Skoltech

Skolkovo Institute of Science and Technology

ON THE TRAINABILITY OF VARIATIONAL QUANTUM CIRCUITS AS ALGORITHMIC MODELS

Doctoral Thesis
by

LUIS ERNESTO CAMPOS ESPINOZA

DOCTORAL PROGRAM IN COMPUTATIONAL AND DATA SCIENCE AND ENGINEERING

Supervisor
Professor Jacob Daniel Biamonte,
Doctor of Philosophy, Doctor of Physical and Mathematical Sciences

I hereby declare that the work presented in this thesis was carried out by myself at Skolkovo Institute of Science and Technology, Moscow, except where due acknowledgement is made, and has not been submitted for any other degree.

Candidate: Luis Ernesto Campos Espinoza

Supervisor: Professor Jacob Daniel Biamonte

Abstract

The variational model of quantum computing has become the standard in the era of noisy intermediate scale quantum (NISQ) computing due to various factors including partial noise resilience. In this model a variational quantum circuit is iteratively trained by a classical co-processor in order to minimize a cost function. This optimization routine, however, becomes a limiting factor of the variational model when training does not result in a good solution in a reasonable amount of time. This difficulty can be caused, among other reasons, by a large number of parameters, a difficult cost landscape, etc. As a result, numerous training strategies have been developed in order to improve the trainability of variational quantum circuits. Still, many of these strategies have not been examined beyond the specific limitations they were designed to alleviate. This thesis studies some of the most prominent training techniques in order to understand their behavior, limitations, and propose improvements. Firstly, we study the layer-wise training strategy, and its generalization to training layers stacks, in the context of the variational quantum compilation of k-Toffoli gates. Here we discover the *abrupt trainability transitions* limitation, which prevents training unless more than a critical number of layers per stack is used. Furthermore, we prove that for this same problem the *identity initialization strategy* starts the training process in a bad local minimum. Secondly, we study layer-wise training in the context of quantum approximate optimization algorithm (QAOA) for the problems of unstructured search, MAX-CUT, and MAX-k-SAT. Here we discover the *training saturation* limitation, which prevents training after a certain depth. Nevertheless, we discovered that the introduction of small probabilities of coherent phase noise prevents training saturation, thus recovering the prospects of layer-wise training. Lastly we study the problem of unstructured search in QAOA, and find an upper bound to the circuit depth necessary to prepare a state which overlap with the target state approaches unity. As a by-product of this result, we obtain a recipe for variational parameters that can be used to bypass the QAOA training process. These results show prospects of a future where variational quantum algorithms will be able to overcome their most prominent training limitations.

Publications

The thesis is based on the following publications:

1. E. Campos, A. Nasrallah, and J. Biamonte. Abrupt transitions in variational quantum circuit training. *Physical Review A*, 103(3):032607, 2021. doi: 10.1103/PhysRevA.103.032607

2. E. Campos, D. Rabinovich, V. Akshay, and J. Biamonte. Training saturation in layerwise quantum approximate optimization. *Physical Review A*, 104(3): L030401, 2021. doi: 10.1103/PhysRevA.104.L030401

3. E. Campos, D. Rabinovich, and A. Uvarov. Depth scaling of unstructured search via quantum approximate optimization. *Physical Review A*, 110(1): 012428, 2024. doi: 10.1103/PhysRevA.110.012428

Other papers published during the PhD studies:

1. E. Campos, S. Venegas-Andraca, and M. Lanzagorta. Quantum tunneling and quantum walks as algorithmic resources to solve hard k-sat instances. *Scientific Reports*, 11(1):16845, 2021. doi: 10.1038/s41598-021-95801-1

2. V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte. Parameter concentrations in quantum approximate optimization. *Physical Review A*, 104(1): L010401, 2021. doi: 10.1103/PhysRevA.104.L010401

3. V. Akshay, H. Philathong, E. Campos, D. Rabinovich, I. Zacharov, X. Zhang, and J. Biamonte. Circuit depth scaling for quantum approximate optimization. *Physical Review A*, 106(4):042438, 2022. doi: 10.1103/PhysRevA.106.042438

4. D. Rabinovich, S. Adhikary, E. Campos, V. Akshay, E. Anikin, R. Sengupta, O. Lakhmanskaya, K. Lakhmanskiy, and J. Biamonte. Ion-native variational ansatz for quantum approximate optimization. *Physical Review A*, 106(3): 032418, 2022. doi: https://doi.org/10.1103/PhysRevA.106.032418

5. D. Rabinovich, R. Sengupta, E. Campos, V. Akshay, and J. Biamonte. Progress towards analytically optimal angles in quantum approximate optimisation. *Mathematics*, 10(15):2601, 2022. doi: 10.3390/math10152601

6. D. Rabinovich, E. Campos, S. Adhikary, E. Pankovets, D. Vinichenko, and J. Biamonte. Robustness of variational quantum algorithms against stochastic parameter perturbation. *Physical Review A*, 109(4):042426, 2024. doi: 10.1103/PhysRevA.109.042426

Acknowledgments

I would like to express my sincere gratitude to my supervisor Professor Jacob Baimonte for his expert guidance, valuable insights, and unwavering support throughout the course of this thesis. I also appreciate the stimulating scientific discussions and camaraderie with my colleagues at the Deep Quantum Lab, which have greatly enhanced my research experience.

I extend my deepest appreciation to my my parents and brother, who have consistently encouraged and supported my pursuit of a career in science.

I would like to extend a special thank you to my wife, Masha, for her unwavering support and encouragement, as well as to our loyal companion, Carrie, for providing a much needed distraction and comfort during the challenging moments of this journey.

Contents

List of symbols and abbreviations

$\mathbb{C}$	Field of complex numbers
$\mathbb{R}$	Field of real numbers
$\mathbb{B}$	Boolean domain
$\mathcal{H}$	Finite dimensional Hilbert space
$\dim(\mathcal{H})$	Dimension of the Hilbert space
$\mathcal{L}(\mathcal{H})$	Set of linear operators acting in $\mathcal{H}$
$A^\dagger$	Conjugate transpose of a matrix
A^*	Complex conjugate of a matrix
$\mathbb{1}$	Identity operator
$\|\psi\rangle$	A unit norm pure quantum state
$\mathrm{herm}_{\mathbb{C}}(d)$	Set of Hermitian $d \times d$ matrices
$\mathbf{U}_{\mathbb{C}}(d)$	Set of unitary $d \times d$ matrices
$\|A\|_p$	Operator p-norm
$X_j,\ Y_j,\ Z_j$	Pauli operators acting on the jth qubit
$\mathcal{O}(g(x))$	Asymptotic upper bound
P	Class of problems that can be solved in polynomial time
NP	Class of problems where a solution can be verified in polynomial time
NISQ	Noisy intermediate scale quantum
VQA	Variational quantum algorithm
QAOA	Quantum aproximate optimization algorithm
HEA	Hardware efficient ansatz
BP	Barren plateau
ATT	Abrupt trainability transitions
TS	Training saturations
$[H_k, \cdots [H_1, H_0] \cdots]$	Nested commutators

List of Figures

Introduction

Topical characterization

The 20th century saw the groundbreaking development of quantum mechanics, leading to the first quantum revolution in technology. This revolution introduced devices utilizing quantum effects for practical applications, such as the laser, atomic clock, and the transistor – ultimately contributing to the creation of computers that profoundly impacted modern life. Advancements in computer science led to the formulation of complexity theory, and the realization that certain problems are intractable even for the most powerful computers [15–17]. Among those problems is the simulation of quantum mechanical systems, as in the worst case, the classical representation grows exponentially with the size of the quantum system. To overcome these limitations, the idea of using quantum mechanical systems to simulate other quantum systems emerged. This concept was formalized by David Deutsch, who established the Church-Turing-Deutsch principle, a physical form of the Church–Turing thesis, stating that a universal quantum computer could simulate any physical system [18, 19].

Quantum computing harnesses unique quantum properties, such as superposition and entanglement, as algorithmic resources to solve computational tasks that would be too difficult for conventional computers. Some notable quantum algorithms like: Deutsch-Josza's [20], Shor's [21], Simon's [22], and Grover's algorithm [23], demonstrate the notion of "quantum advantage", wherein they outperform the best known classical algorithms for specific problems.

Despite promising potential and state of the art quantum computers achieving infidelities as little as 10^{-5} and 10^{-2} for single and two qubit operations respectively

[24–35], errors multiply rapidly limiting the practical number of quantum operations that they can perform. These so called noisy intermediate scale quantum (NISQ) computers [36, 37] are therefore severely limited when implementing "traditional" quantum algorithms like Grover and Shor's beyond some small system sizes [38–41].

To work around these constraints, researchers have developed what are called variational quantum algorithms (VQAs) [12, 42]. In them, a parameterized quantum circuit (a sequence of quantum operations) is iteratively optimized by a classical co-processor in order to minimize a cost function, taking clear inspiration from machine learning. Thanks to their flexibility, VQAs can make use of short quantum circuits that take advantage of the specific quantum hardware [43–46]. Although their resilience to some of the systematic limitations of NISQ computers have made them widely popular [47–52], certain limitations specific to these algorithms have been discovered [1, 2, 12, 53–74].

Of our particular interest are the trainability limitations, which prevent the optimization process from finding good solutions in a reasonable amount of time. These limitation can arise due to a variety of factors: a large number of parameters to be optimized [53–55], an optimizer unfit for a given task [64, 75, 76], a difficult cost landscape with many local minima [65, 68, 77] or very flat as in the case of the infamous *barren plateaus* (BP) [14, 72, 74, 78–83], and others [59, 60, 67]. As a response, a wide variety of training techniques have been proposed to alleviate these limitations. These techniques vary widely in their approach, some of them focusing on: the circuit parameters [70, 84, 85], the circuit structure [62, 86–91], the cost function [56, 92, 93], or others [94–97]. Unfortunately for many of these strategies their behavior is mostly outside their impact on the very specific type of limitation they were designed to alleviate.

Thesis goals

This thesis studies some of the most prominent training techniques for variational quantum algorithms. It focuses on their advantages, limitations, and proposes improvements. To realize these objectives we:

1. Study the so called *layer-wise trainability conjecture* that attests any quantum circuit can be trained by building it sequentially and only optimizing the newly added layers.

2. Perform numerical experiments to study *layer-wise training*, and its generalization to groups of layers, for challenging and computationally relevant problems.

3. Study the *identity initialization strategy* used commonly to avoid the training limiting effect known as *barren plateaus*. This strategy consists of initializing the variational parameters of a quantum circuit to zeros. We study its performance for the same problems as in point 2.

4. Study the performance of layer-wise training on the quantum approximate optimization algorithm (QAOA) for unstructured search, and the NP-hard problems: MAX-CUT and MAX-k-SAT.

5. Perform numerical experiments to study the impact of coherent phase noise in layer-wise training in QAOA for the same problems mentioned in point 4.

6. Study the problem of unstructured search in QAOA to obtain parameters that guarantee an overlap approaching unity with the target, which can be used as initial parameters or to bypass the training process entirely.

Statements to defend

1. The existence of *abrupt trainability transitions* in layer-wise variational quantum compilation, where a minimum number of layers per stack is required for successful training [1].

2. Proof that the cost function for the variational quantum compilation of $k-$Toffoli gates takes extrema when the circuit evaluates to identity [1].

3. The existence of *training saturation* in layer-wise QAOA, where training halts after a certain number of layers [2].

4. Showing that coherent phase noise removes training saturation, thereby recovering prospects of layer-wise training [2].

5. Proof that for unstructured search on n qubits, QAOA requires a depth shorter than $\mathcal{O}\left(N^{\frac{1}{2}+c}\right)$, for any $c > 0$ and $N = 2^n$, to prepare a state which overlap with the target approaches unity [3].

Scientific novelty

1. The discovery of the *abrupt trainability transitions* presents a counterexample to the layer-wise trainability conjecture that asserts a quantum circuit can be trained piecewise.

2. The proof of the identity minima in variational quantum compilation of k-Toffoli gates presents an example where the *identity initialization strategy*, which consists on initializing a variational circuit as identity, starts the optimization process in a bad local minimum.

3. The discovery of *training saturation* presents a second counter example to the layer-wise trainability conjecture.

4. The use of low levels of coherent phase noise to remove training saturations is, to our knowledge, the second reported case (after [98]) of a beneficial use of noise in a variational quantum algorithm.

5. The QAOA sequence for unstructured search we derived are the first to have an overlap that approaches unity while having a sub-classical complexity.

Presentation and validation of results

The main results compiled in this thesis are based on published articles in peer-reviewed journals.

1. E. Campos, A. Nasrallah, and J. Biamonte. Abrupt transitions in variational quantum circuit training. *Physical Review A*, 103(3):032607, 2021. doi: 10.1103/PhysRevA.103.032607

   Contributions: Development of the main ideas, analytical results, and numerical experiments.

2. E. Campos, D. Rabinovich, V. Akshay, and J. Biamonte. Training saturation in layerwise quantum approximate optimization. *Physical Review A*, 104(3): L030401, 2021. doi: 10.1103/PhysRevA.104.L030401

   Contributions: Development of the main ideas, numerical experiments, and verified analytics by Daniil Rabinovich

3. E. Campos, D. Rabinovich, and A. Uvarov. Depth scaling of unstructured search via quantum approximate optimization. *Physical Review A*, 110(1): 012428, 2024. doi: 10.1103/PhysRevA.110.012428

   Contributions: Development of the main ideas, main analytical results, numerical experiments, and verified analytics by Daniil Rabinovich

Some of the results have also been presented as posters in the following conferences:

1. VI International Conference on Quantum Technologies 2021 Digital Edition (July 12-16, 2021, Moscow, online)

2. 4th International Online and Onsite Advanced Course on Data Science and Machine Learning ACDL 2021 (July 19-23, 2021, Tuscany, online)

Validity of the results are supported by numerical experiments and/or mathematical proofs wherever applicable.

Chapter 1

Theoretical background

1.1 Definitions

In quantum computing, the basic unit of quantum information is called a qubit [99]. It is a two level quantum system that can be mathematicaly described by a two dimensional vector in a complex Euclidean space $\mathbb{C}_2$.

Definition 1 (Computational basis [100]) *The computational basis is an orthonormal basis for single qubit states given by*

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{1.1}$$

Any single qubit state can be written as a linear combination of states $|0\rangle$, and $|1\rangle$ as

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \tag{1.2}$$

where $\alpha, \beta \in \mathbb{C}$, and $|\alpha|^2 + |\beta|^2 = 1$. Similarly, multi qubit systems can be represented as vectors in $\mathbb{C}_2^{\otimes n}$, where n is the number of qubits. The basis vectors for an n-qubit system can be obtained by the tensor product of the single qubit computational basis $\{|0\rangle, |1\rangle\}^{\otimes n}$. For simplicity we will write the tensor product of states as $|a_1\rangle \otimes$

$|a_2\rangle \otimes \cdots \otimes |a_n\rangle = |a_1 a_2 \cdots a_n\rangle$. In general any n-qubit state can be written as

$$|\psi\rangle = \sum_{j \in \{0,1\}^{\times n}} \alpha_j |j\rangle = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{2^n} \end{bmatrix}, \tag{1.3}$$

where $\alpha_j \in \mathbb{C}$, and $\sum_j |\alpha_j|^2 = 1$. The covector of a state $|\psi\rangle$ is given by

$$\langle\psi| = \sum_{j \in \{0,1\}^{\times n}} \alpha_j^* \langle j| = [\alpha_1^* \ \alpha_2^* \cdots \alpha_{2^n}^*]. \tag{1.4}$$

Definition 2 (Inner product [101]) *The inner product of two states $|\phi\rangle, |\psi\rangle \in \mathbb{C}_2^{\otimes n}$ is given by*

$$\langle\phi|\,\psi\rangle = \sum_{j}^{2^n-1} \phi_j^* \psi_j \in \mathbb{C}, \tag{1.5}$$

where $\phi_j, \psi_j \in \mathbb{C}$ are the coefficients of $|\phi\rangle, |\psi\rangle$ respectively.

With the addition of the inner product, the n-qubit space forms a finite dimensional Hilbert space $\mathcal{H}$.

Remark 1 *$\mathcal{L}(\mathcal{H})$ denotes the space of linear maps from $\mathcal{H}$ to itself.*

Definition 3 (Hamiltonians and propagators)

1. *Hamiltonians H in* $\mathrm{herm}_{\mathbb{C}}(2^n) \equiv \{H \in \mathcal{L}(\mathcal{H}) \mid H = H^\dagger\}$.
2. *Propagators U in* $\mathbf{U}_{\mathbb{C}}(2^n) \equiv \{U \in \mathcal{L}(\mathcal{H}) \mid U^\dagger U = \mathbb{1}\}$.

Definition 4 (Eigenstates and eigenvalues [102]) *Given an operator $A \in \mathcal{L}(\mathcal{H})$, a state $|\psi\rangle$ is an eigenstate if*

$$A|\psi\rangle = \alpha|\psi\rangle, \tag{1.6}$$

where $\alpha \in \mathbb{C}$ is the corresponding eigenvalue .

Remark 2 (Spectral decomposition of a Hermitian operator [103]) *An operator $H \in herm_{\mathbb{C}}(2^n)$ can be decomposed as*

$$H = \sum_{j=1}^{2^n} \lambda_j \left|\lambda_j\right\rangle\!\left\langle\lambda_j\right|, \tag{1.7}$$

where $\{|\lambda_j\rangle\}_{j=1}^{2^n}$ form an orthonormal basis called the eigenbasis of H, and the corresponding eigenvalues $\lambda_j \in \mathbb{R}$ are called the spectrum of H .

Definition 5 (Pauli matrices [100])

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \; Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \; Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1.8}$$

Pauli matrices together with the identity matrix

$$\mathbb{1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{1.9}$$

form a basis for the 2×2 matrices [100]. For convenience, in some cases the Pauli matrices and the identity will be indexed as $\sigma^0 = \mathbb{1}, \; \sigma^1 = X, \; \sigma^2 = Y, \; \sigma^3 = Z$.

The tensor products of Pauli matrices and the identity $\{\mathbb{1}, X, Y, Z\}^{\otimes n}$, called Pauli strings, form a basis for the linear operators in $\mathcal{L}(\mathbb{C}_2^{\otimes n})$,

$$A = \sum_{\alpha \in \{0,1,2,3\}^{\times n}} a_\alpha \bigotimes_{j=1}^{n} \sigma_j^{\alpha_j}, \tag{1.10}$$

where $a_\alpha \in \mathbb{C}$, and $\sigma_j^{\alpha_j}$ acts on the j-th qubit. For a given matrix A the coefficients a_α can be calculated as

$$a_\alpha = \mathrm{Tr}\left(A \cdot \bigotimes_{j=1}^{n} \sigma^{\alpha_j} \right). \tag{1.11}$$

Note that $A \in \mathrm{herm}_{\mathbb{C}}(2^n)$ whenever $\forall \; a_\alpha \in \mathbb{R}$.

Remark 3 (Hamiltonian evolution [104]) *The evolution of a time independent*

Hamiltonian H is given by the propagator

$$U(\theta) = e^{-iH\theta} = \sum_{j=0}^{\infty} \frac{(-iH\theta)^j}{j!}, \tag{1.12}$$

where $\theta \in \mathbb{R}$.

If H is both unitary and Hermitian then its propagator is

$$U(\theta) = e^{-iH\theta} = \mathbb{1}\cos\theta - iH\sin\theta, \tag{1.13}$$

as is the case for Pauli matrices.

Definition 6 (Pauli rotations [104]) *Pauli rotations are parameterized unitary operators given by*

$$R_x(\theta) = e^{-iX\theta},\ R_y(\theta) = e^{-iY\theta},\ R_z(\theta) = e^{-iZ\theta}. \tag{1.14}$$

Definition 7 (Controlled operators) *Given a unitary $U \in \mathbf{U}_{\mathbb{C}}(2)$, a controlled operator $CU \in \mathbf{U}_{\mathbb{C}}(4)$ can be expressed as*

$$CU = |0\rangle\langle 0| \otimes \mathbb{1} + |1\rangle\langle 1| \otimes U, \tag{1.15}$$

where the first qubit space corresponds to the control qubit and the second to the target qubit [105].

Definition 8 (Measurement [100]) *Given a Von Neumann observable [106] $H \in herm_{\mathbb{C}}(2^n)$ with eigenvalues λ_j, and a state $|\psi\rangle$, the probability of measuring λ_j is given by the Born rule*

$$P(\lambda_j) = \langle\psi| \Pi_j |\psi\rangle, \tag{1.16}$$

where Π_j is a projector into the eigenspace of H corresponding to λ_j.

Definition 9 (Expected value [107]) *The expected value of an observable H with respect to a state $|\psi\rangle$ is the average of its eigenvalues λ_j weighted by the probabilities*

given by the Born rule

$$\langle H \rangle_\psi \equiv \langle \psi | H | \psi \rangle = \sum_{j=0}^{2^n-1} \lambda_j P(\lambda_j). \tag{1.17}$$

1.2 Quantum circuits

Analogous to the classical gate model, the gate model of quantum computing consists of a series of operations which are applied to an input in order to produce a desired output. In it, a series of unitary operations, called quantum gates, are applied sequentially to an input quantum state. The resulting final state is then measured in order to recover information. This sequence of quantum gates together with its input state is called a quantum circuit. It is sometimes useful to represent the quantum circuit diagrammatically. Figure 1-1 illustrates an example of the equivalence between equation and diagrammatic representation.

$$|\psi_{out}\rangle = (\mathbb{1} \otimes U_5)(U_4 \otimes \mathbb{1})(U_1 \otimes U_2 \otimes U_3) |\psi_1\rangle \otimes |\psi_2\rangle \otimes |\psi_3\rangle$$

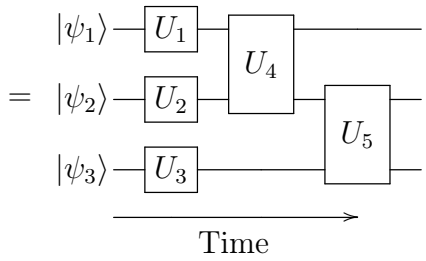


Figure 1-1: Quantum circuit composed of single and two qubit quantum gates applied to a three qubit quantum state in both equation and diagrammatic representations. Usually time flows from left to right in the diagrammatic representation.

1.3 Hamiltonian simulation

One of the prospects of quantum computing is the simulation of Hamiltonians. In it, we start from an initial state $|\psi_0\rangle$ and evolve it according to the rules defined by the underlying Hamiltonian operator H as

$$|\psi(t)\rangle = e^{-iHt/\hbar} |\psi_0\rangle , \tag{1.18}$$

which is a solution to the time independent Schrödinger equation [104],

$$i\hbar\frac{\partial}{\partial t}\left|\psi(t)\right\rangle = H\left|\psi(t)\right\rangle, \tag{1.19}$$

where $\hbar$ is the reduced Planck constant which has units of energy multiplied by time. For simplicity in this thesis we use Planck units $\hbar = 1$ [108].

It is natural to imagine one can decompose the unitary $U(t) = e^{-iHt}$ in a universal [109] set of single and two qubit gates. Unfortunately, we only have knowledge of the generating Hamiltonian H, and for most cases there is no obvious way to decompose $U(t)$. One strategy is to decompose the Hamiltonian as a sum

$$H = \sum_{\mu=1}^{M} H_\mu \tag{1.20}$$

such that each unitary $e^{-iH_\mu t}$ is easily implemented in a particular quantum computer. Unfortunately, for most cases $[H_j, H_{k\neq j}] \neq 0$. An alternative is to use a product formula like the first order Trotter formula [110, 111]

$$S_1(t) = \prod_{\mu=1}^{M} \exp(-iH_\mu t), \tag{1.21}$$

which has error $S_1(t) = U(t) + \mathcal{O}(t^2)$.

In order to further decrease the error, one can use higher order product formulas, like the Trotter-Suzuki formulas [110] which can be recursively generated as,

$$S_2 = (e^{-iH_M t/2}\cdots e^{-iH_1 t/2})(e^{-iH_1 t/2}\cdots e^{-iH_M t/2}) \tag{1.22}$$

$$S_{2k} = S_{2k-2}^2(u_k t)S_{2k-2}((1-4u_k)t)S_{2k-2}^2(u_k t), \tag{1.23}$$

where $u_k = 1/(4 - 4^{1/(2k-1)})$. In general a q-th order Suzuki product formula has the form

$$S_q(t) = \prod_{v=1}^{\Upsilon_q}\prod_{\mu=1}^{M} e^{-ita(v,\mu)H_{\pi(v,\mu)}}, \tag{1.24}$$

where $\Upsilon_q = 2\cdot 5^{k-1}$ is the number of stages in the formula. These higher order

formulas satisfy,

$$S_q(t) = U(t) + \mathcal{O}(t^{q+1}). \tag{1.25}$$

In order to approximate an evolution $U(t)$ with large t one should partition an evolution into r smaller Trotter steps $S_q(t/r)$. The approximation error in this case is given by Theorem 1, which was derived by Childs et al. in [112] (where it appears as Corollary 7),

Theorem 1 (Trotter error with commutator scaling) *Let $H = \sum_{\mu=1}^{M} H_\mu$ with $H_\mu \in \mathrm{herm}_{\mathbb{C}}(2^n)$, and $S_q(t)$ be a q-th order product formula. Then*

$$\|U(t) - S_q^r(t/r)\|_2 \leq \epsilon = \frac{2\Upsilon^{q+1}\delta(t)^{q+1}}{r^q(q+1)}, \tag{1.26}$$

where,

$$\delta = \sum_{\mu_1,\mu_2,\cdots,\mu_{q+1}=1}^{M} \|[H_{\mu_{q+1}}, \cdots [H_{\mu_2}, H_{\mu_1}]\cdots]\|_2. \tag{1.27}$$

1.4 Variational quantum algorithms

Variational quantum algorithms (VQAs) are a class of quantum algorithms designed to take advantage of the current noisy intermediate-scale quantum (NISQ) computers [12]. Such algorithms are composed of a parameterized quantum circuit which is iteratively tuned by a classical co-processor in a quantum to feedback loop reminiscent of classical machine learning. The optimization is done in order to minimize a cost function, typically the expectation value of a so called problem Hamiltonian which has encoded in its ground state the solution of a given problem.

VQAs have been proven to be a universal model for quantum computing [10, 113, 114], although with current NISQ devices the extent of their use is still very limited. VQAs have been proposed for use in a wide range of problems, notable examples being: ground energy approximation [51, 115, 116], Hamiltonian simulation [117–120], optimization [121], compilation [13, 122, 123], machine learning [50, 124–126], among others [56, 127–130]. Notable recent experimental demonstrations include [131–135].

One of the main advantages of this class of algorithms is its versatility, as they do not require prior knowledge of an specific circuit to solve a given problem [10, 127, 136–138]. This in turn makes it suitable to be used with short gate sequences, thus reducing the effect of noise present in NISQ devices [52, 139, 140]. Usually these parameterized quantum circuits are composed of ansatz layers.

Definition 10 (Ansatz Layer) *An ansatz layer $V(\boldsymbol{\theta})$ is a fixed sequence of quantum gates where all or some of them are parameterized.*

Examples of ansatz layers commonly found in the literature are the hardware efficient ansatz (HEA) and the checkerboard ansatz:

1. Hardware efficient ansatz: This ansatz arose due to the spatially fixed qubit connectivity [43, 46] as common on most platforms e.g. superconducting chips [26, 141]. A single layer of the hardware efficient ansatz consists of three single-qubit rotations on each wire followed by a control rotation in a daisy chain. Our implementation makes use of single qubit $R_y(\theta)$ and $R_z(\theta)$ rotations, and controlled rotations $CR_y(\theta)$ as illustrated in Figure 1-2.

2. Checkerboard ansatz (a.k.a. bricklayer ansatz): Appears commonly throughout the literature [11, 45, 142] and can be viewed as the trotterized evolution of a local Hamiltonian in a line. It consists of entangling blocks alternating between even and odd qubits. Our implementation makes use of single qubit $R_z(\theta)$ and $R_x(\theta)$ rotations, and entangling gates $R_{zz}(\theta) = \exp(-iZ \otimes Z(\theta))$ as illustrated in Figure 1-3.

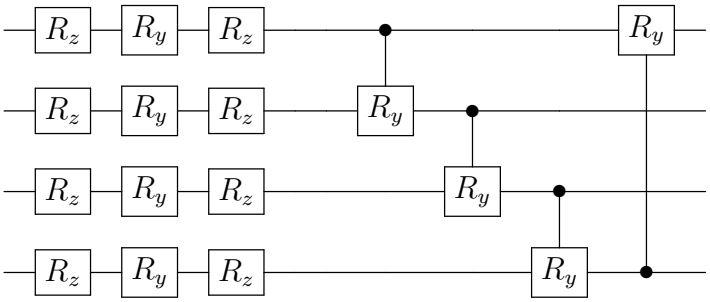


Figure 1-2: Single layer of the Hardware efficient ansatz (HEA) for 4 qubits. Figure reproduced from [10].

