arXiv:2112.08859v3 [quant-ph] 12 Jun 2024

Variational Quantum Algorithms for Semidefinite Programming

Dhrumil Patel[1], Patrick J. Coles[2], and Mark M. Wilde[3,4]

[1]Department of Computer Science, Cornell University, Ithaca, New York, 14850, USA
[2]Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA
[3]Hearne Institute for Theoretical Physics, Department of Physics and Astronomy, and Center for Computation and Technology, Louisiana State University, Baton Rouge, Louisiana 70803, USA
[4]School of Electrical and Computer Engineering, Cornell University, Ithaca, New York 14850, USA

A semidefinite program (SDP) is a particular kind of convex optimization problem with applications in operations research, combinatorial optimization, quantum information science, and beyond. In this work, we propose variational quantum algorithms for approximately solving SDPs. For one class of SDPs, we provide a rigorous analysis of their convergence to approximate locally optimal solutions, under the assumption that they are weakly constrained (i.e., $N \gg M$, where N is the dimension of the input matrices and M is the number of constraints). We also provide algorithms for a more general class of SDPs that requires fewer assumptions. Finally, we numerically simulate our quantum algorithms for applications such as MaxCut, and the results of these simulations provide evidence that convergence still occurs in noisy settings.

Contents

1 Introduction **2**
1.1 Main Idea and Setup . . . 3
1.2 Related Work . . . 6

2 Preliminaries **7**
2.1 Semidefinite Programming . . . 9
2.2 Variational Quantum Algorithms and Parameter-Shift Rule . . . 10

3 Variational Quantum Algorithms for SDPs **13**
3.1 General Form (GF) of SDPs . . . 13
3.1.1 Variational Quantum Algorithm for SDPs in GF . . . 14
3.2 Equality Constrained Standard Form (ECSF) of SDPs . . . 17
3.2.1 Variational Quantum Algorithm for SDPs in ECSF . . . 19
3.2.2 Convergence Rate . . . 20
3.3 Inequality Constrained Standard Form (ICSF) of SDPs . . . 23
3.3.1 Variational Quantum Algorithm for SDPs in ICSF . . . 25

Dhrumil Patel: djp265@cornell.edu
Patrick J. Coles: patrick@normalcomputing.ai, http://patcoles.com/
Mark M. Wilde: wilde@cornell.edu, http://markwilde.com/

4 Numerical Simulations **27**

5 Conclusion **31**

A Appendix **36**
A.1 Proof of Lemma 7 36
A.2 Proof of Lemma 8 37
A.3 Proof of Lemma 16 37
A.4 Proof of Lemma 17 38
A.5 Proof of Lemma 20 39

1 Introduction

Semidefinite programming (SDP) is one of the most important tools in optimization developed over the past few decades. One can see SDPs as a natural extension of the better known linear programs (LPs), in which the vector inequalities of LPs are replaced by matrix inequalities. One of the reasons underlying the importance of SDPs is their applicability to a broad range of problems, including approximation algorithms for combinatorial optimization [1], control theory [2], and sum-of-squares [3]. Additionally, a variety of quantum information problems can be formulated as SDPs, including state discrimination [4, 5], upper bounds on quantum channel capacity [6, 7], and self-testing [8].

The power of SDPs lies with the fact that they can be solved efficiently in polynomial time using classical algorithms such as the celebrated interior-point method [9]. Although SDPs can be solved efficiently using classical techniques, as the size of the input matrices increases, many first-order and second-order algorithms incur significant computational overhead due to the expensive gradient computation at each iteration. For this reason, it is imperative to design more efficient algorithms for solving SDPs.

Given the speed-ups of quantum algorithms over classical algorithms for a variety of problems [10, 11, 12, 13, 14], it is natural to ask if there exists a quantum algorithm that can solve SDPs efficiently. This question was positively answered in [15], wherein a quantum algorithm was proposed and proven to have a quadratic speedup over the classical Arora–Kale algorithm [16]. Following this initial result, more efficient quantum algorithms for solving SDPs were later developed [17, 18].

Although quantum algorithms have been theoretically proven to outperform known classical algorithms for many applications, a fault-tolerant quantum computer is required to reap their benefits. Currently, fault-tolerant quantum computers are not available, and we are instead in the Noisy Intermediate-Scale Quantum (NISQ) era [19]. Google constructed a noisy quantum computer with just 54 qubits [20], but it remains an open challenge to design a fault-tolerant quantum computer that requires millions of qubits for successful operation. Some of the quantum information science research community is focused on determining the power of noisy quantum computers and is designing quantum algorithms that acknowledge the limitations of such devices [21, 22, 23, 24], such as a finite number of gates and qubits, noisy gate execution, and rapid decoherence of qubits.

Variational quantum algorithms (VQAs) constitute an important class of NISQ-friendly algorithms [21, 22]. VQAs can be seen as hybrid quantum-classical algorithms that have a classical computer available for optimization, only calling a quantum subroutine for tasks that are not efficiently solvable by it. Hitherto, VQAs have been proposed for numerous computational tasks, for which there are also known quantum and classical algorithms [21].

Some well studied VQAs include the Variational Quantum Eigensolver (VQE) [25] and the Quantum Approximate Optimization Algorithm (QAOA) [26].

This brings us to the main motivation of our work, where we reformulate three different kinds of SDPs and develop variational quantum algorithms for solving them. In particular, the contributions of our paper are as follows:

1. We present unconstrained reformulations of the constrained general and standard forms of an SDP, by employing a series of reductions (Sections 3.1, 3.2, and 3.3). For the standard form, we consider the case in which the SDP is weakly constrained (i.e., when $N \gg M$, where N is the dimension of the input matrices and M is the number of constraints). On the contrary, we do not require such assumptions when considering the general form, making it applicable to a wide range of problems.

2. We propose variational quantum algorithms for obtaining approximate stationary points of such unconstrained problems, which in turn are approximate stationary points of their respective constrained problems due to the equivalence between them (Sections 3.1.1, 3.2.1, and 3.3.1). Note that stationary points include globally optimal points.

3. We analyze the convergence rate of an algorithm corresponding to the equality constrained standard formulation of an SDP (Section 3.2.2).

4. We provide numerical evidence that showcases how our algorithms work in practice for applications such as MaxCut (Section 4). Specifically, we analyze the convergence of the proposed algorithms by assessing how close the final value is to the actual optimal value evaluated using exact classical solvers. We perform such experiments on a noisy quantum simulator from Qiskit [27] (i.e., QASM simulator) and then compare these results with those on a noiseless simulator. Note that the QASM simulator mimics IBM's quantum computer, which is actually noisy in nature due to gate errors and decoherence.

Additionally, in Section 2, we introduce some notations and definitions, the basics of SDPs, and a brief overview of VQAs, as well as the associated concept of computing partial derivatives on a quantum circuit, i.e., the parameter-shift rule.

1.1 Main Idea and Setup

In this paper, we consider the following three different kinds of SDPs, and we present variational quantum algorithms for all three of them:

- *General Form (GF)*: The general form of an SDP can be concisely written as follows:

$$p^* := \sup_{X \succcurlyeq 0} \{\mathrm{Tr}[CX] : \Phi(X) \preccurlyeq B\}, \tag{1}$$

  where $C \in \mathcal{S}^N$, $B \in \mathcal{S}^M$, and the map Φ is Hermiticity-preserving (Definition 3). Here, the notation $\mathcal{S}^N$ denotes the set of $N \times N$ Hermitian operators. For this case, we do not assume that the SDPs are weakly constrained; i.e., we do not assume that the dimension M of the constraint variable B is much smaller than the dimension N of the objective variable C. Additionally, to generalize it further, we consider an inequality-constrained problem. For solving these general SDPs, we propose a variational quantum algorithm (Algorithm 1).

- *Standard Form (SF)*: Here, we consider the Hermiticity-preserving map Φ and the Hermitian operator B to have a diagonal form, as given in (15). Specifically, in this case, we set

$$\begin{aligned} \Phi(X) &= \operatorname{diag}\left(\operatorname{Tr}[A_1 X], \ldots, \operatorname{Tr}[A_M X]\right), \\ B &= \operatorname{diag}\left(b_1, \ldots, b_M\right), \end{aligned} \tag{2}$$

  where $A_1, \ldots, A_M \in \mathcal{S}^N$ and $b_1, \ldots, b_M \in \mathbb{R}$. This form of an SDP is well known in the convex optimization literature [28], and most exact or approximation algorithms designed for combinatorial optimization problems are based on this form. For this case, we assume that the SDPs are weakly constrained, i.e., $N \gg M$. We further categorize them based on the nature of the constraints as follows:

  - *Equality Constrained Standard Form (ECSF)*: Here, we consider equality constraints, i.e., $\Phi(X) = B$. For solving such SDPs, we propose a variational quantum algorithm (Algorithm 2) and establish its convergence rate and total iteration complexity.
  - *Inequality Constrained Standard Form (ICSF)*: Here, we consider inequality constraints, i.e., $\Phi(X) \preccurlyeq B$, and we propose a variational quantum algorithm for this case (Algorithm 3).

In our methods, we first reduce these constrained optimization problems to their unconstrained forms by employing a series of identities. Second, we express the final unconstrained form as a function of expectation values of the input Hermitian operators, i.e., $C, A_1, \ldots, A_M \in \mathcal{S}^N$.

For solving these final unconstrained formulations using a gradient-based method, we need access to the full gradient of the objective function at each iteration. However, when the dimension N of the input Hermitian operators is large, the evaluation of the gradient of the objective function using a classical computer becomes computationally expensive. Therefore, we delegate this gradient computation to parameterized quantum circuits, and we use a technique called the parameter-shift rule [29, 30, 31] to evaluate the partial derivatives of the objective function with respect to circuit parameters. Please refer to Section 2.2 for more details on the parameter-shift rule.

We design variational quantum algorithms for solving these unconstrained optimization problems. Our methods provide bounds on the optimal values, due to the reduction in the search space, as well as the non-convex nature of the objective function landscape in terms of quantum circuit parameters. We do not assume that the final objective function is convex with respect to these parameters, as generally it is non-convex [32]. In general, finding a globally optimal point for a non-convex function is known to be NP-hard [33, Section 2.1], and so an important question regards the type of solutions that we can guarantee in such a scenario. In the classical optimization literature, the notion of approximate stationary points (Definition 10) is considered when the objective function is non-convex. Therefore, in this paper, we focus on proving the convergence of our algorithms to approximate stationary points, as proving the same for a global optimal point is quite difficult. We would also like to emphasize that VQAs that have been proposed prior to our paper have not considered this notion, which is quite natural when considering non-convex objective functions.

Intuitively, stationary points are those for which the gradient of the function under consideration is equal to zero. Therefore, a stationary point can be a local maximum (including the global maximum), a local minimum (including the global minimum), or a saddle point. Consequently, when using a first-order solver such as gradient descent, one

may get stuck at one of these points. However, it is often desirable to escape unwanted stationary points and move to a more favorable one, depending on the type of optimization problem (maximization or minimization). This can be achieved either through the use of higher-order solvers or by employing a noisy first-order solver [34]. We focus on the latter approach in this paper, as computing higher-order derivatives can be computationally expensive compared to calculating the first-order gradient. Moreover, due to the inherent stochastic nature of variational quantum algorithms (VQAs), a first-order solver can inherently be noisy. This noise can aid in escaping unwanted stationary points and converging to a better stationary point.

The primary reason behind proposing different variational quantum algorithms for three different kinds of SDPs is the nature of the final unconstrained forms of these SDPs. In the general form of an SDP, as we do not consider any assumptions on the input matrices, the final unconstrained optimization problem turns out to be a non-convex–non-concave optimization problem (see (3)). Proving the convergence of the proposed algorithm to an approximate stationary point of this problem requires sophisticated analysis, which we leave for future work. In contrast, although the standard form of SDPs is a special case of the general form, we consider cases where these SDPs are weakly constrained. Studying the standard form of SDPs with this assumption is important because many SDPs of interest are actually large and weakly constrained. Due to this assumption, we observe that the final unconstrained forms turn out to be non-convex–concave optimization problems (see (4) and (6)). Non-convex–concave optimization [35, 36, 37, 38, 39] has a rich literature when compared to that of non-convex–non-concave optimization, as the latter is a harder problem than the former. Therefore, the design and convergence analysis of more sophisticated variational quantum algorithms for solving non-convex–concave optimization problems is possible.

Reformulation of General Form (GF) of SDPs—For the SDPs written as (1), we reduce this form to the following final unconstrained form expressed in terms of quantum circuit parameters $\boldsymbol{\theta}_1 \in [0, 2\pi]^{r_1}$ and $\boldsymbol{\theta}_2 \in [0, 2\pi]^{r_2}$:

$$p^* := \sup_{\substack{\boldsymbol{\theta}_1 \in [0,2\pi]^{r_1}, \\ \lambda \geqslant 0}} \inf_{\substack{\boldsymbol{\theta}_2 \in [0,2\pi]^{r_2}, \\ \mu \geqslant 0}} \left\{ \lambda \langle I \otimes C^\top \rangle_{\boldsymbol{\theta}_1} + \mu \langle I \otimes B \rangle_{\boldsymbol{\theta}_2} - \lambda\mu \langle I \otimes I \otimes \Gamma^\Phi \rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2} \right\}, \tag{3}$$

where Γ^Φ is the Choi operator (Definition 4) of the linear map Φ. See Section 3.1 for more details. For solving the above optimization problem, we propose a VQA called *inexact Variational Quantum Algorithm for General Form* (iVQAGF), in which we have two parameterized quantum circuits competing against each other to maximize/minimize the objective function, and then there is a classical optimizer that updates the parameters of these quantum circuits.

Reformulation of Equality Constrained Standard Form (ECSF) of SDPs—As stated before, for this type of SDP, we make an assumption on it being weakly-constrained, i.e., $N \gg M$. By exploiting this assumption, we design more sophisticated variational quantum algorithms, in which we need just one parameterized quantum circuit. For such SDPs, we arrive at the following final unconstrained form expressed in terms of quantum circuit parameters $\boldsymbol{\theta} \in [0, 2\pi]^r$:

$$p^* := \sup_{\boldsymbol{\theta} \in [0,2\pi]^r} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \left\{ \lambda \langle I \otimes C \rangle_{\boldsymbol{\theta}} + \boldsymbol{y}^\top (\boldsymbol{b} - \lambda \boldsymbol{\Phi}(\boldsymbol{\theta})) + \frac{c}{2} \left\| \boldsymbol{b} - \lambda \boldsymbol{\Phi}(\boldsymbol{\theta}) \right\|^2 \right\}. \tag{4}$$

See Section 3.2 for more details. Throughout this paper, we use the notation

$$\langle H \rangle_{\boldsymbol{\theta}} \equiv \langle \phi(\boldsymbol{\theta}) | H | \phi(\boldsymbol{\theta}) \rangle \tag{5}$$

to represent the expectation value of a Hermitian operator H with respect to $|\phi(\boldsymbol{\theta})\rangle$. For solving the above optimization problem, we propose a VQA called *inexact Variational Quantum Algorithm for Equality Constrained standard form* (iVQAEC). We run iVQAEC on a classical computer, and at any step of the algorithm, the expectation value of a Hermitian operator is evaluated using a parameterized quantum circuit. One of our main results establishes the convergence rate and total iteration complexity of iVQAEC under the assumption of the SDP being weakly-constrained.

Reformulation of Inequality Constrained Standard Form (ICSF) of SDPs—For this case also we make the weakly-constrained assumption on SDPs. Here, we solve the dual problem instead of the primal problem. For this type of SDP, we arrive at the following final unconstrained form expressed in terms of quantum circuit parameters $\boldsymbol{\theta} \in [0, 2\pi]^r$:

$$d' := \sup_{\boldsymbol{\theta}\in[0,2\pi]^r} \inf_{\bar{\boldsymbol{y}}\geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + \frac{b_M}{\gamma} \ln\left(e^{\gamma\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}} + 1\right) \right\}. \tag{6}$$

See Section 3.3 for more details. For solving this problem, we propose a VQA called *inexact Variational Quantum Algorithm for Inequality Constrained standard form* (iVQAIC). For this algorithm, we do not establish its convergence rate, which we leave for future work; instead, we prove a property of the objective function that is necessary for providing such a convergence analysis, i.e., smoothness of the objective function for a fixed $\bar{\boldsymbol{y}}$.

1.2 Related Work

Recently, an approach to semidefinite programming on NISQ devices was proposed in [40], which is non-variational and called NISQ SDP solver (NSS) therein. We should note that this approach does not provide an efficient solution for a general SDP problem. Like our approach, the NSS approach also optimizes over a subset of the positive-semidefinite operator space. The NSS approach assumes the ability to prepare pure states in a set $\{|\psi_i\rangle\}_i$. Using these states, the NSS approach then constructs a hybrid density matrix ansatz: $X_\beta = \sum_{i,j} \beta_{ij} |\psi_i\rangle\langle\psi_j|$, where the entries $\{\beta_{ij}\}_{ij}$ are stored on a classical computer. The NSS approach then transforms the original SDP into a low-dimensional SDP, which is solved by optimizing over these classical entries. Note that the NSS approach does not change the set of pure states for each iteration. As one can see, the NSS approach does not encompass the entire space of positive semidefinite operators; therefore, it is heuristic in nature, similar to ours. This means that there is no guarantee of convergence to the global optimal point.

Additionally, another technique was proposed in [41] where the authors "quantized" the classical randomized cutting plane method for solving semidefinite programs. They used a quantum eigensolver subroutine in order to speedup the classical method. Their results indicate that the robustness of their approach against noise may be useful in implementing their method on NISQ devices.

Our approach here is complementary to both of these approaches because our algorithm is a variational quantum algorithm. Another important point to note here is that, unlike prior works that focus solely on solving SDPs in the standard form, we propose algorithms for solving SDPs in both the standard form and a form considered in [42, 43]. The latter form is prevalent in quantum information theory, as it is used to compute various relevant quantities like fidelity and trace distance. While one can convert this form to the standard form, working in the original form is more convenient, avoiding unnecessary conversion overhead. Comparing the aforementioned approaches to ours is an interesting direction for future work.

2 Preliminaries

In this section, we introduce some notations and definitions.

Notations: We denote the set of real and complex numbers by $\mathbb{R}$ and $\mathbb{C}$, respectively. For a positive integer m, the notation $[m]$ denotes the set $\{1, \ldots, m\}$. We use upper-case letters to denote matrices and bold lower-case letters to denote vectors (e.g., A is a matrix, and $\boldsymbol{x}$ is a vector). Let $\mathcal{H}$ denote a finite-dimensional Hilbert space of dimension N. This Hilbert space represents a system of n qubits, where $n = \lceil \log_2 N \rceil$. The notation $L(\mathcal{H})$ denotes the set of linear operators acting on $\mathcal{H}$. The set of $N \times N$ Hermitian or self-adjoint matrices is denoted by $\mathcal{S}^N \subset L(\mathcal{H})$. The notation $\mathcal{S}^N_+ \subset \mathcal{S}^N$ denotes the set of $N \times N$ positive semidefinite (PSD) matrices, and $\mathcal{D}^N \subset \mathcal{S}^N_+$ denotes the set of $N \times N$ density matrices. Additionally, we use $\|\cdot\|$ to represent the ℓ_2 norm of a vector, as well as the spectral or operator norm of a matrix (its largest singular value), and it should be clear from the context which is being used. Let $\mathrm{Tr}[X]$ denote the trace of a matrix X, i.e., the sum of its diagonal terms. Let $X^\top$ and $X^\dagger$ denote the transpose and Hermitian conjugate (or adjoint) of the matrix X, respectively. The notation $A \succcurlyeq B$ or $A - B \succcurlyeq 0$ indicates that $A - B \in \mathcal{S}^N_+$. For a multivariate function $f : \mathbb{R}^n \to \mathbb{R}$, we use ∇f and $\nabla^2 f$ to denote its gradient and Hessian, respectively. Let $\frac{\partial f(\cdot)}{\partial x_i}$ denote the partial derivative of f with respect to i^{th} component of the vector $\boldsymbol{x}$. For a multivariate vector-valued function $f : \mathbb{R}^n \to \mathbb{R}^m$, we denote its Jacobian by $J_f(\cdot)$.

