I. and Spielman, D. A. Faster approximate lossy generalized flow via interior point algorithms. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 451–460, 2008.

Donoho, D. L. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

Gondzio, J. Interior point methods 25 years later. *European Journal of Operational Research*, 218(3):587–601, 2012.

Gutknecht, M. H. *Software for numerical linear algebra*, volume 2. ETH Zurich, 2008. URL http://www.sam.math.ethz.ch/~mhg/unt/SWNLA/itmethSWNLA08.pdf.

Gutknecht, M. H. and Röllin, S. The Chebyshev iteration revisited. *Parallel Computing*, 28(2):263–283, 2002.

Halko, N., Martinsson, P.-G., and Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

Karmarkar, N. A new polynomial-time algorithm for linear programming. In *Proceedings of the 16th Annual ACM Symposium on Theory of Computing*, pp. 302–311, 1984.

Khachiyan, L. G. A polynomial algorithm in linear programming. In *Doklady Akademii Nauk*, volume 244, pp. 1093–1096. Russian Academy of Sciences, 1979.

Lee, Y. T. and Sidford, A. Solving linear programs with $\sqrt{rank}$ linear system solves. *arXiv preprint arXiv:1910.08033*, 2019.

Lin, T., Ma, S., Ye, Y., and Zhang, S. An ADMM-based interior-point method for large-scale linear programming. *Optimization Methods and Software*, 36(2-3):389–424, 2021.

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

Meshi, O. and Globerson, A. An alternating direction method for dual MAP LP relaxation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 470–483. Springer, 2011.

Mizuno, S. and Jarre, F. Global and polynomial-time convergence of an infeasible-interior-point algorithm using inexact computation. *Mathematical Programming*, 84(1), 1999.

Monteiro, R. D. and O'Neal, J. W. Convergence analysis of a long-step primal-dual infeasible interior-point lp algorithm based on iterative linear solvers. *Georgia Institute of Technology*, 2003.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge University Press, 2007.

Quarteroni, A. and Valli, A. *Numerical Approximation of Partial Differential Equations*. Springer Series in Computational Mathematics, 1996. ISBN 9783540852674. doi: 10.1007/978-3-540-85268-1.

Recht, B., Re, C., Tropp, J., and Bittorf, V. Factoring non-negative matrices with linear programs. In *Advances in Neural Information Processing Systems*, pp. 1214–1222, 2012.

Salahi, M., Peng, J., and Terlaky, T. On mehrotra-type predictor-corrector algorithms. *SIAM Journal on Optimization*, 18(4):1377–1397, 2008.

Schork, L. and Gondzio, J. Implementation of an interior point method with basis preconditioning. *Mathematical Programming Computation*, 12(4):603–635, 2020.

Song, Z. and Yu, Z. Oblivious sketching-based central path method for linear programming. In *International Conference on Machine Learning*, pp. 9835–9847, 2021.

Ubaru, S. and Saad, Y. Fast methods for estimating the numerical rank of large matrices. In *International Conference on Machine Learning*, pp. 468–477. PMLR, 2016.

van den Brand, J., Lee, Y. T., Sidford, A., and Song, Z. Solving tall dense linear programs in nearly linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 775–788, 2020.

Wong, E. and Kolter, Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5286–5295. PMLR, 2018.

Wright, S. J. *Primal-dual interior-point methods*. SIAM, 1997.

Yang, J. and Zhang, Y. Alternating direction algorithms for ℓ_1-problems in compressive sensing. *SIAM Journal on Scientific Computing*, 33(1):250–278, 2011.

Yang, Y. and Yamashita, M. An arc-search $\mathcal{O}(nL)$ infeasible-interior-point algorithm for linear programming. *Optimization Letters*, 12(4):781–798, 2018.

Yuan, M. High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, 11(Aug):2261–2286, 2010.

Zhu, J., Rosset, S., Tibshirani, R., and Hastie, T. J. 1-norm support vector machines. In *Advances in Neural Information Processing Systems*, pp. 49–56, 2004.

On the Convergence of Inexact Predictor-Corrector Methods for Linear Programming (Appendix)

A. Additional Proofs

Lemma A.1. *If the duality measure of the IPM decreases with the relation,*

$$\mu_1 \leq \left(1 - \frac{C_0}{\sqrt{n}}\right)\mu_0 + C_1\epsilon \quad \text{for } C_0 \in (0,1), \ C_1 \in [0, \frac{C_0}{\sqrt{n}}),$$

for all $\mu_0 \geq 2\epsilon$, then the IPM algorithm converges to a point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ with duality measure $\mu^* < 2\epsilon$ in $\frac{\sqrt{n}}{C_0}\log\frac{\mu_0}{\epsilon}$ outer iterations.

Proof. Each algorithm terminates when $\mu_k < 2\epsilon$ indicating convergence has been reached. Therefore, we assume that at each iteration $\mu_k \geq 2\epsilon$.

Although the starting point is assumed to be feasible. This fact can be ignored in the convergence analysis. The constraint matrix of the linear program is assumed to be full rank and the system is undetermined. This implies that for all $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0)$ there exists a vector $\mathbf{b}$ so that the starting point is feasible. Since the duality measure does not depend on $\mathbf{b}$, the decrease in the duality gap occurs whether or not the starting point is feasible.

Given this, we can define a recurrence relation $\mathcal{T}(k)$ such that $\mu_k \leq \mathcal{T}(k)$, where,

$$\mathcal{T}(k) = \left(1 - \frac{C_0}{n^p}\right)\mathcal{T}(k-1) + C_1\epsilon \quad \mathcal{T}(0) = \mu_0.$$

If we define $\xi = \left(1 - \frac{C_0}{\sqrt{n}}\right)$, then we have a recurrence relation of the form $\mathcal{T}(k) = \xi\mathcal{T}(k-1) + C_1\epsilon$. The solution to the recurrence relation is.

$$\mathcal{T}(k) = \frac{\epsilon C_1(1 - \xi^k)}{1 - \xi} + \xi^k\mu_0$$

$$\leq \frac{\epsilon C_1}{C_0/n^p} + \xi^k\mu_0$$

$$\leq \epsilon + \xi^k\mu_0.$$

Therefore, $\mu_k \leq T(k) \leq 2\epsilon$ if $\xi^k\mu_0 \leq \epsilon$. We can prove the outer iteration complexity using a standard argument by substituting back in $\left(1 - \frac{C_0}{\sqrt{n}}\right)$ for ξ and using the identity $\log(1 + \beta) \leq \beta$ for all $\beta > -1$. We prove that $\mu_k < 2\epsilon$ if $k \geq \frac{\sqrt{n}}{C_0}\log\frac{\mu_0}{\epsilon}$.

$$k \geq \frac{\sqrt{n}}{C_0}\log\frac{\mu_0}{\epsilon}$$

$$\Rightarrow \frac{-kC_0}{\sqrt{n}} \leq \log\frac{\epsilon}{\mu_0}$$

$$\Rightarrow k\log\left(1 - \frac{C_0}{\sqrt{n}}\right) \leq \log\frac{\epsilon}{\mu_0}$$

$$\left(1 - \frac{C_0}{\sqrt{n}}\right)^k\mu_0 \leq \epsilon$$

Therefore, the IPM algorithm is guaranteed to converge in $\frac{\sqrt{n}}{C_0} \log \frac{\mu_0}{\epsilon}$ outer iterations. □

Lemma A.2. *Let* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$.

$$\|\mathbf{u} \circ \mathbf{v}\|_2 \le \|\mathbf{u}\|_2 \|\mathbf{v}\|_2$$

Proof.

$$\|\mathbf{u} \circ \mathbf{v}\|_2^2 = \sum_{i=1}^n (\mathbf{u}_i \mathbf{v}_i)^2 = \sum_{i=1}^n \mathbf{u}_i^2 \mathbf{v}_i^2 \le \left(\sum_{i=1}^n \mathbf{u}_i^2 \right) \left(\sum_{i=1}^n \mathbf{v}_i^2 \right) = \|\mathbf{u}\|_2^2 \|\mathbf{v}\|_2^2.$$

□

Lemma A.3. *If* $\mathbf{M}$ *is an* $m \times n$ *matrix of full row rank and* $\mathbf{x}$ *is an arbitrary vector in the row space of* $\mathbf{M}$*, then* $\|\mathbf{M}\mathbf{x}\|_2 \ge \sigma_m(\mathbf{M})\|\mathbf{x}\|_2$.

Proof. Let $\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ where $\mathbf{U}, \boldsymbol{\Sigma} \in \mathbb{R}^{m \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times m}$ constitute the thin-SVD of $\mathbf{M}$. Since $\mathbf{U}$ is an orthonormal matrix, multiplying a vector by $\mathbf{U}$ does not change the ℓ_2-norm. Therefore, we have the following with $\mathbf{y} = \mathbf{V}^T\mathbf{x}$:

$$\|\mathbf{M}\mathbf{x}\|_2^2 = \|\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T\mathbf{x}\|_2^2 = \|\boldsymbol{\Sigma}\mathbf{V}^T\mathbf{x}\|_2^2$$
$$= \|\boldsymbol{\Sigma}\mathbf{y}\|_2^2 = \sum_{i=1}^m \sigma_i(\mathbf{M})^2 \mathbf{y}_i^2$$
$$\ge \sigma_m(\mathbf{M})^2 \sum_{i=1}^m \mathbf{y}_i^2 \ge \sigma_m(\mathbf{M})^2 \|\mathbf{y}\|_2^2,$$

The vectors $\mathbf{y}$ and $\mathbf{x}$ have the same norm since $\mathbf{y}^T\mathbf{y} = \mathbf{x}^T\mathbf{V}\mathbf{V}^T\mathbf{x} = \mathbf{x}^T\mathbf{x}$, and the columns of $\mathbf{V}$ are orthonormal. Therefore, $\|\mathbf{M}\mathbf{x}\|_2^2 \ge \sigma_m(\mathbf{M})^2 \|\mathbf{x}\|_2^2$.

□

Lemma A.4. *(Simplification of Lemma 12 in (*Chowdhury et al., 2020*)) If* $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$*, then,*

$$\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2 \le \sigma \sqrt{\frac{2n\mu}{1-\theta}} + \sqrt{2n\mu}.$$

Proof. By Lemma 7 in (Chowdhury et al., 2020), if the condition given by eqn. (33) is fulfilled by sketching matrix $\mathbf{W}$, then $\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}^2\mathbf{A}^T\mathbf{Q}^{-1/2} - \mathbf{I}_m\|_2 \le \zeta$. Since $\zeta \in (0,1)$, this imples $\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}^2\mathbf{A}^T\mathbf{Q}^{-1/2}\|_2 \le 2$ and so $\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}\|_2 \le \sqrt{2}$. Recall that $\mathbf{p} = -\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n + \mathbf{A}\mathbf{x}$. We can split the terms of $\mathbf{Q}^{-1/2}\mathbf{p}$ by the triangle inequality and then bound these terms separately below. We begin with the first term:

$$\|\sigma\mu\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n\|_2 = \sigma\mu\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{1}_n\|_2$$
$$\le \sigma\mu\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}\|_2\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{1}_n\|_2$$
$$\le \sqrt{2}\sigma\mu\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{1}_n\|_2$$
$$\le \sqrt{2}\sigma\mu\sqrt{\frac{n}{\min \mathbf{x}_i\mathbf{s}_i}}$$
$$\le \sqrt{2}\sigma\mu\sqrt{\frac{n}{(1-\theta)\mu}} = \sigma\sqrt{\frac{2n\mu}{1-\theta}}.$$

Next, we bound the second term:

$$
\begin{aligned}
\|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{x}\|_2 &= \|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}\mathbf{D}^{-1}\mathbf{x}\|_2 \\
&= \|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}(\mathbf{S}^{1/2}\mathbf{X}^{-1/2})\mathbf{X}\mathbf{1}_n\|_2 \\
&= \|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}(\mathbf{S}\mathbf{X})^{-1/2}\mathbf{1}_n\|_2 \\
&\leq \|\mathbf{Q}^{-1/2}\mathbf{A}\mathbf{D}\|_2\|(\mathbf{S}\mathbf{X})^{-1/2}\mathbf{1}_n\|_2 \\
&\leq \sqrt{2}\sqrt{\sum_{i=1}^{n}\mathbf{x}_i\mathbf{s}_i} = \sqrt{2n\mu}.
\end{aligned}
$$

By adding the bounds together, we conclude that:

$$
\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2 \leq \sigma\sqrt{\frac{2n\mu}{1-\theta}} + \sqrt{2n\mu}.
$$

$\square$

B. Inexact Predictor-Corrector IPM

In this section, we analyze the *inexact* predictor-corrector method (Algorithm 3) using an inexact linear solver `Solve`. We follow the proof outline given in Section 3 to prove the convergence guarantees and time complexity of Algorithm 3. As is common in predictor-corrector IPMs (see (Wright, 1997)), we will assume that the matrix $\mathbf{A}$ has full row rank, ie., $rank(\mathbf{A}) = m \leq n$. See Appendix D.3 for extensions alleviating this constraint.

We will prove that Algorithm 3 converges to a point $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$ satisfying $\mu^* < 2\epsilon$ and $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 < \epsilon$ in $\mathcal{O}(\sqrt{n}\log^{\mu_0}/\epsilon)$ outer iterations. First, we start with a technical result to bound $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$.

