MIT Open Access Articles

Improvements to Quantum Interior Point Method for Linear Optimization

Massachusetts Institute of Technology

Improvements to Quantum Interior Point Method for Linear Optimization

MOHAMMADHOSSEIN MOHAMMADISIAHROUDI, Industrial and System Engineering Department, Lehigh University, Bethlehem, United States

ZEGUAN WU, Industrial and System Engineering Department, Lehigh University, Bethlehem, United States

BRANDON AUGUSTINO, Sloan School of Managementhttps://orcid.org/0000-0003-1953-1971, Massachusetts Institute of Technology, Cambridge, United States

ARIELLE CARR, Computer Science and Engineering Department, Lehigh University, Bethlehem, United States

TAMÁS TERLAKY, Industrial and System Engineering Department, Lehigh University, Bethlehem, United States

Quantum linear system algorithms (QLSA) have the potential to speed up Interior Point Methods (IPM). However, a major bottleneck is the inexactness of quantum Tomography to extract classical solutions from quantum states. In addition, QLSAs are sensitive to the condition number, and this sensitivity is exacerbated when the Newton systems arising in IPMs converge to a singular matrix. Recently, an Inexact Feasible Quantum IPM (IF-QIPM) has been developed that addresses the inexactness of QLSAs. However, this method requires a large number of gates and qubits to be implemented. Here, we propose a new IF-QIPM using the normal equation system, which requires less number of gates and qubits. To mitigate the sensitivity to the condition number and other input data-related parameters, we use preconditioning coupled with iterative refinement to obtain better complexity. Finally, we demonstrate the effectiveness of our approach on IBM Qiskit simulators.

CCS Concepts: • **Theory of computation → Linear programming**; **Quantum complexity theory**; **Preconditioning**.

Additional Key Words and Phrases: Quantum Linear System Algorithm, Quantum Interior Point Method, Linear Optimization, Iterative Refinement, Preconditioning

1 Introduction

Mathematical optimization problems arise in many fields and their solution yields significant computational challenges. Researchers have attempted to develop quantum optimization algorithms, such as the Quantum Approximation Optimization Algorithm (QAOA) for unconstrained quadratic binary optimization problems [14], and a quantum subroutine for simplex algorithm [36]. Another class of quantum algorithms are Quantum Interior Point Methods (QIPMs) [4, 23, 31], which are hybrid-classical IPMs that use Quantum Linear System Algorithms (QLSAs) to solve the Newton system at each IPM iteration. Before reviewing prior work on QIPMs for Linear Optimization problems (LOP), we provide the necessary definitions, fundamental results, and properties.

Definition 1.1 (LOP: Standard Formulation). For $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and matrix $A \in \mathbb{R}^{m \times n}$ with rank$(A) = m \le n$, the LOP is defined as

$$
\begin{array}{llll}
 & \min\ c^T x, & & \max\ b^T y, \\
(\text{P}) & \text{s.t.}\ \ Ax = b, & (\text{D}) & \text{s.t.}\ \ A^T y + s = c, \\
 & \qquad x \ge 0, & & \qquad\quad s \ge 0,
\end{array}
$$

where $x \in \mathbb{R}^n$ is the vector of primal variables, and $y \in \mathbb{R}^m$, $s \in \mathbb{R}^n$ are vectors of the dual variables. Problem (P) is called primal problem and problem (D) is called dual problem.

In each step of IPMs, a Newton system is solved to determine the Newton step. There are four approaches:

(1) Full Newton System

$$
\begin{bmatrix} 0 & A & 0 \\ A^T & 0 & I \\ 0 & S & X \end{bmatrix}
\begin{bmatrix} \Delta y \\ \Delta x \\ \Delta s \end{bmatrix} =
\begin{bmatrix} 0 \\ 0 \\ \beta\mu e - Xs \end{bmatrix}; \tag{FNS}
$$

(2) Augmented System

$$
\begin{bmatrix} 0 & A \\ A^T & -D^{-2} \end{bmatrix}
\begin{bmatrix} \Delta y \\ \Delta x \end{bmatrix} =
\begin{bmatrix} 0 \\ s - \beta\mu X^{-1} e \end{bmatrix}; \tag{AS}
$$

(3) Normal Equation System

$$
AD^2 A^T \Delta y = Ax - \beta\mu A S^{-1} e; \tag{NES}
$$

(4) Orthogonal Subspaces System

$$
\begin{bmatrix} -XA^T & SV \end{bmatrix}
\begin{bmatrix} \Delta y \\ \lambda \end{bmatrix} = \beta\mu e - Xs, \tag{OSS}
$$

where $X = \text{diag}(x)$, $S = \text{diag}(s)$, $D = S^{-1/2}X^{1/2}$, $\mu = \frac{x^T s}{n}$, β, $0 < \beta < 1$, and e is an all-one vector. Further, the columns of V form a basis for the null space of A.

System	Size of system	Symmetric	Positive Definite	Rate of Condition Number
FNS	$2n + m$	✗	✗	$O\left(\frac{1}{\mu^2}\right)$
AS	$n + m$	✓	✗	$O\left(\frac{1}{\mu^2}\right)$
NES	m	✓	✓	$O\left(\frac{1}{\mu^2}\right)$
OSS	n	✗	✗	$O\left(\frac{1}{\mu}\right)$

Table 1. Characteristics of Coefficient Matrix of Different Newton Systems

Table 1 shows that NES has a smaller size since in most practical LO problems $m \ll n$. In addition, its symmetric positive definite coefficient matrix is favorable since classically it can be solved faster with Cholesky factorization or conjugate gradient. It is also more adaptable to QLSAs since QLSAs are able to solve linear systems with a Hermitian matrix. To solve linear systems whose matrix is not Hermitian, like (OSS) must be embedded in a bigger system with a Hermitian matrix. Furthermore, solving normal equations has better complexity on quantum machines by using quantum singular value transformation [9]. Thus, NES has a better structure compared to others, however, its condition number grows at a faster rate than the one of the OSS. The key takeaway is that inexact solutions of NES, FNS, and AS may lead to infeasibility, whereas inexact solutions to OSS allow to preserve the feasibility [32]. In this paper, we develop an Inexact Feasible Quantum Interior Point Method (IF-QIPM) based on a modified version of the NES.

1.1 Related Works

Inexact IPMs: IPMs can be divided into two main types: Feasible IPMs and Infeasible IPMs. Feasible IPMs (F-IPMs) necessitate an initial feasible interior point to begin the optimization process. These methods often utilize a self-dual embedding model of the LOP, facilitating the construction of a feasible interior solution [38]. Conversely, Infeasible IPMs (I-IPMs) start with a solution that is infeasible but strictly positive. Theoretical studies indicate that the optimal time complexity for F-IPMs in addressing LO problems is $O(\sqrt{n}L)$, where L represents the binary length of the input data. In contrast, the optimal time complexity for I-IPMs applied to LO problems is $O(nL)$. Although F-IPMs theoretically outperform I-IPMs in terms of time complexity, both types are effective in practically solving LO problems, as noted by [44].

The prevailing approach to solve LOPs is Exact IPMs, in which the Newton direction is calculated by solving NES using Cholesky factorization [38]. Thus, although IPMs enjoy a fast convergence rate, the cost per iteration of IPMs is considerably high when applied to large-scale LOPs. In an effort to reduce the per-iteration cost of IPMs, inexact infeasible IPMs (II-IPMs) were proposed, in which the Newton system is solved with an iterative method, e.g., using conjugate gradient methods (CGMs) [1, 35].

Mizuno and colleagues initially explored the convergence properties of II-IPMs through a series of studies [15, 28]. Subsequently, Baryamureeba and Steihaung demonstrated the convergence of a variant of the I-IPM originally proposed by Kojima [24], incorporating an inexact Newton step [6]. Furthermore, Korzak established that his specific version of an II-IPM operates within polynomial time complexity [25].

As early as the 1980s, partial update techniques were utilized to compute the inexact Newton direction through several rank-one updates to the inverse of the NES matrix. This strategy achieved the total complexity of $O(n^3L)$ arithmetic operations for solving LOPs [38]. This concept has been significantly strengthened in recent years through the adoption of advanced techniques such as fast matrix multiplication, spectral sparsification, and stochastic central path methods [11, 26]. By leveraging these modern techniques, the complexity of IPMs can be reduced to $O(n^wL)$, where $w < 2.3729$ is the matrix multiplication constant [40].

The use of Preconditioned CGMs (PCGMs) in II-IPMs has been extensively studied by several researchers [1, 35]. AL-Jeiroudi and Gondzio utilized the I-IPM framework from Wright [44] to solve (AS) using a PCGM [1]. In a similar vein, Monteiro and O'Neal applied a PCGM to solve (NES) [35]. Inexact linear systems algorithms like CG exhibit favorable dependence on dimension compared to factorization methods and are able to exploit sparsity patterns present in the Newton system. The rub is that these inexact approaches depend on a condition number bound, which could pose a challenge. To tackle the ill-conditioned Newton system, they used the so-called maximum weight basis (MWB) precondition [34].

Further investigations into II-IPMs have been conducted by Bellavia, who examined their convergence for general convex optimization problems [7], and by Zhou and Toh, who developed an II-IPM specifically for Semidefinite Optimization (SDO) problems [47]. It has been established that the best-known bound for the number of iterations required by II-IPMs to solve LOPs is $O(n^2L)$.

All mentioned inexact versions of IPMs tend to be inherently infeasible, as the inexact solutions to the conventional formulations of Newton systems—such as FNS, AS, and NES—result in infeasibility. Gondzio highlighted conceptually that if Newton systems within IPMs can be solved inexactly while still maintaining feasibility, IPMs could achieve the best iteration complexity of $O(\sqrt{n}L)$ for quadratic optimization [21]. To leverage this superior complexity, inexact feasible IPMs (IF-IPMs) have been introduced. These methods utilize the OSS system to derive inexact but feasible Newton directions [4, 32]. In this paper, we present a novel IF-IPM that, at each iteration, employs a modified version of the NES that is solved inexactly. This modification is crucial as it ensures that the inexact direction remains within the feasible region.

Quantum IPMs: QIPMs were first proposed by Kerenidis and Prakash [23], who sought to decrease the cost per iteration by classically estimating the Newton step through the use of a QLSA and quantum state tomography.

Adopting this approach, Casares and Martin-Delgado [8] developed a predictor-correcter QIPM for LO. However, these algorithms were proposed and analyzed using an *exact* IPM framework, which is invalid because the use of quantum subroutines naturally introduces errors into the solution and leads to *inexactness* in the Newton step. Specifically, without further safeguards this inexactness means that the sequence of iterates generated the algorithms in [8, 23] may leave the feasible set, and so convergence cannot be guaranteed.

To address these issues, Augustino et al. [4] proposed an inexact-infeasible QIPM (which closely quantized the II-IPM of [47]) and a novel *inexact-feasible* QIPM using OSS. The latter algorithm was shown to solve LOPs to precision $\zeta \in (0, 1)$ using at most[1]

$$\widetilde{O}_{n,\kappa,\frac{1}{\zeta}}\left(n^2\frac{\kappa^2}{\zeta}\right)$$

QRAM queries and $\widetilde{O}_{n,\kappa,\frac{1}{\zeta}}\left(n^{2.5}\right)$ arithmetic operations, where κ is an upper bound on the Newton system coefficient matrices that arise over the run of the algorithm.

Mohammadisiahroudi et al. [31, 32] specialized the algorithms in [4] to LO and used iterative refinement techniques to exponentially improve the dependence of the algorithms in [4] on precision and the condition number bound. In particular, [31] developed an inexact-infeasible QIPM (II-QIPM), which addresses the inexactness of QLSA. [32] improved this complexity by developing a short-step IF-QIPM for LOPs with

$$\widetilde{O}_{n,\kappa_A,\|A\|,\omega}(n^2L\omega^2\|A\|^2\kappa_A^2)$$

QRAM queries and $O(n^{2.5}L)$ arithmetic operations where ω is an upper bound for the norm of the optimal solution and κ_A is the condition number of matrix A.

Note that the use of iterative refinement techniques indirectly led to another improvement in the complexity, reducing the dependence on a condition number bound κ for the intermediate Newton systems with the condition number κ_A of the input matrix A [33]. IF-QIPMs built on similar techniques have also been developed for linearly constrained quadratic optimization problems in [45] and second-order cone optimization problems in [5].

A month after the original submission of this paper, Apers and Gribling [3] introduced a QIPM for LO that operates independently of any condition number. Under specific conditions[2], and with access to QRAM, this approach reports a quantum speedup for "tall" LOPs. In these problems, all constraints are inequalities, and the number of constraints substantially exceeds the number of variables. Instead of employing QLSAs to resolve the Newton system, the method calculates Newton steps using spectral approximations of the Hessian. Although their worst-case complexity has no dependence on the condition number, for general LO problems their approach can have unfavorable dimension dependence of $n^{7.5}$ and their complexity has a linear dependence on the inverse of the precision. In this paper, we show how iterative refinement and preconditioning can mitigate the impact of condition number on the complexity of the proposed IF-QIPM using modified NES.

