

Quantum Interior Point Methods for Semidefinite Optimization

Brandon Augustino¹, Giacomo Nannicini², Tamás Terlaky¹, and Luis Zuluaga¹

¹*Department of Industrial and Systems Engineering, Quantum Computing and Optimization Lab, Lehigh University*

²*Department of Industrial and Systems Engineering, University of Southern California*

We present two quantum interior point methods for semidefinite optimization problems, building on recent advances in quantum linear system algorithms. The first scheme, more similar to a classical solution algorithm, computes an inexact search direction and is not guaranteed to explore only feasible points; the second scheme uses a nullspace representation of the Newton linear system to ensure feasibility even with inexact search directions. The second is a novel scheme that might seem impractical in the classical world, but it is well-suited for a hybrid quantum-classical setting. We show that both schemes converge to an optimal solution of the semidefinite optimization problem under standard assumptions. By comparing the theoretical performance of classical and quantum interior point methods with respect to various input parameters, we show that our second scheme obtains a speedup over classical algorithms in terms of the dimension of the problem n , but has worse dependence on other numerical parameters.

Brandon Augustino: bra216@lehigh.edu

Giacomo Nannicini: g.nannicini@usc.edu

Tamás Terlaky: terlaky@lehigh.edu

Luis Zuluaga: luis.zuluaga@lehigh.edu

Contents

1	Introduction	3
1.1	Literature Review	3
1.2	Contributions of this paper	6
2	Preliminaries	7
2.1	Basic concepts on block-encoded matrices	8
2.2	Useful results on block-encoded matrices	10
2.3	Extracting the solution of a linear system: tomography algorithm	13
3	Primal and Dual SDOs and the Central Path	14
3.1	Symmetrizing the Newton System	16
4	Newton Linear Systems for QIPMs	18
4.1	An Infeasible Central Path and its Neighborhood	18
4.1.1	Inexact Search Directions	20
4.2	The Nullspace Representation of the Newton Linear System	21
5	Technical Results	24
6	Quantum Interior Point Methods for SDO	30
6.1	An Inexact-Feasible QIPM for SDO	30
6.1.1	Polynomial Convergence of the IF-QIPM	30
6.1.2	Implementing the Nesterov-Todd linear system	33
6.2	An Inexact-Infeasible QIPM for SDO	36
7	Complexity	37
7.1	Running time of the IF-QIPM	38
7.2	Running time of the II-QIPM	40
7.3	Comparison with existing algorithms for SDO	41
7.4	The relationship between κ and optimality in IPMs	44
8	Conclusion	46
A	Block-encoding the Inexact-Infeasible Newton Systems	46
A.1	Implementing the Inexact-Infeasible Newton linear system for Members of the Monteiro-Zhang Family of Search Directions	46
A.1.1	Implementing the HKM linear system	48
A.1.2	Implementing the AHO linear system	49
B	Full proofs of useful results from the literature	49
B.1	Proof of Lemma 19	49

1 Introduction

We develop quantum interior point methods (QIPMs) for the solution of semidefinite optimization problems (SDOPs). Letting $b \in \mathbb{R}^m$, matrices $A^{(1)}, \dots, A^{(m)}, C \in \mathcal{S}^n$, where \mathcal{S}^n is the space of $n \times n$ symmetric matrices, we consider the primal SDOP given as

$$z_P = \inf_X \left\{ C \bullet X : A^{(i)} \bullet X = b_i, \forall i \in [m], X \succeq 0 \right\}, \quad (1)$$

where $[m] = \{1, \dots, m\}$, $U \bullet V = \text{tr}(UV)$ defines the trace inner product of two matrices U and V , and $U \succeq V$ indicates that $U - V$ is a symmetric positive semidefinite matrix. We assume that the matrices $A^{(1)}, \dots, A^{(m)}$ are linearly independent. The corresponding dual problem is given by

$$z_D = \sup_{y, S} \left\{ b^\top y : \sum_{i=1}^m y_i A^{(i)} + S = C, S \succeq 0, y \in \mathbb{R}^m \right\}. \quad (2)$$

The so called Interior Point Condition (IPC) holds if there is a primal feasible X with $X \succ 0$, and there exists a dual feasible (y, S) with $S \succ 0$. Although duality results are generally weaker for SDO when compared to linear optimization (LO), under the IPC strong duality holds; optimal solutions exist for both the primal and dual SDOPs, and their optimal objective values are equal, i.e., $z_P = z_D$.

Interior point methods (IPMs) are algorithms to solve convex optimization problems, including semidefinite optimization problems, in polynomial time. The first polynomial-time algorithm for linear optimization problems (LOPs), a special class of convex optimization problems, is the ellipsoid method [1]. Karmarkar's IPM improved on the complexity of the ellipsoid method [2]. The first polynomial-time IPM for SDOPs is attributed to Nesterov and Nemirovskii [3, 4]. Their framework employs efficiently-computable self-concordant barrier functions.

The interest in SDO originates from the fact that SDO provides a framework that allows one to formulate fundamental problems in control [5], statistics, information theory [6], machine learning [7, 8], finance [9, 10], and quantum information science [11–13], to name a few. Additionally, SDO can provide tight approximations for various combinatorial optimization problems [14, 15], and it can be used to study the properties of convex optimization problems [5]. SDOPs are attractive since they exhibit practical efficiency [16–18] and tractability [3, 19]. LOPs are a special case of SDOPs in which each of the input matrices $C, A^{(1)}, \dots, A^{(m)}$ are diagonal matrices. Considering the vast number of applications of SDO, it is natural to ask if quantum computers can accelerate their solution. Several works have investigated this possibility, which we discuss next.

1.1 Literature Review

When $m \leq \sqrt{n}$, the Cutting Plane Methods of [20, 21] are the best performing classical algorithms for solving SDOPs, and can do so in time

$$\mathcal{O} \left(m(mn^2 + m^2 + n^\omega) \cdot \text{polylog} \left(m, n, R, \frac{1}{\epsilon} \right) \right),$$

where $\omega \in [2, 2.38]$ is the matrix multiplication exponent, R is an upper bound on the trace of primal optimal solutions, ϵ is the precision parameter, and mn^2 is the input size of the SDO problem. However, in practical situations we typically have that m is roughly between n and n^2 , in which case the Cutting Plane Methods given in [20, 21] are outperformed by the IPM for SDO from Jiang et al. [22]. Their IPM exhibits a worst case running time of

$$\mathcal{O} \left(\sqrt{n}(mn^2 + m^\omega + n^\omega) \cdot \text{polylog} \left(m, n, \frac{1}{\epsilon} \right) \right),$$

where the term $m^\omega + n^\omega$ represents the per-iteration cost of inverting the Hessian and dual slack matrices. For IPMs, ϵ -optimality implies that the primal and dual feasible solutions exhibit a *normalized* duality gap bounded by ϵ , i.e.:

$$\frac{X \bullet S}{n} \leq \epsilon.$$

One can also note that when $m = \mathcal{O}(n^2)$, the running time of the IPM in [22] is shown to be $\tilde{\mathcal{O}}_{\frac{1}{\epsilon}}(n^{5.246})$, providing an enhancement over the running time bound of $\tilde{\mathcal{O}}_{\frac{1}{\epsilon}}(n^{6.5})$ found in [23–25]. Here, the notation $\tilde{\mathcal{O}}_{\alpha,\beta}(f(x))$ suppresses polylogarithmic factors in α and β in the running time, i.e., $\tilde{\mathcal{O}}_{\alpha,\beta}(f(x)) = \mathcal{O}(f(x) \cdot \text{polylog}(\alpha, \beta, f(x)))$.

Although IPMs are prevalent in practice, they are not the only approach to solve SDOPs. The classical matrix multiplicative weights update (MWU) algorithm of Arora and Kale [26] is one such algorithm that has been quantized with notable gains. The classical MWU method has a running time of

$$\mathcal{O}\left(mn^2 \cdot \text{poly}\left(\frac{Rr}{\epsilon_{\text{abs}}}\right)\right),$$

where r is an upper bound on the ℓ_1 -norm of dual optimal solutions, and ϵ_{abs} is an additive error to which the optimal objective value is approximated. That is, the objective value attained by the acquired solution is $\text{OPT} \in [\varsigma - \mathcal{O}(\epsilon_{\text{abs}}), \varsigma + \mathcal{O}(\epsilon_{\text{abs}})]$, where ς is a bound on the optimal objective value determined using binary search. The MWU algorithm alternates between candidate solutions to the primal and dual SDO problems, and does not involve the solution of linear systems. A quantum SDO solver based on the Arora-Kale framework was constructed by Brandão and Svore [27], and van Apeldoorn et al. [28]. With successive improvements [29, 30], the running time of the quantum MWU algorithm for SDOPs has been brought down to:

$$\mathcal{O}\left(\left(\sqrt{m} + \sqrt{n} \frac{Rr}{\epsilon_{\text{abs}}}\right) s \left(\frac{Rr}{\epsilon_{\text{abs}}}\right)^4 \cdot \text{polylog}\left(m, n, R, \frac{1}{\epsilon_{\text{abs}}}\right)\right),$$

where s is a sparsity parameter (maximum number of non-zero entries per row). While these methods achieve running times with a more favorable dependence on m and n as compared to classical IPM solvers, their polynomial dependence on the parameters R , r and $1/\epsilon_{\text{abs}}$ is exponentially slower than classical IPMs, which exhibit logarithmic dependence on the inverse precision. (If ϵ_{abs} is considered to be part of the input, it is expressed in binary format and a running time of $1/\epsilon_{\text{abs}}$ is not polynomial in the usual sense.)

The QIPM presented by Kerenidis and Prakash [31] (rather, the earlier arXiv version of that paper) initiated the other major research direction regarding the solution of SDOPs with quantum algorithms. By constructing block-encodings of the Newton linear system matrix, which can be efficiently constructed at runtime using quantum random access memory (QRAM), the authors in [31] solve the Newton linear system in time polylogarithmic in n (but note that this does not include the time to obtain a classical description of the solution). As the solution of the Newton linear system is the most expensive step in classical IPMs, the hope is to obtain a quantum advantage. The paper [31] posits a worst case running time of

$$\mathcal{O}\left(\frac{n^{2.5}}{\xi^2} \mu \kappa^3 \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

for SDOPs, where ϵ is the optimality gap tolerance to which we solve the SDOP, ξ is a feasibility tolerance, κ is an upper bound for the condition number of the Newton systems, and μ is upper bounded by the largest Frobenius norm of the coefficient matrix of the Newton systems. Note, however, that the algorithm of [31] does not converge to an ϵ -optimal solution in the usual sense, for reasons that will be discussed below; thus, this running time expression does not settle the complexity of QIPMs for SDOPs. The term $\mathcal{O}\left(\frac{n^2 \kappa^2}{\xi^2}\right)$ comes from a tomography subroutine that is used to map the quantum-mechanical encoding of the solution of the Newton linear system, to a classical description used to construct the subsequent iterate. This reconstruction is performed in each step to ℓ_2 -norm error ξ/κ . When this algorithm is applied to solve LOPs, the running time is

$$\mathcal{O}\left(\frac{n^{1.5}}{\xi^2} \mu \kappa^3 \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right).$$

Similar ideas have been used in other recent papers [32, 33], all of which follow the same structure: they apply a classical interior point method, and substitute the classical solution of the Newton linear system (or equivalent) with a quantum linear system algorithm, with the goal of improving

the running time. The solution of the linear system is extracted from the quantum computer via tomography. The analysis of the algorithms requires nontrivial machinery and has led to interesting developments. One limitation of some of these existing QIPM works is the use of the feasibility tolerance ξ , that does not translate to the usual concept of primal feasibility. Another limitation is that the analysis of [31] is heavily based on classical exact IPMs, and does not account for some challenges that arise when translating those algorithms to the quantum setting. Similar challenges are found in [32, 33]. These challenges are discussed below, and addressed in this paper for the specific case of semidefinite optimization.

The QIPM of [31] directly quantizes the classical IPM described in [34]. This leads to two major issues. The first is that this algorithm does not guarantee that the solutions to the Newton linear system remain symmetric [35]. As a consequence, the algorithm described in [31] may not converge when applied to SDOPs (this is not an issue for linear optimization, because all the involved matrices are diagonal). Additional safeguards must be employed to guarantee that the resulting solution will be symmetric; numerous techniques have been proposed for this purpose. These methods are generally grouped into families of search directions such as those introduced by Kojima, Shindoh, and Hara [36], Monteiro and Zhang [23, 37–39], Monteiro and Tsuchiya [40], and Tseng [41]. We refer the reader to an overview of these (as well as many other) search directions by Todd [35].

The second major issue is that the analysis of the IPM in [34] assumes that the Newton linear systems are solved exactly. This cannot be done in the quantum setting, because nonnegligible errors are introduced in the tomography step, even if the quantum linear systems algorithm can solve the problem with high precision. Notably, the primal and dual search directions resulting from tomography are not orthogonal, therefore the convergence proof of [34] no longer applies¹. To ensure convergence when the search directions are not exact, we must rely on IPM frameworks that can deal with the errors introduced from tomography.

Infeasible IPMs were originally motivated by the need of having an initial point (and all subsequent iterates) in the interior of the semidefinite cone. It is easy to have an initial interior point which is infeasible [42, 43], hence the use of infeasible IPMs. Further, excessive CPU time and memory requirements may prevent solving large linear systems directly. In such a situation, one can employ an iterative method to determine the solutions to the linear systems to compute search directions, and such solutions are not guaranteed to be exact (e.g., due to early stopping of the iterative method). The resulting algorithm is called inexact-infeasible IPM. The first polynomial time inexact-infeasible IPMs were proposed by Korzak [44] and Mizuno and Jarre [45]. Inexact-infeasible IPMs are natural candidates for quantization. Using this terminology, the “prototypical” classical IPM described in [34] can be called an exact-feasible IPM.

The little-studied inexact-feasible IPM (IF-IPM) bridges the gap between exact-feasible and inexact-infeasible IPMs. In an inexact-feasible IPM, it is typically assumed that primal and dual feasibility can be satisfied exactly, but we allow for a margin of error in solving a reduced Newton system. We argue that this framework is the most well-suited to serve as a foundation for QIPMs, compared to inexact-infeasible IPMs. Indeed, inexact-feasible IPMs exhibit the same iteration complexity as exact-feasible IPMs, and they impose milder restrictions on the inexactness of the Newton linear system; because of this, quantizing an inexact-feasible IPM leads to better theoretical performance than an inexact-infeasible IPM. Gondzio [46] presented an Inexact-Feasible IPM for convex quadratic optimization problems; his complexity analysis is the basis for the analysis of our novel IF-QIPM, although our QIPM bears little resemblance to the classical IF-IPMs. Indeed, when designing an inexact-feasible QIPM we make several choices that are very atypical for classical IPMs, yielding an algorithm with no direct classical analogue, regardless of the fact that its analysis relies on known classical tools. One key distinction being that the IPM in [46] *assumes* that primal and dual feasibility can be satisfied exactly, whereas our IF-QIPM *guarantees* primal-dual feasibility.

¹Lemma 6.3 in [31] requires $dS \bullet dY = 0$, but this property does not hold when the Newton linear system is solved inexactly; it is not clear how to amend the subsequent analysis using the same framework, since even convergence of purely classical algorithms requires rigorous assumptions on the linear system errors.

1.2 Contributions of this paper

This paper highlights several challenges in the literature on QIPMs that have been previously overlooked. To address some of these challenges, it develops two provably convergent QIPMs for SDO. One algorithm is a quantization of the classical inexact-infeasible IPM from [47]; the other algorithm is an inexact-feasible QIPM loosely based on [46], where we introduce significant differences for the quantum setting. The inexact-infeasible QIPM converges to an ϵ -approximate solution to an SDOP in $\mathcal{O}(n^2 \log(\frac{1}{\epsilon}))$ iterations, the inexact-feasible QIPM converges in $\mathcal{O}(\sqrt{n} \log(\frac{1}{\epsilon}))$ iterations.

Our first algorithm can be seen as a direct extension of [31] to properly address the two main issues discussed above, i.e., ensuring the existence of symmetric solutions and dealing with inexact search directions. This leads to the inexact-infeasible QIPM (II-QIPM). The running time of this algorithm, when using the HKM direction [23, 36, 48], is

$$\mathcal{O}\left(\left(n^{5.5} \kappa \kappa_{\mathcal{A}} \rho (\|\mathcal{A}\|_F + \rho n^{1.5})\right) \frac{1}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right),$$

where $\kappa_{\mathcal{A}}$ is the condition number of the matrix \mathcal{A} whose columns are the vectorized constraint matrices $A^{(1)}, \dots, A^{(m)}$, and $\rho > 0$ is used to define the size of the initial infeasible solution and is generally considered a constant in many papers (i.e., it is often ignored in the running time analysis). Comparing the running time of the inexact-infeasible QIPM to classical IPM, there does not seem to be an advantage in any parameter: classical exact-feasible IPMs (with running time $\mathcal{O}(n^{2\omega+0.5} \log 1/\epsilon)$ on dense problems) are superior to the above quantum algorithm. While the II-QIPM does not achieve any advantage compared to classical algorithms, we discuss it because it shows one possible way of dealing with the issues of existing QIPMs in the open literature, highlighting the fact that quantum advantage seems unlikely in this setting: to improve upon classical algorithm at least in some parameters, we need to modify the basic classical IPM scheme.

Our second algorithm, the inexact-feasible QIPM (IF-QIPM), solves the primal-dual SDO pair (1)-(2) with $m = \mathcal{O}(n^2)$ constraints to ϵ -optimality using at most

$$\mathcal{O}\left(n^{3.5} \frac{\kappa^2}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

QRAM accesses and $\mathcal{O}(n^{4.5} \text{polylog}(n, \kappa, \frac{1}{\epsilon}))$ arithmetic operations. When applied to LO problems, the IF-QIPM requires

$$\mathcal{O}\left(n^2 \frac{\kappa^2}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

QRAM accesses and $\mathcal{O}(n^{2.5} \cdot \text{polylog}(n, \kappa, \frac{1}{\epsilon}))$ arithmetic operations.

For SDO problems involving $n \times n$ matrices and $m = \mathcal{O}(n^2)$ constraints, the IF-QIPM achieves speedups in terms of n compared to both classical feasible IPMs, and II-(Q)IPMs existing in the literature, though its dependence on κ and $1/\epsilon$ prohibits an overall speedup, see, Section 7.4. We also de-quantize our framework to obtain a classical IF-IPM, finding that our classical algorithm exhibits an overall running time of

$$\mathcal{O}\left(n^{4.5} \kappa \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right)$$

when applied to SDO problems $m = \mathcal{O}(n^2)$. Like the IF-QIPM, our IF-IPM is faster in n than all existing classical IPMs, and its dependence on ϵ is exponentially faster than its quantum counterpart. Yet, we cannot decisively conclude that the IF-IPM outperforms the current state of the art classical IPMs for SDO; its running time still depends (linearly) on κ . When applied to LO problems, our IF-IPM has a running time of $\mathcal{O}(n^{2.5} \kappa \cdot \text{polylog}(\kappa, \frac{1}{\epsilon}))$. Note that this does not outperform the best classical IPM for LO even with respect to the dimension n : classical randomized IPMs can solve LOPs with $m = n$ constraints in time $\tilde{\mathcal{O}}_{n, \frac{1}{\epsilon}}(n^\omega)$, see [49].

It should be noted that the running time dependence on the condition number κ is due to use of a QLSA for the solution of the Newton linear system. It is known that κ can grow very large over the course of IPMs (regardless of problem class, e.g., LO or SDO), and in fact $\kappa \rightarrow \infty$ as we

approach optimality, see the discussion in Section 7.4. The QIPM has better dependence on $1/\epsilon$ than the quantum MWU, but this comes at a cost of worse dependence on other parameters, most notably n and κ . It is worth noting, however, that QIPMs (as they are presented in this work) classically output the primal-dual optimal solution (X^*, y^*, S^*) , whereas QMWU algorithms only output y^* and a way to construct X^*, S^* . In addition, the solutions we obtain using the IF-QIPM will satisfy primal and dual feasibility *exactly*, while (Q)MWU algorithms can only obtain solutions which satisfy the constraints to some additive error, e.g., in the primal case we have

$$A^{(i)} \bullet X = b_i + \epsilon, \forall i \in [m].$$

It should also be noted that (Q)IPMs achieve a normalized duality gap $\frac{X^* \bullet S^*}{n}$ at most ϵ , whereas with (Q)MWUs the guarantee is ϵ -additive. For a more detailed comparison of (Q)IPMs and (Q)MWUs, see Section 7.3. The QIPMs developed here enjoy a lower complexity than classical IPMs only if ϵ is relatively large, and κ remains polynomially small over the course of the algorithm, a condition that cannot be guaranteed in theory. In practice, it is possible that these algorithms exhibit good performance if we are interested in obtaining a low-precision solution.

Summarizing, based on the current state of the art for classical and quantum algorithms, quantum algorithms for linear systems do not seem sufficient to obtain a speedup over classical IPMs, due to the challenges highlighted above. This was also observed in [50] comparing classical and quantum algorithms for a specific application of SDO in combinatorial optimization. It is known that the dependence of QLSAs on κ cannot in general be reduced to $\kappa^{1-\delta}$ for any $\delta > 0$ unless $\text{BQP} = \text{PSPACE}$ [51]. We therefore conjecture that a quantum speedup would have to rely on different techniques: the direct quantization of a classical IPM, with linear algebra performed on a quantum computer, does not seem to be sufficient for an end-to-end speedup in general.

Our paper identifies some open questions. First, can we reduce the dependence on $1/\epsilon$ from linear to polylogarithmic? A promising direction to attain this objective is that of iterative refinement techniques, which have been successfully applied in the classical optimization literature for LOPs [52–54]. We conjecture that they can be adapted to the setting of SDO, and this may help us improve the ϵ -dependence of the QIPM. Second, is there a way to improve the dependence on κ ? This seems difficult within the QLSA framework, due to the lower bounds of [51, 55], but in the context of QIPM, it would still be advantageous to use techniques that allow us to trade κ for a slightly higher dependence on n – which, crucially, is *exponential* in the number of qubits anyway.

The rest of this paper is organized as follows. In Section 2, we introduce our notation, some necessary concepts in linear algebra, and basic quantum procedures. In Section 3 we discuss the main components of classical IPMs. Section 4 discusses how to deal with tomography errors, as well as algorithmic variants that allow one to capture inexactness and infeasibility. Section 5 provides technical results that are crucial to the convergence analysis of the IF-QIPM. In Section 6 we discuss its quantization, and provide a theoretical analysis of the running time for the Inexact-Feasible and Inexact-Infeasible Quantum Interior Point Methods in Section 7. Section 8 concludes the paper. Some technical results, and results from other papers are presented in Appendices A and B, respectively.

2 Preliminaries

We write A^\dagger to indicate the conjugate transpose of A . We denote by $G \otimes K$ the tensor product of two matrices. For a matrix $G \in \mathbb{R}^{n \times n}$, $\text{vec}(G)$ is the $n^2 \times 1$ vector given by

$$\text{vec}(G) = (g_{11}, g_{21}, \dots, g_{n1}, g_{12}, \dots, g_{nn})^\top.$$

We make extensive use of the functions svec and smat , formally defined as follows.

Definition 1. For $G \in \mathcal{S}^n$, $\text{svec}(G) \in \mathbb{R}^{\frac{1}{2}n(n+1)}$ is given by

$$\text{svec}(G) = \left(g_{11}, \sqrt{2}g_{21}, \dots, \sqrt{2}g_{n1}, g_{22}, \sqrt{2}g_{32}, \dots, \sqrt{2}g_{n2}, \dots, g_{nn} \right)^\top.$$

Further, the operator smat is the inverse operator of svec . That is,

$$\text{smat}[\text{svec}(G)] = G.$$

We also use the symmetric Kronecker product, defined below.

Definition 2. Let $G, K \in \mathcal{S}^n$. Define the $\frac{n(n+1)}{2} \times n^2$ matrix V as:

$$V_{(i,j),(k,l)} = \begin{cases} 1 & \text{if } i = j = k = l, \\ 1/\sqrt{2} & \text{if } i = j \neq k = l, \text{ or } i = l \neq j = k. \\ 0 & \text{otherwise.} \end{cases}$$

Then, the symmetric Kronecker product $G \otimes_s K$ of G and K is defined as:

$$G \otimes_s K = \frac{1}{2} V (G \otimes K + K \otimes G) V^\top.$$

We let \mathcal{S}_+^n and \mathcal{S}_{++}^n denote the cones of symmetric positive semidefinite, and symmetric positive definite matrices, respectively. Given two vectors x, y , we denote by $x \circ y$ the concatenation of the two vectors. For $x \in \mathbb{R}^n$, we denote its amplitude encoding by $|x\rangle$, defined as

$$|x\rangle = \frac{1}{\|x\|} \sum_{j \in [n]} x_j |j\rangle.$$

Notice that $|x\rangle$ is a $(\log n)$ -qubit state; for simplicity, we assume that the sizes of all spaces are powers of 2. All logarithms are base 2. The smallest and largest singular values of a matrix A are denoted $\sigma_{\min}(A), \sigma_{\max}(A)$, and the smallest and largest eigenvalues are denoted $\lambda_{\min}(A), \lambda_{\max}(A)$. The condition number of A is $\kappa_A = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$; we simply write κ when referring the upper bound on the condition number of the Newton linear system. For an element $x \in \mathbb{C}^d$, $\Re(x) \in \mathbb{R}^d$ is its real part. We write ‘‘II’’ for inexact-infeasible and ‘‘IF’’ for inexact-feasible.

2.1 Basic concepts on block-encoded matrices

The QIPMs described in this paper make extensive use of quantum linear algebra subroutines; notably, matrix multiplication and matrix-vector products. We perform all these operations in the framework of block-encodings, as discussed in [56, 57]. The block-encoding framework generalizes earlier work on high-accuracy QLSAs in the sparse-access input model [58] by expanding not only the possible input models, but also the set of operations allowed on the input matrix. Before giving formal definitions and an overview of key results from the literature, we give an informal introduction of the work in [56] to convey intuition.

To get a basic understanding of how to perform matrix-vector and matrix-matrix multiplication using this framework, suppose we want to multiply a matrix A by some vector u . We assume that u is encoded in the quantum state $|u\rangle$, which we want to update as follows:

$$|u\rangle \rightarrow \frac{A|u\rangle}{\|A|u\rangle\|}.$$

A block-encoding U of A is a unitary such that:

$$U = \begin{pmatrix} A & \cdot \\ \alpha & \cdot \end{pmatrix},$$

where α is a normalization factor that is chosen to ensure that U has operator norm at most 1, i.e., $\|U\| \leq 1$. In particular, if A is a w -qubit operator, an (α, a, ξ) -block-encoding of A uses a extra qubits and implements A up to total error ξ . Hence, we have

$$U |u\rangle |0\rangle^{\otimes a} \approx \begin{pmatrix} A & \cdot \\ \alpha & \cdot \end{pmatrix} \begin{pmatrix} |u\rangle \\ |0\rangle \end{pmatrix} = \frac{1}{\alpha} A |u\rangle |0\rangle^{\otimes a} + |\cdot\rangle,$$

and performing approximately $\frac{\alpha}{\|A|u\rangle\|}$ rounds of amplitude amplification yields $\frac{A|u\rangle}{\|A|u\rangle\|}$ with high probability, see e.g., [56].

For matrix multiplication, let U_A be an (α_1, a_1, ξ_1) -block-encoding of an w -qubit operator A , and U_B be a (α_2, a_2, ξ_2) -block-encoding of a w -qubit operator B . Then we have

$$(I_{a_2} \otimes U_A)(I_{a_1} \otimes U_B) = \begin{pmatrix} \frac{B}{\alpha_2} & \cdot \\ \cdot & \cdot \end{pmatrix} \begin{pmatrix} \frac{A}{\alpha_1} & \cdot \\ \cdot & \cdot \end{pmatrix} = \begin{pmatrix} \frac{AB}{\alpha_1 \alpha_2} & \cdot \\ \cdot & \cdot \end{pmatrix},$$

which yields an $(\alpha_1 \alpha_2, a_1 + a_2, \alpha_1 \xi_2 + \alpha_2 \xi_1)$ -block-encoding of AB .

For matrix inversion, let U be a (α, a, ξ) -block encoding of A that can be implemented in time T_U . Then, Chakraborty et al. [56] prove that we can implement a block-encoding V of the inverse of A :

$$V = \begin{pmatrix} \frac{A^{-1}}{2\kappa} & \cdot \\ \cdot & \cdot \end{pmatrix}$$

Hence, given A as a block-encoding and $|u\rangle$, we can solve the linear system

$$Ax = u$$

by outputting the state

$$\frac{A^{-1} |u\rangle}{\|A^{-1} |u\rangle\|},$$

computed by applying amplitude amplification to the state:

$$V |u\rangle |0\rangle^{\otimes a} = \frac{1}{2\kappa} A^{-1} |u\rangle |0\rangle^{\otimes a} + |\cdot\rangle.$$

To construct block-encodings of the matrices used by our QIPMs, we use the following construction, see [56] for details. Suppose we want to construct a matrix A and we have a way of constructing the quantum states $|\psi_i\rangle = \sum_j \frac{A_{ij}}{\|A_i\|} |i, j\rangle$, where A_i is the i -th row of A , and $|\phi_j\rangle = \sum_i \frac{\|A_i\|}{\|A\|_F} |i, j\rangle$. Then we construct:

$$U_R^\dagger U_L = \begin{pmatrix} - & \langle \psi_1 | & - & \cdot \\ & \vdots & & \\ - & \langle \psi_n | & - & \cdot \\ & \cdot & & \end{pmatrix} \begin{pmatrix} | & & | \\ | \phi_1 \rangle & \cdots & | \phi_n \rangle & \cdot \\ | & & | & \\ \cdot & & \cdot & \cdot \end{pmatrix} = \begin{pmatrix} [\langle \psi_i | \phi_j \rangle]_{i,j \in [n]} & \cdot \\ \cdot & \cdot \end{pmatrix} = \begin{pmatrix} \frac{A}{\|A\|_F} & \cdot \\ \cdot & \cdot \end{pmatrix},$$

where we used the fact that

$$\langle \psi_i | \phi_j \rangle = \frac{A_{ij}}{\|A\|_F}.$$

Thus, $U_R^\dagger U_L$ is a block-encoding of A with normalization factor $\alpha = \|A\|_F$. For the above construction to work, we must have a procedure to implement U_R, U_L . Note that these unitaries are easy to obtain starting from controlled operations to prepare the states $|\psi_i\rangle, |\phi_j\rangle$, i.e., operations of the form $|i\rangle |0\rangle \rightarrow |i\rangle |\psi_i\rangle, |j\rangle |0\rangle \rightarrow |j\rangle |\phi_j\rangle$, and these controlled operations can in turn be constructed with a procedure similar to the state preparation algorithm of Grover and Rudolph [59] for efficiently integrable distributions. This can be made more efficient if we allow pre-processing to create certain data structures that can be stored in quantum-accessible storage, i.e., QRAM. A quantum RAM (QRAM) is a form of storage that allows for querying a superposition of addresses. Given a QRAM that stores the classical vector $v_j \in \mathbb{R}^{2^q}$, and a quantum state $\sum_{j=0}^{2^q-1} \beta_j |j\rangle$, the QRAM assumption is that the following mapping can be performed in time $\tilde{\mathcal{O}}_n(1)$, i.e., polylogarithmic in the size of the vector:

$$\sum_{j=0}^{2^q-1} \beta_j |j\rangle \otimes |0\rangle \rightarrow \sum_{j=0}^{2^q-1} (\beta_j |j\rangle \otimes |v_j\rangle).$$

Throughout this paper, we assume that we have access to a classical-write, quantum-read QRAM that is large enough to store all input matrices, as well as a number of vectors so that we can efficiently perform indexed SWAP operations in our tomography subroutine (see Section 2.3),

i.e., $\tilde{\mathcal{O}}_{n, \frac{1}{8}, \frac{1}{\epsilon}} \left(mn^2 + 2 \frac{n^2}{\epsilon} \right)$. The exponential term in the QRAM size comes from the tomography subroutine if we insist on a classical-write QRAM, but as we discuss in 2.3, there are several alternative models to reduce this cost: for example, if the QRAM can be written in superposition, or if we have access to indexed SWAP gates, then the necessary QRAM size is only polynomial. For more details about the QRAM data structure to prepare the amplitude encoding of a vector, or the matrices U_R, U_L discussed above, we refer the reader to [31, 56].