Lemma B.1. *Let $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$ and let $(\Delta\tilde{\mathbf{x}}, \Delta\tilde{\mathbf{y}}, \Delta\tilde{\mathbf{s}})$ denote step calculated from the inexact normal equations (see eqn. (14)). Then*

$$
\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 2\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{A}\mathbf{D})^{\dagger}\mathbf{f}\|_2 + \|(\mathbf{A}\mathbf{D})^{\dagger}\mathbf{f}\|_2^2.
$$

Proof. We start by expressing the inexact step as the difference from the corresponding exact step, using eqns. (21) and (22). This will allow us to leverage results for exact predictor-corrector IPMs in our proof:

$$
\begin{aligned}
\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 &= \|[\Delta\mathbf{x} - (\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})] \circ [\Delta\mathbf{s} - (\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}})]\|_2 \\
&\leq \|\Delta\mathbf{x} \circ \Delta\mathbf{s}\|_2 + \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}) - \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f})\|_2 \\
&\quad + \|(\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}) \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f})\|_2.
\end{aligned}
$$

We will bound each of the three terms in the last inequality separately. Let $\mathcal{B}_1 = \|\Delta\mathbf{x} \circ \Delta\mathbf{s}\|_2$; $\mathcal{B}_2 = \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}) - \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f})\|_2$; and $\mathcal{B}_3 = \|(\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}) \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f})\|_2$. Using eqn. (13), we get a bound on $\mathcal{B}_1$:

$$
\mathcal{B}_1 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu.
$$

Next, we bound $\mathcal{B}_2$. First, we rearrange $\mathcal{B}_2$ using properties of the Hadamard product (see Lemma A.2) and Moore-Penrose pseudoinverse:

$$
\begin{aligned}
\mathcal{B}_2 &= \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}) - \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f})\|_2 \\
&= \|[\mathbf{D}^{-1}\Delta\mathbf{x}] \circ [\mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}] - [\mathbf{D}\Delta\mathbf{s}] \circ [\mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}]\|_2 \\
&= \|[\mathbf{D}^{-1}\Delta\mathbf{x} - \mathbf{D}\Delta\mathbf{s}] \circ [\mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}]\|_2 \\
&\leq \|\mathbf{D}^{-1}\Delta\mathbf{x} - \mathbf{D}\Delta\mathbf{s}\|_2\|(\mathbf{A}\mathbf{D})^{\dagger}\mathbf{f}\|_2. \tag{23}
\end{aligned}
$$

We now bound $\|\mathbf{D}^{-1}\Delta\mathbf{x} - \mathbf{D}\Delta\mathbf{s}\|_2$ by using the definitions of $\Delta\mathbf{y}$, $\Delta\mathbf{s}$ and $\Delta\mathbf{x}$ given in eqn. (7). Thus,

$$
\begin{aligned}
\mathbf{D}^{-1}\Delta\mathbf{x} - \mathbf{D}\Delta\mathbf{s} &= \mathbf{D}^{-1}(-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{D}^2\Delta\mathbf{s}) - \mathbf{D}\Delta\mathbf{s} \\
&= \mathbf{D}^{-1}(-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n) - 2\mathbf{D}\Delta\mathbf{s} \\
&= \mathbf{D}^{-1}(-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n) - 2(\mathbf{AD})^{\dagger}\mathbf{A}(-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n) \\
&= (\mathbf{D}^{-1} - 2(\mathbf{AD})^{\dagger}\mathbf{A})(-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n) \\
&= (\mathbf{D}^{-1} - 2(\mathbf{AD})^{\dagger}\mathbf{A})\mathbf{D}\mathbf{D}^{-1}(-\mathbf{X}\mathbf{1}_n + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n) \\
&= (\mathbf{I} - 2(\mathbf{AD})^{\dagger}\mathbf{AD})[(\mathbf{XS})^{-1/2}(-\mathbf{XS}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)].
\end{aligned}
$$

At this point, we have shown that

$$
\mathcal{B}_2 \le \|\mathbf{I} - 2(\mathbf{AD})^{\dagger}\mathbf{AD}\|_2\|(\mathbf{XS})^{-1/2}(-\mathbf{XS}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)\|_2\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2. \tag{24}
$$

We can now use the fact that the two-norm of $\mathbf{I} - 2(\mathbf{AD})^{\dagger}\mathbf{AD} = \mathbf{I} - (\mathbf{AD})^{\dagger}\mathbf{AD} - (\mathbf{AD})^{\dagger}\mathbf{AD}$ is upper bounded by two, since $\mathbf{I} - (\mathbf{AD})^{\dagger}\mathbf{AD}$ and $(\mathbf{AD})^{\dagger}\mathbf{AD}$ are both projection matrices and have two-norm at most one. Next, we bound the middle term, namely $\|(\mathbf{XS})^{-1/2}(-\mathbf{XS}e + \sigma\mu\mathbf{1}_n)\|_2$; the proof will use the fact that $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$, which implies that $\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2 \le \theta\mu$:

$$
\begin{aligned}
\|(\mathbf{XS})^{-1/2}(-\mathbf{XS}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)\|_2^2 &= \sum_{i=1}^{n} \frac{(-\mathbf{x}_i\mathbf{s}_i + \sigma\mu)^2}{\mathbf{x}_i\mathbf{s}_i} \le \frac{\|\mathbf{x} \circ \mathbf{s} - \sigma\mu\mathbf{1}_n\|_2^2}{\min_i \mathbf{x}_i\mathbf{s}_i} \\
&\le \frac{\|\mathbf{x} \circ \mathbf{s} - \sigma\mu\mathbf{1}_n\|_2^2}{(1-\theta)\mu} \le \frac{\|(\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n) + (1-\sigma)\mu\mathbf{1}_n\|_2^2}{(1-\theta)\mu} \\
&\le \frac{[(\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n) + (1-\sigma)\mu\mathbf{1}_n]^T[(\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n) + (1-\sigma)\mu\mathbf{1}_n]}{(1-\theta)\mu} \\
&\le \frac{\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2^2 + 2(1-\sigma)\mu\mathbf{1}_n^T(\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n) + (1-\sigma)^2\mu^2 n}{(1-\theta)\mu} \\
&\le \frac{\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2^2 + 2(1-\sigma)\mu(n\mu - n\mu) + (1-\sigma)^2\mu^2 n}{(1-\theta)\mu} \\
&\le \frac{\theta^2\mu^2 + (1-\sigma)^2\mu^2 n}{(1-\theta)\mu} \le \frac{(\theta^2 + (1-\sigma)^2 n)\mu}{(1-\theta)}.
\end{aligned}
$$

Inserting the bounds for $\|\mathbf{I} - 2(\mathbf{AD})^{\dagger}\mathbf{AD}\|_2$ and $\|(\mathbf{XS})^{-1/2}(-\mathbf{XS}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)\|_2$ into the previous bound for $\mathcal{B}_2$ gives:

$$
\mathcal{B}_2 \le 2\sqrt{\frac{(\theta^2 + (1-\sigma)^2 n)\mu}{(1-\theta)}}\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2.
$$

Finally, we bound $\mathcal{B}_3$ using properties of the Hadamard product (see Lemma A.2):

$$
\begin{aligned}
\mathcal{B}_3 &= \|[\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}] \circ [\mathbf{D}^2\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}]\|_2 \\
&= \|[\mathbf{DA}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}] \circ [\mathbf{DA}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}]\|_2 \\
&= \|\mathbf{DA}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}\|_2^2 \\
&= \|(\mathbf{AD})^T((\mathbf{AD})(\mathbf{AD})^T)^{-1}\mathbf{f}\|_2^2 \\
&= \|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2^2. \tag{25}
\end{aligned}
$$

Adding the bounds on $\mathcal{B}_1$, $\mathcal{B}_2$, and $\mathcal{B}_3$ gives the final inequality:

$$
\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \le \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 2\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2 + \|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2^2.
$$

$\square$

The next lemma will bound the value of $\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2$, which will allow us to later show that the iterates remain in the correct neighborhood for a given step size.

Lemma B.2. *If $\alpha \in [0,1]$, then*

$$\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2 + \alpha^2\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + \alpha^2\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2^2.$$

Proof. First, we derive an expression for the difference in the duality measure between the exact and inexact step, $(\mu(\alpha) - \tilde{\mu}(\alpha))$. We begin by expanding the definition of $n(\mu(\alpha) - \tilde{\mu}(\alpha))$.

$$
\begin{aligned}
n(\mu(\alpha) - \tilde{\mu}(\alpha)) &= (\mathbf{x} + \alpha\Delta\mathbf{x})^T(\mathbf{s} + \alpha\Delta\mathbf{s}) - (\mathbf{x} + \alpha\Delta\tilde{\mathbf{x}})^T(\mathbf{s} + \alpha\Delta\tilde{\mathbf{s}}) \\
&= (\mathbf{x}^T\mathbf{s} + \alpha\Delta\mathbf{x}^T\mathbf{s} + \mathbf{x}^T\alpha\Delta\mathbf{s} + \alpha^2\Delta\mathbf{x}^T\Delta\mathbf{s}) - (\mathbf{x}^T\mathbf{s} + \alpha\Delta\tilde{\mathbf{x}}^T\mathbf{s} + \mathbf{x}^T\alpha\Delta\tilde{\mathbf{s}} + \alpha^2\Delta\tilde{\mathbf{x}}^T\Delta\tilde{\mathbf{s}}) \\
&= \alpha(\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})^T\mathbf{s} + \alpha\mathbf{x}^T(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}}) + \alpha^2(\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})^T(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}})
\end{aligned}
$$

We next substitute the differences between the exact and inexact steps given by eqns. (21, 22). We do this in two parts for clearer exposition.

$$
\begin{aligned}
\alpha(\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})^Ts + \alpha\mathbf{x}^T(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}}) &= \alpha[\mathbf{D}^2\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}]^T\mathbf{s} + \alpha\mathbf{x}^T[-\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}] \\
&= \alpha\mathbf{f}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{AD}^2\mathbf{s} - \alpha\mathbf{x}^T\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f} \\
&= \alpha\mathbf{f}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{AXS}^{-1}\mathbf{s} - \alpha\mathbf{x}^T\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f} \\
&= \alpha\mathbf{f}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{Ax} - \alpha\mathbf{x}^T\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f} \\
&= 0.
\end{aligned}
$$

Therefore, substituting this result into the previous set of equations, we get the following.

$$
\begin{aligned}
n(\mu(\alpha) - \tilde{\mu}(\alpha)) &= \alpha^2(\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})^T(\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}}) \\
&= \alpha^2[\mathbf{D}^2\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}]^T[-\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}] \\
&= -\alpha^2\mathbf{f}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}(\mathbf{AD}^2\mathbf{A}^T)(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f} \\
&= -\alpha^2\mathbf{f}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f} \Rightarrow
\end{aligned}
$$

$$\mu(\alpha) - \tilde{\mu}(\alpha) = \frac{-\alpha^2}{n}\|(\mathbf{AD})^{\dagger}\mathbf{f}\|_2^2, \tag{26}$$

where the final line follows from properties of the pseudoinverse. Next, we prove two identities that will be useful. For the first identity, we expand the term $\mathbf{x}_i(\alpha)\mathbf{s}_i(\alpha)$ and insert eqn. (12) to cancel terms:

$$
\begin{aligned}
\mathbf{x}_i(\alpha)\mathbf{s}_i(\alpha) - \mu(\alpha) &= \mathbf{x}_i\mathbf{s}_i + \alpha(\mathbf{x}_i\Delta\mathbf{s}_i + \Delta\mathbf{x}_i\mathbf{s}_i) + \alpha^2\Delta\mathbf{x}_i\Delta\mathbf{s}_i - (1 - \alpha(1-\sigma))\mu \\
&= (1-\alpha)\mathbf{x}_i\mathbf{s}_i + \alpha\sigma\mu + \alpha^2\Delta\mathbf{x}_i\Delta\mathbf{s}_i - (1 - \alpha + \alpha\sigma)\mu \\
&= (1-\alpha)(\mathbf{x}_i\mathbf{s}_i - \mu) + \alpha^2\Delta\mathbf{x}_i\Delta\mathbf{s}_i.
\end{aligned}
\tag{27}
$$

For the second identity, we expand terms and substitute eqns. (21) and (22) for $\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}}$ and $\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}}$, respectively:

$$
\begin{aligned}
&\mathbf{x}_i(\alpha)\mathbf{s}_i(\alpha) - \tilde{\mathbf{x}}(\alpha)_i\tilde{\mathbf{s}}(\alpha)_i \\
&= \mathbf{x}_i\mathbf{s}_i + \alpha(\mathbf{x}_i\Delta\mathbf{s}_i + \Delta\mathbf{x}_i s_i) + \alpha^2\Delta\mathbf{x}_i\Delta\mathbf{s}_i - [\mathbf{x}_i\mathbf{s}_i + \alpha(\mathbf{x}_i\Delta\tilde{\mathbf{s}}_i + \Delta\tilde{\mathbf{x}}_i s_i) + \alpha^2\Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i] \\
&= \alpha[\mathbf{x}_i(\Delta\mathbf{s}_i - \Delta\tilde{\mathbf{s}}_i) + \mathbf{s}_i(\Delta\mathbf{x}_i - \Delta\tilde{\mathbf{x}}_i)] + \alpha^2(\Delta\mathbf{x}_i\Delta\mathbf{s}_i - \Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i) \\
&= \alpha[\mathbf{x} \circ (-\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}) + \mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f})]_i + \alpha^2(\Delta\mathbf{x}_i\Delta\mathbf{s}_i - \Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i) \\
&= \alpha[\mathbf{x} \circ (-\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}) + \mathbf{x} \circ (\mathbf{A}^T(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f})]_i + \alpha^2(\Delta\mathbf{x}_i\Delta\mathbf{s}_i - \Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i) \\
&= \alpha^2(\Delta\mathbf{x}_i\Delta\mathbf{s}_i - \Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i).
\end{aligned}
\tag{28}
$$

using the above two identities, we expand and rearrange $\tilde{\mathbf{x}}(\alpha)_i \tilde{\mathbf{s}}(\alpha)_i - \tilde{\mu}(\alpha)$ to get:

$$
\begin{aligned}
\tilde{\mathbf{x}}(\alpha)_i \tilde{\mathbf{s}}(\alpha)_i - \tilde{\mu}(\alpha) &= \mathbf{x}(\alpha)_i \mathbf{s}(\alpha)_i - (\mathbf{x}(\alpha)_i \mathbf{s}(\alpha)_i - \tilde{\mathbf{x}}(\alpha)_i \tilde{\mathbf{s}}(\alpha)_i) - [\mu(\alpha) - (\mu(\alpha) - \tilde{\mu}(\alpha))] \\
&= [\mathbf{x}(\alpha)_i \mathbf{s}(\alpha)_i - \mu(\alpha)] - (\mathbf{x}(\alpha)_i \mathbf{s}(\alpha)_i - \tilde{\mathbf{x}}(\alpha)_i \tilde{\mathbf{s}}(\alpha)_i) + (\mu(\alpha) - \tilde{\mu}(\alpha)) \\
&= [(1-\alpha)(\mathbf{x}_i \mathbf{s}_i - \mu) + \alpha^2 \Delta \mathbf{x}_i \Delta \mathbf{s}_i] - (\mathbf{x}(\alpha)_i \mathbf{s}(\alpha)_i - \tilde{\mathbf{x}}(\alpha)_i \tilde{\mathbf{s}}(\alpha)_i) + (\mu(\alpha) - \tilde{\mu}(\alpha)) \\
&= (1-\alpha)(\mathbf{x}_i \mathbf{s}_i - \mu) + \alpha^2 \Delta \mathbf{x}_i \Delta \mathbf{s}_i - \alpha^2 (\Delta \mathbf{x}_i \Delta \mathbf{s}_i - \Delta \tilde{\mathbf{x}}_i \Delta \tilde{\mathbf{s}}_i) + (\mu(\alpha) - \tilde{\mu}(\alpha)) \\
&= (1-\alpha)(\mathbf{x}_i \mathbf{s}_i - \mu) + \alpha^2 \Delta \tilde{\mathbf{x}}_i \Delta \tilde{\mathbf{s}}_i + (\mu(\alpha) - \tilde{\mu}(\alpha)).
\end{aligned}
$$