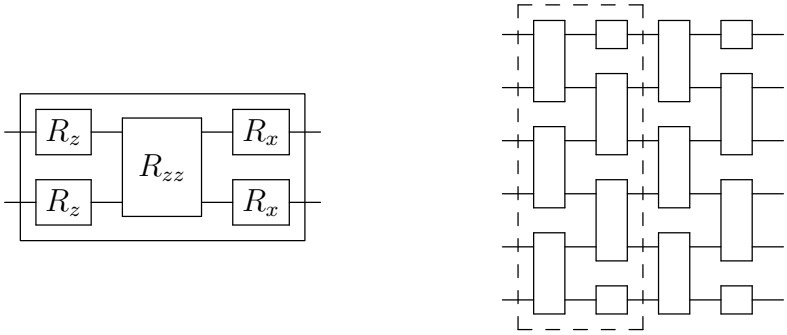


Figure 1-3: Left: single entangling block of the checkerboard ansatz. Right: checkerboard ansatz structure, the dashed line encloses a single layer. Figure reproduced from [11].

Other commonly used ansatz include the quantum approximate optimization ansatz (explained in Section 1.4.2).

The ansatz layers are arranged in a repeated fashion to form an Ansatz circuit, with the number of layers being called the ansatz depth.

Definition 11 (Ansatz circuit) *An ansatz circuit is a parameterized quantum circuit formed by an ansatz layer $V(\boldsymbol{\theta}_j)$ concatenated multiple times,*

$$U(\boldsymbol{\theta}) = \prod_{j=1}^{p} V(\boldsymbol{\theta}_j), \tag{1.28}$$

where p is the ansatz depth and $\boldsymbol{\theta} = \bigcup_j^p \boldsymbol{\theta}_j$.

Similarly we call $|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta}) |0\rangle^{\otimes n}$ an ansatz state.

Definition 12 (Variational state space) *The union of all possible states prepared by a parametrized quantum circuit is called the variational state space*

$$\Omega = \bigcup_{\boldsymbol{\theta}} \{|\psi(\boldsymbol{\theta})\rangle\}. \tag{1.29}$$

Notice that Ω is a subset of $\mathcal{H}$, the Hilbert space of n-qubits. In general, the more parameters an ansatz has the more states in $\mathcal{H}$ it is able to prepare, making it more expressive.

For the variational quantum algorithms whose objective is to minimize a given problem Hamiltonian, the procedure consists of iteratively searching the variational

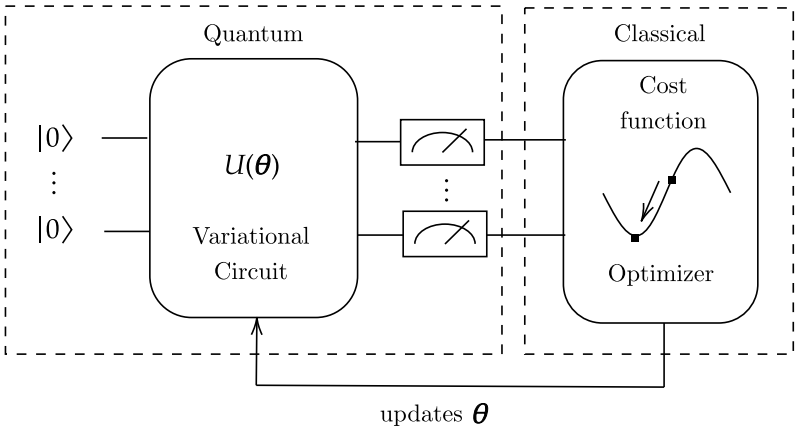


Figure 1-4: Schematic of a variational quantum algorithm depicting the quantum-to-classical feedback loop. Figure reproduced from [12].

state space for the ground state of the problem Hamiltonian H

$$\min_{|\psi\rangle\in\Omega} \langle\psi| H |\psi\rangle \geq \min_{|\phi\rangle\in\mathcal{H}} \langle\phi| H |\phi\rangle . \tag{1.30}$$

Classical optimization techniques can be used to iteratively optimize the parameters, which lowers the burden on the quantum computer. This process is illustrated in Figure 1-4.

1.4.1 Variational quantum compiling

Definition 13 (Quantum compilation problem) *Given a unitary T, we wish to decompose T into a sequence of gates U such that $T = Ue^{i\alpha}$, $\alpha \in \mathbb{R}$.*

This compilation problem dates back to the start of the field of quantum information processing [99]. It is encountered when implementing algorithms on quantum hardware, as the algorithm may require gates that are not part of the native gate set of the quantum computer [143, 144]. This problem is especially commonplace when implementing traditional quantum algorithms, e.g. Grover's algorithm [23] (see Appendix A.1.1). These traditional algorithms make extensive use of CX (controlled X) gates, while actual quantum computers may perform easier to implement entangling gates, e.g. the R_{xx}, also called Mølmer–Sørensen, gates used in ion based quantum computers [145].

Compilation is also used for circuit compression. Given a unitary T composed of q gates, we aim to compile an equivalent unitary U with $r < q$ gates. Additionally quantum compilation can be used to prepare an unknown black-box unitary [132].

The quantum compilation problem has been shown to be in the NP-Complete complexity class [146], and has been approached in multiple ways, notably: via tensor contractions [147, 148], classical deep neural networks [149], through variational quantum algorithms [123, 150, 151], among others [13, 152–155].

Definition 14 (Hilbert-Schmidt inner product [156]) *Given two operators A and B, the Hilbert-Schmidt inner product can be defined as*

$$(A, B)_{\mathrm{HS}} = \mathrm{Tr}(A^\dagger B). \tag{1.31}$$

Definition 15 (Approximate quantum compilation) *Given a target unitary T acting on n qubits and $N = 2^n$, we wish to approximate T by a sequence of gates U such that the distance function,*

$$d(U, T) = 1 - \frac{1}{N}|(U, T)_{\mathrm{HS}}|, \tag{1.32}$$

is less than an error $\epsilon \in \mathbb{R}$.

The approximate compilation of a target unitary T acting on n qubits can be performed by the means of iteratively optimizing the parameters of a variational quantum circuit $U(\boldsymbol{\theta})$ to minimize the cost function

$$d(U(\boldsymbol{\theta}), T) = 1 - \frac{1}{N}|(U(\boldsymbol{\theta}), T)_{\mathrm{HS}}| \in [0, 1]. \tag{1.33}$$

The Hilbert-Schmidt inner product can be calculated in a quantum computer given the fact that

$$\left\langle \Phi^+ \right| \left(U(\boldsymbol{\theta})^\dagger T \otimes \mathbb{1}^{\otimes n}\right) \left| \Phi^+ \right\rangle = \frac{1}{N}(U(\boldsymbol{\theta}), T)_{\mathrm{HS}}, \tag{1.34}$$

where $|\Phi^+\rangle$ is the generalized Bell state

$$\left|\Phi^+\right\rangle = \prod_{j=1}^{n} H_j CX_{j+n}^{j} \left|0\right\rangle^{\otimes 2n} \tag{1.35}$$

$$= \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \left|j\right\rangle \otimes \left|j\right\rangle, \tag{1.36}$$

where H_j is a Hadamard gate applied to the $j-$th qubit, and the upper and lower indices in the CX gates indicate positions of the control and target qubits respectively [157]. Thus one can calculate the cost function (1.33) in a quantum computer by sampling the circuit illustrated in Figure 1-5 since the probability the probability of measuring $|0\rangle^{\otimes 2n}$ is

$$|\left\langle\Phi^+\right| \left(U(\boldsymbol{\theta})^\dagger T \otimes \mathbb{1}\right) \left|\Phi^+\right\rangle|^2 = |\frac{1}{N}(U(\boldsymbol{\theta}), T)_{\mathrm{HS}}|^2. \tag{1.37}$$

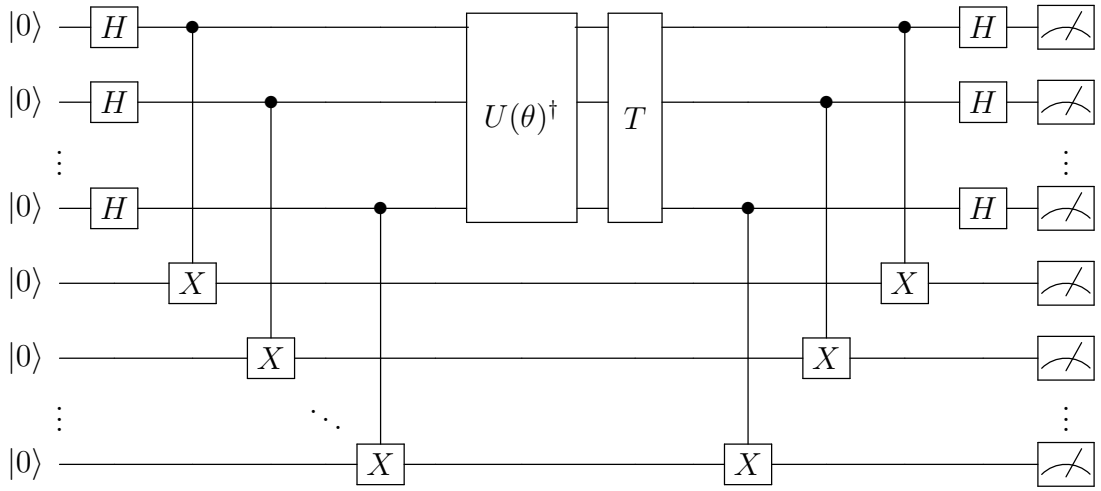


Figure 1-5: Quantum circuit calculating the Hilbert-Schmidt product of $U(\boldsymbol{\theta})$ and T. Here H represents the Hadamard gate. Figure reproduced from [13].

One problem of this approach is that $|\frac{1}{N}(U(\boldsymbol{\theta}), T)_{\mathrm{HS}}|^2$ vanishes exponentially with respect to the system size for a random $U(\boldsymbol{\theta})$. To alleviate this issue a modification to the cost function (1.33) has been proposed [13, 151] where instead of calculating the probability of measuring $|0\rangle^{\otimes n}$, the cost function considers the average probability of measuring qubit pairs as $|00\rangle$.

1.4.2 Quantum approximate optimization algorithm

The quantum approximate optimization algorithm (QAOA) [121] was originally designed to obtain approximate solutions to combinatorial optimization problems [61, 113, 114, 121, 158–166], but it was latter proved to be a universal model of quantum computing [113, 114]. Recent milestones include experimental demonstration of $3p$-QAOA (depth-three, corresponding to six tunable parameters) using 23 qubits [167], depth 2 using 32 qubits [32], analytical results on specific problem instances [2, 3, 5, 163–165, 168], as well as several proposed modifications to improve certain aspects of the original implementation of the algorithm including: constraint awareness [86, 169], the use of quantum walk-based mixing operators [170–174], among others [159–161, 175]. Although QAOA offers a pathway towards quantum advantage [162] limitations are known for low depth QAOA [61, 138, 176]. QAOA takes inspiration from adiabatic quantum computing [177] and can be thought as a trotterized version of it. In fact, for depth $p \to \infty$, QAOA can recover an adiabatic evolution.

QAOA aims to minimize the expectation value of a problem Hamiltonian H, usually diagonal in the computational basis, which encodes the solution of a given problem in its ground state. QAOA uses an ansatz state prepared via the alternated application of two parameterized gates to a state initialized in a uniform superposition in the computational basis,

$$|\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = \prod_{k=1}^{p} e^{-iH_x\beta_k} e^{-iH\gamma_k} |+\rangle^{\otimes n}, \tag{1.38}$$

where where p is the ansatz depth, $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^{\times p}$, $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$, and $H_x = \sum_{j=1}^{n} X_j$ is the standard one-body mixer Hamiltonian with Pauli matrix X_j applied to the j-th qubit. Then the expected value $\langle\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})| H |\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle$ is calculated and the parameters updated. This optimization process is repeated iteratively, resulting in a state

$$|\psi(\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*)\rangle = \underset{|\psi(\boldsymbol{\gamma},\boldsymbol{\beta})\rangle}{\operatorname{argmin}} \langle\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})| H |\psi(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle, \tag{1.39}$$

where $\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*$ are called the optimal parameters, and $|\psi(\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*)\rangle$ approximates a

ground state of H.

1.4.3 Trainability limitations of VQAs

Definition 16 (Trainability of variational quantum circuits) *Given a variational quantum circuit used in a variational quantum algorithm with a cost function $C(\boldsymbol{\theta})$, and $\boldsymbol{\theta}^* = \text{argmin}_{\boldsymbol{\theta}} C(\boldsymbol{\theta})$, the variational quantum circuit is trainable if the optimization process can produce angles $\boldsymbol{\theta}^{\#}$ in a reasonable amount of time, such that $C(\boldsymbol{\theta}^{\#}) \leq C(\boldsymbol{\theta}^*) + \epsilon$, where $\epsilon \geq 0$ is the error tolerance.*

Although variational quantum algorithms are currently the most viable model for use in NISQ devices, they are prone to limiting effects [53–55, 57, 58, 61]. This thesis focuses on the limitations that prevent circuits from being trainable. Similar trainability limitations have been found for classical machine learning algorithms [178–181], but luckily most classical algorithms are trainable. Unfortunately, variational quantum algorithms do not seem to have inherited the nice training properties of their classical counterpart, as they are more prone to experience these limitations [59].

The necessity of overcoming these limitation led to the creation of multiple training strategies, some of which take inspiration from classical machine learning, while others focus on the particular aspects of VQAs. These limitations can be classified according to the mechanism behind them, although in some cases these mechanisms can be related to one another [12, 55]. Consequently, some training strategies have found success at alleviating multiple limitations [5, 6, 62].

The following list includes some of the most relevant types of trainability limitations:

1. Limitations from a large number of parameters. This limitation results from a bottleneck due to the sheer amount of parameters that need to be optimized by the classical co-processor [182]. In some cases these circuits can be *over-parameterized*, meaning the amount of parameters is unnecessarily large for a given problem [53–55].

Strategies that aim to alleviate this limitation include: reducing the number of parameters [183], or take a divide and conquer approach where just a subset of the parameters is optimized at a time. Examples of the latter include: *layer-wise training* [2, 62] where the ansatz circuit is grown layer by layer and each new layer is optimized exclusively, fixing the parameters from previous layers, and *training by layer stacks* [1] which generalizes layer-wise training by training more than one layer at a time. These training strategies are discussed in more detail in Chapters 2 and 3. Both strategies are also commonly applied to the training of classical machine learning models [184–188].

2. Limitations from the optimizer. The classical optimizer of choice plays a significant role on the training of the variational circuit, which can result in slow convergence or getting stuck in a local minimum. Performance in these cases is heavily dependant on: the presence of noise [63], the optimizer hyperparameters[64], among others [65, 66]. As such, many reviews have compared optimizers and their performance under different settings [64, 75, 76, 189].

3. Limitations from layer expressivity. The expressivity of a variational quantum circuit refers to how much of the Hilbert space it is able to cover. Layer and stack wise [1, 2] training techniques can result in limitations that arise due to the low expressivity of individual layers or stacks. This is despite cases where the complete circuit is expressive enough to minimize a given cost function. Known limitations of this type are *abrupt trainability transitions* (ATT) [1] (details in Chapter 2) and *training saturations* (TS) [2] (details in Chapter 3). Strategies that alleviate these limitations include: the use of low probability coherent phase noise (details in Chapter 3) [2], and alternating between layer-wise and global optimization [5, 6].

4. Limitations from a cost landscape full of local minima. It has been shown that the training of certain variational circuits falls in the NP-hard complexity class due to the cost landscape having exponentially many local minima [67–69]. Similarly, it has been shown that some cost landscapes can have a

concentration of local minima far away from the global [59, 60, 70, 71]. These limitations are remarkable as they can occur in shallow depth variational circuits. Strategies against these limitations focus on parameter initialization strategies which give initial parameters with high probability of being in the vicinity of the global minimum. It has been observed that for certain combinatorial problems in QAOA optimal parameters concentrate [5, 163, 190], and theoretical expressions for good initial parameters have also been found for unstructured search in QAOA [3, 168].

5. Limitations from a flat cost landscape. Also known as *barren plateaus* (BPs), it is currently the most studied trainability limitation. It manifests as exponentially vanishing gradients that concentrate around zero in the cost function [72, 73]. Consequently, as the number of measurements necessary to calculate an expectation value with an error ϵ scales as $\mathcal{O}(\epsilon^{-2})$, gradient-based optimizers require an exponential number of measurements to resolve the gradients at a BP. Similarly, gradient-free optimizers require exponentially many measurements as BP suppresses cost differences [83].

   The causes of BPs are varied: they can be caused by the use of problem agnostic ansatz that make it necessary to search in an exponentially large parameter space [12]. Another source of barren plateaus is incoherent phase noise [74] since cost functions are usually minimized by pure states, the noise further erodes the optimization landscape.

   As the most researched trainability limitation, the list of techniques proposed to alleviate BPs is extensive and shares certain strategies used to alleviate other limitations. Some of the most prominent strategies against BPs are: *layer-wise training* as it guaranties the randomness of the initial parameters remains small [62], *identity initialization* where parameters are initialized such that the ansatz evaluates to an identity which reduces randomness [84], initializing near the global minimum [159], using problem specific ansatz that aim is to restrict the space explored by using a problem inspired ansatz which can be trained even with randomly initialized parameters [87, 89], and using local cost

functions which calculate the costs at a subsystem level which remove barren plateaus as long as the circuit depth is logarithmic in n [56, 92, 93], among others [70, 85, 86, 88, 88, 90, 94–97]. As a result of alleviating BPs, these techniques reduce the number of measurements necessary for the optimizers to work properly. Additionally, some VQAs have shown forms of intrinsic resilience to BP: Quantum neural networks (QNN) and quantum convolutional neural networks (QCNN) with tree tensor network architectures do not exhibit BP [191, 192], and variational quantum compilation has an intrinsic resilience to incoherent noise, where optimal parameters are unaffected by its presence [47]. It is currently unclear if other VQAs share this property. It has been also noted that the use of training techniques that avoid BP landscapes may allow for the classical simulability in some cases, by restricting the evolution to polinomially large subspaces [193].

Chapter 2

Abrupt trainability transitions

This chapter studies two training strategies: layer-wise training [62], and the identity initialization strategy [84], with special focus on the former. The chapter begins by recalling the notion of training by layer stacks, which generalizes layer-wise training. We then recall the problem of variational quantum compilation of k-Toffoli gates. Then, we define *abrupt trainability transitions*, which affect variational quantum compilation when using a layer-wise strategy. Numerical experiments where this effect can be observed are presented and discussed. The chapter concludes with the proof of the cost function used for this problem taking extrema when the ansatz circuit evaluates to identity, which has implications for the identity initialization strategy.

2.1 Background

The so-called *layer-wise trainability conjecture* appears in the variational quantum computing literature [62, 132]. The conjecture asserts that a circuit can be trained piece-wise, e.g. that a few layers can be trained to form the first block, then new blocks are added and trained to iteratively minimize a cost function. This strategy originated in classical machine learning, as a way to train neural networks [184–188]. Its most obvious advantage holds true for both quantum and classical algorithms, and is a reduction to the classical computational cost, since the processor only has to optimize a small subset of parameters at a time. Furthermore, it has been discovered

in [62] that training a quantum circuit layer-wise can mitigate barren plateaus.

In the same venue we have the identity initialization strategy. This is an easy to implement strategy aimed at mitigating BP [84], that works by initializing the parameters such that the ansatz evaluates to identity, which reduces randomness in the circuit and the likelihood of BP. Unfortunately, as with other learning strategies, most research has focused on their impact on BP, ignoring possible downsides of employing these strategies.

2.2 Training by layer stacks

Definition 17 (Training by layer stacks) *Training by layer stacks generalizes the layer-wise training, by considering the addition of more than a single ansatz layer at a time [1]. A layer stack S_j is a block composed of j layers of an ansatz V. After an initial stack is optimized, its parameters are fixed and the next stack of j layers is considered*

$$\begin{aligned} S_j^m(\boldsymbol{\theta}_m) &= \prod_{l=1}^{j} V_l^m(\boldsymbol{\theta}_l^m), \;\; \boldsymbol{\theta}_m = \bigcup_{l=1}^{j} \boldsymbol{\theta}_l^m \\ U(\boldsymbol{\theta}) &= \prod_{m=1}^{q} S_j^m(\boldsymbol{\theta}_m), \;\; \boldsymbol{\theta} = \bigcup_{m=1}^{q} \boldsymbol{\theta}_m \end{aligned} \tag{2.1}$$

where m corresponds to the stack index, and the total depth of the circuit is $p = q \cdot j$.

In general, performance increases with the addition of each new stack, although not always as it is later discussed. Figure 2-1 illustrates a circuit divided into stacks of 2 layers.

2.3 Compilation of k-Toffoli gates

We study layer-wise training in the context of compiling k-Toffoli gates, which appear prominently in traditional quantum algorithms [23, 100, 194], and error cor-

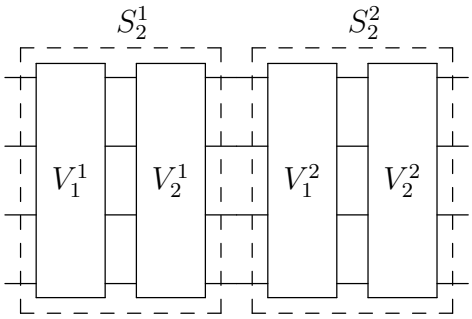


Figure 2-1: Circuit depicting 2 stacks of 2 layers. Upper indices in stacks indicate their position in the circuit, while lower indices indicates number of layers in the stack. Upper indices in layers indicate the position in the circuit of their stack, while lower index indicates position inside the stack. Figure reproduced from [1].

rection studies [195–197]. The k-Toffoli gate can be written as

$$X_b^{a_1 \ldots a_k} = \mathbb{1}_{2^n-2} \oplus X, \tag{2.2}$$

where the $\{a_i\}$ denotes control qubits and b indexes the target qubit.

We recall that the compilation process is performed by iteratively minimizing the distance (1.33)

$$d(U(\boldsymbol{\theta}), X_b^{a_1 \ldots a_k}) = 1 - \frac{1}{N} |(U(\boldsymbol{\theta}), T)_{\mathrm{HS}}| \in [0, 1]. \tag{2.3}$$

We note that the the distance (2.3) between a k-Toffoli gate and the identity is:

$$\begin{aligned} d(X_b^{a_1 \ldots a_k}, \mathbb{1}_{2^n}) &= 1 - \frac{\mathrm{Tr}\left(X_b^{a_1 \ldots a_k \dagger} \mathbb{1}_{2^n}\right)}{2^n} \\ &= 1 - \frac{\mathrm{Tr}(\mathbb{1}_{2^n-2} \oplus X)}{2^n} \\ &= 2^{1-n} = 2^{-k}. \end{aligned} \tag{2.4}$$

This will be used to infer from the cost when the optimization terminates near the identity.

The shrinking distance between the k-Toffoli and the identity under (2.3) may prompt the use of an alternative metric, after all, for an input state $|1\rangle^n$ T and $\mathbb{1}$ are easily distinguishable. This would be useful only in the case where U already agrees with T for the remaining $2^n - 1$ possible input bit strings. This problem is

not present when using (2.3) as it considers all possible inputs during compilation. Another proposal would be use operator norms, unfortunately there are some caveats to this: a) to our knowledge there is no easy circuit that could be used to calculate

$$\min_{\alpha\in\mathbb{R}} \|T - e^{i\alpha} iU(\boldsymbol{\theta})\|, \tag{2.5}$$

and b) as seen from (2.5) it would require additional optimization for α. For the previous reason we use (2.3) in this study.

2.4 Abrupt trainability transitions

In our study we empirically observe that variational quantum compiling can exhibit the limiting effect called *abrupt trainability transitions*.

Definition 18 (Abrupt trainability transitions (ATT)) *Let T be a target gate, $V_j^q = \prod_{m=1}^{q} S_j^m(\boldsymbol{\theta}_m)$ be a circuit trained with j layers per stack, and $d(V_j^q, T) > 0$. Variational quantum compilation experiences ATT if $\exists$ c such that:*

$$\underset{S_j^{q+1}(\boldsymbol{\theta}_{q+1})}{\operatorname{argmin}} \; d\left(V_j^q \cdot S_j^{q+1}(\boldsymbol{\theta}_{q+1}), T\right) = \mathbb{1}e^{i\alpha} \quad \textit{iff } j < c, \quad \alpha \in \mathbb{R}. \tag{2.6}$$

In other words, stacks would train close to the identity for a number of layers per stack $j < c$, but can be trained until the circuit is arbitrarily close to T for $j \geq c$.

2.4.1 Empirical observations

We train using layer stacks composed of layers of two commonly used ansatzes: the hardware efficient ansatz (HEA) (depicted in Figure 1-2) and the checkerboard ansatz (depicted in Figure 1-3). The number of layers per stack is iteratively increased, if the training using j layers per stack is not successful after q stacks, j is increased until the training succeeds. We train for up to 5 qubits, and up to 11 layers per stack of HEA using the L-BFGS-B optimizer [198]. We choose this gradient-based optimizer since we can use the parameter shift rule to easily calculate gradients [199], and to specify bounds for the periodic parameters. The obtained

results are results shown in Figure 2-2. We observe an ATT that depends on the target gate. We observe that bellow that critical point c, HEA stacks always train close to the identity, leading to a cost of $d(k\text{-Toffoli}, \mathbb{1}) = 2^{-k}$.

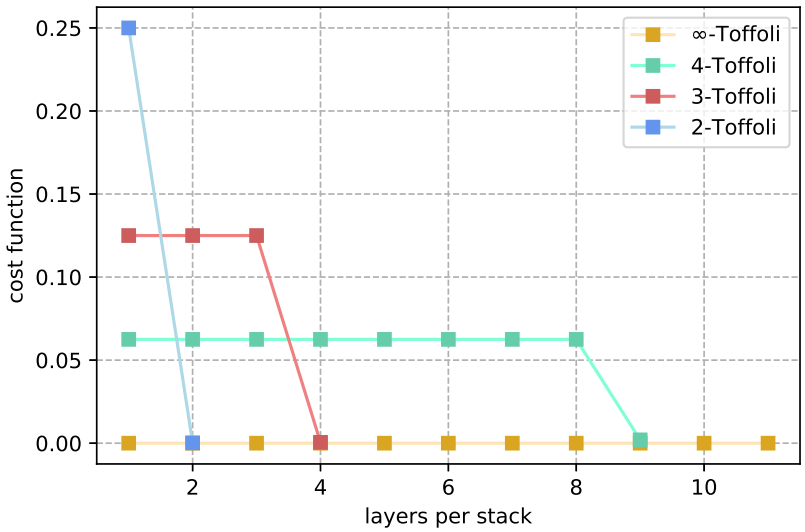


Figure 2-2: Trainability of k-Toffoli gates for multiple HEA layers per stack. Abrupt trainability transitions are observed at different number of layers per stack for different values of k, e.g. the green plot corresponds to the 4-Toffoli and illustrates abrupt trainability transitions occurring at 9 layers per stack. Figure taken from [1].

ATT also appear when using the checkerboard ansatz, albeit in a slightly different fashion as in HEA. Figure 2-3 illustrates the training process for a 2-Toffoli gate. When using stacks of 1 or 2 layers, training stops after a single stack, as the following stacks approximate an identity. This implies the identity is not the global minimum for the first layer stack (as it is the case for HEA), yet it becomes the global minimum following a stack that trained. It is only after $c \geq 3$ we observe training away from the identity with each new stack.

We also find that despite training with stacks of $j \geq c$ layers, it is still common to find the optimization to return an identity. Avoiding the identity extrema becomes more difficult as the number of qubits increases. This stems from the distance between the identity and the k-Toffoli decreasing exponentially with k as $d = 2^{-k}$.

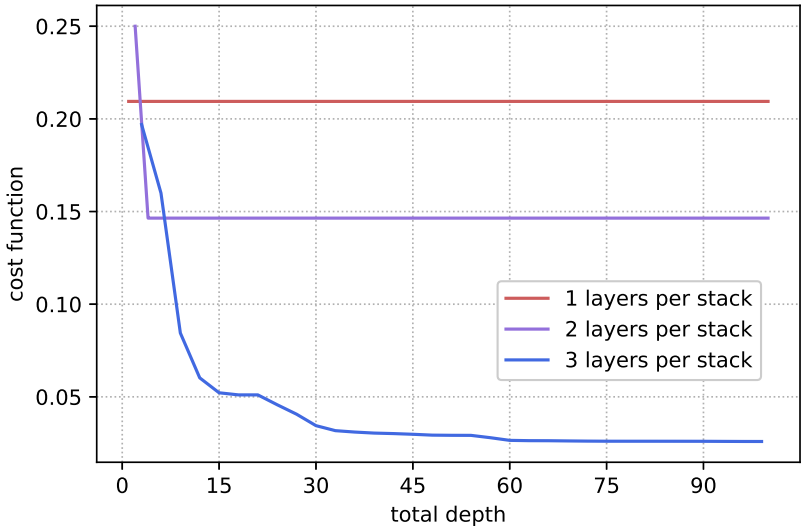


Figure 2-3: Trainability of a 2-Toffoli gate using 1-, 2- and 3-layer stacks of the checkerboard ansatz. Single and 2 layer stacks observe no improvement after the first stack. In contrast, steady improvement is observed with every additional 3-layer stack until minimizing the cost function. Figure taken from [1].