2.2 Useful results on block-encoded matrices

We now provide formal definitions for the concepts informally discussed in the previous section, as well as other results that are used in the remainder of this paper. The material in this section is mostly taken from [56, 57], which provide improvements over the framework discussed in [31, 60].

Definition 3 (block-encoding). *Let $A \in \mathbb{C}^{2^w \times 2^w}$ be a w -qubit operator. Then, a $(w + a)$ -qubit unitary U is an (α, a, ξ) -block-encoding of A if $U = \begin{pmatrix} \tilde{A} & \\ & \cdot \end{pmatrix}$, such that*

$$\|\alpha \tilde{A} - A\| \leq \xi$$

An (α, a, ξ) -block-encoding of A is said to be efficient if it can be implemented in time $T_U = \mathcal{O}(\text{poly}(w))$.

Note that Definition 1 is equivalent to the following property:

$$\|A - \alpha(|0\rangle^{\otimes a} \otimes I_{2^w})U(|0\rangle^{\otimes a} \otimes I_{2^w})\| \leq \xi.$$

The next proposition formalizes an idea discussed in the previous section. We do not provide details on the necessary data structure, referring the reader to [56] for an extensive discussion on this topic.

Proposition 1 (Lemma 50 in [57]). *Let $A \in \mathbb{C}^{m \times m}$ with $m = 2^w$ and $\xi > 0$.*

(i) *Fix $q \in [0, 2]$ and define $\mu_q(A) = \sqrt{n_q(A)n_{(2-q)}(A^\top)}$ where $n_q(A) = \max_i \|a_i\|_q^q$ is the q -th power of the maximum q -norm of the rows of A . If A^q and $(A^{2-q})^\dagger$ are both stored in QRAM data structures, then there exist unitaries U_R and U_L that can be implemented in time $\mathcal{O}(\text{poly}(w \log \frac{1}{\xi}))$ and such that $U_R^\dagger U_L$ is a $(\mu_q(A), w + 2, \xi)$ -block-encoding of A .*

(ii) *If A is stored in a QRAM data structure, then there exist unitaries U_R and U_L that can be implemented in time $\mathcal{O}(\text{poly}(w \log \frac{1}{\xi}))$ and such that $U_R^\dagger U_L$ is an $(\|A\|_F, w + 2, \xi)$ -block-encoding of A .*

Definition 4 (State preparation pair). *Let $y \in \mathbb{C}^m$ and $\|y\|_1 \leq \beta$. The pair of unitaries (P_L, P_R) is called a (β, p, ξ) -state-preparation-pair if $P_L|0\rangle^{\otimes p} = \sum_{j=0}^{2^p-1} c_j |j\rangle$ and $P_R|0\rangle^{\otimes p} = \sum_{j=1}^{2^p-1} d_j |j\rangle$ such that $\sum_{j=0}^{m-1} |\beta(c_j^* d_j) - y_j| \leq \xi$ and for all $j \in m, \dots, 2^p - 1$ we have $c_j^* d_j = 0$.*

Proposition 2 (Lemma 52 in [57]). *(Linear combination of block-encoded matrices, with weights given by a state preparation pair) Let $A = \sum_{j=0}^{m-1} y_j A^{(j)}$ be a w -qubit operator, where $A^{(j)}$ are matrices. Suppose P_L, P_R is a (β, p, ξ_1) -state-preparation pair for y , $W = \sum_{j=0}^{m-1} |j\rangle \langle j| \otimes U_j + ((I - \sum_{j=0}^{m-1} |j\rangle \langle j|) \otimes I_a \otimes I_w)$ is an $(w + a + p)$ -qubit unitary with the property that U_j is an (α, a, ξ_2) -block-encoding of $A^{(j)}$. Then we can implement a $(\alpha\beta, a + p, \alpha\xi_1 + \alpha\beta\xi_2)$ -block-encoding of A with a single use of W, P_R and P_L^\dagger .*

The following two propositions are critical to the efficiency of the QIPMs. They state that one can construct a block-encoding as a product of two block-encoded matrices, with overhead that is merely polylogarithmic in the size of the matrices. The difference between the two propositions is that in the second one, the normalization factor of the block-encoding of the product is fixed, rather than dependent on the input.

Proposition 3 (Lemma 4 [56]). *(Product of block-encoded matrices) If U_A is an (α_1, a_1, ξ_1) -block-encoding of an s -qubit operator A , and U_B is an (α_2, a_2, ξ_2) -block-encoding of an s -qubit operator B , then $(I_{a_2} \otimes U_A)(I_{a_1} \otimes U_B)$ is an $(\alpha_1\alpha_2, a_1 + a_2, \alpha_1\xi_2 + \alpha_2\xi_1)$ -block-encoding of AB .*

Proposition 4 (Lemma 5 in [56]). *(Product of preamplified block-encoded matrices) Suppose A and B have sizes such that AB is a valid matrix product, and $\|A\|, \|B\| \leq 1$. Given an (α_1, a_1, ξ_1) -block-encoding U_A of an s -qubit operator A , and a (α_2, a_2, ξ_2) -block-encoding U_B of an w -qubit operator B , with $\alpha_1, \alpha_2 \geq 1$, we can implement a $(2, a_1 + a_2 + 2, \sqrt{2}(\xi_1 + \xi_2 + \xi_3))$ -block-encoding of AB in time $\mathcal{O}((\alpha_1(T_{U_A} + a_1) + \alpha_2(T_{U_B} + a_2)) \log \frac{1}{\xi_3})$, where T_{U_A} and T_{U_B} are the implementation times for U_A and U_B , respectively.*

If one has a block-encoded matrix, one can also implement a block-encoding of (positive or negative) powers of that matrix, see [56, Lemma 9, 10]. We are now in a position to state how block-encodings can be used to solve quantum linear systems problems, which is formalized in the following result from [56].

Theorem 1 (Theorem 30 in [56]). *(Solution of linear system) Let $r \in (0, \infty)$, $\kappa \geq 2$ and H a Hermitian matrix such that its nonzero eigenvalues lie in $[-1, -1/\kappa] \cup [1/\kappa, 1]$. Suppose that*

$$\xi = o\left(\frac{\delta}{\kappa^2 \log^3 \frac{\kappa^2}{\delta}}\right)$$

and U is an (α, a, ξ) -block-encoding of H , that can be implemented in time T_U . Suppose further that we can prepare a state $|v\rangle$ that is in the image of H in time T_v . Then, for any δ , we can output a state that is δ -close to $H^{-1}|v\rangle / \|H^{-1}v\|$ in time

$$\mathcal{O}\left(\kappa\left(\alpha(a + T_U) \log^2\left(\frac{\kappa}{\delta}\right) + T_v\right) \log \kappa\right).$$

Proposition 5 (Corollary 32 in [56]). *(Norm estimation) Let $p \in (0, \infty)$, $\kappa \geq 2$ and H a Hermitian matrix such that its nonzero eigenvalues lie in $[-1, -1/\kappa] \cup [1/\kappa, 1]$. Suppose that*

$$\xi = o\left(\frac{\delta}{\kappa^2 \log^3 \frac{\kappa^2}{\delta}}\right)$$

and U is an (α, a, ξ) -block-encoding of H , that can be implemented in time T_U . Suppose further that we can prepare a state $|v\rangle$ that is in the image of H in time T_v . Then we can output \tilde{e} such that

$$(1 - \delta)\|H^{-1}|v\rangle\| \leq \tilde{e} \leq (1 + \delta)\|H^{-1}|v\rangle\|$$

in time

$$\mathcal{O}\left(\frac{\kappa}{\delta}\left(\alpha(a + T_U) \log^2\left(\frac{\kappa}{\delta}\right) + T_v\right) \log^3 \kappa \log \frac{\log \kappa}{\xi}\right).$$

Proposition 6. *Let*

$$A = \begin{pmatrix} M_{11} & \dots & M_{1c} \\ \vdots & \ddots & \vdots \\ M_{r1} & \dots & M_{rc} \end{pmatrix} \in \mathbb{R}^{n \times n},$$

where for simplicity r, c are powers of 2, and each M_{ij} is a matrix, of appropriate dimension, that is stored in a QRAM data structure. Suppose further the row norms and column norms of each matrix M_{ij} are known. Then we can construct a $(rc\|A\|_F, \mathcal{O}(\log n), \xi)$ -block-encoding of A in time $\mathcal{O}(\text{poly}(rc, \log n, \log \frac{1}{\xi}))$.

Proof. Let $A^{(1)}, \dots, A^{(m)}$ denote the $m = rc$ matrices in $\mathbb{R}^{n \times n}$ that each store one of the M_{ij} matrices in the same position that it appears in A , but with all other entries being 0. Then, A can be written as a linear combination of the m many matrices $A^{(k)}$ as $A = \sum_{k=1}^m y_k A^{(k)}$, where $y_k = 1$ for all k . We normalize all matrices using the Frobenius norm of A , to ensure that all spectral norms are ≤ 1 .

Notice that if a matrix M_{ij} is stored in QRAM, we can construct the block-encoding of the corresponding $A^{(k)}$ using Proposition 1, with straightforward modifications of the underlying algorithm, because the elements and norms of $A^{(k)}$ can be efficiently computed from those of M_{ij} . Thus, a $(\|A\|_F, \log n + 2, \xi')$ -block-encoding of $A^{(k)}$ can be efficiently constructed, for any $\xi' > 0$.

Let P_L, P_R be a $(m, \log m, 0)$ -state-preparation pair for y , which can be constructed by simply taking $P_L = P_R = H^{\otimes \log m}$ where H is the Hadamard gate. We then use Proposition 2, where W can be obtained by adding a control qubit to the circuits used to construct the block-encoding of each $A^{(k)}$, and the parameter ξ_2 is chosen to be $\xi/(m\|A\|_F)$. This yields a $(m\|A\|_F, \log n + \log m, \xi)$ -block-encoding of A with a single use of W, P_R and P_L^\dagger ; since P_R and P_L^\dagger only require $\mathcal{O}(\log m)$ gates each, and W takes $\mathcal{O}(\text{poly}(m, \log n, \log \frac{1}{\xi}))$ time, the proof is complete. \square

When symmetrizing the Newton linear system, some components of the coefficient matrix can be conveniently defined using Kronecker, and symmetric Kronecker products of the input matrices. We conclude this section by formalizing how one can block-encode $G \otimes_s K$ up to error ξ in time $\tilde{\mathcal{O}}_{\frac{n}{\xi}}(1)$, provided that the matrices G and K are stored in a QRAM data structure.

To do so in an efficient manner, we construct a block-encoding using the sparse-access input model; in particular we use this input model to construct block-encodings of the matrix V and its transpose V^\top which are used to define the symmetric tensor product of two matrices as given in Definition 2. In this setting, it is assumed that the matrix $\mathcal{M} \in \mathbb{C}^{M \times N}$ we wish to block-encode has s_r -sparse rows and s_c -sparse columns. Further, we can query the elements of \mathcal{M} using an oracle:

$$O_{\mathcal{M}} : |i\rangle |j\rangle |0\rangle^{\otimes p} \mapsto |i\rangle |j\rangle |m_{ij}\rangle, \quad \forall i \in [M], j \in [N].$$

Likewise, it is assumed that we have access to oracles O_r and O_c that are capable of querying the indices of non-zero elements of each row and column. This input model is the most suitable to construct V , since we can easily implement an efficient algorithmic description of its sparsity pattern and element values. The following result from [57] establishes how one can efficiently implement block-encodings of sparse-access matrices.

Lemma 1 (Lemma 48 in [57]). *Let $\mathcal{M} \in \mathbb{C}^{2^w \times 2^w}$ be a matrix that is s_r -row-sparse and s_c -column-sparse, and each element of \mathcal{M} has value at most 1. Suppose that we have access to the following sparse-access oracles acting on two $(w + 1)$ qubit registers:*

$$\begin{aligned} O_r : |i\rangle |k\rangle &\mapsto |i\rangle |r_{ij}\rangle \quad \forall i \in [2^w] - 1, k \in [s_r], \text{ and} \\ O_c : |l\rangle |j\rangle &\mapsto |c_{lj}\rangle |j\rangle \quad \forall l \in [s_c], j \in [2^w] - 1, \text{ where} \end{aligned}$$

r_{ij} is the index for the j -th non-zero entry of the i -th row of \mathcal{M} , or if there are less than i non-zero entries, then it is $j + 2^w$, and similarly c_{ij} is the index for the i -th non-zero entry of the j -th column of \mathcal{M} , or if there are less than j non-zero entries, then it is $i + 2^w$. Additionally, assume that we have access to an oracle $O_{\mathcal{M}}$ that returns the entries of \mathcal{M} in a binary description:

$$O_{\mathcal{M}} : |i\rangle |j\rangle |0\rangle^{\otimes p} \mapsto |i\rangle |j\rangle |m_{ij}\rangle, \quad \forall i, j \in [2^w] - 1,$$

where m_{ij} is a p -bit binary description of the ij -matrix element of \mathcal{M} . Then, we can implement a $(\sqrt{s_r s_c}, w + 3, \xi)$ -block-encoding of \mathcal{M} with a single use of O_r, O_c and two uses of $O_{\mathcal{M}}$, and additionally using

$$\mathcal{O} \left(w + \log^{2.5} \left(\frac{s_r s_c}{\xi} \right) \right) \text{ one and two qubit gates while using } \mathcal{O} \left(p, \log^{2.5} \left(\frac{s_r s_c}{\xi} \right) \right) \text{ ancilla qubits.}$$

The above result allows us to establish how to block-encode the Symmetric Kronecker product of two matrices.

Proposition 7. (Block-encoding symmetric Kronecker products) *Let $G, K \in \mathbb{R}^{n \times n}$, be stored in a QRAM data structure. Then we can construct a $(\|G \otimes K\|_F^2, \mathcal{O}(\log n), \xi)$ -block-encoding of $G \otimes_s K$ in time*

$$\mathcal{O} \left(\text{poly} \left(\log n, \log \frac{1}{\xi} \right) \right).$$

Proof. Recall that

$$G \otimes_s K = \frac{1}{2} V (G \otimes K + K \otimes G) V^\top,$$

where V is the $\frac{n(n+1)}{2} \times n^2$ matrix defined in Definition 2. Observe that by definition, V is 1-column-sparse and 2-row-sparse so $s_c s_r = 1 \cdot 2 = 2$.

Using the description of V , we can construct the sparse-access oracles O_r and O_c as defined in Lemma 1 (which act on two $(\log n^2 + 1)$ qubit registers). Additionally, from the definition of V we can construct an oracle O_V that returns the entries of V in a binary description:

$$O_V : |i\rangle |j\rangle |0\rangle^{\otimes p} \mapsto |i\rangle |j\rangle |v_{ij}\rangle, \quad \forall i, j \in [2^{\log n^2}] - 1,$$

where v_{ij} is a p -bit binary description of the ij -matrix element of V . By Lemma 1 we can efficiently implement a $(\sqrt{2}, \mathcal{O}(\log n), \xi_V)$ -block-encoding of V with a single use of O_r , O_c and two uses of O_V . We can construct a $(\sqrt{2}, \mathcal{O}(\log n), \xi_V)$ -block-encoding of V^\top in the same manner.

Given that G and K are stored in a QRAM data structure, we can construct a $(\|G \otimes K\|_F, \mathcal{O}(\log n), \xi_{G \otimes K})$ -block-encoding of $G \otimes K$ in time $\mathcal{O}(\text{poly}(\log n, \log \frac{1}{\xi_{G \otimes K}}))$, and similarly for $K \otimes G$, for some error parameters to be specified later. Indeed, note that a $(\|G \otimes K\|_F, \mathcal{O}(\log n), \xi_{G \otimes K})$ -block-encoding of $G \otimes K$ is trivial to construct: by definition of block-encoding, it suffices to take the tensor product of block-encodings for G and K , constructed using Proposition 1, keeping the ancilla bits separate and ensuring that the sum of errors is less than $\xi_{G \otimes K}$.

From here we use a linear combination of our block-encodings of $G \otimes K$ and $K \otimes G$ to block-encode $G \otimes K + K \otimes G$. Applying Proposition 2 with

$$\xi_{G \otimes K} = \frac{\xi_1}{2\|G \otimes K\|_F} \quad \text{and} \quad \xi_{K \otimes G} = \frac{\xi_1}{2\|G \otimes K\|_F},$$

yields a $(\|G \otimes K\|_F^2, \mathcal{O}(\log n), \xi_1)$ -block-encoding of $G \otimes K + K \otimes G$, and we can implement this block-encoding in time $\mathcal{O}(\text{poly}(\log n, \log \frac{1}{\xi_1}))$, i.e., $\tilde{\mathcal{O}}_{\frac{1}{\xi_1}}(1)$.

Using our block-encodings of V and $G \otimes K + K \otimes G$, we apply Proposition 3 with

$$\xi_V = \frac{\xi_2}{2\sqrt{2}} \quad \text{and} \quad \xi_1 = \frac{\xi_2}{4\|G \otimes K\|_F^2},$$

to implement a $(\sqrt{2}\|G \otimes K\|_F^2, \mathcal{O}(\log n), \xi_2)$ -block-encoding of $V(G \otimes K + K \otimes G)$ in time

$$\mathcal{O}\left(\text{poly}\left(\log n, \log \frac{1}{\xi_2}\right)\right).$$

A final application of Proposition 3 with

$$\xi_V = \frac{\xi}{2\sqrt{2}} \quad \text{and} \quad \xi_2 = \frac{\xi}{2\sqrt{2}\|G \otimes K\|_F^2},$$

yields a $(2\|G \otimes K\|_F^2, \mathcal{O}(\log n), \xi)$ block-encoding of

$$\frac{1}{2}V(G \otimes K + K \otimes G)V^\top = G \otimes_s K,$$

in time $\mathcal{O}(\text{poly}(\log n, \log \frac{1}{\xi}))$, which completes the proof. \square

2.3 Extracting the solution of a linear system: tomography algorithm

In each iteration of our QIPMs, we solve a linear system of equations (known as the Newton linear system) to obtain search directions to progress to the next iterate. Due to the fact that we solve this system using a QLSA, the solution we obtain is encoded in a quantum state, and a classical description is necessary to construct the linear system that arises in the subsequent iteration. Namely, in order to update the current solutions to the primal and dual SDOPs, X and (y, S) , we require a procedure to map the state proportional to the solution of the Newton linear system to a classical solution pair. There are many such procedures (e.g., [61, 62]); in this paper we can take advantage of the fact that we have access to a unitary preparing the quantum state, as well as its inverse.

To obtain an ℓ_2 -norm estimate of the quantum state, we rely on the following result.

Theorem 2 ([63]). Let $|\psi\rangle = \sum_{j=0}^{d-1} y_j |j\rangle$ be a quantum state, $y \in \mathbb{C}^d$ the vector with elements y_j , and $U|0\rangle = |\psi\rangle$. There is a quantum algorithm that, with probability at least $1 - \delta$, outputs $\tilde{y} \in \mathbb{R}^d$ such that $\|\Re(y) - \tilde{y}\|_2 \leq \varepsilon$ using $\mathcal{O}(\frac{d}{\varepsilon} \log \frac{d}{\delta})$ applications of U , $\tilde{\mathcal{O}}_{d, \frac{1}{\delta}, \frac{1}{\varepsilon}}(\frac{d}{\varepsilon})$ indexed-SWAP gates acting on d bits, and $\tilde{\mathcal{O}}_{d, \frac{1}{\delta}, \frac{1}{\varepsilon}}(\frac{d}{\varepsilon})$ additional gates. If we have access to a classical-write, quantum-read QRAM of size $\tilde{\mathcal{O}}_{d, \frac{1}{\delta}, \frac{1}{\varepsilon}}(2^{d/\varepsilon})$, we do not need the indexed-SWAP gates.

Proof. This follows from [63, Prop. 22], setting the ℓ_∞ -norm error to ε/\sqrt{d} . For the gate complexity, [63] uses a ‘‘QRAM-like’’ indexed-SWAP gate, which is used to efficiently implement the mapping from a binary description of a vector x to its amplitude encoding $|x\rangle$. It is straightforward to note that this mapping can also be stored in a classical-write, quantum-read QRAM: as the vectors $x \in [-1, 1]^d$ necessary for the construction are known in advance, and each coordinate requires precision $\mathcal{O}(1/\varepsilon)$, we can employ the usual QRAM data structure for each of these $\mathcal{O}(2^{d/\varepsilon})$ vectors in QRAM. This operation needs to be performed only once and it only depends on the dimension and the precision. Alternatively, the operation can be performed with a $\tilde{\mathcal{O}}_{d, \frac{1}{\delta}, \frac{1}{\varepsilon}}(\frac{d^2}{\varepsilon})$ quantum-writable QRAM (the data structure can be computed in superposition and written onto the QRAM). \square

It is shown in [63] that the above sample complexity is optimal, i.e., the number of applications of U cannot be reduced in general. For purposes of the QIPMs presented in this paper, we treat the algorithm from [63] as a black box oracle that performs our tomography steps. For simplicity, and since we are already assuming access to a classical-write, quantum-read QRAM, we use that input model, although as noted, other models are possible and potentially more efficient.

In light of the normalization of quantum states, in quantum state tomography it is assumed that the vector to be extracted (for our purposes, the solution to a linear system) has unit norm. The tomography error bound is relative, and hence, in order to ensure that we satisfy the absolute error bound for the unscaled vector, we need to divide ξ_k by the maximum norm of a solution vector, which we denote by ϱ . Setting $\varepsilon = \xi/\varrho$ and $T_U = T_{LS}$ (i.e, the time to prepare and solve our linear system using a quantum computer), Theorem 2 asserts that we can extract a classical estimate of a solution to the linear system with Euclidean norm error ξ in time

$$T_{TO}(T_{LS}, \xi) = \tilde{\mathcal{O}}_{d, \varrho, \frac{1}{\xi}}\left(\frac{d}{\xi/\varrho} T_{LS}\right) = \tilde{\mathcal{O}}_{d, \varrho, \frac{1}{\xi}}\left(\frac{d}{\xi} \varrho T_{LS}\right).$$

Note that previous work on QIPMs used the tomography subroutine in [31], which has quadratic dependence on the inverse precision, rather than linear; this improves our final running time, but it is not the only reason why we are able to give a faster QIPM than previously known. Indeed, as we describe in Section 6, the most impactful source of improvement is our development of an inexact-feasible QIPM, that overcomes the issues with feasibility present in other approaches.

3 Primal and Dual SDOs and the Central Path

Recall that the symmetric matrices $A^{(1)}, \dots, A^{(m)}$ are linearly independent by assumption, and $S \in \mathcal{S}^n$ is the slack matrix of the dual problem, as defined in (2). We denote the feasible sets of (1) and (2) by

$$\mathcal{P} = \left\{ X \in \mathcal{S}^n : A^{(i)} \bullet X = b_i, \forall i \in [m], X \succeq 0 \right\},$$

$$\mathcal{D} = \left\{ (y, S) \in \mathbb{R}^m \times \mathcal{S}^n : \sum_{i=1}^m y_i A^{(i)} + S = C, S \succeq 0 \right\}.$$

For a feasible IPM we must assume that a strictly feasible pair X and (y, S) with $X \succ 0$ and $S \succ 0$ exists, i.e., the interior point condition (IPC) is satisfied [64]. It is known that with the self-dual embedding model this condition may be assumed without loss of generality [64]. We define the sets of *interior feasible solutions* by

$$\mathcal{P}^0 = \{X \in \mathcal{P} : X \succ 0\},$$

$$\mathcal{D}^0 = \{(y, S) \in \mathcal{D} : S \succ 0\}.$$

With the IPC satisfied, it is guaranteed that the primal and dual optimal sets:

$$\begin{aligned}\mathcal{P}^* &= \{X \in \mathcal{P} : C \bullet X = z_P\}, \\ \mathcal{D}^* &= \{(y, S) \in \mathcal{D} : b^\top y = z_D\},\end{aligned}$$

are nonempty and bounded, with $z_P = z_D$, i.e., the duality gap is zero. In particular, this holds for all optimal solutions X^* and (y^*, S^*) satisfying

$$C \bullet X^* - b^\top y^* = X^* \bullet S^* = 0, \quad (3)$$

which implies $X^* S^* = S^* X^* = 0$ as X^* and S^* are symmetric positive semidefinite matrices. However, for primal-dual IPMs, the complementarity condition $XS = 0$ is perturbed to:

$$XS = \sigma \nu I, \quad (4)$$

where I is the $n \times n$ identity matrix, $\sigma \in [0, 1]$ is the centering parameter, and $\nu > 0$ is the so-called central path parameter, which is progressively reduced to zero in the course of the optimization algorithm.

For all $\nu > 0$, assuming the IPC and linear independence of the matrices $A^{(i)}$ for $i \in [m]$, the central path equation system

$$\begin{aligned}A^{(i)} \bullet X &= b_i \quad \forall i \in [m], \quad X \succ 0 \\ \sum_{i \in [m]} y_i A^{(i)} + S &= C, \quad S \succ 0 \\ XS &= \sigma \nu I,\end{aligned} \quad (5)$$

has a unique solution [3]. The set of solutions for all $\nu > 0$ gives the central path for the primal and dual SDOPs. In IPMs we aim to follow the central path as $\nu \rightarrow 0$, i.e., as we approach optimality. In fact, we simply seek to stay in a certain neighborhood of the central path.

A classical IPM, as outlined in Algorithm 1, begins with a strictly feasible primal-dual pair $(X_0, S_0) \in \mathcal{P}^0 \times \mathcal{D}^0$ for the primal and dual SDO problems (1) and (2). This pair has a duality gap of $X_0 \bullet S_0 = \nu_0 n$, and a distance to the central path of $d(X_0, S_0) \leq \gamma \nu_0$ for $\gamma \in (0, 1)$, where $\nu_0 = \frac{X_0 \bullet S_0}{n}$, for some appropriate distance metric $d(X, S)$. Without loss of generality, one can assume that $\nu_0 = \mathcal{O}(1)$. As we mentioned earlier, it is always possible obtain an interior feasible solution using the self-dual embedding model, and in particular, the analytic center of our space, i.e., $X_0 = S_0 = n^{-1}I$ is always strictly feasible in this setting.

In our work on the IF-QIPM, we consider the narrow (or, Frobenius) neighborhood

$$\mathcal{N}_F(\gamma) = \left\{ (X, y, S) \in \mathcal{P}^0 \times \mathcal{D}^0 : \left\| X^{1/2} S X^{1/2} - \nu I \right\|_F = \left[\sum_{i=1}^n (\lambda_i(XS) - \nu)^2 \right]^{1/2} \leq \gamma \nu \right\}.$$

Another popular alternative is the so called *negative infinity neighborhood* that is a *large neighborhood*, defined as

$$\mathcal{N}_\infty^-(1 - \gamma) = \{(X, y, S) \in \mathcal{P}^0 \times \mathcal{D}^0 : \lambda_{\min}(XS) \geq \gamma \nu\},$$

where $\gamma \in (0, 1)$. It is well known that classical feasible primal-dual path following IPMs that use the former neighborhood exhibit an iteration complexity of the order $\mathcal{O}(\sqrt{n} \log(1/\epsilon))$ whereas long-step IPMs, using the latter neighborhood, have iteration complexity $\mathcal{O}(n \log(1/\epsilon))$. However, in practice the long step algorithms tend to converge faster than the short step counterpart; in this paper we focus on the theoretical running time and therefore use the narrow (or Frobenius) neighborhood $\mathcal{N}_F(\gamma)$. Thus, we define our centrality measure at a point $(X, S) \in \mathcal{S}_+^n \times \mathcal{S}_+^n$, with $\nu = \frac{X \bullet S}{n}$, to be

$$d(X, S) = \left\| X^{1/2} S X^{1/2} - \nu I \right\|_F. \quad (6)$$

Then, in each iteration, we solve the so-called Newton linear system in order to obtain ΔX and ΔS and update the solutions using the following rule:

$$\begin{aligned}X &= X + \Delta X, \\ S &= S + \Delta S.\end{aligned}$$

If we were to directly linearize the complementarity condition, the Newton linear system would then be written as:

$$\begin{aligned}\Delta XS + \Delta SX &= \sigma\nu I - XS, \\ \Delta X &\in \text{Null}(A^{(1)}, A^{(2)}, \dots, A^{(m)}), \\ \Delta S &\in \mathcal{R}(A^{(1)}, A^{(2)}, \dots, A^{(m)}),\end{aligned}\tag{7}$$

where $\text{Null}(\cdot)$, $\mathcal{R}(\cdot)$ are the nullspace (or, *kernel*) and rowspace of the constraint matrices, respectively (we provide explicit definitions for these subspaces in the following sections). However, it is known that (7) does not have a symmetric matrix solution with respect to ΔX , see, e.g., [19]. This must be addressed, because both the primal and dual solutions must be symmetric.