Iterative refinement. Iterative refinement (IR) is a well-established method used to enhance numerical accuracy when solving linear systems of equations [20, 42]. Gleixner et al. pioneered the adaptation of this technique to LOPs, introducing the first iterative refinement methodology specifically tailored for constrained optimization [19]. They successfully applied this method to achieve precise solutions to LOPs using limited-precision oracles [18].

In parallel developments, other studies have illustrated the utility of IR in achieving high-precision solutions for LOPs, particularly those emerging in the context of integer optimization [12, 13]. More recently, Mohammadisiahroudi et al. applied IR techniques to obtain exact solutions for LOPs within the framework of limited-precision

---

[1]The $\widetilde{O}_{\alpha,\beta}\left(g(x)\right)$ notation indicates that quantities polylogarithmic in α, β and $g(x)$ are suppressed.
[2]The algorithmic speed-up proposed in [3] applies to LO problems where the bit-complexity of each entry in A, b, c is at most polylog(n, m), and $n \geq m^{10}$, with n and m denoting the number of constraints and variables in their standard dual form, respectively.

QIPMs. Their research demonstrated that IR could effectively mitigate the effects of ill-conditioned Newton systems, thereby reducing the overall computational complexity associated with QIPMs [31].

1.2 Contributions of this paper

The contributions of this paper are outlined as follows:

- A novel IF-QIPM is proposed by modifying the NES, which ensures the inexact Newton direction remains within the feasible region. This modification allows our IF-QIPM to achieve the best-known iteration complexity of $O(\sqrt{n}L)$, an improvement over traditional inexact infeasible IPMs that use the NES and exhibit a complexity of $O(n^2 L)$.
- The modified NES features an m-by-m symmetric positive definite matrix, which is advantageous for both classical and quantum linear solvers. Notably, while the state-of-the-art QIPM requires solving systems of dimension $2n$, our proposed IF-QIPM only needs to solve m-dimensional systems and in many practical problems, m is usually much smaller than n. Additionally, the favorable structure of the NES facilitates the use of more efficient matrix inversion techniques on quantum computers [9], enhancing the complexity performance of our IF-QIPM.
- Iterative refinement and preconditioning are utilized to mitigate the effects of the condition number of the Newton system. The integration of these techniques into our IF-QIPM results in a speed-up relative to classical solvers in terms of dimension and compared to previous QIPMs, a speed-up relative to the condition number and other problem-dependent parameters.
- We have implemented a version of our IF-QIPM using the QISKIT platform and the IBM QLSA simulator. Numerical experiments demonstrate that iterative refinement and preconditioning can significantly alleviate issues related to the condition number.

The rest of this paper is structured as follows. Section 2 reviews some notations and preliminaries. In Section 3, a modified NES is utilized to produce an inexact but feasible Newton step, and a short-step Inexact Feasible IPM is developed. Section 4 explores how we use QLSA to solve the modified NES system in order to develop an IF-QIPM. In Section 5, an iterative refinement method coupled with preconditioning is developed to address the impacts of the condition number on the complexity. Finally, numerical experiments using the IBM Qiskit simulator are carried out in Section 6, and Section 7 concludes the paper.

2 Preliminaries

We denote the quantity a to the k-th power by a^k, and the notation $a^{(k)}$, indicates the value of a at iteration k of an algorithm. In the rest of the paper, $\|M\| = \|M\|_2$ is the 2-norm of matrix M, and $\|M\|_F$ is the Frobenius norm of M. Further, $\mathbb{R}^n$ denotes the set of n-dimensional vectors of real numbers and $\mathbb{C}^n$ denotes the set of n-dimensional vectors of complex numbers.

The set of feasible primal-dual solutions is defined as

$$\mathcal{PD} = \left\{ (x, y, s) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \mid Ax = b, \ A^T y + s = c, \ (x, s) \geq 0 \right\}.$$

Then, the set of all feasible interior solutions is defined as

$$\mathcal{PD}^0 = \left\{ (x, y, s) \in \mathcal{PD} \mid (x, s) > 0 \right\}.$$

In this work, we assume $\mathcal{PD}^0$ is not empty. By the Strong Duality theorem, optimal solutions exist and belong to the set $\mathcal{PD}^*$ defined as

$$\mathcal{PD}^* = \left\{ (x, y, s) \in \mathcal{PD} \mid x^T s = 0 \right\}.$$

Let $\zeta \geq 0$, then the set of ζ-optimal solutions is defined as

$$\mathcal{PD}_\zeta = \left\{ (x, y, s) \in \mathcal{PD} \; \middle| \; \frac{x^T s}{n} \leq \zeta \right\}.$$

Assuming integer data, we denote the binary length of the input data by

$$L = mn + m + n + \sum_{i,j} \lceil \log(|a_{ij}| + 1) \rceil + \sum_i \lceil \log(|c_i| + 1) \rceil + \sum_j \lceil \log(|b_j| + 1) \rceil,$$

where a_{ij} represents the ij-element of matrix A. It is well-established that one can find the exact solution of an LOP using a rounding procedure if the solution provided IPM has 2^{-2L} precision [38, 44].

Generally, IPMs approach the optimal set by traversing along a central path. The central path is defined as

$$\mathcal{CP} = \left\{ (x, y, s) \in \mathcal{PD} \; \middle| \; x_i s_i = \mu \;\; \text{for} \;\; i \in \{1, \ldots, n\} \right\},$$

where $\mu = \frac{x^T s}{n}$. As moving exactly on the central path is not possible, Newton steps need to be calculated in a way that the iterates remain in a neighborhood of the central path. For any $\theta \in [0, 1)$, a small neighborhood of the central path is defined as

$$\mathcal{N}(\theta) = \left\{ (x, y, s) \in \mathcal{PD}^0 \; \middle| \; \|XSe - \mu e\|_2 \leq \theta \mu \right\}.$$

Algorithm 1 demonstrate the generic scheme of IPMs. In this paper, we use a feasible short-step version of IPMs

---

Algorithm 1 Conventional IPM

1: Choose $0 < \zeta, \theta, \beta < 1$.
2: $k \leftarrow 0$
3: Choose initial feasible interior solution $(x^0, y^0, s^0) \in \mathcal{N}(\theta)$
4: **while** $(x^{(k)}, y^{(k)}, s^{(k)}) \notin \mathcal{PD}_\zeta$ **do**
5: $\quad \mu^{(k)} \leftarrow \frac{(x^{(k)})^T s^{(k)}}{n}$
6: $\quad (\Delta x^{(k)}, \Delta y^{(k)}, \Delta s^{(k)}) \leftarrow$ Solve FNS, AS, NES, or OSS
7: $\quad$ Choose a proper steplength $\alpha^{(k)} \in (0, 1)$
$$(x^{(k+1)}, y^{(k+1)}, s^{(k+1)}) \leftarrow (x^{(k)}, y^{(k)}, s^{(k)}) + \alpha^{(k)} (\Delta x^{(k)}, \Delta y^{(k)}, \Delta s^{(k)})$$
8: $\quad k \leftarrow k + 1$
9: **end while**
10: Return $(x^{(k)}, y^{(k)}, s^{(k)})$

---

in which β is close to one, i.e., there is a small reduction in the optimality gap at each iteration. Consequently, we can take a full Newton step, $\alpha^{(k)} = 1$, at each iteration [38].

When we refer to complexity, we mean the number of queries to the controlled version of the input oracles and their inverses. We define $O(\cdot)$ as

$$f(x) = O(g(x)) \iff \text{there exist } \gamma \in \mathbb{R}, \; c \in \mathbb{R}_+ \text{ such that } f(x) \leq cg(x) \; \forall x > \gamma.$$

We also define $\widetilde{O}(f(x)) = O(f(x) \cdot \text{polylog}(f(x)))$. When the function depends polylogarithmically on other variables, we write $\widetilde{O}_{a,b}(f(x)) = O(f(x) \cdot \text{polylog}(a, b, f(x)))$. It should be mentioned that the complexity of a classical algorithm refers to the number of arithmetic operations. When we refer to the complexity of quantum algorithms, we mean the number of queries to the controlled version of the input oracles and their inverses, and we also specify the bound for classical arithmetic operations

3 Inexact-Feasible Newton Step using NES

In this section, we discuss how an inexact but feasible Newton direction can be calculated using the NES at each iteration of the proposed IF-QIPM.

To compute the Newton step, we need to determine $(\Delta x^{(k)}, \Delta y^{(k)}, \Delta s^{(k)})$ such that

$$
\begin{aligned}
A\Delta x^{(k)} &= 0, \\
A^T \Delta y^{(k)} + \Delta s^{(k)} &= 0, \\
X^{(k)}\Delta s^{(k)} + S^{(k)}\Delta x^{(k)} &= \beta\mu^{(k)}e - X^{(k)}s^{(k)}.
\end{aligned}
\tag{1}
$$

As discussed in the introduction, we are interested in using NES as it has a smaller symmetric positive definite matrix, favorable for both quantum and classical linear system solvers. First note that an exact solution Δy to NES satisfies

$$
A(D^{(k)})^2 A^T \Delta y^{(k)} = Ax^{(k)} - \beta\mu^{(k)}A(S^{(k)})^{-1}e.
\tag{2}
$$

Having obtained $\Delta y^{(k)}$, we then compute $\Delta s^{(k)}$ and $\Delta x^{(k)}$ using as follows:

$$
\Delta s^{(k)} = -A^T \Delta y^{(k)},
\tag{3a}
$$

$$
\Delta x^{(k)} = \beta\mu(S^{(k)})^{-1}e - x^{(k)} - (D^{(k)})^2\Delta s^{(k)}.
\tag{3b}
$$

Now, when system (2) is solved inexactly, the resulting solution $\overline{\Delta y}^{(k)}$ satisfies

$$
A(D^{(k)})^2 A^T \overline{\Delta y}^{(k)} = Ax^{(k)} - \beta\mu^{(k)}A(S^{(k)})^{-1}e + r^{(k)},
$$

where r is the residual as $\|Ax - \beta\mu A(S^{(k)})^{-1}e - A(D^{(k)})^2 A^T \overline{\Delta y}^{(k)}\|$. In place of (3a) and (3b), we now have

$$
\begin{aligned}
A\Delta x^{(k)} &= r^{(k)}, \\
A^T \Delta y^{(k)} + \Delta s^{(k)} &= 0, \\
X^{(k)}\Delta s^{(k)} + S^{(k)}\Delta x^{(k)} &= \beta\mu^{(k)}e - X^{(k)}s^{(k)}.
\end{aligned}
$$

While dual feasibility is preserved, the same cannot be said for primal feasibility. In order to preserve primal feasibility using inexact solutions to (2), one can alternatively solve

$$
\begin{aligned}
\Delta s^{(k)} &= -A^T \Delta y^{(k)}, \\
\Delta x^{(k)} &= \beta\mu^{(k)}(S^{(k)})^{-1}e - x^{(k)} - (D^{(k)})^2\Delta s^{(k)} - v^{(k)},
\end{aligned}
$$

where $Av^{(k)} = r^{(k)}$. Updating $\Delta x^{(k)}$ in this way, we *correct* primal infeasibility, and one can verify that

$$
\begin{aligned}
A\Delta x^{(k)} &= 0, \\
A^T \Delta y^{(k)} + \Delta s^{(k)} &= 0, \\
X^{(k)}\Delta s^{(k)} + S^{(k)}\Delta x^{(k)} &= \beta\mu^{(k)}e - X^{(k)}s^{(k)} + r'^{(k)},
\end{aligned}
$$

where $r'^{(k)} = -S^{(k)}v^{(k)}$. Next, we describe two procedures to calculate $v^{(k)}$ efficiently.

Procedure A. Since A has full row rank, we can calculate

$$
\hat{A} = A^T(AA^T)^{-1},
$$

as a preprocessing step before the IPM starts. Then, in each iteration, we calculate $v^{(k)} = \hat{A}r^{(k)}$ using classical matrix-vector products. To recover the convergence analysis of [32], the residual must satisfy $\|r'^{(k)}\| \leq \eta\mu^{(k)}$ for

$\eta \in [0, 1)$. One can show that this requirement amounts to

$$\|r^{(k)}\| \le \eta \frac{\mu^{(k)}}{\|s^{(k)}\|_\infty \sigma_{\max}},$$

where $\sigma_{\max}$ is the maximum singular value of A. Since $\|s^{(k)}\|_\infty$ and $\sigma_{\max}$ can be exponentially large, this residual bound can be unacceptably small.