2.4.2 Minima of the cost function

We proved that for both HEA and checkerboard, $d(U(\boldsymbol{\theta}), k\text{-Toffoli})$ takes extrema when the parameters satisfy

$$U(\boldsymbol{\theta}_{\mathbb{1}}) - e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})}\mathbb{1} = 0, \tag{2.7}$$

where $\alpha(\boldsymbol{\theta}_{\mathbb{1}}) \in \mathbb{R}$. Therefore, the identity initialization strategy [62, 84] would prevent any training. We proved this by calculating the first and second derivative of the cost function (1.33) with respect to all parameters of single layer of both ansatzes. This results can be generalized for arbitrary depth and layer stack count as all layers besides the one being differentiated evaluate to identity. The cost function 1.33 can be expressed as

$$d(X_b^{a_1\ldots a_k}, U(\boldsymbol{\theta})) = 1 - \frac{1}{N}\sqrt{\mathrm{Tr}(X_b^{a_1\ldots a_k\dagger}U(\boldsymbol{\theta}))^* \,\mathrm{Tr}(X_b^{a_1\ldots a_k\dagger}U(\boldsymbol{\theta}))}\,, \tag{2.8}$$

where $*$ is the complex conjugate. The first derivative of the cost function is then

$$\begin{aligned}\partial_j d(X_b^{a_1\cdots a_k}, U(\boldsymbol{\theta})) = &- \frac{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\ \partial_j U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))}{2N\sqrt{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))}} \\ &- \frac{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\ \partial_j U(\boldsymbol{\theta}))}{2N\sqrt{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))}} \\ = &- \frac{\mathrm{Re}\left\{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\ \partial_j U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))\right\}}{N\sqrt{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))}},\end{aligned} \tag{2.9}$$

where $\partial_j = \frac{\partial}{\partial\theta_j}$. The second derivative is

$$\begin{aligned}\partial_j^2 d(X_b^{a_1\cdots a_k}, U(\boldsymbol{\theta})) = &\frac{-\mathrm{Re}\left\{\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\ \partial_j^2 U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))\right\} - |\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\ \partial_j U(\boldsymbol{\theta}))|^2}{N\left[\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))\right]^{1/2}} \\ &+ \frac{|\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger}\partial_j U(\boldsymbol{\theta}))\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))|^2}{N\left[\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))^*\,\mathrm{Tr}(X_b^{a_1\cdots a_k\dagger} U(\boldsymbol{\theta}))\right]^{3/2}}.\end{aligned} \tag{2.10}$$

The partial derivative of $U(\boldsymbol{\theta})$ with respect to a single qubit parameter $\boldsymbol{\theta}_j$ can be calculated as follows

$$\begin{aligned}U(\boldsymbol{\theta}) &= \cdots e^{-i\sigma\theta_j}\cdots \\ &= \cdots R\sigma(\theta_j)\cdots, \\ \partial_j U(\boldsymbol{\theta}) &= \cdots R\sigma(\theta_j)(-i\sigma)\cdots,\end{aligned} \tag{2.11}$$

where σ is a Pauli matrix, and $\cdots$ represents the gates to the left and right of $R\sigma(\theta_j)$. If there are no gates, $\cdots$ represents an identity.

Harwdare efficient ansatz

Theorem 2 (Extrema of the HEA) *For a single layer of the HEA, $d(X_b^{a_1\cdots a_k}, V_{hea}(\boldsymbol{\theta}))$ takes extrema for the parameters that satisfy*

$$V_{\mathrm{hea}}(\boldsymbol{\theta}_{\mathbb{1}}) - e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})}\mathbb{1} = 0, \tag{2.12}$$

where $\alpha(\boldsymbol{\theta}_{\mathbb{1}}) \in \mathbb{R}$.

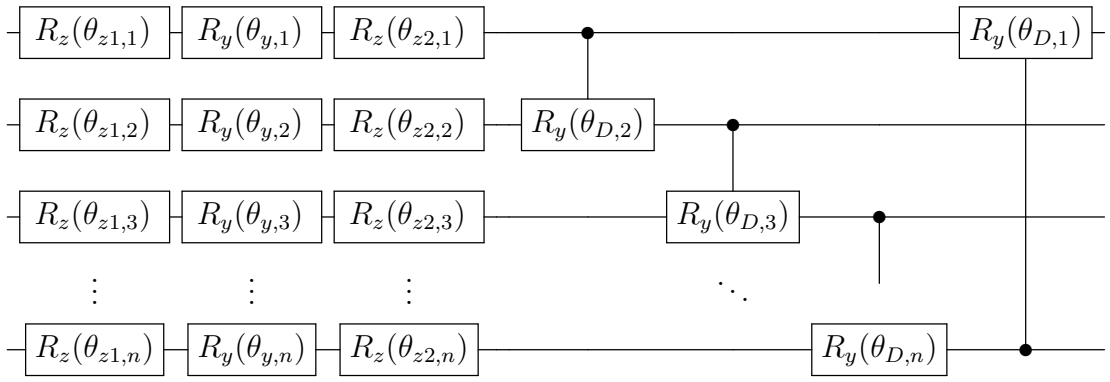


Figure 2-4: Single layer of the hardware efficient ansatz. Figure reproduced from [10].

Proof. We prove $V_{\text{hea}}(\boldsymbol{\theta}_{\mathbb{1}})$ are extrema by showing

$$\begin{aligned} \partial_j d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} &= 0, \\ \partial_j^2 d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} &\geq 0. \end{aligned} \tag{2.13}$$

We use m to refer to qubit indices and X_m, Y_m, Z_m as Pauli gates applied to the $m-$th qubit. Following the notation form Figure 2-4, the partial derivative of $V_{\text{hea}}(\boldsymbol{\theta})$ with respect to the $\theta_{z1,m}$ parameters is

$$\partial_{z1,m} V_{\text{hea}}(\boldsymbol{\theta}) = V_{\text{hea}}(\boldsymbol{\theta})(-iZ_m). \tag{2.14}$$

Evaluating for $\boldsymbol{\theta}_{\mathbb{1}}$ we obtain

$$\partial_{z1,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = -ie^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} Z_m. \tag{2.15}$$

From (2.15) and (2.9), $\partial_{z1,m} d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = 0$. As for the second derivative

$$\partial_{z1,m}^2 V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = -e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \mathbb{1} \tag{2.16}$$

$$\partial_{z1,m}^2 d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} > \ 0, \tag{2.17}$$

which indicates this is a minimum. Similarly, for the $\theta_{z2,m}$ parameter,

$$\partial_{z2,m} V_{\text{hea}}(\boldsymbol{\theta}) = (-iZ_m) V_{\text{hea}}(\boldsymbol{\theta}) \tag{2.18}$$

$$\partial_{z1,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = -ie^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} Z_m. \tag{2.19}$$

From (2.19) and (2.9), $\partial_{z2,m} d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = 0$. Calculating the second derivative we obtain

$$\partial^2_{z2,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = -e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \mathbb{1} \tag{2.20}$$

$$\partial^2_{z2,m} d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} > 0. \tag{2.21}$$

The derivative of $V_{\text{hea}}(\boldsymbol{\theta})$ with respect to a $\theta_{y,m}$ parameter is

$$\partial_{y,m} V_{\text{hea}}(\boldsymbol{\theta}) = \cdots R_z(\theta_{z2,m})(-iY_m) R_y(\theta_{y,m}) R_z(\theta_{z1,m}). \tag{2.22}$$

Evaluating for $\boldsymbol{\theta}_{\mathbb{1}}$ and using (2.12),

$$\begin{aligned} \partial_{y,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} &= e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} R_z^\dagger(\theta_{z1,m})(-iY_m) R_z(\theta_{z1,m}) \\ &= e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \begin{pmatrix} 0 & -e^{i(-2\theta_{z1,m})} \\ e^{i(2\theta_{z1,m})} & 0 \end{pmatrix}. \end{aligned} \tag{2.23}$$

From (2.23) and (2.9), $\partial_{y,m} d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = 0$. The second derivative is

$$\partial^2_{y,m} V_{\text{hea}}(\boldsymbol{\theta}) = \cdots R_z(\theta_{z2,m})(-iY_m)^2 R_y(\theta_{y,m}) R_Z(\theta_{z1,m}) \tag{2.24}$$

$$\partial^2_{y,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = -e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \mathbb{1}. \tag{2.25}$$

Substituting (2.25) in (2.10) gives $\partial^2_{y,m} d(X_b^{a_1 \dots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} > 0$.

The CR_y gate can be written as

$$CR_y(\theta_{D,m}) = e^{-i(\underline{\mathbf{0}}_2 \oplus Y)\theta_{D,m}}$$

where $\underline{\mathbf{0}}_2$ is a 2×2 zeros matrix, and $\oplus$ is the direct sum. The partial derivative of $V_{\text{hea}}(\boldsymbol{\theta})$ with respect to a controlled gate is

$$\partial_{D,m} V_{\text{hea}}(\boldsymbol{\theta}) = (\underline{\mathbf{0}}_2 \oplus (-iY_m)) V_{\text{hea}}(\boldsymbol{\theta}). \tag{2.26}$$

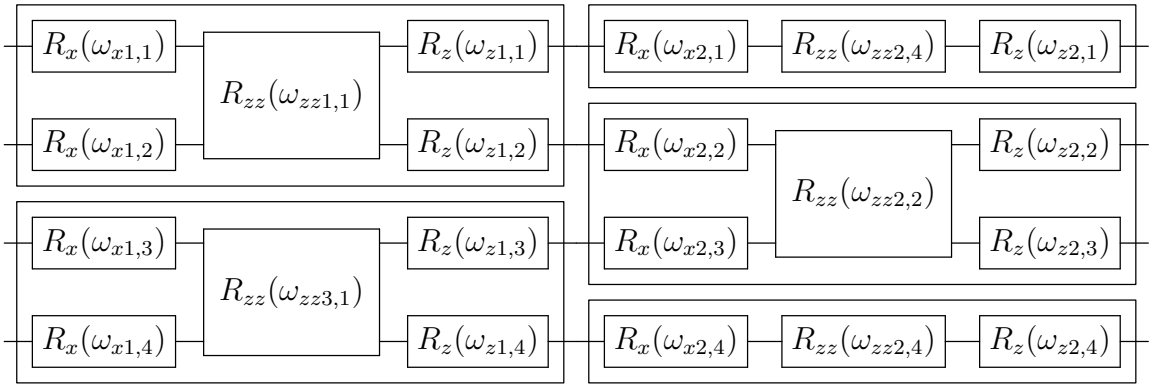


Figure 2-5: Single layer of the checkerboard ansatz for 3 qubits. Figure reproduced from [14].

Evaluating for $\boldsymbol{\theta}_{\mathbb{1}}$

$$\partial_{D,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \underline{\mathbf{0}}_2 \oplus (-iY_m). \tag{2.27}$$

From (2.27) and (2.9) $\partial_{D,m} d(X_b^{a_1 \ldots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = 0$. The second derivative of $CR_y(\theta)$ is

$$\partial^2_{D,m} V_{\text{hea}}(\boldsymbol{\theta}) = \ (\underline{\mathbf{0}}_2 \oplus (-i\mathbb{1}))V_{\text{hea}}(\boldsymbol{\theta}). \tag{2.28}$$

$$\partial^2_{D,m} V_{\text{hea}}(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} = e^{i\alpha(\boldsymbol{\theta}_{\mathbb{1}})} \underline{\mathbf{0}}_2 \oplus (-\mathbb{1}_2). \tag{2.29}$$

From (2.29), (2.10), $\partial^2_{D,m} d(X_b^{a_1 \ldots a_k}, V_{\text{hea}}(\boldsymbol{\theta}))|_{\boldsymbol{\theta}=\boldsymbol{\theta}_{\mathbb{1}}} > 0$. ■

Theorem 3 (Extrema of the checkerboard ansatz) *For a single layer of the checkerboard ansatz, $d(X_b^{a_1 \ldots a_k}, V_{ch}(\boldsymbol{\theta}))$ takes extrema for the roots of*

$$V_{\text{ch}}(\boldsymbol{\omega}_{\mathbb{1}}) - e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})}\mathbb{1} = 0. \tag{2.30}$$

Checkerboard ansatz

Proof. We prove $V_{\text{ch}}(\boldsymbol{\omega}_{\mathbb{1}})$ are extrema by showing

$$\begin{aligned} \partial_j d(V_{ch}(\boldsymbol{\omega}), X_b^{a_1 \ldots a_k})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} &= 0, \\ \partial_j^2 d(V_{ch}(\boldsymbol{\omega}), X_b^{a_1 \ldots a_k})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} &\geq 0. \end{aligned} \tag{2.31}$$

Following the notation from Figure 2-5 and starting from the left column of blocks, the derivative of $V_{\text{ch}}(\boldsymbol{\omega})$ with respect to the $\omega_{x1,m}$ parameter is

$$\partial_{x1,m} V_{\text{ch}}(\boldsymbol{\omega}) = \cdots R_x(\omega_{x1,m})(-iX_m), \tag{2.32}$$

$$\partial_{x1,m} V_{\text{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})}(-iX_m), \tag{2.33}$$

$$\partial_{x1,m} d(X_b^{a_1\ldots a_k}, V_{\text{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = 0. \tag{2.34}$$

The derivative of $V_{\text{ch}}(\boldsymbol{\omega})$ with respect to the $\omega_{z1,m}$ parameter is

$$\partial_{z1,m} V_{\text{ch}}(\boldsymbol{\omega}) = \cdots R_z(\omega_{z2,m}) R_{zz}(\omega_{zz2,m}) R_x(\omega_{x2,m}) R_z(\omega_{z1,m}) R_{zz}(\omega_{zz1,m})(-iZ_m) R_x(\omega_{x1,m}). \tag{2.35}$$

From (2.30) and evaluating for $\boldsymbol{\omega}_{\mathbb{1}}$,

$$\begin{aligned}\partial_{z1,m} V_{\text{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} =& e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})} R_x(\omega_{x1,m})^\dagger (-iZ_m) R_x(\omega_{x1,m}) \\ =& e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})} \begin{pmatrix} -i(\cos^2(\omega_{x1,m}) - \sin^2(\omega_{x1,m})) & -2\sin(\omega_{x1,m})\cos(\omega_{x1,m}) \\ 2\sin(\omega_{x1,m})\cos(\omega_{x1,m}) & i(\cos^2(\omega_{x1,m}) - \sin^2(\omega_{x1,m})) \end{pmatrix}.\end{aligned} \tag{2.36}$$

The matrix elements in opposite corners of (2.36) cancel when evaluating the distance derivative

$$\partial_{z1,m} d(X_b^{a_1\ldots a_k}, V_{\text{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = 0. \tag{2.37}$$

The derivative of $V_{\text{ch}}(\boldsymbol{\omega})$ with respect to the first $\omega_{zz1,m}$ parameter is

$$\partial_{zz1,m} V_{\text{ch}}(\boldsymbol{\omega}) = \cdots R_{zz}(\omega_{zz1,m})(-iZ_m \otimes Z_{m+1})(R_x(\omega_{x1,m}) \otimes R_x(\omega_{x1,m+1})). \tag{2.38}$$

From (2.30) and evaluating for $\boldsymbol{\omega}_{\mathbb{1}}$,

$$\begin{aligned}\partial_{zz1,m} V_{\text{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} =& e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})}(R_x(\omega_{x1,m}) \otimes R_x(\omega_{x1,m+1}))^\dagger(-iZ_m \otimes Z_{m+1})(R_x(\omega_{x1,m}) \otimes R_x(\omega_{x1,m+1})) \\ =& e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})} \begin{pmatrix} -i(\cos^2(\omega_{x1,m}) - \sin^2(\omega_{x1,m})) & -2\sin(\omega_{x1,m})\cos(\omega_{x1,m}) \\ 2\sin(\omega_{x1,m})\cos(\omega_{x1,m}) & i(\cos^2(\omega_{x1,m}) - \sin^2(\omega_{x1,m})) \end{pmatrix} \\ & \otimes \begin{pmatrix} -i(\cos^2(\omega_{x1,m+1}) - \sin^2(\omega_{x1,m+1})) & -2\sin(\omega_{x1,m+1})\cos(\omega_{x1,m+1}) \\ 2\sin(\omega_{x1,m+1})\cos(\omega_{x1,m+1}) & i(\cos^2(\omega_{x1,m+1}) - \sin^2(\omega_{x1,m+1})) \end{pmatrix}.\end{aligned} \tag{2.39}$$

Similar to (2.36), the elements in opposite corners of the matrices in (2.39) are similar but with opposite signs, which makes them cancel when evaluating the distance derivative

$$\partial_{zz1,m} d(X_b^{a_1 \dots a_k}, V_{\text{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = 0. \tag{2.40}$$

Moving to the second block from Figure 2-5, the derivative of $V_{\text{ch}}(\boldsymbol{\omega})$ with respect to the $\omega_{z2,m}$ parameter is

$$\partial_{z2,m} V_{\text{ch}}(\boldsymbol{\omega}) = (-iZ_m) V_{\text{ch}}(\boldsymbol{\omega}). \tag{2.41}$$

Evaluating for $\boldsymbol{\omega}_{\mathbb{1}}$,

$$\partial_{z2,m} V_{\text{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})}(-iZ_m), \tag{2.42}$$

$$\partial_{z2,m} d(X_b^{a_1 \dots a_k}, V_{\text{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} = 0. \tag{2.43}$$

The derivative of $V_{\text{ch}}(\boldsymbol{\omega})$ with respect to $\omega_{x2,m}$ is

$$\partial_{x2,m} V_{\text{ch}}(\boldsymbol{\omega}) = \cdots R_z(\omega_{z2,m}) R_{zz}(\omega_{zz2,m})(-iX_m) R_x(\omega_{x2,m}) R_z(\omega_{z1,m}) R_{zz}(\omega_{zz1,m}) R_x(\omega_{x1,m}). \tag{2.44}$$

From (2.30) and evaluating for $\boldsymbol{\omega}_{\mathbb{1}}$,

$$\begin{aligned}\partial_{x2,m} V_{\text{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}} =& e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})} R_z(\omega_{z2,m}) R_{zz}(\omega_{zz2,m})(-iX_m) R_{zz}(\omega_{zz2,m})^\dagger R_z(\omega_{z2,m})^\dagger \\ =& -ie^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})} R_z(2\omega_{z2,m}) R_{zz}(2\omega_{zz2,m}) X_m.\end{aligned} \tag{2.45}$$

The number $\mathrm{Tr}(X_b^{a_1\dots a_k\dagger}\ \partial_{x2,m}V_{\mathrm{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}})^*\,\mathrm{Tr}(X_b^{a_1\dots a_k\dagger}V_{\mathrm{ch}}(\boldsymbol{\omega}_{\mathbb{1}}))$ is purely complex, thus

$$\partial_{x2,m}d(X_b^{a_1\dots a_k},V_{\mathrm{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}}=0. \tag{2.46}$$

The derivative with respect to $\omega_{zz2,m}$ is

$$\partial_{zz2,m}V_{\mathrm{ch}}(\boldsymbol{\omega})=\cdots(-iZ_m\otimes Z_{m+1})V_{\mathrm{ch}}(\boldsymbol{\omega}). \tag{2.47}$$

Evaluating for $\boldsymbol{\omega}_{\mathbb{1}}$,

$$\begin{aligned}\partial_{zz2,m}V_{\mathrm{ch}}(\boldsymbol{\omega})|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}}&=e^{i\alpha(\boldsymbol{\omega}_{\mathbb{1}})}(-iZ_m\otimes Z_{m+1}),\\ \partial_{zz2,m}d(X_b^{a_1\dots a_k},V_{\mathrm{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}}&=0.\end{aligned} \tag{2.48}$$

The second derivative of $V_{\mathrm{ch}}(\boldsymbol{\omega})$ with respect to any parameter is

$$\partial_j^2V_{\mathrm{ch}}(\boldsymbol{\omega})=-\,V_{\mathrm{ch}}(\boldsymbol{\omega}), \tag{2.49}$$

and therefore $\partial_j^2d(X_b^{a_1\dots a_k},V_{\mathrm{ch}}(\boldsymbol{\omega}))|_{\boldsymbol{\omega}=\boldsymbol{\omega}_{\mathbb{1}}}>0$. ■

2.5 Discussion

We have proven the layer-wise trainability conjecture false by showing the existence of *abrupt trainability transitions*. For the cases studied, training layer-wise with less than a critical number of layers per stack prevented learning, as each stack would train close to an identity. The effect was observed in the HEA and checkerboard ansatzes when trying to approximate k-Toffoli gates, which have a 2^{-k} distance to the identity. The effect seems to affect ansatzes that can approximate an identity up to a global phase but not a target unitary close to the identity. We expect similar difficulties when compiling other k-controlled unitaries that share an distance $O(2^{-k})$ to the identity.

The findings depart from known results: various papers of the area studied similar scenarios but did not encounter ATT. For example, Skolik et al. [62] studied training by layer stacks as a strategy to avoid barren plateaus. The researchers did not

encounter ATT as their problem (MNIST classification) does not have the identity extrema in the cost function. The paper by Sharma et al. [47] studied quantum compilers noise resilience. We did not observe an ATT when training a Toffoli gate as the ansatz they used (checkerboard composed of dressed CNOTs) can not simulate an identity for a single layer.

Also, our results question the benefits of using the identity initialization strategy [62], or a similar initialization strategy where randomly selected gates are initialized as identities [84]. We prove such initialization would start training in a bad local minima.

Avoiding ATT may be possible with the introduction of new training strategies which would bring back the advantages of using layer-wise training. These strategies include: the use of ansatz that can not evaluate to an identity in a few layers, allow for non-monotonic improvement of the cost function with the addition of each stack, and the introduction of variability to the angles during the optimization. In Chapter 3 we study a training strategy similar to the latter to recover layer-wise training, albeit for a different problem. These results have influenced the development of new training techniques [200–202], which take in consideration the possibility of ATT and/or the presence of poor local minima around the identity.

Chapter 3

Training saturation in layer-wise QAOA

In this chapter we investigate layer-wise training in QAOA, as well as the impact of coherent phase noise on this strategy. The chapter starts by recalling the problem of unstructured search in QAOA. Then it defines the limiting effect we call *training saturation* [2], and numerically shows this effect for the problems MAX-CUT, MAX-k-SAT, and unstructured search. For the later we analytically obtain necessary conditions for a state to saturate. Lastly, the chapter presents strategies for avoiding training saturations. The first one consisting on limiting the performance of each individual layer, the second considers the effects of constant probability coherent phase noise, and a third one that considers changing probabilities of coherent phase noise.

3.1 Background

Recent results have shown that classical algorithms can outperform single layer QAOA [138], putting in question the usefulness of low depth QAOA. On the other hand, the use of higher depths comes with a higher optimization cost. An alternative would be to use a layer-wise strategy. As explained in Chapter 2, layer-wise training comes with its own limitations, still, it is a worth using training strategy that has shown success in multiple cases [62, 132, 203].

An aspect worth studying is the interaction of training techniques with noise. Several works have demonstrated certain noise resilience of variational algorithms [48–51, 151], while [98] demonstrated performance benefits that noise can induce for training a quantum autoencoder.

3.2 Unstructured search via QAOA

QAOA can be used for variational search: let $|\omega\rangle$ be the target state in the computational basis, the task is to variationally prepare a candidate state with a high overlap with $|\omega\rangle$. In QAOA, the candidate state $|\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle$—prepared by a $p-$depth circuit—is parametrized as:

$$|\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle = \prod_{k=1}^{p} e^{-i\beta_k H_x} e^{-i\gamma_k H} |+\rangle^{\otimes n}, \tag{3.1}$$

with $\gamma_k \in [0, 2\pi)$, $\beta_k \in [0, \pi)$. Here $H_x = \sum_{j=1}^{n} X_j$ is the standard one-body mixer Hamiltonian with Pauli matrix X_j applied to the j-th qubit, and $H = -|\omega\rangle\langle\omega|$ is the problem Hamiltonian. The optimization task minimizes the expectation value $\langle H \rangle$ which equivalently maximizes the overlap between the candidate state $|\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle$ and the target state $|\omega\rangle$ given by

$$0 \leq \max_{\boldsymbol{\gamma}, \boldsymbol{\beta}} |\langle\omega|\psi_p(\boldsymbol{\gamma}, \boldsymbol{\beta})\rangle|^2 \leq 1. \tag{3.2}$$

QAOA performance for unstructured search is insensitive to the particular target state $|\omega\rangle$ in the computational basis. For any target state $|\omega\rangle$ representing a binary string, there is a $U = U^\dagger$ composed of X and $\mathbb{1}$ operators such that $U|0\rangle^{\otimes n} = |\omega\rangle$. Using the facts that $[H_x, U] = 0$ and $U|+\rangle^{\otimes n} = |+\rangle^{\otimes n}$, we can calculate the overlap

with $|\omega\rangle$ of an arbitrary state prepared by a QAOA sequence as

$$\begin{aligned}
\langle\omega|\psi_p(\boldsymbol{\gamma},\boldsymbol{\beta})\rangle &= \langle\omega| \prod_{k=1}^{p} e^{-i\beta_k H_x} e^{i\gamma_k |\omega\rangle\langle\omega|} |+\rangle^{\otimes n} \\
&= \langle 0|^{\otimes n} U \prod_{k=1}^{p} e^{-i\beta_k H_x} e^{i\gamma_k U(|0\rangle\langle 0|)^{\otimes n} U} |+\rangle^{\otimes n} \\
&= \langle 0|^{\otimes n} U \prod_{k=1}^{p} e^{-i\beta_k H_x} U e^{i\gamma_k (|0\rangle\langle 0|)^{\otimes n}} U |+\rangle^{\otimes n} \\
&= \langle 0|^{\otimes n} \prod_{k=1}^{p} e^{-i\beta_k H_x} e^{i\gamma_k (|0\rangle\langle 0|)^{\otimes n}} |+\rangle^{\otimes n} ,
\end{aligned}$$

which is independent of ω.

The calculation of $\langle H\rangle$ does not require the decomposition of $H = |0\rangle\langle\omega| \, \omega$ into an exponential number of Pauli strings, as it is equivalent to the probability of measuring $|\omega\rangle$. It is worth noting that this optimization based approach to unstructured search is unpractical, as calculating the expectation value would in average require an exponential number of measurements for $\boldsymbol{\gamma}, \boldsymbol{\beta}$ chosen at random. Nonetheless, this problem allows for a greater degree of analysis compared to others, and the results obtained could be extrapolated to more practical applications.

3.3 Training saturation

Definition 19 (Training saturation (TS)) *Training saturates for depth p^*, the smallest depth for which*

$$\langle\psi_{p^*+1}| H |\psi_{p^*+1}\rangle \geq \langle\psi_{p^*}| H |\psi_{p^*}\rangle > \langle\psi_g| H |\psi_g\rangle , \tag{3.3}$$

where $|\psi_g\rangle$ is in the ground space of H, and the circuit is trained layer-wise.

In other words, training saturates when an extra layer (and thus all subsequent layers) does not allow for an improvement.

The numerical experiments in this section are performed using the gradient-free COBYLA optimizer [204]. It was chosen primarily due to the complicated parameter shift rule required to calculate the gradients of QAOA [199]. Additionally,

in our experiments in the presence of noise, when compared to L-BFGS-B with finite differences [198], COBYLA is able to find similarly good solutions with less function evaluations.

3.3.1 Saturation in unstructured search

In the setting of unstructured search, TS can be expressed in terms of overlap with the target state as

$$|\langle\omega|\psi_{p^*+1}\rangle|^2 \leq |\langle\omega|\psi_{p^*}\rangle|^2 < 1. \tag{3.4}$$

Our results demonstrate layer-wise training always saturates for this problem, which limits this method's performance compared to globally optimized QAOA. Specifically we observed an n-dependent threshold depth p^* beyond which overlap cannot be improved further by optimizing additional layers. The induced saturation is also evident as the overlap for any new layer achieves optimality at trivial values of variational parameter β_p, leaving the circuit output state invariant when optimizing subsequent layers. We find that saturation always occurs from the threshold depth $p^* = n$ (see Fig. 3-1).

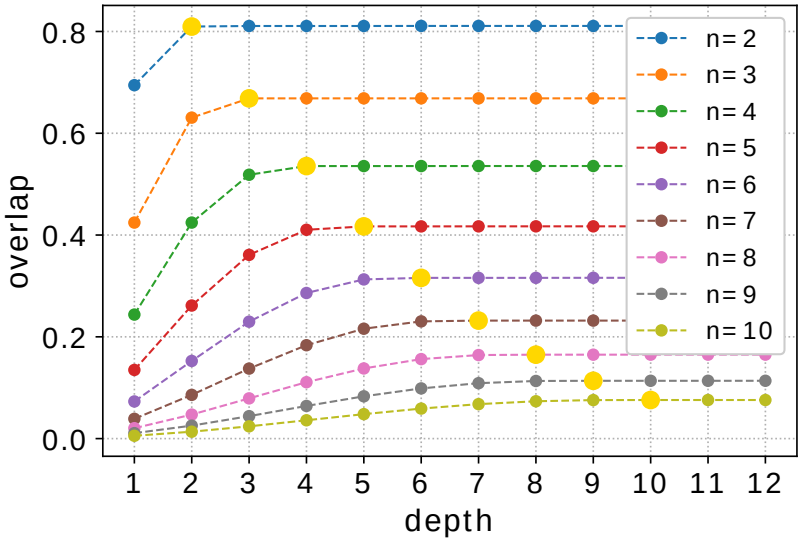


Figure 3-1: Saturation in layer-wise training for unstructured search. Saturation occurs at $p^* = n$ as indicated by the larger markers. Figure taken from [2].