Let $\mathcal{H}$ and $\mathcal{H}'$ be Hilbert spaces of dimensions N and N', respectively.

Definition 1 (Linear map)**.** *A map $\Phi : L(\mathcal{H}) \to L(\mathcal{H}')$ is a linear map if the following holds $\forall X, Y \in L(\mathcal{H})$ and $\forall \alpha, \beta \in \mathbb{C}$,*

$$\Phi(\alpha X + \beta Y) = \alpha \Phi(X) + \beta \Phi(Y). \tag{7}$$

Definition 2 (Adjoint of a linear map)**.** *The adjoint $\Phi^\dagger : L(\mathcal{H}') \to L(\mathcal{H})$ of a linear map $\Phi : L(\mathcal{H}) \to L(\mathcal{H}')$ is the unique linear map such that*

$$\langle Y, \Phi(X) \rangle = \langle \Phi^\dagger(Y), X \rangle \quad \forall X \in L(\mathcal{H}),\ \forall Y \in L(\mathcal{H}'), \tag{8}$$

where $\langle C, D \rangle := \mathrm{Tr}[C^\dagger D]$ is the Hilbert–Schmidt inner product.

Definition 3 (Hermiticity-preserving linear map)**.** *A linear map $\Phi : L(\mathcal{H}) \to L(\mathcal{H}')$ is a Hermiticity-preserving map if $\Phi(X) \in \mathcal{S}^{N'}$ for all $X \in \mathcal{S}^N$. Equivalently, Φ is Hermiticity preserving if and only if $\Phi(X^\dagger) = \Phi(X)^\dagger$ for all $X \in L(\mathcal{H})$.*

Definition 4 (Choi representation of a linear map)**.** *For every linear map $\Phi : L(\mathcal{H}) \to L(\mathcal{H}')$, its Choi representation Γ^Φ is defined as*

$$\Gamma^\Phi := \sum_{i,j=0}^{N-1} |i\rangle\langle j| \otimes \Phi(|i\rangle\langle j|). \tag{9}$$

The operator Γ^Φ is also known as the Choi operator. Additionally, for every linear map Φ, the following holds $\forall X \in L(\mathcal{H}), Y \in L(\mathcal{H}')$:

$$\mathrm{Tr}[Y \Phi(X)] = \mathrm{Tr}[(X^\top \otimes Y)\Gamma^\Phi], \tag{10}$$

which is a direct consequence of the well known fact that

$$\Phi(X) = \mathrm{Tr}_1[\Gamma^\Phi (X^\top \otimes I)], \tag{11}$$

with the partial trace over the first factor in the tensor-product space (see Ref. [43, Proposition 4.2] for a proof of (11)).

Lemma 5. *Given a linear map $\Phi : L(\mathcal{H}) \to L(\mathcal{H}')$, its Choi operator Γ^Φ is a Hermitian operator if and only if Φ is a Hermiticity-preserving map.*

Proof. First, suppose that Φ is Hermiticity preserving. From (9), we can write

$$(\Gamma^\Phi)^\dagger = \left(\sum_{i,j=0}^{N-1} |i\rangle\langle j| \otimes \Phi(|i\rangle\langle j|)\right)^\dagger = \sum_{i,j=0}^{N-1} |i\rangle\langle j|^\dagger \otimes (\Phi(|i\rangle\langle j|))^\dagger \tag{12}$$

$$\overset{(a)}{=} \sum_{i,j=0}^{N-1} |i\rangle\langle j|^\dagger \otimes \Phi(|i\rangle\langle j|^\dagger) = \sum_{i,j=0}^{N-1} |j\rangle\langle i| \otimes \Phi(|j\rangle\langle i|) = \Gamma^\Phi, \tag{13}$$

where equality (a) follows from Definition 3.

Now suppose that Γ^Φ is Hermitian. Then it follows from (11) that $\Phi(X)$ is Hermitian if X is Hermitian, so that Φ is Hermiticity preserving. ■

We now recall some definitions related to the Lipschitz continuity and smoothness of a function.

Definition 6 (Lipschitz continuity)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is L-Lipschitz continuous if there exists $L > 0$, such that $\|f(\boldsymbol{x}) - f(\boldsymbol{x}')\| \leqslant L \|\boldsymbol{x} - \boldsymbol{x}'\|$ for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^n$. We say that L is a Lipschitz constant of f.*

For a univariate function f, suppose that the absolute value of its first derivative on an interval I is bounded from above by a positive real L, i.e., $\forall x \in I : |{}^{df(x)}/_{dx}| \leqslant L$. Then L is a Lipschitz constant of f.

Lemma 7 (Lipschitz constant for a multivariate function)**.** *For a function $f : \mathbb{R}^n \to \mathbb{R}$ with bounded partial derivatives, the value $L = \sqrt{n} \max_i \{\sup_{\boldsymbol{x}} |\partial f(\boldsymbol{x})/\partial x_i|\}$ is a Lipschitz constant of f.*

Proof. See Appendix A.1. ■

Lemma 8 (Lipschitz constant for a multivariate vector-valued function)**.** *For a multivariate vector-valued function $f : \mathbb{R}^n \to \mathbb{R}^m$, if each of its components, f_i, is L_i-Lipschitz, then $L = \sqrt{\sum_{i=1}^m L_i}$ is a Lipschitz constant of f.*

Proof. See Appendix A.2. ■

Definition 9 (Smoothness)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is ℓ-smooth if its gradient is ℓ-Lipschitz, i.e., if there exists $\ell > 0$ such that $\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{x}')\| \leqslant \ell \|\boldsymbol{x} - \boldsymbol{x}'\|$ for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^n$.*

As most of the objective functions that we deal with in this paper are non-convex in some parameters, it is vital to focus on local optimality rather than global optimality because finding globally optimal points of a non-convex function is generally intractable. Therefore, the notion of ϵ-stationary points is important for us. Intuitively, a point is ϵ-stationary if the norm of the gradient at that point is very small. Formally, we define an ϵ-stationary point as follows:

Definition 10 (ϵ-stationary point)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a differentiable function, and let $\epsilon \geqslant 0$. A point $\boldsymbol{x} \in \mathbb{R}^n$ is an ϵ-stationary point of f if $\|\nabla f(\boldsymbol{x})\| \leqslant \epsilon$.*

The above definition applies to first-order stationary points. This definition is important because we use inexact first-order solvers that converge to approximate first-order stationary points, and such a definition acts as a stopping criterion.

Definition 11 (Polyak–Łojasiewicz (PL) Inequality)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}$ satisfies the PL inequality if, for some $\mu > 0$, the following holds for all $\boldsymbol{x} \in \mathbb{R}^n$:*

$$\frac{1}{2} \|f(\boldsymbol{x})\|^2 \geqslant \mu(f(\boldsymbol{x}) - f^*), \tag{14}$$

where f^* is the globally optimal value of f.

In other words, the above inequality implies that every stationary point is a global minimum.

2.1 Semidefinite Programming

In this section, we recall some basic aspects of semidefinite programming [28, 42]. A semidefinite program is an optimization problem for which the goal is to optimize a linear function over the intersection of the positive semidefinite cone with an affine space. SDPs extend linear programs (LPs), such that the vector inequalities of LPs are generalized to matrix inequalities.

To begin with, recall the standard or canonical form of an SDP [28]:

$$\begin{aligned} \sup_{X \geqslant 0} \quad & \mathrm{Tr}[CX] \\ \text{subject to} \quad & \mathrm{Tr}[A_i X] \leqslant b_i; \quad \forall i \in [M], \end{aligned} \tag{15}$$

where $C, A_1, \ldots, A_M \in \mathcal{S}^N$ and $b_1, \ldots, b_M \in \mathbb{R}$. The standard form of an SDP is widely known for designing approximation algorithms for combinatorial optimization problems. A more general form of an SDP, as considered in [42], is as follows:

$$\begin{aligned} p^* := \sup_{X \geqslant 0} \quad & \mathrm{Tr}[CX] \\ \text{subject to} \quad & \Phi(X) \leqslant B, \end{aligned} \tag{16}$$

where $C \in \mathcal{S}^N$, the map Φ is Hermiticity preserving (see Definition 7), $B \in \mathcal{S}^M$, and p^* is the optimal value of the program (16). The aforementioned form is known as the primal form of an SDP, and the corresponding dual form is given as

$$\begin{aligned} d^* := \inf_{Y \geqslant 0} \quad & \mathrm{Tr}[BY] \\ \text{subject to} \quad & \Phi^\dagger(Y) \geqslant C, \end{aligned} \tag{17}$$

where $Y \in S_+^M$, the map $\Phi^\dagger$ is the adjoint of Φ (see Definition 2), and d^* is the optimal value of the program (17).

The duality theorem of SDPs states that if both the primal and dual programs have feasible solutions, then the optimum of the primal program is bounded from above by the optimum of the dual program. Under a very mild condition that the primal has a feasible solution and the dual has a strictly feasible solution (or vice versa), strong duality holds; i.e., the duality gap (difference between p^* and d^*) is closed. In the optimization literature, this condition is well known as Slater's condition (see Theorem 1.18 of [42]). Throughout this paper, we assume that strong duality holds.

2.2 Variational Quantum Algorithms and Parameter-Shift Rule

Variational quantum algorithms are hybrid quantum-classical algorithms, designed for solving optimization tasks with an objective function of the following form:

$$\mathcal{F}(\rho) = \sum_k g_k(\mathrm{Tr}[H_k\rho]), \tag{18}$$

where $\rho \in \mathcal{D}^N$ is a density operator, $\{H_k\}_k$ is a set of problem-specific Hermitian operators, i.e., $H_k \in \mathcal{S}^N$ for all k, and $\{g_k\}_k$ is a problem-specific set of functions [21, 22]. Additionally, each g_k is a function of the expectation value of H_k with respect to ρ. The corresponding optimization problem is as follows:

$$\min_{\rho\in\mathcal{D}^N} \mathcal{F}(\rho). \tag{19}$$

When the dimension N is large, the evaluation of the expectation values of Hermitian operators with respect to ρ is computationally intractable using classical algorithms. VQAs provide a quantum advantage because these hybrid algorithms attempt to circumvent this dimensionality problem by evaluating the expectation values of Hermitian operators using a quantum computer. Specifically, these algorithms utilize a parameterized quantum circuit to explore a problem-specific subspace of density operators and evaluate the expectation values with respect to these density operators. We discuss more about the quantum advantage of VQAs at the end of this section. First, let us discuss what we mean by a parameterized quantum circuit and how this circuit prepares a parameterized quantum state.

Parameterization: Let $|\mathbf{0}\rangle_{RS}$ denote the all-zeros state of systems R and S, each of which consists of n qubits. Let $\rho_S \in \mathcal{D}^{2^n}$, and let U^ρ_{RS} be a quantum circuit that prepares a purification $|\psi\rangle_{RS}$ of ρ_S when U^ρ_{RS} is applied to the initial state $|\mathbf{0}\rangle_{RS}$. Here, the subscripts S and R are used to denote the system of interest and a reference system, respectively. VQAs simulate the space of density operators by parameterizing this quantum circuit as $U_{RS}(\boldsymbol{\theta})$, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_r)^\top \in [0, 2\pi]^r$. Specifically, a VQA applies a parameterized quantum circuit $U_{RS}(\boldsymbol{\theta})$ to the initial pure state $|\mathbf{0}\rangle_{RS}$ to generate a parameterized pure state

$$|\psi(\boldsymbol{\theta})\rangle_{RS} := U_{RS}(\boldsymbol{\theta})|\mathbf{0}\rangle_{RS} \in (\mathbb{C}^2)^{\otimes 2n}. \tag{20}$$

Let $\rho(\boldsymbol{\theta})_S := \mathrm{Tr}_R[|\psi(\boldsymbol{\theta})\rangle\langle\psi(\boldsymbol{\theta})|_{RS}]$ denote the reduced density operator of $|\psi(\boldsymbol{\theta})\rangle_{RS}$. By using the fact that $\mathrm{Tr}[H_k\rho(\boldsymbol{\theta})] = \langle\psi(\boldsymbol{\theta})|_{RS}(I_R \otimes H_k)|\psi(\boldsymbol{\theta})\rangle_{RS}$ and under the assumption that the parameterized circuit $U_{RS}(\boldsymbol{\theta})$ is fully expressive, the objective function in (18) and its associated optimization problem (19) can be written in terms of the parameter $\boldsymbol{\theta}$ as follows:

$$\mathcal{F}(\boldsymbol{\theta}) := \sum_k g_k(\langle\psi(\boldsymbol{\theta})|_{RS}(I_R \otimes H_k)|\psi(\boldsymbol{\theta})\rangle_{RS}) \tag{21}$$

$$= \sum_k g_k(\langle I_R \otimes H_k\rangle_{\boldsymbol{\theta}}), \tag{22}$$

$$\boldsymbol{\theta}^* := \arg\min_{\boldsymbol{\theta}\in[0,2\pi]^r} \mathcal{F}(\boldsymbol{\theta}), \tag{23}$$

where it is implicit that H_k acts on system S.

Assumption 1. *We assume that the objective function in* (22) *is 'faithful,' which means that the minimum of this objective function corresponds to the optimal value of the problem* (19).

A parameterized quantum circuit $U(\boldsymbol{\theta})$ is also known as a variational ansatz, and its choice plays an important role in obtaining an approximation of the optimal value. Throughout this paper, we use the terms "parameterized quantum circuit" and "variational ansatz" interchangeably. Furthermore, there are problem-specific ansatzes as well as problem-independent ansatzes. For our case, we use a problem-independent ansatz having the following form:

$$U(\boldsymbol{\theta}) = U_r(\theta_r)U_{r-1}(\theta_{r-1})\cdots U_1(\theta_1), \tag{24}$$

where each unitary $U_j(\theta_j)$ is written as

$$U_j(\theta_j) = e^{-i\theta_j H_j} W_j, \tag{25}$$

with W_j as an unparameterized unitary.

Assumption 2. *We assume that the number of parameters, r, of a parameterized circuit is $O(\text{poly}(n))$.*

The above assumption is natural in the context of variational quantum algorithms, as we only have access to quantum circuits with short depth.

Each Hermitian operator H_k is arbitrary. In general, we can express a Hermitian operator as a weighted sum of tensor products of Pauli operators, i.e.,

$$H = \sum_{i=1}^{p} w_i \ \sigma_{i,1} \otimes \cdots \otimes \sigma_{i,n}, \tag{26}$$

where $w_i \in \mathbb{R}$, $\sigma_{i,j} \in \{I, \sigma_x, \sigma_y, \sigma_z\}$, and σ_x, σ_y, and σ_z are the Pauli operators. From (26) we see that the expectation value of an arbitrary Hermitian operator is equal to a linear combination of the expectation values of each tensor product of Pauli operators. However, in general, this linear combination may contain many terms.

Assumption 3. *We assume that the number of terms in (26) is polynomial in n; i.e., $p = O(\text{poly}(n))$.*

The above three assumptions are standard in the literature on variational quantum algorithms. Overall, VQAs use a quantum circuit with parameter $\boldsymbol{\theta}$ to estimate the expectation value of a given Hermitian matrix, and they utilize a classical optimizer to solve the optimization problem $\min_{\boldsymbol{\theta}\in[0,2\pi]^r} \mathcal{F}(\boldsymbol{\theta})$. In each round, a VQA updates the parameters of the quantum circuit according to a classical optimization algorithm. We can update these parameters according to gradient-free approaches that include Nelder–Mead [44], Simultaneous Perturbation Stochastic Approximation (SPSA) [45], and Particle Swarm Optimization [46]. The main drawbacks of such gradient-free methods include slower convergence and less robustness against noise. On the contrary, if the evaluation of the gradient of a given objective function is not computationally expensive, then first-order methods like gradient descent are more suitable.

VQAs have an advantage as they do not use a classical method to evaluate gradients. Instead, they evaluate the partial derivatives of $\mathcal{F}(\boldsymbol{\theta})$ with respective to each parameter using the same quantum circuit but with shifted parameters. It is possible to evaluate the partial derivatives on a quantum computer using the parameter-shift rule [29, 30, 31]. Formally, we state the parameter-shift rule for the evaluation of the partial derivatives of $\langle H\rangle_{\boldsymbol{\theta}}$ with respect to its j^{th} component, i.e., θ_j, as

$$\frac{\partial\langle H\rangle_{\boldsymbol{\theta}}}{\partial\theta_j} = \frac{1}{2}\left(\langle H\rangle_{\boldsymbol{\theta}+(\pi/2)\hat{\boldsymbol{e}}_j} - \langle H\rangle_{\boldsymbol{\theta}-(\pi/2)\hat{\boldsymbol{e}}_j}\right), \tag{27}$$

where $\hat{\boldsymbol{e}}_j$ is a unit vector with 1 as its j^{th} element and 0 otherwise.

From (27), we note that for the exact evaluation of the partial derivatives of $\langle H \rangle_{\boldsymbol{\theta}}$ at a parameter value $\boldsymbol{\theta}$, we need to compute the expectation values of this operator at the shifted parameters exactly. However, the exact evaluation of the expectation value of an operator requires an infinite number of measurements on a quantum circuit. In reality, we perform only a limited number of measurements and then take the average of those values. Therefore, it is important to have an unbiased estimator of the expectation value. We assume that this is the case for all of our algorithms. Such an assumption is standard in the literature, as it acts as a good starting point for understanding the nature of the algorithms under such settings.