Procedure B. Letting $\hat{B}$ be an arbitrary basis for matrix A, we can calculate

$$v^{(k)} = (v_{\hat{B}}^{(k)}, v_N^{(k)}) = (A_{\hat{B}}^{-1} r^{(k)}, 0).$$

It is straightforward to verify that $Av^{(k)} = r^{(k)}$. Now, we show that this procedure coupled with an appropriate modification of the NES leads to a favorable residual bound.

Since A has full row rank, one can choose an arbitrary basis $\hat{B}$, and subsequently calculate $A_{\hat{B}}^{-1}$, $\hat{A} = A_{\hat{B}}^{-1} A$, and $\hat{b} = A_{\hat{B}}^{-1} b$. These steps require $O(m^2 n)$ arithmetic operations and take place only once prior to the first iteration of IPM. The cost of this preprocessing can be reduced by leveraging the structure of A. For example, there is no need for this preprocessing if the problem is in the canonical form as

$$\begin{array}{ll}
& \min\ c'^T x, \qquad\qquad\qquad\quad \max\ b'^T y, \\
\text{(P')} & \text{s.t.}\ A'x \ge b', \qquad \text{(D')} \qquad \text{s.t.}\ A'^T y \le c', \\
& \qquad x \ge 0, \qquad\qquad\qquad\qquad\qquad y \ge 0.
\end{array}$$

The reason is that after adding slack variables to the primal constraint, we have $[A', I]$ and the identity part can be used as $A_{\hat{B}}$, which does not require inversion. In this paper, we neglect the cost of preprocessing, since it can be avoided by using the following reformulation.

$$\begin{array}{l}
\min\ c^T x, \\
\text{s.t.}\ Ax + u = b, \\
\quad\ -Ax + u' = -b, \\
\quad\ x, u, u' \ge 0.
\end{array}$$

This is a standard form LOP, but its interior is empty. This issue is remedied upon using the self-dual embedding model [46] and we refer the readers to [32] for details. While this formulation does not require calculation, the price one pays for this case is using a larger system. In the rest of this paper, we assume that we are working with the preprocessed problem with input data that includes an identity matrix.

Now, we can modify system (NES) with coefficient matrix $M^{(k)} = A(D^{(k)})^2 A^T$ and right-hand side vector $\sigma^{(k)} = b - \beta\mu^{(k)} A(S^{(k)})^{-1} e$ to

$$\hat{M}^{(k)} z^{(k)} = \hat{\sigma}^{(k)} \qquad\qquad\qquad \text{(MNES)}$$

where

$$\hat{M}^{(k)} = (D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1} M^{(k)} ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T = (D_{\hat{B}}^{(k)})^{-1} \hat{A}(D^{(k)})^2 \hat{A}^T (D_{\hat{B}}^{(k)})^{-1},$$

$$\hat{\sigma}^{(k)} = (D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1} \sigma^{(k)} = (D_{\hat{B}}^{(k)})^{-1} \hat{b} - \beta\mu^{(k)} (D_{\hat{B}}^{(k)})^{-1} \hat{A}(S^{(k)})^{-1} e.$$

We use the following procedure to find the Newton direction by solving (MNES) inexactly with QLSA+QTA.

Step 1. Find $\tilde{z}^{(k)}$ such that $\hat{M}^{(k)} \tilde{z}^{(k)} = \hat{\sigma}^{(k)} + \hat{r}^{(k)}$ and $\|\hat{r}^{(k)}\| \le \frac{\eta}{\sqrt{1+\theta}} \sqrt{\mu^{(k)}}$.

Step 2. Calculate $\widetilde{\Delta y}^{(k)} = ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T \tilde{z}^{(k)}$.

Step 3. Calculate $v^{(k)} = (v_{\hat{B}}^{(k)}, v_{\hat{N}}^{(k)}) = (D_{\hat{B}}^{(k)} \hat{r}^{(k)}, 0)$.

Step 4. Calculate $\widetilde{\Delta s}^{(k)} = -A^T \widetilde{\Delta y}^{(k)}$.

Step 5. Calculate $\widetilde{\Delta x}^{(k)} = \beta\mu^{(k)}(S^{(k)})^{-1}e - x^{(k)} - (D^{(k)})^2 \widetilde{\Delta s}^{(k)} - v^{(k)}$.

It is noteworthy that, this modification technique is similar to MWB preconditioning techniques of [1, 35]. One major difference is that we modify the NES for a feasible IPM setting, although others apply it for infeasible IPMs. In addition, we preprocess the data initially, before starting IPM although in preconditioning, one needs to do the modification, calculating the precondition with $O(n^3)$ cost, in each iteration. Thus, we show that the complexity of our approach has better dimension dependence $O(n^{2.5})$ although the complexity of other infeasible approaches with preconditioning has $O(n^5)$ dimension dependence. In Section 5, we explore how quantum computing can speed up the preconditioning part.

LEMMA 3.1. *For the Newton direction $(\widetilde{\Delta x}^{(k)}, \widetilde{\Delta y}^{(k)}, \widetilde{\Delta s}^{(k)})$, we have*

$$A\widetilde{\Delta x}^{(k)} = 0,$$

$$A^T\widetilde{\Delta y}^{(k)} + \widetilde{\Delta s}^{(k)} = 0, \tag{4}$$

$$X^{(k)}\widetilde{\Delta s}^{(k)} + S^{(k)}\widetilde{\Delta x}^{(k)} = \beta\mu^{(k)}e - X^{(k)}s^{(k)} + r'^{(k)}.$$

where $r'^{(k)} = -S^{(k)}v^{(k)}$.

PROOF. For the Newton direction $(\widetilde{\Delta x}^{(k)}, \widetilde{\Delta y}^{(k)}, \widetilde{\Delta s}^{(k)})$, one can verify that $\hat{M}^{(k)}\tilde{z}^{(k)} = \hat{\sigma}^{(k)} + \hat{r}^{(k)}$ implies

$$M^{(k)}\widetilde{\Delta y}^{(k)} = \sigma^{(k)} + A_{\hat{B}}D_{\hat{B}}^{(k)}\hat{r}^{(k)}.$$

For the first equation of (4), we can write

$$\begin{aligned}
A\widetilde{\Delta x}^{(k)} &= A(\beta\mu^{(k)}(S^{(k)})^{-1}e - x^{(k)} - (S^{(k)})^{-1}X^{(k)}\widetilde{\Delta s}^{(k)} - v^{(k)}) \\
&= A(\beta\mu^{(k)}(S^{(k)})^{-1}e - x^{(k)} - (S^{(k)})^{-1}X^{(k)}(-A^T\widetilde{\Delta y}^{(k)}) - v^{(k)}) \\
&= \beta\mu^{(k)}A(S^{(k)})^{-1}e - Ax^{(k)} + A(S^{(k)})^{-1}X^{(k)}A^T\widetilde{\Delta y}^{(k)} - Av^{(k)} \\
&= \beta\mu^{(k)}A(S^{(k)})^{-1}e - Ax^{(k)} + \sigma^{(k)} + A_{\hat{B}}D_{\hat{B}}^{(k)}\hat{r}^{(k)} - Av^{(k)} \\
&= 0.
\end{aligned}$$

The second and third equations of (4) are obtained from Steps 4 and 5. □

To have a convergent IPM, we need $\|r'^{(k)}\|_\infty \leq \eta\mu^{(k)}$, where $0 \leq \eta < 1$ is an enforcing parameter. The next lemma gives an analogous residual bound for the modified NES.

LEMMA 3.2. *For the Newton direction $(\widetilde{\Delta x}^{(k)}, \widetilde{\Delta y}^{(k)}, \widetilde{\Delta s}^{(k)})$, if the residual $\|\hat{r}^{(k)}\|_\infty \leq \frac{\eta}{\sqrt{1+\theta}}\sqrt{\mu^{(k)}}$, then $\|r'^{(k)}\|_\infty \leq \eta\mu^{(k)}$.*

PROOF. As $(x^{(k)}, y^{(k)}, s^{(k)}) \in \mathcal{N}(\theta)$, we have

$$(1-\theta)\mu^{(k)} \leq x_i s_i \leq (1+\theta)\mu^{(k)}.$$

Thus,

$$\|(S_{\hat{B}}^{(k)})^{1/2}(X_{\hat{B}}^{(k)})^{1/2}\|_\infty = \max_{i\in\hat{B}}\sqrt{x_i s_i} \leq \max_{i=1}^n \sqrt{x_i s_i} \leq \sqrt{(1+\theta)\mu^{(k)}}.$$

Now we can conclude that

$$\|r'^{(k)}\|_\infty = \|S_{\hat{B}}^{(k)}v_{\hat{B}}^{(k)}\|_\infty = \|S_{\hat{B}}^{(k)}D_{\hat{B}}^{(k)}\hat{r}^{(k)}\|_\infty \leq \|(S_{\hat{B}}^{(k)})^{1/2}(X_{\hat{B}}^{(k)})^{1/2}\|_\infty\|\hat{r}^{(k)}\|_\infty \leq \sqrt{(1+\theta)\mu^{(k)}}\|\hat{r}^{(k)}\|_\infty \leq \eta\mu^{(k)}.$$

□

In the next subsection, we analyze the complexity of QLSA+QTA to solve the (MNES).

3.1 Solving MNES by QLSA+QTA

Before discussing QLSAs, we should mention that the $|z\rangle$ notation represents the quantum state corresponding to the unit classical vector z. We denote the basis state $|i\rangle$, which is a column vector with dimension p, with value one in coordinate i and zero in other coordinates [10].

The first QSLA was proposed by Harrow, Hassidim and Loyd [22] and known as HHL algorithm. It takes as input a sparse, Hermitian matrix M, and prepares a state $|z\rangle = |M^{-1}\sigma\rangle$ that is proportional to the solution of the linear system $Mz = \sigma$. The complexity of the HHL algorithm is $\tilde{O}_d(\tau^2\kappa_M^2/\epsilon)$, where d is the dimension of the problem, τ is the maximum number of non-zeros found in any row of M, and ϵ is the target bound on the error. This complexity bound shows a speed-up with respect to. dimension, although it depends on an upper bound for the condition number κ_M of the coefficient matrix. Generally, the HHL algorithm starts from quantum state $|\sigma\rangle$ and performs an eigendecomposition of A through the following general steps:

(1) Represent σ as a quantum state $|\sigma\rangle = \sum_{i=1}^{d} \sigma_i |i\rangle$.
(2) Apply the conditional Hamiltonian evolution e^{iMt} to $|\sigma\rangle$ for a superposition of different times t. With the phase estimation algorithm, we can decompose $|\sigma\rangle$ in the eigenbasis of M and find the corresponding eigenvalues λ_j. After this stage, the state of the system is close to $\sum_{j=1}^{d} \beta_j |u_j\rangle |\lambda_j\rangle$, where u_j is the eigenbasis of M, and $|\sigma\rangle = \sum_{j=1}^{d} \beta_j |u_j\rangle$.
(3) By uncomputing the $|\lambda_j\rangle$ register, we can invert the eigenvalues and prepare the solution state

$$\sum_{j=1}^{d} \beta_j \lambda_j^{-1} |u_j\rangle = |z\rangle.$$

Following a number of improvements to HHL algorithm [2, 10, 41, 43], the current state-of-the-art QLSA is attributed to Charkraborty et al. [9], who use variable-time *amplitude estimation* and so-called *block-encoded* matrices, while HHL algorithm uses the *sparse-encoding* model [22].

The block-encoding model was formalized in [27], and it assumes that one has access to unitaries that store the coefficient matrix in their top-left block:

$$U = \begin{pmatrix} M/\psi & \cdot \\ \cdot & \cdot \end{pmatrix},$$

where $\psi \geq \|M\|$ is a normalization factor chosen to ensure that U has operator norm at most 1. Assuming access to QRAM, the QLSA of [9] has $\tilde{O}_{d,\kappa_M,\frac{1}{\epsilon}}(\kappa_M\psi)$ complexity, where $\psi = \|M\|_F$. For the details of the QLSA and its complexity analysis, the reader is refered to [4, 9, 17].

If the linear system of equations has the normal equation form, i.e., $EE^Tz = E\sigma$ for a rectangular matrix E, one can use the block encoding of E and apply singular value inversion of E on σ and prepare solution state $|E^\star\sigma\rangle$, where $E^\star = (EE^T)^{-1}E$ is the pseudoinverse (or Moore–Penrose inverse) of E. In this framework, we can avoid the matrix-matrix product aiming the total query complexity is $\tilde{O}_{d,\kappa_E,\frac{1}{\epsilon}}(\kappa_E\|E\|_F)$ [9]. As we see, by singular value inversion, we can improve the complexity of QLSA for normal equations from $\kappa_M = O(\kappa_E^2)$ to κ_E and from $\|M\|_F$ to $\|E\|_F$. As the proposed (MNES) has this favorable structure, we can use this framework to get better complexity than previous QIPMs.

