The Quantum Approximate Optimization Algorithm - Perspectives of Analog Compilation and Incoherent Evolution

Dissertation

zur Erlangung des Grades

des Doktors der Naturwissenschaften

der Naturwissenschaftlich - Technischen Fakultät

der Universität des Saarlandes

von

Thorge Müller

Saarbrücken

2023

Tag des Kolloquiums:	23. Februar 2024
Dekan:	Prof. Dr. Ludger Santen
Berichterstatter:	Prof. Dr. Frank Wilhelm-Mauch Dr. Mikel Sanz
Akad. Mitglied:	Dr. Philipp Hövel
Vorsitz:	Prof. Dr. Peter P. Orth

"The goal is nothing other than the coherence and completeness of the system not only in respect of all details, but also in respect of all physicists of all places, all times, all peoples, and all cultures."

Max Planck

Abstract

The current challenges of quantum computing development pertain to mitigating the effect of noise on the device. The problems of this era, in which Noisy Intermediate-Scale Quantum (NISQ) computers must be used in the absence of error-correcting schemes, are the focus of this thesis.

In one theme of the thesis, we investigate the drop in performance incurred by Quantum Approximate Optimization Algorithm (QAOA) applied to constraint optimization problems Max-kSAT and Max-kXOR, finding that significant changes in performance occur for increasing the number of literals k per constraint. We also investigate the use of annealing-inspired schedules for QAOA, demonstrating that linear schedules outperform those of Trotterized Quantum Annealing.

A second theme of the thesis concerns the co-design of devices for QAOA. Firstly, we consider the tradeoffs in the decomposition of ZZ-generated gates into CZ- and CNOT-gates, with depolarizing and coherent errors affecting each decomposition differently. We find analytical and numerical evidence that both decompositions attain comparable gate fidelities for low noise. We investigate QAOA in the digital-analog scheme, in which individual control of two-qubit gates is relinquished in favor of a global interaction, with device control occurring only via single-qubit gates. We demonstrate that QAOA in this scheme produces the same results as its digital counterpart for fast single-qubit gates.

Zusammenfassung

In der aktuellen Entwicklung des Quantencomputings geht es darum das Rauschen auf der Hardware zu minimieren. Die Probleme dieser Ära, in der Noisy Intermediate-Scale Quantum (NISQ) Computer ohne Fehlerkorrekturverfahren verwendet werden, stehen im Fokus dieser Arbeit.

In einem Schwerpunkt der Arbeit untersuchen wir den Leistungsabfall vom Quantum Approximate Optimization Algorithm (QAOA) für die Optimierungsprobleme Max-kSAT und Max-kXOR. Dabei stellen wir fest, dass signifikante Veränderungen in der Leistung auftreten, wenn die Anzahl der Literale k zunimmt. Des Weiteren untersuchen wir die annealing-inspirierten Initialisierungen für QAOA und zeigen, dass lineare Initialisierungen das Trotterized Quantum Annealing übertreffen.

Ein zweiter Schwerpunkt der Arbeit ist das Co-Design von Hardware für den QAOA. Zunächst betrachten wir ZZ-generierte Gatter in CZ- und CNOT-Gatter Zerlegung, wobei depolarisierende und kohärente Fehler die einzelnen Zerlegungen unterschiedlich beeinflussen. Wir finden analytische und numerische Hinweise darauf, dass beide Zerlegungen vergleichbare Gattertreue bei geringem Rauschen erreichen. Des Weiteren untersuchen wir den QAOA im digital-analogen Ansatz, bei dem die individuelle Steuerung von Zwei-Qubit Gattern zugunsten einer globalen Interaktion aufgegeben wird und die Hardware-Steuerung nur über Ein-Qubit Gatter erfolgt. Wir zeigen, dass der QAOA in diesem Ansatz für schnelle Ein-Qubit Gatter vergleichbare Ergebnisse erzielt wie seine digitale Ausführung.

Publication List

Published

- D. Headley, T. Müller, A. Martin, E. Solano, M. Sanz, F.K. Wilhelm
  Approximating the quantum approximate optimization algorithm with digital-analog interactions
  Phys. Rev. A 106, 042446 (2022)

Submitted

- T. Müller, T. Stollenwerk, D. Headley, M. Epping, F.K. Wilhelm
  Coherent and non-unitary errors in ZZ-generated gates
  arXiv:2304.14212

Acknowledgements

First, I want to thank Frank Wilhelm-Mauch and Achim Basermann for allowing me to research quantum computing at the University of Saarland in collaboration with the German Aerospace Center. Frank Wilhelm-Mauch motivated and inspired me for my research topics. Thank you also to my supervisors at the German Aerospace Center, Michael Epping and Tobias Stollenwerk. Of course, there are many people to mention, like David Headley and Gino Bishop, who were doing their Ph.D. at the same time. A special thanks go out to my parents, and I will never forget their enormous contribution to my work.

Contents

List of Figures

Chapter 1

Introduction

In this chapter, we delve into the fundamental aspects of quantum computing required to comprehend this thesis. We explore quantum physics, which is crucial for understanding the challenges of quantum computing and is the driving force behind my interest in upcoming research topics. To achieve this, we provide a concise overview of the history of quantum computing and the fundamentals of quantum gates in Section 1.1. Subsequently, in Section 1.2, we introduce Noisy Intermediate-Scale Quantum (NISQ) computing. Lastly, in Section 1.3, we delve into the concept of quantum decoherence and its significance in quantum computing.

1.1 Quantum computing

In this section, we will provide an overview of the historical developments and the challenges associated with quantum computing. Subsequently, we will thoroughly examine the essential gates required to construct a universal quantum computer.

1.1.1 History

In the early 1980s, pioneers like Benioff [1] and Feynman [2] laid the groundwork for quantum computing, establishing its fundamental principles. Researchers took their first strides towards practical implementation in the subsequent decade, embarking on experimental endeavors during the 1990s [3, 4, 5, 6, 7].

Quantum computers give rise to a new class of algorithms that are not efficiently simulatable on classical computers due to the exponential scaling incurred when storing and manipulating state vectors. Two of the most noteworthy quantum algorithms proposed to date are Shor's algorithm [8] and Grover's search algorithm [9]. These algorithms hold great promise for the future of quantum computing. Grover's search algorithm, in particular, provides guaranteed quadratic speedup compared to the best-known classical counterparts. Although Shor's algorithm has yet to demonstrate a proven speedup, it holds the potential to accelerate computational speed significantly. For instance, Shor's algorithm can find the prime factors of an integer. The hardness of prime factorization is crucial to security encryption protocols based on RSA. Therefore, Shor's algorithm has the potential to break RSA-based keys exponentially faster than brute force methods. Besides these two well-known algorithms, the HHL algorithm, proposed in 2009 by Harrow et al. [10], has attracted attention in recent years. The HHL algorithm [10] holds significant potential regarding speeding up its classical counterparts. This algorithm can solve linear systems of equations. One challenge for quantum algorithms is that imperfect quantum hardware impacts their performance.

Before the development of quantum hardware, researchers acknowledged that quantum machines are more prone to errors compared to classical computers. As a result, it became essential to incorporate measurement techniques to verify the correct functioning of quantum devices. However, integrating these measurement techniques is challenging due to the requirement of commuting the measurement operator with the system device Hamiltonian, thereby avoiding any violation of Heisenberg's uncertainty principle. Furthermore, classical error correction codes pose a challenge due to the impossibility of replicating the precise state of a qubit for measurement. This limitation arises from the no-cloning theorem [11, 12], which establishes that there is no unitary transformation capable of duplicating an arbitrary quantum state. We can utilize classical codes; however, to ensure that our measurements account for both X and Z errors while maintaining commutation, it becomes necessary to employ two codes simultaneously. Shor [13], among others [14, 15, 16], suggested quantum error correction (QEC) codes to overcome this issue of imperfect quantum computing and impossible copying of the state. These codes require an increased number of physical qubits on the actual device. We can differentiate between logical and physical qubits. Logical qubits are the qubits an error-free algorithm requires. In contrast, physical qubits represent the number of qubits needed to execute a specific algorithm precisely under noisy conditions.

The era of fault-tolerant quantum computing defines a future age in which quantum computations can be performed without losing coherence to fatal errors induced by envi-

ronmental noise. In this era, the error rates of devices fall sufficiently below the threshold values of a quantum error-correcting code in use. Improvements at all levels are needed to pass this fault-tolerance threshold, from the materials used to substantiate qubits to the on-device layout of physical qubits. Small increases in the gate and readout fidelities could drastically impact the physical qubit requirements of a fault-tolerant quantum computer. The number of operations required depends exponentially on the error rate in a concatenated quantum error correcting code [17]. The two-qubit gate error is particularly interesting next to the readout fidelity, with the latter being today's dominant error in quantum hardware. It is worth noting that any quantum computing algorithm can be decomposed into single-qubit and two-qubit gates [18]. Contemporary hardware has attained fidelities for a two-qubit gate above 99 % [19] in superconducting platforms. Generally, current hardware fidelities are approaching the fault-tolerance thresholds of quantum error correcting codes [20]. Despite these achievements, encoding a single logical qubit may require thousands of physical qubits even with fidelities above 99% [21, 22, 23].

However, we need to analyze the error sources of two-qubit gates to improve the coherent and incoherent error regimes. In addition, not only the two-qubit gate error and the readout error are too high. There are still too few qubits available to make fault-tolerant quantum algorithms feasible. Making fault-tolerant quantum algorithms feasible remains a long-term objective, distant from the near future. An unanswered question is whether the quality and quantity of qubits can attain the limit of fault-tolerant quantum computers.

1.1.2 Quantum entanglement and quantum superposition

Quantum phenomena such as quantum entanglement and superposition are the key features of quantum mechanics and are necessary for the advantage. Consequently, we will introduce the two features that set quantum mechanics apart from classical physics. We start with the well-known Schrödinger equation

$$i\hbar\partial_t \left|\psi(t)\right\rangle = \hat{H}(t) \left|\psi(t)\right\rangle , \tag{1.1.1}$$

which is elementary for the comprehension of quantum mechanics, with $|\psi(t)\rangle \in \mathcal{H}$ being a state vector from some Hilbert space $\mathcal{H}$. $\hat{H}(t)$ is the system's Hamiltonian. This equation governs a pure and error-free coherent evolution of the system, meaning that the system is in a well-defined state at every point in time. For further investigation, let us consider a two-level system. We denote the two states of a two-level system with $|0\rangle$ and $|1\rangle$. Let α and β denote the probability amplitudes for the states, which are complex numbers, satisfying the condition $|\alpha|^2+|\beta|^2 = 1$. We will now examine the concept of superposition, demonstrated by the state

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,. \tag{1.1.2}$$

Unlike classical systems, which can only be in state $|0\rangle$ or $|1\rangle$, quantum systems can exist in a superposition of both states.

For quantum entanglement we couple the two-level system with Hilbert space $\mathcal{H}_1$ to another two-level system with Hilbert space $\mathcal{H}_2$. The joint system's state notation can be described as follows: If, for instance, the state $|0\rangle$ is occupied in both systems, we represent this configuration as $|00\rangle$ in the joint system. We exemplify quantum entanglement by

the state

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \in \mathcal{H}_1 \otimes \mathcal{H}_2. \tag{1.1.3}$$

Quantum entanglement refers to the phenomenon where a Hilbert space is composed of $\mathcal{H}_1 \otimes \mathcal{H}_2$, and the state cannot be factorized into $|\psi\rangle = |\psi\rangle_1 \otimes |\psi\rangle_2$, where $|\psi\rangle_1 \in \mathcal{H}_1$ and $|\psi\rangle_2 \in \mathcal{H}_2$. Indeed, the state represented by Equation (1.1.3) cannot be factorized, leading to the observation of quantum entanglement. These effects are of significant interest and illustrate the differences between quantum and classical mechanics.

1.1.3 Universal sets of quantum gates

In this section, we will introduce the basic gate set required for constructing a universal quantum computer. A quantum gate is represented by a unitary matrix. A universal quantum gate set is a collection of quantum logic gates capable of simulating any quantum algorithm and manipulating quantum states according to the desired objectives, thereby enabling the implementation of any quantum operation. As previously mentioned, we can reduce every quantum algorithm into two-qubit and single-qubit gates. Therefore, our universal gate set only requires single and two-qubit gates.
To begin with, we will acquaint ourselves with the Pauli operators X, Y, and Z

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \text{and} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{1.1.4}$$

which are indispensable for this purpose. The Pauli and identity matrices are a basis for all 2 x 2 complex matrices. Notably, the Pauli matrices possess two key properties: Hermiticity and unitarity. By multiplying each matrix with the imaginary unit, the Pauli matrices form a basis for the Lie algebra $\mathfrak{su}(2)$. This Lie algebra serves as the generator for the special unitary group $SU(2)$. The Pauli rotation gates

$$R_X = \exp(-iX\theta/2) = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}, \tag{1.1.5}$$

$$R_Y = \exp(-iY\theta/2) = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}, \tag{1.1.6}$$

$$R_Z = \exp(-iZ\theta/2) = \begin{pmatrix} \exp(-i\theta/2) & 0 \\ 0 & \exp(i\theta/2) \end{pmatrix}, \tag{1.1.7}$$

also belong to $SU(2)$. The three rotation gates R_X, R_Y, and R_Z are capable of reaching every pure state on the Bloch sphere [24, 17]. The Bloch sphere is the surface of a sphere on which all pure states of a two-level system are located. Specifically, R_X rotates the state vector around the x-axis, while R_Y and R_Z rotate it around the y-axis and z-axis, respectively. Since we have the capability to access all pure states in a two-level system, the Pauli rotation gates serve as the first component of our universal gate set.

The second essential component for establishing a universal gate set is a non-local two-qubit gate. Non-local two-qubit gates cannot be decomposed into two separate gates represented as $R_1 \otimes R_2$, where R_1 and R_2 belong to the $SU(2)$ group [25]. A two-qubit Pauli gate like $X_l \otimes X_j = X_l X_j$ with one X gate on qubit j and one on i is of local

structure for instance. One non-local gate is the CNOT gate

$$\mathrm{CNOT} = R_{Y_1}\left(\frac{\pi}{2}\right)\cdot R_{X_1X_2}\left(\frac{\pi}{4}\right)\cdot R_{X_1}\left(\frac{-\pi}{2}\right)\cdot R_{X_2}\left(\frac{-\pi}{2}\right)\cdot R_{Y_1}\left(\frac{-\pi}{2}\right), \tag{1.1.8}$$

which belongs to $SU(4)\setminus(SU(2)\otimes SU(2))$. The Pauli rotation gates and the CNOT gate form one option for a universal quantum gate set.

The two-qubit gates are of significant interest in quantum computing because they are more prone to errors than one-qubit gates. Moreover, quantum entanglement arises from the operations of non-local gates, as illustrated in the following example

$$\mathrm{CNOT}\cdot \mathrm{H}_1\left|00\right\rangle = \frac{1}{\sqrt{2}}\left|00\right\rangle + \frac{1}{\sqrt{2}}\left|11\right\rangle, \tag{1.1.9}$$

We apply the Hadamard gate H_1 to the first qubit to achieve superposition and then apply the CNOT gate to derive quantum entanglement. The two-qubit gate is necessary to achieve significant quantum effects, which may be responsible for quantum speed up in specific algorithms.

Native gates refer to the set of gates available on a given quantum hardware. In the case of trapped-ion-based quantum computers, the native two-qubit gate is

$$R_{X_lX_j}(\theta) = \exp\left(\frac{-iX_lX_j\theta}{2}\right), \tag{1.1.10}$$

Note that this gate is also a non-local gate. Of primary interest for quantum algorithms is the transformed local gate of $R_{X_lX_j}(\theta)$

$$R_{Z_lZ_j}(\theta) = \mathrm{H}_1\cdot\mathrm{H}_2\cdot R_{X_lX_j}(\theta)\cdot\mathrm{H}_1\cdot\mathrm{H}_2 = \exp\left(\frac{-iZ_lZ_j\theta}{2}\right). \tag{1.1.11}$$

Especially for chemical problems and optimization problems, which are formulated into Pauli-Z strings, this gate plays an essential role.

1.2 Noisy Intermediate-Scale Quantum (NISQ) Computing

In the previous section, we explored the potential realization of quantum computers. This section focuses on machines that still exhibit errors but do not incorporate Quantum Error Correction (QEC) called Noisy Intermediate-Scale Quantum (NISQ) computers. We first introduce the general concept in Subsection 1.2.1, outlining how such machines can be constructed. Building upon this understanding, we then delve deeper into the Quantum Approximate Optimization Algorithm (QAOA) in Subsection 1.2.2, which will be the focus of this thesis.

1.2.1 The NISQ concecpt

Before reaching the quantum threshold value to build fault-tolerant quantum computers, we must achieve essential milestones through bridge technologies for quantum computers.

These machines may have qubit fidelities lower than the quantum threshold theorem [26, 21, 22, 23] requires and qubit numbers below 100, but they can run specific algorithms efficiently. Because NISQ devices have a limited number of qubits, it is common to renounce quantum error correction codes and instead use all available physical qubits as logical qubits. This epoch is named Noisy Intermediate-Scale Quantum (NISQ) Computing. We can classify several NISQ hardware types, including superconducting gate-based quantum computers, quantum annealers, and digital-analog quantum computers.

In this era researchers aim to discover algorithms that can demonstrate quantum advantage or supremacy, even in the presence of noise. Quantum supremacy refers to the ability of a quantum device to solve a problem that is not feasible for a classical computer to solve within a reasonable amount of time, regardless of the problem's practical value. In distinction, we obtain quantum advantage if a quantum computer can solve a real-world problem faster than a classical computer. However, Google began the pursuit of quantum supremacy on their Sycamore chip in 2019 [19]. In order to demonstrate quantum supremacy, a random sampling algorithm was employed. Recently in 2021, Feng et al. [27] presented a classical algorithm that may outperform Google's sampling approach on a supercomputer. Despite this, Google's chip still notably improves cross-talk reduction. The Sycamore chip, consisting of 51 qubits, achieves an infidelity of 0.36 % for a single two-qubit gate. When measuring multiple two-qubit gates simultaneously, the infidelity rises to 0.62 %. It is worth noting that both measurement methods yield results of the same order of magnitude.

The algorithms designed for near-term devices are commonly referred to as NISQ algorithms. However, these algorithms are faced with several challenges, such as decoherence, coherent over-rotations, and cross-talk, which are common on NISQ devices. The most well-known algorithms for gate-based NISQ devices are the Variational Quantum Algorithms (VQAs). The main objective of every VQA is to find the optimal parameters for a parametrized circuit in order to minimize the energy of the objective function. Variational quantum algorithms exhibit strong resilience to noise, as demonstrated in several studies [28, 29, 30, 31]. Notably, Sharma et al. [32] showed that variational parameters are highly robust against measurement and Pauli channel noise. The Variational Quantum Eigensolver (VQE) and the Quantum Approximate Optimization Algorithm (QAOA) are the most popular ones. The VQE is particularly interesting since it combines a NISQ algorithm with chemistry optimization problems that are highly relevant to related enterprises. In quantum chemistry, calculating a molecule's ground state energy is one of the most significant challenges, and it forms the foundation for developing energy-efficient fertilizer production [33], among other applications [34]. Molecules are quantum mechanical in nature. As a result, the most efficient hardware for simulating them is a quantum computer, as Feynman once suggested [35]. Classical simulations can only handle a small number of atoms in complex molecules. Therefore, the VQE has emerged as a prominent candidate to demonstrate quantum speedup [36, 37, 38]. In addition, there is hope that this algorithm will exhibit supremacy as the first NISQ algorithm [39, 40].