Unbiased Estimator: First, we consider a simple objective function consisting of a single expectation value term, i.e.,

$$\mathcal{F}(\boldsymbol{\theta}) = \langle H \rangle_{\boldsymbol{\theta}}. \tag{28}$$

Now, we define a k-sample mean unbiased estimator of this expectation value as follows:

Definition 12 (k-sample mean unbiased estimator of expectation value)**.** *Given a quantum circuit $U(\boldsymbol{\theta})$ with parameter $\boldsymbol{\theta} \in [0, 2\pi]^r$, we define $u_k^H(\boldsymbol{\theta})$ as an average of k measurements of the observable H with respect to the pure state $U(\boldsymbol{\theta})|\mathbf{0}\rangle$. It is a k-sample mean unbiased estimator of (28) if the following holds:*

$$\mathbb{E}[u_k^H(\boldsymbol{\theta})] = \langle H \rangle_{\boldsymbol{\theta}}. \tag{29}$$

Next, we define an unbiased estimator of the partial derivatives of $\langle H \rangle_{\boldsymbol{\theta}}$. According to the parameter-shift rule in (27), the partial derivative of the expectation values of a Hermitian operator is a linear combination of its expectation values with shifted parameters. Setting

$$g_j^{H,k}(\boldsymbol{\theta}) := \frac{1}{2}\left(u_k^H(\boldsymbol{\theta} + (\pi/2)\hat{\boldsymbol{e}}_j) - u_k^H(\boldsymbol{\theta} - (\pi/2)\hat{\boldsymbol{e}}_j\right), \tag{30}$$

it follows that $g_j^{H,k}(\boldsymbol{\theta})$ is an unbiased estimator for that partial derivative because

$$\mathbb{E}\Big[g_j^{H,k}(\boldsymbol{\theta})\Big] = \mathbb{E}\left[\frac{1}{2}\big(u_k^H(\boldsymbol{\theta} + (\pi/2)\hat{\boldsymbol{e}}_j) - u_k^H(\boldsymbol{\theta} - (\pi/2)\hat{\boldsymbol{e}}_j)\right] \tag{31}$$

$$= \frac{1}{2}\left(\mathbb{E}[u_k^H(\boldsymbol{\theta} + (\pi/2)\hat{\boldsymbol{e}}_j)] - \mathbb{E}[u_k^H(\boldsymbol{\theta} - (\pi/2)\hat{\boldsymbol{e}}_j)]\right) \tag{32}$$

$$= \frac{1}{2}\left(\langle H \rangle_{\boldsymbol{\theta}+(\pi/2)\hat{\boldsymbol{e}}_j} - \langle H \rangle_{\boldsymbol{\theta}-(\pi/2)\hat{\boldsymbol{e}}_j}\right) \tag{33}$$

$$= \frac{\partial \langle H \rangle_{\boldsymbol{\theta}}}{\partial \theta_j}. \tag{34}$$

We denote an unbiased estimator of the full gradient, i.e., $\nabla_\theta \langle H \rangle_{\boldsymbol{\theta}}$, as $g^{H,k}(\boldsymbol{\theta})$. Furthermore, we evaluate the mean square error of $g^{H,k}(\boldsymbol{\theta})$ as follows,

$$\mathbb{E}\left[\left\|g^{H,k}(\boldsymbol{\theta}) - \nabla_\theta \langle H \rangle_{\boldsymbol{\theta}}\right\|^2\right]$$

$$= \mathbb{E}\left[\sum_{j=1}^{r}\left(g_j^{H,k}(\boldsymbol{\theta}) - \frac{\partial \langle H \rangle_{\boldsymbol{\theta}}}{\partial \theta_j}\right)^2\right] \tag{35}$$

$$= \sum_{j=1}^{r}\mathbb{E}\left[\left(g_j^{H,k}(\boldsymbol{\theta}) - \frac{\partial \langle H \rangle_{\boldsymbol{\theta}}}{\partial \theta_j}\right)^2\right] \tag{36}$$

$$= \sum_{j=1}^{r} \mathrm{Var}\Big(g_j^{H,k}(\boldsymbol{\theta})\Big) \tag{37}$$

$$= \sum_{j=1}^{r} \frac{1}{4}\mathrm{Var}\big(u_k^H(\boldsymbol{\theta} + (\pi/4)\hat{\boldsymbol{e}}_j) - u_k^H(\boldsymbol{\theta} - (\pi/4)\hat{\boldsymbol{e}}_j\big) \tag{38}$$

$$= \frac{1}{4}\sum_{j=1}^{r} \mathrm{Var}\big(u_k^H(\boldsymbol{\theta} + (\pi/4)\hat{\boldsymbol{e}}_j)\big) + \mathrm{Var}\big(u_k^H(\boldsymbol{\theta} - (\pi/4)\hat{\boldsymbol{e}}_j\big) \tag{39}$$

$$= \frac{1}{4}\sum_{j=1}^{r} \frac{1}{k}\mathrm{Var}\big(u_1^H(\boldsymbol{\theta} + (\pi/4)\hat{\boldsymbol{e}}_j)\big) + \frac{1}{k}\mathrm{Var}\big(u_1^H(\boldsymbol{\theta} - (\pi/4)\hat{\boldsymbol{e}}_j)\big)\,, \tag{40}$$

where we denote the variance of a random variable X as $\mathrm{Var}(X)$. The fourth equality follows from the definition of $g_j^{H,k}(\boldsymbol{\theta})$, given by (30). The fifth equality uses the fact that $\mathrm{Var}(X - Y) = \mathrm{Var}(X) + \mathrm{Var}(Y)$ if X and Y are independent random variables. For our case, the sample means $u_k^H(\boldsymbol{\theta}+(\pi/4)\hat{\boldsymbol{e}}_j)$ and $u_k^H(\boldsymbol{\theta}-(\pi/4)\hat{\boldsymbol{e}}_j)$ are independent because they are computed using two different quantum circuit evaluations. For computing gradients in our algorithms, we take a sufficient number of samples, k, such that the mean square error is very small throughout the paper. This implies that the stochastic gradient $(g^{H,k}(\boldsymbol{\theta}))$ is almost equal to the exact gradient $(\nabla_\theta \langle H \rangle_{\boldsymbol{\theta}})$.

Potential quantum advantage of a VQA when N is large: According to (26), we consider a Hermitian operator H consisting of a sum of p weighted Pauli strings where $p = O(\mathrm{poly}(n))$ (Assumption 3). We can measure a given Pauli string P_i of arbitrary size with respect to a given quantum state in constant time. If the desired precision of the expectation value's estimate is ϵ, then we need to make $O(1/\epsilon^2)$ repetitions of the procedure. Here the procedure consists of preparing the quantum state and then measuring the expectation value. As we have p Pauli strings, overall we need no more than $O(\max_i |w_i|^2\, p/\epsilon^2)$ repetitions to estimate the expectation value of H to precision ϵ. In contrast, if the evaluation of the expectation value of a Pauli string P_i with respect to a general quantum state is conducted using known classical algorithms, it appears that we need $O(2^n)$ time, as well as space. This is because the size of P_i is $2^n \times 2^n$, and the same goes for the size of the matrix needed to represent the quantum state. Therefore, if all the above mentioned assumptions hold, then we have an exponential advantage over the best known classical algorithms for estimating the expectation value of a Hermitian operator. A similar argument can be provided for estimating gradients because the partial derivatives of the expectation value with respect to quantum circuit parameters depend on the expectation value evaluated on shifted parameters.

3 Variational Quantum Algorithms for SDPs

3.1 General Form (GF) of SDPs

In this section, we consider the general form of an SDP, as given in (16). We write it concisely as follows:

$$p^* = \sup_{X \succcurlyeq 0} \{\mathrm{Tr}[CX] : \Phi(X) \preccurlyeq B\}\,. \tag{41}$$

Next, we modify the above formulation as follows:

$$p^* = \sup_{X \succcurlyeq 0} \{\mathrm{Tr}[CX] : \Phi(X) \preccurlyeq B\} \tag{42}$$

$$\overset{(a)}{=} \sup_{X \succcurlyeq 0} \left\{ \mathrm{Tr}[CX] + \inf_{Y \succcurlyeq 0} \{\mathrm{Tr}[(B - \Phi(X))Y]\} \right\} \tag{43}$$

$$\overset{(b)}{=} \sup_{X \succcurlyeq 0} \inf_{Y \succcurlyeq 0} \{\mathrm{Tr}[CX] + \mathrm{Tr}[BY] - \mathrm{Tr}[Y\Phi(X)]\} \tag{44}$$

$$\overset{(c)}{=} \sup_{X \succcurlyeq 0} \inf_{Y \succcurlyeq 0} \left\{\mathrm{Tr}[CX] + \mathrm{Tr}[BY] - \mathrm{Tr}[(X^\top \otimes Y)\Gamma^\Phi]\right\} \tag{45}$$

$$= \sup_{X \succcurlyeq 0} \inf_{Y \succcurlyeq 0} \left\{\mathrm{Tr}[C^\top X^\top] + \mathrm{Tr}[BY] - \mathrm{Tr}[(X^\top \otimes Y)\Gamma^\Phi]\right\} \tag{46}$$

$$= \sup_{X \succcurlyeq 0} \inf_{Y \succcurlyeq 0} \left\{\mathrm{Tr}[C^\top X] + \mathrm{Tr}[BY] - \mathrm{Tr}[(X \otimes Y)\Gamma^\Phi]\right\} \tag{47}$$

$$\overset{(d)}{=} \sup_{\substack{\rho \in \mathcal{D}^N, \\ \lambda \geqslant 0}} \inf_{\substack{\sigma \in \mathcal{D}^M, \\ \mu \geqslant 0}} \left\{\lambda \,\mathrm{Tr}[C^\top \rho] + \mu \,\mathrm{Tr}[B\sigma] - \lambda\mu \,\mathrm{Tr}[\Gamma^\Phi(\rho \otimes \sigma)]\right\}. \tag{48}$$

Equality (a) follows due to the equivalence between both problems. If we pick a primal PSD variable X that does not satisfy the constraint, i.e., $\Phi(X) \not\leqslant B$, then the inner minimization results in the value $-\infty$. This is because, in such a case, there exists at least one negative eigenvalue of $B - \Phi(X)$. If we set $Y = s|e_i\rangle\langle e_i|$, where $|e_i\rangle$ is a unit vector in the negative eigenspace corresponding to that eigenvalue, then, due to the inner minimization, we can take the limit $s \to \infty$. This in turn implies that $\inf_{Y \succcurlyeq 0} \{\mathrm{Tr}[(B - \Phi(X))Y]\} = -\infty$. In other words, the inner minimization forces the outer maximization to pick a feasible X and imposes an infinite penalty if chosen otherwise.

Equality (b) follows by taking the infimum outside. This formulation is the Lagrangian of the original primal problem (1), where

$$\mathcal{L}(X, Y) \coloneqq \mathrm{Tr}[CX] + \mathrm{Tr}[BY] - \mathrm{Tr}[Y\Phi(X)] \tag{49}$$

is a Lagrangian and Y is a dual PSD variable.

Equality (c) follows from the definition of the Choi representation of the linear map Φ (see Definition 4), where Γ^Φ is the Choi operator of Φ. According to Lemma 5, the Choi operator Γ^Φ is Hermitian.

Equality (d) follows from the substitution: $X = \lambda\rho$ and $Y = \mu\sigma$, where $\rho \in \mathcal{D}^N, \sigma \in \mathcal{D}^M$, $\lambda = \mathrm{Tr}[X]$, and $\mu = \mathrm{Tr}[Y]$.

Now, we are interested in solving the following unconstrained optimization problem, i.e., the equality (d):

$$p^* = \sup_{\substack{\rho \in \mathcal{D}^N, \\ \lambda \geqslant 0}} \inf_{\substack{\sigma \in \mathcal{D}^M, \\ \mu \geqslant 0}} \left\{\lambda \,\mathrm{Tr}[C^\top \rho] + \mu \,\mathrm{Tr}[B\sigma] - \lambda\mu \,\mathrm{Tr}[\Gamma^\Phi(\rho \otimes \sigma)]\right\}. \tag{50}$$

This optimization problem is now expressed in terms of the expectation values of the Hermitian operators $C^\top$, B, and Γ^Φ with respect to the density operators ρ, σ, and $\rho \otimes \sigma$, respectively.

3.1.1 Variational Quantum Algorithm for SDPs in GF

As stated earlier, when the dimension N of the Hermitian operators is large, solving the problem (50) is generally intractable using a gradient-based classical algorithm. Therefore, we propose a variational quantum algorithm for the optimization problem (50).

First, we introduce a parameterization of the density operators, i.e., ρ and σ, by using parameterized quantum circuits. Second, we optimize the modified objective function of the problem (50) over the parameters of those quantum circuits using our variational quantum algorithm.

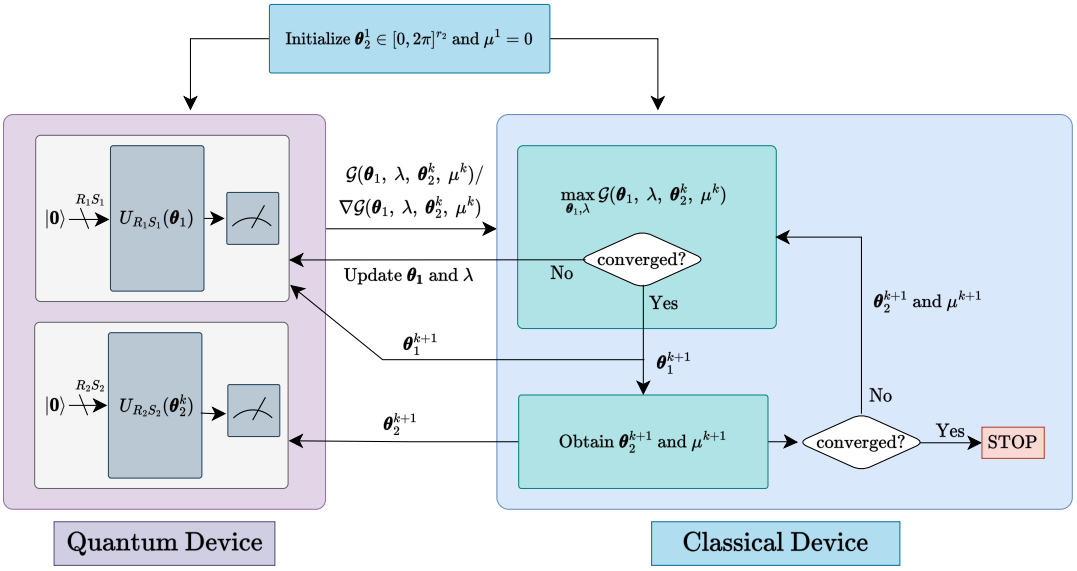


Figure 1: This figure depicts iVQAGF algorithm where we utilize two parameterized quantum circuits, i.e., $U_{R_1S_1}(\boldsymbol{\theta}_1)$ and $U_{R_2S_2}(\boldsymbol{\theta}_2)$.

Parameterization: Let $\rho_{S_1}(\boldsymbol{\theta}_1)$ be the density operator prepared by first applying the quantum circuit $U_{R_1S_1}(\boldsymbol{\theta}_1)$ to the all-zeros state of the quantum system R_1S_1 and then tracing out the system R_1. Similarly, let $\sigma_{S_2}(\boldsymbol{\theta}_2)$ be the density operator prepared by first applying the quantum circuit $U_{R_2S_2}(\boldsymbol{\theta}_2)$ to the all-zeros state of the quantum system R_2S_2 and then tracing out the system R_2. Here, $\boldsymbol{\theta}_1 \in [0, 2\pi]^{r_1}$ and $\boldsymbol{\theta}_2 \in [0, 2\pi]^{r_2}$. Also, we set $r_1, r_2 = O(\text{poly}(n))$. Defining

$$|\psi(\boldsymbol{\theta}_1)\rangle_{R_1S_1} := U_{R_1S_1}(\boldsymbol{\theta}_1)|\mathbf{0}\rangle_{R_1S_1}, \tag{51}$$
$$|\varphi(\boldsymbol{\theta}_2)\rangle_{R_2S_2} := U_{R_2S_2}(\boldsymbol{\theta}_2)|\mathbf{0}\rangle_{R_2S_2}, \tag{52}$$

we have that

$$\rho_{S_1}(\boldsymbol{\theta}_1) = \text{Tr}_{R_1}[|\psi(\boldsymbol{\theta}_1)\rangle\langle\psi(\boldsymbol{\theta}_1)|_{R_1S_1}], \tag{53}$$
$$\sigma_{S_2}(\boldsymbol{\theta}_2) = \text{Tr}_{R_2}[|\varphi(\boldsymbol{\theta}_2)\rangle\langle\varphi(\boldsymbol{\theta}_2)|_{R_2S_2}]. \tag{54}$$

Furthermore, the parameterized quantum circuits, i.e., $U_{R_1S_1}(\boldsymbol{\theta}_1)$ and $U_{R_2S_2}(\boldsymbol{\theta}_2)$, are of the form shown in (24). As these quantum circuits generate purifications of density operators, the following equalities hold:

$$\begin{aligned}
\text{Tr}[C^\top \rho_{S_1}(\boldsymbol{\theta}_1)] &= \langle\psi(\boldsymbol{\theta}_1)|_{R_1S_1} \left(I_{R_1} \otimes C_{S_1}^\top\right) |\psi(\boldsymbol{\theta}_1)\rangle_{R_1S_1} && (55)\\
&= \langle I \otimes C^\top\rangle_{\boldsymbol{\theta}_1}, && (56)\\
\text{Tr}[B\sigma_{S_2}(\boldsymbol{\theta}_2)] &= \langle\varphi(\boldsymbol{\theta}_2)|_{R_2S_2} \left(I_{R_2} \otimes B_{S_2}\right) |\varphi(\boldsymbol{\theta}_2)\rangle_{R_2S_2} && (57)\\
&= \langle I \otimes B\rangle_{\boldsymbol{\theta}_2}, && (58)\\
\text{Tr}[\Gamma^\Phi(\rho_{S_1}(\boldsymbol{\theta}_1) \otimes \sigma_{S_2}(\boldsymbol{\theta}_2))] &= \langle\phi(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)|_{R_1R_2S_1S_2} \left(I_{R_1} \otimes I_{R_2} \otimes \Gamma^\Phi_{S_1S_2}\right) |\phi(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)\rangle_{R_1R_2S_1S_2} \\
&= \langle I \otimes I \otimes \Gamma^\Phi\rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2}, && (59)
\end{aligned}$$

where

$$|\phi(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)\rangle_{R_1R_2S_1S_2} := |\psi(\boldsymbol{\theta}_1)\rangle_{R_1S_1} \otimes |\varphi(\boldsymbol{\theta}_2)\rangle_{R_2S_2}. \tag{60}$$

Due to the parameterization of the density operators $\rho_{S_1}(\boldsymbol{\theta}_1)$ and $\sigma_{S_2}(\boldsymbol{\theta}_2)$, the problem (50) transforms into the following optimization problem:

$$p^* := \sup_{\substack{\boldsymbol{\theta}_1 \in [0,2\pi]^{r_1}, \\ \lambda \geqslant 0}} \inf_{\substack{\boldsymbol{\theta}_2 \in [0,2\pi]^{r_2}, \\ \mu \geqslant 0}} \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu), \tag{61}$$

where

$$\mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu) := \lambda \langle I \otimes C^\top \rangle_{\boldsymbol{\theta}_1} + \mu \langle I \otimes B \rangle_{\boldsymbol{\theta}_2} - \lambda\mu \langle I \otimes I \otimes \Gamma^\Phi \rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2} \tag{62}$$

and we optimize over the space of quantum circuit parameters $\boldsymbol{\theta}_1 \in [0, 2\pi]^{r_1}$ and $\boldsymbol{\theta}_2 \in [0, 2\pi]^{r_2}$. As mentioned before, we assume that the objective function $\mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ is faithful, which means that the global optimal value of the optimization problem (61) is equal to p^*. Additionally, the objective function of (61) is generally non-convex as a function of the quantum circuit parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. Hence, the max-min problem (61) is a non-convex–non-concave optimization problem. Due to the fact that obtaining a globally optimal point of a non-convex–non-concave function is generally NP-hard [33, Section 2.1], we focus on finding first order ϵ-stationary points of (61). As a global optimal point is also a stationary point, if we use techniques to initialize quantum circuit parameters according to the problem at hand such that these parameters lie in the vicinity of a global optimal point, then our algorithm will converge to that point.