Taking vector norms of the above element-wise equality and substituting eqn. (26) for $\mu(\alpha) - \tilde{\mu}(\alpha)$, we conclude,

$$
\begin{aligned}
\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha) \mathbf{1}_n\|_2 &\leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu \mathbf{1}_n\|_2 + \alpha^2 \|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2 + \|(\mu(\alpha) - \tilde{\mu}(\alpha))\mathbf{1}_n\|_2 \\
&\leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu \mathbf{1}_n\|_2 + \alpha^2 \|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2 + \alpha^2 \|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2.
\end{aligned}
$$

$\square$

Next, we prove that the predictor stays in the slightly enlarged neighborhood $\mathcal{N}_2(0.5)$ when starting from the smaller neighborhood $\mathcal{N}_2(0.25)$. We will later show that the corrector step guarantees that the iterate "returns" to the "correct" $\mathcal{N}_2(0.25)$ neighborhood. Recall that $\sigma = 0$ in the predictor step to get the following lemma.

Lemma B.3. *If* $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.25)$,

$$
\alpha = \min\left\{ 1/2, \left(\frac{\mu}{16(\|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2)} \right)^{1/2} \right\}, \quad \|(\mathbf{AD})^\dagger \mathbf{f}\|_2 \leq \frac{\sqrt{\mu}}{2},
$$

then the predictor step $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$.

Proof. We begin with the inequality of Lemma B.2. Recall that $\|\mathbf{x} \circ \mathbf{s} - \mu \mathbf{1}_n\|_2 \leq \mu/4$ since $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}(0.25)$:

$$
\begin{aligned}
\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 &\leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu \mathbf{1}_n\|_2 + \alpha^2\|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2 + \alpha^2\|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2 \\
&\leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu \mathbf{1}_n\|_2 + \frac{\mu(\|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2)}{16(\|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2} + \frac{1}{2^2}\frac{\mu}{4} \\
&\leq \frac{(1-\alpha)\mu}{4} + \frac{1}{8(1-\alpha)}(1-\alpha)\mu \quad (\text{since } (\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.25)) \\
&\leq \frac{(1-\alpha)\mu}{4} + \frac{(1-\alpha)\mu}{4} \quad (\text{since } \alpha \leq 1/2) \\
&\leq \frac{1}{2}(1-\alpha)\mu \leq \frac{1}{2}\tilde{\mu}(\alpha).
\end{aligned}
$$

The last line follows from eqn. (12), which gives $\mu(\alpha) = (1-\alpha)\mu$ for $\sigma = 0$. We also know from eqn. (26) that $\tilde{\mu}(\alpha) \geq \mu(\alpha)$. Therefore, $\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq 1/2\,\tilde{\mu}(\alpha)$. Now, we must show that the condition $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{s}}(\alpha)) > 0$ is fulfilled. First, by eqns. (12, 26), we have $\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu + \frac{\alpha^2}{n}\|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2$, which shows $\tilde{\mu}(\alpha') > 0$ for all positive step sizes $\alpha' \leq \alpha$. From the first part of this proof, we have that $\tilde{\mathbf{x}}_i(\alpha)\tilde{\mathbf{s}}_i(\alpha) \geq 1/2\tilde{\mu}(\alpha)$. We conclude that $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$.

$\square$

The next lemma shows that the step size given in the previous lemma will result in a multiplicative decrease in the duality gap during the predictor step.

Lemma B.4. *If* $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.25)$,

$$
\alpha = \min\left\{ 1/2, \left(\frac{\mu}{16(\|\Delta \tilde{\mathbf{x}} \circ \Delta \tilde{\mathbf{s}}\|_2)} \right)^{1/2} \right\}, \quad \text{and} \quad \|(\mathbf{AD})^\dagger \mathbf{f}\|_2 \leq \frac{\sqrt{\mu}}{8},
$$

then the predictor step $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$ *and there exists a constant* $C_0 \in (0, 1)$ *such that,*

$$
\frac{\tilde{\mu}(\alpha)}{\mu} \leq 1 - \frac{C_0}{\sqrt{n}}.
$$

We note that in the above lemma $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha))$ is the output after the predictor step *only*, before the corrector step is applied.

Proof. To lower bound the decrease in the duality gap at each step, we first find a lower bound for the step size α, using the upper bound for $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$. By Lemma B.1, we have the following inequality:

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 2\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{AD})^\dagger\mathbf{f}\|_2 + \|(\mathbf{AD})^\dagger\mathbf{f}\|_2^2.$$

We can simplify this inequality by substituting $\|(\mathbf{AD})^\dagger\mathbf{f}\|_2 \leq \sqrt{\mu}/8$, $\theta = 0.25$, and $\sigma = 0$ to get:

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{(0.25)^2 + n(1-0)^2}{2^{3/2}(1-0.25)}\mu + 2\sqrt{\frac{(0.25^2 + n(1-0)^2)\mu}{(1-0.25)}}\frac{\sqrt{\mu}}{8} + \frac{\mu}{64}$$

$$\leq \frac{1/16 + n}{2^{3/2}(3/4)}\mu + 2\sqrt{\frac{(1/16 + n)\mu}{3/4}}\frac{\sqrt{\mu}}{8} + \frac{\mu}{64}$$

$$\leq n\mu\left(\frac{1/16 + 1}{2^{3/2}(3/4)} + 2\sqrt{\frac{(1/16 + 1)}{3/4}}\frac{1}{8} + \frac{1}{64}\right)$$

$$\leq 0.82 \cdot n\mu.$$

We can now insert the upper bounds for $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$ and $\|(\mathbf{AD})^\dagger\mathbf{f}\|_2$ to lower bound α:

$$\alpha = \min\left\{1/2, \left(\frac{\mu}{8(\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + \|(\mathbf{AD})^\dagger\mathbf{f}\|_2^2)}\right)^{1/2}\right\}$$

$$\geq \min\left\{1/2, \left(\frac{\mu}{8(0.82)n\mu + \mu/8}\right)^{1/2}\right\}$$

$$\geq \min\left\{1/2, \left(\frac{1}{n(6.56 + 1/8)}\right)^{1/2}\right\}$$

$$\geq \sqrt{0.14/n}.$$

We proceed by using eqns. (12) and (26) to upper bound $\tilde{\mu}(\alpha)$. We substitute the upper bound for $\|(\mathbf{AD})^\dagger\mathbf{f}\|_2$ as well as the upper and lower bounds for α as needed to derive a worst case bound. Recall that the step size α is upper bounded by $1/2$ (by definition):

$$\tilde{\mu}(\alpha) \leq \mu(\alpha) - [\mu(\alpha) - \tilde{\mu}(\alpha)] \leq (1-\alpha)\mu + \frac{\alpha^2}{n}\|(\mathbf{AD})^\dagger\mathbf{f}\|_2^2$$

$$\leq (1-\alpha)\mu + \frac{\alpha^2\mu}{64n} \leq \left(1 - \sqrt{\frac{0.14}{n}}\right)\mu + \frac{\mu}{4 \cdot 64n}$$

$$\leq \left[\left(1 - \sqrt{\frac{0.14}{n}}\right) + \frac{1}{4 \cdot 64\sqrt{n}}\right]\mu \leq \left[1 - \frac{\sqrt{0.14} - (1/256)}{\sqrt{n}}\right]\mu.$$

Then, if we let $C_0 = \sqrt{0.14} - (1/256)$, we get

$$\frac{\tilde{\mu}(\alpha)}{\mu} \leq 1 - \frac{C_0}{\sqrt{n}}, \quad \text{with } C_0 \in (0, 1).$$

$\square$

So far, We have shown that the predictor step results in a multiplicative decrease in the duality gap while keeping the next iterate in the neighborhood $\mathcal{N}_2(0.5)$. We now show that the corrector step returns the iterate to the $\mathcal{N}_2(0.25)$ neighborhood while increasing the duality gap by only a small additive amount.

Lemma B.5. *Let* $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.5)$ *and* $\|(\mathbf{AD})^\dagger \mathbf{f}\|_2 \leq \sqrt{\mu}/2^6$. *Then, the corrector step* $(\tilde{\mathbf{x}}(1), \tilde{\mathbf{y}}(1), \tilde{\mathbf{s}}(1)) \in \mathcal{N}_2(0.25)$ *and* $|\tilde{\mu}(1) - \mu| \leq \|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2/n$.

Proof. First, we show that taking a step with step size $\alpha = 1$ and centering parameter $\sigma = 1$ from a point in $\mathcal{N}_2(0.5)$ "returns" that point to the smaller neighborhood $\mathcal{N}_2(0.25)$. We start with the inequality given by Lemma B.2 with $\alpha = 1$ and then substitute the bound for $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$ from Lemma B.1 to get:

$$\|\tilde{\mathbf{x}}(1) \circ \tilde{\mathbf{s}}(1) - \tilde{\mu}(1)\|_2 \leq \|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + \|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2$$

$$\leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 2\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{AD})^\dagger \mathbf{f}\|_2 + 2\|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2.$$

Next, we use $\|(\mathbf{AD})^\dagger \mathbf{f}\|_2 \leq \sqrt{\mu}/2^6$, $\theta = 0.5$, and $\sigma = 1$:

$$\|\tilde{\mathbf{x}}(1) \circ \tilde{\mathbf{s}}(1) - \tilde{\mu}(1)\|_2 \leq \frac{(0.5)^2 + n(1-1)^2}{2^{3/2}(1-0.5)}\mu + 2\sqrt{\frac{((0.5)^2 + n(1-1)^2)\mu}{(1-0.5)}}\frac{\sqrt{\mu}}{2^6} + \frac{2\mu}{2^{12}}$$

$$\leq \frac{(1/4)}{2^{3/2}(1/2)}\mu + 2\sqrt{\frac{(1/4)\mu}{(1/2)}}\frac{\sqrt{\mu}}{2^6} + \frac{2\mu}{2^{12}}$$

$$\leq \frac{\mu}{2^{9/2}} + \frac{\mu}{2^{11/2}} + \frac{\mu}{2^{11}} \leq \frac{1}{4}\mu.$$

Again, $\tilde{\mu}(\alpha) \geq \mu(\alpha)$ which implies that $\|\tilde{\mathbf{x}}(1) \circ \tilde{\mathbf{s}}(1) - \tilde{\mu}(1)\|_2 \leq 1/4\,\tilde{\mu}(1)$, so we can conclude that $(\tilde{\mathbf{x}}(1), \tilde{\mathbf{y}}(1), \tilde{\mathbf{s}}(1)) \in \mathcal{N}_2(0.25)$.

Finally, we show that the duality gap increases only slightly. In the exact case, the duality gap does not increase at all when $\sigma = 1$, i.e. $\mu(1) = \mu$, as can be seen in eqn. (12). Therefore, by looking at the difference between the exact and inexact duality gaps, i.e. eqn. (26), we get

$$|\tilde{\mu}(1) - \mu(1)| \leq \frac{\|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2}{n} \Rightarrow |\tilde{\mu}(1) - \mu| \leq \frac{\|(\mathbf{AD})^\dagger \mathbf{f}\|_2^2}{n}.$$

$\square$

Finally, we combine the results of the previous lemmas with a standard convergence argument to show the overall correctness and convergence rate of Algorithm 3, namely the *inexact* Predictor-Corrector IPM, thus proving Theorem 1.

Proof. (of Theorem 1) We introduce the following notation to more easily discuss the steps of Algorithm 3. Let $(\Delta\tilde{\mathbf{x}}_p, \Delta\tilde{\mathbf{y}}_p, \Delta\tilde{\mathbf{s}}_p)$ denote the predictor step computed by steps (a-b) and $(\Delta\tilde{\mathbf{x}}_c, \Delta\tilde{\mathbf{y}}_c, \Delta\tilde{\mathbf{s}}_c)$ denote the corrector step computed by steps (e-f).