While QLSAs provide a quantum state proportional to the solution, it is not possible to extract the classical solution by a single measurement. Quantum Tomography Algorithms (QTAs) are needed to extract the classical solution. There are several papers improving QTAs, and the best QTA [39] has $O(\frac{d\varrho}{\epsilon})$ complexity, where ϱ is a bound for the norm of the solution. The direct use of the QLSA from [9] and the QTA by [39] costs

$\tilde{O}_{d,\kappa_M,\frac{1}{\epsilon}}\left(d\kappa_M^2\|M\|_F\frac{\|\sigma\|}{\epsilon}\right)$. Mohammadisiahroudi et al. [29] used an iterative refinement approach employing limited precision QLSA+QTA with $\tilde{O}_{d,\kappa_M,\frac{\|\sigma\|}{\epsilon}}\left(d\|M\|_F\kappa_M^2\right)$ complexity with $\tilde{O}_{\frac{\|\sigma\|}{\epsilon}}\left(d^2\right)$ classical arithmetic operations.

THEOREM 3.3. *Using the linear solver of [29], the* (MNES) *system can be built and solved to obtain a solution* $\tilde{z}^{(k)}$ *satisfying* $\|\hat{r}^{(k)}\| \leq \frac{\eta}{\sqrt{1+\theta}}\sqrt{\mu^{(k)}}$ *with at most* $\tilde{O}_{m,\kappa_E^{(k)},\frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}}\left(m(\kappa_E^{(k)})^2\|E^{(k)}\|_F\right)$ *complexity and* $\tilde{O}_{\frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}}(mn)$ *classical arithmetic operations, where* $E^{(k)} = (D_{\hat{B}}^{(k)})^{-1}\hat{A}D^{(k)}$, *and* $\kappa_E^{(k)}$ *is the condition number of* $E^{(k)}$.

PROOF. Building the (MNES) system in a classical computer requires matrix multiplications, which cost $O(m^2n)$ arithmetic operations. We can write (MNES) as

$$E^{(k)}(E^{(k)})^T\tilde{z}^{(k)} = \hat{\sigma}^{(k)} = E^{(k)}\left((X^{(k)})^{1/2}(S^{(k)})^{1/2}e - \beta\mu^{(k)}(X^{(k)})^{-1/2}(S^{(k)})^{-1/2}e\right),$$

where $E^{(k)} = (D_{\hat{B}}^{(k)})^{-1}\hat{A}D^{(k)}$. Calculating $E^{(k)}$ and $\hat{\sigma}^{(k)}$ requires just $O(mn)$ arithmetic operations. Then, we can use the quantum linear system solver of [29] with $\tilde{O}_{m,\kappa_E^{(k)},\frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}}\left(m(\kappa_E^{(k)})^2\|E^{(k)}\|_F\right)$ complexity and $\tilde{O}_{\frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}}(mn)$ classical arithmetic operations. □

We should remark that the error of the solution from QLSA+QTA can have several sources in practice. Part of the error is from the algorithmic errors of QLSA and QTA. Another part has a physical source of noise, especially in current NISQ devices. In the theoretical analysis of QLSAs, we address the algorithmic error and assume that we have fault-tolerant quantum computers on which the error correction procedure can handle physical noises. However, the proposed framework is designed very robust against errors, which may also handle some level of hardware noise in practice. This is further discussed in Section 6.

In the next section, we apply the proposed modification of NES to develop an IF-QIPM.

4 Inexact-Feasible Quantum Interior Point Method

We develop the IF-QIPM using the modified NES with the short-step version of IPMs.

As the IF-QIPM is a feasible IPM, it requires an initial feasible interior solution to start with. It is well-known in the literature [38] that this is not an issue as one can embed an LO problem in a homogeneous self-dual embedding formulation for which an all-one vector is a feasible and interior solution. We also apply IF-QIPM to the self-dual embedding formulation in practice.

In the next section, we prove the convergence of the IF-QIPM and analyze its complexity.

4.1 Convergence Analysis

To prove the convergence of IF-QIPM, we use the analysis of IF-QIPM in [32]. The only difference is the choice of the Newton system: In [32], the authors use OSS and in the current work we propose the use of MNES, however both compute $(\Delta x, \Delta y, \Delta s)$ such that

$$A\Delta x = 0,$$
$$A^T\Delta y + \Delta s = 0,$$
$$X\Delta s + S\Delta x = \beta\mu e - X^{(k)}s^{(k)} + r',$$

where $\|r'\| \leq \frac{\eta}{\sqrt{1+\theta}}\sqrt{\mu}$.

Next we provide relevant theory for our IF-QIPM in lemmas 4.1 and 4.2 and theorem 4.3. For the relevant proofs, we refer to lemmas 3.1 and 3.2 and theorem 2.6, respectively, in [32].

Algorithm 2 Short-step IF-QIPM using QLSA

1: Choose $\zeta > 0$, $\eta = 0.1$, $\theta = 0.7$ and $\beta = (1 - \frac{0.2}{\sqrt{n}})$.
2: $k \leftarrow 0$
3: Choose initial feasible interior solution $(x^0, y^0, s^0) \in \mathcal{N}(\theta)$
4: **while** $(x^{(k)}, y^{(k)}, s^{(k)}) \notin \mathcal{PD}_\zeta$ **do**
5:     $\mu^{(k)} \leftarrow \frac{(x^{(k)})^T s^{(k)}}{n}$
6:     $\tilde{z}^{(k)} \leftarrow$ Solve $\hat{M}^{(k)} \tilde{z}^{(k)} = \hat{\sigma}^{(k)}$ using IR+QLSA+QTA of [29] with $\|\hat{r}^{(k)}\| \leq \frac{\eta}{\sqrt{1+\theta}}\sqrt{\mu^{(k)}}$.
7:     $\Delta y^{(k)} \leftarrow ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T \tilde{z}^{(k)}$
8:     $v^{(k)} \leftarrow (v_{\hat{B}}^{(k)}, v_{\hat{N}}^{(k)}) = (D_{\hat{B}}^{(k)} \hat{r}^{(k)}, 0)$
9:     $\Delta s^{(k)} \leftarrow c - A^T y^{(k)} - s^{(k)} - A^T \Delta y^{(k)}$.
10:     $\Delta x^{(k)} \leftarrow \beta \mu^{(k)} (S^{(k)})^{-1} e - x^{(k)} - (D^{(k)})^2 \Delta s^{(k)} - v^{(k)}$.
11:     $(x^{(k+1)}, y^{(k+1)}, s^{(k+1)}) \leftarrow (x^{(k)}, y^{(k)}, s^{(k)}) + (\Delta x^{(k)}, \Delta y^{(k)}, \Delta s^{(k)})$
12:     $k \leftarrow k + 1$
13: **end while**
14: Return $(x^{(k)}, y^{(k)}, s^{(k)})$

LEMMA 4.1. *Let the step* $(\Delta x, \Delta y, \Delta s)$ *be calculated from* (MNES) *in each iteration of the IF-IPM. Then*

$$\Delta x^T \Delta s = 0,$$

$$(x + \Delta x)^T (s + \Delta s) \leq (\beta + \frac{\eta}{\sqrt{1+\theta}}) x^T s,$$

$$(x + \Delta x)^T (s + \Delta s) \geq (\beta - \frac{\eta}{\sqrt{1+\theta}}) x^T s.$$

Now, we can show that the iterates of IF-QIPM remain in the neighborhood of the central path in Lemma 4.2, by using results of Lemma 4.1.

LEMMA 4.2. *Let* $(x^0, s^0, y^0) \in \mathcal{N}(\theta)$ *for a given* $\theta \in [0, 1)$, *then* $(x^{(k)}, s^{(k)}, y^{(k)}) \in \mathcal{N}(\theta)$ *for any* $k \in \mathbb{N}$.

Based on Lemma 4.2, IF-IPM remains in the neighborhood of the central path, and it converges to the optimal solution if $\mu^{(k)}$ converges to zero. In Theorem 4.3, we prove that the algorithm reaches ζ-optimal solution after $O(\sqrt{n} \log(\frac{\mu_0}{\zeta}))$ iteration.

THEOREM 4.3. *The sequence* $\mu^{(k)}$ *converges to zero linearly, and we have* $\mu^{(k)} \leq \zeta$ *after* $O(\sqrt{n} \log(\frac{\mu_0}{\zeta}))$ *iterations.*

This demonstrates that the IF-IPM achieves the best-known iteration complexity, and the proof holds for any values satisfying the following two conditions.

$$\beta \leq (1 - \frac{\eta + 0.01}{\sqrt{n}}), \tag{5}$$

$$\frac{\theta^2 + n(1-\beta)^2 + \eta^2}{2^{3/2}(1-\theta)} + \eta \leq \theta(\beta - \frac{\eta}{\sqrt{n}}). \tag{6}$$

It is not hard to check that $\theta = 0.7$ and $\eta = 0.1$ satisfy these conditions.

An exact solution can be calculated by rounding [44], provided that we terminate with $\mu^{(k)} \leq O(2^{-L})$. Accordingly, the IF-IPM may require $O(\sqrt{n}L)$ to determine an exact optimal solution; for more details see [44, Chapter 3].

4.2 Complexity

The next theorem characterizes the total time complexity of the proposed IF-QIPM for finding a ζ optimal solution of an LO problem.

THEOREM 4.4. *The proposed IF-QIPM of Algorithm 2 determines a ζ-optimal solution using at most*

$$\tilde{O}_{m,\kappa_{\hat{A}},\|\hat{A}\|,\|\hat{b}\|,\mu^0,\frac{\omega}{\zeta}} \left(\sqrt{n} m \kappa_{\hat{A}}^2 \|\hat{A}\|_F \frac{\omega^5}{\zeta^5} \right)$$

queries to the QRAM and $\tilde{O}_{\mu^0,\frac{1}{\zeta}} \left(n^{1.5} m \right)$ classical arithmetic operations.

PROOF. The complexity analysis of the different parts of IF-QIPM 2 is outlined as follows:

- After $O(\sqrt{n} \log(\frac{\mu^0}{\zeta}))$ the IF-QIPM obtains a ζ-optimal solution.
- In Theorem 4.2 of [31], the norm and condition number bounds of MNES are derived as

$$\|\sigma^{(k)}\| = O\left(\frac{\|\hat{A}\| + \|\hat{b}\|}{\zeta} \right), \quad \|E^{(k)}\|_F = O\left(\frac{\omega}{\zeta} \|\hat{A}\|_F \right), \quad \kappa_E^{(k)} = O\left(\frac{\omega^2}{\zeta^2} \kappa_{\hat{A}} \right),$$

  where ω is a bound on $\|x^*, s^*\|_\infty$ for all $(x^*, y^*, s^*) \in \mathcal{PD}^*$.
- Applying Theorem 3.3, the complexity of quantum subroutine to solve the MNES is

$$\tilde{O}_{m,\kappa_{\hat{A}},\|\hat{A}\|,\|\hat{b}\|,\frac{\omega}{\zeta}} \left(m \kappa_{\hat{A}}^2 \|\hat{A}\|_F \frac{\omega^5}{\zeta^5} \right).$$

- In each iteration of IF-QIPM, we need to build $E^{(k)}$ classically and load to QRAM with $O(mn)$ complexity. Also, some classical matrix products happen with $O(mn)$ cost. Thus, the classical cost per iteration is $O(mn)$.
- Thus, in the worst case the IF-QIPM requires

$$\tilde{O}_{m,\kappa_{\hat{A}},\|\hat{A}\|,\|\hat{b}\|,\mu^0,\frac{\omega}{\zeta}} \left(\sqrt{n} m \kappa_{\hat{A}}^2 \|\hat{A}\|_F \frac{\omega^5}{\zeta^5} \right)$$

  accesses to the QRAM and $\tilde{O}_{\mu^0,\frac{1}{\zeta}} \left(n^{1.5} m \right)$ classical arithmetic operations.

□

4.3 Improving the error dependence of the IF-QIPM

To get an exact optimal solution, the time complexity contains the exponential term 2^L, due to the polynomial dependence on $\frac{1}{\zeta}$. To address this problem, we can fix $\zeta = 10^{-2}$ and improve the precision by iterative refinement in $O(L)$ iterations [31]. The first iterative refinement method for linear optimization is proposed by Gleixner et al. [19]. Here, we adopt the iterative refinement of [32], which is designed specifically for IF-QIPM as Algorithm 3.