The NISQ era has introduced new quantum hardware and algorithms that could demonstrate quantum advantage or at least quantum supremacy in the near future. The NISQ algorithm QAOA will be studied in detail throughout this thesis. It is worth noting that there is currently no proven quantum advantage of VQAs over their classical counterparts.

1.2.2 A Quantum Approximate Optimization Algorithm

In 2017, Farhi, Goldstone, and Gutmann [41] proposed the Quantum Approximate Optimization Algorithm (QAOA), a heuristic algorithm combining classical and quantum computing. This subsection will provide a comprehensive explanation of the algorithm's functionality. Its challenges and a brief overview of the current state of research will follow this subsection.

The QAOA is designed to tackle combinatorial optimization problems of the succeeding form

$$\hat{C}(\bar{z}) = \sum_{\alpha=1}^{m} C_\alpha(\bar{z}), \tag{1.2.1}$$

where m defines the number of clauses present in the problem, while $\bar{z} = z_1 z_2 \dots z_n$ denotes the n-bit string. Each bit is assigned a value of $z_i \in \{0, 1\}$. A clause is considered satisfied if $C_\alpha(\bar{z}) = 1$. Our objective is to identify the maximum number of clauses, denoted by $\max \hat{C}(\bar{z})$, and to determine the bit string $\bar{z}$ that satisfies the most clauses. The difficulty of solving these instances on a classical computer varies depending on the specific problem. To maintain an overview of the upcoming equations, we will occasionally use the shorthand notation $\hat{C}$ instead of $\hat{C}(\bar{z})$. The number of bits within the string $\bar{z}$ depends on the specific problem under consideration. $\hat{C}$ implies that the problem has already been translated into a quantum operator. The transformation $\{0, 1\} \to \{1, -1\}$ maps the single bits onto the eigenvalues of the Pauli-Z operator, thereby converting the problem to a quantum operator. As we have formulated our problem on a Pauli-Z basis, we use bracket notation $|\bar{z}\rangle = |z_0, ..., z_n\rangle$ instead of bit string notation.

To begin the QAOA approach, every bit string is initialized with the same probability amplitude. This is achieved by applying the Hadamard gate to each qubit in the system, resulting in the state

$$|s\rangle = |+\rangle^{\otimes n} = \mathrm{H}^{\otimes n} |0_0, \dots, 0_{n-1}\rangle . \tag{1.2.2}$$

Next, we introduce the driver Hamiltonian given by

$$\hat{B} = \sum_i X_i, \tag{1.2.3}$$

which is a sum over the Pauli-X operators acting on each qubit in the system. The QAOA method uses an exponential alternating approach between applying the driver Hamiltonian and the cost Hamiltonian $\hat{C}(\bar{z})$ of the problem at hand. This leads to the QAOA ansatz

$$|\vec{\gamma}, \vec{\beta}\rangle = e^{-i\beta_p \hat{B}} e^{-i\gamma_p \hat{C}} \dots e^{-i\beta_2 \hat{B}} e^{-i\gamma_2 \hat{C}} e^{-i\beta_1 \hat{B}} e^{-i\gamma_1 \hat{C}} |s\rangle . \tag{1.2.4}$$

The parameters associated with the driver Hamiltonian, denoted by $\vec{\beta} = (\beta_1, \beta_2, \dots, \beta_n)$, are unconstrained and can be chosen freely. Additionally, we consider the cost function $\hat{C}(\bar{z})$, which is also parameterized by unconstrained parameters denoted by $\vec{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_n)$. The rotation gates for the driver and cost function are defined as

$$\hat{U}(\hat{C}, \gamma_i) = e^{-i\gamma_i \hat{C}} \quad \text{and} \quad \hat{U}(\hat{B}, \beta_i) = e^{-i\beta_i \hat{B}}. \tag{1.2.5}$$

The number of parameters $\vec{\gamma}$, $\vec{\beta}$ used in the QAOA ansatz depends on different aspects and is the subject of the research presented here. After creating the ansatz, the expectation

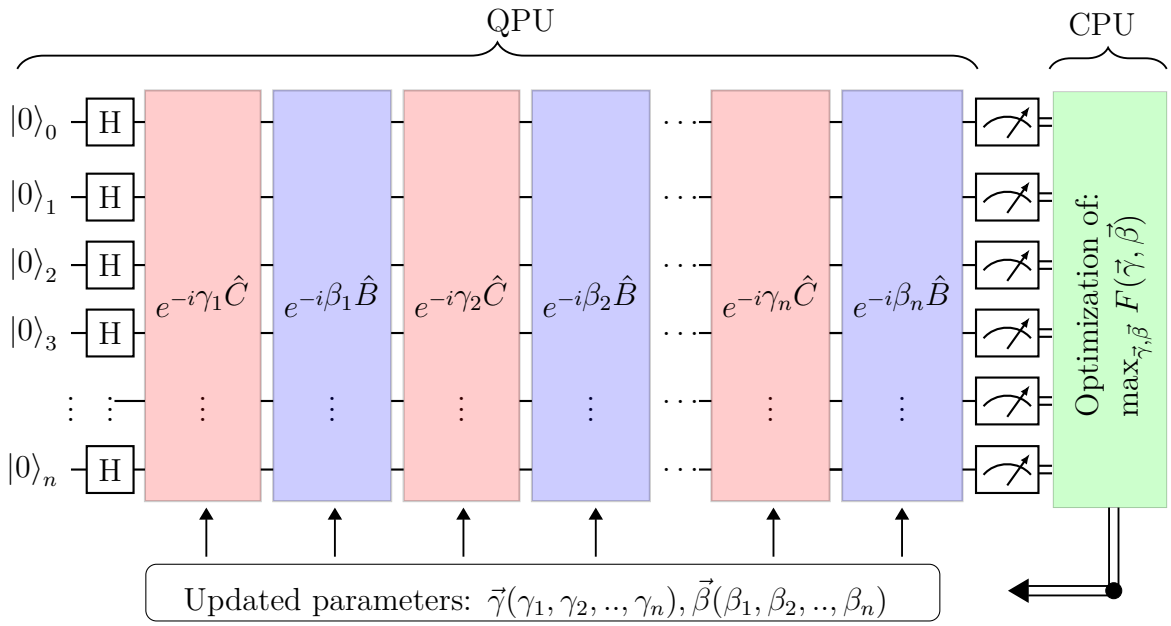


Figure 1.1: QAOA loop: Initially, we incorporate the QAOA ansatz on the Quantum Processing Unit (QPU) by implementing Hadamard gates, as well as the two-qubit gates $\hat{U}(\hat{C},\gamma_i)$ (red boxes) and $\hat{U}(\hat{B},\beta_i)$ (blue boxes). Subsequently, we compute the expectation value $\langle\hat{C}\rangle_{\vec{\gamma},\vec{\beta}}$ of the output state, and the optimizer (green box) then seeks the global minimum on a Central Processing Unit (CPU). The updated angles are reassigned to the QPU to create the QAOA ansatz.

value of the cost function will be computed

$$F_p = \langle\hat{C}\rangle_{\vec{\gamma},\vec{\beta}} = \langle\vec{\gamma},\vec{\beta}|\,\hat{C}\,|\vec{\gamma},\vec{\beta}\rangle\,. \tag{1.2.6}$$

$\langle\hat{C}\rangle_{\vec{\gamma},\vec{\beta}}$ serves as the loss function, and the aim is to determine the values of $\vec{\gamma}$ and $\vec{\beta}$ that maximize $\langle\hat{C}\rangle_{\vec{\gamma},\vec{\beta}}$, leading to

$$M_p = \max_{\vec{\gamma},\vec{\beta}} \langle\hat{C}\rangle_{\vec{\gamma},\vec{\beta}}\,. \tag{1.2.7}$$

We designate the optimal parameters with $\vec{\gamma^*}$ and $\vec{\beta^*}$.

Figure 1.1 illustrates the step-by-step process of the QAOA. First, the desired number of random parameters is used to create the QAOA ansatz on the quantum processing unit (QPU). Next, the output state's expectation value F_p is computed on a CPU. The loss function is then optimized on the CPU, and the updated variables $\vec{\gamma}$ and $\vec{\beta}$ are sent back to the QPU. The QPU then generates a new QAOA ansatz with the updated parameters. This iterative process continues until a point of convergence is achieved. The included parameters in the QAOA approach are called p-depth, p-value, or p. The number of parameters is then $2p$. We suggest obeying the following process to determine the optimal value for p: begin with $p = 1$ and execute the loop procedure illustrated in Figure 1.1. Once convergence is achieved for $p = 1$, increment the p-depth and repeat the process. To improve maximization, we can increase the value from p to $p + 1$, provided that

$$M_p \leqslant M_{p+1} \tag{1.2.8}$$

holds. We increase p until no further improvement in M_p is achieved. It is worth mentioning that for every p, the parameters have to be randomized for the QAOA ansatz. There is no guarantee that the optimal parameters for a given p will be the same as those for $p+1$. The QAOA ansatz is an approximate version of the adiabatic quantum computing approach. Therefore, $p \to \infty$ will enable us to achieve the maximum value

$$\lim_{p\to\infty} M_p = \max_z \hat{C}(\bar{z}). \tag{1.2.9}$$

The problem operator $\hat{C}$ is alternatively referred to as the Hamiltonian operator. As a result, the notation $\hat{H}$ is predominantly employed to represent this operator related to the problem.

In this thesis, we follow the convention that maximizing $\hat{C}(\bar{z})$ entails minimizing $-\hat{C}(\bar{z})$ during the optimization process, thereby allowing us to determine the maximal energy of the optimization problem. This implies that when referring to a global minimum in the objective function or the loss function, we are specifically addressing the global minimum within the optimization process, which corresponds to the global maximum of the respective optimization problem under consideration.

1.2.3 Measuring QAOA performance

To quantify the performance of the QAOA, we introduce two parameters: the mean approximation ratio $\mathcal{M}_p$ and the improvement over random guessing $\mathcal{I}_p$. The subscript p refers to the p-depth. The mean approximation we define by

$$\mathcal{M}_p = \frac{M_p - E_{\min}}{E_{\max} - E_{\min}}. \tag{1.2.10}$$

The improvement over random guessing is defined by

$$\mathcal{I}_p = \frac{M_p - \langle s| \hat{H} |s\rangle}{E_{\max} - \langle s| \hat{H} |s\rangle}. \tag{1.2.11}$$

We first investigate $\mathcal{M}_p$. $E_{\min}$ and $E_{\max}$ are the minimum and maximum energy values of the investigated Hamiltonian $\hat{H}$. The mean approximation ratio compares the result of M_p to the scale of the minimal and maximal energy of the problem's distribution. This is a valuable performance indicator for different instances of the same problem. If M_p supplies the ground state energy then $\mathcal{M}_p = 1$. In the worst-case scenario of the QAOA performance, we achieve $\mathcal{M}_p = 0$.

Now let us investigate the case for different problems. This entails exploring the average of the classical probability density function denoted as $N(E)$. $N(E)$ is the cost function $\hat{H}$ sorted by counts per energy. If we set the QAOA angles $\gamma_i = \beta_i = 0$, the QAOA approach produces the average $\overline{N(E)} = \langle s| \hat{H} |s\rangle$. The QAOA approach provides $\overline{N(E)}$ without using the optimizer. Furthermore, if the optimizer stops in a region of vanishing gradient in the parameter space, we also receive $M_p = \langle s| \hat{H} |s\rangle$. These regions cover most of the parameter space. For this reason, we refer the average energy $\langle s| \hat{H} |s\rangle$ of the cost function to the point where the QAOA result is equal to purely random guessing of the QAOA angles $\vec{\gamma}$ and $\vec{\beta}$. Thus, QAOA shows no improvement over the classical average $\overline{N(E)}$. Figure 1.2 shows two problems that refer to two different energy distributions. The left plot refers to a Gaussian distribution, and the right plot refers

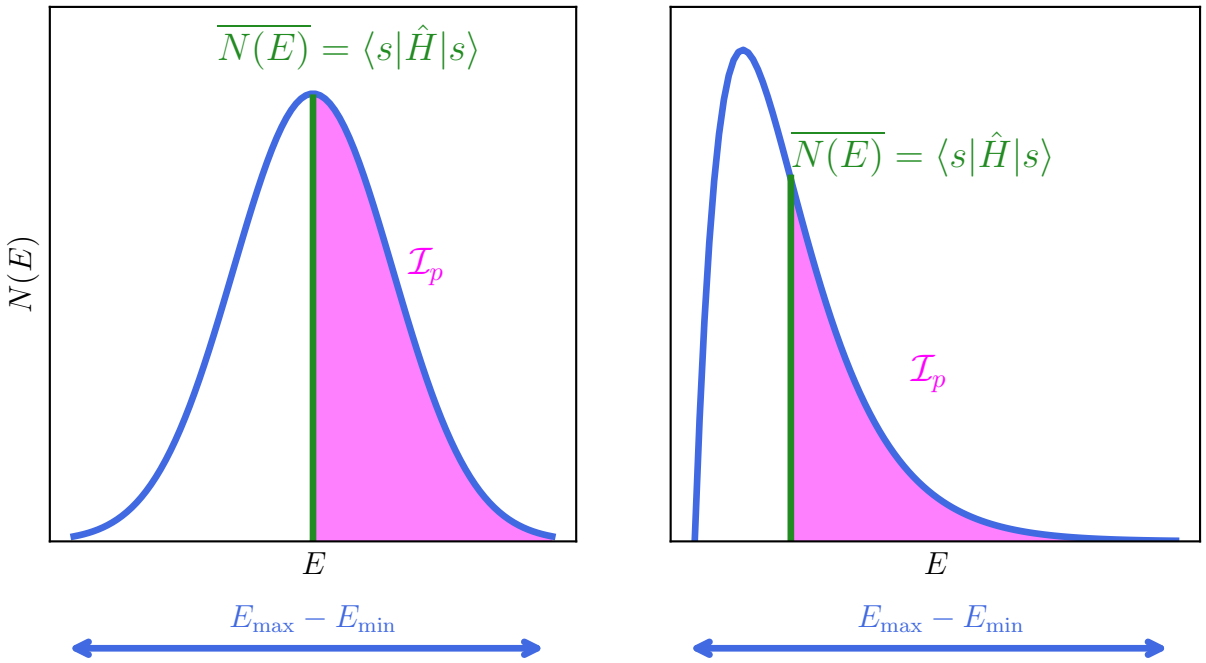


Figure 1.2: Plotting probability density functions $N(E)$ for a problem Hamiltonian $\hat{H}$. E represents the energy associated with a solution string for $\hat{H}$, while N represents the count of solution strings that share the same energy. On the left plot, we have a Gaussian distribution, and on the right plot, a gamma distribution. The green color indicates the expectation value $\overline{N(E)}$, while the blue arrow represents the normalization scale for $\mathcal{M}_p$. Additionally, the magenta-colored curve signifies the normalization scale used for $\mathcal{I}_p$.

to a gamma distribution. We investigate the case $\langle s|\,\hat{H}\,|s\rangle$ for both probability density functions (pdfs), which reflects having no improvement utilizing QAOA. For the Gaussian pdf, we achieve $\mathcal{M}_p = 0.5$, whereas for the gamma pdf, we obtain $\mathcal{M}_p = 0.2$. Indeed, there is a distinction in $\mathcal{M}_p$, but QAOA's performance remains consistent in both cases. For this reason, $\mathcal{M}_p$ is inappropriate for comparing different problems and hence different pdfs.
To assess different problems, we utilize the QAOA's improvement over random guessing, denoted as $\mathcal{I}_p$ (refer to Equation (1.2.11)). For $\mathcal{I}_p$ we normalize M_p with $\overline{N(E)}$ as the lower bound. Hence, if we set the angles $\gamma_i = \beta_i = 0$ for both pdfs we achieve $\mathcal{I}_p = 0$. We conclude that $\mathcal{I}_p$ is for comparing QAOA performance among different problems and $\mathcal{M}_p$ for comparing among different instances of one problem.

1.2.4 Combinatorial optimization

The optimization of the QAOA parameters is one significant challenge to overcome. Figure 1.3 illustrates these challenges by exemplifying the loss function of the QAOA ansatz for F_1 with $p = 1$.

To begin with, F_1 possesses a global minimum and maximum. The three colored traces, magenta, red, and green, represent three problems the optimizer must overcome. We start by examining the magenta curve. Its initial point is close to the local minimum.

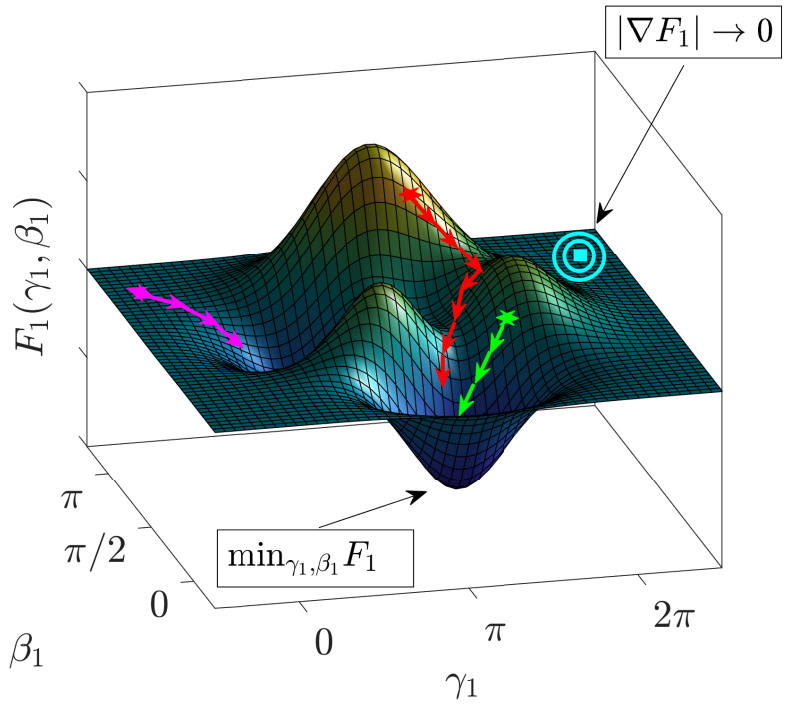


Figure 1.3: Plotting the QAOA solution space $F_1(\gamma_1, \beta_1)$ for $p = 1$ with $\hat{C} = Z_0 Z_1$. $F_1(\gamma_1, \beta_1)$ is represented on the z-axis, while γ_1 and β_1 are on the X-Y plane. The magenta, green, and red traces depict possible paths a local optimizer can take from their respective initial points. The cyan circle indicates a region of vanishing gradient.