Algorithm 1 Classical interior point method

Input: $\epsilon, \delta > 0$; $\sigma = 1 - \delta/\sqrt{n}$; $\gamma \in (0, 1)$

Choose $(X_0, y_0, S_0) \in \mathcal{N}_F(\gamma)$

Choose a nonsingular matrix P from Monteiro-Zhang family of search directions, and compute P_0

Set $\nu_0 \leftarrow \frac{X_0 \bullet S_0}{n}$, $k \leftarrow 0$

while $\nu > \epsilon$ **do**

1. $\nu_k \leftarrow \frac{X_k \bullet S_k}{n}$

2. Compute the scaling matrix P_k

3. Compute the solution $(\Delta X_k, \Delta y_k, \Delta S_k)$ to the Newton linear system (10) defined by choice of P_k

$$\begin{aligned}X_{k+1} &\leftarrow X_k + \Delta X_k, \quad S_{k+1} \leftarrow S_k + \Delta S_k \quad \text{and} \quad y_{k+1} \leftarrow y_k + \Delta y_k \\ k &\leftarrow k + 1\end{aligned}$$

end

3.1 Symmetrizing the Newton System

As has been extensively studied in the literature of IPMs (see, e.g., [19, 24, 25]), the Newton steps ΔS and ΔX have to be symmetric matrices. More specifically, even though ΔS is guaranteed to be symmetric by the definition of a dual feasible solution, there is no symmetric ΔX that solves system (7). Thus, we need a different approach to guarantee symmetry.

In an effort to generalize the scaling methods required for primal-dual symmetry, following Zhang [39], we define the linear transformation $H_P(M)$ that symmetrizes a matrix M using a given invertible matrix P :

$$H_P(M) = \frac{1}{2} [PMP^{-1} + P^{-\top}M^{\top}P^{\top}].\tag{8}$$

Observe that $H_P(M) = \sigma\nu I \iff M = \sigma\nu I$; we can then write the central path equations in symmetric form as: $H_P(XS) = \sigma\nu I$. In this light, we can also symmetrize the linearized complementarity condition of the Newton system $X\Delta S + S\Delta X = \sigma\nu I - XS$ as

$$H_P(X\Delta S + S\Delta X) = \sigma\nu I - H_P(XS).\tag{9}$$

Equation (9) can be explicitly expressed as:

$$P(X\Delta S + S\Delta X)P^{-1} + P^{-\top}(\Delta SX + S\Delta X)P^{\top} = 2\sigma\nu I - PXS P^{-1} - P^{-\top}SXP^{\top}.$$

Defining $\mathcal{A}^{\top} = [\text{vec}(A^{(1)})\text{vec}(A^{(2)}) \cdots \text{vec}(A^{(m)})]$, we can write the symmetrized Newton linear system as

$$\begin{pmatrix} 0 & \mathcal{A}^{\top} & I_{n^2} \\ \mathcal{A} & 0 & 0 \\ \mathcal{E} & 0 & \mathcal{F} \end{pmatrix} \begin{pmatrix} \text{vec}(\Delta X) \\ \Delta y \\ \text{vec}(\Delta S) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \text{vec}(R^c) \end{pmatrix},$$

where I_{n^2} is the identity matrix of order n^2 and

$$\begin{aligned}\mathcal{E} &= P \otimes SP^{-1} + P^{-1}S \otimes P, \\ \mathcal{F} &= PX \otimes P^{-1} + P^{-1} \otimes XP, \\ R^c &= \sigma\nu I - H_P(XS).\end{aligned}$$

Now let us define $\mathcal{A}_s^\top = [\mathbf{svec}(A^{(1)})\mathbf{svec}(A^{(2)}) \cdots \mathbf{svec}(A^{(m)})]$. Following Todd et al. [65], we can also write the Newton linear system using the \mathbf{svec} notation, in which case we have:

$$\begin{pmatrix} 0 & \mathcal{A}_s & \mathcal{I} \\ \mathcal{A}_s^\top & 0 & 0 \\ \mathcal{E}_s & 0 & \mathcal{F}_s \end{pmatrix} \begin{pmatrix} \mathbf{svec}(\Delta X) \\ \Delta y \\ \mathbf{svec}(\Delta S) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ R_s^c \end{pmatrix}, \quad (10)$$

where \mathcal{I} is the identity matrix of order $\frac{n(n+1)}{2}$ and

$$\begin{aligned}\mathcal{E}_s &= P \otimes_s P^{-\top} S, \\ \mathcal{F}_s &= PX \otimes_s P^{-\top}, \\ R_s^c &= \mathbf{svec}(\sigma\nu I - H_P(XS)) = \mathbf{svec}(R^c).\end{aligned}$$

One can note that matrices \mathcal{E}_s and \mathcal{F}_s are nonsingular whenever X and S are positive definite matrices [65]. We have:

$$\mathcal{E}_s = P \otimes_s P^{-\top} S = (I \otimes_s P^{-\top} S P^{-1})(P \otimes_s P), \quad (11a)$$

$$\mathcal{F}_s = PX \otimes_s P^{-\top} = (PXP^\top \otimes_s I)(P^{-\top} \otimes_s P^{-\top}), \quad (11b)$$

where the equalities follow from the definition of the symmetric Kronecker product detailed earlier.

We then need to choose P to obtain specific instances of this system, and we refer to the class of search directions parameterized by P as the Monteiro-Zhang family of search directions [38]. We present results involving three possible choice of P : $P = I$, the AHO direction [19]; $P = S^{1/2}$, the HKM direction [23, 36, 48]; and $P = W^{-1/2}$, where W is the Nesterov-Todd [25] scaling matrix, defined as:

$$W = S^{-1/2}(S^{1/2}XS^{1/2})^{1/2}S^{-1/2} = X^{1/2}(X^{1/2}SX^{1/2})^{-1/2}X^{1/2}. \quad (12)$$

To define the HKM Newton linear system for the k -th iteration, we use symmetry to drop all of the transpose terms involving P and set:

$$\mathcal{E}_{s_k} = S_k^{1/2} \otimes_s S_k^{-1/2} S_k = S_k^{1/2} \otimes_s S_k^{1/2}, \quad (13a)$$

$$\mathcal{F}_{s_k} = S_k^{1/2} X_k \otimes_s S_k^{-1/2}, \quad (13b)$$

$$R_{s_k}^c = \mathbf{svec}(2\sigma\nu_k I - S_k^{1/2} X_k S_k^{1/2}). \quad (13c)$$

In defining the AHO Newton Linear system, taking $P = I$ simplifies the above quantities to

$$\mathcal{E}_{s_k} = I \otimes_s S_k,$$

$$\mathcal{F}_{s_k} = X_k \otimes_s I,$$

$$R_{s_k}^c = \mathbf{svec}(2\sigma\nu I - X_k S_k - S_k X_k).$$

Although this guarantees that our solutions to the Newton system are symmetric, the AHO direction only guarantees a solution if we are within a small (local) neighborhood of the central path: the ∞ -norm neighborhood with opening less than $1/3$. Finally, for the Nesterov-Todd Newton system, we set $P = W^{-1/2}$ and we can write:

$$\mathcal{E}_{s_k} = W_k^{-1/2} \otimes_s W_k^{1/2} S_k, \quad (14a)$$

$$\mathcal{F}_{s_k} = W_k^{-1/2} X_k \otimes_s W_k^{1/2}, \quad (14b)$$

$$R_{s_k}^c = \mathbf{svec} \left(2\sigma\nu I - W_k^{-1/2} X_k S_k W_k^{1/2} - W_k^{1/2} S_k X_k W_k^{-1/2} \right). \quad (14c)$$

Symmetry is not the only issue overlooked by previous works on QIPMs; we still need to account for the consequences of inexact tomography in the context of QIPMs.

4 Newton Linear Systems for QIPMs

We now turn our attention to solving the Newton linear system using quantum linear system solvers. We present two different Newton linear systems that account for the inexactness in the search directions resulting from the use of tomography. The first system, based on the infeasible central path and its neighborhood, allows us to quantize an Inexact-Infeasible IPM. However, in subsequent sections we will show that an algorithm based on this direct methodology leads to an algorithm with a poor overall running time. Thus, we subsequently introduce a new methodology that allows us to design an inexact, but feasible QIPM framework which yields a speedup in n over the classical variants.

4.1 An Infeasible Central Path and its Neighborhood

Our first effort to deal with the tomography errors is to quantize an Inexact-Infeasible IPM framework. That is, we work with a solution to a Newton linear system in which the right hand side includes residuals to capture (i) infeasibility of the iterate sequence and (ii) inexactness of the search direction. In this section we describe the infeasible central path, and the resulting Newton linear system.

To illustrate the issue at hand precisely, previous works on QIPMs [31–33] (if we apply symmetrization to their Newton systems) assume we can exactly solve the Newton linear system (10). Yet, solving the system (10) with a quantum computer introduces errors due to our use of QLSA and tomography. As a result, a classical estimate $(\Delta X, \Delta y, \Delta S)$ of the quantum state $|\Delta X \circ \Delta y \circ \Delta S\rangle$ will indeed only satisfy

$$\begin{pmatrix} 0 & \mathcal{A}^\top & I_{n^2} \\ \mathcal{A} & 0 & 0 \\ \mathcal{E} & 0 & \mathcal{F} \end{pmatrix} \begin{pmatrix} \text{vec}(\Delta X) \\ \Delta y \\ \text{vec}(\Delta S) \end{pmatrix} = \begin{pmatrix} \xi_d \\ \xi_p \\ R^c + \xi_c \end{pmatrix},$$

where (ξ_d, ξ_p, ξ_c) are the errors to which we satisfy primal feasibility, dual feasibility and the complementarity condition, respectively.

The feasible IPM framework, upon which the QIPMs in [31, 33] are built, therefore cannot be applied (even if these works had properly accounted for symmetry). The analysis of feasible primal-dual IPMs relies on the fact that $\Delta X \bullet \Delta S = 0$, this property is used throughout the classical analyses. However, the presence of the errors ξ_d and ξ_p means the primal and dual search directions ΔX and ΔS are not guaranteed to be in orthogonal subspaces. While this alone implies that the QIPMs presently in the literature are not convergent, one also has to account for the inexactness to which we solve the complementarity condition.

Let (X_0, y_0, S_0) be an initial point, not necessarily feasible, where for $\rho > 0$ we have

$$X_0 = S_0 = \rho I. \quad (15)$$

Following [47], it is assumed that ρ is specified such that for a primal-dual optimal solution (X^*, y^*, S^*) , and for some $0 < \gamma_1 < 1$, we have

$$(1 - \gamma_1)X_0 \succeq X^*, \quad (1 - \gamma_1)S_0 \succeq S^* \quad (16a)$$

$$\rho \geq \frac{1}{n} (\text{tr}(X^*) + \text{tr}(S^*)). \quad (16b)$$

Letting $b^\top = [b_1, b_2, \dots, b_m]$, consider the quantities:

$$\mu_0 = \frac{X_0 \bullet S_0}{n} = \rho^2, \quad R_0^p = \text{Avec}(X_0) - b, \quad R_0^d = \mathcal{A}^\top y_0 + \text{vec}(S_0) - \text{vec}(C).$$

Then, for $\tau \in (0, 1]$, the *infeasible* central path equations are given by:

$$\begin{pmatrix} \mathcal{A}^\top y + \text{vec}(S) - \text{vec}(C) \\ \text{Avec}(X) - b \\ XS \end{pmatrix} = \begin{pmatrix} \tau R_0^d \\ \tau R_0^p \\ \tau \mu_0 I \end{pmatrix}, \quad X, S \succ 0. \quad (17)$$

These equations define a set of infeasible centers, that approaches the solution of (1)-(2) as $\tau \rightarrow 0$. Let $\gamma_2 \in (0, 1)$; we can define a neighborhood of the infeasible central path, which we denote \mathcal{N}_I , as:

$$\mathcal{N}_I = \left\{ \begin{array}{l} (\tau, \theta, X, y, S) \in \mathbb{R}_+ \times \mathbb{R}_+ \times \mathcal{S}_+^n \times \mathbb{R}^m \times \mathcal{S}_+^n : \tau \leq \theta \\ \mathcal{A}^\top y + \mathbf{vec}(S) - \mathbf{vec}(C) = \tau(R_0^d + \zeta^d), \|\zeta^d\| \leq \gamma_1 \rho \\ \mathcal{A}\mathbf{vec}(X) - b = \tau(R_0^p + \zeta^p), \|\mathcal{A}^+ \zeta^p\| \leq \gamma_1 \rho \\ \|H_p(XS) - \theta \mu_0 I\| \leq \gamma_2 \theta \mu_0 \end{array} \right\} \quad (18)$$

where $\mathcal{A}^+ = \mathcal{A}^\top (\mathcal{A}\mathcal{A}^\top)^{-1}$ is the right pseudoinverse of \mathcal{A} .

The following results from Zhou and Toh [47] are crucial for the complexity analysis of the II-QIPM later in the paper; they correspond to when the HKM search direction is chosen, which is the one for which we give a full analysis of the method.²

Lemma 2 (Lemma 1 in [47]). *For any r^p and r^d satisfying $\|\mathcal{A}^+ r^p\| \leq \gamma_1 \rho$ and $\|r^d\| \leq \gamma_1 \rho$, there exists $(\tilde{X}, \tilde{y}, \tilde{S})$ that satisfies the following conditions:*

$$\mathcal{A}^\top \tilde{y} + \mathbf{vec}(\tilde{S}) - \mathbf{vec}(C) = R_0^d + r^d \quad (19a)$$

$$\mathcal{A}^\top \mathbf{vec}(\tilde{X}) - b = R_0^p + r^p \quad (19b)$$

$$(1 - \gamma_1)\rho I \preceq \tilde{X} \preceq (1 + \gamma_1)\rho I \quad (19c)$$

$$(1 - \gamma_1)\rho I \preceq \tilde{S} \preceq (1 + \gamma_1)\rho I. \quad (19d)$$

Lemma 3 (Lemma 2 in [47]). *If the conditions (15), (16a) and (16b) hold, then for any $(\tau, \theta, X, y, S) \in \mathcal{N}_I$ with $\theta \in (0, 1]$, we have*

$$\tau \operatorname{tr}(X) = \mathcal{O}(\theta \rho n) \text{ and } \tau \operatorname{tr}(S) = \mathcal{O}(\theta \rho n).$$

Implications of the above lemma are summarized below.

Remark 1 ([47]). *Let $\mathcal{B} = \{(X, y, S) : (\tau, \theta, X, y, S) \in \mathcal{N}_I, \theta \in (0, 1], \tau = \theta\}$. Then, by Lemma 3, the set \mathcal{B} is bounded, because for $(X, y, S) \in \mathcal{B}$ we have $\|X\| \leq \operatorname{tr}(X) \leq \mathcal{O}(\rho n)$ and $\|S\| \leq \operatorname{tr}(S) \leq \mathcal{O}(\rho n)$.*

From the above, one can derive Frobenius norm bounds on the solutions X and S by noting the relationship $\|G\|_F \leq \sqrt{n}\|G\|$ for $G \in \mathbb{R}^{n \times n}$, i.e.,

$$\|X\|_F, \|S\|_F = \mathcal{O}(\rho n^{1.5}). \quad (20)$$

We do however note that the bounds on the operator norms and trace of the solutions

$$\|X\| \leq \operatorname{tr}(X) \leq \mathcal{O}(\rho n) \text{ and } \|S\| \leq \operatorname{tr}(S) \leq \mathcal{O}(\rho n)$$

given in the above remarks are *only relevant* to II-IPMs. They are properties associated with the infeasible central path, and our discussion for the IF-QIPM will not use these bounds.

Remark 2 ([47]). *Suppose we generate a sequence $\{(\tau_k, \theta_k, X_k, y_k, S_k)\}$ in the neighborhood \mathcal{N}_I with the properties that $\theta_k \geq \tau_k \forall k$ and $1 = \theta_0 \geq \theta_k \geq \theta_{k+1} \geq 0$. If $\theta_k \rightarrow 0$ as $k \rightarrow \infty$, then any limit point of the sequence $\{(X_k, y_k, S_k)\}$ is a solution of*

$$\begin{pmatrix} \mathcal{A}^\top y + \mathbf{vec}(S) - \mathbf{vec}(C) \\ \mathcal{A}\mathbf{vec}(X) - b \\ XS \end{pmatrix} = 0, \quad X, S \succeq 0.$$

If $\tau_k = \theta_k$, then the sequence $\{(X_k, S_k)\}$ is also bounded.

²For the other search directions, we will limit ourselves to discussing the running time, without giving all the details.

4.1.1 Inexact Search Directions

Let $\eta_1 \in (0, 1]$, $\eta_2 \in (0, 1)$ with $\eta_1 \geq \eta_2$ and $\tau_0, \theta_0 = 1$. Given a point $(\tau_k, \theta_k, X_k, y_k, S_k) \in \mathcal{N}_I$, we try to generate a new point

$$(\tau_{k+1}, \theta_{k+1}, X_{k+1}, y_{k+1}, S_{k+1}) \in \mathcal{N}_I$$

by computing an update along the search direction $(\Delta X_k, \Delta y_k, \Delta S_k)$ for the solution X, y, S . This naturally leads to an iterative algorithm. We choose the search direction to be a solution of the following system of equations:

$$\begin{pmatrix} 0 & \mathcal{A}^\top & I_{n^2} \\ \mathcal{A} & 0 & 0 \\ \mathcal{E}_k & 0 & \mathcal{F}_k \end{pmatrix} \begin{pmatrix} \text{vec}(\Delta X_k) \\ \Delta y_k \\ \text{vec}(\Delta S_k) \end{pmatrix} = \begin{pmatrix} -\eta_1 R_k^d \\ -\eta_1 R_k^p \\ \text{vec}(R_k^c) \end{pmatrix}, \quad (21)$$

where

$$R_k^c = \text{vec}((1 - \eta_2)\tau_k\mu_0 I - H_P(X_k S_k)).$$

As previously discussed, in the quantum setting both the QLSA and tomography necessarily introduce errors in the solution of (21). We therefore consider an inexact search direction $(\Delta X_k, \Delta y_k, \Delta S_k)$, defined next, following [47]. Let $\{\vartheta_i\}_{i=0}^\infty$ be a sequence of reals in $(0, 1]$, such that

$$\bar{\vartheta} = \sum_{i=0}^{\infty} \vartheta_i < \infty.$$

We say that $(\Delta X_k, \Delta y_k, \Delta S_k)$ is an *inexact search direction* at the k -th iteration if it solves the linear system:

$$\begin{pmatrix} 0 & \mathcal{A}^\top & I_{n^2} \\ \mathcal{A} & 0 & 0 \\ \mathcal{E}_k & 0 & \mathcal{F}_k \end{pmatrix} \begin{pmatrix} \text{vec}(\Delta X_k) \\ \Delta y_k \\ \text{vec}(\Delta S_k) \end{pmatrix} = \begin{pmatrix} -\eta_1(R_k^d + r_k^d) \\ -\eta_1(R_k^p + r_k^p) \\ \text{vec}(R_k^c) + \text{vec}(r_k^c) \end{pmatrix}, \quad (22)$$

where $r_k^c \in \mathcal{S}^n$ and the *residual terms* satisfy:

$$\|r_k^p\| \leq \hat{\vartheta}\gamma_1\rho\tau_k\vartheta_k, \quad \|r_k^d\| \leq \gamma_1\rho\tau_k\vartheta_k, \quad \|r_k^c\| \leq \frac{1}{2}(1 - \eta_2)\gamma_2\theta_k\mu_0, \quad (23)$$

with $0 < \hat{\vartheta} \leq \sigma_{\min}(\mathcal{A})$. Upon obtaining a classical estimate of the solution to (22) using QLSA and tomography, it is possible that the complementarity error r^c is not symmetric (as we have no safeguards to ensure that the tomography errors are symmetric). Therefore, in order to ensure that we always have $r^c \in \mathcal{S}^n$, we take one further precaution and utilize the $\text{svec}(\cdot)$ characterization of the Newton system (22). Let

$$\begin{aligned} R_{s_k}^p &= \mathcal{A}_s \text{svec}(X_k) - b, \\ R_{s_k}^d &= \mathcal{A}_s^\top y_k + \text{svec}(S_k) - \text{svec}(C). \end{aligned}$$

Then, in each iteration of the II-QIPM we use the HKM direction with $P = S^{1/2}$, and solve

$$\begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ \mathcal{E}_{s_k} & 0 & \mathcal{F}_{s_k} \end{pmatrix} \begin{pmatrix} \text{svec}(\Delta X_k) \\ \Delta y_k \\ \text{svec}(\Delta S_k) \end{pmatrix} = \begin{pmatrix} -\eta_1(R_{s_k}^d + r_k^d) \\ -\eta_1(R_{s_k}^p + r_k^p) \\ R_{s_k}^c + \text{svec}(r_k^c) \end{pmatrix}, \quad (24)$$

where \mathcal{E}_{s_k} and \mathcal{F}_{s_k} are defined according to (13a)-(13b), and

$$R_{s_k}^c = \text{svec}\left(2(1 - \eta_2)\tau_k\mu_0 I - S_k^{1/2} X_k S_k^{1/2}\right).$$

We can now give a full description of the classical Inexact-Infeasible IPM, as given in [47], in Algorithm 2. This algorithm converges in polynomial time provided that (23) and (24) hold; this will be discussed in Section 6.

Algorithm 2 Inexact-Infeasible Interior Point Method.

Input: $\tau_0, \theta_0 = 1$; $\epsilon > 0$; $\eta_1 \in (0, 1]$; $\gamma_1, \gamma_2, \eta_2 \in (0, 1)$

Pick a sequence $\{\vartheta_i\}_{i=0}^{\infty}$ in $(0, 1]$, such that $\vartheta = \sum_{i=0}^{\infty} \vartheta_i < \infty$

Choose a nonsingular matrix P from Monteiro-Zhang family of search directions

Find an initial point $(\tau_0, \theta_0, X_0, y_0, S_0) \in \mathcal{N}_I$

Store $(\tau_0, \theta_0, X_0, y_0, S_0)$ in QRAM

while $\tau > \epsilon$ **do**

1. Solve the system (24) to obtain inexact search direction $(\Delta X_k, \Delta y_k, \Delta S_k)$

2. Set

$$\alpha_k = \max \left\{ \alpha : \alpha \in \left[0, \min \left(1, \frac{1}{\eta_1(1+\vartheta)} \right) \right], (\tau_k(\alpha), \theta_k(\alpha), X_k(\alpha), y_k(\alpha), S_k(\alpha)) \in \mathcal{N}_I \right\}$$

$$X_{k+1} \leftarrow X_k + \alpha_k \Delta X_k, S_{k+1} \leftarrow S_k + \alpha_k \Delta S_k \text{ and } y_{k+1} \leftarrow y_k + \alpha_k \Delta y_k$$

$$\tau_{k+1} \leftarrow (1 - \alpha \eta_1) \tau_k, \text{ and } \theta_{k+1} \leftarrow (1 - \alpha \eta_1) \theta_k$$

$$k \leftarrow k + 1$$

end

4.2 The Nullspace Representation of the Newton Linear System

Analogous to Gondzio's [46] IF-IPM for linearly-constrained quadratic optimization (LCQO), in our novel IF-QIPM approach the first two equations of (10) will hold, while we assume that the third equation is only satisfied up to some residual error $R^r \in \mathcal{S}^n$, which in our setting is due to the use of quantum state tomography (subsequent discussion will show that we can guarantee the residual is symmetric):

$$\begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ \mathcal{E}_s & 0 & \mathcal{F}_s \end{pmatrix} \begin{pmatrix} \text{svec}(\Delta X) \\ \Delta y \\ \text{svec}(\Delta S) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ R_s^c + \text{svec}(R^r) \end{pmatrix}, \quad (25)$$

where $R_s^c = \text{svec}(\sigma \nu I - H_P(XS)) = \text{svec}(R^c)$. As is standard in the inexact Newton method literature [46, 66, 67], we assume that the residual term R^r , for some $\beta \in (0, 1)$, satisfies

$$\|R^r\|_F \leq \beta \|R^c\|_F. \quad (\text{AR1})$$

In this way, we ensure that the residual norms are being driven towards zero as we approach optimality, due to the fact that $\|R^c\|_F \rightarrow 0$ as $\nu \rightarrow 0$. Additionally, choosing the level of error in this manner is not very restrictive, as the algorithm terminates upon reaching a solution satisfying $\nu \leq \epsilon$.

The ensuing result from [65] establishes the uniqueness of the solution to the system (25).

Theorem 3 (Theorem 3.1 in [65]). *Suppose X and S are positive definite. Then the system of equations (25) has a unique solution $(\Delta X, \Delta y, \Delta S) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ if $\mathcal{E}_s^{-1} \mathcal{F}_s$ is positive definite (not necessarily symmetric). In particular, this holds when X, S and $H_P(XS)$ are positive definite.*

Proof. The proof is the same as the one given in [65], but we repeat it here for completeness. In what follows, positive definiteness does not imply symmetry. We seek to show that the system

$$\begin{pmatrix} 0 & \mathcal{A}_s & 0 \\ \mathcal{A}_s^\top & 0 & \mathcal{I} \\ 0 & \mathcal{E}_s & \mathcal{F}_s \end{pmatrix} \begin{pmatrix} \Delta y \\ \text{svec}(\Delta X) \\ \text{svec}(\Delta S) \end{pmatrix} = 0 \quad (26)$$

is satisfied only by the all-zero solution. Note that for convenience, we have reordered the block-rows of the Newton system in order to utilize block-Gaussian elimination for the purposes of this proof. From (11a) it follows that \mathcal{E}_s is invertible. Hence, applying block-Gaussian elimination to (26) yields the Schur complement equation:

$$(\mathcal{A}_s \mathcal{E}_s^{-1} \mathcal{F}_s \mathcal{A}_s^\top) \Delta y = 0.$$

By assumption, $\mathcal{E}_s^{-1}\mathcal{F}_s \succ 0$ and \mathcal{A}_s has full row rank, implying that the matrix $\mathcal{A}_s\mathcal{E}_s^{-1}\mathcal{F}_s\mathcal{A}_s^\top$ is positive definite, therefore $\Delta y = 0$. Thus the second equation in (26) gives $\Delta S = -\mathbf{smat}(\mathcal{A}_s^\top \Delta y) = 0$ and from the third equation it follows that $\Delta X = -\mathcal{E}_s^{-1}\mathcal{F}_s\Delta S = 0$. This completes the proof. \square

If the Newton linear system as given by (25) is solved using QLSA and tomography, the first two block equations in this system will not be satisfied exactly, either. To design our IF-QIPM, we re-write the optimality conditions (10) as:

$$\begin{aligned} \Delta X &\in \text{Null}(\mathcal{A}_s) \\ \Delta S &\in \mathcal{R}(\mathcal{A}_s) \\ (P \otimes_s P^{-\top} S)\mathbf{svec}(\Delta X) + (PX \otimes_s P^{-\top})\mathbf{svec}(\Delta S) &= \mathbf{svec}(\sigma\nu I - H_P(XS)) \end{aligned} \quad (27)$$

where $\text{Null}(\mathcal{A}_s)$ denotes the nullspace of \mathcal{A}_s and $\mathcal{R}(\mathcal{A}_s)$ denotes the rowspace of \mathcal{A}_s . Crucially, $\text{Null}(\mathcal{A}_s)$ and $\mathcal{R}(\mathcal{A}_s)$ are orthogonal subspaces to one another; we always have $\Delta X \bullet \Delta S = 0$ for any $\Delta X \in \text{Null}(\mathcal{A}_s)$ and $\Delta S \in \mathcal{R}(\mathcal{A}_s)$.

For $\mathcal{R}(\mathcal{A}_s)$, we choose

$$\{\mathbf{svec}(A^{(1)}), \mathbf{svec}(A^{(2)}), \dots, \mathbf{svec}(A^{(m)})\}$$

as a basis. If the primal-dual pair (1)-(2) is given in canonical form (inequalities instead of equalities), we can trivially obtain bases for the nullspace and rowspace directly from the coefficient matrix. Alternatively, one can recast the SDOP as a slightly larger one using the Self-Dual embedding model, for which the nullspace basis matrix can be constructed trivially from the coefficient matrix (i.e., without needing factorization or Gaussian elimination). We can also calculate a basis of $\text{Null}(\mathcal{A}_s)$ using either Gaussian elimination or the QR-Factorization. Recall that $\mathcal{A}_s \in \mathbb{R}^{m \times \frac{n(n+1)}{2}}$, so $\mathcal{A}_s^\top \in \mathbb{R}^{\frac{n(n+1)}{2} \times m}$. The QR factorization of \mathcal{A}_s^\top is defined as:

$$\mathcal{A}_s^\top = [Q_1 \quad Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix},$$

where $Q_1 \in \mathbb{R}^{\frac{n(n+1)}{2} \times m}$ and $Q_2 \in \mathbb{R}^{\frac{n(n+1)}{2} \times (\frac{n(n+1)}{2} - m)}$. Then, it follows that the columns of Q_2 form a basis for the null space of \mathcal{A}_s . Hence, introducing a variable $\Delta z \in \mathbb{R}^{(\frac{n(n+1)}{2} - m)}$, ΔX and ΔS can be written as

$$\mathbf{svec}(\Delta X) = Q_2 \Delta z, \quad (28a)$$

$$\mathbf{svec}(\Delta S) = -\mathcal{A}_s^\top \Delta y. \quad (28b)$$

The following result defines the Newton linear system to be solved at each iteration of the IF-QIPM.

Proposition 8. *Let P be a nonsingular scaling matrix from the Monteiro-Zhang family of search directions, and define*

$$\begin{aligned} \mathcal{E}_s &= (P \otimes_s P^{-\top} S), \\ \mathcal{F}_s &= (PX \otimes_s P^{-\top}), \\ R_s^c &= \mathbf{svec}(\sigma\nu I - H_P(XS)). \end{aligned}$$

The Newton linear system for the IF-QIPM is given by

$$[\mathcal{E}_s Q_2 \quad \mathcal{F}_s (-\mathcal{A}_s^\top)] \begin{bmatrix} \Delta z \\ \Delta y \end{bmatrix} = R_s^c. \quad (29)$$

Proof. Substituting equations (28a) and (28b) into the left-hand side of the third equation in system (27) yields

$$\begin{aligned} (P \otimes_s P^{-\top} S)\mathbf{svec}(\Delta X) + (PX \otimes_s P^{-\top})\mathbf{svec}(\Delta S) &= (P \otimes_s P^{-\top} S)(Q_2 \Delta z) + (PX \otimes_s P^{-\top})(-\mathcal{A}_s^\top \Delta y) \\ &= [\mathcal{E}_s Q_2 \quad \mathcal{F}_s (-\mathcal{A}_s^\top)] \begin{bmatrix} \Delta z \\ \Delta y \end{bmatrix}. \end{aligned}$$

\square

Observe that the coefficient matrix of equation (29) is full rank, as the columns of A_s^\top and Q_2 are orthogonal, *and* the columns of A_s^\top and Q_2 are linearly independent. Thus, equation (29) has a unique solution. Upon using QLSA to solve the system (29) for $|\Delta z \circ \Delta y\rangle$, we will need to map classical estimates $(\overline{\Delta z}, \overline{\Delta y})$ of the solution to a classical estimate $(\overline{\Delta X}, \overline{\Delta y}, \overline{\Delta S})$ in order to proceed to the next iteration. Define ΔX and ΔS by (28a) and (28b), respectively. As $(\Delta z, \Delta y)$ is the unique solution of (29), consequently ΔX and ΔS are uniquely determined as well.