Given the onset, this result appears to severely limit layer-wise training as an approach towards alleviating the tedious computational cost suffered in global opti-

mization.

Being a greedy algorithm, layer-wise training, in contrast to global training, maximizes overlap increase from layer to layer. Therefore, we observe that for a set of globally optimized parameters $\{\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*\}$ and a set of layer-wise optimized parameters $\{\boldsymbol{\gamma}^\#, \boldsymbol{\beta}^\#\}$, there exists a certain intermediate depth c:

$$|\langle\omega|\psi_c(\boldsymbol{\gamma}^*, \boldsymbol{\beta}^*)\rangle|^2 \leq |\langle\omega|\psi_c(\boldsymbol{\gamma}^\#, \boldsymbol{\beta}^\#)\rangle|^2, \tag{3.5}$$

Note that the resulting overlap in global training is always higher or equal than in layer-wise. It is only at some intermediate depth of the already trained circuit that layer-wise overlap can be higher than global optimization, as seen from Eq. 3.5. From Fig. 3-2 we can observe that greedy gains from layer-wise training stagnate, which results in global optimization having a higher final overlap.

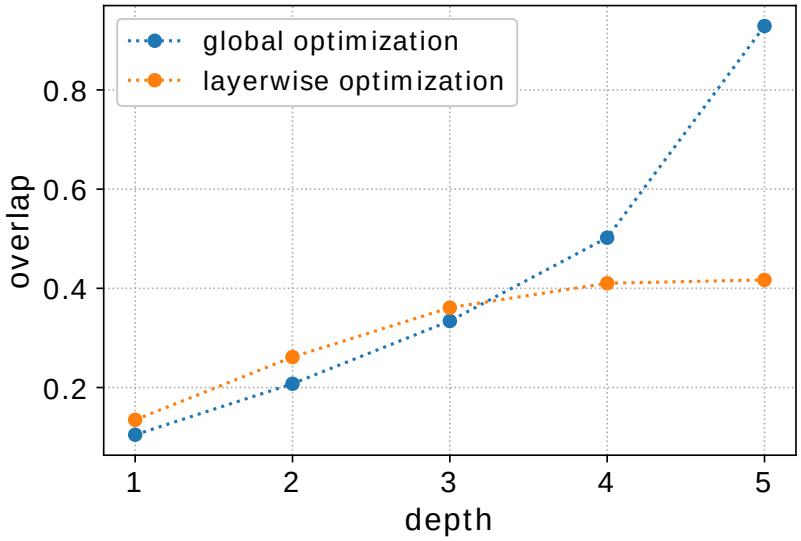


Figure 3-2: Overlap comparison between layer-wise and global training as a function of layers. Global optimization aims at overall overlap improvement, whereas the layer-wise approach maximally increases overlap from layer to layer. Figure taken from [2].

Necessary conditions

To understand the structure of candidate states prepared by layer-wise optimisation and to derive conditions necessary for saturation we recall the following:

Definition 20 (Symmetric subspace) $H_s = Span\{|\psi\rangle \mid \forall\, 1 \le j < k \le n, P_{jk}|\psi\rangle = |\psi\rangle\}$, *where* P_{jk} *is a permutation of a qubit pair* j *and* k.

Definition 21 (Dicke basis vectors) $|e_k\rangle = \frac{1}{\sqrt{C_k^n}} \sum_{z_1+\cdots+z_n=k} |z_1 \cdots z_n\rangle$, *where* $z_i \in \{0,1\}$, *and* C_k^n *are binomial coefficients.*

Dicke state $|e_k\rangle$ is a uniform superposition of all n-qubit states with Hamming weight k, (a.k.a. 1-norm k). These vectors form a $n+1$ dimensional orthonormal basis of H_s. Symmetric subspace H_s plays a key role in our consideration: since H_x commutes with any qubit permutation P_{jk} and $|0\rangle^{\otimes n} \equiv |e_0\rangle, |+\rangle \in H_s$, then all QAOA ansatz states prepared as (3.1) with $|\omega\rangle = |0\rangle^{\otimes n}$ belong to the symmetric subspace. Note that for a general target state $|\omega\rangle$ all the ansatz states still belong to $n+1$ dimensional subspace, isomorphic to the symmetric one, therefore the following analysis is also applicable.

Let us denote a state prepared with optimized c-depth QAOA circuit as $|\psi_c\rangle \equiv \left|\psi_c(\boldsymbol{\gamma}^{\#}, \boldsymbol{\beta}^{\#})\right\rangle$. The ansatz states of different depth circuits are related recursively as $|\psi_c\rangle = e^{-i\beta_c H_x} e^{i\gamma_c |\omega\rangle\langle\omega|} |\psi_{c-1}\rangle$ with the initial state $|\psi_0\rangle = |+\rangle^{\otimes n}$. Since $|\psi_c\rangle \in H_s$, one can expand over the Dicke basis to obtain $n+1$ components $A_k^c = \langle e_k|\psi_c\rangle$.

The optimization task of layer-wise training is to maximize the overlap at each depth c. This overlap can be written in terms of wave function coefficients A_k^{c-1} of the previous layer:

$$
\begin{aligned}
A_0^c = \langle e_0|\psi_c\rangle &= \langle e_0| e^{-i\beta_c H_x} e^{i\gamma_c |e_0\rangle\langle e_0|} |\psi_{c-1}\rangle \\
&= \langle e_0| e^{-i\beta_c H_x} ((e^{i\gamma_c} - 1)|e_0\rangle\langle e_0| + \mathbb{1}) |\psi_{c-1}\rangle \\
&= \langle e_0| e^{-i\beta_c H_x} |e_0\rangle (e^{i\gamma_c} - 1) A_0^{c-1} + \langle e_0| e^{-i\beta_c H_x} |\psi_{c-1}\rangle \\
&= \cos^n \beta_c (e^{i\gamma_c} - 1) A_0^{c-1} + \langle e_0| e^{-i\beta_c H_x} |\psi_{c-1}\rangle \\
&= \cos^n \beta_c (e^{i\gamma_c} - 1) A_0^{c-1} + \\
&+ \sum_{k=0}^{n} (\cos\beta_c)^{n-k} (-i \sin\beta_c)^k \sqrt{C_n^k} \langle e_k| \sum_{l=0}^{n} A_l^{c-1} |e_l\rangle \\
&= A_0^{c-1} \cos^n \beta_c e^{-i\gamma_c} + \\
&+ \sum_{k=1}^{n} (\cos\beta_c)^{n-k} (-i \sin\beta_c)^k A_k^{c-1} \sqrt{C_n^k}.
\end{aligned} \tag{3.6}
$$

All the coefficients A_k^{c-1} are functions of $2(c-1)$ variational parameters of the previous layers. In layer-wise optimization these values are already fixed from previous training iterations, thus leaving A_0^c to be a function of only current layer parameters β_c and γ_c.

Optimization of expression (3.6) over γ_c can be performed using a geometrical fact that for any complex numbers A and B, $\max_{\gamma} |Ae^{i\gamma} + B| = |A| + |B|$:

$$g(\beta) \equiv \max_{\gamma_c} |A_0^c| = \cos^n \beta_c \Big(|A_0^{c-1}| + \Big| \sum_{k=1}^{n} (-i \tan \beta_c)^k A_k^{c-1} \sqrt{C_n^k} \Big| \Big). \tag{3.7}$$

Expression (3.7) should be supplemented with the iterative expression for the amplitudes A_k^{c-1}. One should then calculate the action of the mixer $e^{-i\beta H_x}$ on the Dicke basis vectors, which implies representation of the Hadamard transformation in the Dicke basis. However, the result of this procedure is messy and further solution requires a consideration of the general optimization problem.

As derived by Rabinovich in [2], expression (3.7) allows to identify necessary conditions for saturation at some depth. Assuming that saturation already happened at depth c we study the structure of the output state (coefficients A_k^{c-1}). Saturation implies that the circuit output state cannot be trained further and optimization of additional layer parameters only returns trivial values of β. Thus, (3.7) should have a global maximum at position $\beta_c = 0$.

Proposition 1 (Necessary conditions for saturation) *A saturated state $|\psi\rangle \in H_s$, i.e. its overlap with the target state $|0\rangle^{\otimes n}$ can not be increased by applying a QAOA layer, necessarily satisfies*

$$A_1 = \langle e_1|\psi\rangle \propto \langle 0|^{\otimes n} H_x |\psi\rangle \propto \langle 0|^{\otimes n} X |\psi\rangle = 0, \tag{3.8}$$

$$|A_2| = |\langle e_2|\psi\rangle| \leq \sqrt{\frac{n}{2(n-1)}} |\langle 0|^{\otimes n} |\psi\rangle|. \tag{3.9}$$

Proof. We prove by calculating the first and second derivatives of equation (3.7) and use standard conditions for a maximum, $g'(0) = 0$ and $g''(0) \leq 0$. The first

derivative of $g(\beta)$ evaluated at 0 is,

$$g'(0) = 0 = \sqrt{C_n^1}|A_1| \tag{3.10}$$

$$\implies A_1 = \langle e_1|\psi\rangle \propto \langle 0|^{\otimes n} H_x |\psi\rangle \propto \langle 0|^{\otimes n} X |\psi\rangle = 0, \tag{3.11}$$

which is condition (3.8). The second derivative of $g(\beta)$ evaluated at 0 is,

$$g''(0) = -n|A_0| + |2A_2\sqrt{C_n^2}| \leq 0$$

$$\implies |A_2| \leq |\langle e_2|\psi\rangle| \leq \sqrt{\frac{n}{2(n-1)}}|\langle 0|^{\otimes n}|\psi\rangle|, \tag{3.12}$$

which is condition (3.9). □

Note that the formulated conditions are necessary for observing saturation, but in principle may not be sufficient. We elaborate more on this in the Discussion.

Physically, (3.8) and (3.9) imply that states $|e_1\rangle$ and $|e_2\rangle$ (if their amplitudes in $|\psi\rangle$ are large enough) are the source of state $|0\rangle^{\otimes n}$ under the action of QAOA: probability flows from states $|e_{1,2}\rangle$ to the target state $|0\rangle^{\otimes n}$. If training saturates, this sources should be drained, as formulated in conditions (3.8) and (3.9).

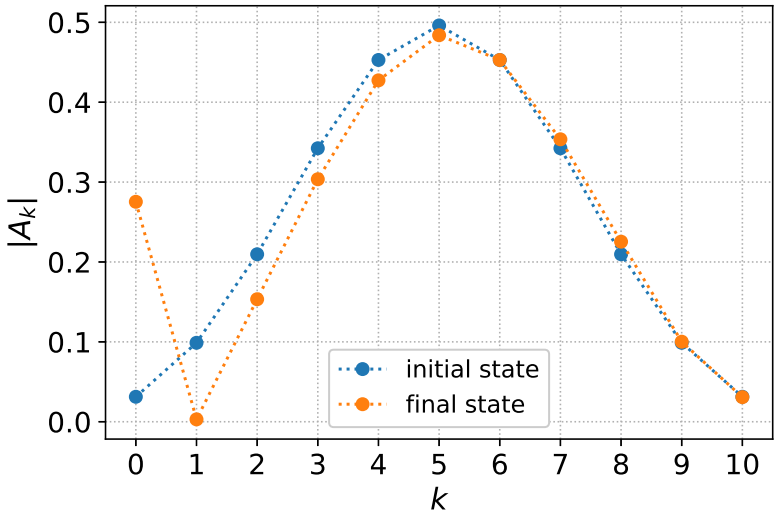


Figure 3-3: Comparison between the coefficients A_k of the initial state $|+\rangle^{\otimes n}$ (blue markers) and the layer-wise QAOA output state (orange markers), for $n = p = 10$ and $k = 0, \ldots, n$. Numerically we confirm the output state is saturated and we observe agreement with the necessary conditions from Proposition 1: $|A_1 = 0|$, and $|A_2|$ is somewhat small compared to $|A_0|$. Figure from [2].

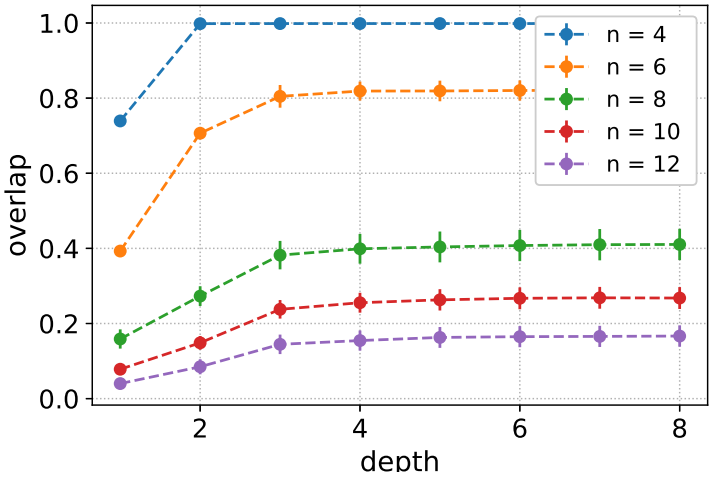


Figure 3-4: Saturation in layer-wise training for MAX-CUT. Data points illustrate the average overlap for 50 randomly generated 3 regular graphs. The spread of the vertical lines illustrate the standard error.

Fig. 3-3 illustrates that when one observes saturation, the circuit output state indeed satisfies necessary saturation conditions (3.8) and (3.9), i.e. $A_1 = 0$, and $|A_2|$ is somewhat small compared to $|A_0|$. If at least one of conditions (3.8) and (3.9) is violated, QAOA would manage to improve state overlap with $|0\rangle^{\otimes n}$. Thus, avoiding states that satisfy (3.8) and (3.9) is key to avoid saturation. We discuss the ways to violate these conditions and to assist training in the next section.

3.3.2 Saturation in combinatorial optimization problems

Saturation is also found in combinatorial optimization problems. We numerically observe this effect in MAX-CUT and MAX-2-SAT (details on this problems in Appendix A.2). For MAX-CUT, we calculate the overlap or the ansatz state with the ground space of 50 randomly generated 3 regular graphs from 4 to 12 qubits. The average overlap and standard error are illustrated in figure 3-4.

The decision of illustrating overlap instead of expected value comes from the variability in the energy spectrum of the randomly generated Hamiltonians, which can complicate the visualization of the saturations. The variability that comes from sampling random graphs also makes harder to determine the saturation depths for a given number of qubits. Similarly, for MAX-2-SAT we calculate overlaps with

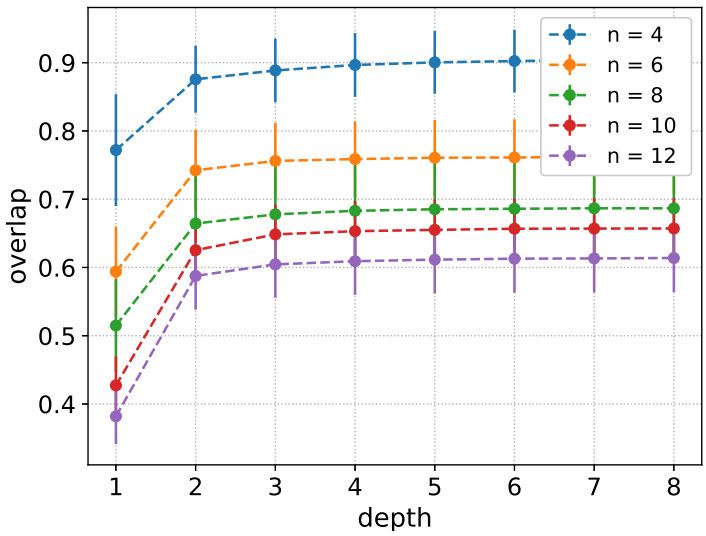


Figure 3-5: Saturation in layer-wise training for MAX-2-SAT. Data points illustrate the average overlap for 50 randomly generated 2-SAT instances of m clauses. The spread of the vertical lines illustrate the standard error.

the ground state of instances of 4 to 12 qubits with a clause to variable ratio of 1. Averages and standard errors of these numerical experiments are depicted in figure 3-5 where the saturations are clearly observed.

Unfortunately, neither necessary nor sufficient conditions were found for these combinatorial optimization problems. This is due to a lack of structure in these problems, making the analysis more difficult compared to unstructured search

3.4 Avoiding saturation

Saturation strongly limits the performance of layer-wise optimization. In unstructured search, it results from the particular structure that the greedy layer-wise algorithm imposes on coefficients A_k. Deviating from this greedy strategy by violating conditions (3.8) and (3.9) and thus altering the structure imposed, one might avoid saturation and thereby improve training. The simplest approach would be to push amplitude A_1 up, for example, by perturbing amplitudes $A_1, ..., A_n$. This will violate condition (3.8), thus non-zero amplitude A_1 might further be transferred to $|0\rangle^{\otimes n}$, removing saturation at the current layer. A similar strategy can be devised for

MAX-2SAT and MAX-CUT, where slight variations in training may prevent falling into saturated states.

3.4.1 Introducing coherent phase noise

Layer-wise optimization may not preserve candidate states in the symmetric subspace H_s when systematic errors (e.g. phase noise or bit-flipping) are considered. Here, (3.8) and (3.9) are no longer applicable and therefore saturation may be avoided in unstructured search. On one hand low probability coherent phase noise could reduce the overlap by just a fraction, thus allowing further training. But on the other hand, bit flip noise would radically alter the state amplitudes thus hampering training. Indeed these are confirmed by our numerical simulations.

We layer-wise train QAOA circuits of $p = n$ with a gradient free optimizer in the presence of coherent phase noise described as:

$$S(\phi) = |0\rangle\langle 0| + e^{i\phi} |1\rangle\langle 1| , \tag{3.13}$$

where ϕ is sampled from a normal distribution centered around 0 with variance 1. The noise is applied with a probability P to any qubit after each gate application. Experimentally, this type of noise engineering has been used to test the limits of variational algorithms in identifying and quantifying quantum phase transitions with noisy qubits [205, 206]. In Fig. 3-6 we illustrate the overlaps obtained in 100 trials at each probability of applying noise for $n = 4, 5, 6, 7$ and $p = n$. The shaded region represents the top 10% of the trials, where the curves are polynomial fits for the best, average and worst of the top 10% overlaps. For a trial emulation of the algorithm, we can expect a 10% probability of obtaining an overlap inside the red shaded area, which for some values of P are higher than the overlap obtained by training layer-wise in an ideal system as illustrated by the dotted green line.

The use of a constant probability of noise is not optimal to obtain the maximum possible overlap, but it is enough to show that its presence can offer an advantage over noiseless systems in certain scenarios. A more systematic approach to introducing noise to the system can result in a higher overlap on average. For example, we

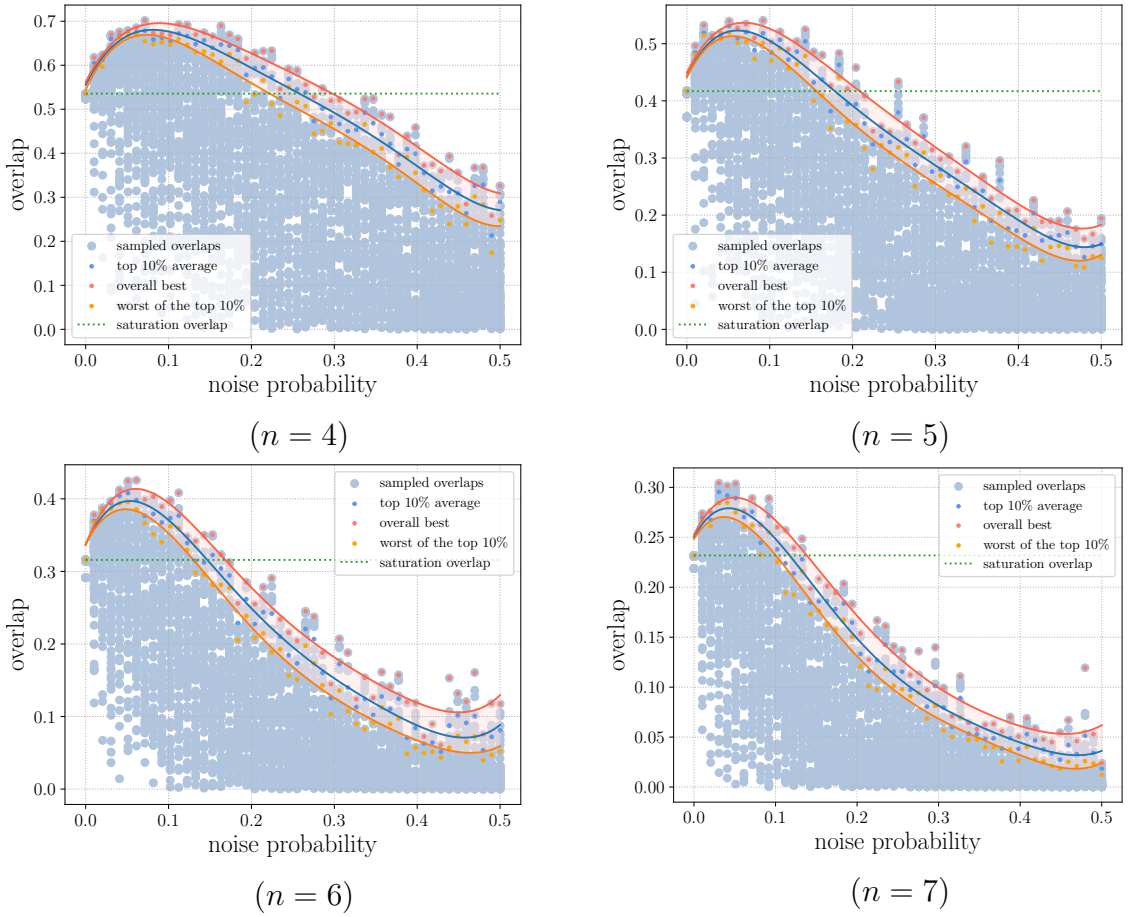


Figure 3-6: Overlaps of unstructured search via layer-wise QAOA in the presence of phase noise. Figure illustrates 100 trials per each noise probability for instances of $n = 4, 5, 6, 7$ and $p = n$. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. For a certain probability of noise, there is a 10% probability the overlap will fall inside the shaded pink region, considerably higher than the noiseless overlap illustrated as a dotted green line. Figure modified from [2].

take inspiration from *simulated annealing* (SA) [207, 208] a probabilistic technique that emulates the physical process of heating and slowly cooling a system to find its minimum energy configuration. We propose a strategy where the probability of noise depends on a "temperature"

$$T = 1 - p_{\text{current}}/p_{\text{final}}, \tag{3.14}$$

where p_{current} is the current depth, and p_{final} is the final depth of the sequence. Thus

$T \in [0, 1]$ decays as the circuit grows. We set the probability P of noise to be

$$P = \exp(-r/T), \tag{3.15}$$

where $r \in \mathbb{R}_+$ parameterizes the rate at which noise probability decays.

Figure 3-7 illustrates overlap of 100 trials versus the rate r for $n = 4, 5, 6, 7$ and $p = n$. We observe a more consistent improvement compared to using a constant probability of noise as in Figure 3-6. Moreover, performance can further be increased with the addition of more layers since we are no longer limited by saturation at $p = n$. Figure 3-8 illustrates overlaps of 100 trials versus the probability of applying noise for $n = 4, 5, 6, 7$ and $p = 2n$.

We run a similar set of numerical experiments for the combinatorial optimization problems. Figures 3-9 and 3-10 illustrate the performance of this strategy for a random instance of each problem. In both cases the depth used was the saturation depth of each particular instance. Notice that in these cases the plots depict expected value instead of overlap, therefore lower values indicate better performance.

We observe performance gains, at least for these sample instances, smaller than in unstructured search. This is in part due to noiseless layer-wise having better performance for these problems than for unstructured search. Performance increases considerably when using the SA inspired training strategy as depicted in Figures 3-11 and 3-12.

3.5 Discussion

The formulated necessary saturation conditions (3.8) and (3.9), in general, may not be sufficient for observing saturation since the amplitudes: $A_3, ..., A_n$ also are a source for A_0. In other words, the necessary saturation conditions ensure that overlap function (3.7) has a maximum at $\beta_c = 0$, which still may just be a local maximum. The absolute maximum for layer-wise training, found by the optimizer, may require $\beta \neq 0$, thus could not be captured by conditions (3.8) and (3.9).

However, based on the values of optimal parameters obtained numerically (Fig.

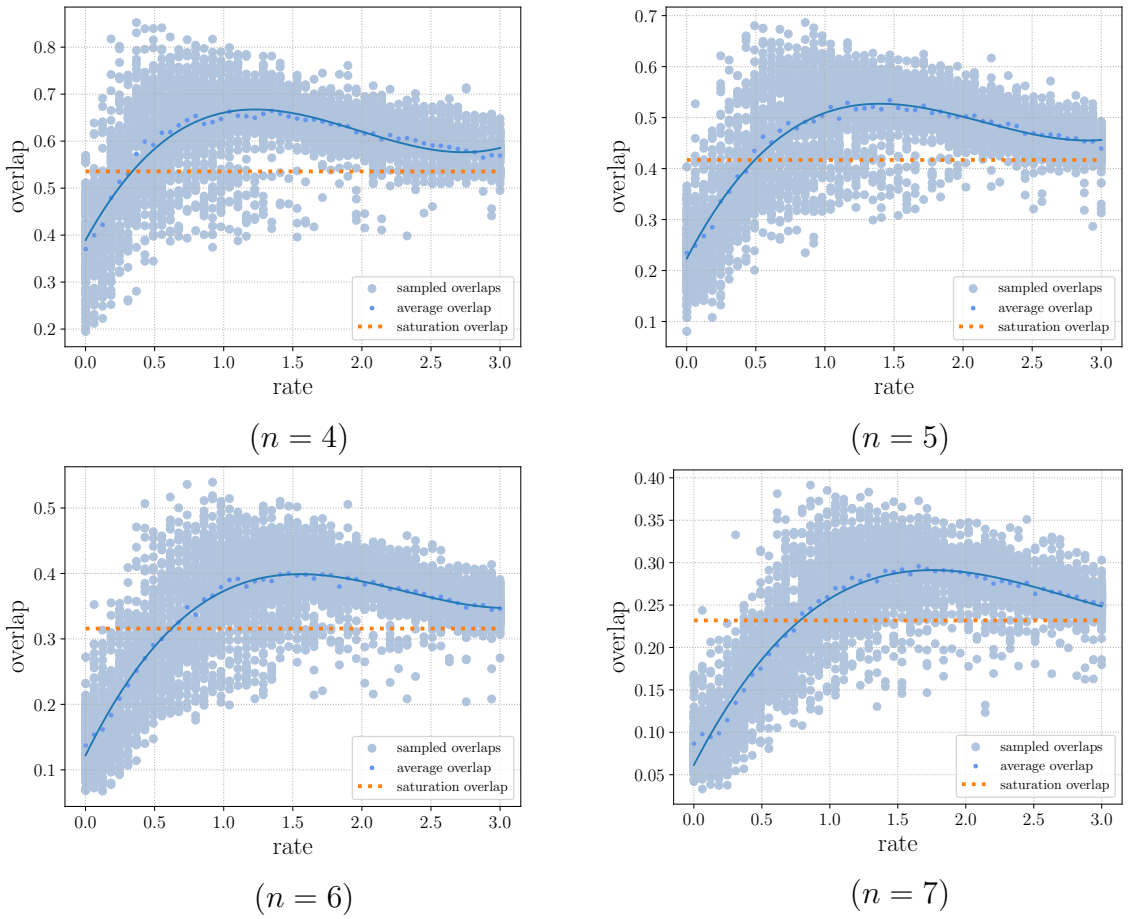


Figure 3-7: Overlaps of unstructured search via layer-wise QAOA in the presence of phase noise with probability decaying with rate r as in (3.15). Figure illustrates the overlaps of 100 trials per each value of r for instances of $n = 4, 5, 6, 7$ and $p = n$. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. There is notable improvement in performance and consistency when compared to using constant probability of noise as in Figure 3-6.

3-13) for large n we see that optimal βs are small. As a result, global maximum of the overlap shifts towards smaller values of β, and it finally stagnates at $\beta = 0$ (causing saturation). Thus we conclude that the formulated conditions for maximum overlap at $\beta_c = 0$ (3.8) and (3.9) are not only necessary for observing saturation, but also sufficient provided that the optimal angles are small.