When a local optimization strategy is employed in this case, the optimizer tends to become stuck. On the other hand, the green trace is initialized close to the global minimum, and the optimizer can locate the global minimum by following the gradient ∇F_1. The red path potentially gets stuck in a local minimum, requiring several optimization calls and, therefore, many QPU calls. The green trace represents a more realistic situation for the optimization process. The initial parameters are not the optimal parameters $\vec{\gamma}^*, \vec{\beta}^*$, but we placed the optimizer in the concave region where the global minimum is located. As a result, the optimizer requires fewer iterations than the other traces and can locate the global minimum. The final scenario is represented by the cyan-colored circles, where the optimizer could not locate the global minimum or any local minima despite the absence of errors on the quantum device. This situation arises when the gradient of the loss function vanishes $|\nabla F_1| \to 0$. It is crucial to distinguish this scenario from the vanishing gradient phenomenon known as the barren plateau, which occurs due to over-parameterization. Barren plateaus typically manifest when employing a random ansatz, like the VQE approach [42, 43, 44]. It is important to note that all the possible impediments in discovering the ground state energy with the QAOA ansatz are based on the assumption of error-free hardware, which is unrealistic. If the incoherent error is increased even slightly, the red trace becomes an impractical scenario for the optimizer.

To overcome these challenges, it is necessary to apply knowledge of the problem or a specific optimization technique to QAOA. One way to achieve this is by introducing layerwise learning to optimize the parameter space of $2p$. Alternatively, one can reuse the parameters from the previous step [45, 46]. However, the critical issue is identifying the most effective layerwise technique to avoid encountering a vanishing gradient

in the parameter space. One possible solution is to group multiple parameters into a single space. Irrespective of the specific sequential approach used, M_p saturates with $\lim_{p\to\infty} M_p \neq \max_{\vec{z}} \hat{C}(\vec{z})$. Campos et al. [47] demonstrated that sequential optimization could cause a sharp transition in the success probability of locating the optimal solution. They also showed that introducing phase noise can overcome the saturation of the variational circuit. This result is based on the outcomes of training a quantum autoencoder [48]. The sequential method reduces the parameter space, but a higher circuit depth is typically required, regardless of whether saturation is achieved or not.

Rather than decreasing the parameter space, an alternative method is to restrict the state space. Instead of providing the optimizer with all possible states after the measurement process, the optimizer utilizes only states with the lowest energy eigenvalues. The expectation value represents the lower tail of the probability distribution of measurement outcomes rather than the distribution's average. This approach simplifies the solution space, which reduces the risk of being trapped in local minima far from the global minimum. The IBM quantum group has implemented this idea [49] into VQE and QAOA, with supporting evidence that this method reduces the number of parametrized circuit calls. This idea has been previously introduced in other scientific fields and is commonly used in finance as conditional value at risk (CVaR) [49]. Furthermore, researchers [50] have proposed a new loss function called Ascending-CVaR that is inspired by quantum annealing. Generally, various techniques from other fields, such as machine learning [51], have been attempted to identify the optimal parameters, but the task remains challenging.

1.2.5 Limitations of QAOA

In this subsection, we will provide a summary of what is currently understood about the application of QAOA to specific problems. Instead of centering our attention on the optimization aspect, within this subsection, our primary emphasis will be on choosing a suitable value for p in order to attain a predefined threshold for the mean approximation ratio and gaining insights into the optimal parameters that have been determined for specific problems. Despite the assurance of attaining the global minimum, determining the optimal angles for a particular problem and selecting an appropriate p value is challenging. Due to the limited coherence time, there is significant interest in decreasing the circuit depth of the QAOA, which can be achieved by using a small p.

Farhi et al. [41] discovered the limitations of the performance of QAOA applied to the ring of disagrees

$$\hat{H}_{\text{ring}} = Z_1 Z_n + \sum_{i=1}^{n-1} Z_i Z_{i+1}. \tag{1.2.12}$$

They determined that the solution depends on the system size with $M_p = n(2p+1)/(2p+2)$. It is also possible to find an analytic solution for small problems, such as two qubit-sized instances with $p = 1$, as demonstrated by Wang et al. [52]. For $n = 2$ and $p = 1$, they calculated $\langle \hat{C} \rangle_{p=1} = 2\sin(4\beta_1)\cos(4\gamma_1)$ with optimal parameters $(\gamma_1^*, \beta_1^*) = \pi(3/8, 1/8)$. Farhi et al. [53, 54] also demonstrated that for a fully connected Ising chain of the $Z_i Z_j$ type, the cost function $\hat{C} = (1/\sqrt{n^3}))\sum_{i,j} m_{i,j} Z_i Z_j$ achieves convergence to optimal global parameters $\vec{\gamma}^*, \vec{\beta}^*$ for $n \to \infty$, regardless of the specific instance. In addition, they found optimal analytical parameters for $p \leqslant 8$ and extrapolated values for $9 \leqslant p \leqslant 12$. The extrapolation trend builds on the analytical results of $p = 8$. Furthermore, they demonstrated that the optimal values $\vec{\gamma}^*, \vec{\beta}^*$ for $n \to \infty$ can also be utilized for $n \geqslant 26$, providing

at least an improvement over random guessing for the mean approximation ratio.

The reachability of the global optimum in dependency of clause density was investigated by Akshay et al. [55]. Reachability deficits were proven for the random boolean satisfiability problems 2SAT and 3SAT. Additionally, Hastings [56] explored the limitations of QAOA. In the research conducted by Biamonte's group [57], they presented analytic evidence of parameter concentration, where the optimal parameters concentrate inversely polynomially based on the problem size. They also provided numerical evidence across different problems. This result is significant compared to the findings of Fahri et al. [53, 54], who presented optimal parameters independent of the instance. Streif et al. [58] also provided numerical evidence of parameter concentration. Jahan Claes et al. [59] studied multi-body coupling problems described by Equation (1.2.1) and found evidence of comparable behavior between two-body and multi-body couplings for $p = 1$. They also explored the optimal parameters in the limit $n \to \infty$ and discovered that they are independent of the specific problem instance. Another approach to mitigate the challenges associated with the $2p$ large parameter space is to use symmetries in the loss function to reduce the $2p$ dimensional space [60].

All these limiting factors raise the question of whether QAOA could even be a candidate to show quantum advantage or at least quantum supremacy [61, 62, 63, 64, 65]. According to Farhi et al. [66], the QAOA may demonstrate quantum supremacy even in a shallow circuit depth.

1.2.6 QAOA applied to real-world problems

So far, we have covered how the QAOA can be applied to generic combinatorial optimization problems. However, we have yet to investigate how QAOA could be applied to real-world problems.

To employ QAOA in solving a real-world problem, the initial step involves converting the problem into a quantum operator. This entails the translation of the problem into Pauli Z-strings. This step is specific to the problem. One example of a real-world problem is the Traveling Salesperson Problem (TSP). TSP is a combinatorial optimization problem. In this problem, a salesperson must find the shortest possible route to visit a set of cities once. The objective is to minimize the total distance traveled, making TSP a fundamental problem. However, applying an algorithm for solving TSP on a quantum computer can be challenging due to the high circuit depth. The constraints in the problem formulation, such as specific weekdays for arriving at specific cities or travel costs, can increase the circuit depth significantly, potentially exceeding the coherence time of the system. One can add penalty terms with a Lagrange parameter to incorporate constraints into the cost function. Hadfield et al. [67] studied TSP for QAOA. Hadfield et al. demonstrated the efficacy of using the two-qubit gate

$$X_iY_j(\theta) = \exp\left(\frac{-i\theta}{2}\left(X_iX_j + Y_iY_j\right)\right), \tag{1.2.13}$$

as a driver in an error-free scenario to incorporate certain constraints [68, 69] of a specific problem. Instead of using the X-driver, the XY-driver helps to stay in the feasible subspace.

Enterprises are exploring potential problems that are well-suited for NISQ devices. One such problem, the Flight Gate Assignment (FGA) problem, has been addressed by Stollenwerk et al. [70]. This particular problem is of great interest to researchers, as

it is known to be NP-hard. The problem arises when connecting flights, which involve passengers from different flights, are required to wait until all passengers have arrived. This can create a congestion point at the gate, obstructing the passage of other aircraft as they wait to access the connecting gate. By implementing more efficient scheduling of flights across all gates, it would be possible to decrease the waiting times for incoming flights. This, in turn, could lead to substantial time and cost savings for both passengers and airlines.

1.2.7 Adiabatic quantum computing

Adiabatic quantum computing, as a NISQ device, establishes a connection with the QAOA. Thus, we provide a concise overview of the adiabatic approximation, founded on the principles of the adiabatic theorem, which forms the foundation for adiabatic quantum computers. This theorem states that when a system undergoes slow changes over time and starts in its ground state, it will remain in the ground state at the final time. Here, we will briefly review the adiabatic approximation by Sarandy et al. [71], presenting the specific conditions that ensure a system remains in its ground state during its time evolution.

To begin, it is important to note that we can diagonalize any Hamiltonian $\hat{H}(t)$ at any time with

$$\hat{U}(t)^{\dagger}\hat{H}(t)\hat{U}(t), \tag{1.2.14}$$

for some suitable unitaries $\hat{U}(t)$. So, we can find at any time instantaneous eigenstates $|\lambda_n\rangle$ with eigenenergies E_λ. In fact, they fulfill the Schrödinger equation and also build an orthonormal basis for the desired time t

$$\langle\lambda_n|\lambda_m\rangle = \delta_{nm}. \tag{1.2.15}$$

For the time evolution of a specific state $|\psi(t)\rangle$, we can express the state as a sum of eigenstates based on their corresponding eigenenergies in the time-dependent Schrödinger equation

$$|\psi(t)\rangle = \sum_{l=1}^{D} a_l(t)\exp\Big(-i\int_0^t E_l(t^{'})dt^{'}\Big)|\lambda_l(t)\rangle\,. \tag{1.2.16}$$

We define

$$\theta_l = -\int_0^t E_l(t^{'})dt^{'} \tag{1.2.17}$$

as the dynamical phase. a_l is a complex function in time and D is the dimension of the Hilbert space. The goal is now to determine a_l. To achieve this, we substitute Equation (1.2.16) into the time-dependent Schrödinger equation, resulting in the following

$$\sum_{l=1}^{D} a_l(\hat{H}(t)\,|\lambda_l\rangle)e^{\theta_l} = \sum_{l=1}^{D}((\partial_t a_l)\,|\lambda_l\rangle + ia_l(\partial_t\theta_l)\,|\lambda_l\rangle + a_l\partial_t\,|\lambda_l\rangle))e^{i\theta_l}. \tag{1.2.18}$$

The left-hand side and the second term on the right-hand side build the Schrödinger equation for instantaneous time and cancel out. Equation (1.2.18) reduces to

$$0 = \sum_{l=1}^{D}((\partial_t a_l)\,|\lambda_l\rangle + a_l\,|\lambda_l\rangle))e^{\theta_l}. \tag{1.2.19}$$

In the next step, we take the scalar product with one eigenstate $|\lambda_m\rangle$ to get an equation of motion for the coefficients a_l

$$\begin{aligned}\partial_t a_m &= -\sum_{l=1}^{D} a_l \langle\lambda_m|\partial_t\lambda_l\rangle\, e^{i(\theta_l-\theta_m)} \\ &= -a_m \langle\lambda_m|\partial_t\lambda_m\rangle - \sum_{l=1,l\neq m}^{D} a_l \langle\lambda_m|\partial_t\lambda_l\rangle\, e^{i(\theta_l-\theta_m)}.\end{aligned} \tag{1.2.20}$$

The term $\langle\lambda_m|\partial_t\lambda_l\rangle$ in the equation can be rewritten in terms of the Hamiltonian with time-independent Schrödinger equation for the instantaneous case at time t for state $|\lambda_l\rangle$. Subsequently, we perform the time derivative of the Schrödinger equation and the scalar product with the state $\langle\lambda_m|$. The result can be plugged into Equation (1.2.20) leading to

$$\partial_t a_m = -a_m \langle\lambda_m|\partial_t\lambda_m\rangle - \sum_{l=1,l\neq m}^{D} a_l \frac{\langle\lambda_m|\partial_t\hat{H}|\lambda_l\rangle}{|E_m-E_l|} e^{-i(\theta_l-\theta_m)}. \tag{1.2.21}$$

By analyzing this equation, we can determine the condition for an adiabatic process. Our objective is to separate the equation of motion for the coefficients a_l from all other energy levels. In order to attain this goal, it is essential to reduce the magnitude of the second term in the previous equation, which consequently results in the following condition

$$\max_{0\leq t\leq T}\left|\frac{\langle\lambda_m|\partial_t\hat{H}|\lambda_l\rangle}{E_{ml}}\right| \ll \min_{0\leq t\leq T}|E_{ml}|, \tag{1.2.22}$$

with T being the total evolution time and

$$E_{ml} = |E_m - E_l| \tag{1.2.23}$$

being the energy gap between two states. Equation (1.2.22) states that to maintain the relative positions of the l^{th} and m^{th} eigenstates in time and prevent them from crossing, the rate of change in time of the Hamiltonian should be much smaller than the energy gap E_{ml}. We express the requirement that the system Hamiltonian changes slowly in time as $\partial_t\hat{H} \approx 0$.

To observe the impact of this condition, let us revisit Equation (1.2.16) to determine the $a(l)'s$. Our current approach involves the omission of the second term in the final equation of (1.2.20), driven by the constraint specified in Equation (1.2.22). This simplification results in

$$\partial_t a_m = -a_m \langle\lambda_m|\partial_t\lambda_m\rangle\,. \tag{1.2.24}$$

In fact, the solution for the complex function in time is

$$a_m(t) = a_m(0)e^{i\gamma_m}, \tag{1.2.25}$$

with

$$\gamma_m = i\int_0^t \langle\lambda_m(t')|\partial_t\lambda_m(t')\rangle\, dt'. \tag{1.2.26}$$

In this case, we can deduce from the inner scalar product that the phase γ is a real value. If we start in a specific state m, at the initial time, with $a_m(0) = 1$ and $a_l(0) = 0$ for all l not equal to m. Under these conditions, all terms in Equation (1.2.16) except one cancel

out yielding

$$|\psi_m(t)\rangle = e^{i\theta_m} e^{i\gamma_m} |\lambda_m(t)\rangle \,. \tag{1.2.27}$$

Thus, we remain for the whole time in the m^{th} state, picking up some additional phases. It is worth noting that the instantaneous eigenstates of $\hat{H}(0)$ and $\hat{H}(t)$ for the m^{th} state are not the same. However, we can assert that the system remains in the same state as long as we satisfy the condition outlined in Equation (1.2.22). This condition also leads to a paradox. It is possible that when we apply the approximation for the same energy eigenstate to two systems, they both satisfy the Hamiltonian yet end up being distinct from each other. Therefore, we must introduce the parameter $s = t/T$, resulting in

$$\max_{s\in[0,1]} \left| \frac{\langle \lambda_m(s)|\partial_s \hat{H}|\lambda_l(s)\rangle}{E_{ml}(s)} \right| << \min_{0\le t\le T} |E_{ml}(s)|. \tag{1.2.28}$$

1.3 Decoherence in quantum theory

Within this section, we aim to emphasize the significance of quantum decoherence in the realm of quantum computing. Additionally, we present an introduction to the fundamental concepts surrounding the simulation of quantum decoherence.

1.3.1 General concept

In this subsection, we focus on exploring the significance of decoherence for quantum computing. Decoherence can be interpreted as a simple scattering process that lacks memory effects in the system-environment interaction, resulting in the irreversible loss of certain information about the system dynamics. The environment often has a large number of degrees of freedom, making it highly complex and difficult to control. Decoherence forms a significant barrier to realizing quantum computing. The study of quantum decoherence is relatively new, as researchers previously concentrated on merging classical and quantum mechanics rather than linking the environment to the quantum system. It was in the early 1970s that Zeh [72, 73] drew attention to quantum decoherence in quantum research. This was followed in the early 1980s by Zurek [74, 75], who established the concept of einselection, which refers to environment-induced superselection. In 1985 and 1986, both Zeh [76] and Zurek [77] proposed the first numerical and theoretical concepts of quantum decoherence.

To grasp the concept of quantum decoherence, we will explore a system interacting with an environment and carry out a measurement of an observable on the system. Consider an arbitrary system S that can exist in two possible states, $|\uparrow\rangle$ and $|\downarrow\rangle$. We allow S to interact with an environment E, which also has two possible states, $|e_1\rangle$ and $|e_2\rangle$. Furthermore, we demand that system-environment is coupled in the following way

$$|\uparrow\rangle |e_1\rangle \,, \quad |\downarrow\rangle |e_2\rangle \,. \tag{1.3.1}$$

Accordingly, the environmental state depends on the system. This leads to system-environment entanglement,

$$|\psi\rangle = \frac{1}{\sqrt{2}} |\uparrow\rangle |e_1\rangle + \frac{1}{\sqrt{2}} |\downarrow\rangle |e_2\rangle \,. \tag{1.3.2}$$

Our next step is to measure the system using an observable $\hat{O}_S$ while leaving the environment unaffected. We express the state vector $|\psi\rangle$ in terms of a density matrix, denoted as ρ, given by $\rho = |\psi\rangle\langle\psi|$. This density matrix, ρ, is in general represented as a sum of classical probabilities a_i and corresponding pure states $|\psi\rangle_i$, with the form $\rho = \sum_i a_i |\psi\rangle_i \langle\psi|_i$. However, we only require a single pure state for our density matrix. To obtain the equation for the system, we need to trace out the environment

$$\begin{aligned}\rho_S &= \mathrm{Tr}_E(\rho) \\ &= |\alpha|^2 |\uparrow\rangle\langle\uparrow| + |\beta|^2 |\downarrow\rangle\langle\downarrow| \\ &+ \alpha\beta^* |\uparrow\rangle\langle\downarrow| \langle e_2|e_1\rangle + \alpha^*\beta |\downarrow\rangle\langle\uparrow| \langle e_1|e_2\rangle .\end{aligned} \tag{1.3.3}$$

Subsequently, we determine the expectation value of $\hat{O}_S$ using the density matrix ρ_S

$$\begin{aligned}\langle\hat{O}_S\rangle &= \mathrm{Tr}_S(\rho\hat{O}_S) \\ &= |\alpha|^2 \langle\uparrow|\hat{O}_S|\uparrow\rangle + |\beta|^2 \langle\downarrow|\hat{O}_S|\downarrow\rangle \\ &+ \alpha\beta^* \langle\uparrow|\hat{O}_S|\downarrow\rangle \langle e_2|e_1\rangle + \alpha^*\beta \langle\downarrow|\hat{O}_S|\uparrow\rangle \langle e_1|e_2\rangle .\end{aligned} \tag{1.3.4}$$

The expectation value includes two types of terms, those with classical probabilities and those with interference between α and β. These interference terms are fragile and responsible for observing coherence effects between the two systems. They describe the non-classical behavior and indicate the transition between classical and quantum nature. The quantum coherent effects cannot be observed when these terms decay to zero. Incoherent processes mostly lead to an exponential decay of the off-diagonal elements [78, 79, 80]

$$\langle e_2|e_1\rangle \propto e^{-t/\tau_d}, \tag{1.3.5}$$

with τ_d being the coherence time.

To overcome decoherence, researchers aim to isolate the qubit system from the environment while maintaining control over it. As mentioned previously, Shor [13], along with other researchers [15, 81], brought forth the concept of quantum error correction to prevent decoherence or rectify errors within the system.

1.3.2 Lindblad equation

Due to environmental complexity, it becomes necessary to develop an equation that describes only the system dynamics in time. In order to describe the effects of decoherence in a time-dependent equation, it is essential to derive the Lindblad equation [79, 82].