Definition 13 (First order ϵ-stationary point of (61))**.** *A point $(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ is a first order ϵ-stationary point of (61) if and only if $\|\nabla \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)\| \leqslant \epsilon$.*

We propose a variational quantum algorithm for obtaining a first order ϵ-stationary point of (61). We call this algorithm *inexact Variational Quantum Algorithm for General Form* (iVQAGF), and the pseudocode for this algorithm is provided in Algorithm 1. It is an inexact version because we solve the subproblem involving a maximization (Step 4) to approximate stationary points instead of solving it until global optimality is reached, due to the non-convex nature of the objective function in terms of the quantum circuit parameters. Furthermore, iVQAGF is a hybrid quantum-classical algorithm, as we have two parameterized quantum circuits for estimating expectation values of Hermitian operators at any given step and a classical optimizer that updates the parameters of these quantum circuits as per the algorithm (see Figure 1).

At any step of the algorithm, the expectation value $\langle H \rangle_{\boldsymbol{\theta}}$ of a Hermitian operator H is evaluated using a quantum circuit with parameter $\boldsymbol{\theta}$. Moreover, the partial derivatives of $\mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ with respect to the parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ depend on the partial derivatives of the expectation values of the Hermitian operators $I \otimes C^\top$, $I \otimes B$, and $I \otimes I \otimes \Gamma^\Phi$ with respect to those parameters. We evaluate the partial derivative of the expectation value of a Hermitian operator with respect to the quantum circuit parameters using the parameter-shift rule (see (27)). We do not explicitly mention these quantum-circuit calls in the main algorithm.

Unbiased Estimators: Due to the fact that the evaluation of the expectation value of a Hermitian operator using a quantum circuit is stochastic in nature, we use unbiased estimators of expectation values and their corresponding partial derivatives (see (29) and (31)). Now, from (62), we can see that $\nabla_{\boldsymbol{\theta}_1} \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ is a linear combination of $\nabla_{\boldsymbol{\theta}_1} \langle I \otimes C^\top \rangle_{\boldsymbol{\theta}_1}$ and $\nabla_{\boldsymbol{\theta}_1} \langle I \otimes I \otimes \Gamma^\Phi \rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2}$. Therefore, the unbiased estimator of $\nabla_{\boldsymbol{\theta}_1} \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ is also a linear combination of the unbiased estimators of $\nabla_{\boldsymbol{\theta}_1} \langle I \otimes C^\top \rangle_{\boldsymbol{\theta}_1}$ and $\nabla_{\boldsymbol{\theta}_1} \langle I \otimes I \otimes \Gamma^\Phi \rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2}$. Similarly, the unbiased estimator of $\nabla_{\boldsymbol{\theta}_2} \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ is a linear combination of the unbiased estimators of $\nabla_{\boldsymbol{\theta}_2} \langle I \otimes B \rangle_{\boldsymbol{\theta}_2}$ and $\nabla_{\boldsymbol{\theta}_2} \langle I \otimes I \otimes \Gamma^\Phi \rangle_{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2}$. Also, the unbiased estimators of $\nabla_\lambda \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ and $\nabla_\mu \mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ depend on a linear combination of the unbiased

Algorithm 1 `iVQAGF`$(\Gamma^{\Phi}, C, B, \eta_1, \eta_2, \epsilon)$

Input: Hermitian operators Γ^{Φ}, C, and B, learning rates $\eta_1, \eta_2 > 0$, precision $\epsilon > 0$.

Initialization: $\boldsymbol{\theta}_2^1 \in [0, 2\pi]^{r_2}$, $\mu^1 = 0$.

 # For any step, expectation values of observables and their gradients are evaluated using parameterized quantum circuits.

for $k = 1, 2, \ldots,$ **do**

Maximize $\mathcal{G}(\cdot,\ \cdot,\ \boldsymbol{\theta}_2^k,\ \mu^k)$ using a first order method such as gradient descent, where $\boldsymbol{\theta}_2^k,\ \mu^k$ are fixed, to obtain $(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1})$ such that the following holds:

$$\left\|\nabla\mathcal{G}(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1},\ \boldsymbol{\theta}_2^k,\ \mu^k)\right\| \leqslant \epsilon.$$

$\mu^{k+1} = \mu^k - \eta_1 \nabla_{\mu}\mathcal{G}(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1},\ \boldsymbol{\theta}_2^k,\ \mu^k)$

$\boldsymbol{\theta}_2^{k+1} = \boldsymbol{\theta}_2^k - \eta_2 \nabla_{\boldsymbol{\theta}_2}\mathcal{G}(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1},\ \boldsymbol{\theta}_2^k,\ \mu^k)$

if $\left\|\nabla\mathcal{G}(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1},\ \boldsymbol{\theta}_2^{k+1},\ \mu^{k+1})\right\| \leqslant \epsilon$ **then**

STOP and return $\mathcal{G}(\boldsymbol{\theta}_1^{k+1},\ \lambda^{k+1},\ \boldsymbol{\theta}_2^{k+1},\ \mu^{k+1})$

estimators of expectation values of Hermitian operators, i.e., $\langle I \otimes C^{\top}\rangle_{\boldsymbol{\theta}_1}$, $\langle I \otimes B\rangle_{\boldsymbol{\theta}_2}$, and $\langle I \otimes I \otimes \Gamma^{\Phi}\rangle_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2}$. Finally, as we have the unbiased estimators of all these partial derivatives, we have an unbiased estimator of the full gradient $\nabla\mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$ of the function $\mathcal{G}(\boldsymbol{\theta}_1, \lambda, \boldsymbol{\theta}_2, \mu)$.

3.2 Equality Constrained Standard Form (ECSF) of SDPs

In this section, we shift our focus to the standard form of SDPs. Due to their specific problem structure and the aforementioned assumption of it being weakly constrained (i.e, $N \gg M$), the design of more sophisticated variational quantum algorithms for solving them is possible. Moreover, we establish a convergence rate for one of the algorithms.

We consider the standard form of SDPs as given in (15). Here, the Hermiticity-preserving map Φ and the Hermitian operator B have a diagonal form:

$$\Phi(X) = \operatorname{diag}\left(\operatorname{Tr}[A_1 X], \ldots, \operatorname{Tr}[A_M X]\right), \tag{63}$$

$$B = \operatorname{diag}\left(b_1, \ldots, b_M\right), \tag{64}$$

where $A_1, \ldots, A_M \in \mathcal{S}^N$ and $b_1, \ldots, b_M \in \mathbb{R}$.

The equality constrained primal SDP is given as follows:

$$p^* = \sup_{X \succcurlyeq 0} \left\{\operatorname{Tr}[CX] : \boldsymbol{\Phi}(X) = \boldsymbol{b}\right\}, \tag{65}$$

where $\boldsymbol{b} = (b_1, \ldots, b_M)^{\top}$ and $\boldsymbol{\Phi}$ is the vector form of the original linear map, i.e., $\boldsymbol{\Phi}(X) = (\operatorname{Tr}[A_1 X], \ldots, \operatorname{Tr}[A_M X])^{\top}$. Taking this formulation into account, we write the optimization over the Lagrangian $\mathcal{L}(X, Y)$ in (49) as

$$p^* = \sup_{X \succcurlyeq 0} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \mathcal{L}(X, \boldsymbol{y}), \tag{66}$$

where

$$\mathcal{L}(X, \boldsymbol{y}) := \mathrm{Tr}[CX] + \boldsymbol{y}^\top (\boldsymbol{b} - \boldsymbol{\Phi}(X)) \tag{67}$$

and $\boldsymbol{y} \in \mathbb{R}^M$ is the dual vector of the Lagrangian $\mathcal{L}(X, \boldsymbol{y})$. As discussed earlier, the inner minimization with respect to $\boldsymbol{y}$ results in $-\infty$ if X violates the constraint in (65). On the contrary, if there exists a PSD operator X that satisfies the constraints, then (66) reduces to $\sup_{X \succcurlyeq 0} \mathrm{Tr}[CX]$. This is because $\inf_{\boldsymbol{y} \in \mathbb{R}^M} \boldsymbol{y}^\top (\boldsymbol{b} - \boldsymbol{\Phi}(X)) = 0$ in this case. Hence, there is an equivalence between (65) and (66), as indicated previously.

For the equality constrained problem (65), we consider the Augmented Lagrangian $\mathcal{L}_c(X, \boldsymbol{y})$ as the objective function, which consists of a quadratic penalty term $\frac{c}{2} \|\boldsymbol{b} - \boldsymbol{\Phi}(X)\|^2$ in addition to the terms of the original Lagrangian $\mathcal{L}(X, \boldsymbol{y})$. Therefore, the optimization problem with the modified objective function can be written as follows:

$$p^* = \sup_{X \succcurlyeq 0} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \mathcal{L}_c(X, \boldsymbol{y}), \tag{68}$$

where $c > 0$ is the penalty parameter and

$$\mathcal{L}_c(X, \boldsymbol{y}) := \mathrm{Tr}[CX] + \boldsymbol{y}^\top (\boldsymbol{b} - \boldsymbol{\Phi}(X)) - \frac{c}{2} \|\boldsymbol{b} - \boldsymbol{\Phi}(X)\|^2 . \tag{69}$$

The method for optimization associated with the Augmented Lagrangian formulation is known as the Augmented Lagrangian Method (ALM), independently introduced in [47] and [48]. This classical method involves the following three steps that iterate until convergence:

1. $X^{k+1} := \arg\max_{X \succcurlyeq 0} \mathcal{L}_c(X, \boldsymbol{y}^k)$,
2. Update the dual variable $\boldsymbol{y}$ according to $\boldsymbol{y}^{k+1} := \boldsymbol{y}^k - c\,(\boldsymbol{b} - \boldsymbol{\Phi}(X))$,
3. Update the penalty parameter c according to $c_{k+1} := c_0 \mu^{k+1}$, where $\mu > 1$.

Due to the equivalence between (65) and (68), iterating through the above steps until convergence leads to the primal optimal value. Hence, we can write

$$p^* = \sup_{X \succcurlyeq 0} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \mathcal{L}_c(X, \boldsymbol{y}) \tag{70}$$

$$= \sup_{X \succcurlyeq 0} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \left\{ \mathrm{Tr}[CX] + \boldsymbol{y}^\top (\boldsymbol{b} - \boldsymbol{\Phi}(X)) - \frac{c}{2} \|\boldsymbol{b} - \boldsymbol{\Phi}(X)\|^2 \right\}. \tag{71}$$

ALM is a well-studied method for solving unconstrained optimization problems with convex objective functions. It is considered superior to methods that exclusively involve the original Lagrangian or exclusively involve the penalty term [49]. ALM converges faster than the original Lagrangian method, as it involves a quadratic penalty term in addition to a linear penalty term of the original Lagrangian. Furthermore, it solves many issues associated with both these methods, such as ill-conditioning arising due to large values of the penalty parameter.

Now, we make a mild assumption that one of the constraints is a trace constraint on the primal PSD operator X. Specifically, let $A_M = \mathbb{I}$ and $b_M = \lambda$. Hence, the constraint is $\mathrm{Tr}[X] = \lambda$. Subsequently, substituting $X = \lambda\rho$, where $\rho \in \mathcal{D}^N$ in (70), we obtain the following:

$$p^* = \sup_{\rho \in \mathcal{D}^N} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \mathcal{L}_c(\lambda\rho, \boldsymbol{y}) \tag{72}$$

$$= \sup_{\rho \in \mathcal{D}^N} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \left\{ \lambda\, \mathrm{Tr}[C\rho] + \boldsymbol{y}^\top (\boldsymbol{b} - \lambda \boldsymbol{\Phi}(\rho)) - \frac{c}{2} \|\boldsymbol{b} - \lambda \boldsymbol{\Phi}(\rho)\|^2 \right\}. \tag{73}$$

This optimization problem is now expressed in terms of the expectation values of the Hermitian operators $C,\ A_1, \ldots, A_M$ with respect to the density operator ρ. According to Assumption 3, the matrices $C,\ A_1, \ldots, A_M$ consist of poly(n) terms.

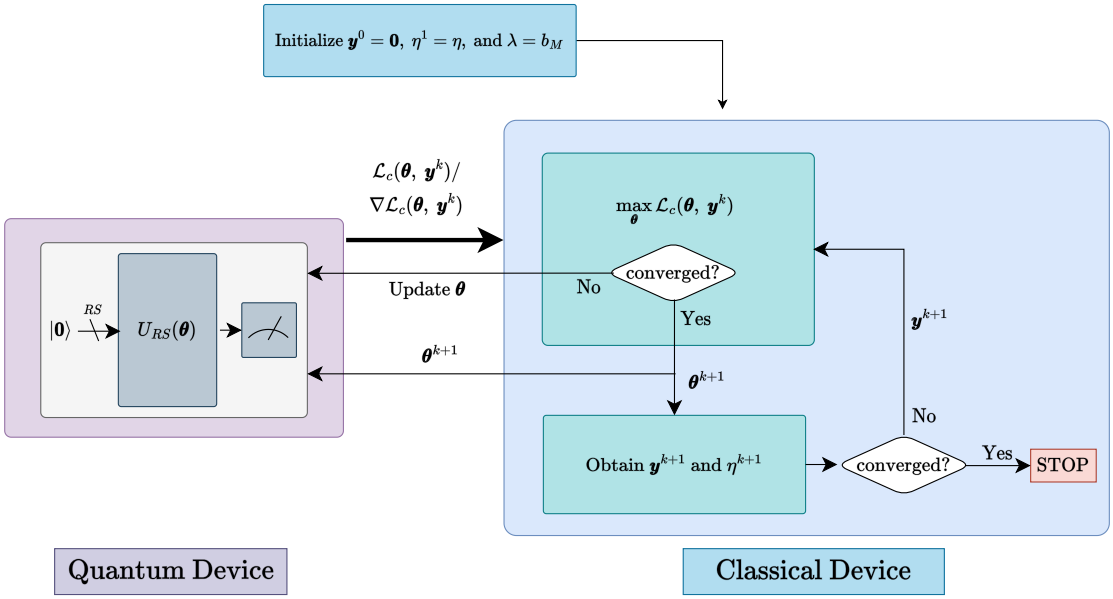


Figure 2: This figure depicts the iVQAEC algorithm in which we utilize one parameterized quantum circuit, i.e., $U_{RS}(\boldsymbol{\theta})$.

3.2.1 Variational Quantum Algorithm for SDPs in ECSF

Solving the unconstrained optimization problem (72) using ALM is computationally intractable if the dimension N of the operators is large. Therefore, we propose a variational quantum algorithm to solve this problem.

As before, we first introduce a parameterization of the density matrix ρ and then optimize the modified objective function over the subspace of those parameters using our variational quantum algorithm.

Parameterization: Let $U_{RS}(\boldsymbol{\theta})$ be a parameterized quantum circuit with parameter $\boldsymbol{\theta} \in [0, 2\pi]^r$, and suppose that it acts on the all-zeros state of the quantum system RS and prepares a purification of the density operator $\rho_S(\boldsymbol{\theta})$. The form of $U_{RS}(\boldsymbol{\theta})$ is given by (24). Moreover, the following equality holds because $U_{RS}(\boldsymbol{\theta})$ generates a purification of $\rho_S(\boldsymbol{\theta})$:

$$\mathrm{Tr}[H\rho_S(\boldsymbol{\theta})] = \langle \mathbf{0}|_{RS} U_{RS}^\dagger(\boldsymbol{\theta}) \left(I_R \otimes H_S\right) U_{RS}(\boldsymbol{\theta}) |\mathbf{0}\rangle_{RS} = \langle I \otimes H \rangle_{\boldsymbol{\theta}}, \tag{74}$$

where $H \in \{C, A_1, \ldots, A_M\}$.

Due to the parameterization of the density operator $\rho_S(\boldsymbol{\theta})$, the problem (72) transforms into the following optimization problem:

$$p^* := \sup_{\boldsymbol{\theta} \in [0,2\pi]^r} \inf_{\boldsymbol{y} \in \mathbb{R}^M} \mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y}), \tag{75}$$

where

$$\mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y}) := \lambda \langle I \otimes C \rangle_{\boldsymbol{\theta}} + \boldsymbol{y}^\top \left(\boldsymbol{b} - \lambda \boldsymbol{\Phi}(\boldsymbol{\theta})\right) - \frac{c}{2} \left\| \boldsymbol{b} - \lambda \boldsymbol{\Phi}(\boldsymbol{\theta}) \right\|^2, \tag{76}$$

$$\boldsymbol{\Phi}(\boldsymbol{\theta}) := \left(\langle I \otimes A_1 \rangle_{\boldsymbol{\theta}}, \ldots, \langle I \otimes A_M \rangle_{\boldsymbol{\theta}}\right)^\top. \tag{77}$$

As mentioned before, we assume that the objective function $\mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y})$ is faithful, which means that the global optimal value of the optimization problem (4) is equal to p^*.

We now focus on solving the optimization problem (4) using the modified Lagrangian $\mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y})$. Note that, in general, the objective function provided above has a non-convex landscape with respect to the quantum circuit parameter $\boldsymbol{\theta}$. Also, it is linear in terms of the dual variable $\boldsymbol{y}$. Therefore, the optimization problem (4) is a non-convex–concave optimization problem. For a convex-concave setting, the natural algorithm to consider is to first perform the maximization until a globally optimal point is reached and then update the dual variable, and repeat these steps until convergence (similar to ALM). On the contrary, for a non-convex–concave setting, solving the maximization until global optimality is reached is generally NP-hard [33, Section 2.1]. Therefore, we focus on obtaining approximate stationary points of (4). As a global optimal point is also a stationary point, if we use techniques to initialize quantum circuit parameters according to the problem at hand, such that these parameters lie in the vicinity of a global optimal point, then our algorithm will converge to that point.

Definition 14 (First order (ϵ, c)-stationary point of (4))**.** *For $\epsilon, c > 0$, a point $\boldsymbol{\theta} \in [0, 2\pi]^r$ is a first order (ϵ, c)-stationary point of (4) if there exists $\boldsymbol{y} \in \mathbb{R}^M$ such that the following hold:*

$$\|\nabla_{\boldsymbol{\theta}} \mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y})\| \leqslant \epsilon, \qquad \|\lambda \boldsymbol{\Phi}(\boldsymbol{\theta}) - \boldsymbol{b}\| \leqslant \epsilon. \tag{78}$$

The above condition acts as a measure of closeness to the first order (ϵ, c)-stationary points of (4). Such a measure is useful to give a stopping criterion for the algorithm.

We now propose a variational quantum algorithm for obtaining a first order (ϵ, c)-stationary point of (4). We call this algorithm *inexact Variational Quantum Algorithm for Equality Constrained standard form* (iVQAEC). It is an inexact version because we solve the subproblem involving the maximization to approximate stationary points instead of solving it until global optimality is reached, due to the nonconcave nature of the objective function in terms of quantum circuit parameters. The pseudocode of iVQAEC is given by Algorithm 2.

We run iVQAEC on a classical computer, and at any step of the algorithm, the expectation value of a Hermitian operator $H \in \{C, A_1, \ldots, A_M\}$ (i.e., $\langle H \rangle_{\boldsymbol{\theta}}$) is evaluated using a quantum circuit with parameter $\boldsymbol{\theta}$ (see Figure 2). Moreover, the partial derivative of $\mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y})$ with respect to the parameter $\boldsymbol{\theta}$ depends on the partial derivatives of the expectation values of the Hermitian operators $I \otimes C$, $I \otimes A_1$, $\ldots$, $I \otimes A_M$ with respect to $\boldsymbol{\theta}$. For evaluating the partial derivatives of the latter with respect to $\boldsymbol{\theta}$, we use the parameter-shift rule (see (27)). Moreover, we do not explicitly mention these calls to quantum circuits in the pseudocode of the algorithm.