For large n it can be understood as follows: It was shown in [5] that first layer optimal parameter $\beta_1 \approx \frac{\pi}{n}$ is small. That implies, the first layer marginally affects amplitudes A_k for large k, leaving the $|+\rangle^{\otimes n}$ state almost unmodified up to a global phase. Therefore, optimal parameters at subsequent layers also remain small, which again preserve $|+\rangle^{\otimes n}$ state. Thus in the output state only the lowest amplitudes are

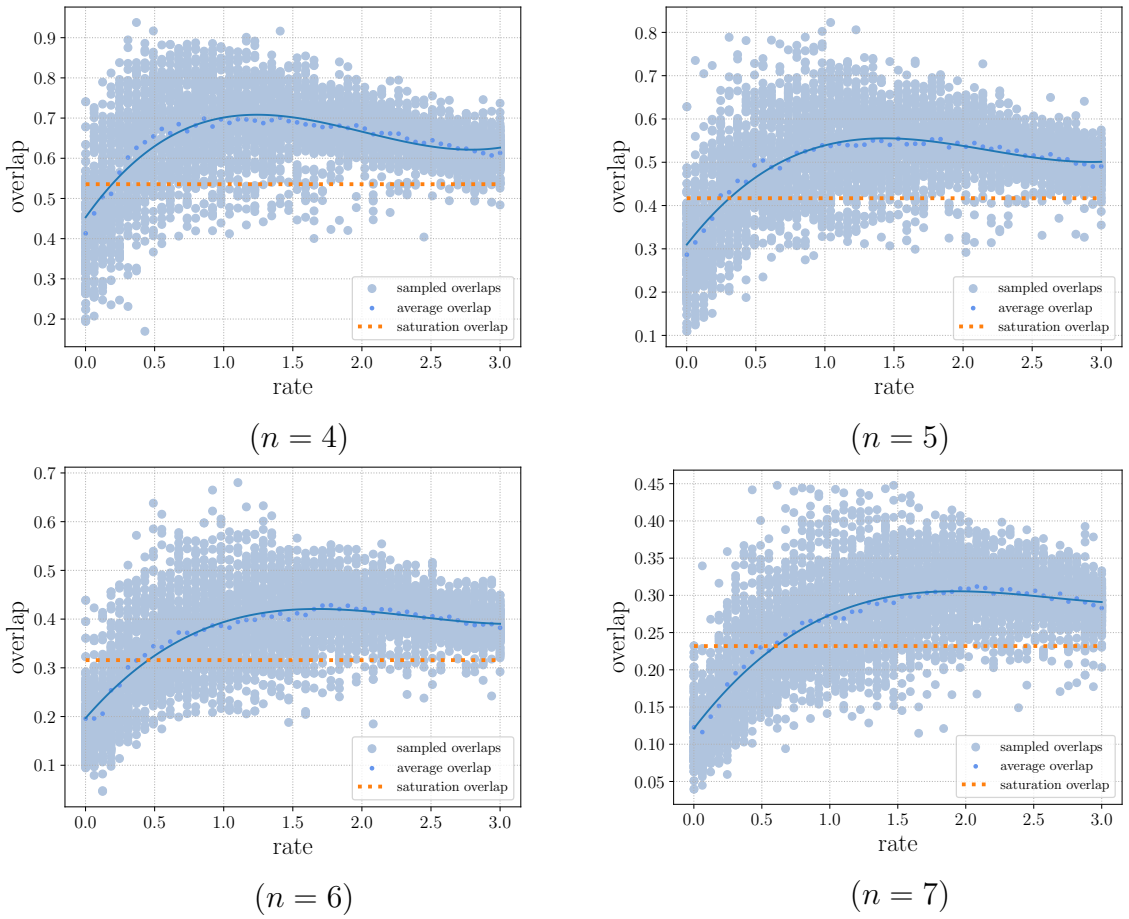


Figure 3-8: Overlaps of unstructured search via layer-wise QAOA in the presence of phase noise with probability decaying with rate r as in (3.15). Figure illustrates the overlaps of 100 trials per each value of r for instances of $n = 4, 5, 6, 7$ and $p = 2n$. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. There is notable improvement in performance with respect to Figure3-6 and 3-7, as we are no longer limited by saturation at $p = n$.

modified, while the others stay almost the same, as illustrated in Fig. 3-3. In other words, layer-wise QAOA manages to modify only slightly the initial state, keeping the large amplitudes almost preserved, causing it to saturate.

The analysis presented also allows to show the existence of several families of states for which layer-wise QAOA does not allow for overlap improvement. Expression (3.7) allows to show that in the symmetric subspace there exist states different from $|0\rangle^{\otimes n}$ that cannot be trained by QAOA to improve the overlap with $|0\rangle^{\otimes n}$. For instance, if one tries to implement QAOA on the state

$$|\psi\rangle = A_0 |0\rangle^{\otimes n} + A_2 |e_2\rangle , \tag{3.16}$$

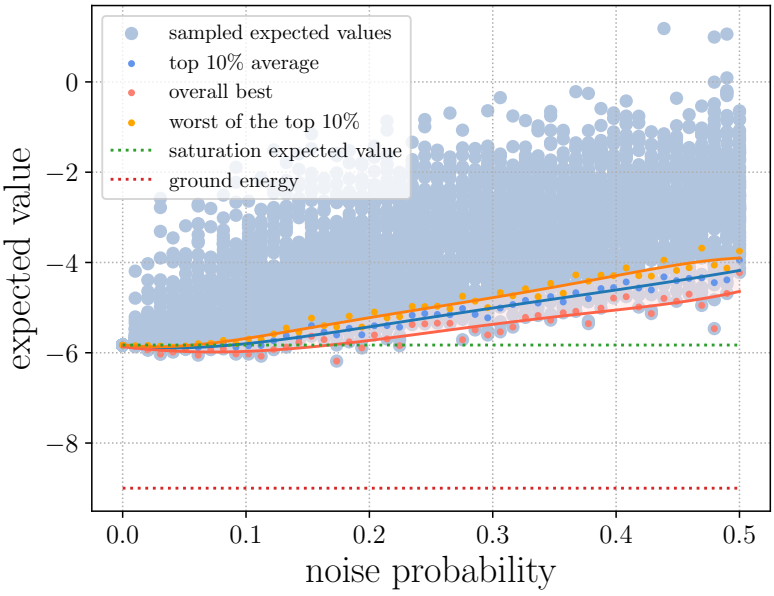


Figure 3-9: Expected values of layer-wise QAOA in the presence of phase noise for MAX-CUT of a randomly generated 3 regular graph of $n = 6$ nodes. Figure illustrates the expected values of 100 trials per each noise probability for a circuit depth set to $p = 6$, the saturation depth. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. For a certain probability of noise, a run of the algorithm exhibits a 10% probability of an expected value inside the shaded pink region.

with $|A_2| \leq \dfrac{|A_0|}{\sqrt{C_n^2}}$, the resulting overlap (3.7) would be

$$g(\beta) = (|A_0| - \sqrt{C_n^2}|A_2|)\cos^n \beta + \sqrt{C_n^2}|A_2|\cos^{n-2}\beta, \tag{3.17}$$

with maximum at $\beta = 0$. Therefore, QAOA is not able to improve overlap of state (3.16) with $|0\rangle$. The observed saturation effect implies that layer-wise QAOA necessarily reaches one of these non-trainable states.

Multiple strategies can be employed in order to avoid non-trainable states in QAOA, including training multiple layers per stack and retraining previous layers, at the cost of longer optimization times. The perturbation to the saturated state has to be able to violate (3.8) and/or (3.9) but not big enough to loose the amplitude A_0 that accumulates with every layer, illustrated in Fig. 3-3. Noise of the form $e^{-i\sigma\phi}$, where σ is a Pauli matrix and $\langle\phi\rangle = 0$, makes small changes to the inner product

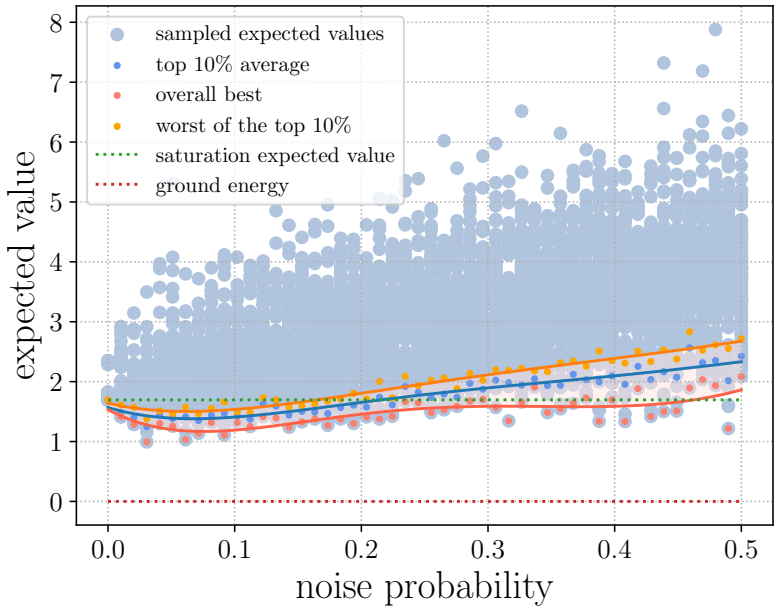
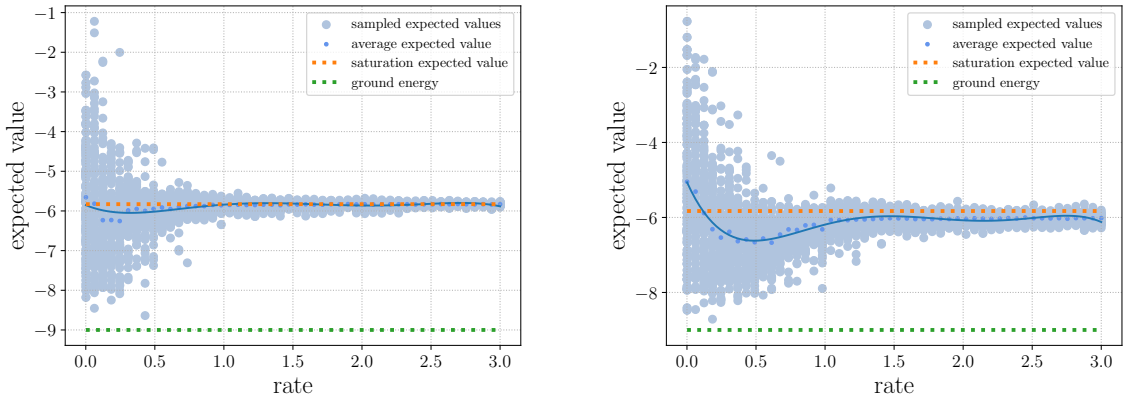
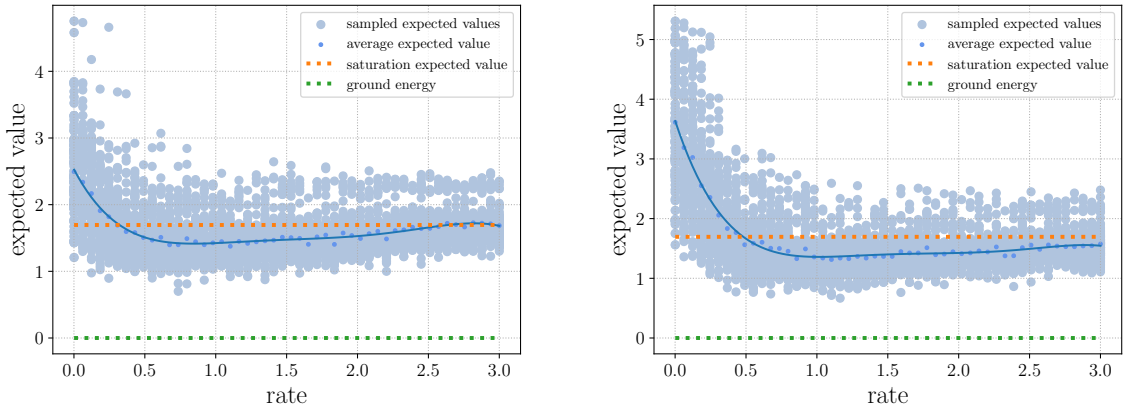


Figure 3-10: Expected values of layer-wise QAOA in the presence of phase noise for a randomly generated 2-SAT instance of $n = 6$ variables and $m = 24$ clauses. Figure illustrates the expected values of 100 trials per each noise probability for a circuit depth set to $p = 6$, the saturation depth. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. For a certain probability of noise, a run of the algorithm exhibits a 10% probability of an expected value inside the shaded pink region.



($p = 6$) ($p = 12$)

Figure 3-11: Expected values of MAX-CUT for a randomly generated graph via layer-wise QAOA in the presence of phase noise with probability decaying with rate r as in (3.15) for $p = 6, 12$. Figure illustrate the expected values of 100 trials per each value of r for a random 3 regular graph of $n = 6$ vertices. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. There is notable improvement in performance and consistency when compared to using constant probability of noise as in Figure 3-9.



$(p = 6)$ $(p = 12)$

Figure 3-12: Expected values for a randomly generated 2-SAT instance via layer-wise QAOA in the presence of phase noise with probability decaying with rate r as in (3.15) for $p = 6, 12$. Figure illustrate the expected values of 100 trials per each value of r for a random 2-SAT instance of $n = 6$ variables and $m = 24$ clauses. The phase noise angle is sampled from a normal distribution centered at 0 with variance 1. There is notable improvement in performance and consistency when compared to using constant noise probability as in Figure 3-10.

for a state $|\psi\rangle$

$$\langle\psi| e^{-i\sigma\phi} |\psi\rangle \approx \langle\psi| \left(1 - i\sigma\phi - \frac{\phi^2}{2}\right) |\psi\rangle = 1 - \frac{\langle\phi^2\rangle}{2}. \tag{3.18}$$

In contrast, bit flip noise does not have a good representation as a series expansion, thus altering the structure of the state in a way that harms the training.

Our numerics on MAX-CUT and MAX-k-SAT instances suggest that TS is a commonly occurring effect of layer-wise training gin QAOA. It is also natural to wonder whether certain problems do not exhibit the effect.

Although we focused on the simplest noise model, our result is among the few that have the potential to be applied in a more realistic setting. Furthermore these results have influenced the development of new training strategies that aim to recover some of the advantages of layer-wise training [6, 209, 210].

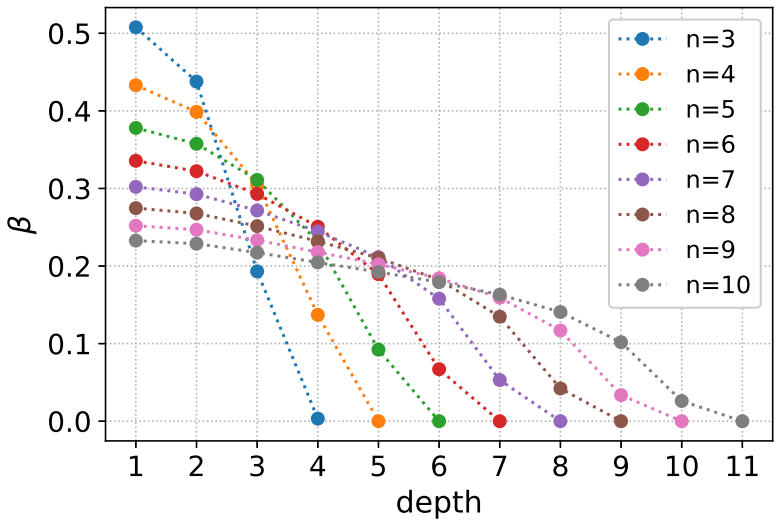


Figure 3-13: Optimal $\boldsymbol{\beta}$ for depths $p = n+1$. For each case, the final layer is affected by saturability, thus returning a trivial optimal angle for β_p. Figure taken from [2].

Chapter 4

Depth scaling of unstructured search via QAOA

This chapter studies the depth scaling of unstructured search via QAOA. To bound the optimal depth for this problem, we build on the known results of unstructured search in a continuous time quantum walk (CTQW), and recent advancements of on the theory of Trotter errors. The chapter starts by recalling unstructured search in a CTQW. Then, trotterizing the CTQW to obtain a QAOA sequence, we upper bound the depth of a QAOA sequence that prepares a state that approaches perfect overlap with the target. Our complexity bound exceeds the well known Grover complexity $\mathcal{O}(N^{\frac{1}{2}})$ but scales slower than $\mathcal{O}\left(N^{\frac{1}{2}+c}\right)$ for any $c > 0$. Finally, the chapter presents numerical experiments for up to 68 qubits that confirm the validity of our analysis.

4.1 Background

Among the promising uses of QAOA is its application to the problem of unstructured search, where the objective is to prepare a particular target state representing a bit string, as it was explained in Chapter 3. Notable results related to unstructured search via QAOA include: the discovery of parameter concentrations [5], optimal depth scaling when using Grover's mixer [211], near optimal depth scaling when using a quantum walk-based mixer [212], and near optimal depth scaling when using

the traditional mixer to prepare a state that approaches an overlap of $\frac{1}{2}$ with the target state [168]. Nevertheless, the depth scaling for approaching perfect overlap when using the traditional mixer, and how it compares to the complexity of Grover's algorithm [23] were unknown.

The problem of unstructured search has been studied in the context of many other models of quantum computing including adiabatic quantum computing [213] and continuous time quantum walks (CTQWs) [214]. The CTQW simulates the dynamics of a quantum system given a graph that describes the transitions between states. In this context, "continuous" refers to the fact that the transition probability from one state to another occurs continuously over time rather than at discrete steps as in a discrete time quantum walk [215]. Originally proposed by Farhi and Gutmann [216], it relates to a classical continuous time random walk by the analogy between a classical transfer matrix and a quantum Hamiltonian. CTQWs are known to provide an exponential speedup in certain problems [217], recover Grover's search [214, 218], and be a computationally universal model [219].

Continuous time evolution in CTQWs can be approximated by discretizing them into a quantum circuit with the use of product formulas, like the Suzuki-Trotter formulas [110, 111]. Notably, a CTQW solving unstructured search recovers a QAOA sequence upon this discretization.

Recently, a variation of QAOA incorporating quantum walks has been explored, referred to as quantum walk-based optimization algorithm (QWOA). The idea behind QWOA is to use a quantum walk instead of the standard mixer in QAOA. The quantum walk can explore the solution space more efficiently by utilizing the underlying graph structure of the problem. This enhances the exploration capabilities of QAOA, potentially offering performance advantages in terms of convergence speed and the quality of solutions. QWOA has been employed for a variety of problems including: continuous multi variable functions [174], bounded NP optimization problems [170], portfolio optimization [173], among others [171, 172, 220]. Notably, it has been proven that a QWOA using a complete identity interdependent network mixer scales similarly to Grover's algorithm while offering an advantage due to an easier to implement mixer [212].

4.2 Unstructured search via CTQW

A continuous time quantum walk is described by a Hamiltonian H which induces a unitary evolution, defined by the operator e^{-iHt}. Here Hamiltonian H is analogous to the transfer matrix of classical continuous time random walks.

In this setting, solving unstructured search in a hypercube graph consists of the evolution under the operator

$$U(\alpha, t) = e^{-i(\alpha H_x + H_\omega)t}, \tag{4.1}$$

where $H_x = \sum_{j=1}^{n} X_j$ is the adjacency matrix of the n-dimensional hypercube graph, with X_j being a Pauli X matrix applied to the j-th qubit; $H_\omega = |\omega\rangle\langle\omega|$ is the projector on the target state $|\omega\rangle \in \{|0\rangle, |1\rangle\}^{\otimes n}$, α is a real tunable parameter, and t is the evolution time. The objective is then to maximize the overlap

$$\max_{\alpha,t} |\langle\omega| U(\alpha, t) |+\rangle^{\otimes n}|^2. \tag{4.2}$$

Similar to unstructured search via QAOA (Section 3.2), we can see that (4.2) is target independent by substituting $|\omega\rangle = U_x |0\rangle^{\otimes n}$, where $U_x \in \{X, \mathbb{1}\}^{\otimes n}$. Therefore, without loss of generality we set $|\omega\rangle = |0\rangle^{\otimes n}$ and $H_\omega = H_0 = (|0\rangle\langle 0|)^{\otimes n}$. The optimal value of α was calculated by Farhi et al. [213] to be

$$\alpha^* = \frac{1}{2^n} \sum_{j=1}^{n} \frac{C_r^n}{r} = \frac{1}{n} + \mathcal{O}\left(\frac{1}{n^2}\right), \tag{4.3}$$

where C_r^n are binomial coefficients (details in Appendix B.1). The evolution $U(\alpha^*, t) |+\rangle^{\otimes n}$ largely occurs in the two-dimensional subspace spanned by the low-energy eigenstates of $\alpha^* H_x + H_0$ (see Appendix B.2). These two eigenstates are approximately given by

$$\begin{aligned} |\psi_+\rangle &= \frac{1}{\sqrt{2}}(|+\rangle^{\otimes n} + |0\rangle^{\otimes n}) + \mathcal{O}\left(\frac{1}{n}\right), \\ |\psi_-\rangle &= \frac{1}{\sqrt{2}}(|+\rangle^{\otimes n} - |0\rangle^{\otimes n}) + \mathcal{O}\left(\frac{1}{n}\right), \end{aligned} \tag{4.4}$$

with an energy gap $\Delta = \frac{2}{\sqrt{2^n}}\left(1 + \mathcal{O}\left(\frac{1}{n}\right)\right)$ [221] (details in Appendix B.2). Thus, up to a global phase we can approximate the evolution as

$$U(\alpha^*, t)\,|+\rangle^{\otimes n} = \frac{1}{\sqrt{2}}\left(|\psi_+\rangle + e^{-i\Delta t}\,|\psi_-\rangle\right) + \mathcal{O}\left(\frac{1}{n}\right), \tag{4.5}$$

allowing one to establish that for $t^* = \frac{\pi}{2}\sqrt{2^n}$ the overlap becomes

$$|\,\langle 0|^{\otimes n}\,U(\alpha^*, t^*)\,|+\rangle^{\otimes n}\,|^2 = 1 + \mathcal{O}\left(\frac{1}{n}\right). \tag{4.6}$$

4.3 Depth scaling of QAOA from a trotterized CTQW

We calculate the depth scaling of unstructured search via QAOA by using the known results of unstructured search in CTQW shown in Section 4.2, and relating both approaches through trotterization, as explained in Section 1.3. To approximate the optimal depth, we make use of Theorem 1, which relates the trotterization error to the Hamiltonians and the number of steps. The resulting depth complexity and its proof are shown in Theorem 4, while some of the auxiliary calculations are presented in Lemma 1.

Lemma 1 (Norm of nested commutators) *Let $H_0 = (|0\rangle\langle 0|)^{\otimes n}$ and $H_1 = H_x\alpha^*$, then the norm of their nested commutators is*

$$\|[H_{\mu_{q+1}}, \cdots [H_{\mu_2}, H_{\mu_1}]\cdots]\|_2 \leq (2\alpha^*\,(n+1))^j\,(n+1), \tag{4.7}$$

where $\{\mu_l\}_{l=1}^{q+1} \in \{0,1\}^{q+1}$, $\mu_1 \neq \mu_2$, and j is the number of times H_1 appears in the sequence.

Proof. We make use of the fact that in unstructured search the evolution happens in a symmetric subspace of dimension $n+1$, as explained in Section 3.3.1. In this subspace, the operator H_x is a tridiagonal matrix with diagonal elements equal to 0. The off-diagonal elements can be calculated from the action of H_x into a Dicke

state,

$$H_x |e_l\rangle = H_x \frac{1}{\sqrt{C_l^n}} \sum_{z_1+\cdots+z_n=l} |z_1 \cdots z_n\rangle \tag{4.8}$$

$$= \frac{l+1}{\sqrt{C_l^n}} \sum_{z_1+\cdots+z_n=l+1} |z_1 \cdots z_n\rangle + \frac{n-l+1}{\sqrt{C_l^n}} \sum_{z_1+\cdots+z_n=l-1} |z_1 \cdots z_n\rangle , \tag{4.9}$$

and then projecting into $|e_{l+1}\rangle$,

$$\langle e_{l+1}| H_x |e_l\rangle = \sqrt{(l+1)(n-l)} \leq \frac{n+1}{2}. \tag{4.10}$$

It follows from (4.10) that the max norm $\|A\|_{\max} = \max_{j,k} |A_{j,k}|$ of $H_1 = H_x \alpha^*$ can be upper bounded as

$$\|H_1\|_{\max} \leq \alpha^* \frac{n+1}{2}. \tag{4.11}$$

As for H_0, in the symmetric subspace it takes the form of an $n+1$ dimensional square matrix with all elements equal to 0 except $\langle e_0| H_0 |e_0\rangle = 1$.

Notice that for an arbitrary matrix A it follows that,

$$\|[H_0, A]\|_{\max} \leq \|A\|_{\max}, \tag{4.12}$$

$$\|[H_1, A]\|_{\max} \leq \max_{j,k} \left(\left(|A_{j+1,k}| + |A_{j-1,k}| + |A_{j,k+1}| + |A_{j,k-1}| \right) \left(\alpha^* \frac{n+1}{2} \right) \right)$$

$$\leq 2\alpha^*(n+1)\|A\|_{\max}. \tag{4.13}$$

Thus, in order to find an upper bound for the norm of nested commutators in (4.20) we focus only on how many times H_1 appears in the sequence. The largest matrix element resulting from any such sequence is bounded by

$$\|[H_{\mu_{q+1}}, \cdots [H_{\mu_2}, H_{\mu_1}] \cdots]\|_{\max} \leq (2\alpha^* (n+1))^j. \tag{4.14}$$

where $\mu_1 \neq \mu_2$, and j being the number of times H_1 appears in the sequence.

We recall the inequality between the spectral and Frobenius norm $\|\cdot\|_{\mathrm{F}}$ [222] as,

$$\|A\|_2 = \sigma_{\max}(A) \leq \|A\|_{\mathrm{F}} = \left(\sum_{j=1}^{m}\sum_{k=1}^{n} |A_{j,k}|^2\right)^{\frac{1}{2}}, \tag{4.15}$$

where $\sigma_{\max}(A)$ is the largest singular value of A. Using (4.14) and (4.15) we find the following expression for the spectral norm of nested commutators

$$\|[H_{\mu_{q+1}}, \cdots [H_{\mu_2}, H_{\mu_1}]\cdots]\|_2 \leq (2\alpha^* (n+1))^j (n+1), \tag{4.16}$$

where $\mu_1 \neq \mu_2$, and j is the number of times H_1 appears in the sequence. ■

Theorem 4 (Complexity of search via QAOA) *A QAOA circuit of depth*

$$p = \mathcal{O}\left(2^{\frac{n}{2}+\sqrt{n}\sqrt{2\cdot\log_2 5}}\right) \tag{4.17}$$

can prepare a state that satisfies $|\langle\omega|\psi_p(\boldsymbol{\gamma},\boldsymbol{\beta})\rangle|^2 = 1 + \mathcal{O}(1/n)$.

Proof. The evolution operator $U(\alpha^*, t^*)$ can be approximated by a qth order product formula with r steps of the form

$$S_q^r(t^*/r) = \prod_{v=1}^{r\Upsilon_q} e^{-it^* b(v) H_1} e^{-it^* a(v) H_0}, \tag{4.18}$$

where $a(v), b(v) \in \mathbb{R}$, $H_1 = H_x\alpha^*$, and $\Upsilon_q = 5^{q/2-1}$ for even q (details in Appendix B.3). Equation (4.18) gives a QAOA sequence of depth $p = r\Upsilon_q$. The objective is to calculate the depth that results in an approximation with an error at most ϵ. This can be done by using equation (1.26),

$$\|U(t) - S_q^r(t/r)\|_2 \leq \epsilon = \frac{2\Upsilon^{q+1}\delta(t^*)^{q+1}}{r^q(q+1)}, \tag{4.19}$$

where,

$$\delta = \sum_{\mu_1,\mu_2,\cdots,\mu_{q+1}=1}^{M} \|[H_{\mu_{q+1}}, \cdots [H_{\mu_2}, H_{\mu_1}]\cdots]\|_2, \tag{4.20}$$

and solving for p,

$$p = \frac{\Upsilon_q^{2+1/q}(2\delta)^{1/q}(t^*)^{1+1/q}}{(\epsilon(q+1))^{1/q}}. \tag{4.21}$$

From Lemma 1, we substitute (4.16) into δ (4.20) and obtain

$$\delta \leq 2(n+1)\sum_{j=1}^{q}(2\alpha^*(n+1))^j\, C_{j-1}^{q-1} \tag{4.22}$$

$$\leq 2(n+1)\,(2\alpha^*(n+1)+1)^q). \tag{4.23}$$

The binomial coefficients in (4.22) come from the number of combinations in which H_1 can appear $j-1$ times outside the first commutator. An extra factor of 2 appears from the first commutator being either $[H_0, H_1]$ or $[H_1, H_0]$.