The starting point for this derivation is the von Neumann equation

$$\partial_t\rho(t) = [\hat{H}(t), \rho(t)]. \tag{1.3.6}$$

This equation defines the evolution of the system and environment over time. This differential equation is not efficiently simulable because of the large degrees of freedom of the environment. As a result, approximations must be made.

To derive a system equation under the influence of the environment, we introduce the concept of master equations. In Equation (1.3.6), $\rho(t)$ represents the qubit system and environment, with the Hamiltonian consisting of three components: $\hat{H}_S(t)$ for pure system dynamics, $\hat{H}_E(t)$ for the environment, and $\hat{H}_{\mathrm{int}}(t)$ for the interaction between the

two. To simplify the von Neumann equation, we assume that the system and environment are not correlated at $t = 0$, meaning there is no entanglement. The density matrix is separable with $\rho(t = 0) = \rho_{\mathrm{S}}(t = 0) \otimes \rho_{\mathrm{E}}(t = 0)$. Subsequently, we assume a weak coupling between the environment and the system called the Born approximation. If the environment is much larger than the system, the influence of the system on it can be considered negligible, and the system and environment can be separated according to

$$\rho(t) \approx \rho_{\mathrm{S}}(t) \otimes \rho_{\mathrm{E}}. \tag{1.3.7}$$

The following essential approximation is the so-called Markov approximation. We assume

$$\tau_{\mathrm{S}} \gg \tau_{\mathrm{E}}, \tag{1.3.8}$$

implying the environment's correlation time τ_{E} is much shorter than the system's τ_{S}. τ is the characteristic time of correlation decay. Consequently, we do not take memory effects of the environment into account. Considering that we use all the approximations mentioned and change to the interaction picture, we reach the Lindblad equation

$$\partial_t \rho_{\mathrm{S}} = [\hat{H}(t), \rho_{\mathrm{S}}] + \sum_n \left(\hat{L}_n \rho_{\mathrm{S}}(t) \hat{L}_n^\dagger - \frac{1}{2} L_n^\dagger \hat{L}_n \rho_{\mathrm{S}}(t) - \frac{1}{2} \rho_{\mathrm{S}}(t) \hat{L}_n^\dagger \hat{L}_n \right). \tag{1.3.9}$$

In order to obtain the above equation, we utilize the rotating wave approximation. This means that we disregard fast-oscillating terms, as they average to zero. The resulting Lindblad-type master equation aids our comprehension of quantum computers' decoherence. On the right-hand side of the equation, the first term represents the coherent evolution of the system, while the second term represents incoherent evolution. The $\hat{L}_n$ operators comprise a set of non-unitary operations, which we will investigate in greater detail. These operators define various physical incoherent processes. The field of research that encompasses the study of the Lindblad equation is known as open quantum system dynamics.

1.3.3 Quantum channels

In the prior subsection, we learned how to model quantum decoherence in a time-dependent equation. This subsection analyzes the possible incoherent noise sources and how they are inserted into quantum circuits. These noise sources are non-unitary to the system. Our attention will be to show that quantum operations lead to the so-called operator sum representation [17, 83].

First, we study pure states in the coherent Schrödinger picture. The time evolution operator $\exp(-i\hat{H}t)$, not only preserves probability amplitudes but also exhibits linearity. It transforms the input state vector into another state vector, making it a quantum operation. Non-unitary operations lead to a mixed-state density matrix. Consequently, the question is, how do we derive the quantum operations $\hat{M}$ that recast density matrices into density matrices:

$$\rho \to \rho' = \hat{M} \rho \hat{M}^\dagger. \tag{1.3.10}$$

The use of the quantum operation for introducing errors to the system is mostly referred to as quantum channels. Quantum channels are denoted with $\mathcal{E}$—consequently, Equation (1.3.10) can also be written as $\mathcal{E}(\rho) = \rho'$. The definition of density matrices gives rise to the properties associated with quantum channels, which are:

- completely positive-preserving: $\rho \geqslant 0 \Rightarrow \rho' \geqslant 0$
- Trace-preserving: $\mathrm{Tr}(\rho) = \mathrm{Tr}(\rho')$
- Linearity: $\mathcal{E}(\alpha\rho_i + \beta\rho_j) = \alpha\mathcal{E}(\rho_i) + \beta\mathcal{E}(\rho_j)$

To show the operator sum representation, we introduce a unitary quantum operation $\hat{U}_{\mathrm{SE}}$ acting on a system and environment. We aim to derive an equation for ρ_{S}. The concept involves having a system ρ_{S} and an environment ρ_{E}. Additionally, we require short interaction and a large bath. This leads to the separability of the system and environment $\rho = \rho_{\mathrm{S}} \otimes \rho_{\mathrm{E}}$ before the operation $\hat{U}_{\mathrm{SE}}$ acts on ρ. We first purify the system with the environment to model the interaction and then discard the environment. Let the environment $\rho_{\mathrm{E}} = |e_0\rangle \langle e_0|$ be in a pure state $|e_0\rangle$. Assuming that the environment starts in a pure state does not restrict generality because if it begins in a mixed state, we have the freedom to introduce an additional system that purifies the environment. To discard the environment, we trace out the environmental degrees

$$\rho'_{\mathrm{S}} = \mathrm{Tr}_{\mathrm{E}}\Big(\hat{U}_{\mathrm{SE}}(\rho_{\mathrm{S}} \otimes \rho_{\mathrm{E}})\hat{U}^{\dagger}_{\mathrm{SE}}\Big), \tag{1.3.11}$$

leading to

$$\rho'_{\mathrm{S}} = \sum_i (\mathbb{1}_{\mathrm{S}} \otimes \langle\lambda_i|)\hat{U}_{\mathrm{SE}}(\rho_{\mathrm{S}} \otimes |e_0\rangle \langle e_0|)\hat{U}^{\dagger}_{\mathrm{SE}}(\mathbb{1}_{\mathrm{S}} \otimes |\lambda_i\rangle), \tag{1.3.12}$$

with λ_i being an orthonormal basis of the environment. Next, we insert the identity $\mathbb{1}_{\mathrm{S}}$ for the system to define

$$\rho_{\mathrm{S}} \otimes |e_0\rangle \langle e_0| = (\mathbb{1}_{\mathrm{S}} \otimes |e_0\rangle)\rho_{\mathrm{S}}(\mathbb{1}_{\mathrm{S}} \otimes \langle e_0|), \tag{1.3.13}$$

and derive the following equation

$$\rho'_{\mathrm{S}} = \sum_i (\mathbb{1}_{\mathrm{S}} \otimes \langle\lambda_i|)\hat{U}_{\mathrm{SE}}(\mathbb{1}_{\mathrm{S}} \otimes |e_0\rangle)\rho_{\mathrm{S}}(\mathbb{1}_{\mathrm{S}} \otimes \langle e_0|)\hat{U}^{\dagger}_{\mathrm{SE}}(\mathbb{1}_{\mathrm{S}} \otimes |\lambda_i\rangle). \tag{1.3.14}$$

We omit them for the expectation value. The operators are defined by

$$\hat{M}_i = \langle\lambda_i| \hat{U}_{\mathrm{SE}} |e_0\rangle \tag{1.3.15}$$

and are also known as Kraus operators [83] or operator sum representation

$$\rho'_{\mathrm{S}} = \sum_i \hat{M}_i \rho_{\mathrm{S}} \hat{M}^{\dagger}_i. \tag{1.3.16}$$

Following from $\hat{U}_{\mathrm{SE}}$ and the completeness of λ_i we can show

$$\sum_i \hat{M}_i \hat{M}^{\dagger}_i = \sum_i \langle\lambda_i| \hat{U}_{\mathrm{SE}} |e_0\rangle \langle e_0| \hat{U}^{\dagger}_{\mathrm{SE}} |\lambda_i\rangle = \mathrm{Tr}_{\mathrm{E}}\Big(\hat{U}_{\mathrm{SE}}\hat{U}^{\dagger}_{\mathrm{SE}}\rho_{\mathrm{E}}\Big) = \mathbb{1}_{\mathrm{S}}. \tag{1.3.17}$$

The set of operators $\hat{M}_i$ build the operational elements of the quantum operation $\hat{U}_{\mathrm{SE}}$. There is a difference between quantum operation and operator sum representation. An operator sum representation is not unique. For one quantum operation, one may find different operator sum representations. The count of $\hat{M}_i$ needed for the decomposition is not given. If the sum only contains one term for illustration, the system would stay in a unitary evolution and would take a pure system state into another pure state.

Chapter 2

QAOA applied to Constrained Optimization Problems

To assess the capabilities of the Quantum Approximate Optimization Algorithm (QAOA), we compare its performance with well-studied constrained optimization problems (COPs) found in the literature, namely Max-kXOR and Max-kSAT. Our main emphasis is on assessing the performance of QAOA in relation to p and k.

This chapter is organized as follows: In Section 2.1, we provide a general introduction to the COPs, specifically the kXOR and kSAT problems. In Section 2.2, we delve into a more detailed examination of QAOA's performance for the Max-kXOR problem related to k and p. Furthermore, in Section 2.3, we analyze the performance of QAOA applied to Max-kSAT for $k = 2, 3, 4, 5$ problems and discuss the results with regard to the phase transition of the decision problem.

2.1 Constrained Optimization Problems

This section focuses on two constrained optimization problems (COPs), which we will then later incorporate into QAOA. COPs consist of a certain number of constraints (clauses). The objective is to maximize the number of satisfied clauses. The first COP is Max-kXOR, and the second is Max-kSAT. We chose these mathematical models because they are well-studied for classical algorithms.

A single constraint or clause consists of k literals that take binary values. Only certain value assignments to the variables satisfy the clause. The literals are picked from a set of variables depending on the problem size. Next to the optimization task for the problem, there is also the decision task for the problem. The decision task is to determine whether a given problem is completely satisfiable or not. To identify the optimization tasks, they are referred to as Max-kXOR and Max-kSAT.

2.1.1 kSAT constraints

In this subsection, we show how kSAT constraints, also known as kSAT clauses, are constructed for the optimization task. Furthermore, we show how a kSAT clause can be implemented on a quantum device.

The kSAT problem involves m clauses, where each clause is composed of k literals selected from n variables and their negations. The literals within a single clause are connected using the logical OR ($\vee$) operator. This representation is also known as the conjunctive normal form (CNF). A general kSAT decision problem can be written in the following form

$$C = \bigwedge_{i}^{m} \bigvee_{\{j_1, j_2, .., j_k\}} (\neg) x_{ij_k}, \tag{2.1.1}$$

where i denotes a specific clause from a set of m clauses, and j represents the chosen variable from the set of n variables. In the context of the optimization problem, the clauses are linked together using addition ($+$) instead of the logical OR ($\wedge$) used for the decision problem. When explicitly listing each clause, the subscript i is omitted. To demonstrate, we consider a randomly generated 2SAT instance

$$\begin{aligned} C = (x_1 \vee x_4) \wedge (\neg x_1 \vee x_4) \wedge (x_2 \vee \neg x_4) \wedge (x_1 \vee \neg x_3) \\ \wedge (x_1 \vee \neg x_2) \wedge (\neg x_2 \vee \neg x_3), \end{aligned} \tag{2.1.2}$$

consisting of 6 clauses. Each clause contains two literals. One way to visualize 2SAT is through an implication graph. In this graph, we can deduce two implications for each clause. Taking the first clause from Equation (2.1.2) as an example, we have

$$(x_1 \vee x_4): \quad \neg x_1 \Rightarrow x_4, \quad \neg x_4 \Rightarrow x_1. \tag{2.1.3}$$

We can construct a directed graph by drawing an arrow from the first variable to the implicated variable, as depicted in Figure 2.1. This graph represents the implication relationships in the 2SAT problem. By analyzing the implication graph, we can determine whether a 2SAT instance is satisfiable. If a variable and its negation are present in the same cycle within the graph, it implies that only certain clauses can be satisfied. This statement holds true for the 2SAT instance depicted in Figure 2.1.

In order to enable the translation of kSAT problems for quantum computers, it is

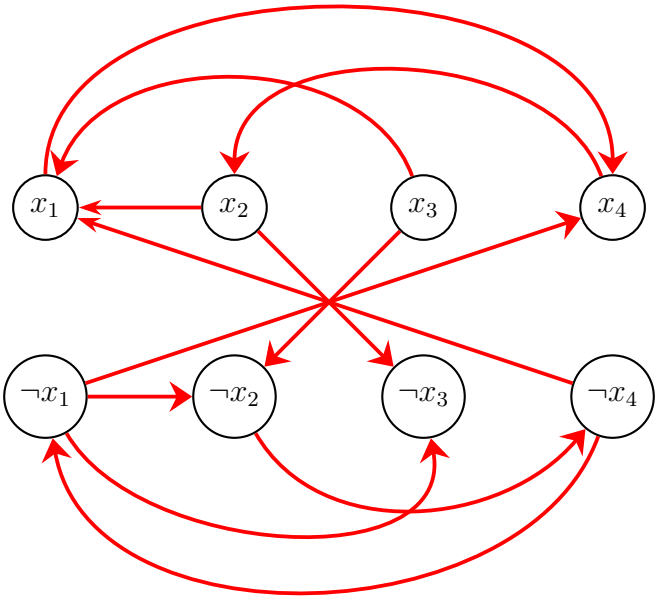


Figure 2.1: Implication graph for Equation (2.1.2). Black circles are the nodes of the variables and their negations. Red arrows are the implications if one of the literals per clause is false.

necessary to represent all clauses as k-qubit gates. We accomplish this by converting the CNF form of the problem, denoted as C, into a quantum operator represented by $\hat{C}$ and representing the variables as states in a Hilbert space. To begin, we convert the binary variables for each element in the set n into the standard basis

$$\{0,1\} \rightarrow \{(1,0)^{\mathrm{T}}, (0,1)^{\mathrm{T}}\}. \tag{2.1.4}$$

By taking the tensor product of all the standard basis states, we construct a Hilbert space of dimension 2^n.

For transforming the problem into an operator, we examine a particular strategy. Our initial focus will be on 3SAT, and subsequently, we will expand this concept to encompass kSAT. In 3SAT problems, each clause consists of three literals. Our target is to use Pauli gates on the quantum device to construct each clause. In this context, every qubit on the device represents one of the n variables. Consequently, for a given 3SAT clause, we require a 3-qubit Pauli gate to simulate a CNF clause. Our focus lies on a Pauli gate decomposition that incorporates $Z_iZ_jZ_k$, Z_iZ_j, and Z_i Pauli gates, where i, j, and k correspond to the respective qubits.

By analyzing the truth Table 2.1 of the 3SAT problem, we construct the operator that represents the clauses. The binary variables are represented as true and false statements according to $\{0,1\} \rightarrow \{\text{false}, \text{true}\}$. By extracting the rightmost column from the truth table in its binary representation, we can generate a column vector. This column vector is then used as the diagonal elements of a matrix, denoted as $\hat{C} = \text{diag}(0,1,1,...,1)$, which serves as our operator. This diagonal matrix we can decompose into the mentioned Pauli

a	b	c	$(a \vee b \vee c)$
false	false	false	false
false	false	true	true
false	true	false	true
false	true	true	true
true	false	false	true
true	false	true	true
true	true	false	true
true	true	true	true

Table 2.1: Truth table for 3SAT problem for the clause $(a \vee b \vee c)$

gates

$$\hat{C} = \text{diag}(0,1,1,1,1,1,1,1) = \alpha_1 \mathbb{1} + \alpha_2 \sum_{i=1}^{3} Z_i + \alpha_3 \sum_{i<j}^{3} Z_i Z_j + \alpha_4 Z_1 Z_2 Z_3. \tag{2.1.5}$$

For clause $(a \vee b \vee c)$ the solution $\vec{\alpha}$ is $\vec{\alpha} = (7/8, -1/8, -1/8, -1/8)$. As a general rule, the decomposition vector $\vec{\alpha}$ and the needed Z gates for any kSAT clause are

$$\begin{aligned} \hat{C} = \text{diag}(0,1,...,1) = \alpha_1 \mathbb{1} + \alpha_2 \sum_{i=1}^{k} Z_i + \alpha_3 \sum_{i<j}^{k} Z_i Z_j + ... + \alpha_{k+1} Z_1 ... Z_k, \\ \text{with } \vec{\alpha} = (\alpha_1, ..., \alpha_k, \alpha_{k+1}) = ((2^k - 1)/2^k, ..., -1/2^k, -1/2^k). \end{aligned} \tag{2.1.6}$$

Clauses with negated literals correspond to permutations of the zero on the diagonal and can be attained by applying $X_i X_j \text{diag}(0,1,...,1) X_i X_j$.

2.1.2 kXOR constraints

The second optimization problem we are investigating consists of kXOR constraints. In contrast to kSAT, the primary distinction is that the variables within a clause are linked together using the exclusive or ($\oplus$) operator. In fact, a random kXOR decision problem can be described as

$$C = \bigwedge_{i}^{m} \bigoplus_{\{j_1, j_2, .., j_k\}} x_{ij_k}. \tag{2.1.7}$$

For example, consider a 2XOR clause containing two literals per clause. An undirected graph can represent a 2XOR problem, denoted as $G(V, E)$, as shown in the right-hand side of Figure 2.2. Here, V represents the set of vertices corresponding to the variables, and E defines the edges, representing the connections between two vertices. Consequently, each set of variables corresponds to a single clause. The graph on the left side of Figure 2.2 illustrates the Max-2XOR problem, also referred to as Max-cut. For the general Max-cut problem, we only use non-negated variables.

To establish a quantum formulation of the problem, we adopt a similar procedure as we did for kSAT constraints. We apply a transformation to kXOR constraints, yielding an operator denoted as $C \to \hat{C}$. For this purpose, we investigate one clause in a 3XOR instance and extend our findings to higher values of k. Our examination involves analyzing the truth table presented in Table 2.2 for the 3XOR clause. The outcome of this truth table in binary representation is then used as the diagonal elements of a matrix, which

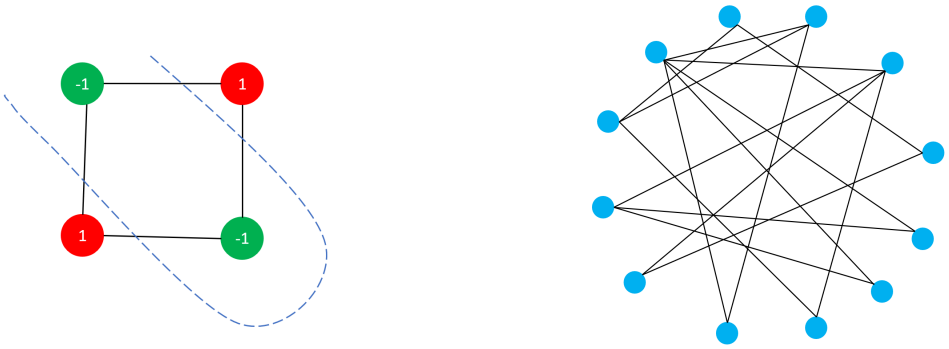


Figure 2.2: Left: G(4,4) 2-regular graph with 4 clauses and a Max-cut going through every edge leading to red and green colored partitions. Right: non-regular Graph with an unknown cut.

represents the operator $\hat{C}$. The final step involves converting the operator $\hat{C}$ into standard

a	b	c	$(a \oplus b \oplus c)$
false	false	false	false
true	true	false	true
false	true	false	true
false	true	true	false
true	false	false	true
true	false	true	false
true	true	false	false
true	true	true	true

Table 2.2: Truth table for one clause C of 3XOR

Pauli gates. A 3XOR clause can be equivalently represented by a single 3-qubit Pauli gate, such as

$$\hat{C} = \mathrm{diag}(0,1,1,0,1,0,0,1) = \frac{1}{2}(\mathbb{1} - Z_{i_1} Z_{j_2} Z_{n_3}). \tag{2.1.8}$$

The generalization of the above equation to a general kXOR reads

$$\hat{C} = \frac{1}{2}(\mathbb{1} - Z_{i_1} Z_{j_2} \dots Z_{n_k}). \tag{2.1.9}$$

When dealing with an odd number of negated variables within a clause, the minus sign in Equation (2.1.9) transforms into a plus sign.