First, we scale the LOP by

$$\rho \leftarrow \max \{\|A\|_F, \|b\|, \|c\|\}$$

and then we solve the scaled problem $(\widetilde{A}, \tilde{b}, \tilde{c}) \leftarrow \frac{1}{\rho}(A, b, c)$ instead of the original problem. Let (x^*, y^*, s^*) be $\frac{\zeta}{\rho}$-optimal solution for the scaled LOP $(\widetilde{A}, \tilde{b}, \tilde{c})$. It is easy to verify that $(x^*, y^*, \rho s^*)$ is an ζ-optimal solution for the original problem. By this scaling, we can improve the complexity from polynomial to polylogarithmic dependence on the norm of input data.

In the IR method, we initially solve the LO problem with limited precision $\hat{\zeta}$ and then iteratively improve the precision by solving the following refining problem with the limited precision $\hat{\zeta}$,

$$
\begin{array}{ll}
\min_{\hat{x}} \ \nabla s^T \hat{x}, & \max_{\hat{y},\hat{s}} \ -\nabla x^T \hat{s}, \\
\text{s.t.} \ \tilde{A}\hat{x} = 0, & \text{s.t.} \ \tilde{A}^T \hat{y} + \hat{s} = \nabla s, \\
\hat{x} \geq -\nabla x, & \hat{s} \geq 0,
\end{array}
\tag{7}
$$

where $\nabla > 1$ is a scaling factor. By changing variables, one can easily reformulate this problem to a standard LOP. The next lemma demonstrates the core idea of the iterative refinement technique.

LEMMA 4.5. *If $(\hat{x}, \hat{y}, \hat{s})$ is a $\hat{\zeta}$-optimal solution for refining problem* (7), *then (x^r, y^r, s^r) is a $\frac{\hat{\zeta}}{\nabla^2}$-optimal solution for LO problem (A, b, c), where $x^r = x + \frac{1}{\nabla}\hat{x}$, $y^r = y + \frac{1}{\nabla}\hat{y}$, and $s^r = c - A^T y^r$.*

The proof for Lemma 4.5 is analogous to the one of Lemma 6.1 in [32]. Based on this result, we develop the IR method as outlined in Algorithm 3.

Algorithm 3 IR-IF-QIPM

Require: $\left(A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, c \in \mathbb{R}^n, \zeta < \hat{\zeta} < 1\right)$
1: $\rho \leftarrow \max\{\|A\|_F, \|b\|, \|c\|\}$
2: $(\widetilde{A}, \tilde{b}, \tilde{c}) \leftarrow \frac{1}{\rho}(A, b, c)$
3: $k \leftarrow 1$ and $\nabla_0 \leftarrow 1$
4: $(x^{(1)}, y^{(1)}, s^{(1)}) \leftarrow$ **solve** $(\widetilde{A}, \tilde{b}, \tilde{c})$ using IF-QIPM of Algorithm 2 with $\hat{\zeta}$ precision
5: **while** $(x^{(k)}, y^{(k)}, s^{(k)}) \notin \mathcal{PD}_{\frac{\zeta}{\rho}}$ **do**
6: $\quad \nabla^{(k)} \leftarrow \frac{1}{\left(x^{(k)}\right)^T s^{(k)}}$
7: $\quad (\hat{x}^{(k)}, \hat{y}^{(k)}, \hat{s}^{(k)}) \leftarrow$ **solve** $(\tilde{A}, 0, \nabla^{(k)} s^{(k)})$ using IF-QIPM of Algorithm 2 with $\hat{\zeta}$ precision
8: $\quad x^{(k+1)} \leftarrow x^{(k)} + \frac{1}{\nabla^{(k)}}\hat{x}^{(k)}$ and $y^{(k+1)} \leftarrow y^{(k)} + \frac{1}{\nabla^{(k)}}\hat{y}^{(k)}$
9: $\quad k \leftarrow k + 1$
10: **end while**
11: **return** $(x^{(k)}, y^{(k)}, \rho s^{(k)})$

THEOREM 4.6. *The proposed IR-IF-QIPM of Algorithm 3 produces a ζ-optimal solution with at most $O\left(\frac{\log(\rho/\zeta)}{\log(1/\hat{\zeta})}\right)$ inquiry to IF-QIPM with precision $\hat{\zeta}$.*

THEOREM 4.7. *The total time complexity of finding an exact optimal solution with the IR-IF-QIPM is*

$$
\tilde{O}_{m,\kappa_{\hat{A}},\|\hat{A}\|,\|\hat{b}\|,\mu^0}\left(\sqrt{n}m\kappa_{\hat{A}}^2\omega^5 L\right)
$$

with $\tilde{O}_{\mu^0}\left(n^{1.5}mL\right)$ classical arithmetic operation.

The proof is similar to the proof of [32, Theorem 6.2].

In the next section, we investigate how preconditioning can help to mitigate the effect of the condition number with respect to $\kappa_{\hat{A}}$ and ω.

5 Iteratively Refined IF-QIPM using preconditioned NES

To mitigate the impact of the condition number, we need to analyze how the matrices $M^{(k)}$ and $E^{(k)}$ evolve through the iterations. As in Theorem 6.8 of [44] or Lemma I.42 of [38], considering the optimal partition $\mathcal{B}$ and $\mathcal{N}$, we have

$$\frac{x_i^{(k)}}{s_i^{(k)}} = \mathcal{O}\left(\frac{\hat{C}}{\mu^{(k)}}\right) \to \infty \text{ for } i \in \mathcal{B} \qquad \text{and} \qquad \frac{x_i^{(k)}}{s_i^{(k)}} = \mathcal{O}\left(\frac{\mu^{(k)}}{\hat{C}}\right) \to 0 \text{ for } i \in \mathcal{N}, \tag{8}$$

where $\hat{C}$ is a constant dependent on the LO problem's parameters. For a more detailed analysis, see pages 121-124 of [44]. To analyze the condition number of NES, we have

$$M^{(k)} = A(D^{(k)})^2 A^T = A_\mathcal{B}(D_\mathcal{B}^{(k)})^2 A_\mathcal{B}^T + A_\mathcal{N}(D_\mathcal{N}^{(k)})^2 A_\mathcal{N}^T.$$

As the sequence of iterates converges to the optimal set, it is easy to see that $A_\mathcal{N}(D_\mathcal{N}^{(k)})^2 A_\mathcal{N}^T \to 0$. Thus, the dominant component is the $A_\mathcal{B}(D_\mathcal{B}^{(k)})^2 A_\mathcal{B}^T$ term. If $A_\mathcal{B}$ includes a basis of A, i.e. $\text{rank}(A_\mathcal{B}) = m$, then the condition number of $M^{(k)}$ will converge to a constant depending on $\frac{\max_{i \in \mathcal{B}} x_i^2}{\min_{i \in \mathcal{B}} x_i^2}$. This implies that when the problem is primal nondegenerate, the condition number will converge to a constant, though that constant may be as big as $\mathcal{O}(2^{2L})$ in the worst case. If $A_\mathcal{B}$ has a rank less than m, then the condition number of $M^{(k)}$ goes to infinity with the rate of $\frac{1}{\mu^2}$, which can be addressed by iterative refinement. In the proposed IF-QIPM, we initially choose basis $\bar{B}$. If in each iteration of IF-QIPM, we choose basis $\hat{B}^{(k)}$ represents the m largest $\frac{x_i^{(k)}}{s_i^{(k)}}$, then by modifying MNES, we can precondition it too. We refer the readers to [34] for the algorithm for determining basis $\hat{B}^{(k)}$. Suppose that the LO problem is non-degenerate. As the trajectory generated by the IF-QIPM converges to the optimal solution, we have

$$\hat{B}^{(k)} \to B$$
$$A_\mathcal{N}(D_\mathcal{N}^{(k)})^2 A_\mathcal{N}^T \to 0$$
$$M^{(k)} \to A_B(D_\mathcal{B}^{(k)})^2 A_\mathcal{B}^T$$
$$\hat{M}^{(k)} = (D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1} M^{(k)} ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T \to I.$$

Thus, $(D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1}$ is a precondition for $M^{(k)}$. When the LO problem is degenerate, which is the more general setting, Theorem 2.2.3 of [37] asserts that the condition number of $\hat{M}^{(k)}$ is bounded by $(\bar{\chi})^2$ where

$$\bar{\chi} = \max\left\{\|A_B^{-1}A\|_F : A_B \text{ is a basis of } A\right\}. \tag{9}$$

Furthermore, based on Lemma 2.2.2 of [37], we have $\|\hat{M}^{(k)}\|_F = \mathcal{O}(\bar{\chi})$.

To utilize this preconditioning method within our IF-QIPM framework, we adopt the following procedure.

Step 1. Choose basis $\hat{B}^{(k)}$ as indices of m largest, $\frac{x_i^{(k)}}{s_i^{(k)}}$ where $A_{\hat{B}^{(k)}}$ are linearly independent

Step 2. Build block-encoding of $(D_{\hat{B}}^{(k)})^{-1}$ and $A_{\hat{B}}$

Step 3. Calculate $A_{\hat{B}}^{-1}$ on the quantum computer

Step 4. Build $\hat{M}^{(k)} = (D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1} M^{(k)} ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T$ on the quantum computer

Step 5. Find $\tilde{z}^{(k)}$ such that $\hat{M}^{(k)} \tilde{z}^{(k)} = \hat{\sigma}^{(k)} + \hat{r}^{(k)}$ and $\|\hat{r}^{(k)}\| \leq \frac{\eta}{\sqrt{1+\theta}} \sqrt{\mu^{(k)}}$ on the quantum computer

Step 6. Calculate $\widetilde{\Delta y}^{(k)} = ((D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1})^T \tilde{z}^{(k)}$.

Step 7. Calculate $v^{(k)} = (v_{\hat{B}}^{(k)}, v_{\hat{N}}^{(k)}) = (D_{\hat{B}}^{(k)} \hat{r}^{(k)}, 0)$.
Step 8. Calculate $\widetilde{\Delta s}^{(k)} = c - A^T y - s^{(k)} - A^T \widetilde{\Delta y}^{(k)}$.
Step 9. Calculate $\widetilde{\Delta x}^{(k)} = \beta \mu^{(k)} (S^{(k)})^{-1} e - x^{(k)} - (D^{(k)})^2 \widetilde{\Delta s}^{(k)} - v^{(k)}$.

As we can see, the only change in the Newton step calculation is that the basis $\hat{B}^{(k)}$ is changing at each iteration, thus we need to calculate $A_{\hat{B}}^{-1}$ at each iteration. One can verify that the Newton steps calculated with the above steps admit properties of Lemma 3.2. It is straightforward to provide a convergence proof of an IF-QIPM that uses this procedure, since it produces feasible-inexact iterates satisfying $\|\hat{r}^{(k)}\| \leq \frac{\eta}{\sqrt{1+\theta}} \sqrt{\mu^{(k)}}$. Thus, the iteration complexity of IF-QIPM using preconditioned NES is $O(\sqrt{n} \log(\frac{\mu^0}{\zeta}))$. However, the cost-per-iteration is different as classically calculating $A_{\hat{B}}^{-1}$ requires $O(nm^2)$ arithmetic operation in the worst case. If only a few columns in $\hat{B}^{(k)}$ are changing from one iteration to another, one can update $A_{\hat{B}}^{-1}$ by Sherman-Morrison-Woodbury formula which is computationally cheaper than the full inversion. Here, we apply the inverse of the block-encoding of $A_{\hat{B}}^{-1}$ and build block-encoding of $\hat{E}^{(k)} = D_{\hat{B}}^{(k)})^{-1} A_{\hat{B}}^{-1} E^{(k)}$ on a quantum machine. As the complexity of applying the inverse of a block-encoding has a logarithmic dependence on $\frac{1}{\zeta}$, we can do that calculation with high-precision, i.e. $\zeta = 2^{-O(L)}$. Thus, errors in those steps are negligible, and analysis of IF-QIPM in Section 4 is valid.

In order to estimate the asymptotic scaling of the overall complexity, we begin by analyzing the cost of solving the Preconditioned NES at each iteration. As discussed in Section 3.1, the preconditioned NES is still a normal equation system and we need to build the block-encoding of $\hat{E}^{(k)}$, then we can use singular value transformation of [9] and find the solution of the preconditioned NES by QLSA+QTA with $\tilde{O}_{m, \kappa_{\hat{E}}^{(k)}, \frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}} \left(m(\kappa_{\hat{E}}^{(k)})^2 \|\hat{E}^{(k)}\|_F \right)$ quantum complexity and $\tilde{O}_{\frac{\|\hat{\sigma}^{(k)}\|}{\mu^{(k)}}} (mn)$ classical arithmetic operations. As in IR, we scale the LO problem initially, so in our analysis here we assume $\|A\|_F \leq 1$.