After substituting $\Upsilon_q = 5^{q/2-1}$, $t^* = \frac{\pi}{2}\sqrt{2^n}$, and (4.23) into (4.21), we derive

$$p \leq p_0\sqrt{2^n}\left(\frac{2\pi(n+1)\sqrt{2^n}}{5\epsilon}\right)^{\frac{1}{q}} 5^q, \tag{4.24}$$

where

$$p_0 = \frac{\pi\,(2\alpha^*(n+1)+1)}{2\cdot 5^{3/2}(q+1)^{\frac{1}{q}}}. \tag{4.25}$$

To minimize (4.24) we take its derivative and set it to zero. Neglecting a small negative contribution, we find the value of q that results in the shortest depth to be

$$q \leq \left(\frac{n\cdot\ln\sqrt{2} + \ln(2\pi(n+1)) - \ln 5\epsilon}{\ln 5}\right)^{\frac{1}{2}}. \tag{4.26}$$

Substituting (4.26) into (4.24) and simplifying we arrive at

$$p \leq p_0\left(\frac{2\pi(n+1)}{5\epsilon}\right)^{\frac{2}{q}} 2^{\frac{n}{2}+\sqrt{n}\sqrt{2\log_2 5}}. \tag{4.27}$$

Importantly, from (4.25) and (4.3) one can show that $p_0 < \frac{1}{2}$.Moreover, since the CTQW is limited to an overlap $1+\mathcal{O}(1/n)$, as given by (4.6), errors smaller than $\epsilon = \mathcal{O}(1/n)$ do not meaningfully alter the overlap of QAOA. Therefore, for errors of this order and large n, the second factor in (4.27) does not grow with n. This leaves the last factor in (4.27) as the dominant contribution, establishing (4.17). ■

Remark 4 (Complexity comparison) *Theorem 4 puts the query complexity of the derived QAOA circuit in between the traditional Grover's algorithm and classical search*

$$\mathcal{O}\left(N^{\frac{1}{2}}\right) < \mathcal{O}\left(N^{\frac{1}{2}} 2^{\sqrt{\log_2 N}\sqrt{2\cdot\log_2 5}}\right) < \mathcal{O}(N). \tag{4.28}$$

It is also worth noting that

$$\mathcal{O}\left(N^{\frac{1}{2}} 2^{\sqrt{\log_2 N}\sqrt{2\cdot\log_2 5}}\right) < \mathcal{O}\left(N^{\frac{1}{2}+c}\right), \tag{4.29}$$

for any $c > 0$.

The angles $\boldsymbol{\gamma}, \boldsymbol{\beta}$ corresponding to the sequence QAOA sequence,

$$\prod_{k=1}^{p} e^{-i\beta_k H_x} e^{-i\gamma_k H_0}, \tag{4.30}$$

can be recovered by expressing $S_q^r(t/r)$ as a product of second order terms by following (1.23) and recursively calculating the parameters t_k that correspond to each second order term as,

$$S_q^r(t/r) = \prod_{k=1}^{p} S_2(t_k). \tag{4.31}$$

Then, the QAOA angles can be recovered by

$$\beta_k = t_k,\ \gamma_{k\neq p} = \frac{t_k + t_{k+1}}{2},\ \gamma_p = \frac{t_p}{2}. \tag{4.32}$$

4.4 Numerical experiments

We begin testing our analytics by confirming that the resulting QAOA sequence has sufficient depth to precisely approximate the evolution of the corresponding CTQW. We compare (i) Grover search, (ii) QAOA from a trotterized CTQW with numerically calculated depth, and (iii) QAOA from a trotterized CTQW with analytically predicted depth. For these three cases, Figure 4-1 illustrates the overlaps increasing through the sequences for $\epsilon = 0.01$ and $n = 42, 46$. Here $n = 46$ is the largest system size with the optimal order $q = 6$, allowing for the numerical estimation of

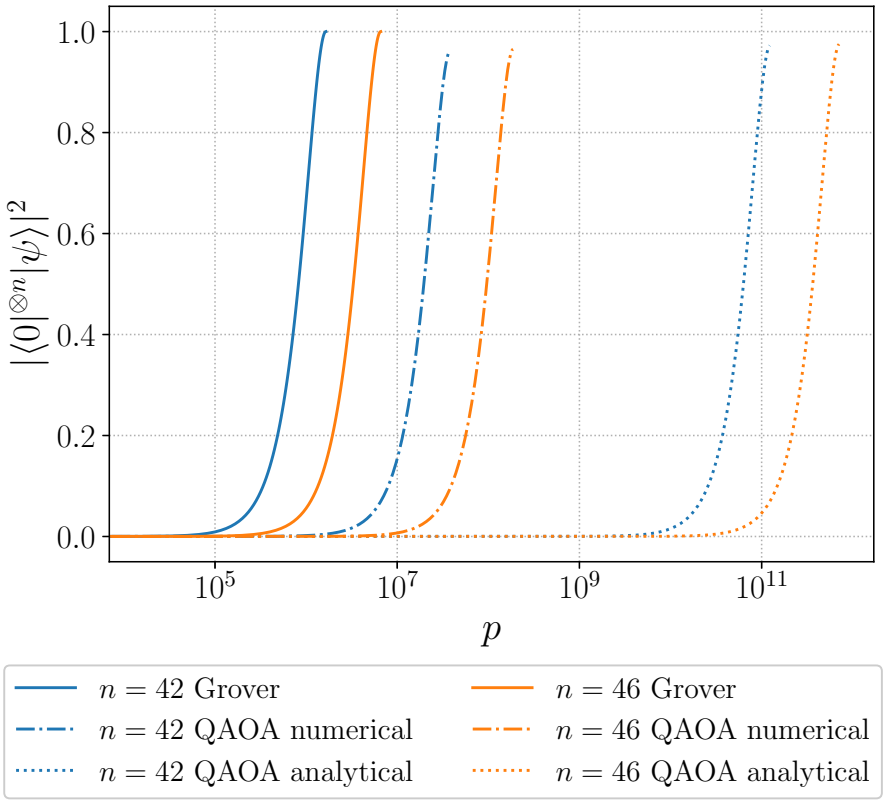


Figure 4-1: Overlap with the target state $|0\rangle^{\otimes n}$ through the circuit for states $|\psi\rangle$ prepared by: (i) Grover search, (ii) QAOA from a trotterized CTQW with numerically calculated depth, and (iii) QAOA from a trotterized CTQW with analytically predicted depth for $n = 42, 46$ and $\epsilon = 0.01$. In agreement with (4.6), the overlaps of the QAOA sequences reach $1 + \mathcal{O}\left(\frac{1}{n}\right)$. Figure taken from [3].

the optimal depth with high precision (details on the numerical execution of these circuits and search for optimal depths can be found in Appendix C.1). For the QAOA sequence obtained analytically we use a $q = 4$ order formula, as calculated from equation (4.26). We observe the overlaps from the QAOA sequences to follow smooth curves reminiscent of those of CTQWs with respect to t, reaching $1 + \mathcal{O}\left(\frac{1}{n}\right)$ as in equation (4.6).

In order to verify the tightness of expression (4.27), we exhaustively calculate the depth required by a trotterized CTQW to have an error below certain threshold ϵ. Figure 4-2 illustrates the ratio between the depth predicted by equation (4.27) and the depth calculated numerically in the range $n \in [22, 68]$ and $\epsilon = \{0.001, 0.01, 0.1\}$. When numerically approximating the optimal depth, for each pair n and ϵ, we use the formula of the order which results in the shortest sequence. The sharp fluctuations

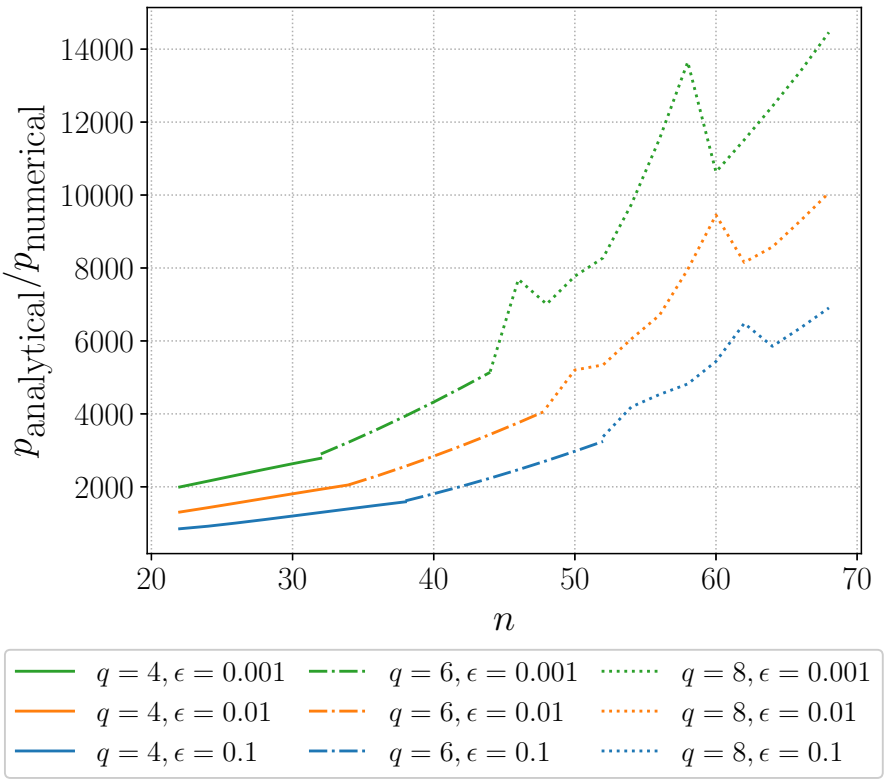


Figure 4-2: Ratios between the analytically calculated depth $p_{\text{analytical}}$, given by (4.27), and numerically calculated depth $p_{\text{numerical}}$ for system sizes $n \in [22, 68]$ and $\epsilon \in \{0.001, 0.01, 0.1\}$. For each pair n and ϵ, $p_{\text{numerical}}$ is calculated using the formula of order q which results in the shortest sequence. Figure taken from [3].

that appear at $q = 8$ are likely due to the longer sequences of every step $S_8(t/r)$, and the more computationally intensive task of approximating depths for the larger system sizes. Across the entire range the ratio demonstrates a seven fold increase, showing a much slower growth compared to the dominant factor $2^{\frac{n}{2}}$.

Figure 4-3 illustrates numerically calculated depth with respect to $\epsilon \in [0.001, 0.1]$ for $n = 40, 42, 44$ and $q = 6$. It can be observed that depth growth slows down as ϵ becomes smaller. In equation (4.27), the impact of ϵ on depth is given by the factor $\epsilon^{-\frac{2}{q}}$ which diminishes for higher order formulas (large systems), and goes to 1 in the limit $n \to \infty$.

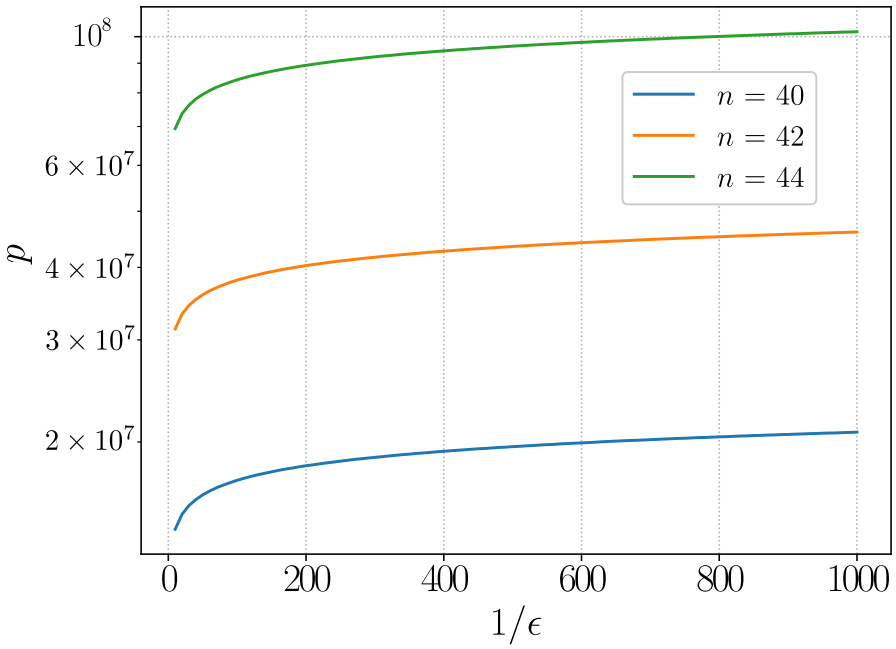


Figure 4-3: Numerically calculated depths for a range of $\epsilon \in [0.001, 0.1]$ for $n = 40, 42, 44$ and $q = 6$. Figure taken from [3].

4.5 Discussion

Our results demonstrate quantum advantage for unstructured search via QAOA where the ansatz state approaches perfect overlap with the target. We prove the query complexity to be higher than that of Grover's algorithm $\mathcal{O}(N^{\frac{1}{2}})$, but notably smaller than $\mathcal{O}\left(N^{\frac{1}{2}+c}\right)$ for any $c > 0$. Compared to the results presented by Jiang et al. [168], our approach offers an improvement to the overlap in exchange for a higher complexity.

It might be tempting to tighten the bound (4.27) by considering evolution in a two dimensional subspace spanned by the low energy eigenstates (B.24). However, the estimate obtained this way does not match the numerical experiments, predicting depths of the order of $\mathcal{O}(2^{\frac{n}{2}})$, lower than what we numerically observe. This discrepancy is likely caused by a probability leakage out of this two dimensional subspace.

Nevertheless, it may still be possible to improve the derived upper bound for depth. Equation (1.26), as introduced by Childs [112], makes use of multiple upper bounds in an attempt to make it as general as possible, which opens the door for

finding a tighter expression tailored to our particular setting. Similarly, in our analysis we make use of some generous upper bounds in order to obtain a closed form of δ, which offers room for improvement. Regardless of this, it is unlikely that any improvement would lower the bound to the point of optimal scaling. As we observe numerically, the suboptimal complexity appears to be a fundamental feature of the trotterization of the considered CTQW. Alternatively, one may consider the trotterization of hybrid adiabatic and CTQW evolutions, similar to those presented in [223]. Such evolutions also present optimal scaling and may result in Trotter sequences of lower complexity.

Furthermore our results provide a recipe for calculating the corresponding QAOA parameters, which can be used as good initial parameters for QAOA. Despite this, for some system sizes the number of QAOA layers can be prohibitively large to be globally optimized, and fall outside the scope of NISQ computers [36]. Fortunately, for small enough ϵ, the training process can be entirely bypassed, in which case our procedure falls into the category of non-variational combinatorial optimization algorithms. These algorithms reduce or eliminate their dependency on a classical optimizer by the use of near optimal parameters which are obtained through: analysis [5, 8, 163, 168], transferring parameters from similar problem instances [190, 224, 225], or by classically optimizing only a small set of parameters [226, 227].

Our result provides insight on the power of QAOA with a standard mixer and makes a step towards closing its gap, in terms of scaling and overlap, with Grover's algorithm and QAOA with modified mixers [211, 212]. If this gap were to be closed, QAOA with a standard mixer could offer a compelling alternative for unstructured search due to its easier to implement mixer. These results, together with proofs of universality [113, 114], may guaranty the relevancy of QAOA beyond the NISQ era.

Chapter 5

Conclusion

The variational model of quantum computing is our current best approach for taking advantage of NISQ computers [36]. The model has been employed in algorithms that tackle a wide variety of problems [13, 51, 115–122], and has seen successful experimental implementations [131–135]. Although originally designed around the limitations of NISQ computers, the variational model is expected to remain relevant beyond the NISQ era, as suggested by universality results [10, 113, 114]. Despite the advantages offered by the model and the continuous improvement of quantum computing hardware, limiting effects fundamental to the variational model [1, 2, 12, 53–74] have been discovered.

Notably, the discovery of *barren plateaus* [72] brought a substantial amount of attention to the trainability limitations of VQAs. Since then, a large portion of the community has put effort on understanding and developing techniques to mitigate BP. Being by far the most studied limitation, as pointed out by [59], have come to be seen by many as equivalent to training limitations. During our exploratory study on training strategies, some of which are known to alleviate BP, we encountered previously unreported training limitations. This led us to extend our focus to limitations beyond BP, as well as to understand the trade offs of certain training strategies.

We studied layer-wise training when applied to the problem of variational quantum compilation of k-Toffoli gates, using training by layer stacks. In our study of layer-wise training we identified a previously unknown trainability limiting effect

we called *abrupt trainability transitions* (ATT) [1]. We empirically observed that bellow a critical number of layers per stack c, these would train close to the identity. Abruptly, for a number of layers per stack equal or larger than c, the quantum circuit would be able to train arbitrarily close to the target gate. Furthermore we proved that for the hardware efficient ansatz, and the checkerboard ansatz, the cost function used for variational quantum compilation of k-Toffolli gates, has a minimum when the circuit evaluates to an identity. The implications of this result put in question the benefits of using the identity initialization strategy [62, 84]. This is important as it can be considered as the easiest technique to implement in order to mitigate BP. Modifications to the standard implementation of layer-wise training may be able to recover its viability in this setting. In [151] the researchers did not observe ATT when compiling a Toffoli gate as, inadvertently, their ansatz of choice could not evaluate to an identity in a single layer. This suggests that a deliberate use of such an ansatz could be a valid strategy to mitigate ATT. Another set of potential strategies would consist on limiting the performance from stack to stack, similar to what we did in in the context of QAOA [2].

The quantum approximate optimization algorithm (QAOA) [121], originally proposed for finding approximate solutions to combinatorial optimization problems, was later proved to be a universal model of quantum computing [113, 114]. The algorithm has been used approximate solutions for problems such as MAX-CUT, MAX-k-SAT, unstructured search, and others. For the previously mentioned problems, we studied the effects of layer-wise training in QAOA. Here we identified a previously unknown trainability limiting effect we called *training saturation* (TS), where a circuit trained layer-wise would stop training after a certain number of layers [2]. This effect was observed empirically for the three problems we studied. In particular for the problem of unstructured search we noticed TS occur at depth $p^* = n$, and were able analytically obtain necessary conditions for TS. Based on this result, we were able to propose strategies that would prevent TS by violating the conditions. By considering coherent phase noise we were able to remove TS, effectively allowing for training beyond the otherwise saturation depth. Although we focused only on the simplest noise model, our result has the potential of being applied to a more

realistic setting.

Finally, we explored the possibility of analytically obtaining near optimal angles for the problem of unstructured search via QAOA. Such a result would allow for training in a region free of BP, or even to bypass the training process entirely. Previous results have hinted at this possibility: in [5] it is shown that parameters concentrate across system sizes, and in [168] the researchers provided angles for an overlap with the target state that approaches 1/2. Our study took inspiration on the well known result of unstructured search in a continuous time quantum walk (CTQW) [214], which has a complexity $\mathcal{O}(N^{\frac{1}{2}})$, the same as Grover's algorithm [23]. Aided by the results presented in [112], we prove that, when using he standard mixer, the QAOA sequence required to prepare a state of an overlap with the target state that approaches unity is larger than that Grover's algorithm but lower than $\mathcal{O}(N^{\frac{1}{2}+c})$, for any $c > 0$. Although the QAOA angles given by this method can be considered good initial parameters with an approximation error to a CTQW of at most ϵ, the exponentially large number of them makes it impossible to train via simultaneous optimization, thus alternative optimization strategies should be employed. Alternatively further decreasing ϵ may allow to entirely bypass training. As for real implementations on quantum hardware, the required depths are beyond the scope of what current NISQ computers can achieve. Nonetheless, QAOA is expected to remain a relevant model beyond the NISQ era, and the simpler implementation of the QAOA mixer operators (both standard and modified [212]) compared to Grover's makes it a more attractive alternative.

The results presented in this thesis have become part of the VQA literature and appeared in multiple reviews [228], have influenced the development of newer training techniques [6, 200–202, 209, 210, 229], and helped the community to better understand training limitations [59, 60]. We plan to continue our research into training strategies, as well as the development of new ones. As we know, this may have as by-product the discovery of new limitations. Our future plans include: the study of new approaches to develop problem specific ansatzes, the use of divide and conquer strategies to better learn specific regions of the input space in quantum machine learning tasks, the effect of error correction strategies in training, the effect

of training limitations on generative quantum algorithms, finding alternative QAOA sequences for unstructured search based on hybrid adiabatic and CTQW evolutions, among others.

While the discovery of more training limitations could be considered a negative result, they provide valuable information on how to possibly avoid them. The absence of this knowledge could otherwise result on somebody's experiment ultimately failing without a clear reason. Adapting to a training limitation was exemplified in Chapter 3, where adding low values of coherent phase noise proved fruitful and led to a strategy that improves on the performance of layer-wise training. This case is specially relevant in now a days mid-size NISQ devices where the presence of noise is defining factor to the practicality of such machines. On the other hand, our results on thee depth scaling of unstructured search via QAOA provide evidence for of the future viability of VQAs in an era of error corrected quantum computers, where the use of VQAs could still be preferred to traditional quantum algorithms.

We anticipate that as understanding of BP increases, there will be renewed interest from the community in other limitations, such as those discussed here. We believe that the insights gained from a broader exploration of training limitations will reinforce the variational model's position as a powerful and relevant approach beyond the NISQ era of quantum computing.

Results

1. Layer-wise variational quantum compilation experiences a trainability limiting effect we called *abrupt trainability transitions*, where a minimum number of layers per stack is required, otherwise training fails. We observe this effect when compiling $k-$Toffoli gates using the hardware efficient ansatz, and checkerboard ansatz. This offers a counterexample to the layer-wise trainability conjecture.

2. The cost function for the variational quantum compilation of $k-$Toffoli gates takes extrema when the circuit evaluates to identity. Therefore in this setting, the *identity initialization strategy* starts training in a bad local minimum.

3. Layer-wise QAOA exhibits a trainability limiting effect we called *training saturations*, where training stops after a certain number of layers. We observe this effect on instances of MAX-CUT, MAX-k-SAT, and unstructured search.

4. The addition of small probabilities of coherent phase noise remove training saturations, thereby recovering prospects of layer-wise training.

5. The query complexity of unstructured search via QAOA required to prepare a state which overlap with the target approaches unity is less than $O\left(N^{\frac{1}{2}+c}\right)$ for any $c < 0$. This complexity lies between that of Grover's algorithm and classical algorithms.

Bibliography

[1] E. Campos, A. Nasrallah, and J. Biamonte. Abrupt transitions in variational quantum circuit training. *Physical Review A*, 103(3):032607, 2021. doi: 10.1103/PhysRevA.103.032607.

[2] E. Campos, D. Rabinovich, V. Akshay, and J. Biamonte. Training saturation in layerwise quantum approximate optimization. *Physical Review A*, 104(3): L030401, 2021. doi: 10.1103/PhysRevA.104.L030401.

[3] E. Campos, D. Rabinovich, and A. Uvarov. Depth scaling of unstructured search via quantum approximate optimization. *Physical Review A*, 110(1): 012428, 2024. doi: 10.1103/PhysRevA.110.012428.

[4] E. Campos, S. Venegas-Andraca, and M. Lanzagorta. Quantum tunneling and quantum walks as algorithmic resources to solve hard k-sat instances. *Scientific Reports*, 11(1):16845, 2021. doi: 10.1038/s41598-021-95801-1.

[5] V. Akshay, D. Rabinovich, E. Campos, and J. Biamonte. Parameter concentrations in quantum approximate optimization. *Physical Review A*, 104(1): L010401, 2021. doi: 10.1103/PhysRevA.104.L010401.

[6] V. Akshay, H. Philathong, E. Campos, D. Rabinovich, I. Zacharov, X. Zhang, and J. Biamonte. Circuit depth scaling for quantum approximate optimization. *Physical Review A*, 106(4):042438, 2022. doi: 10.1103/PhysRevA.106.042438.

[7] D. Rabinovich, S. Adhikary, E. Campos, V. Akshay, E. Anikin, R. Sengupta, O. Lakhmanskaya, K. Lakhmanskiy, and J. Biamonte. Ion-native variational ansatz for quantum approximate optimization. *Physical Review A*, 106(3): 032418, 2022. doi: https://doi.org/10.1103/PhysRevA.106.032418.

[8] D. Rabinovich, R. Sengupta, E. Campos, V. Akshay, and J. Biamonte. Progress towards analytically optimal angles in quantum approximate optimisation. *Mathematics*, 10(15):2601, 2022. doi: 10.3390/math10152601.

[9] D. Rabinovich, E. Campos, S. Adhikary, E. Pankovets, D. Vinichenko, and J. Biamonte. Robustness of variational quantum algorithms against stochastic parameter perturbation. *Physical Review A*, 109(4):042426, 2024. doi: 10.1103/PhysRevA.109.042426.

[10] Jacob Biamonte. Universal variational quantum computation. *Physical Review A*, 103(3):L030401, 2021.

[11] A. V. Uvarov, A. S. Kardashin, and J. D. Biamonte. Machine learning phase transitions with a quantum processor. *Phys. Rev. A*, 102:012415, Jul 2020. doi: 10.1103/PhysRevA.102.012415. URL https://link.aps.org/doi/10.1103/PhysRevA.102.012415.

[12] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9): 625–644, 2021.

[13] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T Sornborger, and Patrick J Coles. Quantum-assisted quantum compiling. *Quantum*, 3:140, 2019.

[14] AV Uvarov and Jacob D Biamonte. On barren plateaus and cost function locality in variational quantum algorithms. *Journal of Physics A: Mathematical and Theoretical*, 54(24):245301, 2021.

[15] Juris Hartmanis and Richard E Stearns. On the computational complexity of algorithms. *Transactions of the American Mathematical Society*, 117:285–306, 1965.

[16] Stephen A Cook. The complexity of theorem-proving procedures. In *Logic, Automata, and Computational Complexity: The Works of Stephen A. Cook*, pages 143–152. 2023.

[17] Richard M Karp. Combinatorics, complexity, and randomness. *Communications of the ACM*, 29(2):98–109, 1986.

[18] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985.

[19] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 11–20, 1993.

[20] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, 1992.

[21] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[22] Daniel R Simon. On the power of quantum computation. *SIAM journal on computing*, 26(5):1474–1483, 1997.

[23] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[24] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[25] Yulin Wu, Wan-Su Bao, Sirui Cao, Fusheng Chen, Ming-Cheng Chen, Xiawei Chen, Tung-Hsun Chung, Hui Deng, Yajie Du, Daojin Fan, et al. Strong quantum computational advantage using a superconducting quantum processor. *Physical review letters*, 127(18):180501, 2021.

[26] Morten Kjaergaard, Mollie E Schwartz, Jochen Braumüller, Philip Krantz, Joel I-J Wang, Simon Gustavsson, and William D Oliver. Superconducting qubits: Current state of play. *Annual Review of Condensed Matter Physics*, 11:369–395, 2020.

[27] TM Graham, Y Song, J Scott, C Poole, L Phuttitarn, K Jooya, P Eichler, X Jiang, A Marra, B Grinkemeyer, et al. Multi-qubit entanglement and algorithms on a neutral-atom quantum computer. *Nature*, 604(7906):457–462, 2022.

[28] TM Graham, L Phuttitarn, R Chinnarasu, Y Song, C Poole, K Jooya, J Scott, A Scott, P Eichler, and M Saffman. Midcircuit measurements on a single-species neutral alkali atom quantum processor. *Physical Review X*, 13(4): 041051, 2023.

[29] Dolev Bluvstein, Simon J Evered, Alexandra A Geim, Sophie H Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, et al. Logical quantum processor based on reconfigurable atom arrays. *Nature*, 626(7997):58–65, 2024.

[30] Dolev Bluvstein, Harry Levine, Giulia Semeghini, Tout T Wang, Sepehr Ebadi, Marcin Kalinowski, Alexander Keesling, Nishad Maskara, Hannes Pichler, Markus Greiner, et al. A quantum processor based on coherent transport of entangled atom arrays. *Nature*, 604(7906):451–456, 2022.

[31] Pavel Hrmo, Benjamin Wilhelm, Lukas Gerster, Martin W van Mourik, Marcus Huber, Rainer Blatt, Philipp Schindler, Thomas Monz, and Martin Ringbauer. Native qudit entanglement in a trapped ion quantum processor. *Nature Communications*, 14(1):2242, 2023.

[32] Steven A Moses, Charles H Baldwin, Michael S Allman, R Ancona, L Ascarrunz, C Barnes, J Bartolotta, B Bjork, P Blanchard, M Bohn, et al. A race-track trapped-ion quantum processor. *Physical Review X*, 13(4):041052, 2023.

[33] Martin Ringbauer, Michael Meth, Lukas Postler, Roman Stricker, Rainer Blatt, Philipp Schindler, and Thomas Monz. A universal qudit quantum processor with trapped ions. *Nature Physics*, 18(9):1053–1057, 2022.

[34] Lars S Madsen, Fabian Laudenbach, Mohsen Falamarzi Askarani, Fabien Rortais, Trevor Vincent, Jacob FF Bulmer, Filippo M Miatto, Leonhard Neuhaus, Lukas G Helt, Matthew J Collins, et al. Quantum computational advantage with a programmable photonic processor. *Nature*, 606(7912):75–81, 2022.

[35] Juan M Arrazola, Ville Bergholm, Kamil Brádler, Thomas R Bromley, Matt J Collins, Ish Dhand, Alberto Fumagalli, Thomas Gerrits, Andrey Goussev, Lukas G Helt, et al. Quantum circuits with many photons on a programmable nanophotonic chip. *Nature*, 591(7848):54–60, 2021.

[36] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2: 79, 2018.

[37] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, and Wojciech Hubert Zurek. Perfect quantum error correcting code. *Physical Review Letters*, 77(1): 198, 1996.

[38] Aamir Mandviwalla, Keita Ohshiro, and Bo Ji. Implementing grover's algorithm on the ibm quantum computers. In *2018 IEEE international conference on big data (big data)*, pages 2531–2537. IEEE, 2018.