Our use of inexact tomography causes Δz and Δy to be calculated with some error. Yet, by using the bases of the nullspace and the row space to calculate ΔX and ΔS , we ensure that these search directions are always coming from orthogonal subspaces, regardless of the errors introduced by tomography or the length of the stepsize. We formalize this result in the following proposition.

Proposition 9. *For a given $X \in \mathcal{P}^0$ and $(y, S) \in \mathcal{D}^0$, let $(\overline{\Delta z}, \overline{\Delta y})$ be an estimate of the solution to (29). Then,*

$$\overline{\Delta z} = \Delta z + \xi_z \text{ and } \overline{\Delta y} = \Delta y + \xi_y,$$

where ξ_z and ξ_y denote the error terms of system (29). Let $\overline{\Delta X}$ and $\overline{\Delta S}$ be given by (28a) and (28b), respectively. Then for any step size α , we have

$$\mathcal{A}_s(\text{svec}(X) + \alpha \text{svec}(\overline{\Delta X})) = b, \text{ and } \mathcal{A}_s^\top(y + \alpha \overline{\Delta y}) + (\text{svec}(S) + \alpha \text{svec}(\overline{\Delta S})) = \text{svec}(C).$$

Proof. First, note that for any $X \in \mathcal{P}$ and $(y, S) \in \mathcal{D}$, we have

$$\begin{aligned} \mathcal{A}_s \text{svec}(X) &= b \\ \mathcal{A}_s^\top y + \text{svec}(S) &= \text{svec}(C). \end{aligned}$$

Therefore,

$$\begin{aligned} \mathcal{A}_s(\text{svec}(X) + \alpha \text{svec}(\overline{\Delta X})) &= \mathcal{A}_s \text{svec}(X) + \alpha \mathcal{A}_s \text{svec}(\overline{\Delta X}) \\ &= b + \alpha \mathcal{A}_s[Q_2(\overline{\Delta z})] \\ &= b + \alpha[\mathcal{A}_s Q_2(\Delta z + \xi_z)] \\ &= b + \alpha[0 \cdot (\Delta z + \xi_z)] \\ &= b. \end{aligned}$$

Similarly,

$$\begin{aligned} \mathcal{A}_s^\top(y + \alpha \overline{\Delta y}) + (\text{svec}(S) + \alpha \text{svec}(\overline{\Delta S})) &= [\mathcal{A}_s^\top y + \text{svec}(S)] + [\mathcal{A}_s^\top \alpha \overline{\Delta y} + \alpha \text{svec}(\overline{\Delta S})] \\ &= \text{svec}(C) + \alpha[\mathcal{A}_s^\top \overline{\Delta y} + \text{svec}(\overline{\Delta S})] \\ &= \text{svec}(C) + \alpha[\mathcal{A}_s^\top(\overline{\Delta y} - \Delta y)] \\ &= \text{svec}(C). \end{aligned}$$

□

Having established the fact that our search directions computed from inexact estimates of the solution to (29) preserve primal and dual feasibility, we next certify the validity of using the system (29) to obtain a solution to the system (25). Further, we demonstrate that if the system (25) has a unique solution, then the system (29) does as well. The proof of the next result is straightforward using the definition of the quantities involved.

Proposition 10. *Suppose X and S are positive definite. Then, any solution $(\Delta z, \Delta y)$ of the system (29), provides a solution $(\Delta X, \Delta y, \Delta S)$ to the system (25).*

The following result from [23] plays a crucial role in the analysis required to establish polynomial convergence of our IF-QIPM.

Lemma 4 (Lemma 2.1 in [23]). *Suppose that $(X, S) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$, $Q \in \mathbb{R}^{n \times n}$ is a nonsingular matrix and $\nu = \frac{X \bullet S}{n}$. Then:*

- (a) $d(\tilde{X}, \tilde{S}) = d(X, S)$ where $\tilde{X} = QXQ^\top$ and $\tilde{S} = Q^{-\top}SQ^{-1}$.
- (b) $d(X, S) \leq \|H_Q(XS - \nu I)\|_F$ with equality holding if $QXSQ^{-1} \in \mathcal{S}^n$.

5 Technical Results

In this section, we present the technical results required to prove that our IF-QIPM for SDO exhibits a polynomial iteration complexity. Our convergence analysis can be viewed as a generalization of Gondzio's [46] analysis from LCQO to SDO. To accomplish this, we adapt the work of Monteiro [23] to account for the inexactness of the complementarity condition in order to prove polynomial convergence of the IF-QIPM presented in Section 6. Though in this work we chose the Nesterov-Todd direction $P = W^{-1/2}$, the convergence analysis we perform is presented in general terms of P , and holds for any member of the Monteiro-Zhang family of search directions [38], which includes the NT, AHO and HKM directions. We leave the results used to establish convergence of the II-QIPM to Appendix B.

Following [23], in this section we assume that $(X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n$ and that $P \in \mathbb{R}^{n \times n}$ is a nonsingular matrix. Additionally, we assume $(\Delta X, \Delta y, \Delta S)$ is a solution of the system

$$A^{(i)} \bullet \Delta X = b_i - A^{(i)} \bullet X, \quad \forall i \in [m], \quad (30a)$$

$$\sum_{i=1}^m \Delta y_i A^{(i)} + \Delta S = C - S - \sum_{i=1}^m y_i A^{(i)}, \quad (30b)$$

$$H_P(\Delta X S + X \Delta S) = \sigma \nu I - H_P(X S) + R^r, \quad (30c)$$

for some $\sigma \in [0, 1]$. Recall that $(\Delta X, \Delta y, \Delta S)$ is a solution to the system (25) obtained from our classical estimate $(\Delta z, \Delta y)$ of the solution $|\Delta z \circ \Delta y\rangle$ to the Newton linear system (29). That is,

$$(\Delta X, \Delta y, \Delta S) = (\mathbf{smat}(Q_2 \Delta z), \Delta y, -\mathbf{smat}(\mathcal{A}_s^\top \Delta y)), \quad (31)$$

where ΔX and ΔS are computed according to (28a) and (28b), respectively. Further, for $\alpha \in \mathbb{R}$ we define the quantities:

$$\widetilde{X} = P X P^\top, \quad \widetilde{S} = P^{-\top} S P^{-1}, \quad (32a)$$

$$\widetilde{\Delta X} = P \Delta X P^\top, \quad \widetilde{\Delta S} = P^{-\top} \Delta S P^{-1}, \quad (32b)$$

$$X(\alpha) = X + \alpha \Delta X, \quad S(\alpha) = S + \alpha \Delta S, \quad y(\alpha) = y + \alpha \Delta y, \quad (32c)$$

$$\widetilde{X}(\alpha) = P X(\alpha) P^\top = \widetilde{X} + \alpha \widetilde{\Delta X}, \quad (32d)$$

$$\widetilde{S}(\alpha) = P^{-\top} S(\alpha) P^{-1} = \widetilde{S} + \alpha \widetilde{\Delta S}, \quad (32e)$$

$$\mathcal{W}_X = \widetilde{X}^{-1/2} \left[\widetilde{\Delta X} \widetilde{S} + \widetilde{X} \widetilde{\Delta S} + \widetilde{X} \widetilde{S} - \sigma \nu I - \widetilde{R}^r \right] \widetilde{X}^{1/2}, \quad (32f)$$

and

$$\nu = \frac{X \bullet S}{n} = \frac{\widetilde{X} \bullet \widetilde{S}}{n}, \quad \nu(\alpha) = \frac{X(\alpha) \bullet S(\alpha)}{n} = \frac{\widetilde{X}(\alpha) \bullet \widetilde{S}(\alpha)}{n}. \quad (33)$$

Lemma 5 (Lemma 3.1 in [23]). $(\Delta X, \Delta y, \Delta S)$ is a solution of the system (30) if and only if $(\widetilde{\Delta X}, \Delta y, \widetilde{\Delta S})$ is a solution of the system

$$\widetilde{A}^{(i)} \bullet \widetilde{\Delta X} = b_i - \widetilde{A}^{(i)} \bullet \widetilde{X}, \quad \forall i \in [m], \quad (34a)$$

$$\sum_{i=1}^m \Delta y_i \widetilde{A}^{(i)} + \widetilde{\Delta S} = \widetilde{C} - \widetilde{S} - \sum_{i=1}^m y_i \widetilde{A}^{(i)} \quad (34b)$$

$$H_I(\widetilde{\Delta X} \widetilde{S} + \widetilde{X} \widetilde{\Delta S}) = \sigma \nu I - H_I(\widetilde{X} \widetilde{S}) + \widetilde{R}^r \quad (34c)$$

where $\widetilde{R}^r = P^{-\top} R^r P^{-1}$, $\widetilde{C} = P^{-\top} C P^{-1}$ and $\widetilde{A}^{(i)} = P^{-\top} A^{(i)} P^{-1}$ for $i \in [m]$.

Proof. The proof follows as a consequence of equations (30), (32a) and (32b). \square

For the rest of this section, we work under the assumption that

$$(X, y, S) \in \mathcal{P}^0 \times \mathcal{D}^0.$$

We remark that in the scaled setting our assumption for the residual term in (AR1) is given by:

$$\|\tilde{R}^r\|_F \leq \beta \|\tilde{R}^c\|_F, \quad (\text{AR2})$$

where $\beta \in (0, 1)$ and $\tilde{R}^c = \sigma\nu I - H_I(\tilde{X}\tilde{S})$. For the analysis in this section we assume that the error tolerances are chosen so that the residual term \tilde{R}^r for the scaled problem (34) in each iteration satisfies

$$\|\tilde{X}^{-1/2}\tilde{R}^r\tilde{X}^{1/2}\|_F \leq \beta \|\tilde{X}^{-1/2}\tilde{R}^c\tilde{X}^{1/2}\|_F \quad (\text{AR3})$$

for some $\beta \in (0, 1)$. Note that these assumptions that we make regarding the residual are mild. In light of Lemma 4(a), R^r and \tilde{R}^r satisfy the same Frobenius norm bound, and the right hand sides of the inequalities in (AR1), (AR2) and (AR3) are all invariant under the stated scaling.

Lemma 6 (Lemma 3.3 in [23]). *The following relations hold:*

$$H_{\tilde{X}^{1/2}}(\mathcal{W}_X) = 0, \quad (35)$$

$$\widetilde{\Delta X} \bullet \widetilde{\Delta S} = \Delta X \bullet \Delta S = 0. \quad (36)$$

Proof. Equation (35) is a direct consequence of (34c). Further from our assumption

$$(X, y, S) \in \mathcal{P}^0 \times \mathcal{D}^0,$$

and applying equations (30c), (30b) and (32b) yields the two identities in (36). \square

The following result can be viewed as the inexact analogue of Lemma 3.4 in [23].

Lemma 7. *For all $\alpha \in \mathbb{R}$ we have*

$$\nu(\alpha) = (1 - \alpha + \sigma\alpha)\nu + \alpha \frac{\text{tr}(R^r)}{n}, \quad (37)$$

$$\begin{aligned} \tilde{X}^{-1/2} \left[\tilde{X}(\alpha)\tilde{S}(\alpha) - \nu(\alpha)I \right] \tilde{X}^{1/2} &= (1 - \alpha)(\tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2} - \nu I) + \alpha\mathcal{W}_X + \alpha^2\tilde{X}^{-1/2}\widetilde{\Delta X}\widetilde{\Delta S}\tilde{X}^{1/2} \\ &+ \alpha\tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\text{tr}(R^r)}{n}I \right] \tilde{X}^{1/2}. \end{aligned} \quad (38)$$

Proof. From (32c) we have

$$X(\alpha)S(\alpha) = (X + \alpha\Delta X)(S + \alpha\Delta S) = XS + \alpha(X\Delta S + \Delta XS) + \alpha^2\Delta X\Delta S.$$

Hence, by the linearity of $H_P(\cdot)$ and equation (30c), it follows

$$\begin{aligned} X(\alpha)S(\alpha) &= (X + \alpha\Delta X)(S + \alpha\Delta S) = H_P(XS) + \alpha H_P(X\Delta S + \Delta XS) + \alpha^2 H_P(\Delta X\Delta S) \\ &= H_P(XS) + \alpha[H_P(\sigma\nu I - XS) + R^r] + \alpha^2 H_P(\Delta X\Delta S) \\ &= (1 - \alpha)H_P(XS) + \alpha\sigma\nu I + \alpha R^r + \alpha^2 H_P(\Delta X\Delta S). \end{aligned}$$

Now, for $M \in \mathbb{R}^{n \times n}$ we have³ $\text{tr}(H_P(M)) = \text{tr}(M)$, and thus

$$\begin{aligned} X(\alpha) \bullet S(\alpha) &= \text{tr}[X(\alpha)S(\alpha)] = \text{tr}[H_P(X(\alpha)S(\alpha))] \\ &= \text{tr}[(1 - \alpha)H_P(XS) + \alpha\sigma\nu I + \alpha R^r + \alpha^2 H_P(\Delta X\Delta S)] \\ &= (1 - \alpha) \text{tr}[H_P(XS)] + \alpha\sigma\nu \text{tr}(I) + \alpha \text{tr}(R^r) + \alpha^2 \text{tr}[H_P(\Delta X\Delta S)] \\ &= (1 - \alpha)X \bullet S + \alpha\sigma\nu n + \alpha \text{tr}(R^r) + \alpha^2 \Delta X \bullet \Delta S \\ &= (1 - \alpha)X \bullet S + \alpha\sigma\nu n + \alpha \text{tr}(R^r), \end{aligned}$$

where the final equality follows from (36). Dividing the above expression by n and applying (33) yields (37).

³This is due to the fact that the eigenvalues of M are invariant under the transformation $H_P(M)$.

From here, we apply equations (32d), (32e), (37) and (32f), and it follows:

$$\begin{aligned}
\tilde{X}(\alpha)\tilde{S}(\alpha) - \nu(\alpha)I &= (\tilde{X} + \alpha\widetilde{\Delta X})(\tilde{S} + \alpha\widetilde{\Delta S}) - \nu(\alpha)I \\
&= \tilde{X}\tilde{S} + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S}) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} - \nu(\alpha)I \\
&= (1 - \alpha)\tilde{X}\tilde{S} + \alpha\tilde{X}\tilde{S} + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S}) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} - \nu(\alpha)I \\
&= (1 - \alpha)\tilde{X}\tilde{S} + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S} + \tilde{X}\tilde{S}) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} - \nu(\alpha)I \\
&= (1 - \alpha)\tilde{X}\tilde{S} + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S} + \tilde{X}\tilde{S}) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} - \nu(\alpha)I + \underbrace{(\alpha\tilde{R}^r - \alpha\tilde{R}^r)}_{=0} \\
&= (1 - \alpha)\tilde{X}\tilde{S} + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S} + \tilde{X}\tilde{S}) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} - \left[(1 - \alpha + \sigma\alpha)\nu + \frac{\alpha \operatorname{tr}(R^r)}{n} \right] I \\
&\quad + (\alpha\tilde{R}^r - \alpha\tilde{R}^r) \\
&= (1 - \alpha)(\tilde{X}\tilde{S} - \nu I) + \alpha(\widetilde{\Delta X}\tilde{S} + \tilde{X}\widetilde{\Delta S} + \tilde{X}\tilde{S} - \sigma\nu I - \tilde{R}^r) + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} \\
&\quad + \alpha \left[\tilde{R}^r - \frac{\operatorname{tr}(R^r)}{n} I \right] \\
&= (1 - \alpha)(\tilde{X}\tilde{S} - \nu I) + \alpha\tilde{X}^{1/2}\mathcal{W}_X\tilde{X}^{-1/2} + \alpha^2\widetilde{\Delta X}\widetilde{\Delta S} + \alpha \left[\tilde{R}^r - \frac{\operatorname{tr}(R^r)}{n} I \right],
\end{aligned}$$

which holds for any $\alpha \in \mathbb{R}$. Multiplying on the left by $\tilde{X}^{-1/2}$ and on the right by $\tilde{X}^{1/2}$ gives (38). \square

The following result from [23] plays a crucial role in the analysis.

Lemma 8 (Lemma 3.5 in [23]). *Let $M \in \mathbb{R}^{n \times n}$ be such that $H_Q(M) = 0$ for some nonsingular $Q \in \mathbb{R}^{n \times n}$. Then,*

$$\|H_I(M)\|_F \leq \frac{1}{2}\|M - M^\top\|_F, \quad (39)$$

$$\|M\|_F \leq \frac{\sqrt{2}}{2}\|M - M^\top\|_F. \quad (40)$$

In particular, if $M = U_1 + U_2$ for some $U_1 \in \mathcal{S}^n$ and $U_2 \in \mathbb{R}^{n \times n}$, then

$$\|M\|_F \leq \sqrt{2}\|U_2\|_F.$$

We can now use the above result in order to prove the following lemma, which is adapted from Lemma 3.6 in [23].

Lemma 9. *For every $\theta \in \mathbb{R}$, we have*

$$\|\mathcal{W}_X\|_F \leq \sqrt{2}\delta_x \left\| \tilde{S}^{1/2}\tilde{X}^{1/2} - \theta\nu\tilde{S}^{-1/2}\tilde{X}^{-1/2} \right\| + \sqrt{2} \left\| \tilde{X}^{-1/2}\tilde{R}^r\tilde{X}^{1/2} \right\|_F, \quad (41)$$

and

$$\begin{aligned}
\left\| \tilde{X}^{-1/2} \left[\tilde{X}(\alpha)\tilde{S}(\alpha) - \nu(\alpha)I \right] \tilde{X}^{1/2} \right\|_F &\leq (1 - \alpha)d(\tilde{X}, \tilde{S}) + \alpha^2\delta_x\delta_s + \alpha \left\| \tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\operatorname{tr}(R^r)}{n} I \right] \tilde{X}^{1/2} \right\|_F \\
&\quad + \alpha\sqrt{2} \left\| \tilde{S}^{1/2}\tilde{X}^{1/2} - \theta\nu\tilde{S}^{-1/2}\tilde{X}^{-1/2} \right\| + \sqrt{2} \left\| \tilde{X}^{-1/2}\tilde{R}^r\tilde{X}^{1/2} \right\|_F,
\end{aligned} \quad (42)$$

for all $\alpha \in [0, 1]$, where

$$\delta_x = \left\| \tilde{X}^{-1/2}\widetilde{\Delta X}\tilde{S}^{1/2} \right\|_F, \quad \delta_s = \left\| \tilde{S}^{-1/2}\widetilde{\Delta S}\tilde{X}^{1/2} \right\|_F. \quad (43)$$

Proof. We follow a similar strategy to [23], but modify the decomposition (namely the term U_2), to account for the inexactness residual term. Let $\theta \in \mathbb{R}$ be given. By (32f), we have $\mathcal{W}_X = U_1 + U_2$, where

$$\begin{aligned} U_1 &= \tilde{X}^{1/2} \widetilde{\Delta S} \tilde{X}^{1/2} + \theta \nu \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{X}^{-1/2} + \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \sigma \nu I \in \mathcal{S}^n, \text{ and} \\ U_2 &= \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} \left(\tilde{S}^{1/2} \tilde{X}^{1/2} - \theta \nu \tilde{S}^{-1/2} \tilde{X}^{-1/2} \right) - \tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2}. \end{aligned}$$

Then, applying (35), it follows $H_Q(\mathcal{W}_X) = 0$ for $Q = \tilde{X}^{1/2}$. This fact, combined with Lemma 8 and the fact that $\|AB\|_F \leq \|A\|_F \|B\|_F$ for every $A, B \in \mathbb{R}^{n \times n}$, from the definition of U_2 and δ_x it follows

$$\|\mathcal{W}_X\|_F \leq \sqrt{2} \|U_2\|_F \leq \sqrt{2} \delta_x \left\| \tilde{S}^{1/2} \tilde{X}^{1/2} - \theta \nu \tilde{S}^{-1/2} \tilde{X}^{-1/2} \right\| + \sqrt{2} \left\| \tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2} \right\|_F,$$

i.e., (41) holds. From here, we apply (6) and (38), and note that for $\alpha \in [0, 1]$, by the definition of δ_x and δ_s we have:

$$\begin{aligned} & \left\| \tilde{X}^{-1/2} \left[\tilde{X}(\alpha) \tilde{S}(\alpha) - \nu(\alpha) I \right] \tilde{X}^{1/2} \right\|_F \\ & \leq (1 - \alpha) \left\| \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \nu I \right\|_F + \alpha \|\mathcal{W}_X\|_F + \alpha \delta_x \delta_s + \alpha \left\| \tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\text{tr}(R^r)}{n} I \right] \tilde{X}^{1/2} \right\|_F \\ & = (1 - \alpha) d(\tilde{X}, \tilde{S}) + \alpha \|\mathcal{W}_X\|_F + \alpha^2 \delta_x \delta_s + \alpha \left\| \tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\text{tr}(R^r)}{n} I \right] \tilde{X}^{1/2} \right\|_F. \end{aligned}$$

Combining the above result with (41) implies (42) and the proof is complete. \square

For ease of notation, we follow [23] in defining the quantity

$$\Phi_\theta(A, B) = \left\| A^{1/2} B^{1/2} - \theta \frac{A \bullet B}{n} A^{-1/2} B^{-1/2} \right\|_F \quad (44)$$

for every $(A, B) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$ and $\theta \in \mathbb{R}$.

Lemma 10 (Lemma 3.7 in [23]). *If $d(X, S) \leq \gamma \nu$ for some $\gamma \in (0, 1)$, then*

$$\left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|^2 \leq \frac{1}{(1 - \gamma) \nu}, \quad (45a)$$

$$\left[\Phi_\theta(\tilde{X}, \tilde{S}) \right]^2 \leq \frac{\gamma^2 + (1 - \theta)^2 n}{1 - \gamma} \nu. \quad (45b)$$

The following result establishes upper bounds on the Frobenius norms of terms involving the residuals.

Lemma 11. *Suppose that the error tolerances are chosen such that the assumptions (AR1), (AR2) and (AR3) hold. For $\beta \in (0, 1)$ we have*

$$\|R^r\|_F \leq \beta \gamma \sigma \nu, \quad \|\tilde{R}^r\|_F \leq \beta \gamma \sigma \nu, \quad (46a)$$

$$\left\| \tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2} \right\|_F \leq \beta \gamma \sigma \nu, \quad (46b)$$

$$\left\| \tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\text{tr}(R^r)}{n} I \right] \tilde{X}^{1/2} \right\|_F \leq 2 \beta \gamma \sigma \nu. \quad (46c)$$

Proof. From equation (34c), we have $\tilde{R}^c = \sigma \nu I - H_I(\tilde{X}, \tilde{S})$. We first note that

$$\begin{aligned} \tilde{X}^{-1/2} \tilde{R}^c \tilde{X}^{1/2} &= \tilde{X}^{-1/2} H_I(\sigma \nu I - \tilde{X} \tilde{S}) \tilde{X}^{1/2} = \tilde{X}^{-1/2} [\sigma \nu I - \tilde{X} \tilde{S}] \tilde{X}^{1/2} = \sigma \nu I - \tilde{X}^{-1/2} \tilde{X} \tilde{S} \tilde{X}^{1/2} \\ &= \sigma \nu I - \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2}. \end{aligned}$$

Thus, it follows

$$\left\| \tilde{X}^{-1/2} \tilde{R}^c \tilde{X}^{1/2} \right\|_F = \left\| \sigma \nu I - \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} \right\|_F = \left\| \tilde{X}^{1/2} \tilde{S} \tilde{X}^{1/2} - \sigma \nu I \right\|_F = d_\sigma(\tilde{X}, \tilde{S}).$$

Further, applying Lemma 4(a) one can observe

$$\|\tilde{R}^c\|_F = \|H_I(\tilde{X}, \tilde{S}) - \sigma\nu I\|_F = d_\sigma(\tilde{X}, \tilde{S}) = d_\sigma(X, S) \leq \sigma\gamma\nu.$$

Combining this fact with assumptions (AR1) and (AR2) we have

$$\|R^r\|_F \leq \beta\|R^c\|_F \leq \beta\sigma\gamma\nu, \text{ and } \|\tilde{R}^r\|_F \leq \beta\|\tilde{R}^c\|_F \leq \beta\sigma\gamma\nu,$$

and hence (46a) holds. By assumption (AR3), it follows that

$$\|\tilde{X}^{-1/2}\tilde{R}^r\tilde{X}^{1/2}\|_F \leq \beta\|\tilde{X}^{-1/2}\tilde{R}^c\tilde{X}^{1/2}\|_F \leq \beta\sigma\gamma\nu.$$

Recall that for two matrices $A, B \in \mathbb{R}^{n \times n}$, $\text{tr}(AB) \leq \|A\|_F\|B\|_F$, thus

$$\text{tr}(R^r) = \text{tr}(R^r I) \leq \|R^r\|_F\|I\|_F = \sqrt{n}\|R^r\|_F.$$

Then, noting that $\|R^r\|_F \leq \beta\|R^c\|_F \leq \beta\sigma\gamma\nu$, it follows

$$\begin{aligned} \left\| \tilde{X}^{-1/2} \left[\tilde{R}^r - \frac{\text{tr}(R^r)}{n} I \right] \tilde{X}^{1/2} \right\|_F &\leq \left\| \tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2} \right\|_F + \left\| \frac{\text{tr}(R^r)}{n} I \right\|_F \\ &\leq \beta\sigma\gamma\nu + \frac{|\text{tr}(R^r)|}{n} \sqrt{n} \\ &\leq \beta\sigma\gamma\nu + \|R^r\|_F \leq 2\beta\sigma\gamma\nu, \end{aligned}$$

which completes the proof. \square

In the next result, we establish an upper bound on the terms δ_x and δ_s by adapting Lemma 3.8 of [23] to the inexact setting.

Lemma 12. *If $(X, y, S) \in \mathcal{N}_F(\gamma)$ for some $\gamma > 0$ satisfying*

$$2\sqrt{2}\frac{\gamma}{1-\gamma} \leq 1, \quad (47)$$

and $\beta \in (0, 1)$ such that

$$\beta\sigma \leq \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}},$$

then

$$\max\{\delta_x, \delta_s\} \leq 2 \left(\Phi_\sigma(\tilde{X}, \tilde{S}) + \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right),$$

where δ_x , δ_s and $\Phi_\sigma(\cdot, \cdot)$ are defined by (43) and (44).

Proof. From (32f), one can see

$$\begin{aligned} \tilde{X}^{1/2} \widetilde{\Delta S} \tilde{S}^{-1/2} + \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} &= \mathcal{W}_X \tilde{X}^{-1/2} \tilde{S}^{-1/2} + \sigma\nu \tilde{X}^{-1/2} \tilde{S}^{-1/2} - \tilde{X}^{1/2} \tilde{S}^{1/2} \\ &\quad + (\tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2}) \tilde{X}^{-1/2} \tilde{S}^{-1/2}. \end{aligned} \quad (48)$$

Since $\gamma \in (0, 1)$, it follows that $(1-\gamma) \in (0, 1)$ so we have $(1-\gamma) < (1-\gamma)^{1/2}$ and hence

$$\frac{1}{(1-\gamma)^{1/2}} < \frac{1}{1-\gamma}.$$

From (36), observe that the two terms on the left hand side of (48) are orthogonal. Combining this fact with (43), (48), (44), (33), (41) with $\theta = 1$; and (45a), (45b) with $\theta = 1$ and (47), it follows that:

$$\max\{\delta_x, \delta_s\} \leq (\delta_x^2 + \delta_s^2)^{1/2} = \left(\left\| \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} \right\|_F^2 + \left\| \tilde{S}^{-1/2} \widetilde{\Delta S} \tilde{X}^{1/2} \right\|_F^2 \right)^{1/2}$$

$$\begin{aligned}
&= \left\| \tilde{X}^{-1/2} \widetilde{\Delta X} \tilde{S}^{1/2} + \tilde{X}^{1/2} \widetilde{\Delta S} \tilde{S}^{-1/2} \right\|_F \\
&\leq \|\mathcal{W}_X\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \sigma \nu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \left\| (\tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2}) \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|_F \quad (\text{by (48)}) \\
&= \|\mathcal{W}_X\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \sigma \nu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \left\| \tilde{X}^{-1/2} \tilde{R}^r \tilde{S}^{-1/2} \right\|_F \\
&\leq \sqrt{2} \delta_x \left\| \tilde{X}^{1/2} \tilde{S}^{1/2} - \nu \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| + \Phi_\sigma(\tilde{X}, \tilde{S}) + (1 + \sqrt{2}) \left\| \tilde{X}^{-1/2} \tilde{R}^r \tilde{X}^{1/2} \right\|_F \left\| \tilde{X}^{-1/2} \tilde{S}^{-1/2} \right\| \\
&\leq \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + (1 + \sqrt{2}) \beta \sigma \gamma \nu \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} \\
&= \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \left[(1 + \sqrt{2}) \frac{\gamma}{(1-\gamma)^{1/2}} \right] \beta \sigma \nu^{1/2} \\
&< \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \left[(1 + \sqrt{2}) \frac{\gamma}{(1-\gamma)} \right] \beta \sigma \nu^{1/2} \\
&\leq \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \left[2\sqrt{2} \frac{\gamma}{(1-\gamma)} \right] \beta \sigma \nu^{1/2} \\
&\leq \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \beta \sigma \nu^{1/2} \\
&\leq \sqrt{2} \delta_x \frac{\gamma \nu^{1/2}}{(1-\gamma)^{1/2}} \frac{1}{(1-\gamma)^{1/2} \nu^{1/2}} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \\
&\leq \frac{\delta_x}{2} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \leq \frac{1}{2} \max\{\delta_x, \delta_s\} + \Phi_\sigma(\tilde{X}, \tilde{S}) + \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu,
\end{aligned}$$

and the proof is complete. \square

The final result of this section, analogous to Lemma 3.9 of [23], follows from the above lemmas.