Unbiased Estimators: As mentioned earlier in the previous section, due to the fact that evaluation of the expectation value of a Hermitian operator using a quantum circuit is stochastic in nature, we use unbiased estimators of expectation values and their corresponding partial derivatives (see (29) and (31)). Furthermore, it suffices to have *independent* unbiased estimators of $\nabla_{\boldsymbol{\theta}} \langle I \otimes C \rangle_{\boldsymbol{\theta}}$, $\nabla_{\boldsymbol{\theta}} \langle I \otimes A_1 \rangle_{\boldsymbol{\theta}}$, $\ldots$, $\nabla_{\boldsymbol{\theta}} \langle I \otimes A_M \rangle_{\boldsymbol{\theta}}$, $\langle I \otimes A_1 \rangle_{\boldsymbol{\theta}}$, $\ldots$, $\langle I \otimes A_M \rangle_{\boldsymbol{\theta}}$ in order to construct an unbiased estimator of $\nabla_{\boldsymbol{\theta}} \mathcal{L}_c(\boldsymbol{\theta}, \boldsymbol{y})$.

3.2.2 Convergence Rate

In this section, we provide the convergence rate and total iteration complexity of iVQAEC in terms of the number of iterations of the for-loop at Step 3 of Algorithm 2 needed to arrive at an approximate stationary point of (75).

Recently, the authors of [50] studied the following non-convex optimization problem:

$$\inf_{x \in \mathbb{R}^d} \{f(x) + g(x) : \boldsymbol{A}(x) = 0\}, \tag{79}$$

Algorithm 2 `iVQAEC`$(C, \{A_i\}_{i=1}^M,\ \boldsymbol{b},\ \epsilon',\ \eta,\ \mu)$

Input: Hermitian operators $(C, A_1, \ldots, A_M)$, vector $\boldsymbol{b}$, final precision $\epsilon' > 0$, learning rate $\eta > 0$, and a constant $\mu > 1$.

Initialization: $\boldsymbol{y}^1 = \boldsymbol{0}, \eta^1 = \eta, \lambda = b_M$

 # For any step, expectation values of observables and their gradients are evaluated using a parameterized quantum circuit.

for $k = 1, 2, \ldots,$ **do**

  $c_k = -\mu^k,\ \epsilon_{k+1} = 1/\mu^k$

  Maximize $\mathcal{L}_{c_k}(\cdot, \boldsymbol{y}^k)$, where $\boldsymbol{y}^k$ is kept fixed, to obtain $\boldsymbol{\theta}^{k+1}$ such that the following holds:

$$\left\| \nabla \mathcal{L}_{c_k}(\boldsymbol{\theta}^{k+1}, \boldsymbol{y}^k) \right\| \leqslant \epsilon_{k+1}$$

  $\eta^{k+1} = \eta^1 \min\left\{ \frac{\left\|\lambda \boldsymbol{\Phi}(\boldsymbol{\theta}^1) - \boldsymbol{b}\right\| \ln^2 2}{\left\|\lambda \boldsymbol{\Phi}(\boldsymbol{\theta}^{k+1}) - \boldsymbol{b}\right\| (k+1) \ln^2 (k+2)},\ 1 \right\}$

  $\boldsymbol{y}^{k+1} = \boldsymbol{y}^k - \eta^{k+1} \left(\boldsymbol{b} - \lambda \boldsymbol{\Phi}(\boldsymbol{\theta}^{k+1}) \right)$

  if $\left\| \nabla_{\boldsymbol{\theta}} \mathcal{L}_{c_k}(\boldsymbol{\theta}^{k+1}, \boldsymbol{y}^{k+1}) \right\| + \left\| \lambda \boldsymbol{\Phi}(\boldsymbol{\theta}^{k+1}) - \boldsymbol{b} \right\| \leqslant \epsilon'$ **then**

    STOP and return $\mathcal{L}_{c_k}(\boldsymbol{\theta}^{k+1}, \boldsymbol{y}^{k+1})$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a continuously-differentiable non-convex function, $\boldsymbol{A} : \mathbb{R}^d \to \mathbb{R}^m$ is a nonlinear operator, and $g : \mathbb{R}^d \to \mathbb{R}$ is a convex function. The constrained optimization problem (79) is equivalent to the following unconstrained minimax formulation:

$$\inf_{\boldsymbol{x}\in\mathbb{R}^d} \sup_{\boldsymbol{y}\in\mathbb{R}^m} \{\mathcal{L}_\beta(x, \boldsymbol{y}) + g(\boldsymbol{x})\}, \tag{80}$$

where

$$\mathcal{L}_\beta(x, \boldsymbol{y}) := f(\boldsymbol{x}) + \boldsymbol{y}^\top A(\boldsymbol{x}) + \frac{\beta}{2} \|\boldsymbol{A}(\boldsymbol{x})\|^2, \tag{81}$$

$\beta > 1$, and $\mathcal{L}_\beta(x, y)$ is the augmented Lagrangian. As the classic ALM algorithm is suited for solving optimization problems involving convex objective functions, the authors proposed an inexact Augmented Lagrangian method, known as the iALM algorithm for obtaining an approximate first order stationary point of (80). It is an inexact version because the subproblem involving a maximization is over a non-convex function. They claim that if we use an inexact solver for that subproblem at each iteration, then their algorithm converges to an approximate first order stationary point (see Theorem 4.1 of [50]). They also provide the total iteration complexity of iALM in terms of the number of iterations of the for-loop (see Corollary 4.2 of [50]).

Our problem (75) also consists of an augmented Lagrangian term, which is non-convex in terms of the quantum circuit parameters $\boldsymbol{\theta}$, and $g(\boldsymbol{\theta}) = 0$. Additionally, for solving our problem, we use an inexact solver for the subproblem (Step 5) of iVQAEC, and the update rule for modifying the learning rate η at each iteration (Step 6) is similar to their iALM algorithm.

Now, the following theorem characterizes iVQAEC's convergence rate and total iteration complexity for finding an approximate stationary point of the problem (75) in terms of the number of iterations of the for-loop (Step 3). The proof of this theorem is provided by the proofs of Theorem 4.1 and Corollary 4.2 of [50].

Theorem 15 (Convergence rate and total iteration complexity of iVQAEC)**.** *For integers $k_1 \geqslant k_0 \geqslant 2$, consider the interval $K := \{k_0, \ldots, k_1\}$, and let $\{\boldsymbol{\theta}^k\}_{k\in K}$ be the output sequence of iVQAEC on the interval K, where $\boldsymbol{\theta}^k \in [0, 2\pi]^r$ for all $k \in K$. Suppose that $f(\boldsymbol{\theta}) := \langle I \otimes C \rangle_{\boldsymbol{\theta}}$ and $\boldsymbol{A}(\boldsymbol{\theta}) := \lambda \boldsymbol{\Phi}(\boldsymbol{\theta}) - \boldsymbol{b}$ have Lipschitz constants L_f and $L_{\boldsymbol{A}}$. Also suppose that the function $\mathcal{L}_c(\cdot, \boldsymbol{y})$ is smooth for every $\boldsymbol{y} \in \mathbb{R}^M$ (i.e., its gradient is $L_{c,\boldsymbol{y}}$-Lipschitz continuous). Additionally, suppose that there exists $\nu > 0$ such that*

$$\nu \|\boldsymbol{A}(\boldsymbol{\theta}^k)\| \leqslant \left\| -J_{\boldsymbol{A}}(\boldsymbol{\theta}^k)^\top \boldsymbol{A}(\boldsymbol{\theta}^k) \right\|, \tag{82}$$

for every $k \in K$. If an inexact solver is used in Step 5 of the algorithm, then $\boldsymbol{\theta}^k$ is a first order (ϵ_k, c_k)-stationary point of (75) with

$$\epsilon_k = \frac{1}{c_{k-1}} \left(\frac{2(L_f + y_{\max} L_{\boldsymbol{A}})(1 + \eta^k L_{\boldsymbol{A}})}{\nu} + 1 \right) =: \frac{Q(f, \boldsymbol{A}, \eta^1)}{c_{k-1}}, \tag{83}$$

for every $k \in K$, where $y_{\max}(\boldsymbol{\theta}^1, \boldsymbol{y}^0, \eta^1)$ is given as

$$y_{\max}(\boldsymbol{\theta}^1, \boldsymbol{y}^0, \eta^1) = \|\boldsymbol{y}^0\| + \eta^1 \|\boldsymbol{A}(\boldsymbol{\theta}^1)\| \ln^2 2. \tag{84}$$

Additionally, if we use the Accelerated Proximal Gradient Method (APGM) from [51] as an inexact solver for the subproblem (Step 5), then the algorithm finds a first-order (ϵ_T, c_T)-stationary point, after T calls to the solver, where

$$T = \mathcal{O}\left(\frac{rQ^3}{\epsilon_T^4} \log_\mu \left(\frac{Q}{\epsilon_T} \right) \right) = \tilde{\mathcal{O}}\left(\frac{rQ^3}{\epsilon_T^4} \right), \tag{85}$$

with $Q \equiv Q(f, \boldsymbol{A}, \eta^1)$, and μ is a constant as mentioned in Algorithm 2.

The inequality given by (82) is known as the PL inequality for minimizing $\|\boldsymbol{A}(\boldsymbol{\theta})\|^2$ [52]. This regularity condition is used in the proof of Theorem 4.1 of [50] for obtaining a bound on $\|\boldsymbol{A}(\boldsymbol{\theta}^k)\|$ for all $k \in K$. For our purposes, we assume that this condition holds for all $\boldsymbol{\theta} \in [0, 2\pi]^r$. Then, Theorem 15 is a direct consequence of the following statements:

(A) **Smoothness of $\mathcal{L}_c(\cdot, \boldsymbol{y})$**: The function $\mathcal{L}_c(\cdot, \boldsymbol{y})$ is smooth for every $\boldsymbol{y} \in \mathbb{R}^M$ (i.e., its gradient is $L_{c,\boldsymbol{y}}$-Lipschitz continuous).

(B) **Lipschitz continuity of $f(\boldsymbol{\theta})$ and $\boldsymbol{A}(\boldsymbol{\theta})$**: The functions $f(\boldsymbol{\theta})$ and $\boldsymbol{A}(\boldsymbol{\theta})$ are Lipschitz continuous with constants L_f and $L_{\boldsymbol{A}}$, respectively.

We prove the statements (A) and (B) in Lemmas 17 and 18, respectively. Before we prove these statements, we state the following lemma that concerns Lipschitz continuity of the function $h : [0, 2\pi]^r \to \mathbb{R}$ defined as

$$h(\boldsymbol{\theta}) := \langle I \otimes O \rangle_{\boldsymbol{\theta}} = \langle \mathbf{0} | U^\dagger(\boldsymbol{\theta})(I \otimes O)U(\boldsymbol{\theta}) | \mathbf{0} \rangle), \tag{86}$$

and its gradient $\nabla h(\boldsymbol{\theta})$, where $O \in \mathcal{S}^N$.

Lemma 16 (Lipschitz continuity of $h(\boldsymbol{\theta})$ and $\nabla h(\boldsymbol{\theta})$)**.** *The function $h : [0, 2\pi]^r \to \mathbb{R}$ and its gradient, i.e., the vector-valued function $\nabla h : [0, 2\pi]^r \to \mathbb{R}^r$, are L_h-Lipschitz and $L_{\nabla h}$-Lipschitz continuous, respectively, for some $L_h, L_{\nabla h} > 0$.*

Proof. The proof is given in Appendix A.3. ■

Next, we formally write the statements (A) and (B) as lemmas.

Lemma 17 (Smoothness of $\mathcal{L}_c(\cdot, \boldsymbol{y})$)**.** *For all $\boldsymbol{y} \in \mathbb{R}^M$, there exists $L_{c,\boldsymbol{y}} > 0$ such that the gradient of the function $\mathcal{L}_c(\cdot, \boldsymbol{y}) : [0, 2\pi]^r \to \mathbb{R}$ is $L_{c,\boldsymbol{y}}$-Lipschitz continuous.*

Proof. The proof is given in Appendix A.4. ■

Lemma 18 (Lipschitz continuity of $f(\boldsymbol{\theta})$ and $\boldsymbol{A}(\boldsymbol{\theta})$)**.** *The function $f : [0, 2\pi]^r \to \mathbb{R}$, where $f(\boldsymbol{\theta}) = \langle I \otimes C \rangle_{\boldsymbol{\theta}}$, and the linear map $\boldsymbol{A}(\boldsymbol{\theta}) : [0, 2\pi]^r \to \mathbb{R}^M$ are L_f-Lipschitz and $L_{\boldsymbol{A}}$-Lipschitz continuous, respectively, for some $L_f, L_{\boldsymbol{A}} > 0$.*

Proof. The proof directly follows from the Lipschitz continuity of $h(\boldsymbol{\theta})$ (Lemma 16). ■

3.3 Inequality Constrained Standard Form (ICSF) of SDPs

In this section, we consider the following inequality constrained primal form of SDPs:

$$\begin{aligned} p^* = \sup_{X \succcurlyeq 0} \quad & \mathrm{Tr}[CX] \\ \text{subject to} \quad & \boldsymbol{\Phi}(X) \preccurlyeq \boldsymbol{b}. \end{aligned} \tag{87}$$

Here we take vector forms of the Hermiticity-preserving linear map Φ and the Hermitian operator B because both have a diagonal form. Furthermore, we assume that the last constraint is the trace constraint on the primal PSD variable X. As such, we set $A_M = I$ and $\mathrm{Tr}[X] \leqslant b_M$. Additionally, for this case also, we make an assumption that the SDPs are weakly constrained, i.e., $N \gg M$. The corresponding dual form is given as follows:

$$\begin{aligned} d^* = \inf_{\boldsymbol{y} \succcurlyeq 0} \quad & \boldsymbol{b}^\top \boldsymbol{y} \\ \text{subject to} \quad & \Phi^\dagger(\boldsymbol{y}) \succcurlyeq C, \end{aligned} \tag{88}$$

where $\boldsymbol{y} = (y_1, \ldots, y_M)^\top$ is a dual variable. We can determine the dual map $\Phi^\dagger$ as follows, using the definition of the adjoint of a linear map (see Definition (8)):

$$\boldsymbol{y}^\top \boldsymbol{\Phi}(X) = \sum_{i=1}^{M-1} y_i \operatorname{Tr}[A_i X] + y_M \operatorname{Tr}[X] \tag{89}$$

$$= \operatorname{Tr}\left[\left(\sum_{i=1}^{M-1} y_i A_i + y_M I \right) X \right], \tag{90}$$

which implies that

$$\Phi^\dagger(\boldsymbol{y}) = \sum_{i=1}^{M-1} y_i A_i + y_M I. \tag{91}$$

We derive unconstrained formulations for the primal and dual forms of inequality constrained SDPs. First, by taking the primal form of SDPs into account, we introduce slack variables and convert the inequality constrained problem (87) to the following equality constrained problem:

$$\begin{aligned} p^* = \sup_{X \succcurlyeq 0, \boldsymbol{z} \geqslant 0} \quad & \operatorname{Tr}[CX] \\ \text{subject to} \quad & \boldsymbol{b} - \boldsymbol{\Phi}(X) = \boldsymbol{z}, \end{aligned} \tag{92}$$

where $\boldsymbol{z} = (z_1, \ldots, z_M)^\top$ is a vector of slack variables and $z_1, \ldots, z_M \geqslant 0$. Now, with the problem (92) being an equality constrained problem, we can use iVQAEC (see Algorithm 2) to solve it.

Next, we focus on the dual form of inequality constrained SDPs as follows:

$$d^* = \inf_{\boldsymbol{y} \geqslant 0} \left\{ \boldsymbol{b}^\top \boldsymbol{y} : \Phi^\dagger(\boldsymbol{y}) \succcurlyeq C \right\} \tag{93}$$

$$= \inf_{y_1, \ldots, y_M \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + y_M b_M : \sum_{i=1}^{M-1} y_i A_i + y_M I \succcurlyeq C \right\} \tag{94}$$

$$= \inf_{y_1, \ldots, y_M \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + y_M b_M : y_M I \succcurlyeq C - \sum_{i=1}^{M-1} y_i A_i \right\}. \tag{95}$$

Now, consider that

$$\inf_{t \geqslant 0} \{ t : tI \succcurlyeq H \} = \max\{0, \lambda_{\max}(H)\} = \max \left\{ 0, \sup_{\rho \in \mathcal{D}^N} \operatorname{Tr}[H\rho] \right\}, \tag{96}$$

where H is a Hermitian operator, $\lambda_{\max}(H)$ is the maximum eigenvalue of H, and the supremum is over the set $\mathcal{D}^N$ of density operators. Using this fact, we convert the constrained problem (95) into the following unconstrained problem:

$$d^* = \inf_{y_1, \ldots, y_{M-1} \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + b_M \cdot \max \left\{ 0, \sup_{\rho \in \mathcal{D}^N} \operatorname{Tr} \left[\left(C - \sum_{i=1}^{M-1} y_i A_i \right) \rho \right] \right\} \right\} \tag{97}$$

$$= \inf_{\bar{\boldsymbol{y}} \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + b_M \cdot \max \left\{ 0, \sup_{\rho \in \mathcal{D}^N} \operatorname{Tr} \left[H(\bar{\boldsymbol{y}}) \rho \right] \right\} \right\}, \tag{98}$$

where we set

$$\bar{\boldsymbol{y}} := (y_1, \ldots, y_{M-1})^\top, \tag{99}$$

$$H(\bar{\boldsymbol{y}}) \coloneqq C - \sum_{i=1}^{M-1} y_i A_i. \tag{100}$$

Furthermore, as the first term is independent of ρ, we have that

$$d^* = \inf_{\bar{\boldsymbol{y}} \geqslant 0} \sup_{\rho \in \mathcal{D}^N} \left\{ \sum_{i=1}^{M} b_i y_i + b_M \cdot \max\left\{0, \operatorname{Tr}\left[H(\bar{\boldsymbol{y}})\rho\right]\right\} \right\}. \tag{101}$$

We can use Sion's minimax theorem [53] and interchange the supremum and infimum because the set $\mathcal{D}^N$ of density operators is compact and convex, the objective function is convex with respect to $\bar{\boldsymbol{y}}$ for all $\rho \in \mathcal{D}^N$, and it is quasi-concave with respect to $\rho \in \mathcal{D}^N$ for all $\bar{\boldsymbol{y}} \geqslant 0$. Hence,

$$d^* = \sup_{\rho \in \mathcal{D}^N} \inf_{\bar{\boldsymbol{y}} \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + b_M \cdot \max\left\{0, \operatorname{Tr}\left[H(\bar{\boldsymbol{y}})\rho\right]\right\} \right\}. \tag{102}$$

The above problem is not differentiable at some parameter values due to the max operation in the objective function. In order to smoothen the sharp corners that result from this operation, we modify the above problem in the following way by introducing $\gamma > 0$:

$$d' \coloneqq \sup_{\rho \in \mathcal{D}^N} \inf_{\bar{\boldsymbol{y}} \geqslant 0} \left\{ \sum_{i=1}^{M-1} b_i y_i + \frac{b_M}{\gamma} \ln\left(e^{\gamma \operatorname{Tr}[H(\bar{\boldsymbol{y}})\rho]} + 1\right) \right\}. \tag{103}$$

In the above, we used the scaled version of the log-sum-exp function [28, Section 3.1.5] as an approximation to the original max function in (102). This approximation can be controlled by varying the parameter γ because the following holds for all $x, y \in \mathbb{R}$:

$$\max\{x, y\} \leqslant \frac{\ln(e^{\gamma x} + e^{\gamma y})}{\gamma} \leqslant \max\{x, y\} + \frac{\ln(2)}{\gamma}. \tag{104}$$

By using this approximation, the objective function in (103) does not have any sudden corners where the partial derivatives can change drastically. In fact, the objective function of (103) is infinitely differentiable. It is also equivalent to the previous objective function in (102) in the limit $\gamma \to \infty$. For all finite values of γ, the value d^* is bounded from above by d'.