[39] Clément Godfrin, Abdelkarim Ferhat, Rafik Ballou, Svetlana Klyatskaya, Mario Ruben, Wolfgang Wernsdorfer, and Franck Balestro. Operating quantum states in single magnetic molecules: implementation of grover's quantum algorithm. *Physical review letters*, 119(18):187702, 2017.

[40] Thomas Monz, Daniel Nigg, Esteban A Martinez, Matthias F Brandl, Philipp Schindler, Richard Rines, Shannon X Wang, Isaac L Chuang, and Rainer Blatt. Realization of a scalable shor algorithm. *Science*, 351(6277):1068–1070, 2016.

[41] Austin G Fowler, Simon J Devitt, and Lloyd CL Hollenberg. Implementation of shor's algorithm on a linear nearest neighbour qubit array. *arXiv preprint quant-ph/0402196*, 2004.

[42] Jacob Biamonte. On the mathematical structure of quantum models of computation based on hamiltonian minimisation. *arXiv preprint arXiv:2009.10088*, 2020.

[43] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, 2017.

[44] Ho Lun Tang, VO Shkolnikov, George S Barron, Harper R Grimsley, Nicholas J Mayhall, Edwin Barnes, and Sophia E Economou. qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor. *PRX Quantum*, 2(2):020310, 2021.

[45] Kouhei Nakaji and Naoki Yamamoto. Expressibility of the alternating layered ansatz for quantum computation. *Quantum*, 5:434, 2021.

[46] Lorenzo Leone, Salvatore FE Oliviero, Lukasz Cincio, and M Cerezo. On the practical usefulness of the hardware efficient ansatz. *arXiv preprint arXiv:2211.01477*, 2022.

[47] Kunal Sharma, Sumeet Khatri, M Cerezo, and Patrick J Coles. Noise resilience of variational quantum compiling. *New Journal of Physics*, 22(4): 043006, apr 2020. doi: 10.1088/1367-2630/ab784c. URL https://doi.org/10.1088%2F1367-2630%2Fab784c.

[48] Laura Gentini, Alessandro Cuccoli, Stefano Pirandola, Paola Verrucchi, and Leonardo Banchi. Noise-resilient variational hybrid quantum-classical optimization. *Physical Review A*, 102(5):052414, 2020.

[49] Lukasz Cincio, Kenneth Rudinger, Mohan Sarovar, and Patrick J Coles. Machine learning of noise-resilient quantum circuits. *PRX Quantum*, 2(1):010324, 2021.

[50] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016.

[51] Jarrod R McClean, Mollie E Kimchi-Schwartz, Jonathan Carter, and Wibe A De Jong. Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states. *Physical Review A*, 95(4):042308, 2017.

[52] Enrico Fontana, Nathan Fitzpatrick, David Muñoz Ramo, Ross Duncan, and Ivan Rungger. Evaluating the noise resilience of variational quantum algorithms. *Physical Review A*, 104(2):022403, 2021.

[53] Junyu Liu, Khadijeh Najafi, Kunal Sharma, Francesco Tacchino, Liang Jiang, and Antonio Mezzacapo. Analytic theory for the dynamics of wide quantum neural networks. *Physical Review Letters*, 130(15):150601, 2023.

[54] Xuchen You, Shouvanik Chakrabarti, and Xiaodi Wu. A convergence theory for over-parameterized variational quantum eigensolvers. *arXiv preprint arXiv:2205.12481*, 2022.

[55] Martin Larocca, Nathan Ju, Diego García-Martín, Patrick J Coles, and Marco Cerezo. Theory of overparametrization in quantum neural networks. *Nature Computational Science*, 3(6):542–551, 2023.

[56] Marco Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature communications*, 12(1):1791, 2021.

[57] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to variational quantum optimization from symmetry protection. *Physical review letters*, 125(26):260505, 2020.

[58] Giacomo De Palma, Milad Marvian, Cambyse Rouzé, and Daniel Stilck França. Limitations of variational quantum algorithms: a quantum optimal transport approach. *PRX Quantum*, 4(1):010309, 2023.

[59] Eric R Anschuetz and Bobak T Kiani. Quantum variational algorithms are swamped with traps. *Nature Communications*, 13(1):7760, 2022.

[60] Eric R Anschuetz. Critical points in quantum generative models. *arXiv preprint arXiv:2109.06957*, 2021.

[61] V. Akshay, H. Philathong, M. E.S. Morales, and J. D. Biamonte. Reachability Deficits in Quantum Approximate Optimization. *Physical Review Letters*, 124(9):090504, mar 2020. ISSN 10797114. doi: 10.1103/PhysRevLett.124.090504. URL https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.124.090504.

[62] Andrea Skolik, Jarrod R McClean, Masoud Mohseni, Patrick van der Smagt, and Martin Leib. Layerwise learning for quantum neural networks. *Quantum Machine Intelligence*, 3(1):1–11, 2021.

[63] Daniel Stilck França and Raul Garcia-Patron. Limitations of optimization algorithms on noisy quantum devices. *Nature Physics*, 17(11):1221–1227, 2021.

[64] Xavier Bonet-Monroig, Hao Wang, Diederick Vermetten, Bruno Senjean, Charles Moussa, Thomas Bäck, Vedran Dunjko, and Thomas E O'Brien. Performance comparison of optimization methods on variational quantum algorithms. *Physical Review A*, 107(3):032407, 2023.

[65] David Wierichs, Christian Gogolin, and Michael Kastoryano. Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer. *Physical Review Research*, 2(4):043246, 2020.

[66] Roeland Wiersema and Nathan Killoran. Optimizing quantum circuits with riemannian gradient flow. *Physical Review A*, 107(6):062421, 2023.

[67] Lennart Bittel and Martin Kliesch. Training variational quantum algorithms is np-hard. *Physical review letters*, 127(12):120502, 2021.

[68] Xuchen You and Xiaodi Wu. Exponentially many local minima in quantum neural networks. In *International Conference on Machine Learning*, pages 12144–12155. PMLR, 2021.

[69] Joel Rajakumar, John Golden, Andreas Bärtschi, and Stephan Eidenbenz. Trainability barriers in low-depth qaoa landscapes. *arXiv preprint arXiv:2402.10188*, 2024.

[70] Roeland Wiersema, Cunlu Zhou, Yvette de Sereville, Juan Felipe Carrasquilla, Yong Baek Kim, and Henry Yuen. Exploring entanglement and optimization within the hamiltonian variational ansatz. *PRX Quantum*, 1(2):020319, 2020.

[71] Bobak Toussi Kiani, Seth Lloyd, and Reevu Maity. Learning unitaries by gradient descent. *arXiv preprint arXiv:2001.11897*, 2020.

[72] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9(1):1–6, 2018.

[73] Martin Larocca, Supanut Thanasilp, Samson Wang, Kunal Sharma, Jacob Biamonte, Patrick J Coles, Lukasz Cincio, Jarrod R McClean, Zoë Holmes, and M Cerezo. A review of barren plateaus in variational quantum computing. *arXiv preprint arXiv:2405.00781*, 2024.

[74] Samson Wang, Enrico Fontana, Marco Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nature communications*, 12(1):6961, 2021.

[75] Owen Lockwood. An empirical review of optimization techniques for quantum variational circuits. *arXiv preprint arXiv:2202.01389*, 2022.

[76] Aidan Pellow-Jarman, Ilya Sinayskiy, Anban Pillay, and Francesco Petruccione. A comparison of various classical optimizers for a variational quantum linear solver. *Quantum Information Processing*, 20(6):202, 2021.

[77] Javier Rivera-Dean, Patrick Huembeli, Antonio Acín, and Joseph Bowles. Avoiding local minima in variational quantum algorithms with neural networks. *arXiv preprint arXiv:2104.02955*, 2021.

[78] Aram W Harrow and Richard A Low. Random quantum circuits are approximate 2-designs. *Communications in Mathematical Physics*, 291:257–302, 2009.

[79] Fernando GSL Brandao, Aram W Harrow, and Michał Horodecki. Local random quantum circuits are approximate polynomial-designs. *Communications in Mathematical Physics*, 346:397–434, 2016.

[80] Zoë Holmes, Kunal Sharma, Marco Cerezo, and Patrick J Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum*, 3(1):010313, 2022.

[81] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. Entanglement-induced barren plateaus. *PRX Quantum*, 2(4):040316, 2021.

[82] Zoë Holmes, Andrew Arrasmith, Bin Yan, Patrick J Coles, Andreas Albrecht, and Andrew T Sornborger. Barren plateaus preclude learning scramblers. *Physical Review Letters*, 126(19):190501, 2021.

[83] Andrew Arrasmith, Marco Cerezo, Piotr Czarnik, Lukasz Cincio, and Patrick J Coles. Effect of barren plateaus on gradient-free optimization. *Quantum*, 5: 558, 2021.

[84] Edward Grant, Leonard Wossnig, Mateusz Ostaszewski, and Marcello Benedetti. An initialization strategy for addressing barren plateaus in parametrized quantum circuits. *Quantum*, 3:214, dec 2019. ISSN 2521-327X. doi: 10.22331/q-2019-12-09-214. URL https://quantum-journal.org/papers/q-2019-12-09-214/.

[85] Tyler Volkoff and Patrick J Coles. Large gradients via correlation in random parameterized quantum circuits. *Quantum Science and Technology*, 6(2): 025008, 2021.

[86] Stuart Hadfield, Zhihui Wang, Bryan O'gorman, Eleanor G Rieffel, Davide Venturelli, and Rupak Biswas. From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34, 2019.

[87] Andrew G Taube and Rodney J Bartlett. New perspectives on unitary coupled-cluster theory. *International journal of quantum chemistry*, 106(15):3393–3401, 2006.

[88] Guillaume Verdon, Michael Broughton, Jarrod R McClean, Kevin J Sung, Ryan Babbush, Zhang Jiang, Hartmut Neven, and Masoud Mohseni. Learning to learn with quantum neural networks via classical neural networks. *arXiv preprint arXiv:1907.05415*, 2019.

[89] Kishor Bharti and Tobias Haug. Iterative quantum-assisted eigensolver. *Physical Review A*, 104(5):L050401, 2021.

[90] Kishor Bharti and Tobias Haug. Quantum-assisted simulator. *Physical Review A*, 104(4):042418, 2021.

[91] Phattharaporn Singkanipa and Daniel A Lidar. Beyond unital noise in variational quantum algorithms: noise-induced barren plateaus and fixed points. *arXiv preprint arXiv:2402.08721*, 2024.

[92] Kunal Sharma, Marco Cerezo, Lukasz Cincio, and Patrick J Coles. Trainability of dissipative perceptron-based quantum neural networks. *Physical Review Letters*, 128(18):180505, 2022.

[93] M Cerezo, Kunal Sharma, Andrew Arrasmith, and Patrick J Coles. Variational quantum state eigensolver. *npj Quantum Information*, 8(1):113, 2022.

[94] Lucas Friedrich and Jonas Maziero. Avoiding barren plateaus with classical deep neural networks. *Physical Review A*, 106(4):042433, 2022.

[95] Taylor L Patti, Khadijeh Najafi, Xun Gao, and Susanne F Yelin. Entanglement devised barren plateau mitigation. *Physical Review Research*, 3(3):033090, 2021.

[96] Stefan H Sack, Raimel A Medina, Alexios A Michailidis, Richard Kueng, and Maksym Serbyn. Avoiding barren plateaus using classical shadows. *PRX Quantum*, 3(2):020365, 2022.

[97] Abhinav Anand, Matthias Degroote, and Alán Aspuru-Guzik. Natural evolutionary strategies for variational quantum computation. *Machine Learning: Science and Technology*, 2(4):045012, 2021.

[98] Chenfeng Cao and Xin Wang. Noise-assisted quantum autoencoder. *Physical Review Applied*, 15(5):054012, 2021.

[99] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011. ISBN 1107002176.

[100] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[101] Paul Adrien Maurice Dirac. A new notation for quantum mechanics. In *Mathematical proceedings of the Cambridge philosophical society*, volume 35, pages 416–418. Cambridge University Press, 1939.

[102] Claude Cohen-Tannoudji, Bernard Diu, and Franck Laloë. *Quantum Mechanics Volume 2*. Hermann, 1986.

[103] Joel N Franklin. *Matrix theory*. Courier Corporation, 2012.

[104] David J Griffiths and Darrell F Schroeter. *Introduction to quantum mechanics*. Cambridge university press, 2018.

[105] Colin P Williams. *Explorations in quantum computing*. Springer Science & Business Media, 2010.

[106] Alexander S Holevo. *Statistical structure of quantum theory*, volume 67. Springer Science & Business Media, 2003.

[107] Ola Bratteli and Derek William Robinson. *Operator algebras and quantum statistical mechanics: Volume 1: C*-and W*-Algebras. Symmetry Groups. Decomposition of States*. Springer Science & Business Media, 2012.

[108] Frank Wilczek. On absolute units, i: Choices. *Physics Today*, 58(10):12–13, 2005.

[109] David Elieser Deutsch, Adriano Barenco, and Artur Ekert. Universality in quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 449(1937):669–677, 1995.

[110] Masuo Suzuki. Generalized trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems. *Communications in Mathematical Physics*, 51(2):183–190, 1976.

[111] Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, 270:359–371, 2007.

[112] Andrew M Childs, Yuan Su, Minh C Tran, Nathan Wiebe, and Shuchen Zhu. Theory of trotter error with commutator scaling. *Physical Review X*, 11(1): 011020, 2021.

[113] Seth Lloyd. Quantum approximate optimization is computationally universal. *arXiv preprint arXiv:1812.11075*, 2018.

[114] Mauro ES Morales, JD Biamonte, and Zoltán Zimborás. On the universality of the quantum approximate optimization algorithm. *Quantum Information Processing*, 19(9):1–26, 2020.

[115] 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.

[116] A Garcia-Saez and JI Latorre. Addressing hard classical problems with adiabatically assisted variational quantum eigensolvers. *arXiv preprint arXiv:1806.02287*, 2018.

[117] Xiao Yuan, Suguru Endo, Qi Zhao, Ying Li, and Simon C Benjamin. Theory of variational quantum simulation. *Quantum*, 3:191, 2019.

[118] Cristina Cirstoiu, Zoe Holmes, Joseph Iosue, Lukasz Cincio, Patrick J Coles, and Andrew Sornborger. Variational fast forwarding for quantum simulation beyond the coherence time. *npj Quantum Information*, 6(1):82, 2020.

[119] Joe Gibbs, Kaitlin Gili, Zoë Holmes, Benjamin Commeau, Andrew Arrasmith, Lukasz Cincio, Patrick J Coles, and Andrew Sornborger. Long-time simulations with high fidelity on quantum hardware. *arXiv preprint arXiv:2102.04313*, 2021.

[120] Suguru Endo, Jinzhao Sun, Ying Li, Simon C Benjamin, and Xiao Yuan. Variational quantum simulation of general processes. *Physical Review Letters*, 125(1):010501, 2020.

[121] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[122] Tyson Jones and Simon C Benjamin. Robust quantum compilation and circuit optimisation via energy minimisation. *Quantum*, 6:628, 2022.

[123] Zhimin He, Lvzhou Li, Shenggen Zheng, Yongyao Li, and Haozhen Situ. Variational quantum compiling with double q-learning. *New Journal of Physics*, 23(3):033002, 2021.

[124] Jin-Min Liang, Shu-Qian Shen, Ming Li, and Lei Li. Variational quantum algorithms for dimensionality reduction and classification. *Physical Review A*, 101(3):032323, 2020.

[125] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, 2019.

[126] Guillaume Verdon, Michael Broughton, and Jacob Biamonte. A quantum algorithm to train neural networks using low-depth circuits. *arXiv preprint arXiv:1712.05304*, 2017.

[127] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

[128] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3):032331, 2019.

[129] Xiaosi Xu, Simon C Benjamin, and Xiao Yuan. Variational circuit compiler for quantum error correction. *arXiv preprint arXiv:1911.05759*, 2019.

[130] Alexey Uvarov, Jacob D. Biamonte, and Dmitry Yudin. Variational quantum eigensolver for frustrated quantum systems. *Phys. Rev. B*, 102:075104, Aug 2020. doi: 10.1103/PhysRevB.102.075104. URL https://link.aps.org/doi/10.1103/PhysRevB.102.075104.

[131] Jonathan Romero and Alán Aspuru-Guzik. Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions. *Advanced Quantum Technologies*, 4(1):2000003, 2021.

[132] Jacques Carolan, Masoud Mohseni, Jonathan P Olson, Mihika Prabhu, Changchen Chen, Darius Bunandar, Murphy Yuezhen Niu, Nicholas C Harris, Franco NC Wong, Michael Hochberg, et al. Variational quantum unsampling on a quantum photonic processor. *Nature Physics*, 16(3):322–327, 2020.

[133] Amir H Karamlou, William A Simon, Amara Katabarwa, Travis L Scholten, Borja Peropadre, and Yudong Cao. Analyzing the performance of variational quantum factoring on a superconducting quantum processor. *npj Quantum Information*, 7(1):156, 2021.

[134] Abhinav Anand, Jonathan Romero, Matthias Degroote, and Alán Aspuru-Guzik. Noise robustness and experimental demonstration of a quantum generative adversarial network for continuous distributions. *Advanced Quantum Technologies*, 4(5):2000069, 2021.

[135] Abhinav Anand, Jonathan Romero, Matthias Degroote, and Alán Aspuru-Guzik. Experimental demonstration of a quantum generative adversarial network for continuous distributions. *arXiv preprint arXiv:2006.01976*, 2020.

[136] Oscar Higgott, Daochen Wang, and Stephen Brierley. Variational quantum computation of excited states. *Quantum*, 3:156, 2019.

[137] William M Kirby and Peter J Love. Contextuality test of the nonclassicality of variational quantum eigensolvers. *Physical Review Letters*, 123(20):200501, 2019.

[138] Matthew B Hastings. Classical and quantum bounded depth approximation algorithms. *arXiv preprint arXiv:1905.07047*, 2019.

[139] Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S Kottmann, Tim Menke, et al. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1):015004, 2022.

[140] Samson Wang, Piotr Czarnik, Andrew Arrasmith, Marco Cerezo, Lukasz Cincio, and Patrick J Coles. Can error mitigation improve trainability of noisy variational quantum algorithms? *Quantum*, 8:1287, 2024.

[141] Irfan Siddiqi. Engineering high-coherence superconducting qubits. *Nature Reviews Materials*, 6(10):875–891, 2021.

[142] Carlos Bravo-Prieto, Josep Lumbreras-Zarapico, Luca Tagliacozzo, and José I Latorre. Scaling of variational quantum circuit depth for condensed matter systems. *Quantum*, 4:272, 2020.

[143] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*. American Mathematical Society, USA, 2002. ISBN 0821832298.

[144] Marco Maronese, Lorenzo Moro, Lorenzo Rocutto, and Enrico Prati. Quantum compiling. In *Quantum Computing Environments*, pages 39–74. Springer, 2022.

[145] Anders Sørensen and Klaus Mølmer. Quantum computation with ions in thermal motion. *Physical review letters*, 82(9):1971, 1999.

[146] Adi Botea, Akihiro Kishimoto, and Radu Marinescu. On the complexity of quantum circuit compilation. In *Proceedings of the International Symposium on Combinatorial Search*, volume 9, pages 138–142, 2018.

[147] Roman Schutski, Taras Khakhulin, Ivan Oseledets, and Dmitry Kolmakov. Simple heuristics for efficient parallel tensor contraction and quantum circuit simulation. *Physical Review A*, 102(6):062614, 2020.

[148] Feng Pan and Pan Zhang. Simulation of quantum circuits using the big-batch tensor network method. *Physical Review Letters*, 128(3):030501, 2022.

[149] Lorenzo Moro, Matteo GA Paris, Marcello Restelli, and Enrico Prati. Quantum compiling by deep reinforcement learning. *Communications Physics*, 4 (1):178, 2021.

[150] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T. Sornborger, and Patrick J. Coles. Quantum-assisted quantum compiling. *Quantum*, 3:140, May 2019. ISSN 2521-327X. doi: 10.22331/q-2019-05-13-140. URL https://doi.org/10.22331/q-2019-05-13-140.

[151] Kunal Sharma, Sumeet Khatri, Marco Cerezo, and Patrick J Coles. Noise resilience of variational quantum compiling. *New Journal of Physics*, 22(4): 043006, 2020.

[152] Liam Madden and Andrea Simonetto. Best approximate quantum compiling problems. *ACM Transactions on Quantum Computing*, 3(2):1–29, 2022.

[153] Layla Hormozi, Georgios Zikos, Nicholas E Bonesteel, and Steven H Simon. Topological quantum compiling. *Physical Review B*, 75(16):165310, 2007.

[154] Yuan-Hang Zhang, Pei-Lin Zheng, Yi Zhang, and Dong-Ling Deng. Topological quantum compiling with reinforcement learning. *Physical Review Letters*, 125(17):170501, 2020.

[155] Y Zhiyenbayev, VM Akulin, and Aikaterini Mandilara. Quantum compiling with diffusive sets of gates. *Physical Review A*, 98(1):012325, 2018.

[156] John B Conway. *A course in functional analysis*, volume 96. Springer, 2019.

[157] Jacob Biamonte and Ville Bergholm. Tensor networks in a nutshell. *arXiv preprint arXiv:1708.00006*, 2017.

[158] Murphy Yuezhen Niu, Sirui Lu, and Isaac L Chuang. Optimizing qaoa: Success probability and runtime dependence on circuit depth. *arXiv preprint arXiv:1905.12134*, May 2019.

[159] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X*, 10:021067, Jun 2020. doi: 10.1103/PhysRevX.10.021067. URL https://link.aps.org/doi/10.1103/PhysRevX.10.021067.

[160] Zhihui Wang, Nicholas C Rubin, Jason M Dominy, and Eleanor G Rieffel. X y mixers: Analytical and numerical results for the quantum alternating operator ansatz. *Physical Review A*, 101(1):012320, 2020.

[161] Lucas T. Brady, Christopher L. Baldwin, Aniruddha Bapat, Yaroslav Kharkov, and Alexey V. Gorshkov. Optimal Protocols in Quantum Annealing and Quantum Approximate Optimization Algorithm Problems. *Physical Review Letters*, 126(7):070505, Feb 2021. ISSN 0031-9007. doi: 10.1103/PhysRevLett.126.070505. URL https://link.aps.org/doi/10.1103/PhysRevLett.126.070505.

[162] Edward Farhi and Aram W Harrow. Quantum supremacy through the quantum approximate optimization algorithm. *arXiv preprint arXiv:1602.07674*, 2016.

[163] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. The quantum approximate optimization algorithm and the sherrington-kirkpatrick model at infinite size. *Quantum*, 6:759, 2022.

[164] Matteo M Wauters, Glen Bigan Mbeng, and Giuseppe E Santoro. Polynomial scaling of qaoa for ground-state preparation of the fully-connected p-spin ferromagnet. *arXiv preprint arXiv:2003.07419*, 2020.

[165] Jahan Claes and Wim van Dam. Instance independence of single layer quantum approximate optimization algorithm on mixed-spin models at infinite size. *arXiv preprint arXiv:2102.12043*, 2021.

[166] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Phys. Rev. X*, 10:021067, Jun 2020. doi: 10.1103/PhysRevX.10.021067. URL https://link.aps.org/doi/10.1103/PhysRevX.10.021067.

[167] Matthew P Harrigan, Kevin J Sung, Matthew Neeley, Kevin J Satzinger, Frank Arute, Kunal Arya, Juan Atalaya, Joseph C Bardin, Rami Barends, Sergio Boixo, et al. Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nature Physics*, 17(3):332–336, 2021.

[168] Zhang Jiang, Eleanor G Rieffel, and Zhihui Wang. Near-optimal quantum circuit for grover's unstructured search using a transverse field. *Physical Review A*, 95(6):062317, 2017.

[169] Zhihui Wang, Nicholas C Rubin, Jason M Dominy, and Eleanor G Rieffel. Xy mixers: Analytical and numerical results for the quantum alternating operator ansatz. *Physical Review A*, 101(1):012320, 2020.

[170] JB Wang. A quantum walk-assisted approximate algorithm for bounded np optimisation problems. *Quantum Information Processing*, 18(3):61, 2019.

[171] Combinatorial optimization via highly efficient quantum walks. *Physical Review Research*, 2(2):023302, 2020.

[172] Tavis Bennett, Edric Matwiejew, Sam Marsh, and Jingbo B Wang. Quantum walk-based vehicle routing optimisation. *Frontiers in Physics*, 9:730856, 2021.

[173] Nicholas Slate, Edric Matwiejew, Samuel Marsh, and Jingbo B Wang. Quantum walk-based portfolio optimisation. *Quantum*, 5:513, 2021.

[174] Edric Matwiejew, Jason Pye, and Jingbo B Wang. Quantum optimisation for continuous multivariable functions by a structured search. *Quantum Science and Technology*, 8(4):045013, 2023.

[175] Panagiotis Kl Barkoutsos, Giacomo Nannicini, Anton Robert, Ivano Tavernelli, and Stefan Woerner. Improving variational quantum optimization using cvar. *Quantum*, 4:256, 2020.

[176] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to State Preparation and Variational Optimization from Symmetry

Protection. *Physical Review Letters*, 125(26), Oct 2019. doi: 10.1103/PhysRevLett.125.260505. URL http://arxiv.org/abs/1910.08980http://dx.doi.org/10.1103/PhysRevLett.125.260505.

[177] Tameem Albash and Daniel A Lidar. Adiabatic quantum computation. *Reviews of Modern Physics*, 90(1):015002, 2018.

[178] Avrim Blum, Merrick Furst, Jeffrey Jackson, Michael Kearns, Yishay Mansour, and Steven Rudich. Weakly learning dnf and characterizing statistical query learning using fourier analysis. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 253–262, 1994.

[179] Ricard Gavaldà, Gabor Lugosi, Thomas Zeugmann, and Sandra Zilles. *Algorithmic Learning Theory: 20th International Conference, ALT 2009, Porto, Portugal, October 3-5, 2009, Proceedings*, volume 5809. Springer, 2009.

[180] Surbhi Goel, Aravind Gollakota, and Adam Klivans. Statistical-query lower bounds via functional gradients. *Advances in Neural Information Processing Systems*, 33:2147–2158, 2020.

[181] Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Failures of gradient-based deep learning. In *International Conference on Machine Learning*, pages 3067–3075. PMLR, 2017.

[182] Dominik Janzing, Pawel Wocjan, and Thomas Beth. " non-identity-check" is qma-complete. *International Journal of Quantum Information*, 3(03):463–473, 2005.

[183] Yunseong Nam, Neil J Ross, Yuan Su, Andrew M Childs, and Dmitri Maslov. Automated optimization of large quantum circuits with continuous parameters. *npj Quantum Information*, 4(1):23, 2018.

[184] Loc Quang Trinh. *Greedy layerwise training of convolutional neural networks*. PhD thesis, Massachusetts Institute of Technology, 2019.

[185] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19, 2006.

[186] Hannes Schulz and Sven Behnke. Deep learning: Layer-wise learning of feature hierarchies. *KI-Künstliche Intelligenz*, 26:357–363, 2012.

[187] Qi Teng, Kun Wang, Lei Zhang, and Jun He. The layer-wise training convolutional neural networks using local loss for sensor-based human activity recognition. *IEEE Sensors Journal*, 20(13):7265–7274, 2020.

[188] Hugo Larochelle, Yoshua Bengio, Jérôme Louradour, and Pascal Lamblin. Exploring strategies for training deep neural networks. *Journal of machine learning research*, 10(1), 2009.

[189] Wim Lavrijsen, Ana Tudor, Juliane Müller, Costin Iancu, and Wibe De Jong. Classical optimizers for noisy intermediate-scale quantum devices. In *2020 IEEE international conference on quantum computing and engineering (QCE)*, pages 267–277. IEEE, 2020.

[190] Michael Streif and Martin Leib. Training the quantum approximate optimization algorithm without access to a quantum processing unit. *Quantum Science and Technology*, 5(3):034008, 2020.

[191] Arthur Pesah, Marco Cerezo, Samson Wang, Tyler Volkoff, Andrew T Sornborger, and Patrick J Coles. Absence of barren plateaus in quantum convolutional neural networks. *Physical Review X*, 11(4):041011, 2021.

[192] Kaining Zhang, Min-Hsiu Hsieh, Liu Liu, and Dacheng Tao. Toward trainability of quantum neural networks. *arXiv preprint arXiv:2011.06258*, 2020.

[193] Marco Cerezo, Martin Larocca, Diego García-Martín, Nelson L Diaz, Paolo Braccia, Enrico Fontana, Manuel S Rudolph, Pablo Bermejo, Aroosa Ijaz, Supanut Thanasilp, et al. Does provable absence of barren plateaus imply classical simulability? or, why we need to rethink variational quantum computing. *arXiv preprint arXiv:2312.09121*, 2023.

[194] Yunseong Nam, Yuan Su, and Dmitri Maslov. Approximate quantum fourier transform with o (n log (n)) t gates. *NPJ Quantum Information*, 6(1):26, 2020.

[195] David G Cory, MD Price, W Maas, Emanuel Knill, Raymond Laflamme, Wojciech H Zurek, Timothy F Havel, and Shyamal S Somaroo. Experimental quantum error correction. *Physical Review Letters*, 81(10):2152, 1998.