2.2 Limitations of QAOA applied to random Max-kXOR

In this section, we focus on assessing the effectiveness of the QAOA for solving Max-kXOR problems, specifically concerning the dependence on k. We begin by providing an overview in Subsection 2.2.1 of the selected Max-kXOR problems discussed in Subsection 2.1.2. In Subsection 2.2.2 we examine the correlation of the performance of QAOA applied to Max-kXOR and the clause-to-variable ratio r. We further investigate QAOA's performance

dependence on k for Max-kXOR problems in Subsection 2.2.3.

Finally, in Subsection 2.2.4, we analyze how QAOA's performance compares across various Max-kXOR problems, particularly for cases involving large values of p.

2.2.1 Max-kXOR problems under investigation

In this section, we explore the performance of the QAOA in the context of Max-kXOR problems, with a particular focus on scenarios where $k > 2$. Notably, the application of QAOA to Max-Cut problems ($k = 2$) has been extensively scrutinized in prior research [84, 52, 41]. We specifically select random graph instances to assess QAOA's performance when applied to random Max-kXOR problems. It is worth noting that Marwaha et al. [85] have previously investigated QAOA's performance in Max-kXOR problems characterized by a unique constraint structure. In this specialized context, problems are empty of triangles, implying that no pair of variables appears in more than one constraint, and they remain non-adjacent. However, these constrained instances are rather specific and may not furnish a comprehensive overview of QAOA's overall performance trends. Furthermore, they demonstrated the bounds for the constrained Max-kXOR problems with respect to the classical threshold algorithm. Contrastingly, Basso et al. [86] have highlighted limitations in QAOA's performance when applied to random Max-kXOR problems featuring constraint graphs with even values of k. Nevertheless, to gain a more comprehensive and nuanced understanding of QAOA's performance, it is imperative to engage in numerical studies encompassing a wide spectrum of values for k and encompassing random instances of the Max-kXOR problem. Such an extensive investigation will contribute to a more robust comprehension of QAOA's performance characteristics in this domain.

In Section 2.1, we gained knowledge about how to translate the Max-kXOR problem into the following operator representation

$$\hat{C}_k = \sum_{i\ldots k \in E} g_{i\ldots k}\, Z_i \ldots Z_k, \tag{2.2.1}$$

with E being the set of edges of the problem graph. All edges appear with the same weight in the generated problem graph expressed by $g_{i\ldots k} = 1$. To quantify the performance among different system sizes and different Max-kXOR problems, we introduce the clause-to-variable ratio $r = |C|/n$. $|C|$ is the count of clauses in the problem. The number of edges is the number of variables n times the clause-to-variable ratio r. To construct a Max-kXOR instance, we randomly select a clause from the entire set of constraints $\binom{n}{k}$ with probability

$$p = \frac{r \cdot n}{\binom{n}{k}}. \tag{2.2.2}$$

Max-kXOR, like Max-cut, falls under the complexity class NP-hard [87, 88]. However, it is important to note that the decision problem of determining whether a kXOR problem is satisfiable or not belongs to the complexity class P [89].

2.2.2 QAOA performance for Max-kXOR with respect to r

In this subsection, we apply the QAOA algorithm to random Max-kXOR problems. Figure 2.3 displays the results for QAOA applied to Max-kXOR in dependence on the clause-to-variable ratio r. We investigate the performance for $p = 1, 2, 3, 4$ and $k = 3, 4, 5, 6$. First, we observe that the performance $\mathcal{I}_p$ exhibits a significant decrease for small increments

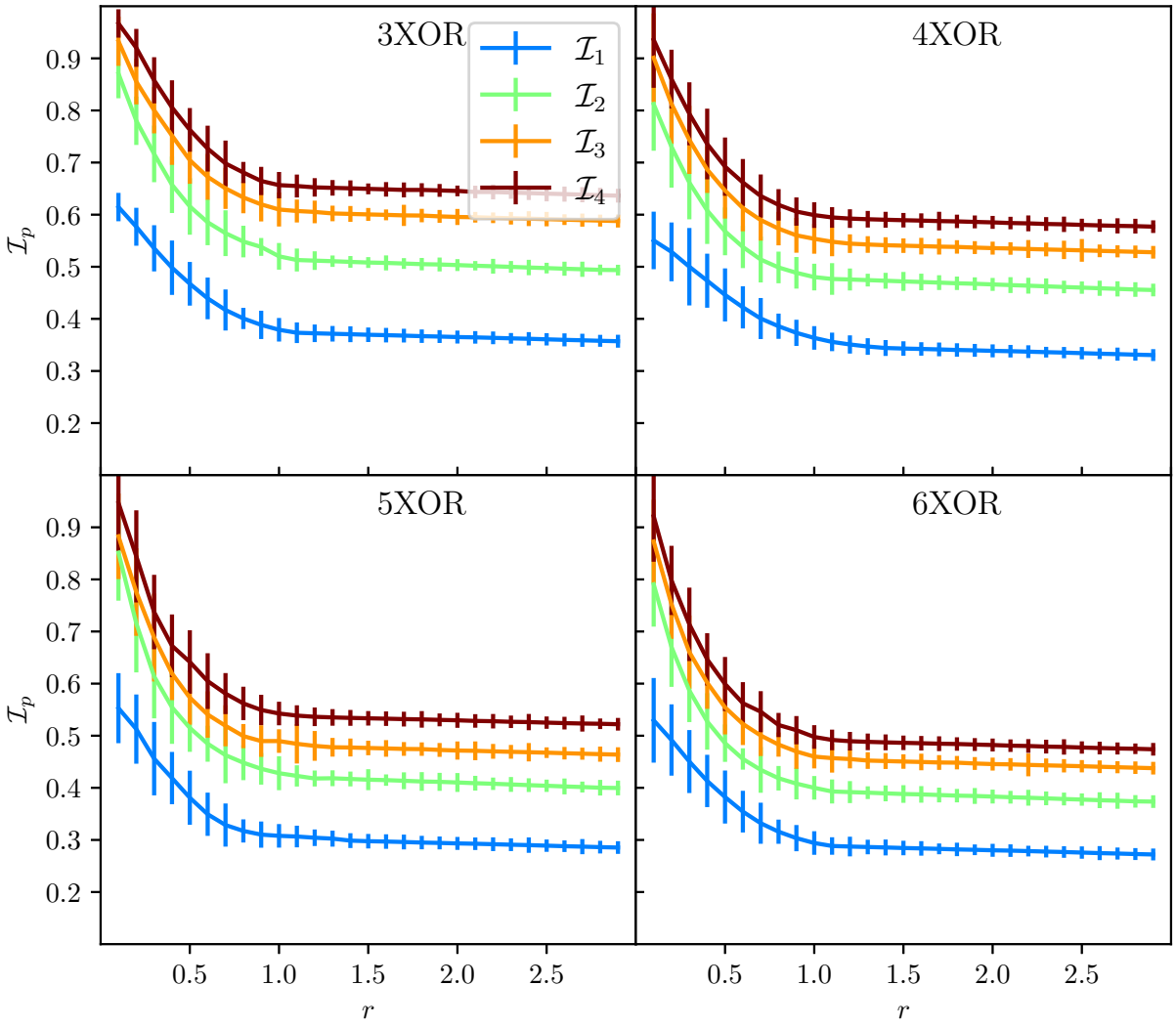


Figure 2.3: Plotting QAOA improvement over random guessing $\mathcal{I}_p$ against the clause-to-variable ratio r for Max-kXOR for $k = 4, 6, 8, 10$. For every plot, we calculate the performance of QAOA for $p = 1, 2, 3, 4$. For every r, k, p, we average over 10000 randomly created Erdős–Rényi instances for a qubit size 18.

in dr, regardless of the specific Max-kXOR problem or p for $r < 1$. For $r < 1$, the problem instances are under-constrained, and therefore, multiple solutions exist. These problems are easy to solve for an algorithm regardless of whether it is a quantum or classical algorithm. For increasing $r > 1$, the decrease in performance slows down for all p and k. For $r > 1$, the problem instances have fewer solutions than for $r < 1$ and become harder to solve for an algorithm.

In sum, we observe a dependence of the performance of QAOA applied to Max-kXOR problems on r. The QAOA performances for all k and p only differ in a vertical shift along the $\mathcal{I}_p$-axis. Furthermore, the relative performance improvement p to $p + 1$ and $p + 1$ to $p + 2$ decreases.

2.2.3 QAOA performance in dependence on k

In this subsection, we investigate how the performance of QAOA changes when applied to random Max-kXOR instances for large k. The results are shown in Figure 2.4. For $r < 1$, we have a high standard deviation. For $r > 1$, the standard deviation for all k decreases. For $r > 1$, we observe that the performance $\mathcal{I}_p$ decreases for increasing k, so we provide first indication that for random Max-kXOR for increasing k, the problems become harder to solve for QAOA.

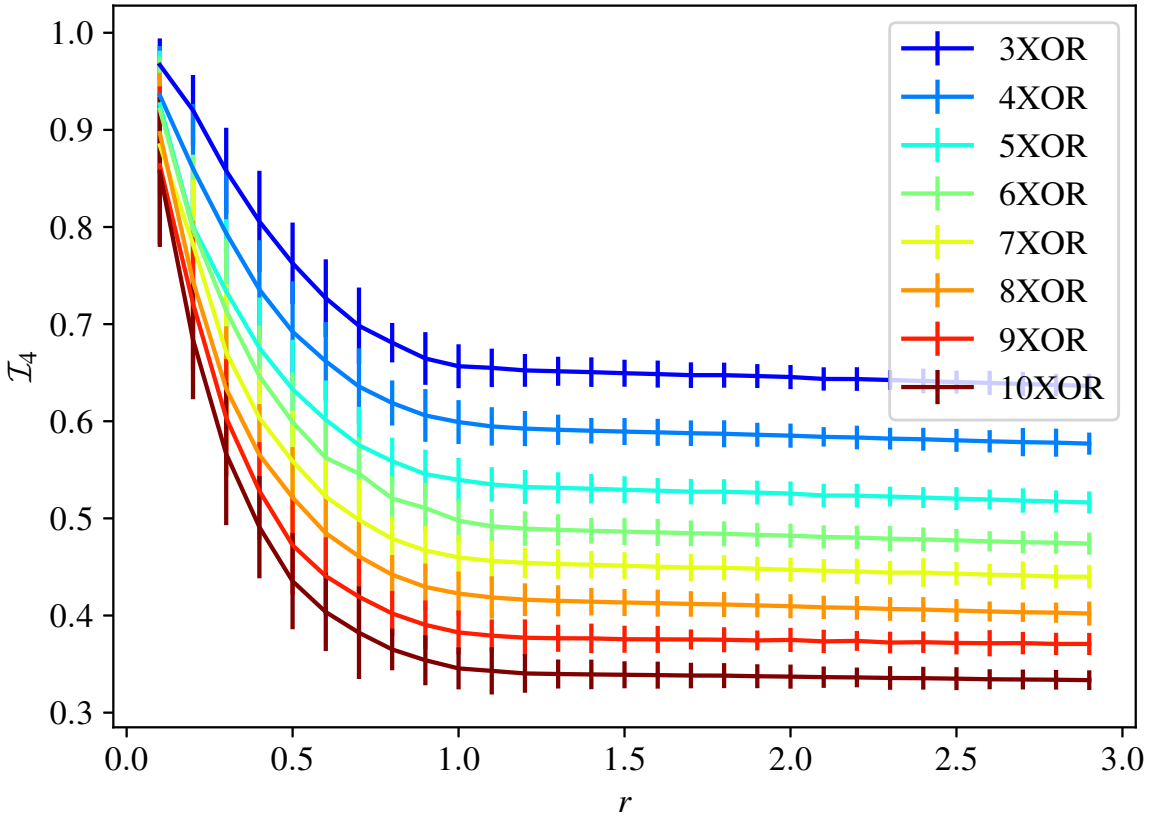


Figure 2.4: Plotting the performance of QAOA $\mathcal{I}_p$ as a function of the parameter r for randomly generated Max-kXOR instances. Our analysis covers the range of $k \in [3, 10]$. We calculate the average performance over 10000 instances for each combination of r and k.

2.2.4 QAOA performance at high circuit depth

In this section, we analyze how QAOA performs on hard-to-solve random instances of the Max-kXOR problem for large p. As previously discussed, solving instances with $r > 1$ is more difficult than those with $r < 1$, regardless of the value of k. The performance of QAOA degrades linearly with a slight negative trend for $r > 1$. Consequently, we anticipate that there won't be a significant relative difference in QAOA's performance for instances with $r > 1$, unless we approach a complete problem graph, meaning $r \cdot n \approx \binom{n}{k}$.

In Figure 2.5 we plot the results for $\mathcal{I}_p$ with $1 \leqslant p \leqslant 8$ and $3 \leqslant k \leqslant 10$. When

we compare the difference in improvement at $p = 1$ and $p = 8$ among two k values, we determine an increase in the performance difference, expressed as

$$\mathcal{I}_p^{k\text{xor}} - \mathcal{I}_p^{(k+n)\text{xor}} < \mathcal{I}_{p+m}^{k\text{xor}} - \mathcal{I}_{p+m}^{(k+n)\text{xor}}, \tag{2.2.3}$$

with $p, m > 0$. This implies that the performance improvement of the QAOA is not characterized by a linear shift when applied to different instances of Max-kXOR problems. Instead, as we increase the parameter p in QAOA, the relative improvement reduces with increasing k. Consequently, we conclude that as k grows, Max-kXOR problems become progressively more challenging to solve using the QAOA algorithm. Building upon the findings of Basso et al. [86], who demonstrated the infeasibility of QAOA for solving Max-kXOR problems with even values of k, we provide numerical evidence to substantiate that this assertion holds for all values of k. This extends and complements the investigation conducted by Marwaha et al. [85], which focused exclusively on Max-kXOR instances that were empty of triangles.

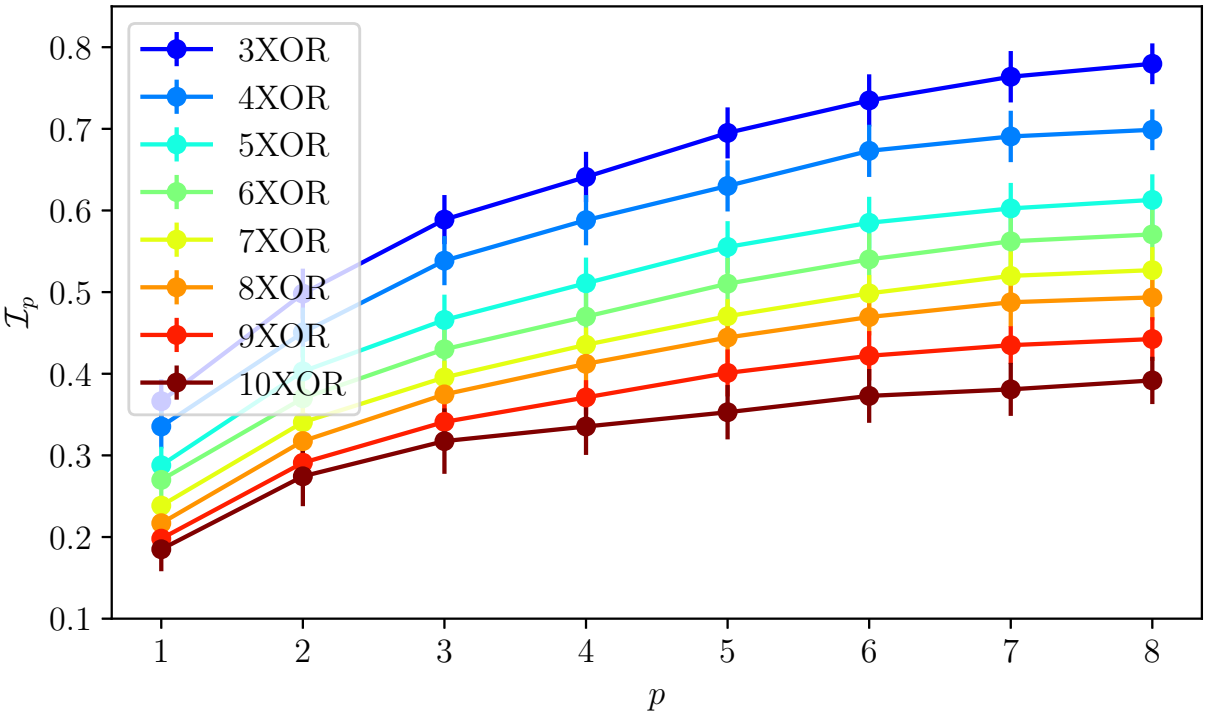


Figure 2.5: Plotting the performance $\mathcal{I}_p$ against p for Max-kXOR with $k = 4, 6, 8, 10$. We set $r = 1.5$ to investigate hard-to-solve instances for QAOA. Per p and Max-kXOR, we average over 1000 random instances.

2.3 Limitations for QAOA applied to random Max-kSAT

In this section, our focus is on examining the performance of the QAOA for Max-kSAT with $k = 2, 3, 4, 5$, as well as exploring its relationship to the satisfiability threshold r^*_{sat}. We begin by providing a brief introduction in Subsection 2.3.1, discussing what is currently known regarding Max-kSAT for classical algorithms and QAOA. Subsequently,

we analyze the performance of QAOA for Max-kSAT in Subsection 2.3.2. In Subsection 2.3.3 we investigate the performance for high circuit depth.

2.3.1 Max-kSAT problems under investigation

In this subsection, we present the random Max-kSAT problems under examination. For Max-kSAT, we have at most 2^k possible clauses per set of k variables. Therefore, the clause-to-variable ratio is

$$r = \frac{\binom{n}{k}}{n} 2^k p. \tag{2.3.1}$$

p represents the probability, where $p \in [0, 1]$, indicating the number of clauses from the complete set that we use to generate a random instance of Max-kSAT.

The classical SAT solver for the decision problem like WalkSAT [90] and Survey Propagation [91] have a dramatic decrease in performance close to the satisfiability phase transition of kSAT. For $r > r^*_{\text{sat}}$, the probability of having a solution string that satisfies all clauses decreases to zero. For $r \ll r^*_{\text{sat}}$, the probability of having a solution string that satisfies all clauses is close to 1. As the following table shows, r^*_{sat} increases with increasing k. For $r \gg r^*_{\text{sat}}$ and $r \ll r^*_{\text{sat}}$, solving the decision problem is easier compared to

kSAT	r^*_{sat}
2	1
3	4.27
4	9.93
5	21.12

Table 2.3: Table for satifiability clause-to-variable ratio r^*_{sat} for random kSAT.

being close to the satisfiability threshold. This structure is often referred to as the easy-hard-easy pattern. In the context of the Max-kSAT optimization problem, an easy-hard pattern becomes apparent. This pattern entails that as we increase the clause-to-variable ratio, the complexity of the optimization problems generally intensifies, making them more challenging to solve.