Algorithm 3 first computes the predictor step from $\Delta\tilde{\mathbf{y}}_p = \texttt{Solve}(\mathbf{AD}^2\mathbf{A}^T, \mathbf{Ax}, \delta_{\epsilon,n})$ and then computes the corrector step from $\Delta\tilde{\mathbf{y}}_c = \texttt{Solve}(\mathbf{AD}^2\mathbf{A}^T, -\mu\mathbf{AS}^{-1}\mathbf{1}_n + \mathbf{Ax}, \delta_{\epsilon,n})$. Let $\mathbf{f}_p$ and $\mathbf{f}_c$ denote the error vectors incurred when solving for the predictor and corrector steps, respectively. Guarantee (i) of $\texttt{Solve}$ (see eqn. (5)) allows us to bound the term $\|(\mathbf{AD})^\dagger \mathbf{f}\|_2$ as follows:

$$\|\Delta\tilde{\mathbf{y}} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}(-\sigma\mu\mathbf{AS}^{-1}\mathbf{1}_n + \mathbf{Ax})\|_{\mathbf{AD}^2\mathbf{A}^T} \leq \delta_{\epsilon,n}$$

$$\Rightarrow \|\Delta\tilde{\mathbf{y}} - \Delta\mathbf{y}\|_{\mathbf{AD}^2\mathbf{A}^T} \leq \delta_{\epsilon,n}$$

$$\Rightarrow \|\Delta\tilde{\mathbf{y}} - \Delta\mathbf{y}\|_{\mathbf{AD}^2\mathbf{A}^T} \leq \delta_{\epsilon,n}$$

$$\Rightarrow \|(\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{f}\|_{\mathbf{AD}^2\mathbf{A}^T} \leq \delta_{\epsilon,n}$$

$$\Rightarrow \|(\mathbf{AD})^\dagger \mathbf{f}\|_2 \leq \delta_{\epsilon,n}. \tag{29}$$

Thus, we have the following bounds, using the value of $\delta_{\epsilon,n}$ in Theorem 1:

$$\|(\mathbf{AD})^\dagger \mathbf{f}_p\|_2, \ \|(\mathbf{AD})^\dagger \mathbf{f}_c\|_2 \leq \sqrt{\epsilon}/2^6.$$

Algorithm 3 sets the step size α for the predictor step to the value given by Lemma B.3. If $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0) \in \mathcal{N}_2(0.25)$ and $\mu_0 \geq 2\epsilon$, then Lemma B.4 guarantees that the predictor step will reduce the duality gap by a multiplicative factor of $1 - C_0/\sqrt{n}$, while keeping the iterate in the neighborhood $\mathcal{N}_2(0.5)$. Lemma B.5 then guarantees that the ensuing corrector step will return the iterate to the neighborhood $\mathcal{N}_2(0.25)$, while increasing the duality measure by at most $\delta_{\epsilon,n}/n$. Therefore, a single iteration of Algorithm 3, starting from a point $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0) \in \mathcal{N}_2(0.25)$ such that $\mu_0 \geq 2\epsilon$ guarantees:

$$\mu_1 \leq \left(1 - \frac{C_0}{\sqrt{n}}\right)\mu_0 + \frac{\epsilon C_0}{2\sqrt{n}\log\mu_0/\epsilon}.$$

This fulfills the conditions of Lemma A.1 with $C_0 \in (0,1)$ and $C_1 \leq \frac{C_0}{2\sqrt{n}\log\mu_0/\epsilon} \leq C_0/\sqrt{n}$. Therefore, we conclude that Algorithm 3 converges after $\frac{\sqrt{n}}{C_0}\log\frac{\mu_0}{\epsilon}$ outer iterations iterations.

We now prove that the final iterate of Algorithm 3, which we denote by $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$, is ϵ-feasible. In a single iteration, the primal variable changes by $\Delta\tilde{\mathbf{x}} = \Delta\tilde{\mathbf{x}}_p + \Delta\tilde{\mathbf{x}}_c$. Let $\mathbf{f} = \mathbf{f}_p + \mathbf{f}_c$ and recall that by the second guarantee of $\texttt{Solve}$, $\|\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2 \leq \delta_{\epsilon,n}$. This is equivalent to $\|\mathbf{f}_p\|_2, \|\mathbf{f}_c\|_2 \leq \delta_{\epsilon,n}$ for all iterations of Algorithm 3. We proceed by showing that the change in the primal residual at the k-th iteration is exactly $\mathbf{f}_k$:

$$\|\mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1}\|_2 = \|\mathbf{A}(\mathbf{x}_{k-1} + \Delta\tilde{\mathbf{x}}_{k-1}) - \mathbf{A}\mathbf{x}_{k-1}\|_2 = \|\mathbf{A}\Delta\tilde{\mathbf{x}}_{k-1}\|_2$$
$$= \|\mathbf{A}(\Delta\mathbf{x}_{k-1} - \Delta\tilde{\mathbf{x}}_{k-1})\|_2 = \|\mathbf{A}(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{f}_{k-1})\|_2 = \|\mathbf{f}_{k-1}\|_2.$$

We can use the bound $\|\mathbf{f}_i\|_2 \leq 2\delta_{\epsilon,n}$ to bound the primal residual at the k-th iteration.

$$\|\mathbf{b} - \mathbf{A}\mathbf{x}_k\|_2 = \|\mathbf{b} - \mathbf{A}\mathbf{x}_1 + (\mathbf{A}\mathbf{x}_2 - \mathbf{A}\mathbf{x}_1) + ... + (\mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1})\|_2$$
$$= \|(\mathbf{A}\mathbf{x}_0 - \mathbf{A}\mathbf{x}_1) + (\mathbf{A}\mathbf{x}_2 - \mathbf{A}\mathbf{x}_1) + ... + (\mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1})\|_2$$
$$\leq \|\mathbf{A}\mathbf{x}_0 - \mathbf{A}\mathbf{x}_1\|_2 + \|\mathbf{A}\mathbf{x}_2 - \mathbf{A}\mathbf{x}_1\| + ... + \|\mathbf{A}\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1}\|_2$$
$$\leq \|\mathbf{f}_0\|_2 + \|\mathbf{f}_1\|_2 + ... + \|\mathbf{f}_{k-1}\|_2 \leq 2k\delta_{\epsilon,n}.$$

We previously concluded in this proof by Lemma A.1 that the Algorithm 3 will converge after $k = \frac{\sqrt{n}}{C_0}\log\frac{\mu_0}{\epsilon}$ iterations. By the conditions of Theorem 1, $\delta_{\epsilon,n} < \frac{\epsilon C_0}{2\sqrt{n}\log\mu_0/\epsilon}$. Therefore, we conclude that $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 \leq \epsilon$, i.e. the solution is ϵ-primal feasible.

$\square$

C. Error-Adjusted Predictor-Corrector IPM

Lemma C.1. *Let $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$ and let $(\Delta\tilde{\mathbf{x}}, \Delta\tilde{\mathbf{y}}, \Delta\tilde{\mathbf{s}})$ be the step calculated from the inexact normal equations with error-adjustment (see eqn. (15)). Then,*

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2 + 2\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2^2.$$

Proof. This proof has a similar structure to Lemma B.1. However, there are additional steps needed due to the correction vector and we will use several of our previous bounds throughout the proof. First, we start by expressing $\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}$ as a function of the exact step and then we substitute the difference between the exact and error-adjusted steps (eqns. (17) and (18)):

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 = \|[\Delta\mathbf{x} - (\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}})] \circ [\Delta\mathbf{s} - (\Delta\mathbf{s} - \Delta\tilde{\mathbf{s}})]\|_2$$
$$\leq \|\Delta\mathbf{x} \circ \Delta\mathbf{s}\|_2 + \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v})$$
$$- \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} - \mathbf{S}^{-1}\mathbf{v})\|_2$$
$$+ \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} - \mathbf{S}^{-1}\mathbf{v}) \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v})\|_2.$$

We will bound each of these three terms in the last inequality separately. Let $\mathcal{B}_1 = \|\Delta\mathbf{x} \circ \Delta\mathbf{s}\|_2$; $\mathcal{B}_2 = \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}) - \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} - \mathbf{S}^{-1}\mathbf{v})\|_2$; and
$\mathcal{B}_3 = \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} - \mathbf{S}^{-1}\mathbf{v}) \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v})\|_2$. Using eqn. (13), we get a bound on $\mathcal{B}_1$:

$$\mathcal{B}_1 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu.$$

Next, we bound $\mathcal{B}_2$ by splitting it into two parts:

$$\begin{aligned}
\mathcal{B}_2 &= \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}) - \Delta\mathbf{s} \circ (\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} - \mathbf{S}^{-1}\mathbf{v})\|_2 \\
&= \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}) - \Delta\mathbf{s} \circ \mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \Delta\mathbf{s} \circ \mathbf{S}^{-1}\mathbf{v}\|_2 \\
&\leq \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}) - \Delta\mathbf{s} \circ \mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 + \|\Delta\mathbf{s} \circ \mathbf{S}^{-1}\mathbf{v}\|_2.
\end{aligned}$$

Let $\mathcal{B}_2^{(a)} = \|\Delta\mathbf{x} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}) - \Delta\mathbf{s} \circ \mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2$ and $\mathcal{B}_2^{(b)} = \|\Delta\mathbf{s} \circ \mathbf{S}^{-1}\mathbf{v}\|_2$. Notice that $\mathcal{B}_2^{(a)}$ can be bounded in the same way as $\mathcal{B}_2$ was bounded in eqn. (24) in the predictor-corrector proof without error-adjustment of Section 5 by setting $\mathbf{f} = \mathbf{A}\mathbf{S}^{-1}\mathbf{v}$:

$$\mathcal{B}_2^{(a)} \leq \|\mathbf{I} - 2(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}\|_2\|(\mathbf{X}\mathbf{S})^{-1/2}(-\mathbf{X}\mathbf{S}e + \sigma\mu\mathbf{1}_n)\|_2\|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2. \tag{30}$$

Furthermore, the first two terms of the above inequality were already bounded in the predictor-corrector proof without error-adjustment:

$$\|\mathbf{I} - 2(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}\|_2 \leq 2 \quad \text{and} \quad \|(\mathbf{X}\mathbf{S})^{-1/2}(-\mathbf{X}\mathbf{S}e + \sigma\mu\mathbf{1}_n)\|_2 \leq \sqrt{\frac{(\theta^2 + (1-\sigma)^2n)\mu}{(1-\theta)}}.$$

Next, we note that

$$\|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 = \|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2 \leq \|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}\|_2\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2 \leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2.$$

Substituting the above three inequalities into eqn. (30) we get:

$$\mathcal{B}_2^{(a)} \leq 2\sqrt{\frac{(\theta^2 + (1-\sigma)^2n)\mu}{(1-\theta)}}\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2.$$

We now bound $\mathcal{B}_2^{(b)}$. Using the definition of $\Delta\tilde{\mathbf{s}}$ (eqn. (17)), as well as $\|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}\|_2 \leq 1$ and the bound on $\|(\mathbf{X}\mathbf{S})^{-1/2}(-\mathbf{X}\mathbf{S}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)\|_2$, we get. Again, we apply Lemma A.2 to go from the fourth to the fifth line.

$$\begin{aligned}
\mathcal{B}_2^{(b)} &= \|\mathbf{S}^{-1}\mathbf{v} \circ \Delta\mathbf{s}\|_2 \\
&= \|\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}(\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{A}\mathbf{x})\|_2 \\
&= \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v} \circ \mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}(\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{A}\mathbf{x})\|_2 \\
&= \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v} \circ (\mathbf{A}\mathbf{D})^\dagger(\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{A}\mathbf{x})\|_2 \\
&\leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2\|(\mathbf{A}\mathbf{D})^\dagger(\sigma\mu\mathbf{A}\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{A}\mathbf{x})\|_2 \\
&\leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2\|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}(\sigma\mu(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{1}_n - (\mathbf{X}\mathbf{S})^{1/2}\mathbf{1}_n)\|_2 \\
&\leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2\|(\mathbf{A}\mathbf{D})^\dagger\mathbf{A}\mathbf{D}\|_2\|(\mathbf{X}\mathbf{S})^{-1/2}(-\mathbf{X}\mathbf{S}\mathbf{1}_n + \sigma\mu\mathbf{1}_n)\|_2 \\
&\leq \sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2.
\end{aligned}$$

We can now use the bounds on $\mathcal{B}_2^{(a)}$ and $\mathcal{B}_2^{(b)}$ to obtain the desired bound on $\mathcal{B}_2$:

$$\mathcal{B}_2 \leq \mathcal{B}_2^{(a)} + \mathcal{B}_2^{(b)} \leq 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2.$$

Finally, we bound $\mathcal{B}_3$. We distribute the terms in $\mathcal{B}_3$ and split the norm into two components using the triangle inequality:

$$
\begin{aligned}
\mathcal{B}_3 &= \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \mathbf{S}^{-1}\mathbf{v}) \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v})\|_2 \\
&= \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \mathbf{S}^{-1}\mathbf{v} \circ (\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 \\
&\leq \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 + \|\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2.
\end{aligned}
$$

Let $\mathcal{B}_3^{(a)} = \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2$ and $\mathcal{B}_3^{(b)} = \|\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2$. We first bound $\mathcal{B}_3^{(a)}$ following similar ideas to the derivation of the bound of $\mathcal{B}_3$ in the predictor-corrector proof without error-adjustment:

$$
\begin{aligned}
\mathcal{B}_3^{(a)} &= \|(\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 \\
&= \|(\mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 \\
&= \|(\mathbf{A}\mathbf{D})^\dagger \mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2^2 \leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2^2.
\end{aligned}
$$

We also bound $\mathcal{B}_3^{(b)}$:

$$
\begin{aligned}
\mathcal{B}_3^{(b)} &= \|\mathbf{S}^{-1}\mathbf{v} \circ \mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 = \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v} \circ \mathbf{D}\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v}\|_2 \\
&\leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_\infty \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2 \leq \|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2^2.
\end{aligned}
$$

Combining the above two inequalities, we get the overall bound for $\mathcal{B}_3$:

$$
\|\mathcal{B}_3\|_2 \leq \mathcal{B}_3^{(a)} + \mathcal{B}_3^{(b)} \leq 2\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2^2.
$$

Finally, summing up all bounds for $\mathcal{B}_1$, $\mathcal{B}_2$, and $\mathcal{B}_3$ gives our final inequality:

$$
\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2 + 2\|(\mathbf{X}\mathbf{S})^{-1/2}\mathbf{v}\|_2^2.
$$

$\square$

Recall that a point $(\mathbf{x}, \mathbf{y}, \mathbf{s})$ is in the neighborhood $\mathcal{N}_2(\theta)$ if $\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2 \leq \mu\theta$. We bound the left hand side of this condition after a step of size α is taken.