3.3.1 Variational Quantum Algorithm for SDPs in ICSF

Solving the unconstrained optimization problem (103) using a gradient based classical technique is computationally expensive if the dimension N of the operators is large. Hence, in this section, we propose a variational quantum algorithm to solve this problem.

First, we introduce a parameterization of density operators using parameterized quantum circuits, in order to efficiently estimate the expectation values of the Hermitian operators $C, A_1, \ldots, A_{M-1}$. Second, we optimize the resulting objective function using our variational quantum algorithm.

Parameterization: Let $U_{RS}(\boldsymbol{\theta})$ be a parameterized quantum circuit with a parameter $\boldsymbol{\theta} \in [0, 2\pi]^r$, and let it act on the all-zeros state of the quantum system RS and prepare a purification of the density operator $\rho_S(\boldsymbol{\theta})$. The structure of $U_{RS}(\boldsymbol{\theta})$ is given by (24). Furthermore, the following equality holds because $U_{RS}(\boldsymbol{\theta})$ generates a purification of $\rho_S(\boldsymbol{\theta})$:

$$\operatorname{Tr}[H\rho_S(\boldsymbol{\theta})] = \langle \mathbf{0}|_{RS} U_{RS}^\dagger(\boldsymbol{\theta}) \left(I_R \otimes H_S\right) U_{RS}(\boldsymbol{\theta}) |\mathbf{0}\rangle_{RS} = \langle I \otimes H \rangle_{\boldsymbol{\theta}}, \tag{105}$$

Algorithm 3 iVQAIC$(C, \{A_i\}_{i=1}^M, \{b_i\}_{i=1}^M, \eta, \epsilon, \gamma)$

Input: Hermitian operators $(C, \{A_i\}_{i=1}^M)$, scalars $\{b_i\}_{i=1}^M$, learning rate $\eta > 0$, precision $\epsilon > 0$, constant $\gamma > 0$.

Initialization: $\bar{\boldsymbol{y}}^1 = \mathbf{0}$.

 # For any step, expectation values of observables and their gradients are evaluated using a parameterized quantum circuit.

for $k = 1, 2, \ldots,$ **do**

  Maximize $\mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}}^k)$, where $\bar{\boldsymbol{y}}^k$ is fixed, to obtain $\boldsymbol{\theta}^{k+1}$ such that the following holds:

$$\left\| \nabla \mathcal{F}_\gamma(\boldsymbol{\theta}^{k+1}, \bar{\boldsymbol{y}}) \right\| \leqslant \epsilon.$$

  $\bar{\boldsymbol{y}}^{k+1} = \bar{\boldsymbol{y}}^k - \eta \nabla_{\bar{\boldsymbol{y}}} \mathcal{F}_\gamma(\boldsymbol{\theta}^{k+1}, \bar{\boldsymbol{y}}^k)$

  if $\left\| \nabla \mathcal{F}_\gamma(\boldsymbol{\theta}^{k+1}, \bar{\boldsymbol{y}}^{k+1}) \right\| \leqslant \epsilon$ **then**

    STOP and return $\mathcal{F}_\gamma(\boldsymbol{\theta}^{k+1}, \bar{\boldsymbol{y}}^{k+1})$

where $H \in \{C, A_1, \ldots, A_{M-1}\}$.

Due to the parameterization of the density operator $\rho_S(\boldsymbol{\theta})$, the problem (103) transforms into the following optimization problem:

$$d' := \sup_{\boldsymbol{\theta} \in [0, 2\pi]^r} \inf_{\bar{\boldsymbol{y}} \geqslant 0} \mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}}), \tag{106}$$

where

$$\mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}}) := \sum_{i=1}^{M-1} b_i y_i + \frac{b_M}{\gamma} \ln\left(e^{\gamma \langle I \otimes H(\bar{\boldsymbol{y}}) \rangle_{\boldsymbol{\theta}}} + 1 \right) \tag{107}$$

and we optimize over the space of quantum circuit parameters $\boldsymbol{\theta} \in [0, 2\pi]^r$. As mentioned before, we assume that the objective function $\mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}})$ is faithful, which means that the global optimal value of the optimization problem (106) is equal to d'.

The objective function of (106) is in general non-convex with respect to the quantum circuit parameter $\boldsymbol{\theta}$. On the contrary, it is concave in $\bar{\boldsymbol{y}}$. Hence, the optimization problem (106) is a non-convex–concave optimization problem. For such a setting, finding a globally optimal solution is generally NP-hard [33, Section 2.1]. Therefore, we focus on obtaining approximate stationary points of (106). As a global optimal point is also a stationary point, if we use techniques to initialize quantum circuit parameters according to the problem at hand such that these parameters lie in the vicinity of a global optimal point, then our algorithm will converge to that point.

Definition 19 (First order ϵ-stationary points of (106))**.** *For $\epsilon > 0$, a point $(\boldsymbol{\theta}, \bar{\boldsymbol{y}}) \in [0, 2\pi]^r \times \mathbb{R}^{M-1}$ is a first order ϵ-stationary point of (106) if the following holds:*

$$\|\nabla \mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}})\| \leqslant \epsilon. \tag{108}$$

We use the above condition as a stopping criterion for our variational quantum algorithm.

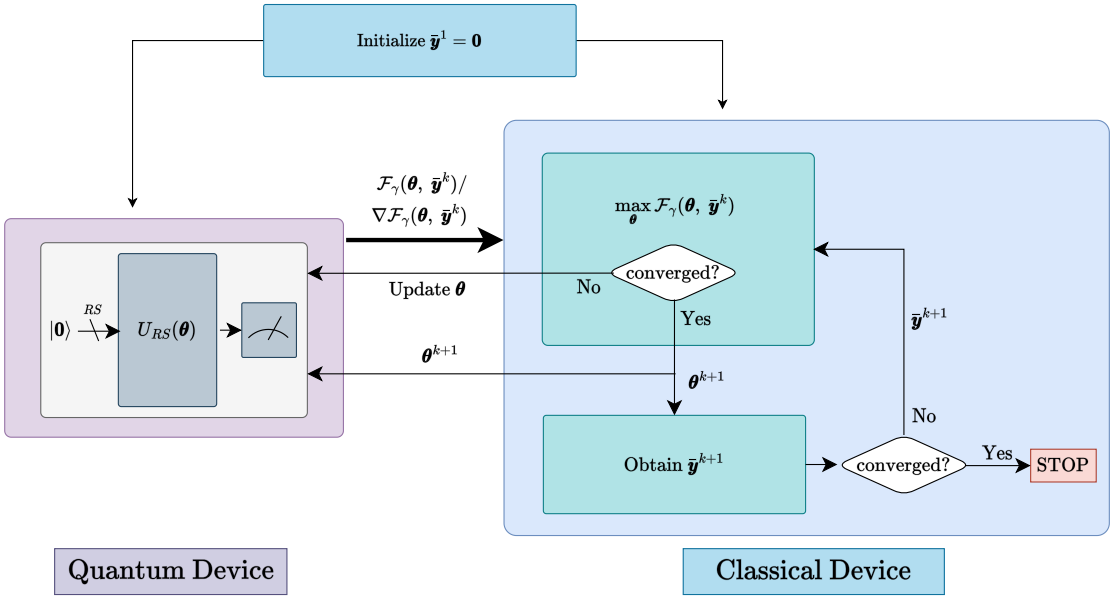


Figure 3: This figure depicts the iVQAIC algorithm in which we utilize one parameterized quantum circuit, i.e., $U_{RS}(\boldsymbol{\theta})$.

In order to obtain a first order ϵ-stationary point of (106), we propose a variational quantum algorithm (see Algorithm 3). We call this algorithm *inexact Variational Quantum Algorithm for Inequality Constrained standard form* (iVQAIC). It is an inexact version because we solve the subproblem involving the maximization to an approximate stationary point instead of solving it until global optimality is reached, due to the non-convex nature of the objective function in terms of quantum circuit parameters. Its pseudocode is provided in Algorithm 3, and it is depicted in Figure 3.

We run iVQAIC on a classical computer and utilize a parameterized quantum circuit $U_{RS}(\boldsymbol{\theta})$ with a parameter $\boldsymbol{\theta} \in [0, 2\pi]^r$ for estimating the expectation values of the Hermitian operators. Moreover, the partial derivative of $\mathcal{F}_\gamma(\boldsymbol{\theta}, \bar{\boldsymbol{y}})$ with respect to the quantum circuit parameters $\boldsymbol{\theta}$ depends on the partial derivatives of the expectation values of the Hermitian operators $I \otimes C$, $I \otimes A_1$, $\ldots$, $I \otimes A_{M-1}$ with respect to $\boldsymbol{\theta}$. In order to compute the partial derivatives of the latter with respect to $\boldsymbol{\theta}$, we use the parameter-shift rule (see (27)). Note that we do not explicitly mention these quantum-circuit calls in the pseudocode of the algorithm.

For this case, we do not state a theorem that indicates the convergence rate for finding approximate stationary points of (106) in terms of the number of iterations in the for-loop of the algorithm. Rather, we prove a property of the objective function that is necessary for providing such a convergence analysis, i.e., smoothness of $\mathcal{F}(\cdot, \bar{\boldsymbol{y}})$ for a fixed $\bar{\boldsymbol{y}} \geqslant 0$. We state it formally in the following lemma:

Lemma 20 (Smoothness of $\mathcal{F}_\gamma(\cdot, \bar{\boldsymbol{y}})$)**.** *For all $\bar{\boldsymbol{y}} \geqslant 0$, there exists $L_{\gamma,\bar{\boldsymbol{y}}} > 0$ such that the gradient of the function $\mathcal{F}_\gamma(\cdot, \bar{\boldsymbol{y}}) : [0, 2\pi]^r \to \mathbb{R}$ is $L_{\gamma,\bar{\boldsymbol{y}}}$-Lipschitz continuous.*

Proof. The proof is given in Appendix A.5. ■

4 Numerical Simulations

For validating our reformulations of SDPs and verifying the convergence of their respective algorithms to approximate stationary points, we randomly generated SDPs such that they

contain a valid feasible region. To verify the convergence of the algorithms proposed in this paper, we focus on three cases based on the number M of constraints and the dimension N of the input Hermitian operators, and whether the problem is an equality or inequality constrained problem:

1. $N \approx M$: Here, we consider a well-known and extensively studied MaxCut problem. First, we briefly recall what the problem is and its SDP relaxation in the next subsection. This problem is taken into account because the number of constraints and the dimension of the matrices are equal ($N = M$). For this case, we evaluated the performance of iVQAGF (see Algorithm 1), as it does not make the weakly-constrained assumption on SDPs. Figure 4 shows the convergence of iVQAGF for solving randomly generated SDP instances of the MaxCut problem. Furthermore, we performed these simulations for different dimensions of the input matrices. Specifically, we considered $N \in \{8, 16, 32\}$.

2. $N \gg M$: We divide this case further according to the type of constraints: (2a) equality constraints and (2b) inequality constraints. For randomly generated equality-constrained problems, we report the performance of iVQAEC (see Algorithm 2). Similarly, we analyse the performace of iVQAIC (see Algorithm 3) for randomly generated instances of an inequality constrained problem. Here also we consider $N \in \{8, 16, 32\}$. Figure 5 and Figure 6 showcase the convergence of iVQAEC and iVQAIC, respectively, for randomly generated instances of their respective problems.

We take into account Assumption 3 while creating the input Hermitian matrices. We assume that the Pauli string decomposition of these input matrices are provided beforehand. Additionally, for the analysis of iVQAGF for solving MaxCut, we assume that we are provided with the Pauli string decomposition of $C^\top$ and the Choi operator of the linear map Φ, i.e., Γ^Φ.

We executed our algorithms using Pennylane's Python libraries where we set QASM simulator of the Qiskit Python package as a backend.* Pennylane is an open-source Python library developed by Xanadu for differential programming of quantum computers [54]. Similarly, Qiskit is an open-source package/interface developed by IBM to interact with the underlying quantum computer [27]. Additionally, the QASM simulator simulates a real IBM Quantum Backend, which is actually noisy in nature due to gate errors and decoherence. We integrated Pennylane and Qiskit's QASM simulator using the Pennylane-Qiskit plugin.

We use the Strong Entangling Layers template of Pennylane as our variational ansatz, where each layer consists of $O(\text{poly}(n))$ single-qubit rotations and entangling gates. We then repeat this layer $O(\text{poly}(n))$ number of times, where n is the number of qubits. Therefore, the overall gate complexity is $O(\text{poly}(n))$.

In order to assess the convergence of our methods to a globally optimal point, we initialize the quantum circuit parameter such that it lies in the convex region of that globally optimal point. We then report the results and their associated analyses on how well our algorithms perform on the QASM noisy simulator and compare the results with a noiseless simulator. We also report the time complexity (number of iterations of the for-loops) of our algorithms, i.e., how fast our algorithms converge to an actual globally optimal point.

First, for the sake of completeness, we recall the definition of a cut and a MaxCut of a given graph.

*The source code is available here.

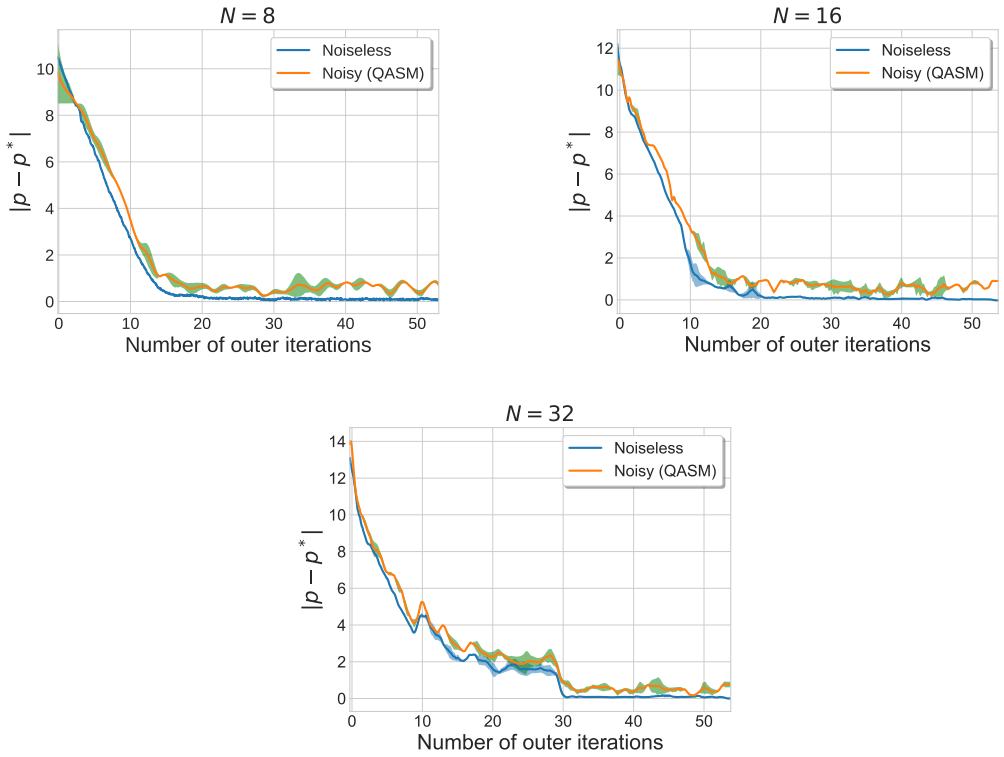


Figure 4: Convergence of iVQAGF for three randomly generated MaxCut-SDP instances with different numbers of vertices in the graph: $N \in \{8, 16, 32\}$.

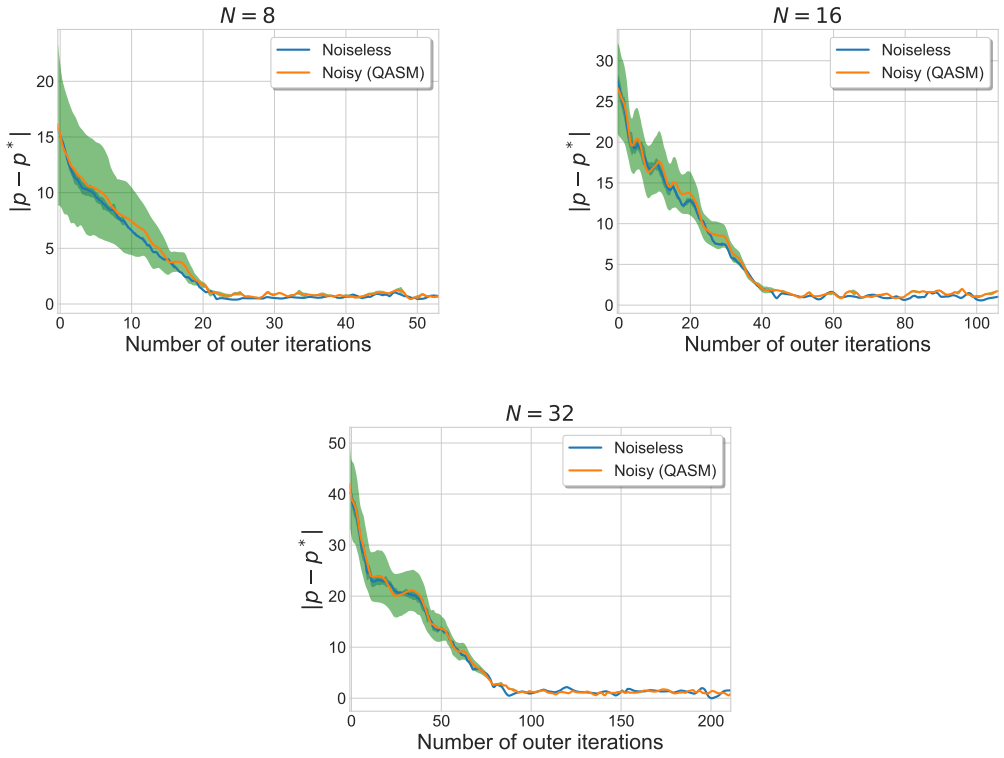


Figure 5: Convergence of iVQAEC for three separate cases of randomly generated equality constrained semidefinite programs and nonempty feasbile regions: $N \in \{8, 16, 32\}$.