Biamonte et al. [55] studied the relationship between QAOA performance and the clause-to-variable ratio in Max-2SAT and Max-3SAT problems. Their empirical findings confirmed the existence of a threshold in the clause-to-variable ratio that significantly impacts the performance of QAOA. Once this threshold is exceeded, a noticeable decrease in QAOA performance occurs. No connection was discovered between the performance of QAOA applied to Max-2SAT and Max-3SAT and the phase transition for the respective decision problem.

2.3.2 QAOA performance of Max-kSAT with respect to r

In this subsection, we explore the performance of Max-kSAT denoted as $\mathcal{I}_1$ for the case of $p = 1$. Figure 2.6 illustrates the results of $\mathcal{I}_1$ for problem sizes $n = 15$. As anticipated, when the clause-to-variable ratio is low ($0 < r < 1$), all Max-kSAT instances exhibit similar performance, achieving $\mathcal{I}_1$ values between 0.8 and 0.85. These instances are easily solvable for all values of k, as the instances are under-constrained. This implies the

existence of multiple solutions for a single instance in this regime. Moreover, for $0 < r < 1$, we are below the satisfiability threshold $r^*_{k\text{sat}}$ of the respective decision problem for all kSAT.

For $r > 1$, the performance $\mathcal{I}_1$ starts to decrease, indicating more challenging instances appear for all Max-kSAT. In Figure 2.6, the Max-2SAT problem has a more significant

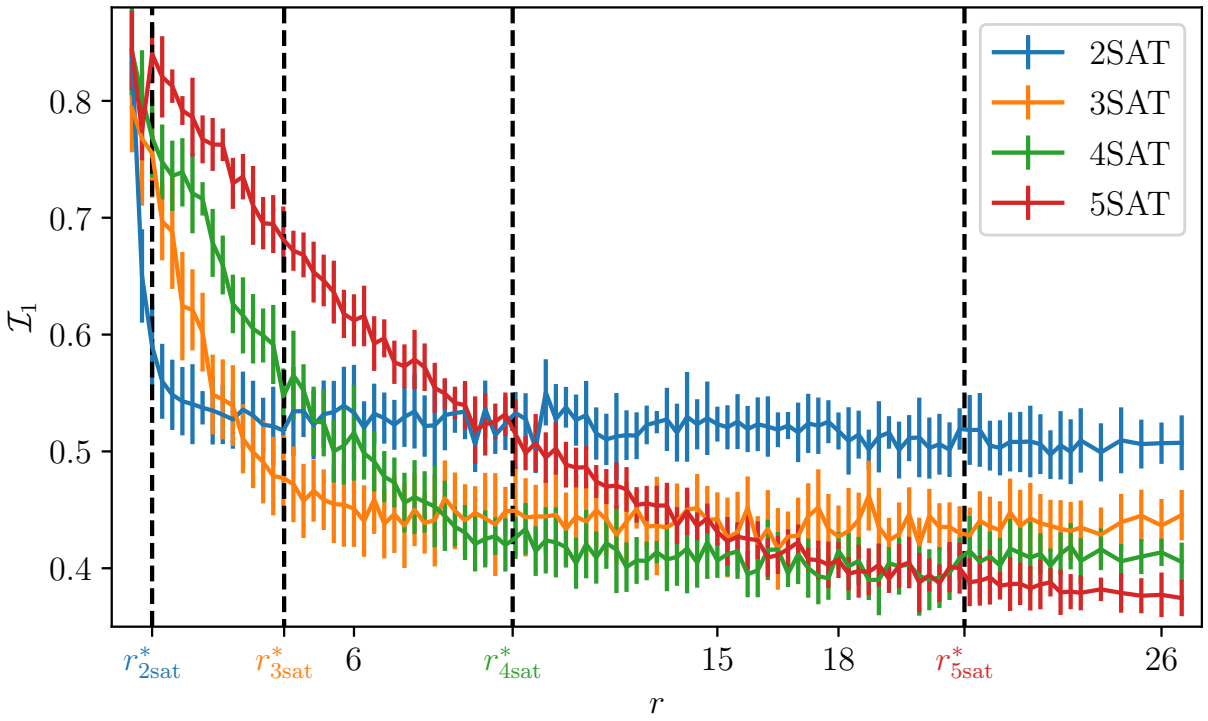


Figure 2.6: Plotting the improvement over random guessing $\mathcal{I}_1$ against clause-to-variable ratio r for $p = 1$. The blue, orange, green, and red graphs are the results for Max-kSAT with $k = 2, 3, 4, 5$. r^*_{sat} is the phase transition for the respcted kSAT decision problem. For every r of every kSAT, we average over 10000 random instances. The problem size is $n = 15$.

QAOA performance decrease for small increments of dr than for $k = 3, 4, 5$. Upon further increase of r beyond $r^*_{2\text{sat}}$ of the respected decision problem, QAOA's performance for the Max-2SAT problem demonstrates a slower change in QAOA performance for small changes dr for $r^*_{3\text{sat}} > r > r^*_{2\text{sat}}$. However, as r surpasses the threshold $r^*_{3\text{sat}}$ while still remaining below $r^*_{4\text{sat}}$, the performance of QAOA applied to Max-3SAT problem is the worst among all Max-kSAT in Figure 2.6. For Max-4SAT and Max-5SAT, we observe the same QAOA performance decrease with respect to r and the satisfiability thresholds.

To sum up, in our examination of QAOA on Max-kSAT, we observe the following trend. The worst QAOA performance among all Max-kSAT problems is observed when r crosses the threshold of satisfiability for the respective decision problem while still staying below the subsequent threshold. This trend holds true for Max-kSAT problems with k ranging from 2 to 5. These findings suggest a correlation between the satisfiability threshold and the observed performance drop in $\mathcal{I}_p(r)$.

We also determine in Figure 2.6 for $r \gg r^*_{5\text{sat}}$, the performance of QAOA decreases with increasing k. We define this phenomenon as the delocalization effect of COPs. The effect of delocalization we classify as follows. Increasing k means more literals from the

set of n variables are included in one clause. Consequently, if we change one bit of one specific solution string, we change the energy rate more for higher kSAT. In other words, if we pick a solution string for QAOA and change the string by a few Hamming distances, the result for the QAOA differs significantly from the original solution. The quantity of delocalization connects the structure of the problem to the local properties of the QAOA.

One local property of the QAOA is characterized by including Pauli-X operators in the driver Hamiltonian employed within the algorithm. This driver makes it possible to change among solution strings for QAOA in shallow depth if we have problems like $\hat{H} = \sum_{i,j} Z_i Z_j$. As soon as we increase the number of qubits in the interaction, we gain delocalization effects. Following, the QAOA with a single qubit X-driver decreases in performance. This assumption is substantiated by the results of Biamonte et al. [55]. They showed an increase in performance using the Grover driver

$$\hat{H}_{\text{Grover}} = \sum_i X_i + \sum_{i,j} X_i X_j + \ldots + \sum_{i,..,n} X_i \ldots X_n. \tag{2.3.2}$$

The Grover driver is a complete driver. $\hat{H}_{\text{Grover}}$ allows all possible transitions between bit strings and thus is of non-local structure. The study conducted in [55] demonstrated that the utilization of $\hat{H}_{\text{Grover}}$ as the driver yielded enhanced performance compared to Pauli-X operators in the context of solving 2SAT and 3SAT problems.

2.3.3 QAOA performance at high circuit depth

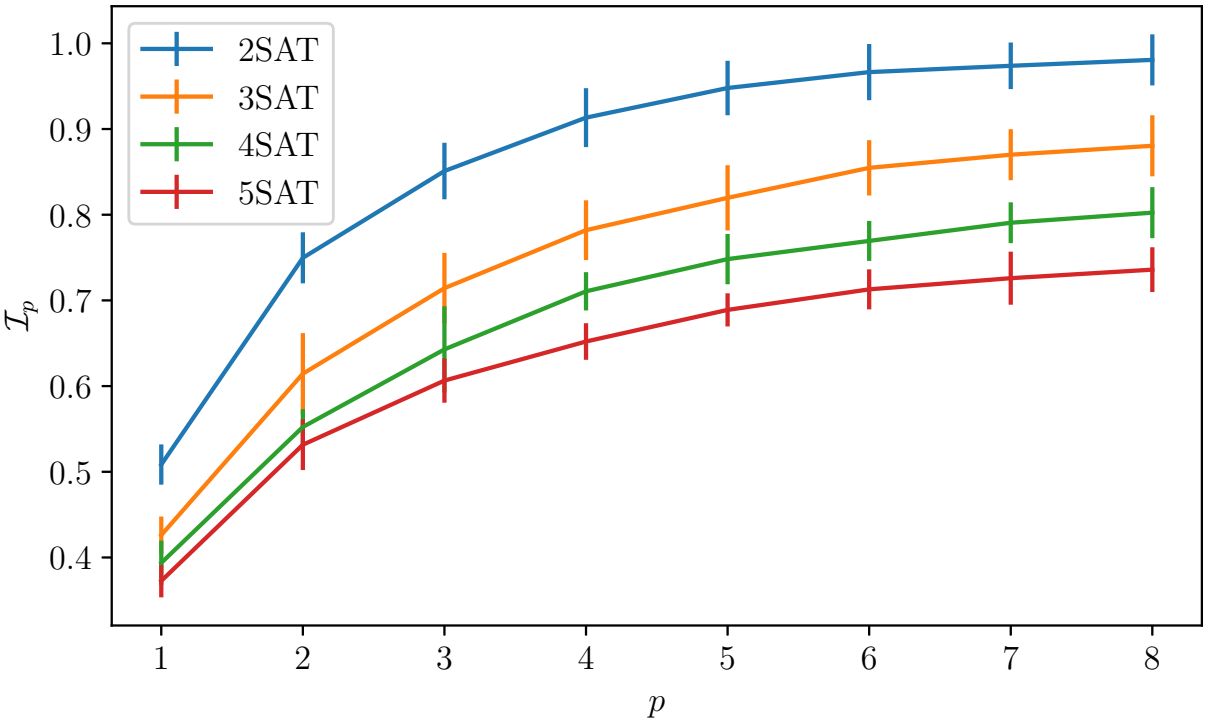


Figure 2.7: Plotting performance of QAOA $\mathcal{I}_p$ against p for kSAT problems. The plot shows the performance for k = 2(blue), 3(orange), 4(green), 5(red). We average over 1000 random Max-kSAT instances per k and per p value with $r = 25$.

In this subsection, we investigate the relative performance decrease of QAOA for large

p on hard-to-solve Max-kSAT problems. To study the performance differences among the Max-kSATs, we set $r = 25$, ensuring that we surpass the phase transition for the decision problem for every k. In Figure 2.7, we present how QAOA's performance diminishes as we increment the value of p for a specific k. Upon a closer examination of the relative performance differences, expressed as

$$\mathcal{I}_p^{k\text{sat}} - \mathcal{I}_p^{(k+n)\text{sat}} < \mathcal{I}_{p+m}^{k\text{sat}} - \mathcal{I}_{p+m}^{(k+n)\text{sat}}, \tag{2.3.3}$$

for $n, m > 0$, a clear pattern emerges. We observe that relative improvement from p to $p + 1$ decreases for increasing k. This trend suggests that Max-kSAT problems become increasingly challenging to solve using QAOA as k increases. In other words, to achieve a comparable level of performance between two Max-kSAT problems, a significantly higher value of p is required for instances with larger values of k. Generally speaking, the relative performance decrease for a specific Max-kSAT problem indicates that QAOA is unsuitable for hard-to-solve Max-kSAT problems. This is primarily due to the impracticality of running QAOA on current quantum devices when a high circuit depth is necessary to maintain acceptable performance levels.

2.4 Conclusion

In this section, we provide numerical evidence that the performance of QAOA decreases with increasing k for random Max-kXOR problems. This stands in sharp contrast to the findings of Marwaha et al. [85], who reported that QAOA's performance improves with increasing k for triangle-free instances. Furthermore, we showed that the relative improvement of QAOA applied to Max-kXOR decreases for increasing k and also for increasing p. We further showed that the performance of QAOA depends on the clause-to-variable ratio r for Max-kXOR problems. Overall, we provide evidence that QAOA applied to random Max-kXOR problems gets impracticable to run on a real device for increasing k.

We also delved into the connection between QAOA performance applied to Max-kSAT problems and the relation to the parameter k. Our investigation revealed that if r is above the phase transition point of the decision problem for all values of k, the performance of QAOA deteriorates the most when dealing with the largest values of k. Additionally, our analysis demonstrated that as k increases, the relative improvement achieved by applying QAOA to Max-kSAT problems diminishes. Consequently, we can deduce that Max-kSAT problems, where the clause-to-variable ratio exceeds the phase transition threshold of the decision problem, become increasingly impractical to solve on real quantum devices as k increases.

Chapter 3

Annealing-inspired initialization for QAOA

The Quantum Approximate Optimization Algorithm (QAOA) is a variational algorithm designed to minimize the loss function by searching for the optimal parameters, represented as $\vec{\gamma}^*$ and $\vec{\beta}^*$. To achieve this, an optimizer is employed. It is essential to have initial parameters $\vec{\gamma}$, $\vec{\beta}$ in proximity to the optimal parameters $\vec{\gamma}^*$, $\vec{\beta}^*$ to avoid the optimizer becoming stuck in a region of vanishing gradient within the parameter space. The QAOA ansatz can be viewed as an approximation of an annealing process. Sack et al. [92] were the first to investigate QAOA parameter optimization inspired by annealing protocols. Thus, our objective is to investigate various schedules for optimizing the QAOA's initial parameters. In particular, we demonstrate that linear schedules outperform the Trotterized Quantum Annealing (TQA) protocol introduced by Sack et al. [92].

This chapter primarily focuses on optimizing annealing-inspired linear schedules for the initial parameters $\vec{\gamma}$, $\vec{\beta}$. We compare linear schedules with the TQA initialization method. The chapter is structured as follows: Section 3.1 introduces the concept behind the annealing-inspired initialization technique for QAOA. In Section 3.2, we perform a detailed analysis, comparing the effectiveness of linear schedules and the TQA protocol as initialization methods for QAOA. Section 3.3 compares sigmoid schedules and TQA. Finally, in Section 3.4, we compare linear schedules with random initialization and the Fourier method, which is not annealing-inspired.

3.1 Annealing-inspired parameter schedule

In this section, we investigate the procedure to apply an annealing-inspired subroutine to find initial parameters for QAOA. We propose incorporating a subroutine that employs linear schedules for initializing the $2p$ parameters. This builds upon the TQA protocol of Sack et al. [92]. To differentiate between these two approaches, we will refer to our method as the linear schedule subroutine and to the method proposed by Sack et al. [92] as the TQA subroutine. Both subroutines optimize within a parameter space of lower dimensionality than the $2p$. The origin of the TQA method lies in the observation that a single execution of the QAOA can be viewed as a discrete representation of the quantum annealing process

$$\psi(\tau) = \hat{U}(\tau)\,|s\rangle = \mathcal{T}\exp\left(\int_0^\tau \hat{H}(t)dt\right)|s\rangle\,. \tag{3.1.1}$$

$\mathcal{T}$ is the time ordering operator and the Hamiltonian is

$$\hat{H}(\tau) = (1-\tau)\hat{H}_\mathrm{D} + \tau\hat{H}_\mathrm{P}. \tag{3.1.2}$$

$\hat{H}_\mathrm{D}$ defines the driver Hamiltonian for which we possess knowledge of the ground state, while $\hat{H}_\mathrm{P}$ corresponds to the problem Hamiltonian with an unknown ground state. The annealing parameter $\tau \in [0,1]$ is defined as $\tau = t/T$. Here, t represents the time, and T is the total runtime needed to make a smooth adiabatic transition. The total runtime depends on the distance between the ground and the first excited state (minimal energy gap ΔE). During this process, the initial state $|s\rangle$ changes from the driver's to the problem Hamiltonian's ground state. If this time evolution is slow enough, the adiabatic theorem ensures that the system remains in the ground state. In order to establish a system that relies on the driver Hamiltonian initially and transitions to the problem Hamiltonian at the final time, we build the Hamiltonian as delineated in Equation (3.1.2), aiming to satisfy the two properties: $\hat{H}(\tau=1) = \hat{H}_\mathrm{P}$ and $\hat{H}(\tau=0) = \hat{H}_\mathrm{D}$. Also, the Hamiltonian $\hat{H}$ must gradually change over time, implying that $\partial_t\hat{H} \approx 0$. Every other schedule which does not have these two properties is called a diabatic schedule [93]. We are interested in diabatic schedules because, for adiabatic quantum annealing, long runtime [94] and, therefore, high p for QAOA is necessary. TQA method discretizes the annealing time evolution into p slices

$$e^{-i\Delta t\hat{H}} \approx e^{-i\Delta t\hat{H}_\mathrm{D}}e^{-i\Delta t\hat{H}_\mathrm{P}}, \tag{3.1.3}$$

with $\gamma = \Delta t(t/T)$ and $\beta = \Delta t(1-t/T)$, and optimizes Δt. In this chapter, we want to find suitable annealing-like initializations of $\vec{\gamma}$, $\vec{\beta}$ for p, and we compare them to the TQA method.