Lemma C.2. *If $\alpha \in [0,1]$, then*

$$
\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2 + \alpha^2\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + 2\alpha\|\mathbf{v}\|_2.
$$

Proof. We start by expanding the expression for $\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha)$:

$$
\begin{aligned}
\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) &= (\mathbf{x} + \alpha\Delta\tilde{\mathbf{x}}) \circ (\mathbf{s} + \alpha\Delta\tilde{\mathbf{s}}) \\
&= \mathbf{x} \circ \mathbf{s} + \alpha(\mathbf{x} \circ \Delta\tilde{\mathbf{s}} + \mathbf{s} \circ \Delta\tilde{\mathbf{x}}) + \alpha^2\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}} \\
&= \mathbf{x} \circ \mathbf{s} + \alpha(\mathbf{x} \circ \Delta\tilde{\mathbf{s}} + \mathbf{s} \circ (-\mathbf{x} + \sigma\mu\mathbf{S}^{-1}\mathbf{1}_n - \mathbf{D}^2\Delta\tilde{\mathbf{s}} - \mathbf{S}^{-1}\mathbf{v})) + \alpha^2\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}} \\
&= \mathbf{x} \circ \mathbf{s} + \alpha(-\mathbf{x} \circ \mathbf{s} + \sigma\mu\mathbf{1}_n - \mathbf{v}) + \alpha^2\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}} \\
&= (1-\alpha)\mathbf{x} \circ \mathbf{s} + \alpha\sigma\mu\mathbf{1}_n - \alpha\mathbf{v} + \alpha^2\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}.
\end{aligned}
$$

Left-multiplying the final expression by the vector $\mathbf{1}_n^T$ and dividing by n, gives an expression for $\tilde{\mu}(\alpha)$. Notice that $\Delta\tilde{\mathbf{x}}^T\Delta\tilde{\mathbf{s}} = -\Delta\tilde{\mathbf{x}}^T\mathbf{A}^T\Delta\tilde{\mathbf{y}} = 0$ by substituting the definition of the $\Delta\tilde{\mathbf{s}}$ without error-adjustment and using the fact that $\mathbf{A}\mathbf{x} = \mathbf{b}$ at each step:

$$
\begin{aligned}
\tilde{\mu}(\alpha) &= \frac{1}{n}\mathbf{1}_n^T[(1-\alpha)\mathbf{x} \circ \mathbf{s} + \alpha\sigma\mu\mathbf{1}_n - \alpha\mathbf{v} + \alpha^2\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}] \\
&= [1 - \alpha(1-\sigma)]\mu - \alpha/n\,\mathbf{v}^T\mathbf{1}_n.
\end{aligned} \tag{31}
$$

For simplicity of exposition, we first look at individual elements of the vector $\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)$:

$$\tilde{\mathbf{x}}_i(\alpha)\tilde{\mathbf{s}}_i(\alpha) - \tilde{\mu}(\alpha) = (1-\alpha)\mathbf{x}_i\mathbf{s}_i + \alpha\sigma\mu - \alpha\mathbf{v}_i + \alpha^2\Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i - \mu + \alpha\mu - \alpha\sigma\mu + \frac{\alpha\mathbf{v}^T\mathbf{1}_n}{n}$$

$$= (1-\alpha)(\mathbf{x}_i\mathbf{s}_i - \mu) + \alpha^2\Delta\tilde{\mathbf{x}}_i\Delta\tilde{\mathbf{s}}_i - \alpha\left(\mathbf{v}_i - \frac{\mathbf{v}^T\mathbf{1}_n}{n}\right).$$

We bound the norm of the last summand as follows:

$$\frac{1}{n}|\mathbf{v}^T\mathbf{1}_n| \leq \frac{1}{n}\|\mathbf{v}\|_2\|\mathbf{1}_n\|_2 \leq \frac{1}{\sqrt{n}}\|\mathbf{v}\|_2 \Rightarrow \left\|\mathbf{v} - \frac{(\mathbf{v}^T\mathbf{1}_n)}{n}\mathbf{1}_n\right\|_2 \leq \|\mathbf{v}\|_2 + \frac{1}{\sqrt{n}}\|\mathbf{v}\|_2 \leq 2\|\mathbf{v}\|_2.$$

Therefore, we can conclude that,

$$\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq (1-\alpha)\|\mathbf{x}\circ\mathbf{s} - \mu\mathbf{1}_n\|_2 + \alpha^2\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2 + 2\alpha\|\mathbf{v}\|_2.$$

$\square$

Given the previous bound, we can now derive a step size α which guarantees that the iterate remains in $\mathcal{N}_2(0.5)$ after the predictor step.

Lemma C.3. *If* $(\mathbf{x},\mathbf{y},\mathbf{s}) \in \mathcal{N}_2(0.25)$, $\alpha = \min\left\{1/2, (\mu/16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2)^{1/2}\right\}$, *and* $\|\mathbf{v}\|_2 \leq \mu/32$, *then the predictor step* $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$.

Proof. Our starting point is the bound of Lemma C.2. By definition, α is upper-bounded by both $1/2$ and the term depending on μ:

$$\|\tilde{\mathbf{x}}(\alpha) \circ \tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq (1-\alpha)\|\mathbf{x}\circ\mathbf{s} - \mu\mathbf{1}_n\|_2 + \alpha^2\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2 + 2\alpha\|\mathbf{v}\|_2$$

$$\leq \frac{(1-\alpha)\mu}{4} + \frac{\mu}{16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2}\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2 + \frac{2}{2}\|\mathbf{v}\|_2$$

$$\leq \frac{(1-\alpha)\mu}{4} + \frac{\mu}{16} + \frac{\mu}{32}$$

$$\leq \frac{(1-\alpha)\mu}{4} + \frac{\mu}{8} - \frac{\mu}{32}$$

$$\leq \frac{(1-\alpha)\mu}{4} + \frac{(1-\alpha)\mu}{8(1-\alpha)} - \frac{\mu}{32}$$

$$\leq \frac{1}{2}(1-\alpha)\mu - \frac{\mu}{32}$$

$$\leq \frac{1}{2}\tilde{\mu}(\alpha).$$

The last step follows from eqn. (31), which states $\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu - \alpha/n\,\mathbf{v}^T\mathbf{1}_n$. By applying the Cauchy-Schwarz inequality to $\mathbf{v}^T\mathbf{1}_n$ as done previously and $\sigma = 0$, we obtain $\tilde{\mu}(\alpha) \geq (1-\alpha)\mu - \frac{\alpha}{\sqrt{n}}\|\mathbf{v}\|_2$, which allows us to conclude that $\|\tilde{\mathbf{x}}(\alpha)\circ\tilde{\mathbf{s}}(\alpha) - \tilde{\mu}(\alpha)\mathbf{1}_n\|_2 \leq \frac{1}{2}\tilde{\mu}(\alpha)$. Now, we must show that the condition $(\tilde{\mathbf{x}}(\alpha),\tilde{\mathbf{s}}(\alpha)) > 0$ is fulfilled. First, by eqns. (12, 31), we have $\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu - \alpha/n\,\mathbf{v}^T\mathbf{1}_n$, which shows $\tilde{\mu}(\alpha') > 0$ for all positive step sizes $\alpha' \leq \alpha$. From the first part of this proof, we have that $\tilde{\mathbf{x}}_i(\alpha)\tilde{\mathbf{s}}_i(\alpha) \geq 1/2\tilde{\mu}(\alpha)$. We conclude that $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha)) \in \mathcal{N}_2(0.5)$. $\square$

We now show that the predictor step with step size α as given in the above lemma guarantees a multiplicative decrease in the duality gap. Recall that $\sigma = 0$ in the predictor step when solving the normal equations.

Lemma C.4. *If* $(\mathbf{x},\mathbf{y},\mathbf{s}) \in \mathcal{N}_2(0.25)$, $\alpha = \min\left\{1/2, (\mu/16\|\Delta\tilde{\mathbf{x}}\circ\Delta\tilde{\mathbf{s}}\|_2)^{1/2}\right\}$, *and* $\|\mathbf{v}\|_2 \leq \mu/32$, *then the predictor step* $(\tilde{\mathbf{x}}(\alpha), \tilde{\mathbf{y}}(\alpha), \tilde{\mathbf{s}}(\alpha))$ *remains in* $\mathcal{N}_2(0.5)$ *and there exists a constant* $C_0 \in (0,1)$ *such that,*

$$\frac{\tilde{\mu}(\alpha)}{\mu} \leq 1 - \frac{C_0}{\sqrt{n}}.$$

Proof. Lemma C.3 already shows that this value of α ensures that the next iterate remains in $\mathcal{N}_2(0.5)$. Therefore, we just need to prove the multiplicative decrease in the duality measure. Towards that end, we will again need to lower bound the step size α, starting from the upper bound for $\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2$. By Lemma C.1, we get the following inequality:

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 + 2\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2^2.$$

We now derive a bound on $\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2$ using the bound on $\|\mathbf{v}\|_2$. We use the fact that for $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$, $\mathbf{x}_i\mathbf{s}_i \geq (1-\theta)\mu$ to get:[4]

$$\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 \leq \|(\mathbf{XS})^{-1/2}\|_2\|\mathbf{v}\|_2 \leq \frac{1}{\sqrt{\min \mathbf{x}_i\mathbf{s}_i}}\|\mathbf{v}\|_2 \leq \frac{1}{\sqrt{(1/4)\mu}}\frac{\mu}{9} \leq \frac{2\sqrt{\mu}}{9}.$$

Next, we simplify the inequality from Lemma C.1 by substituting $\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 \leq \frac{2\sqrt{\mu}}{9}$, $\theta = 0.25$, and $\sigma = 0$ to get

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{(0.25)^2 + n(1-0)^2}{2^{3/2}(1-0.25)}\mu + 3\sqrt{\frac{((0.25)^2 + n(1-0)^2)\mu}{(1-0.25)}}\frac{2\sqrt{\mu}}{9} + \frac{2 \cdot 2^2\mu}{9^2}$$

$$\leq \frac{(1/16) + n}{2^{3/2}(3/4)}\mu + 3\sqrt{\frac{(1/16) + n}{(3/4)}}\frac{2\mu}{9} + \frac{2^3\mu}{9^2}$$

$$\leq n\mu\left(\frac{(1/16) + 1}{2^{3/2}(3/4)} + 3\sqrt{\frac{(1/16) + 1}{(3/4)}}\frac{2}{9} + \frac{2^3}{9^2}\right)$$

$$\leq 1.4 \cdot n\mu.$$

The above upper bound can now be used to lower-bound α:

$$\alpha = \min\left\{\frac{1}{2}, \left(\frac{\mu}{16\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2}\right)^{1/2}\right\}$$

$$\geq \min\left\{\frac{1}{2}, \left(\frac{\mu}{16n\mu(1.4)}\right)^{1/2}\right\}$$

$$\geq \frac{0.2}{\sqrt{n}}.$$

Eqn. (31) states that $\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu - \alpha/n\,\mathbf{v}^T\mathbf{1}_n$. Combining it with our upper and lower bounds for α we can bound the decrease in the duality measure $\tilde{\mu}$ as follows:

$$\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu - \frac{\alpha}{n}\mathbf{v}^T\mathbf{1}_n$$

$$\leq [1 - \alpha(1-\sigma)]\mu + \frac{\alpha}{\sqrt{n}}\|\mathbf{v}\|_2 \quad \text{(By Cauchy-Schwarz)}$$

$$\leq \left(1 - \frac{0.2}{\sqrt{n}}\right)\mu + \frac{\mu}{2\sqrt{n} \cdot 9}$$

$$\leq \left(1 - \frac{0.2 - 1/18}{\sqrt{n}}\right)\mu$$

$$\leq \left(1 - \frac{0.14}{\sqrt{n}}\right)\mu. \tag{32}$$

$\square$

---

[4]The inequality $\mathbf{x}_i\mathbf{s}_i \geq (1-\theta)\mu$ follows from the definition of $\mathcal{N}_2(\theta)$.

We have shown that the predictor step results in a multiplicative decrease in the duality gap, while keeping the next iterate in the neighborhood $\mathcal{N}_2(0.5)$. We now show that the corrector step returns the iterate to the $\mathcal{N}_2(0.25)$ neighborhood, while increasing the duality gap by a small additive amount.

Lemma C.5. *Let* $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(0.5)$ *and* $\|\mathbf{v}\|_2 \leq {}^\mu/2^7$. *Then, the corrector step* $(\tilde{\mathbf{x}}(1), \tilde{\mathbf{y}}(1), \tilde{\mathbf{s}}(1)) \in \mathcal{N}_2(0.25)$ *and* $|\tilde{\mu}(1) - \mu| \leq \frac{1}{\sqrt{n}}\|\mathbf{v}\|_2$.