Lemma 13. *If $(X, y, S) \in \mathcal{N}_F(\gamma)$ for some $\gamma > 0$ satisfying (47), then, for every $\alpha \in [0, 1]$, $\beta \in (0, 1)$, and $\eta > 0$, we have*

$$\begin{aligned}
&\left\| \tilde{X}^{-1/2} \left[\tilde{X}(\alpha) \tilde{S}(\alpha) - \nu(\alpha) I \right] \tilde{X}^{1/2} \right\|_F \\
&\leq \left((1-\alpha)\gamma + 4\sqrt{2}\alpha \frac{\gamma [\gamma^2 + (1-\sigma)^2 n]^{1/2}}{1-\gamma} + 16\alpha^2 \frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma} + \alpha(2 + \sqrt{2})\beta\sigma\gamma \right) \nu.
\end{aligned}$$

Proof. We apply (42) with $\theta = 1$, Lemma 4 (a), (32a), the assumption $d(X, S) \leq \gamma\nu$, Lemma 11, Lemma 12 and (45b) where we choose $\theta = 1$ and $\theta = \sigma$. We have:

$$\begin{aligned}
&\left\| \tilde{X}^{-1/2} \left[\tilde{X}(\alpha) \tilde{S}(\alpha) - \nu(\alpha) I \right] \tilde{X}^{1/2} \right\|_F \\
&\leq (1-\alpha)d(X, S) + \sqrt{2}\alpha\delta_x\Phi_1(\tilde{X}, \tilde{S}) + \alpha^2\delta_x\delta_s + \alpha(2 + \sqrt{2})\beta d_\sigma(X, S) \\
&\leq (1-\alpha)\gamma\nu + 2\sqrt{2}\alpha \left(\Phi_\sigma(\tilde{X}, \tilde{S}) + \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right) \Phi_1(\tilde{X}, \tilde{S}) + \alpha^2 \left[2\Phi_\sigma(\tilde{X}, \tilde{S}) + 2\sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right]^2 \\
&\quad + \alpha(2 + \sqrt{2})\beta\sigma\gamma\nu \\
&\leq (1-\alpha)\gamma\nu + 2\sqrt{2}\alpha \left(2\sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right) \Phi_1(\tilde{X}, \tilde{S}) + \alpha^2 \left[4\sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right]^2 + \alpha(2 + \sqrt{2})\beta\sigma\gamma\nu \\
&= (1-\alpha)\gamma\nu + 4\sqrt{2}\alpha \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \cdot \Phi_1(\tilde{X}, \tilde{S}) + 16\alpha^2 \left[\sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}} \nu \right]^2 + \alpha(2 + \sqrt{2})\beta\sigma\gamma\nu \\
&\leq \left((1-\alpha)\gamma + 4\sqrt{2}\alpha \frac{\gamma [\gamma^2 + (1-\sigma)^2 n]^{1/2}}{1-\gamma} + 16\alpha^2 \frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma} + \alpha(2 + \sqrt{2})\beta\sigma\gamma \right) \nu.
\end{aligned}$$

The proof is complete. \square

6 Quantum Interior Point Methods for SDO

In this section we present our Inexact-Feasible and Inexact-Infeasible QIPMs. For the IF-QIPM, we establish polynomial convergence, then discuss how to implement the block-encoding of the Newton linear system, and conduct the overall running time analysis. Though we present the II-QIPM in its entirety, we leave the discussion on convergence, and implementing the block-encodings for the corresponding Newton linear system to the Appendix.

6.1 An Inexact-Feasible QIPM for SDO

We quantize a general IF-IPM and the resulting scheme is described in Algorithm 3. There is no classical counterpart in the literature for SDO, as the nullspace representation of the Newton system destroys the symmetry of the Newton system, and is thus impractical for classical computing. The algorithm has several parameters, the most important of which is the optimality gap tolerance ϵ . Details for all steps of Algorithm 3 are discussed subsequently in this paper.

In our initialization steps, we must calculate bases of the nullspace and row space of \mathcal{A}_s . For this step, one can either use Gaussian elimination, or the QR-factorization of \mathcal{A}_s , which is typically more stable from a numerical point of view. Theoretically this can be accomplished using $\mathcal{O}(n^{2\omega})$ classical arithmetic operations where $\omega = 2.37$ is the matrix multiplication exponent [68, 69], although this cost can be potentially reduced much closer to $\mathcal{O}(n^4)$ if the initial data is sparse and we use sparse matrix-vector multiplication, see [70]. It is also possible to trivially obtain a basis for the nullspace and the range space for “free” if the SDO problem is given in canonical form (inequalities instead of equalities): in that case, using the Self-Dual embedding model for that form, the nullspace basis matrix can be constructed trivially too from the coefficient matrix (without needing a QR-factorization or Gaussian elimination). Crucially, the computation of these bases need only be carried out once. That is, we calculate \mathcal{A}_s and Q_2 one time, and store these matrices in QRAM before the algorithm begins, as they remain unchanged for the duration of the algorithm. This step is akin to the preprocessing steps (e.g., Cholesky factorization, ordering of basis) typical of many optimization algorithms.

In steps 2 and 3 of Algorithm 3 we require classical matrix multiplication and inversion of $n \times n$ matrices. These steps take $\mathcal{O}(n^\omega)$ classical arithmetic operations. A detailed analysis for step 3 can be found in Proposition 13. In steps 4 and 5, we employ a QLSA and the tomography routine which reconstructs, with high probability, a classical description of the solution to the Newton linear system. Following our discussion on tomography in Section 2.3, this can be accomplished in $\tilde{\mathcal{O}}_{n, \frac{1}{\xi_k}} \left(\frac{n^2}{\xi_k} T_{LS} \right)$ time, where, for the NT direction, T_{LS} is the time needed to prepare and solve the NT linear system, and ξ_k is the error to which we perform our tomography steps to extract a classical estimate of the solution to the Newton linear system. A discussion on how to set ξ_k at each iterate is provided at the beginning of Section 6.1.1. Then, in step 6, we use our classical estimates of $\bar{\Delta}z$ and $\bar{\Delta}y$ to classically calculate $\bar{\Delta}X$ and $\bar{\Delta}S$. This step amounts to matrix-vector multiplication of dimension n^2 , and requires $\mathcal{O}(n^4)$ arithmetic operations. Finally, we classically update the current solutions to the SDOP X_k , y_k and S_k , and the central path parameter ν_k .

Here we establish that Algorithm 3 converges for a specific choice of the error parameters, discuss how to construct the Newton linear system for the Nesterov-Todd direction, and provide a complexity analysis for the chosen parameters.

6.1.1 Polynomial Convergence of the IF-QIPM

In order to have a convergent algorithm, we need to ensure that the errors ξ_k introduced by tomography are properly controlled. As we discussed earlier, the tomography error bound is relative (it is stated in additive form, but assumes that the vector to be extracted has unit norm): to ensure that we satisfy the absolute error bound for the unscaled vector, we need to divide ξ_k by the maximum norm of a solution vector. Now, by definition $\|R_k^c\|_F \leq \sigma\gamma\nu_k$ holds at every iteration, and we terminate once $\nu_k \leq \epsilon$. Hence, for every iteration of the algorithm before termination with an ϵ -optimal solution, the inequality

$$\|R_k^c\|_F > \sigma\gamma\epsilon,$$

Algorithm 3 Inexact-Feasible Quantum Interior Point Method

Input: $\epsilon, \delta > 0$; $\sigma = 1 - \delta/\sqrt{n}$; $\beta, \gamma \in (0, 1)$

Choose a nonsingular matrix P from Monteiro-Zhang family of search directions

Choose $(X_0, y_0, S_0) \in \mathcal{N}_F(\gamma)$

Set $\nu_0 = \frac{X_0 \bullet S_0}{n}$

Store (X_0, y_0, S_0) in QRAM

Compute bases for $\mathcal{R}(\mathcal{A}_s) = \mathcal{A}_s^\top$ and $\text{Null}(\mathcal{A}_s) = Q_2$

Store \mathcal{A}_s and Q_2 in QRAM

while $\nu > \epsilon$ **do**

1. Update central path parameter $\nu_k \leftarrow \frac{X_k \bullet S_k}{n}$
2. Compute matrices X_k^{-1} and S_k^{-1} classically, and store these matrices in QRAM
3. Compute scaling matrix P_k , right hand side matrix, and necessary matrix products for Newton system (29) and store in QRAM
4. Using block-encodings, solve (29) to construct inexact search direction $|\Delta z_k \circ \Delta y_k\rangle$
5. Obtain classical estimate $\overline{\Delta z}_k, \overline{\Delta y}_k$ of $\Delta z_k, \Delta y_k$ using vector state tomography
6. Use classical estimate $\overline{\Delta z}_k, \overline{\Delta y}_k$ to obtain classical estimate $\overline{\Delta X}_k, \overline{\Delta S}_k$ of $\Delta X_k, \Delta S_k$
7. Update current solution

$$X_{k+1} \leftarrow X_k + \Delta X_k, S_{k+1} \leftarrow S_k + \Delta S_k \text{ and } y_{k+1} \leftarrow y_k + \Delta y_k$$

$$k \leftarrow k + 1$$

end

holds, where $\gamma = \frac{1}{20}$ and $\sigma \approx 1$ for large n . In other words, the requirement on the residual bound never becomes much smaller than ϵ , if we aim at ϵ -optimality. Thus, we can set

$$\xi_k = \frac{\beta}{\varrho} \cdot \max \left\{ \|R_k^c\|_F, \frac{1}{25}\epsilon \right\}. \quad (49)$$

where ϱ is the maximum norm of a solution to the Newton linear system.

Lemma 14. *Choosing ξ_k according to (49), the inexact search direction $(\Delta z, \Delta y)$ computed at step 5 of Algorithm 3 satisfies (AR1).*

Proof. For feasible IPMs, the complementarity gap is updated as $\nu_{k+1} = (1 - \frac{\delta}{\sqrt{n}})\nu_k$, which implies that $\|R_{k+1}^c\|_F \leq (1 - \frac{\delta}{\sqrt{n}})\|R_k^c\|_F$, due to the fact that $\|R_{k+1}^c\|_F \leq \sigma\gamma\nu_{k+1}$. \square

Next, we provide the inexact analogue of Theorem 4.1 from [23], which yields the analysis of one iteration of Algorithm 3 for suitable choices of γ , δ and β .

Theorem 4. *Let $\gamma, \beta \in (0, 1)$ and $\delta \in [0, n^{1/2})$ be constants satisfying*

$$\frac{2\sqrt{2}\gamma}{1-\gamma} \leq 1, \quad \beta\sigma \leq \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}}, \quad \beta \leq 1 - \frac{\gamma}{\sqrt{n}} - \frac{21.7(\gamma^2 + \delta^2)}{(2 + \sqrt{2}) \left(1 - \frac{\delta}{\sqrt{n}}\right) \gamma(1-\gamma)}. \quad (50)$$

Suppose that $(X, y, S) \in \mathcal{N}_F(\gamma)$ and let $(\Delta X, \Delta y, \Delta S)$ denote the solution that we obtain from solving system (29), where $\sigma = 1 - \delta/\sqrt{n}$, $\nu = (X \bullet S)/n$ and $P \in \mathbb{R}^{n \times n}$ is a nonsingular matrix. Then,

$$(a) \quad (\widehat{X}, \widehat{y}, \widehat{S}) = (X + \Delta X, y + \Delta y, S + \Delta S) \in \mathcal{N}_F(\gamma);$$

$$(b) \widehat{X} \bullet \widehat{S} = \left(1 - \frac{\delta}{\sqrt{n}}\right) (X \bullet S).$$

Proof. In the following proof, we seek to show that a new iterate, after a step α in this direction remains in the neighborhood of the central path. That is, the new iterate satisfies

$$\left\| \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \nu(\alpha) I \right] \widetilde{X}^{1/2} \right\|_F \leq \gamma \nu(\alpha) = \gamma(1 - \alpha + \sigma \alpha) \nu + \alpha \gamma \frac{\text{tr}(R^r)}{n}.$$

Now, $|\text{tr}(R^r)| \leq \sqrt{n} \|R^r\|_F \leq \sqrt{n} \beta \sigma \gamma \nu < \sqrt{n} \sigma \gamma \nu$. However, since we cannot make any conclusions regarding the sign of the quantity $\text{tr}(R^r)$, it suffices to show

$$\left\| \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \nu(\alpha) I \right] \widetilde{X}^{1/2} \right\|_F \leq \gamma(1 - \alpha + \sigma \alpha) \nu - \alpha \gamma \frac{\sigma \gamma \nu}{\sqrt{n}}. \quad (*)$$

First, we establish an upper bound on the quantity

$$\left\| \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \nu(\alpha) I \right] \widetilde{X}^{1/2} \right\|_F,$$

and show that this upper bound satisfies the inequality (*) for appropriately chosen γ, β and δ . Applying Lemma 13, along with the definition of σ and (50), for all $\alpha \in [0, 1]$ we have

$$\begin{aligned} & \left\| \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \nu(\alpha) I \right] \widetilde{X}^{1/2} \right\|_F \\ & \leq \left((1 - \alpha) \gamma + 4\sqrt{2} \alpha \frac{\gamma [\gamma^2 + (1 - \sigma)^2 n]^{1/2}}{1 - \gamma} + 16\alpha^2 \frac{\gamma^2 + (1 - \sigma)^2 n}{1 - \gamma} + \alpha(2 + \sqrt{2}) \beta \sigma \gamma \right) \nu \\ & \leq \left((1 - \alpha) \gamma + 21.7 \alpha \frac{\gamma^2 + (1 - \sigma)^2 n}{1 - \gamma} + \alpha(2 + \sqrt{2}) \beta \sigma \gamma \right) \nu \\ & = \left((1 - \alpha) \gamma + \alpha \frac{21.7(\gamma^2 + (1 - \sigma)^2 n)}{1 - \gamma} + \alpha(2 + \sqrt{2}) \beta \sigma \gamma \right) \nu \\ & = \left((1 - \alpha) \gamma + \alpha \frac{21.7(\gamma^2 + \delta^2)}{1 - \gamma} + \alpha(2 + \sqrt{2}) \beta \sigma \gamma \right) \nu. \quad (**) \end{aligned}$$

Substituting the upper bound (**) into the inequality (*), one can see that we must have

$$\left(\alpha \frac{21.7(\gamma^2 + \delta^2)}{1 - \gamma} + \alpha(2 + \sqrt{2}) \beta \sigma \gamma \right) \nu \leq \alpha \left(1 - \frac{\delta}{\sqrt{n}} \right) \gamma \nu - \alpha \gamma \frac{\sigma \gamma \nu}{\sqrt{n}}.$$

Dropping like terms on both sides of the inequality, this implies

$$\frac{21.7(\gamma^2 + \delta^2)}{1 - \gamma} + (2 + \sqrt{2}) \beta \sigma \gamma \leq \left(1 - \frac{\delta}{\sqrt{n}} \right) \gamma - \frac{\sigma \gamma^2}{\sqrt{n}}.$$

Using the fact that $\sigma = 1 - \delta/\sqrt{n}$, we can further simplify this expression to

$$\beta \leq 1 - \frac{\gamma}{\sqrt{n}} - \frac{21.7(\gamma^2 + \delta^2)}{(2 + \sqrt{2}) \left(1 - \frac{\delta}{\sqrt{n}} \right) \gamma (1 - \gamma)}. \quad (51)$$

Choosing $\beta \in (0, 1)$ such that (51) is satisfied, it follows

$$\begin{aligned} \left\| \widetilde{X}^{-1/2} \left[\widetilde{X}(\alpha) \widetilde{S}(\alpha) - \nu(\alpha) I \right] \widetilde{X}^{1/2} \right\|_F & \leq \left((1 - \alpha) \gamma + \alpha \left(1 - \frac{\delta}{\sqrt{n}} \right) \gamma \right) \nu + \alpha \gamma \frac{\text{tr}(R^r)}{n} \\ & = ((1 - \alpha) \gamma + \alpha \gamma \sigma) \nu + \alpha \gamma \frac{\text{tr}(R^r)}{n}. \end{aligned} \quad (52)$$

By equation (37)

$$\nu(\alpha) = (1 - \alpha + \sigma \alpha) \nu + \alpha \frac{\text{tr}(R^r)}{n}.$$

Hence,

$$\left\| \tilde{X}^{-1/2} \left[\tilde{X}(\alpha) \tilde{S}(\alpha) - \nu(\alpha) I \right] \tilde{X}^{1/2} \right\|_F \leq \gamma(1 - \alpha + \alpha\sigma)\nu + \gamma \frac{\text{tr}(R^r)}{n} = \gamma\nu(\alpha).$$

Therefore, defining

$$G(\alpha) = \left(\tilde{X}^{-1/2} \tilde{X}(\alpha) \tilde{S}(\alpha) \tilde{X}^{1/2} \right) / \nu(\alpha),$$

it follows that $G(\alpha)$ is nonsingular for all $\alpha \in [0, 1]$ as $\|G(\alpha) - I\| \leq \gamma < 1$. As a consequence, $\tilde{X}(\alpha)$ and $\tilde{S}(\alpha)$ are also nonsingular for $\alpha \in [0, 1]$.

From here, we follow [23] by combining (32d) and (32e) with a simple continuity argument which demonstrates that for each $\alpha \in [0, 1]$, we have that $(\tilde{X}(\alpha), \tilde{S}(\alpha))$ and $(X(\alpha), S(\alpha))$ are in $\mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$. Noting our assumption $(X, y, S) \in \mathcal{P}^0 \times \mathcal{D}^0$, along with equations (30b) and (30a), one can observe that $(X(\alpha), y(\alpha), S(\alpha))$ satisfies

$$\begin{aligned} A^{(i)} \bullet X - b_i &= 0, \quad i = 1, \dots, m, \\ \sum_{i \in [m]} y_i A^{(i)} + S - C &= 0, \end{aligned}$$

and thus $(X(\alpha), y(\alpha), S(\alpha)) \in \mathcal{P}^0 \times \mathcal{D}^0$ for every $\alpha \in [0, 1]$. Finally, we apply Lemma 4(a) with $(X, S) = (X(1), S(1))$ and $Q = P$, along with Lemma 4(b) with $(X, S) = (\tilde{X}(1), \tilde{S}(1))$ and $Q = \tilde{X}^{-1/2}$, and equations (32d), (32e), (37) with $\alpha = 1$, and (52) with $\alpha = 1$, which yields

$$\begin{aligned} d(X(1), S(1)) &= d(\tilde{X}(1), \tilde{S}(1)) \leq \left\| H_{\tilde{X}^{-1/2}} \left[\tilde{X}(1) \tilde{S}(1) - \nu(1) I \right] \right\|_F \\ &\leq \left\| \tilde{X}^{-1/2} \left[\tilde{X}(1) \tilde{S}(1) - \nu(1) I \right] \tilde{X}^{1/2} \right\|_F \leq \gamma\nu(1). \end{aligned}$$

Therefore, $(\hat{X}, \hat{y}, \hat{S}) = (X + \Delta X, y + \Delta y, S + \Delta S) \in \mathcal{N}_F(\gamma)$, and hence (a) holds. Further, (b) follows immediately from the definition of σ and equation (37) with $\alpha = 1$. \square

Following from Theorem 4, we obtain the convergence result for Algorithm 3.

Corollary 1. *Let γ, δ and β be constants in $(0, 1)$ and satisfying*

$$\frac{2\sqrt{2}\gamma}{1-\gamma} \leq 1, \quad \beta\sigma \leq \sqrt{\frac{\gamma^2 + (1-\sigma)^2 n}{1-\gamma}}, \quad \beta \leq 1 - \frac{\gamma}{\sqrt{n}} - \frac{21.7(\gamma^2 + \delta^2)}{(2 + \sqrt{2}) \left(1 - \frac{\delta}{\sqrt{n}}\right) \gamma(1-\gamma)}.$$

Then, every iterate (X_k, y_k, S_k) generated by the IF-QIPM in Algorithm 3 is an element of the neighborhood $\mathcal{N}_F(\gamma)$, and satisfies the relation $X_k \bullet S_k = (1 - \delta/\sqrt{n})^k (X_0 \bullet S_0)$. Further, the IF-QIPM terminates in at most $\mathcal{O}(\sqrt{n} \log(1/\epsilon))$ iterations.

Examples of constants γ, δ and β satisfying the conditions of Corollary 1 are $\gamma = \delta = 1/20$ and $\beta = 1/4$.

6.1.2 Implementing the Nesterov-Todd linear system

As in [71], we factorize the NT Newton linear system given by (29), and construct block-encodings for its factors. In this paper we use block-encodings and QSLA within an IPM by pre-computing some matrix inverses and products *classically*, and storing the results in QRAM so that the block-encodings of the factors of the Newton linear system can be constructed in time $\tilde{\mathcal{O}}_{\frac{\epsilon}{2}}(1)$, where ϵ is some error tolerance. This is more efficient than other schemes that we explored. Note that the largest matrices appearing in the Newton system, namely $Q_2 \in \mathbb{R}^{\frac{n(n+1)}{2} \times (\frac{n(n+1)}{2} - m)}$ and $\mathcal{A}_s^\top \in \mathbb{R}^{\frac{n(n+1)}{2} \times m}$, are only calculated once before the algorithm begins so their size is not prohibitive. Hence, compared to solving the Newton linear system, classical computation of matrix powers and products has an insignificant cost, and we circumvent increased dependence on the condition number.

Proposition 11. *Let*

$$M_1 = [(P \otimes_s P^{-\top} S)Q_2 \quad 0].$$

Let P , Q_2 , and $P^{-\top} S$ be stored in QRAM. Then a $(\|M_1\|_F, \mathcal{O}(\log n), \xi_{M_1})$ -block-encoding of M_1 can be constructed in time $\tilde{\mathcal{O}}_{\xi_{M_1}} \frac{n}{\xi_{M_1}}(1)$.

Proof. First, given that Q_2 is stored in QRAM, we can construct a $(\|Q_2\|_F, \mathcal{O}(\log n), \xi_1)$ -block-encoding of Q_2 in time $\tilde{\mathcal{O}}_{\xi_1} \frac{n}{\xi_1}(1)$. Next, with P and $P^{-\top} S$ stored in QRAM, we can apply Proposition 7 so that we can construct a $(\|P \otimes_s P^{-\top} S\|_F, \mathcal{O}(\log n), \xi_2)$ -block-encoding of $P \otimes_s P^{-\top} S$ in time $\tilde{\mathcal{O}}_{\xi_2} \frac{n}{\xi_2}(1)$.

From here we apply Proposition 3 with

$$\xi_1 = \frac{\xi_{M_1}}{2\|P \otimes_s P^{-\top} S\|_F} \text{ and } \xi_2 = \frac{\xi_{M_1}}{2\|Q_2\|_F},$$

yielding a $(\|M_1\|_F, \mathcal{O}(\log n), \xi_{M_1})$ -block-encoding of M_1 in time $\tilde{\mathcal{O}}_{\xi_{M_1}} \frac{n}{\xi_{M_1}}(1)$, as desired. \square

Proposition 12. *Let*

$$M_2 = [0 \quad (PX \otimes_s P^{-\top})(-\mathcal{A}_s^\top)].$$

Let $P^{-\top}$, \mathcal{A}_s , PX and $P^{-\top} S$ be stored in QRAM. Then a $(\|M_2\|_F, \mathcal{O}(\log n), \xi_{M_2})$ -block-encoding of M_2 can be constructed in time $\tilde{\mathcal{O}}_{\xi_{M_2}} \frac{n}{\xi_{M_2}}(1)$.

Proof. The steps of the proof follow that of the previous result. With $-\mathcal{A}_s^\top$ stored in QRAM, we can construct a $(\|\mathcal{A}_s^\top\|_F, \mathcal{O}(\log n), \xi_1)$ -block-encoding of $-\mathcal{A}_s^\top$ in time $\tilde{\mathcal{O}}_{\xi_1} \frac{n}{\xi_1}(1)$. Further, with PX and $P^{-\top}$ stored in QRAM, we apply Proposition 7 such that we can construct a $(\|PX \otimes_s P^{-\top}\|_F, \mathcal{O}(\log n), \xi_2)$ -block-encoding of $PX \otimes_s P^{-\top}$ in time $\tilde{\mathcal{O}}_{\xi_2} \frac{n}{\xi_2}(1)$.

Applying Proposition 3 with

$$\xi_1 = \frac{\xi_{M_2}}{2\|PX \otimes_s P^{-\top}\|_F} \text{ and } \xi_2 = \frac{\xi_{M_2}}{2\|\mathcal{A}_s^\top\|_F},$$

we obtain a $(\|M_2\|_F, \mathcal{O}(\log n), \xi_{M_2})$ -block-encoding of M_2 , and the proof is complete. \square

Proposition 13. *The Nesterov-Todd linear system matrix (29) can be written compactly as:*

$$M_{NT} = [(P \otimes_s P^{-\top} S)Q_2 \quad (PX \otimes_s P^{-\top})(-\mathcal{A}_s^\top)]. \quad (53)$$

If P , $P^{-\top}$, $-\mathcal{A}_s^\top$, Q_2 , PX and $P^{-\top} S$ are stored in QRAM. Then a

$$(\|M_{NT}\|_F, \mathcal{O}(\log n), \xi / (\|M_{NT}\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$$

block-encoding of M_{NT} can be constructed in time $\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi}}(1)$.

Proof. Carrying out the calculations shows that (53) corresponds to the Nesterov-Todd linear system. We construct the two following block-encodings:

$$\begin{aligned} M_1 &= [(P \otimes_s P^{-\top} S)Q_2 \quad 0] \\ M_2 &= [0 \quad (PX \otimes_s P^{-\top})(-\mathcal{A}_s^\top),] \end{aligned}$$

using Propositions 11-12. We choose the precision of this step so that we obtain

$$(\|M_1\|_F, \mathcal{O}(\log n), \xi / (\|M_1\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$$

and $(\|M_2\|_F, \mathcal{O}(\log n), \xi / (\|M_2\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encodings of M_1 and M_2 , respectively, where κ refers to the condition number of (29), here and in the sequel. We add these two block-encodings together using Proposition 2, obtaining a $(\max\{\|M_1\|_F, \|M_2\|_F\}, \mathcal{O}(\log n), \xi / (\kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encoding of (53), in time $\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi}}(1)$. Since $\max\{\|M_1\|_F, \|M_2\|_F\} = \mathcal{O}(\|M_{NT}\|_F)$, we obtain the claimed result. \square

One should note that the construction of the Newton linear system provided in Proposition 13 is only one of the many possible ways to obtain a block-encoding of our Newton linear system while utilizing the Nesterov-Todd direction. Further, though the Nesterov-Todd direction is the search direction we study in this manuscript, note that our construction is written in general terms of P , thus this construction can be used for any suitable choice of P , such as the AHO ($P = I$) and HKM ($P = S^{1/2}$) directions. Note that by employing a factorization of the Newton linear system, we are able to theoretically improve the running time of implementing the block-encodings of the Newton linear system; which in turn allows us to improve the efficiency with which we solve the system. Next, we use this factorization to solve the Newton system.

Theorem 5. *Using the Nesterov-Todd direction $P = W^{-1/2}$, there is a quantum algorithm, which given $|\mathbf{svec}(R^c)\rangle$ and access to QRAM data structures encoding P , $P^{-\top}$, $-\mathcal{A}_s^\top$, Q_2 , PX and $P^{-\top}S$, outputs a state ξ -close to $|\Delta z \circ \Delta y\rangle$ in time*

$$\tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(\kappa \max\{\|M_1\|_F, \|M_2\|_F\}).$$

We can also output an estimate of $\|\Delta z \circ \Delta y\|$ with relative error δ by increasing the running time by a factor of $\frac{1}{\delta}$.

Proof. This is a direct consequence of Proposition 13 and Theorem 1. \square

Proposition 14. *The norm ϱ of the solution of the quantum Newton linear system (29) satisfies*

$$\varrho = \mathcal{O}(\|M_{NT}^{-1}\|).$$

Proof. We have

$$\begin{bmatrix} \Delta z \\ \Delta y \end{bmatrix} = (M_{NT})^{-1} \mathbf{svec}(R^c + R^r).$$

By Lemma 11, we have

$$\|R^c + R^r\|_F \leq \|R^c\|_F + \|R^r\|_F \leq (1 + \beta)\sigma\gamma\nu. \quad (54)$$

Choosing a constant $c_0 \geq (1 + \beta)\sigma\gamma$, it follows that $\|R^c + R^r\|_F \leq c_0\nu$, and therefore

$$\|R^c + R^r\|_F = \mathcal{O}(\nu) = \mathcal{O}(1),$$

where the final equality follows from the definition of condition number and $\nu = \mathcal{O}(1)$. To see this, note that across every iteration of the QIPM, we have $\max_k\{\nu_k\} = \nu_0 = \frac{X_0 \bullet S_0}{n} = \mathcal{O}(1)$, as in each iteration we strictly decrease the quantity $X \bullet S$ (which consequently decreases ν).

The bound on the norm of the solution is thus given by

$$\|(\Delta z, \Delta y)\|_F = \|M_{NT}^{-1} \mathbf{svec}(R^c + R^r)\|_F \leq \|M_{NT}^{-1}\| \cdot \|(R^c + R^r)\|_F = \mathcal{O}(\|M_{NT}^{-1}\|).$$

The proof is complete. \square

Recall (see page 30) that T_{LS} denotes the time needed to prepare and solve the Newton linear system (29) with the Nesterov-Todd direction.

Proposition 15. *Let ξ be the precision to which we prepare the quantum state proportional to the solution of the Newton linear system (29) using a QLSA. Then, we have $\varrho T_{LS} = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(n\kappa^2)$.*

Proof. The result follows trivially from Proposition 14 and noting the fact that $\max\{\|M_1\|_F, \|M_2\|_F\} = \mathcal{O}(\|M_{NT}\|_F)$. First, observe

$$T_{LS} = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(\kappa \max\{\|M_1\|_F, \|M_2\|_F\}) = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(\kappa \|M_{NT}\|_F) = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(n\kappa \cdot \|M_{NT}\|).$$

Then,

$$\varrho T_{LS} = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(n\kappa \cdot \|M_{NT}\| \cdot \|M_{NT}^{-1}\|) = \tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(n\kappa^2),$$

where the final equality follows from the definition of condition number. \square

Algorithm 4 Inexact-Infeasible Quantum Interior Point Method

Input: $\nu_0, \theta_0 = 1$; $\epsilon, \delta > 0$; $\eta_1 \in (0, 1]$; $\gamma_1, \gamma_2, \eta_2 \in (0, 1)$

Pick a sequence $\{\vartheta_i\}_{i=0}^\infty$ in $(0, 1]$, such that $\vartheta = \sum_{i=0}^\infty \vartheta_i < \infty$

Choose a nonsingular matrix P from Monteiro-Zhang family of search directions

Find an initial point $(\nu_0, \theta_0, X_0, y_0, S_0) \in \mathcal{N}_I$

Store $(\nu_0, \theta_0, X_0, y_0, S_0)$ in QRAM

while $\nu > \epsilon$ **do**

1. Compute matrix X_k^{-1} and S_k^{-1} classically, and store these matrices in QRAM
2. Compute scaling matrix P_k , right hand side matrix and necessary matrix products for Newton system factorization and store in QRAM
3. Using quantum linear systems algorithm, solve the system (24) to construct inexact search direction $|\mathbf{svec}(\Delta X_k) \circ \Delta y_k \circ \mathbf{svec}(\Delta S_k)\rangle$
4. Obtain classical estimate of $\mathbf{svec}(\Delta X_k), \Delta y_k, \mathbf{svec}(\Delta S_k)$ using tomography

5. Set

$$\alpha_k \leftarrow \max \left\{ \alpha : \alpha \in \left[0, \min \left(1, \frac{1}{\eta_1(1+\vartheta)} \right) \right], (\nu_k(\alpha), \theta_k(\alpha), X_k(\alpha), y_k(\alpha), S_k(\alpha)) \in \mathcal{N} \right\}$$

6. Update solution

$$\begin{aligned} X_{k+1} &\leftarrow X_k + \alpha_k \Delta X_k, \quad y_{k+1} \leftarrow y_k + \alpha_k \Delta y_k \quad \text{and} \quad S_{k+1} \leftarrow S_k + \alpha_k \Delta S_k \\ \nu_{k+1} &\leftarrow (1 - \alpha \eta_1) \nu_k, \quad \text{and} \quad \theta_{k+1} \leftarrow (1 - \alpha \eta_1) \theta_k \\ k &\leftarrow k + 1 \end{aligned}$$

end

6.2 An Inexact-Infeasible QIPM for SDO

The II-QIPM follows a similar algorithmic structure to that of the IF-QIPM, with the key distinction being the Newton linear system to be solved at each iteration, and that we do not need to compute the bases of the nullspace and rowspace of \mathcal{A} .