The linear schedule subroutine works as follows: First, the p has to be fixed for the problem we want to solve. Second, decide for a schedule. Even though we can use various schedules, since we always begin with linear schedules and investigate them the most in the underlying thesis, we call the subroutine linear schedules. The freedom to choose the schedule is an extension of the TQA method. The schedule determines how many schedule parameters must be optimized in the linear schedule subroutine. For instance, three schedule parameters must be optimized for a non-symmetric linear schedule with fixed $\gamma_1 = 0$. The essential point is that the initial QAOA parameters $\vec{\gamma}(\vec{\alpha}_\mathrm{init})$, $\vec{\beta}(\vec{\alpha}_\mathrm{init})$

depend on the schedule parameters $\vec{\alpha}$ with

$$\vec{\gamma}(\vec{\alpha}_{\text{init}}) = (\gamma_1(\vec{\alpha}_{\text{init}}), \ldots, \gamma_p(\vec{\alpha}_{\text{init}})), \quad \vec{\beta}(\vec{\alpha}_{\text{init}}) = (\beta_1(\vec{\alpha}_{\text{init}}), \ldots \beta_p, (\vec{\alpha}_{\text{init}})). \tag{3.1.4}$$

When a subscript init is present, it indicates a particular type of schedule. If there is no subscript, it means no specific schedule is being referred to. For example, for a linear schedule, the QAOA parameters are distributed according to

$$\vec{\gamma}(a, b) = (a\,1 + b, \ldots a\,p + b), \quad \vec{\beta}(c, d) = (c\,1 + d, \ldots c\,p + d), \tag{3.1.5}$$

with the parameters a, b, c, d to be optimized. We proceed with the optimization of the schedule parameters $\vec{\alpha}$ in order to determine the optimal values $\vec{\alpha}^*$. The state

$$|\vec{\alpha}\rangle = \prod_{p'=1}^{p} e^{i\beta_{p'}(\vec{\alpha})\hat{H}_{\text{D}}} e^{i\gamma_{p'}(\vec{\alpha})\hat{H}_{\text{P}}} |s\rangle\,, \tag{3.1.6}$$

will be prepared on the quantum device, and the loss function

$$\max_{\vec{\alpha}} \langle \hat{H} \rangle_{\vec{\alpha}} = \langle \vec{\alpha}| \hat{H} |\vec{\alpha}\rangle \tag{3.1.7}$$

will be optimized on the CPU. The mean approximation ratio of the optimized schedule parameters we call

$$\mathcal{L}_p(\vec{\alpha}^*) = \frac{\langle \hat{H} \rangle_{\vec{\alpha}^*} - E_{\text{min}}}{E_{\text{max}} - E_{\text{min}}}, \tag{3.1.8}$$

and the mean approximation ratio for executing QAOA with optimized schedule parameters $\mathcal{M}_p(\vec{\alpha}^*)$. The depicted process in Figure 3.1 outlines the recommended approach for the subroutine aimed at achieving a low $\mathcal{L}_p(\vec{\alpha}^*) \geqslant L$. The linear schedule subroutine will

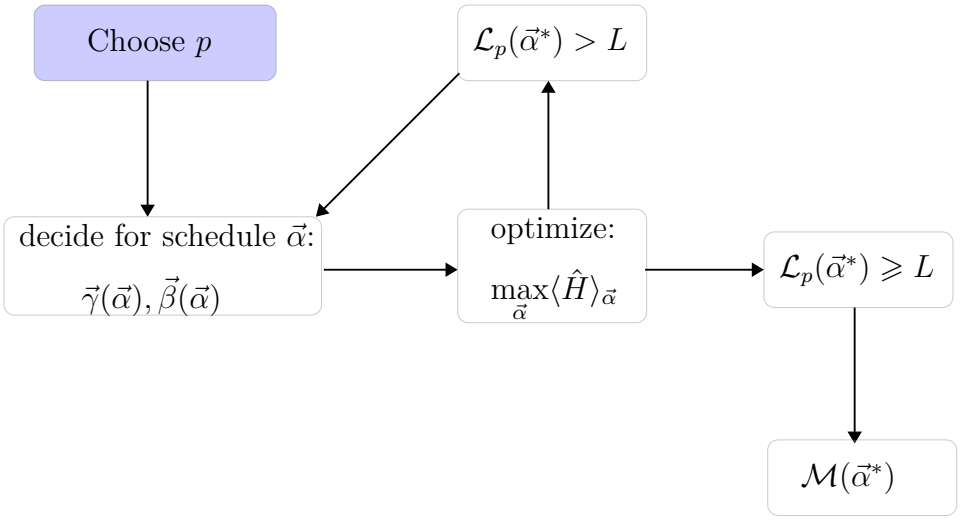


Figure 3.1: Flowchart for optimizing QAOA's initial parameters with linear schedule subroutine.

be executed until $\mathcal{L}_p(\vec{\alpha}^*) \geqslant L$, with L being a threshold value. If $\mathcal{L}_p(\vec{\alpha}^*) \geqslant L$, we have the option to run the QAOA to compute $\mathcal{M}_p(\vec{\alpha}^*)$. Alternatively, if the final state of the subroutine yields an energy that closely approximates the expected minimum energy of the problem Hamiltonian $\hat{H}_{\text{P}}$, our objective is accomplished. If $\mathcal{L}_p(\vec{\alpha}^*) < L$, a new sched-

ule $\vec{\alpha}$ can be chosen. One possible course of action could be: first, try a symmetric linear schedule with one parameter; second, choose a symmetric sigmoid schedule with two parameters; third, try a non-symmetric linear schedule with three parameters and $\gamma_1 = 0$. A significant advantage of the linear schedule subroutine is its capability to work regardless of the problem size and p-depth. Especially for highly non-degenerate and large-sized problems, this approach can help find suitable initial parameters. Another way to achieve a high $\mathcal{L}_p$ is to increase p until attaining coherence time to reach annealing-like conditions for QAOA. If the annealing time is sufficiently large, the quantum annealing algorithm will find the optimal solution due to the adiabatic theorem.

We will provide evidence that this subroutine is less complicated due to more local minima close to the global minimum for a particular p. This leads to less state preparation on the quantum computer than executing QAOA with random initial parameters. We will provide evidence demonstrating that the linear scheduling subroutine surpasses the TQA protocol in cases of shallow circuit depth when considering the mean approximation ratios $\mathcal{L}_p(\vec{\alpha}^*)$ and $\mathcal{M}_p(\vec{\alpha}^*)$. Furthermore, we will show that linear schedules and the TQA method outperform random initialization methods due to less required optimizer executions.

3.2 Applying linear schedules to QAOA parameters

In this section, we present the enhanced initialization of QAOA by utilizing linear schedules. Firstly, in Subsection 3.2.1, we demonstrate the superior performance of linear schedules compared to TQA in terms of $\mathcal{L}_p$ in shallow circuit depth. Building upon this outcome, in Subsection 3.2.2, we delve into the examination of the performance of linear schedules and TQA on $\mathcal{M}_p$. Subsequently, in Subsection 3.2.3, we showcase the extent to which each parameter deviates from its initial values and the final QAOA parameters for both TQA and linear schedules. Additionally, in Subsection 3.2.4, we provide a time perspective on the parameter results from the previous subsections. Finally, in Subsection 3.2.5, we establish the independence of our results from the system size as n increases.

3.2.1 Optimized QAOA start parameters

First, we apply linear schedules to a modified version of the Sherrington-Kirkpatrick [95] (SK) model

$$\hat{H} = \sum_{i,j} J_{i,j} Z_i Z_j + \sum_i h_i Z_i. \tag{3.2.1}$$

In this case, we have single-qubit and two-qubit coupling with $J_{i,j}$, and h_i coupling strengths picked from a Gaussian distribution with mean zero $\langle J_{i,j} + h_i \rangle = 0$ and variance $\sigma(J_{i,j}) = \sigma(h_i) = 1$. The problem graph is fully connected. h_i is an exterior magnetic field. Barahona et al. [96] demonstrated that the problem of locating the lowest eigenvalue of $\hat{H}$ is NP-hard, thus proving the impossibility of finding a polynomial-time algorithm for its solution. This challenge makes it particularly appealing to quantum computing. The original SK model has no exterior magnetic field, and the problem size normalizes the coupling strength $J_{i,j}$. The only parameter to determine for TQA is β_1. Equation (3.1.6) reduces to

$$|\vec{\alpha}\rangle = \prod_{p'=1}^{p} e^{i((-\beta_1/p)\,(p'-1)+\beta_1)\,\hat{H}_\mathrm{D}} e^{i(\beta_1/p)\,(p'-1)\hat{H}_\mathrm{P}} |s\rangle \tag{3.2.2}$$

and for a general linear schedule with $\gamma_1 = 0$ to

$$|\vec{\alpha}\rangle = \prod_{p'=1}^{p} e^{i((-(\beta_p-\beta_1)/p)\,(p'-1)+\beta_1)\,\hat{H}_{\mathrm{D}}} e^{i(\gamma_p/p)\,(p'-1)\hat{H}_{\mathrm{P}}}\,|s\rangle\,. \tag{3.2.3}$$

For TQA, we use the notation $\vec{\alpha}_{\mathrm{TQA}}$ to represent the parameter schedule, while for a linear schedule, we utilize $\vec{\alpha}_{\mathrm{ls}}$. Additionally, we introduce $\vec{\alpha}_{\mathrm{TQA},p=9}$ and $\vec{\alpha}_{\mathrm{ls},p=9}$ to signify the linear schedule and the TQA protocol for $p = 9$. Figure 3.2 shows the result for

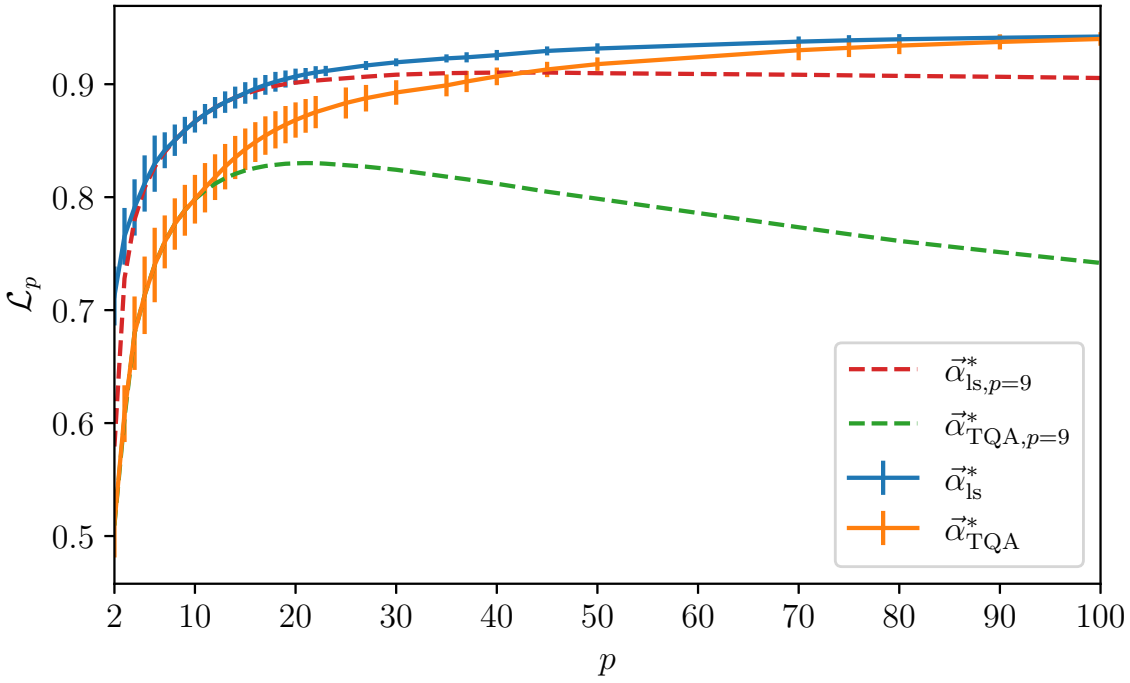


Figure 3.2: Depicting the mean approximation ratio $\mathcal{L}_p$ plotted against the parameter p, ranging from $p = 2$ to $p = 100$, considering the optimal schedule parameters $\vec{\alpha}^*_{\mathrm{TQA}}$ and $\vec{\alpha}^*_{\mathrm{ls}}$. Furthermore, we illustrate $\mathcal{L}_p$ for $\vec{\alpha}^*_{\mathrm{TQA},p=9}$ and $\vec{\alpha}^*_{\mathrm{ls},p=9}$. The parameter space for $\vec{\alpha}_{\mathrm{ls}}$ is set to $[\beta_1, \beta_p, \gamma_p] \in [0, 2\pi] \times [0, 2\pi] \times [0, 2\pi]$, while for $\vec{\alpha}_{\mathrm{TQA}}$, it is constrained to $\beta_1 \in [0, 2\pi]$. The optimization of these schedules is carried out using the BFGS optimizer. Our analysis encompasses an ensemble averaging over 1000 instances for each value of p and its corresponding schedule. The Powell optimizer yields comparable outcomes despite its independence from gradient-based techniques.

$\mathcal{L}_p$ for the schedules $\vec{\alpha}^*_{\mathrm{ls}}$, $\vec{\alpha}^*_{\mathrm{TQA}}$, $\vec{\alpha}^*_{\mathrm{ls},p=9}$, $\vec{\alpha}^*_{\mathrm{TQA},p=9}$ up to $p = 100$. Accordingly, the mean approximation ratio for random guessing will be exactly $\mathcal{L}_p = 0.5$. Thus, $\mathcal{L}_p$=0.5 means we have no improvement over random guessing. For $1 < p < 10$, the schedules' mean approximation ratio difference is at least

$$\mathcal{L}_p(\vec{\alpha}^*_{\mathrm{ls}}) - \mathcal{L}_p(\vec{\alpha}^*_{\mathrm{TQA}}) > 0.08. \tag{3.2.4}$$

For $p = 2$ and $\vec{\alpha}_{\mathrm{ls}}$, we achieve an improvement over random guessing of 0.2. Whereas for TQA, we reach an improvement of 0.02. Sack et al. [92] achieved similar mean approximation ratios when applying TQA protocols to Max-cut instances. For $p \leqslant 10$ and $\vec{\alpha}_{\mathrm{TQA}^*,p=9}$,

$\mathcal{L}_p(\vec{\alpha}_{\mathrm{TQA}^*,p=9})$ exhibits an optimal value due to accordance with $\mathcal{L}_p(\vec{\alpha}^*_{\mathrm{TQA}})$. Similarly, it is observed that $\mathcal{L}_p(\vec{\alpha}_{\mathrm{ls}^*,p=9})$ aligns with $\mathcal{L}_p(\vec{\alpha}^*_{\mathrm{ls}})$ within the range of $3 < p < 13$. Drawing from this insight, optimized scheduling parameters for a given p yield an optimal mean approximation ratio for at least $p-1$ and $p+1$, corresponding to $\mathcal{L}_{\mathrm{ls},p-1}(\vec{\alpha}^*_{\mathrm{ls},p}) \approx \mathcal{L}_{\mathrm{ls},p-1}(\vec{\alpha}^*_{\mathrm{ls},p-1})$ and $\mathcal{L}_{\mathrm{ls},p+1}(\vec{\alpha}^*_{\mathrm{ls},p}) \approx \mathcal{L}_{\mathrm{ls},p+1}(\vec{\alpha}^*_{\mathrm{ls},p+1})$, respectively. This suggests that as p increases, there is a gradual displacement of the global minimum's position within the $\vec{\alpha}$-parameter space. This result was also ascertained for Max-cut 3-regular unweighted graphs [92].

To summarize, linear schedules outperform TQA protocols regarding the mean approximation ratios. Moreover, as illustrated in Figure 3.2, we have shown that both TQA and linear schedules allow us to approach a mean approximation ratio close to the global minimum for high circuit depth. This observation holds significance because as the value of p increases, the optimization task becomes more challenging for QAOA with randomly initialized parameters, primarily due to the increased likelihood of encountering barren plateaus, as noted in prior research [97]. The number of required optimizer executions increases exponentially $2^{O(p)}$ [45]. In the case of TQA and linear schedules, achieving a solution that outperforms the mean approximation ratio achieved by random parameter guessing requires only a single execution of the optimizer.

3.2.2 Executing QAOA with optimized start parameters

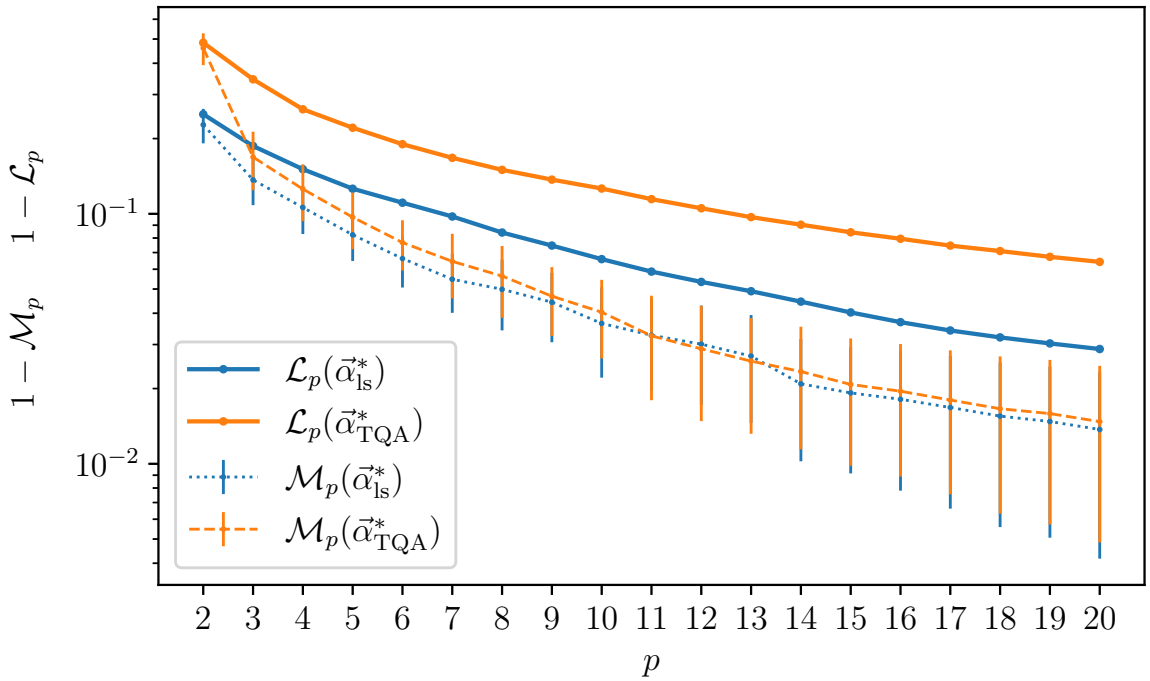


Figure 3.3: Plotting $1-\mathcal{M}_p$ and $1-\mathcal{L}_p$ against p in a log-linear scale. Solid lines indicate mean approximation ratio $\mathcal{L}_p$ for optimized schedules for TQA (orange line) and linear schedules (blue line) taking from 3.2. The dotted blue line ($\mathcal{M}_p(\vec{\alpha}^*_{\mathrm{ls}})$) and dashed orange line ($\mathcal{M}_p(\vec{\alpha}^*_{\mathrm{TQA}})$) show the respective QAOA results. We use BFGS optimizer on 1000 random SK instances per p. Remark: For every instance, two optimizer executions are applied.