PROPOSITION 5.1. *Suppose A and $A_{\hat{B}}$ are stored in a QRAM data structure. Then, one can prepare a block-encoding of $\hat{E}^{(k)}$ accesses to the QRAM and $\widetilde{O}(n)$ arithmetic operations.*

PROOF. First, observe that we always have classical access to $x^{(k)}$ and $s^{(k)}$. We can therefore store the nonzero entries of the matrices $D^{(k)}$ and $\left(D_{\hat{B}}^{(k)} \right)^{-1}$ in QRAM using $\widetilde{O}_n(n)$ classical operations. From here, applying [17, Lemma 50] asserts that $\widetilde{O}_{n, \frac{1}{\xi_D}}(1)$ accesses to QRAM suffices to construct an $(\alpha_D, \log(n) + 2, \xi_D)$-block-encoding of $D^{(k)}$ and an $\left(\alpha_{D_{\hat{B}}}, \log(n) + 2, \xi_D \right)$-block-encoding of $(D_{\hat{B}}^{(k)})^{-1}$. As $(D_{\hat{B}}^{(k)}$ and $(D_{\hat{B}}^{(k)})^{-1}$ are diagonal matrices, they can be efficiently encoded on QRAM and the normalization parameters $\alpha_{D^{-1}}$ and α_D have upper bound $O(\frac{\omega}{\epsilon})$ as analyzed in [31]. Likewise, with A and $A_{\hat{B}}$ stored in QRAM, invoking [17, Lemma 50] we can construct a $(\|A\|_F, \log(n) + 2, \xi_A)$-block-encoding of A and a $\left(\|A_{\hat{B}}\|_F, \log(n) + 2, \xi_A \right)$-block-encoding of $A_{\hat{B}}$, using $\widetilde{O}_{m, n, \frac{1}{\xi_A}}(1)$ accesses to the QRAM.

Having prepared block-encodings of A and $D^{(k)}$, we can take their product prepare an $(\alpha_D \|A\|_F, O(\log(n)), \xi_M)$-block-encoding of $M^{(k)}$ as

$$U_A U_{D^2} U_{A^\top} = U_M.$$

Similarly, having prepared a $\left(\|A_{\hat{B}}\|_F, \log(n) + 2, \xi_A \right)$-block-encoding of $A_{\hat{B}}$ for $A_{\hat{B}}$, applying [16, Corollary 3.4.13], we can prepare a $\left(\|A_{\hat{B}}\|_F \kappa_{A_{\hat{B}}}, \log(n) + 2, \xi_A \right)$-block-encoding of $\left(A_{\hat{B}} \right)^{-1}$. From here, applying [9, Lemma 4], we can

take the product of block-encodings of $\left(A_{\hat{B}}\right)^{-1}$ and $(D_{\hat{B}}^{(k)})^{-1}$, which yields an $\left(\alpha_{D^{-1}}\kappa_{A_{\hat{B}}}\left\|A_{\hat{B}}\right\|_F, O\left(\log(n)\right), \xi_P\right)$-block-encoding of P.

Observe that we have prepared block-encodings U_E and U_P for $E^{(k)}$ and P such that $U_P U_E$ corresponds to a $\left(\alpha_{D^{-1}}\alpha_D\kappa_{A_{\hat{B}}}\left\|A\right\|_F\left\|A_{\hat{B}}\right\|_F, O\left(\log(n)\right), \xi_P\right)$-block-encoding of P, since it is defined as the product of block-encodings [9, Lemma 4]. The complexity result follows from observing that constructing the unitary $U_{\hat{E}}$ has a gate cost of $\widetilde{O}_{m,n,\frac{1}{\epsilon}}(1)$, where we have properly set the parameters ξ_A and ξ_D, such that $U_{\hat{M}}$ implements $\hat{M}$ up to error ϵ. We also needed $\widetilde{O}_n(n)$ classical operations to create the QRAM data structures for D^{-1} and D. The proof is complete. □

COROLLARY 5.2. *Suppose A, $A_{\hat{B}}$, $x^{(k)}$, $s^{(k)}$ and $\hat{\sigma}^{(k)}$ are stored in a QRAM data structure and define*

$$\alpha := \alpha_{D^{-1}}\alpha_D.$$

Then, one can obtain an $\frac{\eta}{\sqrt{1+\theta}}\sqrt{\mu^{(k)}}$-precise solution (in ℓ_∞-norm) to the linear system

$$\hat{M}^{(k)}\tilde{z}^{(k)} = \hat{\sigma}^{(k)},$$

using at most

$$\widetilde{O}_{m,n,\frac{1}{\epsilon}}\left(m\alpha\kappa_{\hat{E}}^2\right)$$

QRAM accesses and $O\left(mn\right)$ arithmetic operations.

PROOF. Using the linear systems algorithm from [29] with the subnormalization factor $\alpha_{D^{-1}}\alpha_{D^2}^2\kappa_{A_{\hat{B}}}^2\left\|A\right\|_F^2\left\|A_{\hat{B}}\right\|_F^2$ and condition number $\kappa_{\hat{M}}$ gives the result. □

For both $\alpha_{D^{-1}}$ and α_D we can use upper bound $O(\frac{\omega}{\epsilon})$. Thus $\alpha = O(\frac{\omega^2}{\epsilon^2})$. We are now in a position to state the complexity of the Iteratively Refined IF-QIPM using preconditioned NES.

THEOREM 5.3. *Suppose that the LO problem data A, b, c is stored in QRAM. Then, the Iteratively Refined IF-QIPM using preconditioned NES obtains an ϵ-precise solution to the primal-dual LO pair $(P) - (D)$ using at most*

$$\widetilde{O}_{n,\|A\|,\|b\|,\kappa_A,\frac{\mu^0}{\epsilon}}\left(\sqrt{n}m\bar{\chi}^2\omega^2\right)$$

QRAM accesses and $\widetilde{O}_{\|A\|,\|b\|,\frac{1}{\epsilon}}\left(mn^{1.5}\right)$ arithmetic operations.

PROOF. The result follows upon adjusting the proof of Theorem 4.4 to account for the result in Corollary 5.2. □

Based on the analysis in [37], we have $\alpha = O(\omega^2)$ and $\kappa_{\hat{E}} = O(\bar{\chi})$. Thus, we can simplify the quantum complexity to

$$\widetilde{O}_{n,\|A\|,\|b\|,\kappa_A,\frac{\mu^0}{\epsilon}}\left(\sqrt{n}m\bar{\chi}^2\omega^2\right)$$

with $\widetilde{O}_{\|A\|,\|b\|,\frac{1}{\epsilon}}\left(mn^{1.5}\right)$ arithmetic operations.

6 Numerical Experiments

In this section, we present a series of numerical experiments aimed at elucidating the behavior of various QIPMs. Additionally, we compare the impact of employing iterative refinement versus preconditioning techniques. It is important to note that our analysis presupposes access to QRAM; however, it is essential to highlight that physical QRAM infrastructure has yet to be realized. Likewise, QLSAs remain beyond the capabilities of existing quantum hardware.

Consequently, our experiments are conducted using the IBM Qiskit HHL simulator, and it is imperative to acknowledge that our numerical results cannot be extrapolated to gauge the performance of QIPMs and QLSAs on actual quantum hardware. Simulating quantum computers on classical computers is known to be exponentially time-consuming, which precludes any empirical time comparison between classical and quantum methodologies at this juncture. Notwithstanding, we are capable of simulating QLSAs for problems with limited numbers of variables and manageable condition numbers.

Our primary focus therefore centers on presenting numerical findings pertaining to QIPMs, and we refrain from presenting QLSA results in this paper. Interested readers are referred to [29] for comprehensive numerical experiments related to QLSAs. Each of the algorithms discussed in this paper have been implemented in Python and are readily accessible on our GitHub repository at https://github.com/QCOL-LU. Our Python package encompasses a versatile array of QIPMs designed to solve linear, semidefinite, and second-order cone optimization problems. To enhancing their versatility and efficacy, we have also incorporated iterative refinement techniques into both Quantum Linear System Algorithms (QLSAs) and QIPMs. Users are offered the flexibility to conduct experiments with QIPMs using either classical or quantum linear solvers, with the option to employ preconditioning.

For our experimental setup, we employ the LOP generators described in [30]. These generators have been demonstrated to produce randomly generated Linear Optimization Problems (LOPs) with predefined optimal and interior solutions, thereby facilitating the evaluation of IF-QIPMs. Furthermore, these generators offer users the flexibility to control various characteristics of the problems, including the condition number of the coefficient matrix—a critical parameter for assessing QIPMs' performance. Our numerical experiments were conducted on a workstation equipped with Dual Intel Xeon® CPU E5-2630 @ 2.20 GHz, featuring 20 cores and 64 GB of RAM.

To illustrate how the condition numbers of different Newton systems evolve in IPMs, Fig. 1 shows the condition number of different linear systems trend for four problems. As Fig. 1a shows, the condition number of FNS, OSS, and NES converge to a constant for nondegenerate LOPs with a well-conditioned matrix A. However, the condition number of the augmented system may go to infinity, even for nondegenerate well-conditioned problems, as approaching the unique optimal solution. For nondegenerate problems with ill-conditioned matrices, Fig. 1b, the condition number of NES, OSS, and FNS still converge to a constant which can be very large, like 10^{12}. If the LO problem is degenerate, Fig. 1c, the condition number of all Newton systems goes to infinity as approaching the optimal solution. However, the FNS and OSS have a better rate than the NES and AS. The worst case happens when the problem is degenerate and matrix A is ill-conditioned, Fig. 1d. In this case, the condition number of NES can be as large as 10^{20} for $\mu = 10^{-6}$. As these figures illustrate, the condition number of the Newton systems is affected by the condition number of matrix A and the degeneracy status of the problem. Generally, OSS has a better condition number than the NES. In the next figures, we show how iterative refinement and preconditioning can mitigate the condition number of Newton systems, especially for NES.

Fig. 2 shows the performance of different IF-QIPMs with respect to condition number to solve a nondegenerate problem with an ill-conditioned matrix A. As we can see, preconditioned NES (PNES) has a significantly smaller condition number, even better than OSS. However, iterative refinement is not helping with the condition number since for this type of problem, early stopping the IF-QIPM and restarting it will not change the condition number as it is almost constant, dependent on the condition number of A. The performance of IF-QIPMs for solving a primal degenerate LOP with well-conditioned coefficient matrix A is depicted in Fig. 3. Although for PNES,

(a) A nondegenerate LO with $\kappa_A = 10$

(b) A nondegenerate LO with $\kappa_A = 10^6$

(c) A degenerate LOP with $\kappa_A = 10$
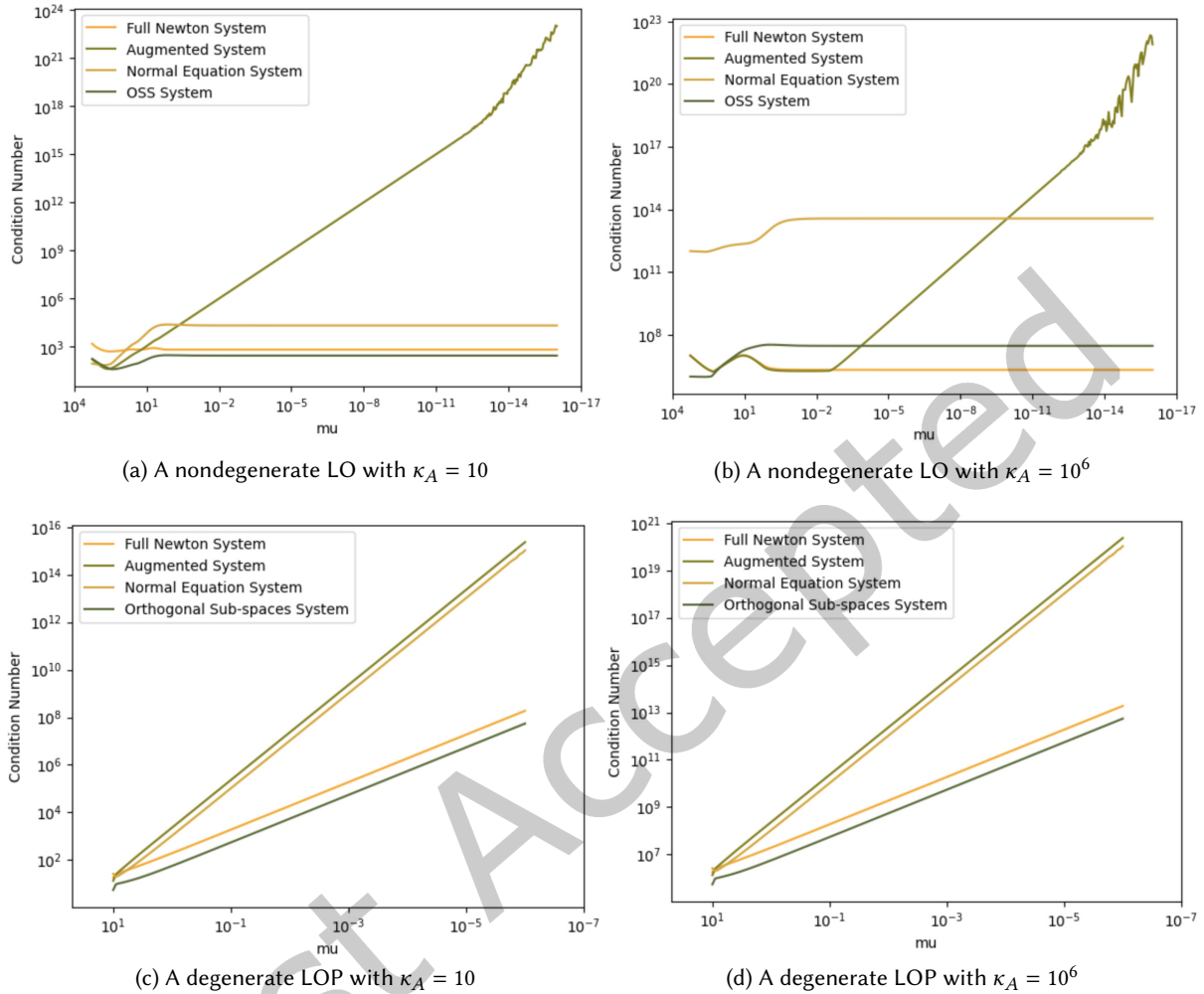
(d) A degenerate LOP with $\kappa_A = 10^6$