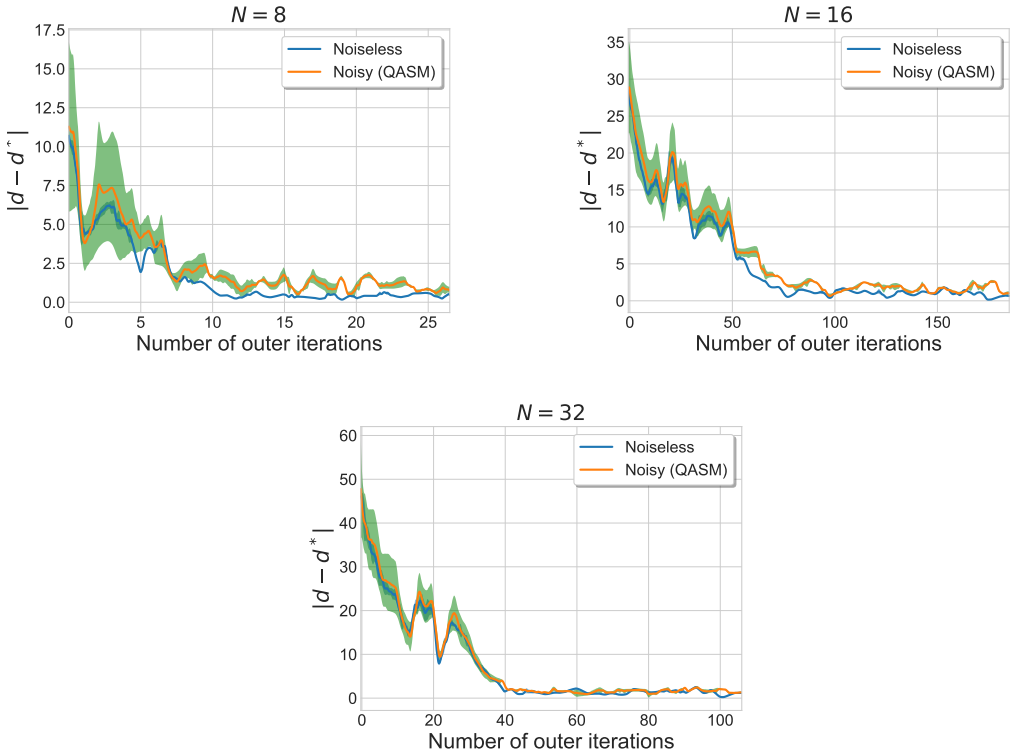


Figure 6: Convergence of iVQAIC for three separate cases of randomly generated inequality constrained semidefinite programs with nonempty feasbile regions: $N \in \{8, 16, 32\}$.

Definition 21 (Cut and MaxCut)**.** *A cut is a bi-partition W of the vertex set V of a graph $G = (V, E)$, where $|V| = N$. An edge $(i, j) \in E$ is part of the cut set if its vertices i and j lie in separate partitions. A MaxCut is the largest cut possible of a graph G.*

The problem of finding a MaxCut of a graph can be formulated as a Quadratic Integer Program (QIP) [55]:

$$\begin{aligned} [\text{QIP}]: \quad & \sup \sum_{(i,j)\in E} \frac{1}{4}(x_i - x_j)^2 \\ & \text{subject to} \;\; x_i \in \{-1, 1\}; \;\; \forall i \in V. \end{aligned} \tag{109}$$

Here, the optimal value of the above optimization is the optimal cut size (i.e., the number of the edges in the MaxCut). This quadratic integer program is in general computationally intractable [56]. However, there exist LP and SDP relaxations for the above program [57].

$$\begin{aligned} [\text{LP}]: \quad & \sup \sum_{(i,j)\in E} \frac{1}{4}\|\boldsymbol{v}_i - \boldsymbol{v}_j\|^2 \\ & \text{subject to} \;\; \|\boldsymbol{v}_i\|^2 = 1, \boldsymbol{v}_i \in \mathbb{R}^N; \;\; \forall i \in V. \end{aligned} \tag{110}$$

The above mentioned LP formulation is not good enough as it is a $1/2$-approximation to the original problem. It is well known that this can be extended to an SDP formulation in which an algorithm proposed in [57] gives a 0.879-approximation of the original QIP

in (109):

$$\begin{aligned} [\text{SDP}]: \quad & \sup \sum_{(i,j)\in E} \frac{1}{4}(X_{ii} - 2X_{ij} + X_{jj}) \\ & \text{subject to} \;\; X \succcurlyeq 0, \;\; X_{ii} = 1, \;\; \forall i \in V. \end{aligned} \tag{111}$$

For our case, we numerically simulate iVQAGF for solving the aforementioned SDP relaxation of a MaxCut problem. Recasting the constraints of the above MaxCut SDP as constraints of the general form of SDP, we can write them as $\Phi(X) = B$, where

$$B = I, \qquad \Phi(X) = \text{diag}\left(\text{Tr}[A_1 X], \ldots, \text{Tr}[A_M X]\right), \tag{112}$$

and $A_i = |i\rangle\langle i|$. Therefore, the Choi operator for this linear map is given as

$$\Gamma^{\Phi} = \sum_{i,j=0}^{N-1} |i\rangle\langle j| \otimes \Phi(|i\rangle\langle j|) = \sum_{i=0}^{N-1} |i\rangle\langle i| \otimes |i\rangle\langle i|. \tag{113}$$

Additionally, the objective function can be written as $\text{Tr}[CX]$, where $C = L/4$ and L is the Laplacian matrix for the graph G.

The Pauli decomposition of the above Choi operator consists of $N = 2^n$ Pauli strings. This gives the impression that we need to compute 2^n expectation values in order to compute the expectation value of Γ^{Φ}. However, all these Pauli strings can be constructed from just I and σ_z operators, because it is clearly a diagonal matrix. This implies that all the Pauli strings are mutually commuting, and their expectation values can be estimated simultaneously [58]. Therefore, we just need to compute a single expectation value to evaluate the expectation value of Γ^{Φ}.

The shaded regions in Figure 4, 5, and 6 signify the variance in the values for different runs (specifically 20) of the algorithm. The y-axis measures the difference between the cost function value at each iteration and the actual optimal value evaluated with the CVXPY package [59]. The x-axis shows the number of outer iterations of algorithms, i.e., the time needed to converge to the actual optimal value.

The numerical simulations demonstrate that all three algorithms indeed converge to their respective optimal values approximately. This numerical evidence suggests that all three proposed algorithms work well in practice. In other words, convergence to the optimal parameters in the presence of noise showcases noise resilience of our variational quantum algorithms.

5 Conclusion

In this paper, we proposed variational quantum algorithms for solving semidefinite programs. We considered three constrained formulations of SDPs, which were first converted to unconstrained forms by employing a series of reductions. When the dimension N of the input Hermitian operators of SDPs is large and for these unconstrained forms, the computation of the objective function's gradient is difficult when using known classical techniques. To address this problem, we utilized parameterized quantum circuits to estimate these gradients. We also established the convergence rate and total iteration complexity of one of our proposed VQAs. Finally, we numerically simulated our variational quantum algorithms for different instances of SDPs, and the results of these simulations provide evidence that convergence still occurs in noisy settings.

The estimation of the gradients using parameterized quantum circuits is stochastic in nature. In this paper, we assumed that we have unbiased estimators of these gradients,

and the variance of these estimators is also small. Therefore, it remains open to study the effect of the variance of these estimators on the convergence rate of our algorithms.

Acknowledgments

DP and MMW acknowledge support from the National Science Foundation under Grant No. 1907615. PJC acknowledges initial support from the Los Alamos National Laboratory (LANL) ASC Beyond Moore's Law project, and later support from the U.S. Department of Energy (DOE), Office of Science, Office of Advanced Scientific Computing Research, under the Accelerated Research in Quantum Computing (ARQC) program. DP acknowledges Prof. Rahul Shah's helpful suggestions and support from the National Science Foundation under Grant No. 2137057.

Author Contributions

The following describes the different contributions of all authors of this work, using roles defined by the CRediT (Contributor Roles Taxonomy) project [60]:

DP: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization.

PC: Writing - Review & Editing.

MMW: Conceptualization, Methodology, Formal Analysis, Writing - Review & Editing, Supervision, Project administration, Funding acquisition.

References

[1] László Lovász and Alexander Schrijver. "Cones of matrices and set-functions and 0-1 optimization". Society for Industrial and Applied Mathematics Journal on Optimization **1**, 166–190 (2006).

[2] Anirudha Majumdar, Georgina Hall, and Amir Ali Ahmadi. "Recent scalability improvements for semidefinite programming with applications in machine learning, control, and robotics". Annual Review of Control, Robotics, and Autonomous Systems **3**, 331–360 (2020).

[3] Pablo A. Parrilo. "Semidefinite programming relaxations for semialgebraic problems". Mathematical Programming **96**, 293–320 (2003).

[4] Horace Yuen, Robert Kennedy, and Melvin Lax. "Optimum testing of multiple hypotheses in quantum detection theory". IEEE Transactions on Information Theory **21**, 125–134 (1975).

[5] Yonina C. Eldar. "A semidefinite programming approach to optimal unambiguous discrimination of quantum states". IEEE Transactions on Information Theory **49**, 446–456 (2003).

[6] Xin Wang, Wei Xie, and Runyao Duan. "Semidefinite programming strong converse bounds for classical capacity". IEEE Transactions on Information Theory **64**, 640–653 (2018).

[7] Xin Wang. "Semidefinite optimization for quantum information". PhD thesis. University of Technology Sydney. Centre for Quantum Software and Information, Faculty of Engineering and Information Technology (2018). url: http://hdl.handle.net/10453/127996.

[8] Ivan Supic and Joseph Bowles. "Self-testing of quantum systems: a review". Quantum **4**, 337 (2020).

[9] Florian A. Potra and Stephen J. Wright. "Interior-point methods". Journal of Computational and Applied Mathematics **124**, 281–302 (2000).

[10] Peter W. Shor. "Algorithms for quantum computation: Discrete logarithms and factoring". In 35th Annual Symposium on Foundations of Computer Science. Pages 124–134. Santa Fe, New Mexico, USA (1994). IEEE Computer Society.

[11] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum algorithm for linear systems of equations". Physical Review Letters **103**, 150502 (2009).

[12] Lov K. Grover. "Quantum mechanics helps in searching for a needle in a haystack". Physical Review Letters **79**, 325–328 (1997).

[13] Christoph Dürr and Peter Høyer. "A quantum algorithm for finding the minimum" (1996). arXiv:quant-ph/9607014.

[14] Ramesh Hariharan and Vishwanathan Vinay. "String matching in $O(n+m)$ quantum time". Journal of Discrete Algorithms **1**, 103–110 (2003).

[15] Fernando G. S. L. Brandão and Krysta M. Svore. "Quantum speed-ups for solving semidefinite programs". In 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS). Pages 415–426. IEEE Computer Society (2017).

[16] Sanjeev Arora and Satyen Kale. "A combinatorial, primal-dual approach to semidefinite programs". Journal of the ACM **63**, 1–35 (2016).

[17] 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 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece. Volume 132 of LIPIcs, pages 27:1–27:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2019).

[18] Joran van Apeldoorn, András Gilyén, Sander Gribling, and Ronald de Wolf. "Quantum SDP-solvers: Better upper and lower bounds". Quantum **4**, 230 (2020).

[19] John Preskill. "Quantum computing in the NISQ era and beyond". Quantum **2**, 79 (2018).

[20] Frank Arute, Kunal Arya, Ryan Babbush, et al. "Quantum supremacy using a programmable superconducting processor". Nature **574**, 505–510 (2019).

[21] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon Benjamin, Suguro Endo, Keisuke Fujii, Jarrod Ryan McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick Coles. "Variational quantum algorithms". Nature Reviews Physics **3**, 625–644 (2021).

[22] Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S. Kottmann, Tim Menke, Wai-Keong Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. "Noisy intermediate-scale quantum algorithms". Reviews of Modern Physics **94**, 015004 (2022). arXiv:2101.08448.

[23] Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. "The complexity of NISQ". Nature Communications **14**, 6001 (2023).

[24] Armando Angrisani. "The disparate impact of noise on quantum learning algorithms". Theses. Sorbonne Université. (2023). url: https://theses.hal.science/tel-04511706.

[25] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. "A variational eigenvalue solver on a photonic quantum processor". Nature Communications **5**, 4213 (2014).

[26] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm" (2014). arXiv:1411.4028.

[27] Matthew Treinish et al. "Qiskit: An open-source framework for quantum computing" (2023). https://doi.org/10.5281/zenodo.2573505.

[28] Stephen P. Boyd and Lieven Vandenberghe. "Convex optimization". Cambridge University Press. (2004).

[29] Jun Li, Xiaodong Yang, Xinhua Peng, and Chang-Pu Sun. "Hybrid quantum-classical approach to quantum optimal control". Physical Review Letters **118**, 150503 (2017).

[30] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. "Quantum circuit learning". Physical Review A **98**, 032309 (2018).

[31] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. "Evaluating analytic gradients on quantum hardware". Physical Review A **99**, 032331 (2019).

[32] Patrick Huembeli and Alexandre Dauphin. "Characterizing the loss landscape of variational quantum circuits". Quantum Science and Technology **6**, 025011 (2021).

[33] Marina Danilova, Pavel Dvurechensky, Alexander Gasnikov, Eduard Gorbunov, Sergey Guminov, Dmitry Kamzolov, and Innokentiy Shibaev. "Recent theoretical advances in non-convex optimization". Volume 191, pages 79–163. Springer International Publishing. (2022). arXiv:2012.06188.

[34] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan. "How to escape saddle points efficiently". In Doina Precup and Yee Whye Teh, editors, Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. Volume 70 of Proceedings of Machine Learning Research, pages 1724–1732. PMLR (2017). url: http://proceedings.mlr.press/v70/jin17a.html.

[35] Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. "Training GANs with optimism". In International Conference on Learning Representations. (2018). arXiv:1711.00141.

[36] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. "GANs trained by a two time-scale update rule converge to a local Nash equilibrium". In Proceedings of the 31st International Conference on Neural Information Processing Systems. Page 6629–6640. NIPS'17Red Hook, NY, USA (2017). Curran Associates Inc. arXiv:1706.08500.

[37] Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. "Optimistic mirror descent in saddle-point problems: Going the extra(-gradient) mile". In International Conference on Learning Representations. (2019). arXiv:1807.02629.

[38] Eric V. Mazumdar, Michael I. Jordan, and S. Shankar Sastry. "On finding local Nash equilibria (and only local Nash equilibria) in zero-sum games" (2019). arXiv:1901.00838.

[39] Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. "Weakly-convex–concave min–max optimization: provable algorithms and applications in machine learning". Optimization Methods and Software **37**, 1087–1121 (2022). arXiv:1810.02060.

[40] Kishor Bharti, Tobias Haug, Vlatko Vedral, and Leong-Chuan Kwek. "Noisy intermediate-scale quantum algorithm for semidefinite programming". Physical Review A **105**, 052445 (2022). arXiv:2106.03891.

[41] Jakub Marecek and Albert Akhriev. "A cutting-plane method for semidefinite programming with potential applications on noisy quantum devices" (2021). arXiv:2110.03400.

[42] John Watrous. "The theory of quantum information". Cambridge University Press. (2018).

[43] Sumeet Khatri and Mark M. Wilde. "Principles of quantum communication theory: A modern approach" (2020). arXiv:2011.04672v2.

[44] John A. Nelder and Roger Mead. "A simplex method for function minimization". The Computer Journal **7**, 308–313 (1965).

[45] James C. Spall. "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation". IEEE Transactions on Automatic Control **37**, 332–341 (1992).

[46] James Kennedy and Russell Eberhart. "Particle swarm optimization". In Proceedings of ICNN'95-international conference on neural networks. Volume 4, pages 1942–1948. IEEE (1995).

[47] Magnus R. Hestenes. "Multiplier and gradient methods". Journal of Optimization Theory and Applications **4**, 303–320 (1969).

[48] Michael J. D. Powell. "A method for nonlinear constraints in minimization problems". Optimization, Pages 283–298 (1969). url: https://cir.nii.ac.jp/crid/1574231876073595264.

[49] Dimitri P. Bertsekas. "Multiplier methods: A survey". Automatica **12**, 133–145 (1976).

[50] Mehmet Fatih Sahin, Armin Eftekhari, Ahmet Alacaoglu, Fabian Latorre Gómez, and Volkan Cevher. "An inexact augmented Lagrangian framework for nonconvex optimization with nonlinear constraints". In Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. Pages 13943–13955. (2019). url: https://proceedings.neurips.cc/paper/2019/hash/866c7ee013c58f01fa153a8d32c9ed57-Abstract.html.

[51] Huan Li and Zhouchen Lin. "Accelerated proximal gradient methods for nonconvex programming". In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems. Volume 28. Curran Associates, Inc. (2015). url: https://proceedings.neurips.cc/paper_files/paper/2015/file/f7664060cc52bc6f3d620bcedc94a4b6-Paper.pdf.

[52] Hamed Karimi, Julie Nutini, and Mark Schmidt. "Linear convergence of gradient and proximal-gradient methods under the Polyak–Lojasiewicz condition". In Machine Learning and Knowledge Discovery in Databases. Pages 795–811. Springer, Cham, Switzerland (2016).

[53] Maurice Sion. "On general minimax theorems". Pacific Journal of Mathematics **8**, 171–176 (1958).

[54] Ville Bergholm, Josh Izaac, Maria Schuld, Christian Gogolin, et al. "Pennylane: Automatic differentiation of hybrid quantum-classical computations" (2022). arXiv:1811.04968.

[55] Bojan Mohar and Svatopluk Poljak. "Eigenvalues and the max-cut problem". Czechoslovak Mathematical Journal **40**, 343–352 (1990). url: http://eudml.org/doc/13856.

[56] Christos H. Papadimitriou and Mihalis Yannakakis. "Optimization, approximation, and complexity classes". Journal of Computer and System Sciences **43**, 425–440 (1991).

[57] Michel X. Goemans and David P. Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming". Journal of the ACM **42**, 1115–1145 (1995).

[58] Ophelia Crawford, Barnaby van Straaten, Daochen Wang, Thomas Parks, Earl Campbell, and Stephen Brierley. "Efficient quantum measurement of Pauli operators in the presence of finite sampling error". Quantum **5**, 385 (2021).

[59] Steven Diamond and Stephen Boyd. "CVXPY: A Python-embedded modeling language for convex optimization". Journal of Machine Learning Research **17**, 1–5 (2016). url: https://www.jmlr.org/papers/volume17/15-408/15-408.pdf.

[60] NISO. "Credit – contributor roles taxonomy". https://credit.niso.org/, Accessed 2024-06-07.