Proof. We start by simplifying the inequality of Lemma C.1 for the corrector step. Recall:

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{\theta^2 + n(1-\sigma)^2}{2^{3/2}(1-\theta)}\mu + 3\sqrt{\frac{(\theta^2 + n(1-\sigma)^2)\mu}{(1-\theta)}}\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 + 2\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2^2.$$

We bound $\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2$ using the bound on $\|\mathbf{v}\|_2$ from the condition of the lemma:

$$\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 \leq \|(\mathbf{XS})^{-1/2}\|_2\|\mathbf{v}\|_2 \leq \frac{1}{\sqrt{\min \mathbf{x}_i\mathbf{s}_i}}\|\mathbf{v}\|_2 \leq \frac{1}{\sqrt{(1/2)\mu}}\frac{\mu}{2^7} \leq \frac{\sqrt{\mu}}{2^6}.$$

We then simplify the inequality from Lemma C.1 by substituting $\|(\mathbf{XS})^{-1/2}\mathbf{v}\|_2 \leq \frac{\sqrt{\mu}}{2^6}$, $\theta = 0.5$, and $\sigma = 1$:

$$\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 \leq \frac{(0.5)^2 + n(1-1)^2}{2^{3/2}(1-0.5)}\mu + 3\sqrt{\frac{((0.5)^2 + n(1-1)^2)\mu}{(1-0.5)}}\frac{\sqrt{\mu}}{2^6} + 2\frac{\mu}{2^{14}} \leq \frac{\mu}{2^{5/2}} + \frac{3\mu}{2^{13/2}} + \frac{\mu}{2^{13}}.$$

Next, we show that taking a step with step size $\alpha = 1$ and centering parameter $\sigma = 1$ from a point in the "larger" neighborhood $\mathcal{N}_2(0.5)$ returns the iterate to the "smaller" neighborhood $\mathcal{N}_2(0.25)$. We start from the result of Lemma C.2 with $\alpha = 1$:

$$\begin{aligned}
\|\tilde{\mathbf{x}}(1) \circ \tilde{\mathbf{s}}(1) - \tilde{\mu}(1)\|_2 &\leq (1-\alpha)\|\mathbf{x} \circ \mathbf{s} - \mu\mathbf{1}_n\|_2 + \alpha^2\|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + 2\alpha\|\mathbf{v}\|_2 \\
&\leq \|\Delta\tilde{\mathbf{x}} \circ \Delta\tilde{\mathbf{s}}\|_2 + 2\|\mathbf{v}\|_2 \\
&\leq \frac{\mu}{2^{5/2}} + \frac{3\mu}{2^{13/2}} + \frac{\mu}{2^{13}} + \frac{2\mu}{2^7} \\
&\leq \frac{\mu}{2^{5/2}} + \frac{3\mu}{2^{13/2}} + \frac{\mu}{2^{13}} + \frac{2\mu}{2^7} + \frac{\mu}{2^7} - \frac{\mu}{2^7} \\
&\leq \frac{\mu}{4} - \frac{\mu}{2^7} \leq \frac{\tilde{\mu}}{4}.
\end{aligned}$$

The last step follows from $\tilde{\mu}(\alpha) \geq \mu - \frac{\alpha}{\sqrt{n}}\|\mathbf{v}\|_2$, which can be derived from eqn. (31).

This implies that the corrector step will return the iterate to the neighborhood $\mathcal{N}_2(0.25)$.

Finally, by eqn. (31), we know that $\tilde{\mu}(\alpha) = [1 - \alpha(1-\sigma)]\mu - {}^\alpha/n\mathbf{v}^T\mathbf{1}_n$. Substituting $\alpha = 1$ and $\sigma = 1$ allows us to bound $\tilde{\mu}(1) - \mu$:

$$\tilde{\mu}(1) = \mu - \frac{1}{n}\mathbf{v}^T\mathbf{1}_n \Rightarrow \tilde{\mu}(1) - \mu = \frac{-1}{n}\mathbf{v}^T\mathbf{1}_n \Rightarrow \tilde{\mu}(1) - \mu \leq \frac{1}{n}\sqrt{n}\|\mathbf{v}\|_2 \Rightarrow \tilde{\mu}(1) - \mu \leq \frac{1}{\sqrt{n}}\|\mathbf{v}\|_2.$$

$\square$

We are now ready to combine the results of the previous lemmas to show the overall correctness and convergence rate of Algorithm 1, the error-adjusted inexact predictor-corrector IPM.

Proof. (of Theorem 2) By the guarantees of $\mathtt{Solve}^\mathbf{v}$, we know that the error-adjusted normal equations are solved for a given σ and $\|\mathbf{v}\|_2 < {}^\epsilon/2^7 < {}^\mu/2^7$ at each iteration. First, Lemma C.3 guarantees that the intermediate point computed at step (d) of Algorithm 1 remains in the neighborhood $\mathcal{N}_2(0.5)$. Lemma C.4 guarantees that the predictor step decreases the duality measure of the iterate by at least a multiplicative factor of the form $(1 - {}^{C_0}/\sqrt{n})$ for some constant C_0.

Next, Lemma C.5 ensures that the corrector step of Algorithm 1 returns the iterate to the neighborhood $\mathcal{N}_2(0.25)$, while increasing the duality measure by at most $1/\sqrt{n}\|\mathbf{v}\|_2 \leq \delta_{\epsilon,n}/\sqrt{n}$. Therefore, a single iteration of Algorithm 1 starting from a point $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0) \in \mathcal{N}_2(0.25)$ such that $\mu_0 \geq 2\epsilon$ guarantees the following inequality:

$$\mu_1 \leq \left(1 - \frac{C_0}{\sqrt{n}}\right)\mu_0 + \frac{\epsilon}{2^7\sqrt{n}}.$$

This fulfills the conditions of Lemma A.1 with $C_0 \in (0,1)$ and $C_1 \leq \frac{1}{2^7\sqrt{n}} \leq C_0/\sqrt{n}$ (see eqn. (32)). Therefore, we conclude that Algorithm 1 converges in $\mathcal{O}(\sqrt{n}\log{\mu_0}/\epsilon)$ iterations.

Finally, we prove that the final iterate is primal-feasible, i.e. $\|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|_2 = 0$. By eqn. (18), at each step of Algorithm 1, $\Delta\mathbf{x} - \Delta\tilde{\mathbf{x}} = -\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \mathbf{S}^{-1}\mathbf{v}$. We left multiply this expression by $\mathbf{A}$ to get

$$\mathbf{A}\Delta\mathbf{x} - \mathbf{A}\Delta\tilde{\mathbf{x}} = -\mathbf{A}\mathbf{D}^2\mathbf{A}^T(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{A}\mathbf{S}^{-1}\mathbf{v} + \mathbf{A}\mathbf{S}^{-1}\mathbf{v} = \mathbf{0}_m.$$

This implies that the change in the primal-residual of the error-adjusted algorithm is the same as the change in the primal-residual of the exact algorithm at every iteration. Therefore, since the exact algorithm returns a primal-feasible solution, the error-adjusted algorithm does as well. □

D. Implementing `Solve` and `Solve`ᵛ using randomized linear algebra

We now discuss how to implement the solvers that are needed in our inexact predictor-corrector IPMs, with and without the correction vector, using standard preconditioned solvers, such as the preconditioned conjugate gradient (PCG) method. We use well-known sketching-based approaches to construct the preconditioner, leveraging results from the randomized linear algebra literature.

We first focus on (full row-rank) constraint matrices $\mathbf{A} \in \mathbb{R}^{m \times n}$ that are short-and-fat, ie., $m \ll n$. Clearly such matrices have rank $m \ll n$. In Appendix D.3 below we will discuss how to reduce general LP problems with exact low-rank constraint matrices to this setting. Moreover, in Appendix D.2, we also discuss how to handle the tall-and-thin constraint matrices. Towards that end, consider the LP of eqn. (1) with an input matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $m \ll n$. First, we prove that the preconditioned conjugate gradient (PCG) method of Algorithm 2 (see also (Chowdhury et al., 2020)) can fulfill the requirements of `Solve` in $\mathcal{O}\left(\log\frac{\sigma_{\max}(\mathbf{A}\mathbf{D})n\mu}{\delta}\right)$ iterations and the guarantees of `Solve`ᵛ in $\mathcal{O}\left(\log\frac{n\mu}{\delta}\right)$ iterations.

Let $\mathbf{A}\mathbf{D} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}}$ be the thin SVD representation and $\mathbf{W} \in \mathbb{R}^{n \times w}$ be an oblivious sparse sketching matrix which satisfies:[5]

$$\|\mathbf{V}\mathbf{W}\mathbf{W}^T\mathbf{V}^T - \mathbf{I}_m\|_2 \leq \frac{\zeta}{2}, \tag{33}$$

with probability at least $1 - \eta$. The work of Cohen et al. (2016) shows how to construct such a matrix $\mathbf{W}$ fulfilling this guarantee with sketch size $w = \mathcal{O}(m/\zeta^2 \cdot \log m/\eta)$ and $\mathcal{O}(1/\zeta \cdot \log m/\eta)$ non-zero entries per row. One possible construction is to uniformly sample $s = \mathcal{O}(1/\zeta \cdot \log m/\eta)$ entries per row of $\mathbf{W}$ without replacement and set each of the selected entries to $\pm 1/s$ independently and uniformly randomly. Next, we use the above sketching matrix to define $\mathbf{Q} = \mathbf{A}\mathbf{D}\mathbf{W}\mathbf{W}^T\mathbf{D}\mathbf{A}^T$; we note that $\mathbf{Q}$ is not explicitly constructed in Algorithm 2. Then, with probability at least $1 - \eta$, the vector $\tilde{\mathbf{z}}^t$ computed by Algorithm 2 fulfills the following inequality (see Equation 7 in (Chowdhury et al., 2020)).

$$\|\mathbf{Q}^{-1/2}(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\tilde{\mathbf{z}}^t - \mathbf{Q}^{-1/2}\mathbf{p}\|_2 \leq \zeta^t\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2, \quad \zeta \in (0,1). \tag{34}$$

Recall that the function `Solve` is defined to have the following guarantees:

$$\Delta\tilde{\mathbf{y}} = \texttt{Solve}(\mathbf{A}\mathbf{D}^2\mathbf{A}^T, \mathbf{p}, \delta) \Rightarrow \|\Delta\tilde{\mathbf{y}} - (\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{A}\mathbf{D}^2\mathbf{A}^T} \leq \delta \text{ and } \|\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2 \leq \delta.$$

The next lemma shows that Algorithm 2 fulfills the conditions of `Solve`.

Lemma D.1. *If Algorithm 2 is used to compute* $\Delta\tilde{\mathbf{y}} = \texttt{Solve}(\mathbf{A}\mathbf{D}^2\mathbf{A}^T, \mathbf{p}, \delta)$, $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{N}_2(\theta)$, *and* $t = \mathcal{O}\left(\log\frac{\sigma_{\max}(\mathbf{A}\mathbf{D})n\mu}{\delta}\right)$, *then, with probability at least* $1 - \eta$, $\Delta\tilde{\mathbf{y}}$ *satisfies*

$$\|\Delta\tilde{\mathbf{y}} - (\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{A}\mathbf{D}^2\mathbf{A}^T} \leq \delta \text{ and } \|\mathbf{A}\mathbf{D}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2 \leq \delta.$$

---

[5]Let $\|\mathbf{A}\|_F^2 = \sum_{i,j} A_{ij}^2 = \text{tr}(\mathbf{A}^T\mathbf{A})$ denote the (square of the) Frobenius norm of matrix $\mathbf{A}$.

Proof. We start by bounding $\|\Delta\tilde{\mathbf{y}} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{AD}^2\mathbf{A}^T}$, where $\Delta\tilde{\mathbf{y}} = \mathbf{Q}^{-1/2}\tilde{\mathbf{z}}^t$, using the guarantee given by eqn. (34) and Lemma 2 of (Chowdhury et al., 2020), which guarantees $(1 + \zeta/2)^{-1} \leq \sigma_i^2(\mathbf{Q}^{-1/2}\mathbf{AD}) \leq (1 - \zeta/2)^{-1}$ for all $i = 1\ldots m$, when $\mathbf{W}$ fulfills eqn. (33):

$$\|\mathbf{Q}^{-1/2}(\mathbf{AD}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{Q}^{-1/2}\mathbf{p}\|_2 = \|\mathbf{Q}^{-1/2}(\mathbf{AD}^2\mathbf{A}^T)(\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p})\|_2$$

$$\geq \frac{1}{\sqrt{1 + \zeta/2}}\|\mathbf{DA}^T(\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p})\|_2$$

$$\geq \frac{1}{\sqrt{1 + \zeta/2}}\|(\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p})\|_{(\mathbf{AD}^2\mathbf{A}^T)}.$$

The first step is justified by Lemma A.3, since the column space of $\mathbf{DA}^T$ is the row space of $\mathbf{Q}^{-1/2}\mathbf{AD}$. We show in Lemma A.4 that $\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2 \leq C\sqrt{n\mu}$ for some constant C depending only on σ and θ. Combining this bound with eqn. (34) gives:

$$\zeta^t(1 + \zeta/2)^{1/2}C\sqrt{n\mu} \geq \|(\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p})\|_{(\mathbf{AD}^2\mathbf{A}^T)}. \tag{35}$$

This implies that $\|\Delta\tilde{\mathbf{y}} - (\mathbf{AD}^2\mathbf{A}^T)^{-1}\mathbf{p}\|_{\mathbf{AD}^2\mathbf{A}^T} \leq \delta$ after $t = \mathcal{O}\left(\log\frac{n\mu}{\delta}\right)$ iterations. Next, we bound $\|\mathbf{AD}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2$ using the guarantee of eqn. (34):

$$\|\mathbf{Q}^{-1/2}(\mathbf{AD}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{Q}^{-1/2}\mathbf{p}\|_2 \geq \sigma_{\min}(\mathbf{Q}^{-1/2})\|\mathbf{AD}^2\mathbf{A}^T\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{p}\|_2$$

$$\Rightarrow \zeta^t\sigma_{\max}(\mathbf{Q}^{1/2})\|\mathbf{Q}^{-1/2}\mathbf{p}\|_2 \geq \|\mathbf{AD}^2\mathbf{A}^T\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{p}\|_2$$

$$\Rightarrow \zeta^t\sigma_{\max}(\mathbf{Q}^{1/2})C\sqrt{n\mu} \geq \|\mathbf{AD}^2\mathbf{A}^T\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{p}\|_2.$$

Again, the first step follows from Lemma A.3. Since $\zeta \in (0,1)$, we conclude that $\|\mathbf{AD}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}\|_2 < \delta$ after $t = \mathcal{O}\left(\log\frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta}\right)$ iterations. Therefore, both guarantees of Solve can be achieved with probability at least $1 - \eta$ in $t = \mathcal{O}\left(\log\frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta}\right)$ iterations. $\square$

Satisfying eqn. (34). Exploiting the properties of the preconditioner $\mathbf{Q}^{-1/2}$, (Chowdhury et al., 2020) showed how to satisfy eqn. (34) using popular solvers beyond conjugate gradient. Such solvers include steepest descent and Richardson iteration. We could do the same in our work and prove similar results for, say, the Chebyshev iteration (Barrett et al., 1994; Gutknecht, 2008; Gutknecht & Röllin, 2002). Indeed, the preconditioner $\mathbf{Q}^{-1/2}$ can be combined with Theorem 1.6.2 of (Gutknecht, 2008) to satisfy eqn. (34). Chebyshev iteration avoids the computation of the inner products which is typically needed for CG or other inexact methods. As a result, Chebyshev iteration offers several advantages in a parallel environment as it does not need to evaluate communication-intensive inner products for computing the recurrence parameters.