Fig. 1. Condition number trend of different Newton systems for different types of LOPs.

there is a theoretical uniform condition number bound, which is independent of the algorithm's iterates and depends only on the problem's input data. However, in this problem, as is the case for many other problems such as degenerate problems, this upper bound can be extremely large. The condition number can grow with a slightly lower rate but at the same rate as the one for MNES. On the other hand, for degenerate problems, iterative refinement can help with condition numbers. As Fig. 3b shows, in the iterative refinements steps, we stop IF-QIPMs early, when $\mu = 10^{-2}$ and so the condition number remains bounded. Then we restart the IF-QIPM for the refining problems, where the condition number is as low as the initial condition number. By IR, the condition number will not grow above an upper bound.

Fig. 4 shows how iterative refinement coupled with preconditioning can keep the condition number of the NES bounded during iterations of the QIPM for this challenging degenerate LOP with an ill-conditioned matrix. All in
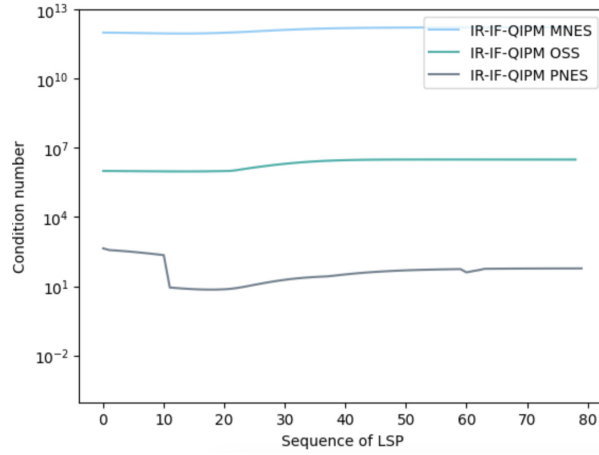
Fig. 2. Effect of preconditioning on the condition number sequence of linear systems arising in IR-IF-QIPM to solve a nondegenerate LOP with $\kappa_A = 10^6$

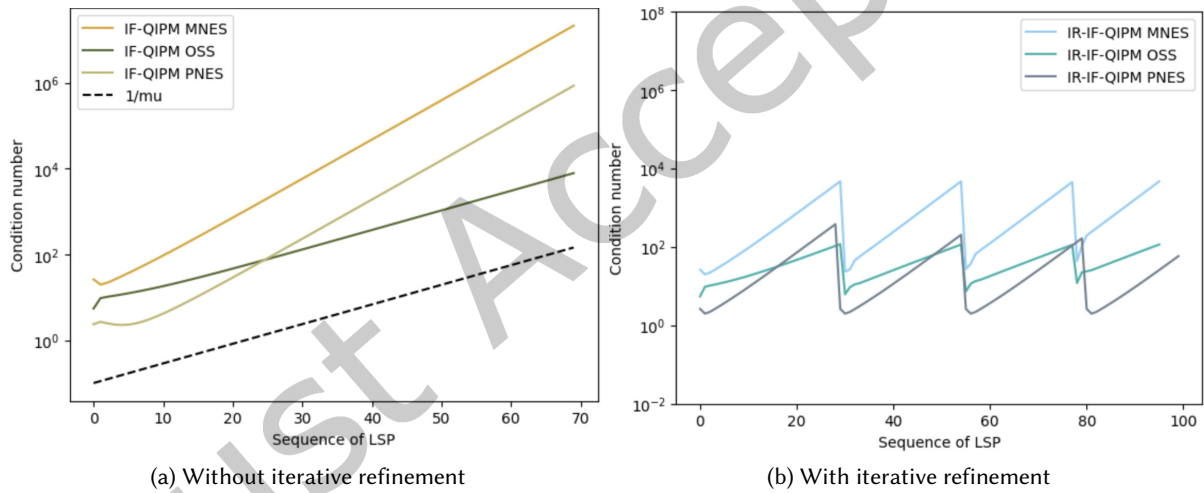(a) Without iterative refinement

(b) With iterative refinement

Fig. 3. Condition number sequence of linear systems arising in different IF-QIPM to solve a degenerate LOP with $\kappa_A = 10$

all, iterative refinement can mitigate the impact of degeneracy on the condition number of Newton systems. On the other hand, preconditioning is effective for addressing problems with an ill-conditioned matrix A.

We also solved 100 randomly generated problems with different IF-QIPMs using the IBM QISKIT simulator. Fig. 5 shows some statistics of solved problems. As we can see, with the OSS system we could not solve problems with more than 8 variables as the size of the linear systems that could be solved by the Qiskit simulator is limited to 16. In addition, the OSS has a nonsymmetric n-by-n coefficient matrix. However, with MNES, we could solve an LOP with a million variables and 16 constraints as the dimension of MNES is dependent on the number of constraints. These results show that the proposed IF-QIPM is more adaptable to near-term devices. In addition,

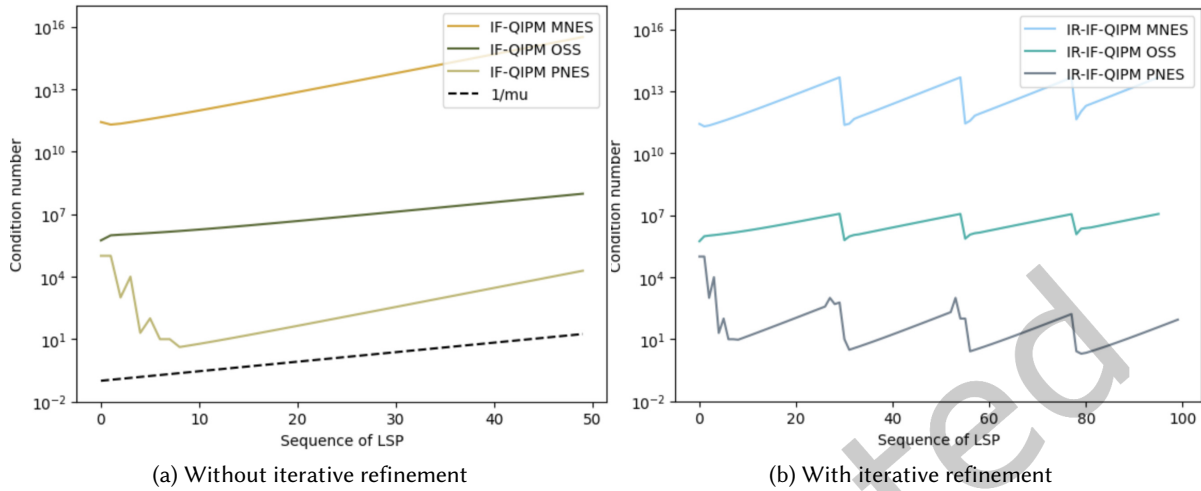(a) Without iterative refinement (b) With iterative refinement

Fig. 4. Condition number sequence of linear systems arising in different IF-QIPM to solve a degenerate LOP with $\kappa_A = 10^6$
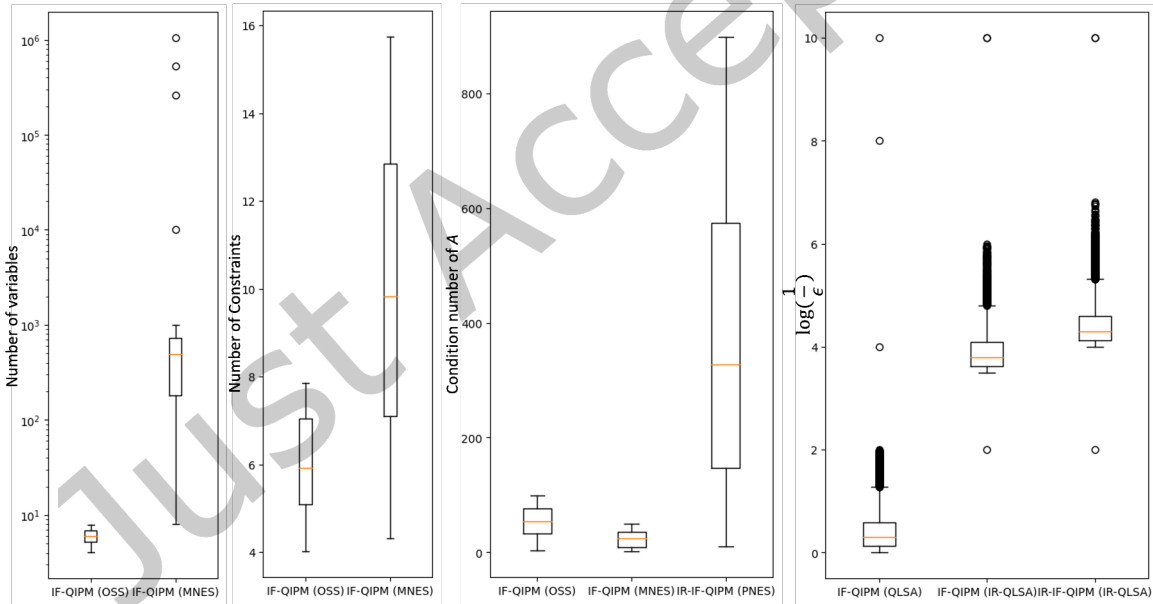

Fig. 5. Statistics of 100 randomly generated LOPs solved by IF-QIPM using QISKIT Simulator (max time =2 hrs)

we can see that the iterative refinement coupled with preconditioning enables the solution of the problem with a larger condition number. In addition, with both inner and outer iterative refinement, we could improve the precision from 10^{-1} to 10^{-4} on average.

Algorithm	System	Linear System Solver	Quantum Complexity	Classical Complexity	Bound for κ
IPM with Partial Updates [38]	NES	Low rank updates		$O(n^3L)$	
Feasible IPM [38]	NES	Cholesky		$O(n^{3.5}L)$	
II-IPM [35]	PNES	PCG		$O(n^5L\bar\chi^2)$	$\bar\chi^2$
II-QIPM [31]	NES	QLSA+QTA	$\tilde{O}_{n,\kappa_A,\omega}(n^4L\kappa_A^4\omega^{19}\|A\|)$	$\tilde{O}_\omega(n^4L)$	$O(\kappa_{\hat A}^2\omega^5)$
IF-QIPM [32]	OSS	QLSA+QTA	$\tilde{O}_{n,\kappa_A,\omega}(n^2L\kappa_A^2\omega^5\|A\|)$	$\tilde{O}_{\mu^0}(n^{2.5}L)$	$O(\kappa_{\hat A}\omega^2)$
The proposed IR-IF-IPM	MNES	CG		$\tilde{O}_{\mu^0}(n^{2.5}L\kappa_{\hat A}^2\omega^4)$	$O(\kappa_{\hat A}^2\omega^4)$
The proposed IR-IF-IPM	PNES	PCG		$\tilde{O}_{\mu^0}(n^{3.5}L\bar\chi^2)$	$\bar\chi^2$
The proposed IR-IF-QIPM	MNES	QLSA+QTA	$\tilde{O}_{n,\kappa_{\hat A},\|\hat A\|,\|\hat b\|,\mu^0}(\sqrt{n}mL\kappa_{\hat A}^2\omega^5)$	$\tilde{O}_{\mu^0}(n^{2.5}L)$	$O(\kappa_{\hat A}^2\omega^4)$
The proposed IR-IF-QIPM	PNES	QLSA+QTA	$\tilde{O}_{n,\kappa_A,\|A\|,\|b\|,\mu^0}(\sqrt{n}mL\bar\chi^2\omega^2)$	$\tilde{O}_{\mu^0}(n^{2.5}L)$	$\bar\chi^2$

Table 2. Complexity of different IPMs for LO

7 Conclusion

In this paper, we propose an inexact-feasible quantum interior point method in which we solve a modified normal equation system with QLSA+QTA. In addition, we apply an iterative refinement and preconditioning to mitigate the effect of condition number on the complexity of QIPMs. Applications of these classical ideas lead to improvements of QIPMs as follows:

- By modifying the NES, in each iteration of the proposed IF-QIPM, we solve a linear system with m-by-m symmetric positive definite matrix, which is smaller than the OSS with n-by-n nonsymmetric matrix. In other words, the proposed IF-QIPM needs fewer Qubits and gates.
- We use an iterative refinement scheme coupled with preconditioning the NES that builds a uniform bound on the condition number and speeds up QIPMs with respect to. precision, and condition number.
- By preconditioning the NES in the quantum setting, we achieved speed up with respect to. the dimension compared to classical inexact approaches.