We now discuss convergence of the algorithm, using the main result from the work of Zhou and Toh [47], adapted to our case. Note that the proof in [47] is only explicitly given for the HKM direction ($P = S^{1/2}$), but convergence for the other cases can be shown analogously. For a detailed discussion on the analysis required to prove convergence, and implementing the block-encodings for the II-QIPM Newton linear system, we refer the reader to the Appendix.

To prove convergence, we first need to show that Algorithm 4 satisfies several properties. More specifically, we need to show that the inexact search direction computed at the end of step 4 satisfies (23)-(24). This requires choosing error parameters for the QLSA and for the tomography step. Recall the required bounds on the inexactness residuals, reported for convenience from (23):

$$\|r_k^d\| \leq \gamma_1 \rho \tau_k \vartheta_k, \quad \|r_k^p\| \leq \hat{\vartheta} \gamma_1 \rho \tau_k \vartheta_k, \quad \|r_k^c\| \leq \frac{1}{2} (1 - \eta_2) \gamma_2 \theta_k \rho^2.$$

Note that to obtain a classical estimate in step 4, we must determine the norm of the inexact search direction. Recall that QLSA also introduces some error in the inexact search direction. In line with our convergence discussion for the IF-QIPM we assume that all of the error comes from tomography.

To determine appropriate error parameters, we first discuss the sequence $\{\vartheta_k\}_{k=0}^\infty$, as ϑ_k determines the rate at which the residual norms must be decreased with respect to τ_k , where $\tau \in (0, 1]$ is defined in (17). We choose the sequence:

$$\vartheta_k = \begin{cases} \min \left(1, \frac{1}{\log n \log(1/\epsilon)} \right) & \text{if } k < N := (\log n) n^2 \lceil \log(1/\epsilon) \rceil \\ \frac{1}{(k+1-N)^{1.1}} & \text{if } k \geq N. \end{cases} \quad (55)$$

This sequence, which is admissible according to [47], has the advantage of a relatively slow decrease of the error terms, and it greatly simplifies our running time analysis. Then $\bar{\vartheta} \leq n^2 + 10$ and for a sufficiently large n ,

$$\vartheta_k \approx \frac{1}{\log n \log(1/\epsilon)}$$

for all iterations k before reaching ϵ -optimality. Therefore, choosing the sequence as defined by (55), the residual norms will decrease $1/(\log n \log(1/\epsilon))$ faster than τ_k .

At iteration k , we then choose ξ_k satisfying:

$$\xi_k \leq \frac{1}{\varrho} \min\{\|r_k^d\|, \|r_k^p\|, \|r_k^c\|\},$$

where ϱ is the maximum norm of a solution to the Newton linear system (24) corresponding to the II-QIPM. Setting ϑ_k according to (55), for sufficiently large n , we simply have $\vartheta_k = \frac{1}{\log n \log 1/\epsilon}$ before convergence. Hence, we can then take the error ξ_k to be:

$$\xi_k = \frac{\epsilon}{\varrho(\log n \log 1/\epsilon)}. \quad (56)$$

Lemma 15. *Choosing ξ_k according to (56), the inexact search direction computed at the end of step 4 of Algorithm 4 satisfies (23).*

Proof. For infeasible IPMs, the termination condition requires the feasibility gap be less than or equal to ϵ , and we scale down by ϱ due to normalization. \square

Theorem 6. *Let $\epsilon > 0$ be a small constant. Suppose that the conditions (15), (16a) and (16b) hold. Suppose further that ξ_k is chosen according to (49). Then, after $k = \mathcal{O}(n^2 \ln(1/\epsilon))$ iterations of Algorithm 4, we have $\theta_k \leq \epsilon$.*

Proof. Follows from [47, Theorem. 1], after noting that, as a consequence of Lemma 15, the inexact search direction used in Algorithm 4 satisfies the conditions of [47]. \square

By the definition of the neighborhood of the infeasible central path, \mathcal{N}_I , we have $\tau_k \leq \theta_k$ for every iteration k . In view of Theorem 6, after $k = \mathcal{O}(n^2 \ln(1/\epsilon))$ iterations we have $\theta_k \leq \epsilon$, which implies $\tau_k \leq \epsilon$. Accordingly, the infeasible central path parameter has been sufficiently reduced to obtain an ϵ -optimal solution.

Having established convergence of both the IF- and II-QIPMs, we now turn our attention to their overall running times, this is discussed next.

7 Complexity

In this section, we formally analyze the worst case overall running times of the IF- and II-QIPMs. The per-iteration cost of either QIPM can be generally written as

$$T_{iter} = T_{NT} + T_{progress},$$

where T_{NT} is the quantum gate complexity of obtaining a classical solution to the Newton linear system (29), and $T_{progress}$ is the number of classical arithmetic operations required to prepare the next iteration. Thus, upon bounding T_{iter} for the IF- and II-QIPMs one can obtain the worst-case overall complexity of each algorithm using the iteration bounds that were established in the previous section.

We conclude this section by comparing the performance of our methods to known classical and quantum algorithms for SDO from the literature, before discussing the impact of dependence on the condition number bound κ on the overall running time of QIPMs.

7.1 Running time of the IF-QIPM

The following result establishes per-iteration cost of the IF-QIPM.

Proposition 16. *Each iteration k of the IF-QIPM in Algorithm 3 has a quantum gate complexity of*

$$\mathcal{O}\left(n^3 \frac{\kappa^2}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

and requires $\mathcal{O}(n^4 \log(\kappa))$ classical arithmetic operations.

Proof. In every iteration, we need to prepare and solve the NT Newton linear system (29), and obtain a classical estimate of the quantum state encoding its solution. Furthermore, we apply the state tomography algorithm to obtain a classical description of the solution of the Newton linear system; we denote the corresponding running time by $T_{TO}(T_{LS}, \xi_k)$ for precision ξ_k . For simplicity and without loss of generality, we consider the tomography precision ξ_k but neglect the QLSA precision, as only the tomography precision appears polynomially in the final expression for the iteration cost. Applying Theorem 2 with $d = n^2$ and $T_U = T_{LS}$, running the tomography step to precision ξ_k chosen according to (49) implies

$$T_{NT}^{IF} = T_{TO}(T_{LS}, \xi_k) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi_k}}\left(\frac{n^2}{\xi_k} T_{LS}\right) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi_k}}\left(n^2 \frac{\varrho}{\epsilon} T_{LS}\right) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}\left(n^3 \frac{\kappa^2}{\epsilon}\right).$$

QRAM accesses are required to prepare the Newton linear system and obtain a classical estimate of the solution at iteration k .

To progress to the iteration, we must compute the primal and dual search directions ΔX and ΔS from the nullspace representation provided in equations (28a)-(28b), and update the solution (X, y, S) to the SDO before proceeding to the next iteration. We must also compute our scaling matrix P and its inverse. Computing ΔX and ΔS using (28a)-(28b) amounts to classically computing matrix-vector products in which the dimension is n^2 , and hence requires $\mathcal{O}(n^4)$ arithmetic operations, while updating the solution is matrix addition involving $n \times n$ matrices, and can be accomplished using $\mathcal{O}(n^2)$ arithmetic operations. Finally, noting that all of the matrices involved in the computation of P have size $n \times n$, we can compute both P and P^{-1} using $\mathcal{O}(n^\omega)$ arithmetic operations. Summarizing, we have a total of

$$T_{progress}^{IF} = \mathcal{O}(n^4 + n^2 + n^\omega) = \mathcal{O}(n^4),$$

arithmetic operations at each iterate, as computing ΔX and ΔS from our estimate of the solution to (29) is the dominant operation.

Note however, at each iterate the QLSA circuit requires the condition number of the Newton linear system coefficient matrix M_{NT} (or, an upper bound on it) as part of the input. Computing the condition number classically is too costly, and so alternatively we use the following scheme to ensure correctness of steps 4 and 5 in Algorithm 3 (i.e., to ensure that $\tilde{\kappa}$ was chosen large enough). Use $\tilde{\kappa} = 1$ as a starting guess, and then

1. Execute steps 4 and 5 of Algorithm 3 with $\tilde{\kappa}$ as input
2. Use classical estimate of the solution to compute

$$\varepsilon = \left\| M_{NT} \begin{bmatrix} \overline{\Delta z_k} \\ \overline{\Delta y_k} \end{bmatrix} - \mathbf{svec}(R_k^c) \right\|_2$$

3. If $\varepsilon \leq \varrho \xi_k$, then **accept** $(\overline{\Delta z_k}, \overline{\Delta y_k})$ and terminate loop. Otherwise, set $\tilde{\kappa} = 2 \cdot \tilde{\kappa}$ and **go to Step 1**.

One can observe that ε can be computed using $\mathcal{O}(n^4)$ arithmetic operations, and at most $\log(\kappa)$ such verification steps will be necessary in the worst case. Summarizing, we arrive at a per-iteration cost of $\log(\kappa) T_{NT}^{IF} = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}\left(n^3 \frac{\kappa^2}{\epsilon}\right)$ QRAM accesses and $\mathcal{O}(n^4 \log(\kappa))$ classical arithmetic operations, which completes the proof. \square

We are now in a position to bound the overall running time of our IF-QIPM given in Algorithm 3, which is formalized in the following result.

Corollary 2. *A quantum implementation of Algorithm 3 with access to QRAM outputs an ϵ -optimal solution (X^*, y^*, S^*) to the primal-dual SDO pair (1)-(2) using at most*

$$\mathcal{O}\left(n^{3.5} \frac{\kappa^2}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

QRAM accesses and $\mathcal{O}\left(n^{4.5} \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right)$ arithmetic operations.

Proof. By Corollary 1, the IF-QIPM requires $\mathcal{O}(\sqrt{n} \log(1/\epsilon))$ iterations to converge to an ϵ -optimal solution. Hence, letting χ be a nonnegative constant and applying Proposition 16, it follows that Algorithm 3 requires at most

$$\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}\left(\sum_{k=1}^{\chi \sqrt{n} \log(1/\epsilon)} \left(n^3 \frac{\kappa^2}{\epsilon}\right)\right) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}\left(\sqrt{n} \left(n^3 \frac{\kappa^2}{\epsilon}\right)\right),$$

QRAM accesses, and $\mathcal{O}\left(\sum_{k=1}^{\chi \sqrt{n} \log(1/\epsilon)} n^4 \log(\kappa)\right) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}(n^{4.5})$ arithmetic operations, which completes the proof. \square

Recall that LOPs are simply a special instance of SDO in which all of the matrices are diagonal. Hence, our algorithm can also be applied to these problems as well.

Corollary 3. *A quantum implementation of Algorithm 3 with access to QRAM outputs an ϵ -optimal solution $(X^*, y^*, S^*) = (\text{diag}(x^*), y^*, \text{diag}(s^*))$ to a Linear Optimization problem using at most*

$$\mathcal{O}\left(n^2 \frac{\kappa^2}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

QRAM accesses and $\mathcal{O}\left(n^{2.5} \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right)$ arithmetic operations.

Proof. The quantum gate complexity can be obtained upon repeating the analysis we used in the proof of Corollary 2, noting that for LO the dimension of the problem is $d = n$ instead of $d = n^2$. Then, recall that in the case of LO, all of the matrices are diagonal, and therefore no symmetrization is required. Hence, the only classical progression steps in this case are computing the primal and dual search directions from their nullspace representation, and subsequently updating the solution, and each of these steps require $\mathcal{O}(n^2)$ arithmetic operations. \square

Since our IF-QIPM utilizes a hybrid quantum-classical scheme, one can trivially obtain a fully classical IF-IPM by replacing the quantum subroutine for solving the Newton linear system with an inexact classical linear systems algorithm. Note that the system we obtain in (29) using the nullspace representation is not guaranteed to be positive semidefinite, so we cannot directly apply the Conjugate Gradient Method (CG) to solve it. For $M \in \mathbb{R}^{d \times d}$ and $v \in \mathbb{R}^d$ with $M \notin \mathcal{S}_+^n$, one could first symmetrize the system $Mu = v$ as

$$M^\top Mu = M^\top v,$$

and subsequently apply CG, in which case the cost of this step would be

$$\mathcal{O}\left(d^\omega + d^2 \sqrt{\kappa_M^2} \log(1/\xi)\right) = \mathcal{O}\left(d^\omega + d^2 \kappa_M \log(1/\xi)\right),$$

because we need to carry out matrix-matrix multiplication, and the condition number of the resulting coefficient matrix is κ_M^2 . Alternatively, we could employ one of the generalizations of CG such as the Generalized Minimum Residual Method (GMRES), however, not only is the coefficient matrix we employ indefinite, but it is nonsymmetric and dense: applying GMRES⁴ to solve a

⁴We do not review CG or GMRES in detail, and instead refer the reader to Chapter 6 of [72].

dense, nonsymmetric and indefinite system of dimension d to precision ξ leads to a complexity of $\mathcal{O}(d^3 \log(1/\xi))$. In our case $d = n^2$, so classically performing this step will require time $\mathcal{O}(n^{2\omega} + n^4 \kappa \log(1/\xi))$ if we symmetrize and use CG, or time $\mathcal{O}(n^6 \log(1/\xi))$ using GMRES.

This outcome is undesirable because it would lead to a slower algorithm than what already exists in the classical literature. Alternatively, one can use a polynomial approximation q , of $1/u$ on the interval $[1/\kappa_M^2, 1]$. The solution would then be $q(M^2)Mv$, which is approximately $M^{-2}Mv = M^{-1}v$. Such a polynomial q with degree roughly $\kappa_M \log(1/\xi)$ exists (see, e.g., [72, Section 6.11]), and the approximate solution would be $q(M^2)Mv$, which is close to $M^{-1}v$ whenever the linear system $Mu = v$ has a solution. Crucially, $q(M^2)Mv$ can be obtained using $2 \cdot \deg(q) + 1$ matrix-vector products, and does not require computing (or even writing down) the matrix M^2 , thus avoiding the $n^{2\omega}$ present term in the complexity associated with symmetrization and subsequently using CG. Rather, in our case the complexity becomes

$$\mathcal{O}(d^2 \sqrt{\kappa_M} \log(1/\xi)) = \mathcal{O}\left((n^2)^2 \sqrt{\kappa^2} \log(1/\xi)\right) = \mathcal{O}(n^4 \kappa \log(1/\xi)).$$

For an explicit discussion of this method, we refer the reader to Section 16.5 of the monograph [73].

This argument is the basis for the next result, which bounds the running time of the classical IF-IPM for SDO.

Corollary 4. *A classical implementation of Algorithm 3 obtains an ϵ -optimal solution (X^*, y^*, S^*) to the primal-dual SDO pair (1)-(2) with overall complexity*

$$\mathcal{O}\left(n^{4.5} \kappa \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right).$$

Proof. Simply note that applying [73, Theorem 16.6] with $d = n^2$ one has

$$T_{NT}^{IF} = \mathcal{O}\left(n^4 \kappa \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right),$$

and we still have $T_{progress}^{IF} = \mathcal{O}(n^4)$. Just as in the case of the IF-QIPM, the linear system solver we use for our IF-IPM takes the condition number of the Newton linear system as input. Using the verification scheme described in the proof of Proposition 16 implies that we might need to repeat these steps $\log(\kappa)$ times to ensure the solution we obtain is accepted. The stated result then follows upon accounting for the iteration bound provided by Corollary 1. \square

Corollary 5. *A classical implementation of Algorithm 3 obtains an ϵ -optimal solution $(X^*, y^*, S^*) = (\text{diag}(x^*), y^*, \text{diag}(s^*))$ to a Linear Optimization problem with overall complexity*

$$\mathcal{O}\left(n^{2.5} \kappa \cdot \text{polylog}\left(\kappa, \frac{1}{\epsilon}\right)\right).$$

7.2 Running time of the II-QIPM

Similar to our analysis for the IF-QIPM, we begin by formally stating the per-iteration cost of the II-QIPM.

Proposition 17. *Every iteration k of the II-QIPM in Algorithm 4 has a quantum gate complexity of*

$$\mathcal{O}\left(n^{3.5} \kappa \kappa_{\mathcal{A}} \rho (\|\mathcal{A}\|_F + \|S\|_F) \frac{1}{\epsilon} \cdot \text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right)$$

and requires $\mathcal{O}(n^\omega \log(\kappa))$ classical arithmetic operations.

Proof. In each iteration of the II-QIPM, we prepare and solve the HKM linear system; and let T_{LS}^{II} denote the time required to do so. As in the IF-QIPM, we utilize vector state tomography algorithm from [63] to obtain a classical description of the solution of the Newton linear system (24); and similar to the proof of Proposition 16, denote the corresponding running time by $T_{TO}(T_{LS}^{II}, \xi_k)$

for precision ξ_k . Applying Proposition 23, we have $T_{LS}^{II} = \tilde{\mathcal{O}}_{n,\kappa}(\kappa(\|\mathcal{A}\|_F + \|S\|_F))$, and hence, we have:

$$T_{NT}^{II} = T_{TO}(T_{LS}^{II}, \xi_k) = \tilde{\mathcal{O}}_{n,\kappa, \frac{1}{\xi_k}} \left(\frac{n^2}{\xi_k} T_{LS}^{II} \right) = \tilde{\mathcal{O}}_{n,\kappa, \frac{1}{\epsilon}} \left(\frac{n^2}{\epsilon} \varrho \left(\kappa(\|\mathcal{A}\|_F + \rho n^{1.5}) \right) \right).$$

From here, we point out that for II-(Q)IPMs, $\varrho = \mathcal{O}(\kappa_{\mathcal{A}} \rho n^{1.5})$; this is a consequence of the known bounds $\|\Delta X\|, \|\Delta S\| = \mathcal{O}(\rho n)$ (see, e.g., [47]), the relationship between spectral norm and Frobenius norm, and the definition of the neighborhood of the central path (18). We therefore have

$$T_{NT}^{II} = \tilde{\mathcal{O}}_{n,\kappa, \frac{1}{\epsilon}} \left(\frac{n^2}{\epsilon} \kappa_{\mathcal{A}} \rho n^{1.5} \left(\kappa(\|\mathcal{A}\|_F + \|S\|_F) \right) \right) = \tilde{\mathcal{O}}_{n,\kappa, \frac{1}{\epsilon}} \left(n^{3.5} \kappa_{\mathcal{A}} \rho \left(\|\mathcal{A}\|_F + \rho n^{1.5} \right) \frac{1}{\epsilon} \right),$$

as $\|S\|_F = \mathcal{O}(\rho n^{1.5})$ by (20).

We must also update the solution to the SDOP, and compute our scaling matrix P and its inverse. Updating the solution is again standard matrix addition involving $n \times n$ matrices, and can be accomplished using $\mathcal{O}(n^2)$ arithmetic operations, and the computation of $P = S^{1/2}$ and its inverse can be performed using $\mathcal{O}(n^\omega)$ arithmetic operations, as all of the involved matrices are of size $n \times n$.

Just as in the case of the IF-QIPM, the QLSA we use for our II-QIPM takes the condition number of the Newton linear system as input. Using the scheme described in the proof of Proposition 16 implies that we might need to repeat these steps (both quantum and classical) $\log(\kappa)$ times to ensure the solution we obtain is accepted. Thus, each iteration of Algorithm 4 has a quantum gate complexity of

$$\mathcal{O} \left(n^{3.5} \kappa_{\mathcal{A}} \rho \left(\|\mathcal{A}\|_F + \rho n^{1.5} \right) \frac{1}{\epsilon} \cdot \text{polylog} \left(n, \kappa, \frac{1}{\epsilon} \right) \right)$$

and requires $\mathcal{O}(n^\omega \log(\kappa))$ classical arithmetic operations. The proof is complete. \square

Combining the preceding result with the iteration bound in Theorem 6 immediately gives the overall running time of the II-QIPM, which is formalized next.

Corollary 6. *A quantum implementation of Algorithm 4 with access to QRAM converges to an ϵ -optimal solution to the primal-dual SDO pair (1)-(2) in time*

$$\mathcal{O} \left(n^{5.5} \kappa_{\mathcal{A}} \rho \left(\|\mathcal{A}\|_F + \rho n^{1.5} \right) \frac{1}{\epsilon} \cdot \text{polylog} \left(n, \kappa, \frac{1}{\epsilon} \right) \right).$$

Proof. We can begin by expanding the expression for the per-iteration cost of the algorithm provided in Proposition 17. Then, noting that by Theorem 6, the II-QIPM requires $\mathcal{O}(n^2 \ln 1/\epsilon)$ iterations, we can express the total running time of the algorithm as

$$\sum_{k=1}^{\chi n^2 \log(1/\epsilon)} n^{3.5} \kappa_{\mathcal{A}} \rho \left(\|\mathcal{A}\|_F + \rho n^{1.5} \right) \frac{1}{\epsilon} = \mathcal{O} \left(n^{5.5} \kappa_{\mathcal{A}} \rho \left(\|\mathcal{A}\|_F + \rho n^{1.5} \right) \frac{1}{\epsilon} \cdot \text{polylog} \left(n, \kappa, \frac{1}{\epsilon} \right) \right),$$

and the proof is complete. \square

7.3 Comparison with existing algorithms for SDO

Before proceeding further, we remind the reader of a number of key differences between (Q)IPMs and (Q)MWU algorithms. The output of (Q)IPMs is a classical primal-dual pair (X, y, S) such that

$$\begin{aligned} A^{(i)} \bullet X &= b_i \quad \forall i \in [m], \quad X \succ 0 \\ \sum_{i \in [m]} y_i A^{(i)} + S &= C \quad S \succeq 0 \end{aligned}$$

with

$$\frac{X \bullet S}{n} \leq \epsilon.$$

Conversely, primal-dual (Q)MWUs output a dual solution $y \in \mathbb{R}^m$ such that setting

$$X = R \cdot \frac{\exp\left(-\sum_{i=1}^m y_i A^{(i)}\right)}{\text{tr}\left(\exp\left(-\sum_{i=1}^m y_i A^{(i)}\right)\right)}$$

the primal-dual pair (X, y) satisfies

$$\begin{aligned} \left|A^{(i)} \bullet X - b_i\right| &\leq \epsilon_{\text{abs}} \quad \forall i \in [m], \quad X \succ 0 \\ S = C - \sum_{i \in [m]} y_i A^{(i)} &\succeq -\epsilon_{\text{abs}} I \end{aligned}$$

and the objective value attained by this solution is $\text{OPT} \in [\varsigma - \mathcal{O}(\epsilon_{\text{abs}}), \varsigma + \mathcal{O}(\epsilon_{\text{abs}})]$, where ς is a bound on the optimal objective value determined using binary search. Observe that this is another distinction between (Q)IPMs and (Q)MWUs; the output of our QIPMs always satisfies primal and dual feasibility exactly, whereas the solution obtained by (Q)MWU methods satisfies primal and dual feasibility approximately.

The QMMWUs (specifically, all papers by Brandao et al. [27, 50, 74], as well as the papers by van Apeldoorn et al. [28, 30] also assume that the SDO problem data is normalized:

$$\|A^{(1)}\|, \dots, \|A^{(m)}\|, \|C\| \leq 1.$$

Our QIPMs do not require this assumption. For the QIPMs presented in this work, only the Newton linear system must be normalized in each iteration (so we can apply a QLSA to solve it), and this normalization is accounted for in our accuracy requirements and the subsequent running time analysis. (For classical MMWUs, the choice of the normalization can vary depending on the papers; the analysis of the classical algorithm in [28] assumes the above normalization.)

Assuming the data is normalized with respect to the operator norm means that for more general problems (i.e., without normalized data) the error scales as $\theta \epsilon_{\text{abs}}$, where

$$\theta = \max \left\{ \|A^{(1)}\|, \dots, \|A^{(m)}\|, \|C\| \right\}.$$

Thus, for MMWUs one would lose a factor θ in the precision. Note that being able to compute θ to normalize the data as $\theta^{-1}A^{(1)}, \dots, \theta^{-1}A^{(m)}, \theta^{-1}C$ could require classical access to these matrices, and an additional $\mathcal{O}(mns)$ cost in the running time to compute θ .

Table 1 presents the running time of our algorithms alongside well-known algorithms for solving SDOPs. For ease of comparison, we assume that $m = \mathcal{O}(n^2)$ and that the problem is fully dense (i.e., $s = n$). It can be readily seen that the II-QIPM is worse in every parameter when compared to the other algorithms. Conversely, the IF-QIPM obtains a speedup with respect to the dimension n over each of the listed classical solution methodologies. Yet, the IF-QIPM exhibits a quadratic dependence on the condition number bound κ , and linear dependence on the inverse precision, implying an asymptotically worse running time overall.

References	Method	Runtime
[23–25]	IPM	$\tilde{\mathcal{O}}_{n, \frac{1}{\epsilon}}(n^{6.5})$
[22]	IPM	$\tilde{\mathcal{O}}_{\frac{1}{\epsilon}}(n^{5.246})$
[20, 21]	CPM	$\tilde{\mathcal{O}}_{n, R, \frac{1}{\epsilon}}(n^6)$
[30]	QMWU	$\tilde{\mathcal{O}}_{n, \frac{1}{\epsilon_{\text{abs}}}} \left(\left(n^2 + n^{1.5} \frac{Rr}{\epsilon_{\text{abs}}} \right) \left(\frac{Rr}{\epsilon_{\text{abs}}} \right)^4 \right)$
This work (Quantum)	IF-QIPM	$\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}} \left(\sqrt{n} \left(n^3 \frac{\kappa^2}{\epsilon} + n^4 \right) \right)$
This work (Classical)	IF-IPM	$\tilde{\mathcal{O}}_{\kappa, \frac{1}{\epsilon}}(n^{4.5} \kappa)$
This work (Quantum)	II-QIPM	$\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}(n^{5.5} \kappa \kappa_{\mathcal{A}} \rho (\ \mathcal{A}\ _F + \rho n^{1.5}) \frac{1}{\epsilon})$

Table 1: Total running times for classical and quantum algorithms to solve (1)-(2) with $m = \mathcal{O}(n^2)$.

When compared to the QMWU algorithm, it can be observed that the QIPM achieves a favorable dependence on the error. Although the QMWU is faster with respect to dimension, the QMWU has polynomial dependence on the trace bound R , and for certain applications this may negate any speedup, such as the SDO approximation of Quadratic Unconstrained Binary Optimization (QUBO) problems, i.e., problems of the form

$$\max_{x \in \{-1, 1\}^n} x^\top C x. \quad (57)$$

QUBOs play an important role in quantum information sciences, and the resulting SDO approximation is given by

$$\max_{X \in \mathcal{S}^n} \{C \bullet X : \text{diag}(X) = e, X \succeq 0\}, \quad (58)$$

where e is the all ones vector of dimension n . Hence, any optimal solution of (58) satisfies $\text{tr}(X) = n$, suggesting that for these instances the worst case running time of the QMWU is

$$\tilde{\mathcal{O}}_{n, \frac{1}{\epsilon_{\text{abs}}}} \left(n^{6.5} \left(\frac{r}{\epsilon_{\text{abs}}} \right)^5 \right),$$

as $R = n$ and $m = n$ for (58). Thus, the IF-QIPM could outperform the QMWU for this class of SDOPs, provided that $\kappa = \mathcal{O} \left(nr^{2.5} \left(\frac{1}{\epsilon} \right)^2 \right)$.

For SDOPs with diagonal constraints such as (58), the Arora-Kale method can be specialized (see, e.g., [75]) to obtain an $\tilde{\mathcal{O}}(n\epsilon^{-3.5})$ algorithm. Note that to achieve this running time, the algorithm avoids explicitly computing or reporting the primal solution X , and instead makes use of techniques to estimate its diagonal entries and trace inner products of the form $A \bullet X$. As an alternative, these methods report a “gradient” $G \in \mathcal{S}^n$ such that $X = W \exp(G)W$ for a diagonal matrix W . While the algorithm of Lee and Padmanabhan [75] offers speedups with respect to the dimension, its running time is exponentially slower than classical IPMs with respect to the inverse precision.

To obtain an additive ϵ -convergence using the algorithm found in [75], the error parameter would have to be set to

$$\tilde{\epsilon} = \frac{\epsilon}{\|C\|_{\ell_1}}.$$

When C is dense and $\|C\|_{\ell_1} = \mathcal{O}(n)$, the running time of their algorithm therefore scales as

$$\mathcal{O}(n^2(\tilde{\epsilon})^{-3.5}) = \mathcal{O}(n^2(n/\epsilon)^{3.5}) = \mathcal{O}(n^{5.5}\epsilon^{-3.5}).$$

For our QIPM to reach duality gap ϵ would require complexity

$$\mathcal{O} \left(\sqrt{n} \left(n^3 \kappa^2 \frac{1}{\epsilon/n} + n^4 \right) \text{polylog}(n, \epsilon^{-1}, \kappa) \right) = \tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}} \left(n^{4.5} \frac{\kappa^2}{\epsilon} \right).$$

It can then be seen that in this case, our QIPM scales better than the classical MWU in n and ϵ^{-1} in this case, but depends linearly on κ . However, one could solve the QUBO problem to ϵ -optimality using the IPM from Jiang et al. [22] in time $\mathcal{O}(n^{\omega+0.5} \log(n/\epsilon))$, or even our classical IF-IPM with overall running time $\tilde{\mathcal{O}}_{\kappa, \frac{n}{\epsilon}}(n^{4.5}\kappa)$.