D.1. Computing the error-adjustment vector **v** for Algorithm 1

In this section we discuss how to efficiently compute the correction vector $\mathbf{v}$ for our "corrected" inexact predictor-corrector IPM. Recall eqn. (15): the correction vector must satisfy $\mathbf{AS}^{-1}\mathbf{v} = (\mathbf{AD}^2\mathbf{A}^T)(\Delta\tilde{\mathbf{y}} - \Delta\mathbf{y})$. One possible construction of such a vector $\mathbf{v}$ is the following:

$$\mathbf{v} = (\mathbf{XS})^{1/2}\mathbf{W}(\mathbf{ADW})^{\dagger}(\mathbf{AD}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p}). \tag{36}$$

Notice that this vector can be constructed efficiently from quantities already computed in Algorithm 2. Left-multiplying by $\mathbf{AS}^{-1}$ immediately proves that this construction for $\mathbf{v}$ satisfies $\mathbf{AS}^{-1}\mathbf{v} = (\mathbf{AD}^2\mathbf{A}^T)(\Delta\tilde{\mathbf{y}} - \Delta\mathbf{y})$, with probability at least $1 - \eta$. We now prove the following lemma:

Lemma D.2. *Let $\Delta\tilde{\mathbf{y}}$ be computed by Algorithm 2 and let the correction vector $\mathbf{v}$ be computed by eqn. (36). Then, $\|\mathbf{v}\|_2 \leq \delta$ after $t = \mathcal{O}\left(\log\frac{n\mu}{\delta}\right)$ iterations.*

Proof. Lemma 5 from (Chowdhury et al., 2020) (using our notation) guarantees that:

$$\|\mathbf{v}\|_2 \leq \sqrt{3n\mu}\|\mathbf{Q}^{-1/2}(\mathbf{AD}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{Q}^{-1/2}\mathbf{p}\|_2.$$

Using eqn. (35),

$$\|\mathbf{v}\|_2 \leq \sqrt{3n\mu}\|\mathbf{Q}^{-1/2}(\mathbf{A}\mathbf{D}^2\mathbf{A}^T)\mathbf{Q}^{-1/2}\mathbf{z}^{(t)} - \mathbf{Q}^{-1/2}\mathbf{p}\|_2$$
$$\leq \sqrt{3n\mu}(\zeta^t(1+\zeta/2)^{1/2}Cn\sqrt{\mu}).$$

Again, since $\zeta \in (0,1)$, we can conclude that $\|\mathbf{v}\|_2 \leq \delta$ after $t = \mathcal{O}\left(\log \frac{n\mu}{\delta}\right)$ iterations of Algorithm 2 with probability at least $1 - \eta$. $\qquad\square$

D.2. Constraint matrices with $m \gg n$ and $\mathrm{rank}(\mathbf{A}) = n$

So far, we only focused on constraint matrices that have full row-rank and are wide *i.e.*, $m \ll n$. By considering the dual problem, our methods also address constraint matrices that are tall-and-thin and have full column rank *i.e.*, $m \gg n$. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be the constraint matrix with $m \gg n$ and $\mathrm{rank}(\mathbf{A}) = n$ such that the primal LP is given by

$$\min \mathbf{c}^\mathsf{T}\mathbf{x}, \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{b}, \ \mathbf{x} \geq \mathbf{0}. \tag{37}$$

The associated dual problem is,

$$\max \mathbf{b}^\mathsf{T}\mathbf{y}, \text{ subject to } \mathbf{A}^\mathsf{T}\mathbf{y} + \mathbf{s} = \mathbf{c}, \mathbf{s} \geq \mathbf{0}, \tag{38}$$

Note that the dual variable $\mathbf{y}$ is a free variable *i.e.* it can have both non-negative and non-positive entries. However, we can always rewrite $\mathbf{y}$ as the difference between two non-negative vectors. Therefore, let $\mathbf{y} = \mathbf{y}^+ - \mathbf{y}^-$, where both $\mathbf{y}^+, \mathbf{y}^- \geq \mathbf{0}$. Now, if we rewrite eqn. (38) in terms of $\mathbf{y}^+$ and $\mathbf{y}^-$ and change $\max \mathbf{b}^T\mathbf{y}$ to $\min -\mathbf{b}^T\mathbf{y}$, it becomes

$$\min -\mathbf{b}^\mathsf{T}\mathbf{y}, \text{ subject to } \mathbf{A}^\mathsf{T}\mathbf{y}^+ - \mathbf{A}^\mathsf{T}\mathbf{y}^- + \mathbf{s} = \mathbf{c}, \text{ and } \mathbf{y}^+, \mathbf{y}^-\mathbf{s} \geq \mathbf{0} \tag{39}$$

Now, we can express eqn. (39) as

$$\min \bar{\mathbf{b}}^\mathsf{T}\bar{\mathbf{y}}, \text{ subject to } \bar{\mathbf{A}}\bar{\mathbf{y}} = \mathbf{c}, \ \bar{\mathbf{y}} \geq \mathbf{0}, \tag{40}$$

where $\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A}^\mathsf{T} & -\mathbf{A}^\mathsf{T} & \mathbf{I}_n \end{bmatrix} \in \mathbb{R}^{n \times (2m+n)}$, $\bar{\mathbf{b}} = \begin{pmatrix} -\mathbf{b} \\ \mathbf{b} \\ \mathbf{0}_n \end{pmatrix} \in \mathbb{R}^{2m+n}$, and $\bar{\mathbf{y}} = \begin{pmatrix} \mathbf{y}^+ \\ \mathbf{y}^- \\ \mathbf{s} \end{pmatrix} \in \mathbb{R}^{2m+n}$.

Note that $\bar{\mathbf{A}}$ is short-and-fat as $2m + n \gg n$ and it also has full row-rank. Therefore, eqn. (40) can be solved using our framework.

D.3. A generalization to low-rank constraint matrices

We will now discuss how to apply randomized preconditioners and iterative solvers to LPs where $\mathbf{A}$ can be any $m \times n$ matrix with $\mathrm{rank}(\mathbf{A}) = k \ll \min\{m, n\}$, which we assume to be known[6]. In addition, we further emphasize that we also assume the set of primal-dual solutions of the LP is non-empty *i.e.* there exists at least one feasible point.

First, we briefly discuss the approximate SVD "proto-algorithm" of (Halko et al., 2011) that will be instrumental in translating the low-rank LP into our sketching-based framework. The single-iteration "proto-algorithm" of (Halko et al., 2011) returns a matrix $\mathbf{Z} \in \mathbb{R}^{m \times (\ell+2)}$ with $\mathbf{Z}^\mathsf{T}\mathbf{Z} = \mathbf{I}_{(\ell+2)}$ ($\ell \leq k$) such that for some constant $\varepsilon_0 \geq 0$, the following inequality holds with high probability:[7]

$$\|\mathbf{A} - \mathbf{Z}\mathbf{Z}^\mathsf{T}\mathbf{A}\|_2 \leq (1 + \epsilon_0)\|\mathbf{A} - \mathbf{A}_\ell\|_2, \tag{41}$$

where $\mathbf{A}_\ell$ is the *best* ℓ-rank approximation of $\mathbf{A}$. The computation of $\mathbf{Z}$ is dominated by the cost of multiplying $\mathbf{A}$ by a vector and thus, can be computed in $\mathcal{O}(\ell \cdot \mathrm{nnz}(\mathbf{A}))$ time. By taking $\ell = k$, we have $\mathbf{A} = \mathbf{A}_k$ which makes the right hand side of eqn. (41) equal to zero. Therefore, letting $\widetilde{\mathbf{A}} = \mathbf{Z}\mathbf{Z}^\mathsf{T}\mathbf{A}$ directly yields $\mathbf{A} = \widetilde{\mathbf{A}}$.

---

[6]When k is not known in advance, one can efficiently estimate it using trace estimation techniques (Avron & Toledo, 2011; Ubaru & Saad, 2016).

[7]Here, $\epsilon_0 = 9\sqrt{\ell + p} \cdot \sqrt{\min\{m, n\}}$. We set $p = 2$ (the minimal allowed value), which suffices for our purposes, since the matrix $\mathbf{A} = \mathbf{A}_\ell$ has exact low-rank.

Now, as we already have $\mathbf{A} = \widetilde{\mathbf{A}}$ from eqn. (41) with $\ell = k$,

$$\operatorname*{argmin}_{\mathbf{Ax}=\mathbf{b},\,\mathbf{x}\geq\mathbf{0}} \mathbf{c}^{\mathsf{T}}\mathbf{x} = \operatorname*{argmin}_{\widetilde{\mathbf{A}}\mathbf{x}=\mathbf{b},\,\mathbf{x}\geq\mathbf{0}} \mathbf{c}^{\mathsf{T}}\mathbf{x}\,.$$

Now, the matrix $\mathbf{Z}$ is orthogonal, so it has full column-rank. Therefore, when multiplying from the left, it keeps the same rank. So $\operatorname{rank}(\mathbf{Z}^{\mathsf{T}}\mathbf{A}) = \operatorname{rank}(\mathbf{Z}\mathbf{Z}^{\mathsf{T}}\mathbf{A}) = \operatorname{rank}(\widetilde{\mathbf{A}}) = k$.

Next, let $\mathbf{x}$ is a feasible point, then

$$\mathbf{b} = \widetilde{\mathbf{A}}\mathbf{x} = \mathbf{Z}\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{Z}\mathbf{Z}^{\mathsf{T}}\mathbf{b}\,. \tag{42}$$

Let $\mathcal{F}_1 = \{\mathbf{x} : \widetilde{\mathbf{A}}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$ and $\mathcal{F}_2 = \{\mathbf{x} : \mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{Z}^{\mathsf{T}}\mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$ be two sets. If $\mathbf{u} \in \mathcal{F}_1$, then

$$\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{u} = (\mathbf{Z}^{\mathsf{T}}\mathbf{Z})\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{u} = \mathbf{Z}^{\mathsf{T}}(\mathbf{Z}\mathbf{Z}^{\mathsf{T}}\mathbf{A})\mathbf{u} = \mathbf{Z}^{\mathsf{T}}\widetilde{\mathbf{A}}\mathbf{u} = \mathbf{Z}^{\mathsf{T}}\mathbf{b}, \quad i.e.\ \mathbf{u} \in \mathcal{F}_2\,.$$

The last equality above holds as $\mathbf{u} \in \mathcal{F}_1$. Therefore, $\mathcal{F}_1 \subseteq \mathcal{F}_2$. Now, we need to prove $\mathcal{F}_2 \subseteq \mathcal{F}_1$. For this, let $\mathbf{u} \in \mathcal{F}_2$. Then, $\widetilde{\mathbf{A}}\mathbf{x} = \mathbf{Z}(\mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{u}) = \mathbf{Z}\mathbf{Z}^{\mathsf{T}}\mathbf{b} = \mathbf{b}$, where the last equality follows from eqn. (42). Therefore, we have $\mathcal{F}_1 = \mathcal{F}_2$ *i.e.* the feasible region induced by $(\widetilde{\mathbf{A}}, \mathbf{b})$ is identical to the feasible region induced by $(\mathbf{Z}^{\mathsf{T}}\mathbf{A}, \mathbf{Z}^{\mathsf{T}}\mathbf{b})$. Therefore, the LP $\min_{\widetilde{\mathbf{A}}\mathbf{x}=\mathbf{b},\,\mathbf{x}\geq\mathbf{0}} \mathbf{c}^{\mathsf{T}}\mathbf{x}$ can be restated as

$$\min \mathbf{c}^{\mathsf{T}}\mathbf{x}\,, \text{ subject to } \mathbf{Z}^{\mathsf{T}}\mathbf{A}\mathbf{x} = \mathbf{Z}^{\mathsf{T}}\mathbf{b}\,, \mathbf{x} \geq \mathbf{0}\,. \tag{43}$$

Note that we have already shown $\operatorname{rank}(\mathbf{Z}^{\mathsf{T}}\mathbf{A}) = k$ (which is $\ll n$). However, $\mathbf{Z}^{\mathsf{T}}\mathbf{A} \in \mathbb{R}^{(k+2)\times m}$ does not have full-rank. Therefore, we can use Gaussian elimination to get the k linearly independent rows of $\mathbf{Z}^{\mathsf{T}}\mathbf{A}$ in $\mathcal{O}(nk^2)$ time and solve eqn. (43) using our framework. See Appendix D.4 for the running time of our algorithms for low-rank constraint matrices.