[196] Adam Paetznick and Ben W Reichardt. Universal fault-tolerant quantum computation with only transversal gates and error correction. *Physical review letters*, 111(9):090505, 2013.

[197] Jérémie Guillaud and Mazyar Mirrahimi. Repetition cat qubits for fault-tolerant quantum computation. *Physical Review X*, 9(4):041053, 2019.

[198] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.

[199] David Wierichs, Josh Izaac, Cody Wang, and Cedric Yen-Yu Lin. General parameter-shift rules for quantum gradients. *Quantum*, 6:677, 2022.

[200] Jonas Stein, Navid Roshani, Maximilian Zorn, Philipp Altmann, Michael Kölle, and Claudia Linnhoff-Popien. Improving parameter training for vqes by sequential hamiltonian assembly. *arXiv preprint arXiv:2312.05552*, 2023.

[201] Ali Rad, Alireza Seif, and Norbert M Linke. Surviving the barren plateau in variational quantum circuits with bayesian learning initialization. *arXiv preprint arXiv:2203.02464*, 2022.

[202] Xiaoyuan Liu, Anthony Angone, Ruslan Shaydulin, Ilya Safro, Yuri Alexeev, and Lukasz Cincio. Layer vqe: A variational approach for combinatorial optimization on noisy quantum computers. *IEEE Transactions on Quantum Engineering*, 3:1–20, 2022.

[203] Junxiang Xiao, Jingwei Wen, Shijie Wei, and Guilu Long. Reconstructing unknown quantum states using variational layerwise method. *Frontiers of Physics*, 17(5):51501, 2022.

[204] Michael JD Powell. *A direct search optimization method that models the objective and constraint functions by linear interpolation*. Springer, 1994.

[205] OV Borzenkova, GI Struchalin, AS Kardashin, VV Krasnikov, NN Skryabin, SS Straupe, SP Kulik, and JD Biamonte. Variational simulation of schwinger's hamiltonian with polarization qubits. *Applied Physics Letters*, 118(14):144002, 2021.

[206] Alexander Pechen. Engineering arbitrary pure and mixed quantum states. *Physical Review A*, 84(4):042106, 2011.

[207] Dimitris Bertsimas and John Tsitsiklis. Simulated annealing. *Statistical science*, 8(1):10–15, 1993.

[208] Peter JM Van Laarhoven, Emile HL Aarts, Peter JM van Laarhoven, and Emile HL Aarts. *Simulated annealing*. Springer, 1987.

[209] Xinwei Lee, Ningyi Xie, Dongsheng Cai, Yoshiyuki Saito, and Nobuyoshi Asai. A depth-progressive initialization strategy for quantum approximate optimization algorithm. *Mathematics*, 11(9):2176, 2023.

[210] Xinwei Lee, Xinjian Yan, Ningyi Xie, Dongsheng Cai, Yoshiyuki Saito, and Nobuyoshi Asai. Iterative layerwise training for the quantum approximate optimization algorithm. *Physical Review A*, 109(5):052406, 2024.

[211] Mauro ES Morales, Timur Tlyachev, and Jacob Biamonte. Variational learning of grover's quantum search algorithm. *Physical Review A*, 98(6):062333, 2018.

[212] Samuel Marsh and JB Wang. Deterministic spatial search using alternating quantum walks. *Physical Review A*, 104(2):022216, 2021.

[213] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106*, 2000.

[214] Andrew M Childs and Jeffrey Goldstone. Spatial search by quantum walk. *Physical Review A*, 70(2):022314, 2004.

[215] Andrew M Childs. On the relationship between continuous-and discrete-time quantum walk. *Communications in Mathematical Physics*, 294:581–603, 2010.

[216] Edward Farhi and Sam Gutmann. Quantum computation and decision trees. *Physical Review A*, 58(2):915, 1998.

[217] Andrew M Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A Spielman. Exponential algorithmic speedup by a quantum walk. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 59–68, 2003.

[218] Dengke Qu, Samuel Marsh, Kunkun Wang, Lei Xiao, Jingbo Wang, and Peng Xue. Deterministic search on star graphs via quantum walks. *Physical review letters*, 128(5):050501, 2022.

[219] Andrew M Childs. Universal computation by quantum walk. *Physical review letters*, 102(18):180501, 2009.

[220] Edric Matwiejew and Jingbo B Wang. Quop_mpi: A framework for parallel simulation of quantum variational algorithms. *Journal of Computational Science*, 62:101711, 2022.

[221] Andrew M Childs, Enrico Deotto, Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Andrew J Landahl. Quantum search by measurement. *Physical Review A*, 66(3):032314, 2002.

[222] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[223] James G Morley, Nicholas Chancellor, Sougato Bose, and Viv Kendon. Quantum search with hybrid adiabatic–quantum-walk algorithms and realistic noise. *Physical review A*, 99(2):022339, 2019.

[224] Fernando GSL Brandao, Michael Broughton, Edward Farhi, Sam Gutmann, and Hartmut Neven. For fixed control parameters the quantum approximate optimization algorithm's objective function value concentrates for typical instances. *arXiv preprint arXiv:1812.04170*, Dec 2018.

[225] Alexey Galda, Xiaoyuan Liu, Danylo Lykov, Yuri Alexeev, and Ilya Safro. Transferability of optimal qaoa parameters between random graphs. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 171–180. IEEE, 2021.

[226] Tavis Bennett, Lyle Noakes, and Jingbo B Wang. Analysis of the non-variational quantum walk-based optimisation algorithm. *arXiv preprint arXiv:2408.06368*, 2024.

[227] Tavis Bennett, Lyle Noakes, and Jingbo Wang. Non-variational quantum combinatorial optimisation. *arXiv preprint arXiv:2404.03167*, 2024.

[228] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H Booth, et al. The variational quantum eigensolver: a review of methods and best practices. *Physics Reports*, 986:1–128, 2022.

[229] Michael Broughton, Guillaume Verdon, Trevor McCourt, Antonio J Martinez, Jae Hyeon Yoo, Sergei V Isakov, Philip Massey, Murphy Yuezhen Niu, Ramin Halavati, Evan Peters, et al. Tensorflow quantum: A software framework for quantum machine learning. *arXiv preprint arXiv:2003.02989*, 2020.

[230] Christof Zalka. Grover's quantum searching algorithm is optimal. *Physical Review A*, 60(4):2746, 1999.

[231] Richard M Karp. *Reducibility among combinatorial problems*. Springer, 2010.

Appendix A

Relevant classical computing problems

This appendix explains the classical computing problems and some of their properties relevant to this thesis.

A.1 Unstructured search

Definition 22 (Unstructured search) *Consider an unstructured database S indexed by $\{0,1\}^{\times n}$. Let $f : \{0,1\}^{\times n} \to \{0,1\}$ be a Boolean function (aka. black box) such that:*

$$f(j) = \begin{cases} 1 & \textit{iff } j = \omega \\ 0 & \textit{otherwise.} \end{cases} \tag{A.1}$$

Find $\omega \in \{0,1\}^{\times n}$.

Definition 23 (Query complexity) *Consider a algorithm that uses a black box f. The query complexity of the algorithm is defined as the number of evaluations of f required for the termination of the algorithm.*

Remark 5 (Query complexity of classic unstructured search) *Given an unstructured database S of size $N = 2^n$, the query complexity of a search algorithm in S is $\frac{N+1}{2}$ on average.*

A.1.1 Grover's algorithm

Grover's algorithm is a quantum algorithm for unstructured search invented in 1996 by Lov Grover [23], which has quadratic speedup over classical algorithms.

The algorithm starts with a state in uniform superposition $|s\rangle = |+\rangle^{\otimes n}$ to which two operators are applied iteratively in alternating fashion:

1. $U_\omega = \mathbb{1} - 2|\omega\rangle\langle\omega|$, where $|\omega\rangle$ is the target state in the computational basis, acts as
$$U_\omega |j\rangle = (-1)^{f(j)} |j\rangle . \tag{A.2}$$

2. $U_s = 2|s\rangle\langle s| - \mathbb{1}$, when applied to a n arbitrary state $|\psi\rangle = \sum_{j=\{0,1\}^{\times n}} a_j |j\rangle$ acts as
$$U_s |\psi\rangle = \sum_{j=\{0,1\}^{\times n}} (2\langle a\rangle - a_j) |j\rangle , \tag{A.3}$$
where $\langle a\rangle = \frac{\sum a_j}{N}$, which is sometimes called a reflection around the amplitudes mean.

In order to maximize the probability of measuring $|\omega\rangle$, the algorithm requires $\sim \frac{\pi}{4}\sqrt{N}$ iterations.

Remark 6 (Query complexity of Grover's algorithm) *Consider the plane spanned by $|\omega\rangle$ and $|s'\rangle = \frac{1}{\sqrt{N-1}} \sum_{j\neq\omega} |j\rangle$. The initial state $|s\rangle$ lies in this plane and can be expressed as*
$$|s\rangle = a|s'\rangle + b|\omega\rangle , \tag{A.4}$$
where $|a|^2 + |b|^2 = 1$. The operator $U_\omega = \mathbb{1} - 2|\omega\rangle\langle\omega|$ is then a reflection on the hyper plane orthogonal to $|\omega\rangle$, and $U_s = 2|s\rangle\langle s| - \mathbb{1}$ is a reflection through $|s\rangle$.

$$\begin{aligned} U_s U_\omega |\omega\rangle &= -U_s |\omega\rangle && \text{(A.5)}\\ &= (\mathbb{1} - 2|s\rangle\langle s|) |\omega\rangle && \text{(A.6)}\\ &= |\omega\rangle - \frac{2}{\sqrt{N}} |s\rangle && \text{(A.7)}\\ &= \frac{N-2}{N} |\omega\rangle - \frac{2\sqrt{N-1}}{N} |s'\rangle && \text{(A.8)} \end{aligned}$$

similarly

$$U_s U_\omega \left|s'\right\rangle = \frac{N-2}{N}\left|s'\right\rangle + \frac{2\sqrt{N-1}}{N}\left|\omega\right\rangle . \tag{A.9}$$

Thus in the subspace spaned by $|s'\rangle$ *and* $|\omega\rangle$, $U_s U_\omega$ *is equivalent to*

$$U_s U_\omega = \begin{pmatrix} \frac{N-2}{N} & -\frac{2\sqrt{N-1}}{N} \\ \frac{2\sqrt{N-1}}{N} & \frac{N-2}{N} \end{pmatrix} . \tag{A.10}$$

This can be written as aplying a rotation with $\cos(\theta) = \frac{N-2}{N}$, *and* $\sin(\theta) = \frac{2\sqrt{N-1}}{N}$, *where one can see that* $\theta = 2\arcsin\left(\frac{1}{\sqrt{N}}\right)$. *The initial state* $|s\rangle$ *can be written in terms of* θ *as*

$$\left|s\right\rangle = \sin(\theta/2)\left|\omega\right\rangle + \cos(\theta/2)\left|s'\right\rangle . \tag{A.11}$$

After r *iterations the state is*

$$(U_s U_\omega)^r \left|s\right\rangle = \sin((r+1/2)\theta)\left|\omega\right\rangle + \cos((r+1/2)\theta)\left|s'\right\rangle , \tag{A.12}$$

and the probability of measuring $|\omega\rangle$ *is*

$$\left|\left\langle\omega\right|(U_s U_\omega)^r\left|s\right\rangle\right|^2 = \sin^2((1/2+r)\theta), \tag{A.13}$$

which is maximal for

$$\sin^2((1/2+r)\theta) = 1 \tag{A.14}$$

$$(1/2+r)\theta = \pi/2 \tag{A.15}$$

$$r = \frac{\pi}{2\theta} - \frac{1}{2} \tag{A.16}$$

$$r = \frac{\pi}{4\arcsin\left(\frac{1}{\sqrt{N}}\right)} - \frac{1}{2}. \tag{A.17}$$

For $N \gg 1$, $\arcsin\left(\frac{1}{\sqrt{N}}\right) \approx \frac{1}{\sqrt{N}}$, *thus* (A.17) *becomes* $r \approx \frac{\pi}{4}\sqrt{N}$, *which was proven optimal in [230].*

A.2 Combinatorial optimization problems

A combinatorial optimization problem consists on finding the optimal object from a finite set of objects. Since the problems explained in this section have instances that can be represented as pseudo-Boolean functions, $f : \mathbb{B}^{\times n} \to \mathbb{R}$, the optimization problem is to determine

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}\in\mathbb{B}^{\times n}}{\operatorname{argmin}} f(\boldsymbol{x}) \tag{A.18}$$

or equivalently the value $f(\boldsymbol{x}^*)$.

A.2.1 Satisfiability

Satisfiability, or SAT, was the first problem proved to be NP-complete [15–17], that is, all instances from problems in the complexity class NP can be reduced to SAT instances in polynomial number of steps, and verified in polynomial time. The problem consists on determining if a given Boolean formula written in its conjunctive normal form (CNF) can evaluate to true by a variable assignment. If the number of literals in each clause of the Boolean formula is equal to k, it forms a k-SAT instance,

$$f(x_1, x_2, \cdots, x_n) = \bigwedge_{l=1}^{m} \left(\bigvee_{j=1}^{k} x_{l,j} \right), \tag{A.19}$$

where $x_{l,j} \in \{x_1, x_2, \cdots, x_n, \neg x_1, \neg x_2, \cdots, \neg x_n\}$, $\neg$ is the logical negation, and m is the number of clauses. Like the standard SAT, k-SAT is also in NP-complete for $k \geq 3$. The optimization version of k-SAT, named MAX-k-SAT, consists on satisfying as many clauses as possible.

Definition 24 (MAX-k-SAT) *Given a k-SAT instance, the optimization version of it replaces the conjunction of clauses of (A.19) with a summation as*

$$g(\boldsymbol{x}) = \sum_{l=1}^{m} \bigvee_{j=1}^{k} x_{l,j}. \tag{A.20}$$

The objective is then to determine

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\mathbb{B}^{\times n}} g(\boldsymbol{x}). \tag{A.21}$$

MAX-k-SAT is the canonical NP-hard problem, that is, any instance of an NP problem can be reduced to MAX-k-SAT in polynomial number of steps [231].

In order to create a problem Hamiltonian that encodes a k-SAT instance, corresponding to (A.19), we use the following mapping

$$\begin{aligned} x_j \to P_j^1, \quad \neg x_j \to P_j^0, \\ \vee \to \otimes, \quad \wedge \to +, \end{aligned} \tag{A.22}$$

where $P_j^0 = |0\rangle\langle 0|_j$, $P_j^1 = |1\rangle\langle 1|_j$ are projectors acting on the jth qubit. Notice that the projectors can be written in terms of Pauli matrices as

$$P_j^\alpha = \frac{1}{2}(\mathbb{1} + (-1)^\alpha Z_j). \tag{A.23}$$

Thus for a 2-SAT instance, its corresponding problem Hamiltonian can be represented in the form

$$H = \sum_{j<l} J_{j,l} Z_j Z_l + \sum_j h_j Z_j, \tag{A.24}$$

where $J_{j,l}, h_j \in \mathbb{R}$.

A.2.2 MAX-CUT

Given an undirected graph, MAX-CUT consists on partitioning the nodes into two complementary subsets such that the number of edges across subsets, called cut value, is maximized. Like MAX-k-SAT, MAX-CUT is an NP-hard problem.

Let G be a graph $G = (V, E)$, where V are vertices ($|V| = n$) and E edges. The vertices are represented by the variables $\{x_j\}_{j=1}^n$, and will take values $x_j = 0$ if it is in subset S, or $x_j = 1$ if it is in the complementary subset S'. For an edge $\langle j, k\rangle \in E$, the function

$$g(x_j, x_l) = x_j + x_l - 2x_j x_l, \tag{A.25}$$

will be 1 when $x_j \neq x_l$, and 0 otherwise. Then, summing over all the edges gives the pseudo-Boolean function we look to maximize.

Definition 25 (MAX-CUT) *Given an undirected graph $G = (V, E)$, where V are vertices ($|V| = n$) and E edges, the cut value given an assignment $\boldsymbol{x} \in \mathbb{B}^{\times n}$ is*

$$g(\boldsymbol{x}) = \sum_{\langle j,l \rangle \in E} x_j + x_l - 2x_j x_l. \tag{A.26}$$

The objective is then to determine

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x} \in \mathbb{B}^{\times n}} g(\boldsymbol{x}). \tag{A.27}$$

In order to create a Hamiltonian that encodes a MAX-CUT instance, corresponding to (A.26), we use the mapping

$$x_j \to \frac{1}{2}(\mathbb{1} + Z_j), \tag{A.28}$$

which results in the Hamiltonian

$$H = \frac{1}{2} \sum_{\langle j,l \rangle \in E} (\mathbb{1} - Z_j Z_l). \tag{A.29}$$

Notice that the state that maximizes (A.29) is the same that minimizes

$$H' = \sum_{\langle j,l \rangle \in E} Z_j Z_l. \tag{A.30}$$

Appendix B

Properties of unstructured search in a CTQW

B.1 Optimal value of α for unstructured search in a CTQW

Let $H = \alpha H_x + |\mathbf{0}\rangle\langle\mathbf{0}|$, where $|\mathbf{0}\rangle \equiv |0\rangle^{\otimes n}$. We seek to find α for which the energy gap Δ is minimum [213].

Let $|e_k\rangle$ be the Dicke states as in Definition 21, and $|h_k\rangle = H_a^{\otimes n} |e_k\rangle$ where H_a is the Hadamard gate. It can be seen that

$$\left(\sum_{i=1}^{n} Z_i\right) |e_k\rangle = (n - 2k) |e_k\rangle , \tag{B.1}$$

$$H_x |h_k\rangle = (n - 2k) |h_k\rangle . \tag{B.2}$$

We solve for the eigenvalues of H, $H |\psi\rangle = E |\psi\rangle$. Multiplying it by $\langle h_k|$,

$$\langle h_k| H |\psi\rangle = E \langle h_k|\psi\rangle , \tag{B.3}$$

substituting H and using property (B.2) after a series of algebraic manipulations we

end up with (B.6):

$$\alpha(n-2k)\langle h_k|\psi\rangle + \langle h_k|\mathbf{0}\rangle\langle\mathbf{0}|\psi\rangle = E\langle h_k|\psi\rangle, \tag{B.4}$$

$$[E-\alpha(n-2k)]\langle h_k|\psi\rangle = \langle h_k|\mathbf{0}\rangle\langle\mathbf{0}|\psi\rangle, \tag{B.5}$$

$$\langle h_k|\psi\rangle = \frac{\langle h_k|\mathbf{0}\rangle\langle\mathbf{0}|\psi\rangle}{E-\alpha(n-2k)}. \tag{B.6}$$

We multiply both sides by $\langle\mathbf{0}|h_k\rangle$ and sum over k,

$$\sum_{k=0}^{n}\langle\mathbf{0}|h_k\rangle\langle h_k|\psi\rangle = \sum_{k=0}^{n}\frac{|\langle h_k|\mathbf{0}\rangle|^2\langle\mathbf{0}|\psi\rangle}{E-\alpha(n-2k)}, \tag{B.7}$$

$$1 = \sum_{k=0}^{n}\frac{P_k}{E-\alpha(n-2k)}, \tag{B.8}$$

where we used the fact that $|h_k\rangle$ form a basis in the symmetric subspace. Here $P_k = |\langle h_k|o\rangle|^2 = \frac{C_n^k}{2^n}$. We introduce a change of variables

$$\lambda = E/\alpha \implies \alpha = \sum_{k=0}^{n}\frac{P_k}{\lambda-(n-2k)}. \tag{B.9}$$

Let $\alpha > 0$, the right hand side of (B.9) approaches $+\infty$ as $\lambda \to n+0, n-2+0, \cdots -n+0$, and goes to $-\infty$ as $\lambda \to n-0, n-2-0, \cdots -n-0$. We will prove that there exist two roots exponentially close to $\lambda = n$ for

$$\alpha^* = \frac{1}{2}\sum_{k=1}^{n}\frac{P_k}{k}. \tag{B.10}$$

Substituting α^* from (B.10) to (B.9), we get

$$\frac{1}{2}\sum_{k=1}^{n}\frac{P_k}{k} = \sum_{k=0}^{n}\frac{P_k}{\lambda+2k-n} \iff \sum_{k=1}^{n}\frac{P_k(\lambda-n)}{2k(\lambda+2k-n)} = \frac{P_0}{\lambda-n} = \frac{1}{2^n(\lambda-n)}. \tag{B.11}$$

As we search for $|\lambda-n| << 1$, we neglect the term in the denominator of the LHS of (B.11),

$$\frac{1}{4}\sum_{k=1}^{n}\frac{P_k}{k^2} = \frac{1}{2^n(\lambda-n)^2} \implies \lambda = n \pm \xi, \tag{B.12}$$

where $\xi = \frac{2}{\sqrt{2^n}} \left(\sum_{k=1}^{n} \frac{P_k}{k^2}\right)^{-\frac{1}{2}}$. The energy gap is then

$$\Delta = 2\alpha^* \xi, \tag{B.13}$$

which is correct up to exponential precision $O(2^{-n})$. However, to obtain a tractable expression we approximate sums in the expression for α^* and ξ

$$\sum_{k=0}^{n} \frac{P_k}{k} = \frac{2}{n} + O\left(\frac{1}{n^2}\right), \ \sum_{k=0}^{n} \frac{P_k}{k^2} = \frac{4}{n^2} + O\left(\frac{1}{n^3}\right), \tag{B.14}$$

allowing us to conclude

$$\implies \Delta = \frac{2}{\sqrt{2^n}} \left(1 + O\left(\frac{1}{n}\right)\right). \tag{B.15}$$

B.2 Low energy eigenstates of unstructured search in a CTQW

From equations (B.6) and (B.10) we have,

$$\alpha^* \langle h_k|\psi\rangle = \frac{\sqrt{P_k}\,\langle \mathbf{0}|\psi\rangle}{\lambda - n + 2k}, \tag{B.16}$$

which for $k = 0$ simplifies to

$$\alpha^* \langle h_0|\psi\rangle = \pm \frac{\sqrt{P_0}}{\xi} \langle \mathbf{0}|\psi\rangle . \tag{B.17}$$

Note that $|h_0\rangle = |+\rangle^{\otimes n}$. For $k \neq 0$ exponentially small $\lambda - n$ can be neglected, giving

$$\alpha^* \langle h_k|\psi\rangle = \pm \frac{\sqrt{P_k}}{2k} \langle \mathbf{0}|\psi\rangle . \tag{B.18}$$

$$\sum_{k=0}^{n} |\langle h_k|\psi\rangle|^2 = 1 = \frac{|\langle \mathbf{0}|\psi\rangle|^2}{(\alpha^*)^2} \left(\frac{P_0}{\xi^2} + \sum_{k=1}^{n} \frac{P_k}{4k^2}\right) \tag{B.19}$$

Thus,

$$(\alpha^*)^2 = |\langle \mathbf{0}|\psi\rangle|^2 \left(\frac{P_0}{\xi^2} + \sum_{k=1}^{n} \frac{P_k}{4k^2}\right) = |\langle \mathbf{0}|\psi\rangle|^2 \left(\frac{2}{2^n \xi^2}\right) \tag{B.20}$$

Where we used equation (B.11) at the last transition. Finally, we conclude

$$\langle \mathbf{0}|\psi_{\pm}\rangle = \pm\frac{\sqrt{2^n}\Delta}{2\sqrt{2}}. \tag{B.21}$$

From equations (B.17) and (B.21)

$$\langle h_0|\psi_{\pm}\rangle = \frac{1}{\sqrt{2}} + O(2^{-n}), \tag{B.22}$$

Similarly using (B.18), (B.21) for $k \neq 0$,

$$\langle h_k|\psi_{\pm}\rangle = \pm\frac{\sqrt{P_k}}{k\sqrt{2}}\left(\sum_{k=0}^{n}\frac{P_k}{k^2}\right)^{-\frac{1}{2}} + O(2^{-n}), \tag{B.23}$$

From equations (B.22) and (B.23) we obtain the eigenstates

$$|\psi_{\pm}\rangle = \frac{1}{\sqrt{2}}\left(|+\rangle^{\otimes n} \pm \left(\sum_{k=0}^{n}\frac{P_k}{k^2}\right)^{-\frac{1}{2}}\sum_{k=1}^{n}\frac{\sqrt{P_k}}{k}|h_k\rangle\right) + O(2^{-n}). \tag{B.24}$$

Note, the second therm has high overlap with $|\mathbf{0}\rangle$

$$\left(\sum_{k=0}^{n}\frac{P_k}{k^2}\right)^{-\frac{1}{2}}\langle \mathbf{0}|\sum_{k=1}^{n}\frac{\sqrt{P_k}}{k}|h_k\rangle = \left(\sum_{k=0}^{n}\frac{P_k}{k^2}\right)^{-\frac{1}{2}}\sum_{k=1}^{n}\frac{P_k}{k} \tag{B.25}$$

$$= \left(\frac{4}{n^2} + O\left(n^{-3}\right)\right)^{-\frac{1}{2}}\left(\frac{2}{n} + O\left(n^{-2}\right)\right) \tag{B.26}$$

$$= 1 + O\left(n^{-1}\right), \tag{B.27}$$

which justifies use of expressions (4.4). Notice, however, that while these expressions are only polynomially correct, this imprecision comes from approximating the second part of (B.24) with state $|\mathbf{0}\rangle$. The original form of the eigenstates (B.24) remains exponentially precise.

B.3 QAOA sequence from a trotterized CTQW

The second order Trotter sequence for the evolution of a Hamiltonian with two terms $H = H_0 + H_1$ is

$$S_2(t) = e^{-iH_1\frac{t}{2}}e^{-iH_0\frac{t}{2}}e^{-iH_0\frac{t}{2}}e^{-iH_1\frac{t}{2}} \tag{B.28}$$

$$= e^{-iH_1\frac{t}{2}}e^{-iH_0 t}e^{-iH_1\frac{t}{2}}. \tag{B.29}$$

Similarly, after grouping neighboring terms, higher order Suzuki sequences take the form

$$S_q(t) = \left(\prod_{v=1}^{5^{q/2-1}} e^{-iH_1 t a(v)}e^{-iH_0 t b(v)}\right) e^{-iH_1 tc}, \tag{B.30}$$

where $a, b, c \in \mathbb{R}$. In the case of unstructured search, by setting $H_0 = |0\rangle\langle 0|^{\otimes n}$, $H_1 = H_x\alpha^*$, a step $S_q(t/r)$ takes the form

$$S_q(t/r)\,|+\rangle^{\otimes n} = e^{-i\frac{\alpha^* ntc}{r}} \prod_{v=1}^{5^{q/2-1}} e^{-iH_x\frac{\alpha^* t a(v)}{r}}e^{-iH_0\frac{tb(v)}{r}}\,|+\rangle^{\otimes n}\,, \tag{B.31}$$

which has $\Upsilon_q = 5^{q/2-1}$ stages. Using r steps in the sequence (4.18), and grouping terms in neighboring steps one gets

$$\begin{aligned} S_q^r(t/r)\,|+\rangle^{\otimes n} &= \left\{\left(\prod_{v=1}^{5^{q/2-1}} e^{-iH_x\frac{\alpha^* t a(v)}{r}}e^{-iH_0\frac{tb(v)}{r}}\right) e^{-iH_x\frac{\alpha^* tc}{r}}\right\}^r |+\rangle^{\otimes n} \\ &= e^{-i\alpha^* ntc'} \prod_{v=1}^{r5^{q/2-1}} e^{-iH_x\alpha^* t a'(v)}e^{-iH_0 t b'(v)}\,|+\rangle^{\otimes n}\,, \end{aligned} \tag{B.32}$$

with $a', b', c' \in \mathbb{R}$, which recovers a QAOA sequence of depth $p = r\Upsilon_q = r5^{q/2-1}$.

Appendix C

Numerical details

C.1 Simulating QAOA from a trotterized CTQW

The numerics presented in Chapter 4 were performed by simulating the QAOA circuits in the $n+1$ dimensional symmetric subspace. Due to the QAOA angles repeating across Trotter steps, we simulate these circuit by calculating powers of a step $S_q(t/r)$, which can be performed efficiently.

In order to numerically approximate the optimal depth for a given n, ϵ and q we:

1. Define the number of steps as $r_{dl} = d \cdot 2^{n/2-l}$, where d and l are integers initially set to $d = 1$ and $l = 0$.

2. Numerically calculate $S_q^{r_{dl}}(t/r_{dl})$ and iteratively increase the value of d one by one until finding $d = d'$ such that:

$$|\langle 0|^{\otimes n} S_q^{r_{d'l}}(t/r_{d'l}) |+\rangle^{\otimes n}|^2 \geq |\langle 0|^{\otimes n} U(\alpha^*, t^*) |+\rangle^{\otimes n}|^2 - \epsilon. \tag{C.1}$$

3. Once condition (C.1) is fulfilled we perform binary search. This is done by setting the new initial value of $d \to 2d' - 1$ and $l \to l + 1$, and repeating step 2.

4. Steps 2 and 3 get repeated for a fixed number of iterations. Specifically for the numerics presented in the manuscript, we used 15 iterations.

5. The resulting approximated optimal depth is given by $p_{\text{numerical}} = r_{dl} \cdot 5^{q/2-1}$.

The codes used in this thesis are written in Python and are available on reasonable request.