One potential use case of our quantum and classical IF-IPMs would be as a limited precision oracle in an iterative refinement method. Iterative refinement (IR) methods are used to solve optimization problems to extended precision by using (possibly fixed) low-precision calls to an optimization subroutine (such as an IPM) to solve a sequence of simpler problems [52–54]. In this setting, low-precision solutions are completely acceptable, and since both the classical and quantum IF-IPMs we propose classically report the solution, the IF-QIPM would seem to be the preferable choice of subroutine.

However, the behavior of κ as we traverse the central path towards optimality poses a significant hurdle to the performance of QIPMs. This is highlighted in detail next.

7.4 The relationship between κ and optimality in IPMs

Noting the presence of the condition number κ in our running time results for the QIPMs, it is of crucial importance to determine whether or not bounds on κ can be established. This is of particular importance as the coefficient matrices M_{NT} change at each iteration of the QIPMs. It has been well studied that without any numerical safeguards, in theory the condition number κ will go to ∞ as we approach the optimal set to the optimization problem at hand. In what follows, we highlight how the condition number κ is related to the optimality gap ϵ , and also note that ξ , the precision of the quantum linear solver has to be also in the order of ϵ or less.

To keep our discussion simple, let us first consider the case of LO. Following Roos, Terlaky and Vial [76] for the LO case, let

$$\begin{aligned} B &= \{i : x_i > 0 \text{ for some optimal solutions}\}, \\ N &= \{i : s_i > 0 \text{ for some optimal solutions}\}, \end{aligned}$$

where x and s denote the diagonals of X and S . Then, (B, N) forms the so called *optimal partition* of the LO case of (1). It is known that

$$B \cup N = \{1, \dots, n\}.$$

Further, letting \mathcal{SP}^* denote the primal-dual optimal set, we define the quantities

$$\begin{aligned} \sigma_{\mathcal{SP}^*}^x &:= \min_{i \in B} \max_{x \in \mathcal{SP}^*} \{x_i\}, \\ \sigma_{\mathcal{SP}^*}^s &:= \min_{i \in N} \max_{x \in \mathcal{SP}^*} \{s_i\}, \end{aligned}$$

where $\sigma_{\mathcal{SP}^*}^x = \infty$ if $B = \emptyset$ and $\sigma_{\mathcal{SP}^*}^s = \infty$ if $N = \emptyset$. By the boundedness of the optimal set \mathcal{SP}^* , it follows that $\sigma_{\mathcal{SP}^*}^x$ is finite if B is nonempty, and similarly $\sigma_{\mathcal{SP}^*}^s$ is finite if N is nonempty. As Roos, Terlaky and Vial [76] note, by the definition of B and N , at least one must be nonempty. Therefore, taking

$$\sigma_{\text{SP}} = \min\{\sigma_{\mathcal{SP}^*}^x, \sigma_{\mathcal{SP}^*}^s\},$$

it follows that the condition number of the LO problem must be positive and finite. As we follow the central path, for the case of LO we have $s_i x_i = \nu$ for all i . As we approach optimality we have

Lemma 16 (Lemma I.43 in [76]). *For any $\nu > 0$,*

$$\begin{aligned} x_i(\nu) &\geq \frac{\sigma_{\text{SP}}}{n}, \quad \forall i \in B, & x_i(\nu) &\leq \frac{n\nu}{\sigma_{\text{SP}}}, \quad \forall i \in N, \\ s_i(\nu) &\leq \frac{n\nu}{\sigma_{\text{SP}}}, \quad \forall i \in B, & s_i(\nu) &\geq \frac{\sigma_{\text{SP}}}{n}, \quad \forall i \in N. \end{aligned}$$

As in the Newton system of LO, at each iteration we need to calculate the coefficient matrix $A^\top X S^{-1} A$ of the Newton system in compact form (see e.g., [76]). The condition number of this matrix will be in the order of the largest element of the diagonal matrix $X S^{-1}$ divided by the smallest diagonal element. Thus, considering the bounds of Lemma 16, we have that this ratio is at least

$$\left(\frac{\sigma_{\text{SP}}}{n} \right) / \left(\frac{n\nu}{\sigma_{\text{SP}}} \right) = \left(\frac{\sigma_{\text{SP}}^2}{n^2\nu} \right) / \left(\frac{n^2\nu}{\sigma_{\text{SP}}^2} \right) = \frac{\sigma_{\text{SP}}^4}{n^4\nu^2},$$

which clearly goes to infinity as $\nu \rightarrow 0$, as the QIPM approaches the optimal set. It is also known that the condition number σ of the LO problem can be as small as 2^L if the problem data is integer, see, e.g., [76].

Now, let us consider, as customary in the classical setting [76], that all of the data of the LO problem is integer. Then, the binary encoding of the LO problem requires at most

$$L = \sum_i \sum_j (1 + \log_2(|a_{ij}| + 1)) + \sum_i (1 + \log_2(|b_i| + 1)) + \sum_j (1 + \log_2(|c_j| + 1))$$

bits. It is also known that $\delta \geq 2^{-L}$ and by [76, Theorem I.53], that if $\epsilon < \frac{\sigma_{\text{SP}}^5}{4n^{1.5}}$ then an exact optimal solution can be identified by solving a linear system of equations. For simplicity, the bound given here is not the tightest possible bound, some quantities in the bound of [76, Theorem I.53] are estimated by σ_{SP} .

Another way to look at it, is to estimate the total running time to reach a near-optimal primal-dual feasible solution pair, with optimality gap less than ϵ . To reach this precision, we need to have

$$n\nu < \epsilon < \frac{\sigma_{\text{SP}}^5}{4n^{1.5}},$$

i.e. $\nu = \mathcal{O}\left(\frac{\sigma_{\text{SP}}^5}{4n^{1.5}}\right)$. Thus we can identify an exact optimal solution of the LO problem if

$$\nu = \mathcal{O}\left(\frac{2^{-5L}}{n^{2.5}}\right).$$

Then, we have

$$\kappa \geq \frac{\sigma_{\text{SP}}^4}{n^4 \nu^2} \geq \frac{\sigma_{\text{SP}}^4}{n^4 \frac{\sigma_{\text{SP}}^{10}}{n^5}} = n\sigma_{\text{SP}}^{-6} = \Omega(n2^{6L}).$$

We also need

$$\xi < \epsilon < c_0,$$

where $c_0 = \mathcal{O}\left(\frac{2^{-5L}}{n^{2.5}}\right)$.

By Corollary 3, when applied to LOPs, the IF-QIPM has a worst case overall running time of

$$\mathcal{O}\left(\sqrt{n}\left(n^{1.5}\frac{\kappa^2}{\epsilon} + n^2\right)\text{polylog}\left(n, \kappa, \frac{1}{\epsilon}\right)\right).$$

Substituting all of these in the total running time bound of our IF-QIPM yields

$$\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}\left(n^{1.5}\frac{n^2 2^{12L}}{\frac{2^{-5L}}{n^{2.5}}} + n^{2.5}\right) = \tilde{\mathcal{O}}_{n, \kappa}(n^{6.5} 2^{17L} L)$$

as the time complexity of our QIPM to identify an exact solution, compared to the $\mathcal{O}(n^3 L)$ arithmetic operations needed for the classical IPM to do the same. Carrying out similar calculations for the II-QIPM yields a worst case running time bound of

$$\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\epsilon}}(n^{4.5} \kappa \kappa_{\mathcal{A}\rho} (\|\mathcal{A}\|_F + \rho n^{1.5})) = \tilde{\mathcal{O}}_{n, \kappa}(n^{5.5} \kappa_{\mathcal{A}\rho} (\|\mathcal{A}\|_F + \rho n^{1.5}) 2^{6L} L)$$

when applied to LOPs.

As LO is a special case of SDO, the general bound for the condition number of the Newton linear systems (24) and (29) are bounded below by the condition number bound of LO problems. For the condition number of SDO problems, we refer to Mohammad-Nezhad and Terlaky [77]. Let

$$\begin{aligned} \mathcal{B} &= \mathcal{R}(X^*), \\ \mathcal{N} &= \mathcal{R}(S^*), \end{aligned}$$

where $\mathcal{R}(\cdot)$ denotes the range space and (X^*, y^*, S^*) denotes a maximally complementary solution. Define $n_{\mathcal{B}} = \dim(\mathcal{B})$ and $n_{\mathcal{N}} = \dim(\mathcal{N})$. Then, letting $Q = [Q_{\mathcal{B}}, Q_{\mathcal{T}}, Q_{\mathcal{N}}]$ be orthonormal bases for \mathcal{B} , \mathcal{N} and \mathcal{T} , respectively, it follows from [77] that for every primal-dual optimal solution $(X, y, S) \in \mathcal{P}^* \times \mathcal{D}^*$, one can write X and S as:

$$\begin{aligned} X &= Q_{\mathcal{B}} U_X Q_{\mathcal{B}}^{\top} \\ S &= Q_{\mathcal{N}} U_S Q_{\mathcal{N}}^{\top}, \end{aligned}$$

where U_X and U_S are symmetric $n_{\mathcal{B}} \times n_{\mathcal{B}}$ and $n_{\mathcal{N}} \times n_{\mathcal{N}}$ matrices, respectively. Then, similar to the result for linear optimization, defining

$$\sigma_{\mathcal{B}} = \max_{(X \in \mathcal{P}^*)} \lambda_{\min}(Q_{\mathcal{B}}^{\top} X Q_{\mathcal{B}}) = \max_{Q_{\mathcal{B}} \in \Gamma_{\mathcal{B}}} \max_{X \in \mathcal{P}^*} \lambda_{\min}(\bar{Q}_{\mathcal{B}}^{\top} X \bar{Q}_{\mathcal{B}}) \quad (59a)$$

$$\sigma_{\mathcal{N}} = \max_{(y, S) \in \mathcal{D}^*} \lambda_{\min}(Q_{\mathcal{N}}^{\top} S Q_{\mathcal{N}}) = \max_{Q_{\mathcal{N}} \in \Gamma_{\mathcal{N}}} \max_{(y, S) \in \mathcal{D}^*} \lambda_{\min}(\bar{Q}_{\mathcal{N}}^{\top} S \bar{Q}_{\mathcal{N}}) \quad (59b)$$

$$\sigma = \min\{\sigma_{\mathcal{B}}, \sigma_{\mathcal{N}}\} \quad (59c)$$

where $\Gamma_{\mathcal{B}}$ and $\Gamma_{\mathcal{N}}$ denote the sets of all orthonormal bases for \mathcal{B} and \mathcal{N} , respectively. Finally, we arrive at the following results from [77].

Lemma 17 (Lemma 2 in [77]). *The condition number σ is positive.*

Lemma 18 (Lemma 8 in [77]). *Let the SDO problems (P) and (D) be given by integer data, L denote the binary input length of the largest absolute value of the entries in b , C and $A^{(i)}$ for $i = 1, \dots, m$ and $\|\cdot\|_F$ be the Frobenius norm. Then for the condition number σ , we have*

$$\sigma \geq \min \left\{ \frac{1}{r_{\mathcal{P}^*} \sum_{i=1}^m \|A^{(i)}\|_F}, \frac{1}{r_{\mathcal{D}^*}} \right\}.$$

This implies that σ is positive for SDO as well, therefore $\kappa \rightarrow \infty$.

8 Conclusion

In this work we present two provably convergent quantum interior point methods for semidefinite optimization problems. They are building on recent advances in quantum linear system solvers. The quantization of classical interior point methods is the subject of several recent papers in the literature. We compare the theoretical performance of classical and quantum interior point methods with respect to various input parameters, concluding that our Inexact-Feasible QIPM provides a speedup in terms of the size of the problem. Yet, due to the dependence on the condition number of the QLSA, as well as a $1/\epsilon$ dependence picked up from repeated tomography, it is not faster than the classical small neighborhood IPM in the worst case. The dependence on the condition number in particular presents a significant obstacle toward a theoretical quantum speedup. In practice, the QIPM may still have an advantage if low-precision solutions are acceptable.

Acknowledgements

The authors would like to thank the anonymous referees of Quantum for very helpful comments that improved the presentation, and for pointing us to [73] to improve the running time of our classical IF-IPM.

This project has been carried out thanks to funding by the Defense Advanced Research Projects Agency (DARPA), ONISQ grant W911NF2010022, titled The Quantum Computing Revolution and Optimization: Challenges and Opportunities. This research also used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725. G. Nannicini is supported by ONR award # N000142312585.

A Block-encoding the Inexact-Infeasible Newton Systems

A.1 Implementing the Inexact-Infeasible Newton linear system for Members of the Monteiro-Zhang Family of Search Directions

For the version of our QIPM that uses the NT direction, we factorize the corresponding linear system matrix and construct block-encodings for its factors. Note that in principle we could construct a $(2, \tilde{\mathcal{O}}(1), \xi)$ -block-encoding for $W = S^{-1/2}(S^{1/2}XS^{1/2})^{1/2}S^{-1/2}$ in time $\tilde{\mathcal{O}}(\|S\|_F^4 \kappa_S^5 \kappa_X)$, starting from the individual factors S, X and with repeated applications of Propositions 4 and [56, Lemma 9 and 10]. However, due to the heavy dependence on the condition number, we choose to classically compute all the factors (including matrix powers), and store them in QRAM. This is more efficient: all the involved matrices are $n \times n$, so classical computation of matrix powers has negligible cost compared to the solution of the Newton linear system, which is of size $\mathcal{O}(n^2 \times n^2)$, and we can avoid increasing the dependence on the condition number. The following two propositions establish how one can construct block-encodings of these factors, and bound the corresponding running time.

Proposition 18. *Let*

$$M_1 = \begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Let \mathcal{A}_s be stored in QRAM. Then, a $(\|M_1\|_F, \mathcal{O}(\log n), \xi_{M_1})$ -block-encoding of M_1 can be constructed in time $\tilde{\mathcal{O}}_{\frac{n}{\xi_{M_1}}}(1)$.

Proof. The result follows directly from Proposition 1. \square

Proposition 19. *Let $P = W^{-1/2}$, where W is the Nesterov-Todd scaling matrix and*

$$M_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ P \otimes_s P^{-\top} S & 0 & 0 \end{pmatrix}.$$

Let P and $P^{-\top} S$, be stored in QRAM. Then, a $(\|M_2\|_F, \mathcal{O}(\log n), \xi_{M_2})$ -block-encoding of M_2 can be constructed in time $\tilde{\mathcal{O}}_{\frac{n}{\xi_{M_2}}}(1)$.

Proof. With P and $P^{-\top} S$ stored in QRAM, it follows from Proposition 7 that we can construct a $(\|P \otimes_s P^{-\top} S\|_F, \mathcal{O}(\log n), \xi_{M_2})$ -block-encoding of $P \otimes_s P^{-\top}$ in time $\tilde{\mathcal{O}}_{\frac{n}{\xi_{M_2}}}(1)$. M_2 can then be obtained using, e.g., Lemma 1 and product of block-encoded matrices to place $P \otimes_s P^{-\top}$ in the right position. \square

Proposition 20. *Let $P = W^{-1/2}$, where W is the Nesterov-Todd scaling matrix and*

$$M_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & PX \otimes_s P^{-\top} \end{pmatrix}.$$

If PX , and $P^{-\top}$ are stored in QRAM. Then, a $(\|M_3\|_F, \mathcal{O}(\log n), \xi_{M_3})$ -block-encoding of M_3 can be constructed in time $\tilde{\mathcal{O}}_{\frac{n}{\xi_{M_3}}}(1)$.

Proof. The proof follows the same argument as in the proof of Proposition 19. \square

Proposition 21. *Let $P = W^{-1/2}$, where W is the Nesterov-Todd scaling matrix. The Nesterov-Todd linear system matrix (14) can be written compactly as:*

$$M_{NT} = \begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ P \otimes_s P^{-\top} S & 0 & PX \otimes_s P^{-\top} \end{pmatrix}, \quad (60)$$

If \mathcal{A}_s , P , $P^{-\top}$, PX , and $P^{-\top} S$ are stored in QRAM, then a $(\|M_{NT}\|_F, \mathcal{O}(\log n), \xi/(\kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encoding of (60), can be constructed in time $\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi}}(1)$.

Proof. Carrying out the calculations $M_1 + M_2 + M_3$ shows that (60) corresponds to the Nesterov-Todd linear system. We construct the three following block-encodings:

$$M_1 = \begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad M_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ P \otimes_s P^{-\top} S & 0 & 0 \end{pmatrix} \quad M_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & PX \otimes_s P^{-\top} \end{pmatrix},$$

using Propositions 18-20. We choose the precision of this step so that we obtain $(\|M_1\|_F, \mathcal{O}(\log n), \xi/(\|M_1\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$, $(\|M_2\|_F, \mathcal{O}(\log n), \xi/(\|M_2\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$ and $(\|M_3\|_F, \mathcal{O}(\log n), \xi/(\|M_3\|_F \kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encodings, respectively, where κ refers to the condition number of (60), here and below. We add these two block-encodings together using Proposition 2, obtaining a $(\max\{\|M_1\|_F, \|M_2\|_F, \|M_3\|_F\}, \mathcal{O}(\log n), \xi/(\kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encoding of (60), in time $\tilde{\mathcal{O}}_{n, \kappa, \frac{1}{\xi}}(1)$. Since $\max\{\|M_1\|_F, \|M_2\|_F, \|M_3\|_F\} = \mathcal{O}(\|M_{NT}\|_F)$, we obtain the claimed result. \square

The construction described in Proposition 21 is only one of the possible ways to obtain a block-encoding of the Newton linear system corresponding to the Nesterov-Todd direction. We choose this specific decomposition because among the ones that we analyzed, it minimizes the amount of classical work while keeping the quantum gate complexity unchanged. We can then use this factorization to solve the Newton system.

Theorem 7. *Let $P = W^{-1/2}$, where W is the Nesterov-Todd scaling matrix. There is a quantum algorithm that given*

$$|-\eta_1 R_{s_0}^d \circ -\eta_1 R_{s_0}^p \circ \mathbf{svect}((1 - \eta_2)\tau\mu_0 I - H_P(XS))\rangle$$

and access to QRAM data structures encoding \mathcal{A}_s , P , $P^{-\top}$, PX , and $P^{-\top}S$, outputs a state ξ -close to $|\Delta X \circ \Delta y \circ \Delta S\rangle$ in time:

$$\tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(\kappa \max\{\|M_1\|_F, \|M_2\|_F, \|M_3\|_F\}),$$

using the NT direction. We can also output an estimate of $\|\Delta X \circ \Delta y \circ \Delta S\|$ with relative error δ by increasing the running time by a factor $\frac{1}{\delta}$.

Proof. This is a direct consequence of Proposition 21 and Theorem 1. \square

A.1.1 Implementing the HKM linear system

We now analyze the version of our II-QIPM that uses the HKM direction. As before, we could implement a $(2\kappa_S^{1/2}, \tilde{\mathcal{O}}(1), \xi_2)$ -block-encoding for $S^{1/2}XS^{1/2}$ in time $\tilde{\mathcal{O}}(\|S\|_F\kappa_S)$, using Proposition 1 and [56, Lemma 9 and 10], but it is more efficient to perform these computations classically — see the discussion in the previous subsection.

Proposition 22. *The HKM linear system matrix can be written compactly as:*

$$M_{HKM} = \begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ S^{1/2} \otimes_s S^{1/2} & 0 & S^{1/2}X \otimes_s S^{-1/2} \end{pmatrix}, \quad (61)$$

If $\mathcal{A}_s, S^{1/2}, S^{-1/2}, S^{1/2}X$, are stored in QRAM, then a $(\|M_{HKM}\|_F, \mathcal{O}(\log n), \xi/(\kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encoding of (61), can be constructed in time $\tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(1)$.

Proof. Follows the same proof as Proposition 21, as this is simply the case in which $P = S^{1/2}$. \square

Having established a compact factorization of the Newton linear system corresponding to the HKM direction, similar to our work for the NT direction, we may now use the factorization to solve the Newton system. The following two results establish the validity of our factorization and bound the running time for solving this system in a quantum setting, respectively.

Proposition 23. *Using the HKM direction $P = S^{1/2}$, there is a quantum algorithm, which given*

$$|-\eta_1 R_{s_0}^d \circ -\eta_1 R_{s_0}^p \circ \mathbf{svect}((1 - \eta_2)\tau\mu_0 I - H_P(XS))\rangle$$

and access to QRAM data structures encoding \mathcal{A}_s , P , $P^{-\top}$, PX , and $P^{-\top}S$, outputs a state ξ -close to $|\Delta X \circ \Delta y \circ \Delta S\rangle$ in time

$$\tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(\kappa(\|\mathcal{A}\|_F + \|S\|_F)).$$

We can also output an estimate of $\|\Delta X \circ \Delta y \circ \Delta S\|$ with relative error δ by increasing the running time by a factor of $\frac{1}{\delta}$.

Proof. Observe, adding (i.e., a linear combination of) M_1 , M_2 and M_3 yields

$$\begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ S^{1/2} \otimes_s S^{1/2} & 0 & S^{1/2}X \otimes_s S^{-1/2} \end{pmatrix}.$$

Then, the result follows from Theorem 7. \square

A.1.2 Implementing the AHO linear system

We now analyze the version of our II-QIPM that uses the AHO direction.

Proposition 24. *The AHO linear system matrix can be written compactly as:*

$$M_{AHO} = \begin{pmatrix} 0 & \mathcal{A}_s^\top & \mathcal{I} \\ \mathcal{A}_s & 0 & 0 \\ I \otimes_s S^{1/2} & 0 & X \otimes_s I \end{pmatrix}, \quad (62)$$

If $\mathcal{A}, S^{1/2}, S^{-1/2}, XS^{1/2}, S^{1/2}X$, are stored in QRAM, then a $(\|M_{HKM}\|_F, \mathcal{O}(\log n), \xi/(\kappa^2 \log^2 \frac{\kappa}{\xi}))$ -block-encoding of (62), can be constructed in time $\tilde{\mathcal{O}}_{n,\kappa,\frac{1}{\xi}}(1)$.

Proof. The result follows the same proof as Proposition 21, as this is simply the special case in which $P = I$. \square

B Full proofs of useful results from the literature

We begin with a proof of Theorem 6.

Proof. Let

$$\bar{\alpha}_k = \min \left(1, \frac{1}{\eta_1(1 + \bar{\vartheta})}, \frac{0.5 \cdot (1 - \eta_2)\gamma_2\theta_k\mu_0}{\|H_P(\Delta X_k \Delta S_k)\|} \right). \quad (63)$$

Then, by Lemma 19, at the k -th iteration, there exists α_k satisfying

$$\alpha_k \geq \bar{\alpha}_k = \min \left(1, \frac{1}{\eta_1(1 + \bar{\vartheta})}, \frac{c}{n^2} \right), \quad (64)$$

where $c > 0$ is a constant which is independent of $\bar{\vartheta}$. Using the fact that

$$\begin{aligned} \theta_k &= \prod_{i=0}^{k-1} (1 - \alpha_i \eta_2) \leq (1 - \bar{\alpha} \eta_2)^k, \text{ and} \\ \frac{1}{\bar{\alpha}} &= \mathcal{O}(n^2), \end{aligned}$$

it follows that $\theta_k \leq \epsilon$ for $k = \mathcal{O}(n^2 \ln(1/\epsilon))$. \square

Lemma 19 (Lemma 4 in [47]). *Suppose that the conditions (15), (16a) and (16b) hold. Then*

$$\|H_P(\Delta X_k \Delta S_k)\| = \mathcal{O}(n^2 \theta_k \mu_0). \quad (65)$$

B.1 Proof of Lemma 19

Following the framework presented in [47], we make use of the quantities that originally appeared in [39]. Choosing $P = S^{1/2}$ (i.e., the HKM direction), define

$$Z_k := \mathcal{F}_k \mathcal{E}_k = \frac{1}{2} \left(S_k^{\frac{1}{2}} X_k S_k^{\frac{1}{2}} \otimes I + I \otimes S_k^{\frac{1}{2}} X_k S_k^{\frac{1}{2}} \right), \quad (66a)$$

$$D_k := Z_k^{-\frac{1}{2}} \mathcal{F}_k = Z_k^{\frac{1}{2}} \mathcal{E}_k^{-1}. \quad (66b)$$

Next, consider the eigenvalue decomposition of

$$S_k^{\frac{1}{2}} X_k S_k^{\frac{1}{2}} = Q_k \Lambda_k Q_k^\top, \quad (67)$$

where $Q_k^\top Q_k = I$, and $\Lambda_k = \text{diag}(\lambda_1^k, \lambda_2^k, \dots, \lambda_n^k)$ with $\lambda_1^k \geq \lambda_2^k \geq \dots \geq \lambda_n^k > 0$. Now, if $(\tau, \theta, X, y, X) \in \mathcal{N}_I$, it can be shown that

$$(1 - \gamma_2)\theta\mu_0 I \preceq H_P(XS) = S_k^{\frac{1}{2}} X_k S_k^{\frac{1}{2}} \preceq (1 + \gamma_2)\theta\mu_0 I, \quad (68)$$

and we have

$$(1 - \gamma_2)\theta_k\mu_0 \leq \lambda_n^k \leq \dots \leq \lambda_1^k \leq (1 + \gamma_2)\theta_k\mu_0. \quad (69)$$

Then, as noted in [47], from equations (67) and (69) the eigenvalues of Z_k are defined to be

$$\left\{ \frac{1}{2}(\lambda_i^k + \lambda_j^k) : i, j = 1, 2, \dots, n \right\},$$

and thus

$$\|Z_k\|_2 \leq (1 + \gamma_2)\theta_k\mu_0, \quad \|Z_k^{-1}\|_2 \leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0}. \quad (70)$$

Next, define

$$G_k = \frac{1}{2} \left(X_k^{-\frac{1}{2}} S_k^{-1} X_k^{-\frac{1}{2}} \otimes I + I \otimes X_k^{-\frac{1}{2}} S_k^{-1} X_k^{-\frac{1}{2}} \right). \quad (71)$$

Following [47], $X_k^{\frac{1}{2}} S_k X_k^{\frac{1}{2}}$ and $S_k^{\frac{1}{2}} X_k S_k^{\frac{1}{2}}$ have the same spectrum, which implies that the eigenvalues of G_k are defined to be

$$\left\{ \frac{1}{2} \left(\frac{1}{\lambda_i^k} + \frac{1}{\lambda_j^k} \right) : i, j = 1, 2, \dots, n \right\}.$$

Therefore,

$$\frac{2}{(1 + \gamma_2)\theta_k\mu_0} \leq \frac{1}{\lambda_i^k} + \frac{1}{\lambda_j^k} \leq \frac{2}{(1 - \gamma_2)\theta_k\mu_0}, \quad i, j = 1, 2, \dots, n. \quad (72)$$

In what follows, we present lemmas from [47] which lead to a proof of Lemma 19.