In Table 2, the complexities of some recent classical and quantum IPMs are provided. As we can see, IR-IF-QIPM with MNES achieves the best complexity of IR-IF-QIPM using OSS with slightly better dependence on $\|A\|$, due to using quantum solver of [29]. By switching to preconditioned NES, complexity improves with respect to dimension compared with classical preconditioned IPMs. It is natural that calculating preconditioner on a quantum machine will have better complexity with respect to dimension but the challenge is addressing normalization factors in block-encoding. Another difference is dependence on $\bar\phi$ instead of ω which is an upper bound for norm of optimal solution. It should be mentioned that both ω and $\bar\phi$ are constants depending on input data. However, for some problems, they can be extremely large. As numerical results demonstrate, one advantage of iterative refinement and preconditioning is bounding the condition number. Mostly iterative refinement avoids the growing condition number of the Newton system in degenerate problems, and preconditioning alleviates the impact of the condition number of matrix A. All in all, QIPMs have the potential to speed up the solution of LOPs with respect to dimension compared to classical but they are more dependent on condition number and norm of the coefficient matrix. In this paper, we explored some classical ideas like using a better formulation of Newton's system and using iterative refinement coupled with preconditioning to shorten this gap. Table 2 demonstrates that the proposed IR-IF-QIPM using modified NES or preconditioned NES outperforms previous quantum and classical IPMs with respect to worst-case complexity.

Using MNES also enables regularizing the Newton system. It is worth exploring the regularization in the quantum setting to address the impact of condition number on QIPMs. In addition, the proposed IR-IF-QIPM with preconditioned NES can be generalized to other conic problems such as Semi-definite optimization where the size of Newton systems may grow quadratically for large-scale problems.

Acknowledgments

References

[1] Ghussoun Al-Jeiroudi and Jacek Gondzio. 2009. Convergence analysis of the inexact infeasible interior-point method for linear optimization. *Journal of Optimization Theory and Applications* 141, 2 (2009), 231–247. https://doi.org/10.1007/s10957-008-9500-5

[2] Andris Ambainis. 2012. Variable time amplitude amplification and quantum algorithms for linear algebra problems. In *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*, Vol. 14. LIPIcs, 636–647.

[3] Simon Apers and Sander Gribling. 2023. Quantum speedups for linear programming via interior point methods. *arXiv preprint arXiv:2311.03215* (2023).

[4] Brandon Augustino, Giacomo Nannicini, Tamás Terlaky, and Luis F. Zuluaga. 2023. A Quantum Interior Point Method for Semidefinite Optimization Problems. *Quantum* 7 (2023), 1110.

[5] Brandon Augustino, Tamás Terlaky, Mohammadhossein Mohammadisiahroudi, and Luis F Zuluaga. 2021. *An inexact-feasible quantum interior point method for second-order cone optimization.* Technical Report 21T-009. ISE, Lehigh University.

[6] Venansius Baryamureeba and Trond Steihaug. 2006. On the convergence of an inexact primal-dual interior point method for linear programming. In *Large-Scale Scientific Computing*, Ivan Lirkov, Svetozar Margenov, and Jerzy Waśniewski (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 629–637.

[7] Stefania Bellavia. 1998. Inexact interior-point method. *Journal of Optimization Theory and Applications* 96, 1 (1998), 109–121. https://doi.org/10.1023/A:1022663100715

[8] PAM Casares and MA Martin-Delgado. 2020. A quantum interior-point predictor–corrector algorithm for linear programming. *Journal of Physics A: Mathematical and Theoretical* 53, 44 (2020), 445305. https://doi.org/10.1088/1751-8121/abb439

[9] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. 2018. The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation. *arXiv preprint arXiv:1804.01973* (2018).

[10] Andrew M Childs, Robin Kothari, and Rolando D Somma. 2017. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM J. Comput.* 46, 6 (2017), 1920–1950.

[11] Michael B. Cohen, Yin Tat Lee, and Zhao Song. 2021. Solving linear programs in the current matrix multiplication time. *J. ACM* 68, 1, Article 3 (jan 2021), 39 pages. https://doi.org/10.1145/3424305

[12] Fabio D'andreagiovanni and Ambros M Gleixner. 2016. Towards an accurate solution of wireless network design problems. In *Combinatorial Optimization: 4th International Symposium, ISCO 2016, Vietri sul Mare, Italy, May 16-18, 2016, Revised Selected Papers 4.* Springer, 135–147.

[13] Leon Eifler and Ambros Gleixner. 2023. A computational status update for exact rational mixed integer programming. *Mathematical Programming* 197, 2 (2023), 793–812.

[14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. 2014. A quantum approximate optimization algorithm. *arXiv preprint* (2014). https://arxiv.org/abs/1411.4028

[15] Roland W. Freund, Florian Jarre, and Shinji Mizuno. 1999. Convergence of a class of inexact interior-point algorithms for linear programs. *Mathematics of Operations Research* 24, 1 (1999), 50–71. https://doi.org/10.1287/moor.24.1.50

[16] András Gilyén. 2019. *Quantum singular value transformation & its algorithmic applications.* Ph. D. Dissertation. University of Amsterdam.

[17] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. 2019. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, Moses Charikar and Edith Cohen (Eds.). 193–204.

[18] Ambros Gleixner and Daniel E. Steffy. 2020. Linear programming using limited-precision oracles. *Mathematical Programming* 183, 1 (2020), 525–554.

[19] Ambros M. Gleixner, Daniel E. Steffy, and Kati Wolter. 2016. Iterative refinement for linear programming. *INFORMS Journal on Computing* 28, 3 (2016), 449–464.

[20] Gene H Golub and Charles F Van Loan. 2013. *Matrix Computations.* JHU Press, Baltimore, MD, USA.

[21] Jacek Gondzio. 2013. Convergence analysis of an inexact feasible interior point method for convex quadratic programming. *SIAM Journal on Optimization* 23, 3 (2013), 1510–1527.

[22] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. 2009. Quantum algorithm for linear systems of equations. *Physical Review Letters* 103, 15 (2009). https://doi.org/10.1103/PhysRevLett.103.150502

[23] Iordanis Kerenidis and Anupam Prakash. 2020. A quantum interior point method for LPs and SDPs. *ACM Transactions on Quantum Computing* 1, 1 (2020), 1–32. https://doi.org/10.1145/3406306

[24] Masakazu Kojima, Nimrod Megiddo, and Shinji Mizuno. 1993. A primal—dual infeasible-interior-point algorithm for linear programming. *Mathematical Programming* 61, 1 (1993), 263–280. https://doi.org/10.1007/BF01582151

[25] Janos Korzak. 2000. Convergence analysis of inexact infeasible-interior-point algorithms for solving linear programming problems. *SIAM Journal on Optimization* 11, 1 (2000), 133–148. https://doi.org/10.1137/S1052623497329993

[26] Yin Tat Lee and Aaron Sidford. 2015. Efficient inverse maintenance and faster algorithms for linear programming. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. 230–249. https://doi.org/10.1109/FOCS.2015.23

[27] Guang Hao Low and Isaac L Chuang. 2019. Hamiltonian simulation by qubitization. *Quantum* 3 (2019), 163.

[28] Shinji Mizuno and Florian Jarre. 1999. Global and polynomial-time convergence of an infeasible-interior-point algorithm using inexact computation. *Mathematical Programming* 84, 1 (1999), 105–122. https://doi.org/10.1007/s101070020a

[29] Mohammadhossein Mohammadisiahroudi, Brandon Augustino, Ramin Fakhimi, Giacomo Nannicini, and Tamás Terlaky. 2023. *Accurately Solving Linear Systems with Quantum Oracles*. Technical Report 23T-006. ISE, Lehigh University.

[30] Mohammadhossein Mohammadisiahroudi, Ramin Fakhimi, Brandon Augustino, and Tamás Terlaky. 2024. Generating linear, semidefinite, and second-order cone optimization problems for numerical experiments. *Optimization Methods and Software* 0, 0 (2024), 1–31. https://doi.org/10.1080/10556788.2024.2308677

[31] Mohammadhossein Mohammadisiahroudi, Ramin Fakhimi, and Tamás Terlaky. 2024. Efficient Use of Quantum Linear System Algorithms in Inexact Infeasible IPMs for Linear Optimization. *Journal of Optimization Theory and Applications* (2024), 1–38.

[32] Mohammadhossein Mohammadisiahroudi, Ramin Fakhimi, Zeguan Wu, and Tamás Terlaky. 2023. An inexact feasible interior point method for linear optimization with high adaptability to quantum computers. *arXiv preprint arXiv:2307.14445* (2023).

[33] Mohammadhossein Mohammadisiahroudi and Tamás Terlaky. 2020. *Quantum IPMs for Linear Optimization*. Springer International Publishing, Cham, 1–11. https://doi.org/10.1007/978-3-030-54621-2_851-1

[34] Renato DC Monteiro, Jerome W O'Neal, and Takashi Tsuchiya. 2004. Uniform boundedness of a preconditioned normal matrix used in interior-point methods. *SIAM Journal on Optimization* 15, 1 (2004), 96–100.

[35] Renato DC Monteiro and Jerome W O'Neal. 2003. *Convergence analysis of a long-step primal-dual infeasible interior-point LP algorithm based on iterative linear solvers*. Technical Report 30332. ISyE, Georgia Institute of Technology. http://www.optimization-online.org/DB_FILE/2003/10/768.pdf

[36] Giacomo Nannicini. 2022. Fast quantum subroutines for the simplex method. *Operations Research* (2022).

[37] Jerome W O'Neal. 2006. *The use of preconditioned iterative linear solvers in interior-point methods and related topics*. Georgia Institute of Technology.

[38] Cornelis Roos, Tamás Terlaky, and J-Ph Vial. 2005. *Interior Point Methods for Linear Optimization*. Springer New York, NY. https://doi.org/10.1007/b100325

[39] Joran van Apeldoorn, Arjan Cornelissen, András Gilyén, and Giacomo Nannicini. 2023. Quantum tomography using state-preparation unitaries. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 1265–1318.

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

[41] Almudena Carrera Vazquez, Ralf Hiptmair, and Stefan Woerner. 2022. Enhancing the quantum linear systems algorithm using Richardson extrapolation. *ACM Transactions on Quantum Computing* 3, 1 (2022), 1–37.

[42] James Hardy Wilkinson. 1963. *Rounding Errors in Algebraic Processes*. Prentice Hall, Englewood Cliffs, NJ, USA.

[43] Leonard Wossnig, Zhikuan Zhao, and Anupam Prakash. 2018. Quantum linear system algorithm for dense matrices. *Physical Review Letters* 120, 5 (2018), 050502.

[44] Stephen J Wright. 1997. *Primal-Dual Interior-Point Methods*. SIAM. https://doi.org/10.1137/1.9781611971453

[45] Zeguan Wu, Mohammadhossein Mohammadisiahroudi, Brandon Augustino, Xiu Yang, and Tamás Terlaky. 2023. An inexact feasible quantum interior point method for linearly constrained quadratic optimization. *Entropy* 25, 2 (2023), 330.

[46] Yinyu Ye, Michael J. Todd, and Shinji Mizuno. 1994. An $O(\sqrt{nL})$-iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research* 19, 1 (1994), 53–67. https://doi.org/10.1287/moor.19.1.53

[47] Guanglu Zhou and Kim-Chuan Toh. 2004. Polynomiality of an inexact infeasible interior point algorithm for semidefinite programming. *Mathematical Programming* 99, 2 (2004), 261–282. https://doi.org/10.1007/s10107-003-0431-5