A Appendix

A.1 Proof of Lemma 7

The proof is rather straightforward. First, let us consider a function with two parameters x and y, so that $f : \mathbb{R}^2 \to \mathbb{R}$. Suppose that the function $(\cdot) \to f(\cdot, y)$ is Lipschitz continuous with Lipschitz constant L_X, for all $y \in \mathbb{R}$, and suppose that the function $(\cdot) \to f(x, \cdot)$ is Lipschitz continuous with Lipschitz constant L_Y, for all $x \in \mathbb{R}$. Therefore, the following holds according to the definition of Lipschitz continuity (recall Definition 6):

$$\left|f(x,y) - f(x',y)\right| \leqslant L_X \left|x - x'\right| \quad \forall x, x', y \in \mathbb{R}, \tag{114}$$
$$\left|f(x,y) - f(x,y')\right| \leqslant L_Y \left|y - y'\right| \quad \forall x, y, y' \in \mathbb{R}. \tag{115}$$

Now consider the following:

$$\begin{aligned}
\left|f(x,y) - f(x',y')\right| &= \left|f(x,y) - f(x',y) + f(x',y) - f(x',y')\right| && (116)\\
&\overset{(a)}{\leqslant} \left|f(x,y) - f(x',y)\right| + \left|f(x',y) - f(x',y')\right| && (117)\\
&\overset{(b)}{\leqslant} L_X \left|x - x'\right| + L_Y \left|y - y'\right| && (118)\\
&\leqslant \max\{L_X, L_Y\} \left(\left|x - x'\right| + \left|y - y'\right|\right) && (119)
\end{aligned}$$

$$\overset{(c)}{\leqslant} \sqrt{2}\max\{L_X, L_Y\}\left\|(x,y)-(x',y')\right\|, \tag{120}$$

where the inequality (a) follows from the triangle inequality, inequality (b) follows from (114)–(115), and the last inequality (c) follows from the fact that $\|\cdot\|_1 \leqslant \sqrt{2}\,\|\cdot\|$ for the two-variable case. Therefore, $L = \sqrt{2}\max\{L_X, L_Y\}$ is a Lipschitz constant for f.

The proof given above for two variables can be easily extended to a function f with an n-variable input, using the fact that $\|\cdot\|_1 \leqslant \sqrt{n}\,\|\cdot\|$, where $L = \sqrt{n}\max_i\{L_i\}_i$ is a Lipschitz constant given in terms of

$$L_i = \sup_{\boldsymbol{x}} \left|\frac{\partial f(\boldsymbol{x})}{\partial x_i}\right|. \tag{121}$$

Here, $\boldsymbol{x} = (x_1, \ldots, x_n)^\top$.

A.2 Proof of Lemma 8

By hypothesis, each component f_i of the vector-valued function $f : \mathbb{R}^n \to \mathbb{R}^m$ is L_i-Lipschitz continuous. Hence, the following holds for all $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^n$ and $i \in \{1, \ldots, m\}$:

$$\left|f_i(\boldsymbol{x}) - f_i(\boldsymbol{x}')\right| \leqslant L_i \left\|\boldsymbol{x} - \boldsymbol{x}'\right\|. \tag{122}$$

Now consider the following:

$$\left\|f(\boldsymbol{x}) - f(\boldsymbol{x}')\right\|^2 = \sum_{i=1}^{m} \left|f_i(\boldsymbol{x}) - f_i(\boldsymbol{x}')\right|^2 \tag{123}$$

$$\leqslant \sum_{i=1}^{m} L_i^2 \left\|\boldsymbol{x} - \boldsymbol{x}'\right\|^2 \tag{124}$$

$$= \left(\sum_{i=1}^{m} L_i^2\right) \left\|\boldsymbol{x} - \boldsymbol{x}'\right\|^2. \tag{125}$$

Hence, $L = \sqrt{\sum_{i=1}^{m} L_i}$ is a Lipschitz constant for the vector-valued function f.

A.3 Proof of Lemma 16

According to Lemma 7, a Lipschitz constant for the multivariate function $h(\boldsymbol{\theta}) = \langle I \otimes O \rangle_{\boldsymbol{\theta}}$ is as follows:

$$L_h = \sqrt{r}\max_i \left\{\sup_{\boldsymbol{\theta}} \left|\frac{\partial h(\boldsymbol{\theta})}{\partial \theta_i}\right|\right\}_i \tag{126}$$

$$\overset{(a)}{=} \sqrt{r}\max_i \left\{\sup_{\boldsymbol{\theta}} \left|2\,\mathrm{Re}\left[\langle \mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i}|\mathbf{0}\rangle\right]\right|\right\}_i \tag{127}$$

$$\overset{(b)}{\leqslant} 2\sqrt{r}\max_i \left\{\sup_{\boldsymbol{\theta}} \left|\langle \mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)U_f(\boldsymbol{\theta})e^{-i\theta_i H_i}(-iH_i)U_b(\boldsymbol{\theta})|\mathbf{0}\rangle\right|\right\}_i \tag{128}$$

$$\overset{(c)}{\leqslant} 2\sqrt{r}\max_i \left\{\sup_{\boldsymbol{\theta}} \left\|U^\dagger(\boldsymbol{\theta})(I\otimes O)U_f(\boldsymbol{\theta})e^{-i\theta_i H_i}(-iH_i)U_b(\boldsymbol{\theta})\right\|\right\}_i \tag{129}$$

$$\overset{(d)}{\leqslant} 2\sqrt{r}\max_i \left\{\sup_{\boldsymbol{\theta}} \left\|U^\dagger(\boldsymbol{\theta})\right\| \|O\| \left\|U_f(\boldsymbol{\theta})\right\| \left\|e^{-i\theta_i H_i}\right\| \|-iH_i\| \left\|U_b(\boldsymbol{\theta})\right\|\right\}_i \tag{130}$$

$$\overset{(e)}{\leqslant} 2\sqrt{r}\,\|O\| \max_i \left\{\|H_i\|\right\}_i. \tag{131}$$

Here, equality (a) is a consequence of the following fact:

$$\frac{\partial h(\boldsymbol{\theta})}{\partial \theta_i} = \langle \mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i}|\mathbf{0}\rangle + \langle \mathbf{0}|\frac{\partial U^\dagger(\boldsymbol{\theta})}{\partial \theta_i}(I\otimes O)U(\boldsymbol{\theta})|\mathbf{0}\rangle, \tag{132}$$

where we used the product rule. Inequality (b) follows from the fact that $|\mathrm{Re}[z]| \leqslant |z|$ for all $z \in \mathbb{C}$. Additionally, we assume $U(\boldsymbol{\theta}) = U_f(\boldsymbol{\theta})e^{-i\theta_i H_i}U_b(\boldsymbol{\theta})$, so that

$$\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_i} = U_f(\boldsymbol{\theta})e^{-i\theta_i H_i}(-iH_i)U_b(\boldsymbol{\theta}). \tag{133}$$

Inequality (c) follows from the fact that $\|A\| = \sup_{|\psi\rangle,|\phi\rangle}\{|\langle\psi|A|\phi\rangle| : \||\psi\rangle\| = \||\phi\rangle\| = 1\}$. Inequality (d) follows from two properties of the spectral norm of a matrix: $\|AB\| \leqslant \|A\|\|B\|$ and $\|A\otimes B\| \leqslant \|A\|\|B\|$. The inequality (e) follows from the fact that $\|V\| = 1$ for every unitary V. Similarly, we can evaluate $L_{\nabla h}$ because each component of ∇h, i.e., $\frac{\partial h(\boldsymbol{\theta})}{\partial \theta_i}$ is Lipschitz continuous. This is because according to the parameter-shift rule, we can write $\frac{\partial h(\boldsymbol{\theta})}{\partial \theta_i}$ in terms of the linear combination of $h(\boldsymbol{\theta} + (\pi/4)\hat{\boldsymbol{e}}_i)$ and $h(\boldsymbol{\theta} - (\pi/4)\hat{\boldsymbol{e}}_i)$.

A.4 Proof of Lemma 17

The multivariate vector-valued function $\nabla_{\boldsymbol{\theta}}\mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})$, where $\mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})$ is defined in (76)–(77), is $L_{c,\boldsymbol{y}}$-Lipschitz continuous for a fixed $\boldsymbol{y} \in \mathbb{R}^M$ if all its components, i.e., $\left\{\frac{\partial \mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})}{\partial \theta_i}\right\}_{i=1}^r$, are Lipschitz continuous for a fixed $\boldsymbol{y} \in \mathbb{R}^M$. In order to prove that $\frac{\partial \mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})}{\partial \theta_i}$ is $L^i_{c,\boldsymbol{y}}$-Lipschitz continuous, we need to bound the Lipschitz constant $L^i_{c,\boldsymbol{y}}$ from above. According to Lemma 7, we state the following:

$$L^i_{c,\boldsymbol{y}} = \sqrt{r}\max_j\left\{\sup_{\boldsymbol{\theta}}\left|\frac{\partial^2\mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})}{\partial\theta_i\partial\theta_j}\right|\right\}_j \tag{134}$$

$$\begin{aligned}
&\leqslant \sqrt{r}\max_j\Bigg\{\sup_{\boldsymbol{\theta}}\Bigg\{\lambda\left|\frac{\partial^2\langle I\otimes C\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right| + \lambda\sum_{m=1}^M|y_m|\left|\frac{\partial^2\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right| \\
&\qquad + \lambda c\sum_{m=1}^M\Bigg(|b_m|\left|\frac{\partial^2\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right| + \lambda\left|\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}\right|\left|\frac{\partial^2\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right| \\
&\qquad\qquad + \lambda\left|\frac{\partial\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}}{\partial\theta_i}\right|\left|\frac{\partial\langle I\otimes A_m\rangle_{\boldsymbol{\theta}}}{\partial\theta_j}\right|\Bigg)\Bigg\}\Bigg\}_j,
\end{aligned} \tag{135}$$

where $j \in \{1,\ldots,r\}$, and the aforementioned inequality follows from the triangle inequality. Let us first bound $\sup_{\boldsymbol{\theta}}\left|\frac{\partial^2\langle I\otimes O\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right|$ from above as follows, where $O \in \{C, A_1,\ldots,A_M\}$:

$$\begin{aligned}
&\sup_{\boldsymbol{\theta}}\left|\frac{\partial^2\langle I\otimes O\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j}\right| \qquad (136)\\
&\overset{(a)}{=} \sup_{\boldsymbol{\theta}}\left\{\left|2\,\mathrm{Re}\left[\langle\mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)\frac{\partial^2U(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j}|\mathbf{0}\rangle\right] + 2\,\mathrm{Re}\left[\langle\mathbf{0}|\frac{\partial U^\dagger(\boldsymbol{\theta})}{\partial\theta_i}(I\otimes O)\frac{\partial U(\boldsymbol{\theta})}{\partial\theta_j}|\mathbf{0}\rangle\right]\right|\right\}\\
&\overset{(b)}{\leqslant} 2\sup_{\boldsymbol{\theta}}\left\{\left|\mathrm{Re}\left[\langle\mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)\frac{\partial^2U(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j}|\mathbf{0}\rangle\right]\right| + \left|\mathrm{Re}\left[\langle\mathbf{0}|\frac{\partial U^\dagger(\boldsymbol{\theta})}{\partial\theta_i}(I\otimes O)\frac{\partial U(\boldsymbol{\theta})}{\partial\theta_j}|\mathbf{0}\rangle\right]\right|\right\}\\
&\overset{(c)}{\leqslant} 2\sup_{\boldsymbol{\theta}}\left\{\left|\langle\mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)U_{ff}(\boldsymbol{\theta})e^{-i\theta_jH_j}(-iH_j)U_{fb}(\boldsymbol{\theta})e^{-i\theta_iH_i}(-iH_i)U_b(\boldsymbol{\theta})|\mathbf{0}\rangle\right|\right.
\end{aligned}$$

$$+\left|\langle \mathbf{0}|U_b^\dagger(\boldsymbol{\theta})e^{i\theta_i H_i}(iH_i)U_f^\dagger(\boldsymbol{\theta})(I\otimes O)U_{ff}(\boldsymbol{\theta})e^{-i\theta_j H_j}(-iH_j)U_{b'}(\boldsymbol{\theta})|\mathbf{0}\rangle\right|\Bigg\}$$

$$\overset{(d)}{\leqslant} 4\,\|O\|\,\|H_j\|\,\|H_i\|\,, \tag{137}$$

where equality (a) follows from the chain rule applied twice (see (132)). Inequality (b) follows from the triangle inequality. Inequality (c) follows from the fact that $|\mathrm{Re}[z]| \leqslant |z|$ for all $z \in \mathbb{C}$, as well as from the assumption that $U(\boldsymbol{\theta})$ has the following decomposition:

$$U(\boldsymbol{\theta}) = U_{ff}(\boldsymbol{\theta})e^{-i\theta_j H_j}U_{fb}(\boldsymbol{\theta})e^{-i\theta_i H_i}U_b(\boldsymbol{\theta}). \tag{138}$$

Additionally, let $U_{ff}(\boldsymbol{\theta})e^{-i\theta_j H_j}U_{fb}(\boldsymbol{\theta}) = U_f(\boldsymbol{\theta})$ and $U_{fb}(\boldsymbol{\theta})e^{-i\theta_i H_i}U_b(\boldsymbol{\theta}) = U_{b'}(\boldsymbol{\theta})$. Inequality (d) follows from a set of arguments similar to those in the proof of Lemma 16.

Second, we bound $\sup_{\boldsymbol{\theta}} |\langle I\otimes O\rangle_{\boldsymbol{\theta}}|$ from above as follows, where $O \in \{A_1, \ldots, A_M\}$:

$$\sup_{\boldsymbol{\theta}} |\langle I\otimes O\rangle_{\boldsymbol{\theta}}| = \sup_{\boldsymbol{\theta}} \left|\langle \mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)U(\boldsymbol{\theta})|\mathbf{0}\rangle\right| \tag{139}$$

$$\leqslant \|O\|\,. \tag{140}$$

Finally, we bound $\sup_{\boldsymbol{\theta}} \left|\frac{\partial\langle I\otimes O\rangle_{\boldsymbol{\theta}}}{\partial\theta_i}\right|$ from above as follows, where $O \in \{A_1, \ldots, A_M\}$:

$$\sup_{\boldsymbol{\theta}} \left|\frac{\partial\langle I\otimes O\rangle_{\boldsymbol{\theta}}}{\partial\theta_i}\right| = \sup_{\boldsymbol{\theta}} \left|2\,\mathrm{Re}\left[\langle \mathbf{0}|U^\dagger(\boldsymbol{\theta})(I\otimes O)\frac{\partial U(\boldsymbol{\theta})}{\partial\theta_i}|\mathbf{0}\rangle\right]\right| \tag{141}$$

$$\leqslant 2\,\|O\|\,\|H_i\|\,. \tag{142}$$

The above inequality follows from the similar set of arguments made in the proof of Lemma 16.

Note that the upper bounds in (137), (140), and (142) are independent of $\boldsymbol{\theta}$. Hence, using these upper bounds, we bound $L^i_{c,\boldsymbol{y}}$ from above as follows:

$$L^i_{c,\boldsymbol{y}} \leqslant \sqrt{r}\max_j \Bigg\{4\lambda\,\|C\|\,\|H_j\|\,\|H_i\| + \lambda\sum_{m=1}^{M} 4\,|y_m|\,\|A_m\|\,\|H_j\|\,\|H_i\| \tag{143}$$

$$+\lambda c\sum_{m=1}^{M}\left(4\,|b_m|\,\|A_m\|\,\|H_j\|\,\|H_i\| + 4\lambda\,\|A_m\|^2\,\|H_j\|\,\|H_i\| + 4\lambda\,\|A_m\|^2\,\|H_j\|\,\|H_i\|\right)\Bigg\}_j$$

$$= 4\lambda\sqrt{r}\,\|H_i\|\left(\|C\| + \sum_{m=1}^{M}\left((|y_m| + c\,|b_m|)\,\|A_m\| + 2c\lambda\,\|A_m\|^2\right)\right)\max_j\{\|H_j\|\}_j \tag{144}$$

From (144) we conclude that the Lipschitz constant $L^i_{c,\boldsymbol{y}}$ is bounded from above by a positive number. Hence, the Lipschitz constant $L_{c,\boldsymbol{y}}$ of $\nabla_{\boldsymbol{\theta}}\mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})$ is also bounded from above by a positive number because $L_{c,\boldsymbol{y}} = \sqrt{\sum_{i=1}^r L^i_{c,\boldsymbol{y}}}$ according to Lemma 8. Thus, the function $\mathcal{L}_c(\boldsymbol{\theta},\boldsymbol{y})$ is $L_{c,\boldsymbol{y}}$-smooth for a fixed $\boldsymbol{y} \in \mathbb{R}^M$.

A.5 Proof of Lemma 20

The multivariate vector-valued function $\nabla_{\boldsymbol{\theta}}\mathcal{F}_\gamma(\boldsymbol{\theta},\bar{\boldsymbol{y}})$ for a fixed $\bar{\boldsymbol{y}} \geqslant 0$, is $L_{\gamma,\bar{\boldsymbol{y}}}$-Lipschitz continuous if all its components, i.e., $\left\{\frac{\partial\mathcal{F}_\gamma(\boldsymbol{\theta},\bar{\boldsymbol{y}})}{\partial\theta_i}\right\}_{i=1}^r$, are Lipschitz continuous for a fixed

$\bar{\boldsymbol{y}} \geqslant 0$. In order to prove that $\frac{\partial \mathcal{F}_\gamma(\boldsymbol{\theta},\bar{\boldsymbol{y}})}{\partial \theta_i}$ is $L^i_{\gamma,\bar{\boldsymbol{y}}}$-Lipschitz continuous, we first bound the Lipschitz constant $L^i_{\gamma,\bar{\boldsymbol{y}}}$ from above as follows:

$$L^i_{\gamma,\bar{\boldsymbol{y}}} = \sqrt{r}\max_j \left\{ \sup_{\boldsymbol{\theta}} \left| \frac{\partial^2 \mathcal{F}_\gamma(\boldsymbol{\theta},\bar{\boldsymbol{y}})}{\partial\theta_i \partial\theta_j} \right| \right\}_j \tag{145}$$

$$= b_M\sqrt{r}\max_j \left\{ \sup_{\boldsymbol{\theta}} \left| \frac{e^{\gamma\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}}{\left(e^{\gamma\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}+1\right)} \frac{\partial^2\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j} \right.\right. \tag{146}$$
$$\left.\left. + \frac{e^{\gamma\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}}{\left(e^{\gamma\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}+1\right)^2}\gamma \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i} \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_j} \right| \right\}_j$$

$$\leqslant b_M\sqrt{r}\max_j \left\{ \sup_{\boldsymbol{\theta}} \left| \frac{\partial^2\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j} + \gamma \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i} \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_j} \right| \right\}_j \tag{147}$$

$$\leqslant b_M\sqrt{r}\max_j \left\{ \sup_{\boldsymbol{\theta}} \left| \frac{\partial^2\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i\partial\theta_j} \right| + \gamma \left| \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_i} \right| \left| \frac{\partial\langle I\otimes H(\bar{\boldsymbol{y}})\rangle_{\boldsymbol{\theta}}}{\partial\theta_j} \right| \right\}_j \tag{148}$$

$$\leqslant 4b_M\sqrt{r}\left\|H_i\right\| \left(\left\|H(\bar{\boldsymbol{y}})\right\| + 4\gamma\left\|H(\bar{\boldsymbol{y}})\right\|^2 \right) \max_j \left\{\left\|H_j\right\|\right\}_j . \tag{149}$$

The last inequality follows from (142) and (137). We see that $L^i_{\gamma,\bar{\boldsymbol{y}}}$ is bounded from above by a positive number for a fixed $\bar{\boldsymbol{y}}$. Therefore, the Lipschitz constant $L_{\gamma,\bar{\boldsymbol{y}}}$ is also bounded from above by a positve number because according to Lemma 8, we have $L_{\gamma,\bar{\boldsymbol{y}}} = \sqrt{\sum_{i=1}^r L^i_{\gamma,\bar{\boldsymbol{y}}}}$. Thus, function $\mathcal{F}_\gamma(\boldsymbol{\theta},\bar{\boldsymbol{y}})$ is $L_{\gamma,\bar{\boldsymbol{y}}}$-smooth for a fixed $\bar{\boldsymbol{y}} \geqslant 0$.