Lemma 20 (Lemma 5 in [47]). *For any $\mathcal{M} \in \mathbb{R}^{n \times n}$*

$$\begin{aligned} \|D_k^{-\top} \mathbf{vec}(\mathcal{M})\|^2 &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \|S_k^{\frac{1}{2}} \mathcal{M} S_k^{\frac{1}{2}}\|^2 \\ \|D_k \mathbf{vec}(\mathcal{M})\|^2 &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \|X_k^{\frac{1}{2}} \mathcal{M} X_k^{\frac{1}{2}}\|^2. \end{aligned}$$

Proof. From the definition of the matrices $\mathcal{E}_k, \mathcal{F}_k, G_k, Z_k$ and D_k , it follows

$$\begin{aligned} \|D_k^{-\top} \mathbf{vec}(\mathcal{M})\|^2 &= (\mathbf{vec}(\mathcal{M}))^\top \mathcal{E}_k Z_k^{-1} \mathcal{E}_k \mathbf{vec}(\mathcal{M}) \\ &= \mathbf{vec} \left(S_k^{\frac{1}{2}} \mathcal{M} S_k^{\frac{1}{2}} \right)^\top Z_k^{-1} \mathbf{vec} \left(S_k^{\frac{1}{2}} \mathcal{M} S_k^{\frac{1}{2}} \right) \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \left\| \mathbf{vec} \left(S_k^{\frac{1}{2}} \mathcal{M} S_k^{\frac{1}{2}} \right) \right\|^2 \\ &= \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \left\| S_k^{\frac{1}{2}} \mathcal{M} S_k^{\frac{1}{2}} \right\|^2. \end{aligned}$$

Additionally,

$$\begin{aligned} \|D_k \mathbf{vec}(\mathcal{M})\|^2 &= (\mathbf{vec}(\mathcal{M}))^\top \mathcal{E}_k^{-1} Z_k \mathcal{E}_k^{-1} \mathbf{vec}(\mathcal{M}) \\ &= (\mathbf{vec}(\mathcal{M}))^\top \mathcal{E}_k^{-1} \mathcal{F}_k \mathbf{vec}(\mathcal{M}) \\ &= \mathbf{vec} \left(X_k^{\frac{1}{2}} \mathcal{M} X_k^{\frac{1}{2}} \right)^\top G_k \mathbf{vec} \left(X_k^{\frac{1}{2}} \mathcal{M} X_k^{\frac{1}{2}} \right) \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \left\| \mathbf{vec} \left(X_k^{\frac{1}{2}} \mathcal{M} X_k^{\frac{1}{2}} \right) \right\|^2 \\ &= \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \left\| X_k^{\frac{1}{2}} \mathcal{M} X_k^{\frac{1}{2}} \right\|^2. \end{aligned}$$

□

Lemma 21 (Lemma 6 in [47]). *Let $\kappa = \frac{\lambda_1^k}{\lambda_n^k} \leq \frac{1 + \gamma_2}{1 - \gamma_2}$. Then*

$$\begin{aligned} \|D_k^{-\top} \mathbf{vec}(\Delta X_k)\|^2 + \|D_k \mathbf{vec}(\Delta S_k)\|^2 + 2\Delta X_k \cdot \Delta S_k &= \|Z_k^{-\frac{1}{2}} (\mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c))\|^2, \\ \|H_P(\Delta X_k \Delta S_k)\| &\leq \frac{\sqrt{\kappa}}{2} (\|D_k^{-\top} \mathbf{vec}(\Delta X_k)\|^2 + \|D_k \mathbf{vec}(\Delta S_k)\|^2). \end{aligned}$$

Proof. See Lemmas 3.1 and 3.3 in [39]. □

Lemma 22 (Lemma 7 in [47]). *We have*

$$\|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c))\|^2 = \mathcal{O}(n\theta_k\mu_0).$$

Proof. From (23) and (70) it follows

$$\begin{aligned} \|Z_k^{-\frac{1}{2}}(\mathbf{vec}(r_k^c))\|^2 &= (\mathbf{vec}(r_k^c))^\top Z_k^{-1} \mathbf{vec}(r_k^c) \\ &\leq \|Z_k^{-1}\|_2 \|r_k^c\|^2 \\ &\leq \frac{0.25[(1-\eta_2)\gamma_2\theta_k\mu_0]^2}{(1-\gamma_2)\theta_k\mu_0} \\ &= \frac{[(1-\eta_2)\gamma_2]^2\theta_k\mu_0}{4(1-\gamma_2)}. \end{aligned} \tag{73}$$

Then, from equation (67) it follows

$$\mathbf{vec}(R_k^c) = (Q_k \otimes Q_k) \mathbf{vec}((1-\eta_2)\theta_k\mu_0 I - \Lambda_k)$$

and hence,

$$\begin{aligned} \|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c))\|^2 &= (\mathbf{vec}(R_k^c))^\top Z_k^{-1} \mathbf{vec}(R_k^c) \\ &\leq \|Z_k^{-1}\|_2 \|\mathbf{vec}(R_k^c)\|^2 \\ &\leq \frac{1}{(1-\gamma_2)\gamma_2\theta_k\mu_0} \sum_{i=1}^n ((1-\eta_2)\theta_k\mu_0 - \lambda_i^k)^2 \\ &\leq \frac{1}{(1-\gamma_2)\gamma_2\theta_k\mu_0} \sum_{i=1}^n (|\theta_k\mu_0 - \lambda_i^k| + \eta_2\theta_k\mu_0)^2 \\ &\leq \frac{n\theta_k\mu_0}{1-\gamma_2} (\gamma_2 + \eta_2)^2. \end{aligned} \tag{74}$$

Finally, from (73) and (74) it follows

$$\begin{aligned} \|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c))\|^2 &= 2\|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c))\|^2 + 2\|Z_k^{-\frac{1}{2}}(\mathbf{vec}(r_k^c))\|^2 \\ &\leq \frac{2n\theta_k\mu_0}{1-\gamma_2} (\gamma_2 + \eta_2)^2 + \frac{[(1-\eta_2)\gamma_2]^2\theta_k\mu_0}{2(1-\gamma_2)} \\ &= \mathcal{O}(n\theta_k\mu_0). \end{aligned}$$

The proof is complete. □

To complete the analysis, as in [47], we consider a point $(\tilde{X}, \tilde{y}, \tilde{S})$. Note that it is guaranteed that such a point exists by Lemma 2, and at iteration k , we face the following linear system:

$$\mathcal{A}^\top \tilde{y} + \mathbf{vec}(\tilde{S}) - \mathbf{vec}(X) = R_0^d + r^d, \quad \|\zeta_k^d\| \leq \gamma_1 \rho \tag{75a}$$

$$\mathbf{Avec}(\tilde{X}) - b = R_0^p + r^p, \quad \|\mathcal{A}^+ \zeta_k^p\| \leq \gamma_1 \rho. \tag{75b}$$

Therefore, following [47], by Lemma 2 there exists $(\tilde{X}_k, \tilde{y}_k, \tilde{S}_k)$ satisfying

$$\mathcal{A}^\top \tilde{y}_k - \mathbf{vec}(\tilde{S}_k) - \mathbf{vec}(C) = R_0^d + \zeta_k^d \tag{76a}$$

$$\mathbf{Avec}(\tilde{X}_k) - b = R_0^p + \zeta_k^p \tag{76b}$$

$$(1-\gamma_1)\rho I \preceq \tilde{X}_k \preceq (1+\gamma_1)\rho I \tag{76c}$$

$$(1-\gamma_1)\rho I \preceq \tilde{S}_k \preceq (1+\gamma_1)\rho I. \tag{76d}$$

Lemma 23 (Lemma 8 in [47]). Let $\mathcal{R}_k^p, \mathcal{R}_k^d \in \mathbb{R}^{n \times n}$ be the unique matrices such that

$$\mathbf{vec}(\mathcal{R}_k^p) = \mathcal{A}^+ r_k^p \quad (77a)$$

$$\mathbf{vec}(\mathcal{R}_k^d) = r_k^d. \quad (77b)$$

Then, the following conditions hold

$$(X_k - X_* - \tau_k(\tilde{X}_k - X_*)) \bullet (S_k - S_* - \tau_k(\tilde{S}_k - S_*)) = 0 \quad (78a)$$

$$(\Delta X_k + \eta_1 \tau_k(\tilde{X}_k - X_*) + \eta_1 \mathcal{R}_k^p) \bullet (\Delta S_k + \eta_1 \tau_k(\tilde{S}_k - S_*) + \eta_1 \mathcal{R}_k^d) = 0. \quad (78b)$$

Proof. Given

$$\mathcal{A}^\top y_* + \mathbf{vec}(S_*) = \mathbf{vec}(C)$$

$$\mathcal{A} \mathbf{vec}(X_*) = b,$$

it follows from (75a)-(76b) that

$$\mathcal{A}^\top (\mathbf{vec}(X_k - X_*) - \tau_k \mathbf{vec}(\tilde{X}_k - X_*)) = 0 \quad (79)$$

$$\mathcal{A} (y_k - y_* - \tau_k(\tilde{y}_k - y_*)) + (\mathbf{vec}(S_k - S_*) - \tau_k \mathbf{vec}(\tilde{S}_k - S_*)) = 0. \quad (80)$$

As a consequence of (79) we have

$$(\mathbf{vec}(X_k - X_*) - \tau_k \mathbf{vec}(\tilde{X}_k - X_*))^\top (\mathbf{vec}(S_k - S_*) - \tau_k \mathbf{vec}(\tilde{S}_k - S_*)) = 0 \quad (81)$$

and thus (78a) is proved.

Next, by (24), and (75a)-(76b) we have

$$\mathcal{A} (\mathbf{vec}(\Delta X_k) + \eta_1 \tau_k \mathbf{vec}(\tilde{X}_k - X_*) + \eta_1 \mathcal{A}^+ r_k^p) = 0 \quad (82)$$

$$\mathcal{A}^\top (\Delta y_k + \eta_1 \tau_k(\tilde{y}_k - y_*)) + (\mathbf{vec}(\Delta S_k) + \eta_1 \tau_k \mathbf{vec}(\tilde{S}_k - S_*) + \eta_1 r_k^d) = 0 \quad (83)$$

from which (78b) follows. \square

Define

$$t = (\|D_k^{-\top} \mathbf{vec}(\Delta X_k)\|^2 + \|D_k \mathbf{vec}(\Delta S_k)\|^2)^{\frac{1}{2}} \quad (84)$$

$$\beta = (\|D_k^{-\top} \mathbf{vec}(\tilde{X}_k - X_*)\|^2 + \|D_k \mathbf{vec}(\tilde{S}_k - S_*)\|^2)^{\frac{1}{2}} \quad (85)$$

$$\delta = (\|D_k^{-\top} \mathcal{A}^+ r_k^p\|^2 + \|D_k r_k^d\|^2)^{\frac{1}{2}}, \quad (86)$$

then from [47], we have the subsequent lemma.

Lemma 24 (Lemma 9 in [47]).

$$t \leq 2\eta_1(\tau_k \beta + \delta) + \sqrt{\varrho}$$

where

$$\varrho = \|Z_k^{-\frac{1}{2}} \mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c)\|^2 + 2(\eta_1 \tau_k)^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) + 2\eta_1^2 (\tau_k \beta + \delta) \delta.$$

Proof. It follows from (78b)

$$\begin{aligned} \Delta X_k \bullet \Delta S_k &= -\eta_1 \tau_k ((\Delta X_k \bullet (\tilde{S}_k - S_*) + (\tilde{X}_k - X_*) \bullet \Delta S_k)) \\ &= -(\eta_1 \tau_k)^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) - \eta_1 (\Delta X_k \bullet \mathcal{R}_k^p + \mathcal{R}_k^d \Delta S_k) \\ &= -\eta_1^2 \tau_k ((\tilde{X}_k - X_*) \bullet \mathcal{R}_k^d + \mathcal{R}_k^p \bullet (\tilde{S}_k - S_*)) - \eta_1^2 \mathcal{R}_k^p \bullet \mathcal{R}_k^d \\ &\geq -\eta_1 \tau_k \beta t - \eta_1 \delta t - (\eta_1 \tau_k)^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) - \eta_1^2 \tau_k \beta \delta - \frac{1}{2} \eta_1^2 \delta^2 \\ &\geq -\eta_1 (\tau_k \beta + \delta) t - (\eta_1 \tau_k)^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) - \eta_1^2 \tau_k \beta \delta - \eta_1^2 (\tau_k \beta + \delta) \delta. \end{aligned}$$

Applying Lemma 21 in addition to the above inequality yields

$$t^2 - 2\eta_1(\tau_k\beta + \delta)t - \varrho \leq 0.$$

Therefore,

$$\begin{aligned} t &\leq \eta_1(\tau_k\beta + \delta) + \sqrt{\eta_1^2(\tau_k\beta + \delta)^2 + \varrho} \\ &\leq 2\eta_1(\tau_k\beta + \delta) + \sqrt{\varrho}. \end{aligned}$$

□

Lemma 25 (Lemma 10 in [47]). *We have*

$$\delta^2 = \mathcal{O}(n^2\theta_k\mu_0).$$

Proof. Recall, from (23), we have

$$\|\mathcal{A}^+ r_k^d\| \leq \tau_k \gamma_1 \rho, \text{ and } \|r_k^p\| \leq \tau_k \gamma_1 \rho. \quad (87)$$

Applying Lemma 20, (77) and using the fact that $\|\mathcal{M}\|_F \leq \text{tr}(\mathcal{M})$ for $\mathcal{M} \in \mathcal{S}_+^n$ it follows

$$\begin{aligned} \|D_k^{-\top} \mathcal{A}^+ r_k^p\|^2 &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \|S_k\|^2 \|\mathcal{A}^+ r_k^p\|^2 \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} (\text{tr}(S_k))^2 \|\mathcal{A}^+ r_k^p\|^2 \\ &\leq \frac{\gamma_1^2 \rho^2}{(1 - \gamma_2)\theta_k\mu_0} \tau_k^2 (\text{tr}(S_k))^2 \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \mathcal{O}(n^2\theta_k^2\rho^4) \\ &= \mathcal{O}(n^2\theta_k\mu_0). \end{aligned}$$

Additionally, for $\|D_k r_k^d\|^2$, we have

$$\begin{aligned} \|D_k r_k^d\|^2 &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \|X_k\|^2 \|r_k^d\|^2 \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} (\text{tr}(X_k))^2 \|r_k^d\|^2 \\ &\leq \frac{\gamma_1^2 \rho^2}{(1 - \gamma_2)\theta_k\mu_0} \tau_k^2 (\text{tr}(X_k))^2 \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k\mu_0} \mathcal{O}(n^2\theta_k^2\rho^4) \\ &= \mathcal{O}(n^2\theta_k\mu_0). \end{aligned}$$

Thus, $\delta^2 = \mathcal{O}(n^2\theta_k\mu_0)$. □

Lemma 26 (Lemma 11 in [47]). *Under the conditions (15), (16a) and (16b),*

$$(\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) = \mathcal{O}(n\mu_0).$$

Proof. Applying (76c) and (76d) and using the fact that $\tilde{X}_k, \tilde{S}_k, X_k, S_k \in \mathcal{S}_+^n$ we have

$$\begin{aligned} (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) &= \tilde{X}_k \bullet \tilde{S}_k - \tilde{X}_k \bullet S_* - X_* \bullet \tilde{S}_k + X_* \bullet S_* \\ &\leq \tilde{X}_k \bullet \tilde{S}_k \\ &\leq (1 + \gamma_2)^2 n \rho^2 \\ &= (1 + \gamma)^2 n \mu_0 = \mathcal{O}(n\mu_0). \end{aligned}$$

□

Lemma 27 (Lemma 12 in [47]). *Under the conditions (15), (16a) and (16b),*

$$\tau_k^2 \beta^2 = \mathcal{O}(n^2 \theta_k \mu_0).$$

Proof. Applying Lemma 20 and using the fact that $\tilde{X}_k - X_* \succeq 0$ it follows

$$\begin{aligned} \|D_k^{-\top} \mathbf{vec}(\tilde{X}_k - X_*)\|^2 &\leq \frac{1}{(1 - \gamma_2)\theta_k \mu_0} \|S_k^{\frac{1}{2}}(\tilde{X}_k - X_*)S_k^{\frac{1}{2}}\|^2 \\ &\leq \frac{1}{(1 - \gamma_2)\theta_k \mu_0} \text{tr}\left(S_k^{\frac{1}{2}}(\tilde{X}_k - X_*)S_k^{\frac{1}{2}}\right)^2 \\ &= \frac{1}{(1 - \gamma_2)\theta_k \mu_0} ((\tilde{X}_k - X_*) \bullet S_k)^2, \end{aligned}$$

and hence

$$\|D_k \mathbf{vec}(\tilde{S}_k - S_*)\|^2 \leq \frac{1}{(1 - \gamma_2)\theta_k \mu_0} (X_k \bullet (\tilde{S}_k - S_*))^2.$$

Therefore

$$\tau_k \beta \leq \frac{1}{\sqrt{(1 - \gamma_2)\theta_k \mu_0}} \tau_k ((\tilde{X}_k - X_*) \bullet S_k + X_k \bullet (\tilde{S}_k - S_*)).$$

Next, following [47] we use (78a) and that $X_* \bullet S_* = 0$, so we have

$$\begin{aligned} &\tau_k ((\tilde{X}_k - X_*) \bullet S_k + X_k \bullet (\tilde{S}_k - S_*)) \\ &= X_k \bullet S_k - X_k \bullet S_* - X_* \bullet S_k + \tau_k (\tilde{Y}_k \bullet S_* + X_* \bullet \tilde{S}_k) \\ &\quad + \tau_k^2 (\tilde{X}_k \bullet \tilde{S}_k - \tilde{X}_k \bullet S_* - \tilde{X}_* \bullet \tilde{S}_k) \\ &\leq X_k \bullet S_k + n u_k (\tilde{X}_k \bullet S_* + X_* \bullet \tilde{S}_k) + \tau_k^2 \tilde{X}_k \bullet \tilde{S}_k \\ &\leq (1 + \gamma_2)n\theta_k \mu_0 + \theta_k (1 + \gamma_1)\rho(I \bullet S_* + X_* \bullet I) + \theta_k^2 (1 + \gamma_1)^2 n \mu_0 \\ &\leq 6n\theta_k \mu_0 + 2\tau_k \rho (\text{tr}(X_*) + \text{tr}(S_*)) \\ &\leq 6n\theta_k \mu_0 + 2\tau_k \rho^2 \\ &= 8n\theta_k \mu_0. \end{aligned}$$

Thus,

$$\theta_k^2 \beta^2 \leq \frac{64n^2 \theta_k \mu_0}{1 - \gamma_2} = \mathcal{O}(n^2 \theta_k \mu_0),$$

which completes the proof. \square

Proof. Proof of Lemma 19 Applying Lemma 24 and using the fact that $(a + b)^2 \leq 2a^2 + 2b^2$, we have

$$\begin{aligned} t^2 &\leq (2(\tau_k \beta + \delta) + \sqrt{\varrho})^2 \leq 8(\tau_k \beta + \delta)^2 + 2\varrho \\ \varrho &\leq \|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c))\|^2 + 2\tau_k^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) + 2(\tau_k \beta + \delta)\delta. \end{aligned}$$

Hence,

$$\begin{aligned} t^2 &\leq 8(\tau_k \beta + \delta)(\tau_k \beta + 2\delta) + 2\|Z_k^{-\frac{1}{2}}(\mathbf{vec}(R_k^c) + \mathbf{vec}(r_k^c))\|^2 + 4\tau_k^2 (\tilde{X}_k - X_*) \bullet (\tilde{S}_k - S_*) \\ &\leq \mathcal{O}(n^2 \theta_k \mu_0) + \mathcal{O}(n\theta_k \mu_0) + \mathcal{O}(n\theta_k^2 \mu_0). \end{aligned}$$

Note that, as in [47], the last inequality results from applying Lemmas 22, 25, 26 and 27. Finally, from Lemma 21, it follows that

$$\|H_P(\Delta X_k \Delta S_k)\| \leq \frac{1}{2} \sqrt{\frac{1 + \gamma_2}{1 - \gamma_2}} t^2 = \mathcal{O}(n^2 \theta_k \mu_0),$$

and the proof of Lemma 19 is complete. \square

References

- [1] Leonid G. Khachiyan. “Polynomial algorithms in linear programming”. *USSR Computational Mathematics and Mathematical Physics* **20**, 53–72 (1980).
- [2] Narendra Karmarkar. “A new polynomial-time algorithm for linear programming”. *Combinatorica* Pages 373–395 (1984).
- [3] Yurii E. Nesterov and Arkadi Nemirovskii. “A general approach to polynomial-time algorithms design for convex programming”. Report, Central Economical and Mathematical Institute, USSR Academy of Sciences, Moscow (1988).
- [4] Yurii E. Nesterov and Arkadi Nemirovskii. “Interior-Point Polynomial Algorithms in Convex Programming”. Volume 13. SIAM. (1995).
- [5] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. “Linear Matrix Inequalities in System and Control Theory”. SIAM. (1994).
- [6] Eric M. Rains. “A semidefinite program for distillable entanglement”. *IEEE Transactions on Information Theory* **47**, 2921–2933 (2001).
- [7] Gert R.G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. “Learning the kernel matrix with semidefinite programming”. *Journal of Machine Learning Research* **5**, 27–72 (2004).
- [8] Kilian Q. Weinberger and Lawrence K. Saul. “Unsupervised learning of image manifolds by semidefinite programming”. *International Journal of Computer Vision* **70**, 77–90 (2006).
- [9] Alexandre d’Aspremont, Laurent El Ghaoui, Michael I. Jordan, and Gert R.G. Lanckriet. “A direct formulation for sparse PCA using semidefinite programming”. *SIAM Review* **49**, 434–448 (2007). [arXiv:https://doi.org/10.48550/arXiv.cs/0406021](https://doi.org/10.48550/arXiv.cs/0406021).
- [10] Henry Wolkowicz, Romesh Saigal, and Lieven Vandenbergh. “Handbook of semidefinite programming: Theory, algorithms, and applications”. Springer Science & Business Media. (2012).
- [11] Yonina C. Eldar. “A semidefinite programming approach to optimal unambiguous discrimination of quantum states”. *IEEE Transactions on Information Theory* **49**, 446–456 (2003).
- [12] Aram W. Harrow, Anand Natarajan, and Xiaodi Wu. “An improved semidefinite programming hierarchy for testing entanglement”. *Communications in Mathematical Physics* **352**, 881–904 (2017).
- [13] John Watrous. “Semidefinite programs for completely bounded norms” (2009).
- [14] Michel X. Goemans and David P. Williamson. “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming”. *Journal of the ACM (JACM)* **42**, 1115–1145 (1995).
- [15] László Lovász. “On the Shannon capacity of a graph”. *IEEE Transactions on Information Theory* **25**, 1–7 (1979).
- [16] Erling D. Andersen and Knud D. Andersen. “The MOSEK interior point optimizer for linear programming: an implementation of the homogeneous algorithm”. In Hans Frenk, Kees Roos, Tamás Terlaky, and Shuzhong Zhang, editors, High Performance Optimization. Pages 197–232. Springer (2000).
- [17] Jos F. Sturm. “Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones”. *Optimization Methods and Software* **11**, 625–653 (1999).
- [18] Kim-Chuan Toh, Michael J. Todd, and Reha H. Tütüncü. “SDPT3—a MATLAB software package for semidefinite programming, version 1.3”. *Optimization Methods and Software* **11**, 545–581 (1999).
- [19] Farid Alizadeh, Jean-Pierre A. Haeberly, and Michael L. Overton. “Primal-dual interior-point methods for semidefinite programming: convergence rates, stability and numerical results”. *SIAM Journal on Optimization* **8**, 746–768 (1998).
- [20] Haotian Jiang, Yin Tat Lee, Zhao Song, and Sam Chiu-wai Wong. “An improved cutting plane method for convex optimization, convex-concave games, and its applications”. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing. Pages 944–953. (2020).
- [21] Yin Tat Lee, Aaron Sidford, and Sam Chiu-wai Wong. “A faster cutting plane method and its implications for combinatorial and convex optimization”. In Rafail Ostrovsky and Venkatesan

- Guruswami, editors, 2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS). Pages 1049–1065. IEEE (2015).
- [22] Haotian Jiang, Tarun Kathuria, Yin Tat Lee, Swati Padmanabhan, and Zhao Song. “A faster interior point method for semidefinite programming”. In Sandy Irani, Lisa O’Conner, and Patrick Kellenberger, editors, 2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS). Pages 910–918. IEEE (2020).
- [23] Renato D.C. Monteiro. “Polynomial convergence of primal-dual algorithms for semidefinite programming based on the Monteiro and Zhang family of directions”. *SIAM Journal on Optimization* **8**, 797–812 (1998).
- [24] Yurii E. Nesterov and Michael J. Todd. “Self-scaled barriers and interior-point methods for convex programming”. *Mathematics of Operations Research* Pages 1–42 (1997).
- [25] Yurii E. Nesterov and Michael J. Todd. “Primal-dual interior-point methods for self-scaled cones”. *SIAM Journal on Optimization* **8**, 324–364 (1998).
- [26] Sanjeev Arora, Elad Hazan, and Satyen Kale. “The multiplicative weights method: a meta-algorithm and its applications”. *Theory of Computing*, 8(6) 121-164 (2012).
- [27] Fernando G.S.L. Brandão and Krysta M. Svore. “Quantum speed-ups for solving semidefinite programs”. In Rafail Ostrovsky and Chris Umans, editors, 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS). Pages 415–426. IEEE (2017).
- [28] Joran van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. “Quantum SDP-solvers: Better upper and lower bounds”. *Quantum* **4**, 230 (2020).
- [29] Sander Gribling. “Applications of Optimization to Factorization Ranks and Quantum Information Theory”. PhD. Thesis, CentER, Tilburg University. (2019).
- [30] Joran van Apeldoorn and András Gilyén. “Improvements in Quantum SDP-Solving with Applications”. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019). Volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 99:1–99:15. Dagstuhl, Germany (2019). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [31] Iordanis Kerenidis and Anupam Prakash. “A quantum interior point method for LPs and SDPs”. *ACM Transactions on Quantum Computing* **1**, 1–32 (2020).
- [32] Pablo A.M. Casares and Miguel Angel Martin-Delgado. “A quantum interior-point predictor-corrector algorithm for linear programming”. *Journal of Physics A: Mathematical and Theoretical* **53**, 445305 (2020).
- [33] Iordanis Kerenidis, Anupam Prakash, and Dániel Szilágyi. “Quantum algorithms for second-order cone programming and support vector machines”. *Quantum* **5**, 427 (2021).
- [34] Aharon Ben-Tal and Arkadi Nemirovskii. “Lectures on modern convex optimization: Analysis, algorithms, and engineering applications”. *SIAM*. (2001).
- [35] Michael J. Todd. “A study of search directions in primal-dual interior-point methods for semidefinite programming”. *Optimization Methods and Software* **11**, 1–46 (1999).
- [36] Masakazu Kojima, Susumu Shindoh, and Shinji Hara. “Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices”. *SIAM Journal on Optimization* **7**, 86–125 (1997).
- [37] Renato D.C. Monteiro. “Primal–dual path-following algorithms for semidefinite programming”. *SIAM Journal on Optimization* **7**, 663–678 (1997).
- [38] Renato D.C. Monteiro and Yin Zhang. “A unified analysis for a class of long-step primal-dual path-following interior-point algorithms for semidefinite programming”. *Mathematical Programming* **81**, 281–299 (1998).
- [39] Yin Zhang. “On extending some primal–dual interior-point algorithms from linear programming to semidefinite programming”. *SIAM Journal on Optimization* **8**, 365–386 (1998).
- [40] Renato D.C. Monteiro and Takashi Tsuchiya. “Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming”. *SIAM Journal on Optimization* **9**, 551–577 (1999).
- [41] Paul Tseng. “Search directions and convergence analysis of some infeasible path-following methods for the monotone semi-definite LCP”. *Optimization Methods and Software* **9**, 245–268 (1998).

- [42] Florian A. Potra and Rongqin Sheng. “A superlinearly convergent primal-dual infeasible-interior-point algorithm for semidefinite programming”. *SIAM Journal on Optimization* **8**, 1007–1028 (1998).
- [43] Yin Zhang. “On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem”. *SIAM Journal on Optimization* **4**, 208–227 (1994).
- [44] Janos Korzák. “Convergence analysis of inexact infeasible-interior-point algorithms for solving linear programming problems”. *SIAM Journal on Optimization* **11**, 133–148 (2000).
- [45] Shinji Mizuno and Florian Jarre. “Global and polynomial-time convergence of an infeasible-interior-point algorithm using inexact computation”. *Mathematical Programming* **84** (1999).
- [46] Jacek Gondzio. “Convergence analysis of an inexact feasible interior point method for convex quadratic programming”. *SIAM Journal on Optimization* **23**, 1510–1527 (2013).
- [47] Guanglu Zhou and Kim-Chuan Toh. “Polynomiality of an inexact infeasible interior point algorithm for semidefinite programming”. *Mathematical programming* **99**, 261–282 (2004).
- [48] Christoph Helmberg, Franz Rendl, Robert J. Vanderbei, and Henry Wolkowicz. “An interior-point method for semidefinite programming”. *SIAM Journal on Optimization* **6**, 342–361 (1996).
- [49] Michael B. Cohen, Yin Tat Lee, and Zhao Song. “Solving linear programs in the current matrix multiplication time”. *Journal of the ACM (JACM)* **68**, 1–39 (2021).
- [50] Fernando G.S.L. Brandão, Richard Kueng, and Daniel Stilck França. “Faster quantum and classical SDP approximations for quadratic binary optimization”. *Quantum* **6**, 625 (2022).
- [51] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. “Quantum algorithm for linear systems of equations”. *Physical Review Letters* **103**, 150502 (2009).
- [52] Ambros M. Gleixner and Daniel E. Steffy. “Linear programming using limited-precision oracles”. *Mathematical Programming* **183**, 525–554 (2020).
- [53] Ambros M. Gleixner, Daniel E. Steffy, and Kati Wolter. “Improving the accuracy of linear programming solvers with iterative refinement”. In Joris van der Hoeven and Mark van Hoeij, editors, Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation. Pages 187–194. (2012).
- [54] Ambros M. Gleixner, Daniel E. Steffy, and Kati Wolter. “Iterative refinement for linear programming”. *INFORMS Journal on Computing* **28**, 449–464 (2016).
- [55] Rolando D. Somma and Yiğit Subaşı. “Complexity of quantum state verification in the quantum linear systems problem”. *PRX Quantum* **2**, 010315 (2021).
- [56] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. “The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation”. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019). Volume 132, pages 33:1–33:14. Dagstuhl, Germany (2019). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [57] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics”. In Moses Charikar and Edith Cohen, editors, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing. Pages 193–204. (2019).
- [58] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. “Quantum algorithm for systems of linear equations with exponentially improved dependence on precision”. *SIAM Journal on Computing* **46**, 1920–1950 (2017).
- [59] Lov Grover and Terry Rudolph. “Creating superpositions that correspond to efficiently integrable probability distributions” (2002). [arXiv:https://doi.org/10.48550/arXiv.quant-ph/0208112](https://doi.org/10.48550/arXiv.quant-ph/0208112).
- [60] Iordanis Kerenidis and Anupam Prakash. “Quantum recommendation systems” (2016). [arXiv:https://doi.org/10.48550/arXiv.1603.08675](https://doi.org/10.48550/arXiv.1603.08675).
- [61] Michael Keyl. “Quantum state estimation and large deviations”. *Reviews in Mathematical Physics* **18**, 19–60 (2006).
- [62] Ryan O’Donnell and John Wright. “Efficient quantum tomography”. In Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing. Pages 899–912. (2016).

- [63] Joran van Apeldoorn, Arjan Cornelissen, András Gilyén, and Giacomo Nannicini. “Quantum tomography using state-preparation unitaries”. In Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA). Pages 1265–1318. SIAM (2023).
- [64] Etienne de Klerk, Cornelis Roos, and Tamás Terlaky. “Initialization in semidefinite programming via a self-dual skew-symmetric embedding”. *Operations Research Letters* **20**, 213–221 (1997).
- [65] Michael J. Todd, Kim-Chuan Toh, and Reha H. Tütüncü. “On the Nesterov–Todd direction in semidefinite programming”. *SIAM Journal on Optimization* **8**, 769–796 (1998).
- [66] Ron S. Dembo, Stanley C. Eisenstat, and Trond Steihaug. “Inexact Newton Methods”. *SIAM Journal on Numerical Analysis* **19**, 400–408 (1982).
- [67] Carl T. Kelley. “Iterative methods for linear and nonlinear equations”. SIAM. (1995).
- [68] Peter Bürgisser, Michael Clausen, and Mohammad A. Shokrollahi. “Algebraic complexity theory”. Springer Science & Business Media. (2013).
- [69] Volker Strassen. “Gaussian elimination is not optimal”. *Numerische Mathematik* **13**, 354–356 (1969).
- [70] Raphael Yuster and Uri Zwick. “Fast sparse matrix multiplication”. *ACM Transactions On Algorithms (TALG)* **1**, 2–13 (2005).
- [71] Iordanis Kerenidis and Anupam Prakash. “A quantum interior point method for LPs and SDPs” (2018). [arXiv:https://doi.org/10.48550/arXiv.1808.09266](https://doi.org/10.48550/arXiv.1808.09266).
- [72] Yousef Saad. “Iterative methods for sparse linear systems”. SIAM. (2003).
- [73] Nisheeth K. Vishnoi. “ $Lx = b$: Laplacian Solvers and Their Algorithmic Applications”. *Foundations and Trends® in Theoretical Computer Science* **8**, 1–141 (2013).
- [74] Fernando G.S.L. Brandão, Amir Kalev, Tongyang Li, Cedric Yen-Yu Lin, Krysta M. Svore, and Xiaodi Wu. “Quantum SDP Solvers: Large Speed-Ups, Optimality, and Applications to Quantum Learning”. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019). Volume 132, pages 27:1–27:14. Dagstuhl, Germany (2019). Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [75] Yin Tat Lee and Swati Padmanabhan. “An $\tilde{O}(m/\varepsilon^{3.5})$ -cost algorithm for semidefinite programs with diagonal constraints”. In Jacob Abernethy and Shivani Agarwal, editors, Conference on Learning Theory. Pages 3069–3119. PMLR (2020).
- [76] Cornelis Roos, Tamás Terlaky, and Jean-Phillipe Vial. “Interior Point Methods for Linear Optimization”. Springer Science & Business Media. (2005).
- [77] Ali Mohammad-Nezhad and Tamás Terlaky. “On the identification of the optimal partition for semidefinite optimization”. *INFOR: Information Systems and Operational Research* **58**, 1–39 (2019). [arXiv:https://doi.org/10.1080/03155986.2019.1572853](https://doi.org/10.1080/03155986.2019.1572853).