D.4. Running times for Algorithm 2, `Solve`, and `SolveV`

Finally, we discuss the running times of Algorithm 2, `Solve`, and `SolveV`.

Lemma D.3. *Algorithm 2 called with input matrix $\mathbf{AD} \in \mathbb{R}^{m\times n}$, failure probability η and iteration count t has a total time complexity $\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta + m^3 \log {}^m\!/_\eta + mt + \mathsf{nnz}(\mathbf{A}) \cdot t\right)$.*

Proof. First, recall that $\mathbf{W}$ has $\log {}^m\!/_\eta$ non-zero entries per row and $\mathbf{D}$ is a diagonal matrix. Therefore, $\mathbf{ADW}$ can be computed in $\mathcal{O}(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta)$ time. Then, computing $\mathbf{Q}^{-1/2}$ via the SVD of $\mathbf{ADW}$ takes $\mathcal{O}(m^3 \log {}^m\!/_\eta)$ time. We conclude that the overall time complexity to compute $\mathbf{Q}^{-1/2}$ is $\mathcal{O}(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta + m^3 \log {}^m\!/_\eta)$.

After computing the preconditioner, each inner iteration requires multiplying $\tilde{\mathbf{z}}$ with $\mathbf{Q}^{-1/2}\mathbf{AD}^2\mathbf{A}^T = \mathbf{Q}^{-1/2}(\mathbf{AD})(\mathbf{AD})^T$. Multiplying a vector by $(\mathbf{AD})(\mathbf{AD})^T$ takes $\mathcal{O}(\mathsf{nnz}(\mathbf{A}))$ time and multiplying a vector by $\mathbf{Q}^{-1/2}$ takes $\mathcal{O}(m)$ time. Therefore, the overall time complexity of Algorithm 2 is

$$\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta + m^3 \log {}^m\!/_\eta + mt + \mathsf{nnz}(\mathbf{A}) \cdot t\right)\,.$$

$\square$

We can then immediately derive the time complexity of `Solve` by combining Lemma D.3 and Lemma D.1. We conclude that `Solve` can be implemented by Algorithm 2 with probability at least $1 - \eta$ in time

$$\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta + m^3 \log {}^m\!/_\eta + m \log \frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta} + \mathsf{nnz}(\mathbf{A}) \cdot \log \frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta}\right)\,. \tag{44}$$

We can similarly derive the time complexity of implementing `Solve`$^{\mathsf{v}}$ by combining Lemma D.3 and Lemma D.2. Observe from eqn. (36) that computing $\mathbf{v}$ does not affect the time complexity, since it is a single matrix-vector product using values already computed by Algorithm 2, except pre-multiplying the vector $(\mathbf{ADW})^{\dagger}(\mathbf{AD}^2\mathbf{A}^T\Delta\tilde{\mathbf{y}} - \mathbf{p})$ by $\mathbf{W}$ that takes time $\mathcal{O}(\mathsf{nnz}(\mathbf{A}).\log {}^m\!/_\eta)$ (assuming $\mathsf{nnz}(\mathbf{A}) \geq n$), which is dominated by the cost of computing $(\mathbf{ADW})^{\dagger}$. Therefore, Algorithm 2 combined with eqn. (36) can implement `Solve`$^{\mathsf{v}}$ in time,

$$\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log {}^m\!/_\eta + m^3 \log {}^m\!/_\eta + m \log \frac{n\mu}{\delta} + \mathsf{nnz}(\mathbf{A}) \cdot \log \frac{n\mu}{\delta}\right)\,. \tag{45}$$

Recall that η is the failure probability of the algorithm.

We note that it is straightforward to obtain an overall time complexity for Algorithms 3 and 1 using the above results by setting $\eta = O\left(1/\sqrt{n}\log(\mu_0/\epsilon)\right)$ and applying the union bound.

Running time for Low-rank constraint matrices. In Appendix D.3, the approximate SVD takes $\mathcal{O}(k \cdot \mathsf{nnz}(\mathbf{A}))$ time, computing $\mathbf{Z}^\top \mathbf{A}$ takes another $\mathcal{O}(k \cdot \mathsf{nnz}(\mathbf{A}))$ time, and performing the Gaussian elimination to get k linearly independent rows of $\mathbf{Z}^\top \mathbf{A}$ takes $\mathcal{O}(nk^2)$ time. Therefore, overall it takes $\mathcal{O}(k^2 \cdot \mathsf{nnz}(\mathbf{A}))$ time to preprocess the data (assuming $\mathsf{nnz}(\mathbf{A}) \geq n$). Now, Solve can be implemented in $\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log k/\eta + k^3 \log k/\eta + k \log \frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta} + \mathsf{nnz}(\mathbf{A}) \cdot \log \frac{\sigma_{\max}(\mathbf{AD})n\mu}{\delta}\right)$ time and similarly, Solve$^\mathbf{v}$ can be implemented in $\mathcal{O}\left(\mathsf{nnz}(\mathbf{A}) \cdot \log k/\eta + k^3 \log k/\eta + k \log \frac{n\mu}{\delta} + \mathsf{nnz}(\mathbf{A}) \cdot \log \frac{n\mu}{\delta}\right)$ time.

E. Experiments

We experimentally validated the key predictions of our results. First, we measure the number of iterations needed for Algorithm 1 to converge in relation to the number of variables n, and the precision of the final solution ϵ.

E.1. Generating the random LP

To construct a random LP, we first sample $\mathbf{x}_0 \in \mathbb{R}^n$, $\mathbf{y}_0 \in \mathbb{R}^m$, and $\mathbf{A} \in \mathbb{R}^{m \times n}$, where the entries of $\mathbf{x}_0$ are sampled uniformly from $[0, 10]$ and the entries of $\mathbf{y}_0$ and $\mathbf{A}$ are sampled uniformly from $[-10, 10]$. We then set $\mathbf{s}_0 \in \mathbb{R}^n$ by $[\mathbf{s}_0]_i = 20 \cdot [\mathbf{x}_0]_i^{-1}$. This guarantees that $\mu_0 = 1/n \cdot \mathbf{s}_0^T \mathbf{x}_0 = 20$ and $\|\mathbf{s}_0 \circ \mathbf{x}_0 - \mu_0 \mathbf{1}_n\|_2 = 0$. The generated constraint matrix $\mathbf{A}$ and initial primal-dual point $(\mathbf{x}_0, \mathbf{y}_0, \mathbf{s}_0)$ along with the assumption that the initial point is primal-dual feasible is enough information to exactly describe the linear program.

E.2. Testing Algorithm 1

We first test the predictions of Theorem 2 under a simple instantiation of Solve$^\mathbf{v}$. To implement Solve$^\mathbf{v}$, we sample a random vector $\mathbf{v} \in \mathbb{R}^n$ randomly from the unit sphere and rescale it so that $\mathbf{AS}^{-1}\mathbf{v} = \delta$, where δ is the accuracy parameter of Solve$^\mathbf{v}$. We then use a standard linear system solver to solve the perturbed system given by Equation 4. Note that this instantiation of Solve$^\mathbf{v}$ would not be useful in practice, but it is nevertheless useful to test whether the outer iteration complexity of Theorem 2 holds empirically. Figures 1 and 2 summarize our results on the relationship between the number of outer iterations versus n and ϵ.

We find that, in all displayed experiments, primal infeasibility is around 10^{-10} and does not change substantially with n or ϵ. We conclude that the error-adjustment effectively keeps the iterates primal-feasible, modulo minor numerical errors.

E.3. Testing an iterative instantiation of Solve$^\mathbf{v}$ (Algorithm 2)

We repeat the above two experiments while using the iterative linear system solver described in Appendix D. We note that the iterative solver only requires a few number of iterations (< 20) in the parameter regime we test. We avoid a more in-depth analysis of the PCG iteration complexity, as this was already performed in (Chowdhury et al., 2020). Overall, we find that there is no notable difference between using the perturbed Solve$^\mathbf{v}$ method or the iterative instantiation. Results of our experiments are summarized in Figures 3 and 4 below.
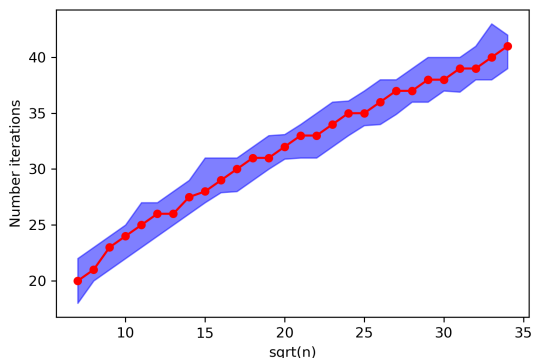
Figure 3. This graph demonstrates that the linear relationship between the number of iterations and $\sqrt{n}$ continues to hold when using the iterative instantiation of $\texttt{Solve}^{\textbf{v}}$. The line shows the median number of iterations and the intervals designate the 10% and 90% quantiles out of 60 repetitions. Other parameters are $m = 20$, $\epsilon = 0.1$, $\delta = 0.001$, and (sketch size) $w = 60$.
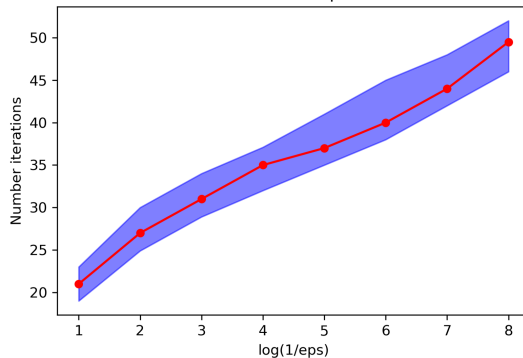

Figure 4. This graph demonstrates that the linear relationship between the number of iterations and $\log(1/\epsilon)$ continues to hold when using the iterative instantiation of $\texttt{Solve}^{\textbf{v}}$. The line shows the median number of iterations and the intervals designate the 10% and 90% quantiles out of 60 repetitions. Other parameters are $m = 30$, $n = 70$, $\delta(\epsilon) = \epsilon$, and (sketch size) $w = 60$.

E.4. Running time of Algorithm 1

Finally, we evaluated how the use of an inexact solver affects the empirical running time on randomly generated LPs. Figure 5 shows the empirical running time of Algorithm 1 versus the same IPM algorithm where the normal equation is solved by directly inverting the normal matrix and computing $\Delta\mathbf{y} = (\mathbf{A}\mathbf{D}^2\mathbf{A}^T)^{-1}\mathbf{p}$. When implementing Algorithm 1, we instantiated $\texttt{Solve}^{\textbf{v}}$ using Algorithm 2, where the matrix $\mathbf{W}$ is a normalized Gaussian sketching matrix. We show the running time of the inexact IPM method *(1)* in total, *(2)* excluding the time to form $\mathbf{W}$, and *(3)* excluding the time to execute $\texttt{Solve}^{\textbf{v}}$.

Our experiments show that inexactly solving the normal equations at each step can result in improved running time when the time to inexactly solve the normal equations is negligible. However, our basic implementation of the $\texttt{Solve}^{\textbf{v}}$ is insufficient to show improved total performance at the scale of the LPs in our experiments. First steps towards improving the running time of our inexact implementation would be *(1)* using a sparse or structured sketching matrix for $\mathbf{W}$, rather than a dense Gaussian sketch, *(2)* computing the preconditioner with a more efficient decomposition, rather than SVD, and *(3)* using sparse matrix libraries.

The fact that, even at small scales, our experiments show inexact solutions to the normal equations can improve running time when the $\texttt{Solve}^{\textbf{v}}$ cost is excluded is a promising sign for future benefit in solving sparse large-scale LPs, as running time improvements to using randomized linear algebra and, in particular, matrix sketching for large matrices has already been established in prior literature (Halko et al., 2011; Dahiya et al., 2018; Avron et al., 2010). Likewise, the running time benefits of PCG on sparse matrices has also been established in prior literature (Quarteroni & Valli, 1996; Press et al., 2007).
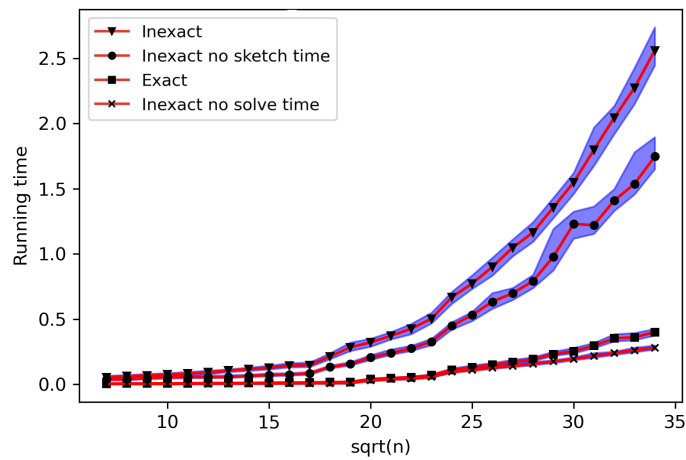
Figure 5. This shows the average running time in seconds for the exact and inexact IPM algorithms, where the inexact (PCG) running time is shown in total, excluding the time to construct the sketch, and excluding the time to construct the sketch/run PCG. The line shows the median number of iterations and the intervals designate the 10% and 90% quantiles out of 60 repetitions. Other parameters are $m = 20$, $\epsilon = 0.1$, $\delta = 0.001$, and (sketch size) $w = 60$.