We now examine the difference in mean approximation ratios $\mathcal{M}(\vec{\alpha}^*_{\text{ls}})$, $\mathcal{M}(\vec{\alpha}^*_{\text{TQA}})$ for QAOA with TQA and linear schedules. Figure 3.3 shows $1-\mathcal{M}_p$ and $1-\mathcal{L}_p$ for TQA and linear schedules. We use the local optimizer BFGS for this simulation. For every p, we observe an improvement from $1-\mathcal{L}_p$ to $1-\mathcal{M}_p$ for TQA and linear schedules parameters. Consequently, the linear schedules approach does not lead to the problem of starting on a barren plateau. Only for $p=2$, we almost have no improvement between $1-\mathcal{L}_p$ and $1-\mathcal{M}_p$ for both schedules. The enhancement grows as p increases for both schedules, which agrees with Guerreschi et al.[98]. Guerreschi et al. [98] proved that a greater number of local minima in proximity to the global minimum manifest as the parameter space expands. Furthermore, we determine that for $p \leqslant 10$, we always have a greater $\mathcal{M}_p$ for linear schedules than for TQA. This is essential since only short circuits can be executed on a quantum computer with high accuracy. Hence, linear schedules outperform TQA protocols. For $p > 10$, the difference between TQA and linear schedules initialization in $\mathcal{M}_p$ is less than 1%. For further investigations, we introduce three new quantities

$$d_{\text{init}} = \left|(\overrightarrow{\gamma\beta}(\vec{\alpha}^*_{\text{init}}) - (\overrightarrow{\gamma\beta})^*_{\text{init}}\right|, \; d^* = \left|(\overrightarrow{\gamma\beta})^*_{\text{TQA}} - (\overrightarrow{\gamma\beta})^*_{\text{ls}}\right|, \tag{3.2.5}$$
$$\Delta_{\text{init}} = \mathcal{M}_p(\vec{\alpha}^*_{\text{init}}) - \mathcal{L}_p.$$

d is a distance in parameter space, and Δ is the difference in the mean approximation ratio. The subscript indicates the schedule method used for the subroutine. d_{init} is the distance in parameter space between $\vec{\gamma}$, $\vec{\beta}$ using optimized schedule parameters and $\vec{\gamma}^*$, $\vec{\beta}^*$. d^* is the distance in parameter space between optimal QAOA angles $\vec{\gamma}^*$, $\vec{\beta}^*$ initialized by TQA $\vec{\gamma}(\vec{\alpha}^*_{\text{TQA}})$, $\vec{\beta}(\vec{\alpha}^*_{\text{TQA}})$ and linear schedules $\vec{\gamma}(\vec{\alpha}^*_{\text{ls}})$, $\vec{\beta}(\vec{\alpha}^*_{\text{ls}})$. Finally, we compare the relative improvement Δ_{init} in the mean approximation ratio before and after executing QAOA. Table 3.1 shows the quantifier results. All the quantities in the table have been obtained

Quantity	$p=3$	$p=7$	$p=12$	$p=15$	$p=18$
$\langle\Delta_{\text{TQA}}\rangle$	0.17	0.1	0.076	0.063	0.056
$\langle\Delta_{\text{ls}}\rangle$	0.05	0.042	0.023	0.021	0.017
$\langle d_{\text{TQA}}\rangle$	0.41	0.42	0.78	0.78	0.89
$\langle d_{\text{ls}}\rangle$	0.17	0.18	0.22	0.38	0.71
$\langle d^*\rangle$	0.68	0.65	1.2	1.35	1.4

Table 3.1: Depicted are 3 quantifiers for 5 different p values for TQA and linear schedules to show the difference in both methods. The quantifiers Δ_{init}, d^*, d_{init} we refer to 3.2.2, init is the respected initialization method, and $\langle . \rangle$ is the average over 1000 instances.

using the BFGS local optimizer. Similar values for the quantities can be achieved when using other local optimizers. The average distance between $\vec{\gamma}^*$, $\vec{\beta}^*$ and $\vec{\gamma}(\vec{\alpha}^*_{\text{TQA}})$, $\vec{\beta}(\vec{\alpha}^*_{\text{TQA}})$ indicated by $\langle d_{\text{TQA}}\rangle$ is larger than for linear schedules. Therefore, we conclude that linear schedules' optimal parameters are closer to local minima in the $2p$ dimensional parameter space than TQA's schedule. Executing QAOA with linear schedules initialization leads to a greater $\mathcal{M}$ than for TQA initialization. Consequently, the optimizer is less likely to explore a decreasing gradient region. The statement gains additional reinforcement from the fact that the average distance of $\langle d^*\rangle$ slightly increases as p grows. As a result, TQA

and linear schedules converge to distinct points in the QAOA parameter space for their final parameters $\vec{\gamma}^*(\vec{\alpha}^*_{\mathrm{TQA}})$, $\vec{\beta}^*(\vec{\alpha}^*_{\mathrm{TQA}})$ and $\vec{\gamma}^*(\vec{\alpha}^*_{\mathrm{ls}})$, $\vec{\beta}^*(\vec{\alpha}^*_{\mathrm{ls}})$. Besides, the relative improvement from $\mathcal{L}_p$ to $\mathcal{M}_p$ increases more for TQA than for linear schedules, as expressed by $\Delta_{\mathrm{TQA}} > \Delta_{\mathrm{ls}}$.

3.2.3 Parameter deviation after executing QAOA

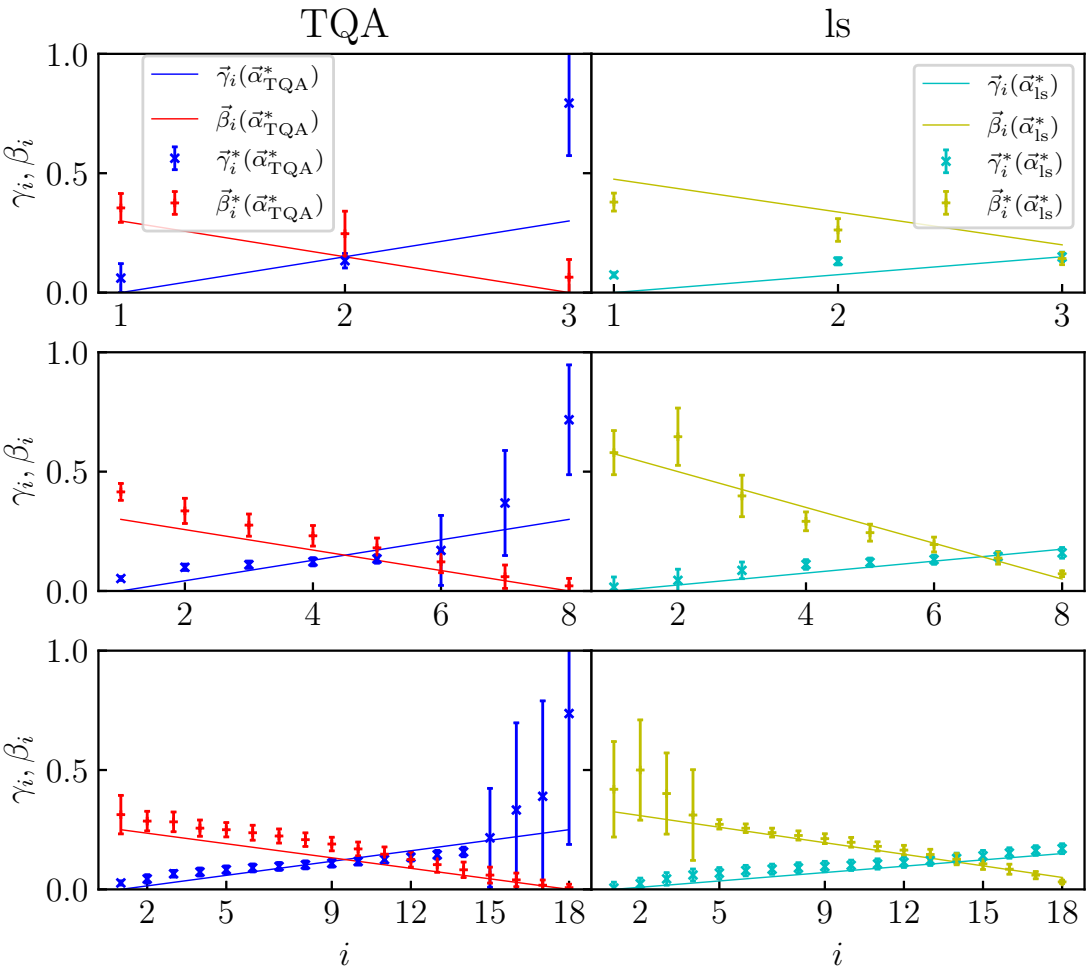


Figure 3.4: Plotting parameter values γ_i, β_i against multiple p. Left column: solid blue, red line indicates the starting parameters $\gamma_i(\vec{\alpha}^*_{\mathrm{TQA}})$, $\beta_i(\vec{\alpha}^*_{\mathrm{TQA}})$ with the optimal TQA initialization. Blue, red markers indicate the final parameters $\gamma^*_i(\vec{\alpha}^*_{\mathrm{TQA}}), \beta^*_i(\vec{\alpha}^*_{\mathrm{TQA}})$. The 3 rows correspond to $p = 3, 8, 18$ (top to bottom). Right column: solid cyan, yellow lines indicate optimal linear schedule initializations for $\gamma_i(\vec{\alpha}^*_{\mathrm{ls}})$, $\beta_i(\vec{\alpha}^*_{\mathrm{ls}})$. Cyan, and yellow markers are the respected final parameters $\gamma^*_i(\vec{\alpha}^*_{\mathrm{ls}})$, $\beta^*_i(\vec{\alpha}^*_{\mathrm{ls}})$. For every single plot, we average over 1000 random Sk instances.

In this subsection, we examine the change in $\vec{\gamma}(\vec{\alpha}^*)$, $\vec{\beta}(\vec{\alpha}^*)$ and $\vec{\gamma}^*(\vec{\alpha}^*)$, $\vec{\beta}^*(\vec{\alpha}^*)$ regarding TQA and linear schedules initialization. Figure 3.4 illustrates the outcomes for $p = 3, 8, 18$. We commence with the examination of Figure 3.4 for $p = 3$. For TQA, we

observe that $\vec{\gamma}(\vec{\alpha}^*_{\text{TQA}})$, $\vec{\beta}(\vec{\alpha}^*_{\text{TQA}})$ and $\vec{\gamma}^*(\vec{\alpha}^*_{\text{TQA}})$, $\vec{\beta}^*(\vec{\alpha}^*_{\text{TQA}})$ do not overlay within the standard deviations. $\beta(\vec{\alpha}^*_{\text{TQA}})$ and the optimal QAOA parameters $\beta^*(\vec{\alpha}^*_{\text{TQA}})$ are significant close, with a difference of only 0.1 and low standard deviation (≈ 0.075 on average) across all p_i. However, TQA suffers from a poor initialization in $\vec{\gamma}(\vec{\alpha}^*_{\text{TQA}})$. The optimal QAOA parameters $\vec{\gamma}^*(\vec{\alpha}^*_{\text{TQA}})$ exhibit a notable difference in $\gamma^*_3(\vec{\alpha}^*_{\text{TQA}})$ compared to $\gamma^*_3(\vec{\alpha}^*_{\text{TQA}})$, with a high standard deviation. Whereas for linear schedules at $p = 3$, $\vec{\gamma}(\vec{\alpha}^*_{\text{ls}})$, $\vec{\beta}(\vec{\alpha}^*_{\text{ls}})$ and $\vec{\gamma}^*(\vec{\alpha}^*_{\text{ls}})$, $\vec{\beta}^*(\vec{\alpha}^*_{\text{ls}})$ differ by less than 0.1 for every element.

Next, we investigate the scenario with $p = 8$, corresponding to the second row in Figure 3.4. TQA and linear schedules deviations between $\vec{\gamma}(\vec{\alpha}^*)$, $\vec{\beta}(\vec{\alpha}^*)$ and $\vec{\gamma}^*(\vec{\alpha}^*)$, $\vec{\beta}^*(\vec{\alpha}^*)$ are similar to $p = 3$. The parameters of linear schedules have fewer deviations between optimal schedule and optimal QAOA parameters. Whereas for TQA, the most deviations arise between $\gamma^*_i(\vec{\alpha}^*_{\text{TQA}})$ and $\gamma_i(\vec{\alpha}^*_{\text{TQA}})$. For higher p, the deviation between the initial and final parameters increases. Additionally, the standard deviations increase.

For $p = 18$ in Figure 3.4, the disparities between $\vec{\gamma}(\vec{\alpha}^*)$, $\vec{\beta}(\vec{\alpha}^*)$, and $\vec{\gamma}^*(\vec{\alpha}^*)$, $\vec{\beta}^*(\vec{\alpha}^*)$ are largely comparable to those observed for $p = 3$ and $p = 8$. However, there is an exception as we observe an increase in the deviation between $\vec{\beta}^*(\vec{\alpha}^*_{\text{ls}})$ and $\vec{\beta}(\vec{\alpha}^*_{\text{ls}})$ for $p \leqslant 4$. Linear schedules initialization is generally more stable than TQA. As we increase the value of p we find that there is no significant deviation between $\vec{\gamma}^*(\vec{\alpha}^*_{\text{ls}})$ and $\vec{\gamma}(\vec{\alpha}^*_{\text{ls}})$. Additionally, we discover that the standard deviation of each parameter, $\vec{\gamma}^*(\vec{\alpha}^*_{\text{ls}})$ and $\vec{\beta}^*(\vec{\alpha}^*_{\text{ls}})$, is consistently lower than 0.02 for $p \geqslant 4$. While the standard deviation for the first one-third of $\vec{\beta}^*(\vec{\alpha}^*_{\text{ls}})$ elements increases as p increases, this increase is not as significant as the rise in TQA standard deviation noted in $\vec{\gamma}^*(\vec{\alpha}^*_{\text{TQA}})$. These conclusions align with the results in Table 3.1.

3.2.4 QAOA parameters as time evolution

In this subsection, our attention is directed toward examining the time representation of $\vec{\gamma}(\vec{\alpha}_{\text{ls}}), \vec{\gamma}(\vec{\beta}_{\text{ls}})$.The time analysis aims to elucidate how optimal and near-optimal linear schedules change with respect to the mean approximation ratio as the value of p increases, and we can obtain the operational time.

The QAOA can also be interpreted as an approach where we turn on and off two Hamiltonians for certain amounts of time Δt_i with the unitary time evolution

$$\begin{aligned} U(T_{\text{tot}}) = \exp\Big(-i\hat{H}_{\text{D}}\Delta t^{\text{D}}_p\Big)\exp\Big(-i\hat{H}_{\text{P}}\Delta t^{\text{P}}_p\Big)\exp\Big(-i\hat{H}_{\text{D}}\Delta t^{\text{D}}_{p-1}\Big)\exp\Big(-i\hat{H}_{\text{P}}\Delta t^{\text{P}}_{p-1}\Big)\times \\ \ldots \times \exp\Big(-i\hat{H}_{\text{D}}\Delta t^{\text{D}}_1\Big)\exp\Big(-i\hat{H}_{\text{P}}\Delta t^{\text{P}}_1\Big). \end{aligned} \tag{3.2.6}$$

The problem $\hat{H}_{\text{P}}$ or the driver Hamiltonian $\hat{H}_{\text{D}}$ are turned on for a duration of Δt_i, during which the other Hamiltonian is turned off. This approach to the QAOA is commonly referred to as the bang-bang approach and has been studied in [99, 100]. The sum of the times per Hamiltonian can be viewed as the system's operational time

$$T_{\text{tot}} = \sum_{i=1}^{p}\Big(\Delta t^{\text{D}}_i + \Delta t^{\text{P}}_i\Big). \tag{3.2.7}$$

We define

$$T_{\text{D}} = \sum_{i=1}^{p}\Delta t^{\text{D}}_i \quad \text{and} \quad T_{\text{P}} = \sum_{i=1}^{p}\Delta t^{\text{P}}_i \tag{3.2.8}$$

as the driver and the problem time. If we restrict to TQA protocols, we limit the total time to

$$\sum_{i=1}^{p} \Delta t_i^{\mathrm{D}} = \sum_{i=1}^{p} \Delta t_i^{\mathrm{P}} = \frac{T_{\mathrm{tot}}}{2}. \tag{3.2.9}$$

When using linear schedules, we consider the times (parameters) Δt_p^{P} and Δt_p^{D} to be independent of each other. These parameters correspond to the angles γ_i and β_i. It is necessary to establish a specific range for these parameters. In the context of the Sherrington-Kirkpatrick model, the range of values for Δt_p^{D} and Δt_p^{P} is constrained to the interval $[0, 2\pi]$. This constraint on Δt_p^{D} arises from the periodicity of Pauli-X rotation gate. Conversely, when examining $\exp(-i\hat{H}_{\mathrm{P}}\Delta t_i^{\mathrm{P}})$ in the context of the SK model, this periodic behavior does not occur within the interval $[0, 2\pi]$. Notably, in the SK model, an observation is made that γ^* and consequently $\Delta t_p^{\mathrm{P}*}$ exhibit a tendency to remain within the interval $[0, 2\pi]$. By combining the definition presented in Equation (3.2.4) for T_{tot} with the definition of linear schedules in Equation (3.1) with $\gamma_1 = 0$, it is possible to determine the value of T_{tot} with the help of γ_1, β_1, γ_2, and β_2. The initial step in this process is to calculate T_{D}

$$\begin{aligned} T_{\mathrm{D}} &= \sum_{i=1}^{p}(\beta_1 + (i-1)(\beta_2 - \beta_1)) = (2\beta_1 - \beta_2)p + (\beta_2 - \beta_1)\sum_{i=1}^{p} i \\ &= (2\beta_1 - \beta_2)p + (\beta_2 - \beta_1)\frac{p(p+1)}{2}. \end{aligned} \tag{3.2.10}$$

Analog to the previous equation, we can deduce the equation for T_{P}. Both equations allow us to to associate the times T_{D} and T_{P} as a function of $\beta_1\beta_p, \gamma_1\gamma_p$. Because we convert 4 parameters from the schedule into two time parameters

$$\{\gamma_1, \gamma_p\} \to T_{\mathrm{P}}, \quad \text{and} \quad \{\beta_1, \beta_p\} \to T_{\mathrm{D}}, \tag{3.2.11}$$

the conversion into $T_{\mathrm{D}}, T_{\mathrm{P}}$ space is not unique. For example, setting $\beta_1 = 1, \beta_p = 0$ or $\beta_1 = 0, \beta_p = 1$ results in different schedules and mean approximation ratios, but the driver's time T_{D} remain the same. In order to exclude solutions with the same time but a lower $\mathcal{L}_p$, we will focus on schedules with an improvement over random guessing, meaning $\mathcal{L}_p > 0.5$.

Figure 3.5 displays the mean approximation ratio for linear schedules $\mathcal{L}_p(\vec{T}_{\mathrm{ls}})$ and TQA $\mathcal{L}_p(\vec{T}_{\mathrm{TQA}})$ in the T_{D}, T_{P} space for $p = 3, 4, 6, 18$. For $p = 3$, the optimal linear schedule attains an average mean approximation ratio of $\vec{T}^*_{\mathrm{ls}} = 0.85$. In the parameter space, $\vec{T}^*_{\mathrm{ls}}$ is situated amidst numerous local minima, each yielding $\mathcal{L}_3(\vec{T}_{\mathrm{ls}})$ values that closely approach the global minimum. These local minima have an approximation ratio that differs from the optimal solution by approximately $\mathcal{L}_3(\vec{T}^*_{\mathrm{ls}}) - \mathcal{L}_3(\vec{T}^*_{\mathrm{ls}} + \delta\vec{T}_{\mathrm{ls}}) \approx 0.005$. Solutions above the threshold of $\mathcal{L}_3(\vec{T}_{\mathrm{ls}}) > 0.8$ favor a shorter T_{P} time and a longer T_{D} time. This indicates that the linear schedules of the local minima are approximately in alignment with the optimal schedule in Figure 3.4 for $p = 3$.

The top-right plot of Figure 3.5 indicates that achieving the optimal parameters for TQA for a specific problem might prove to be more difficult due to great difference in $\mathcal{L}_3(\vec{T}^*_{\mathrm{TQA}})$ across different instances. Assuming that we have optimal values T^*_{D} and T^*_{P} for TQA, a small change to these optimal parameters can result in a significant change in the approximation ratio, with $\mathcal{L}_3(\vec{T}^*_{\mathrm{TQA}}) - \mathcal{L}_3(\vec{T}^*_{\mathrm{TQA}} - \delta\vec{T}_{\mathrm{TQA}}) \approx 0.2$. This value exhibits an order-of-magnitude increase compared to linear schedules. Regarding the overall time

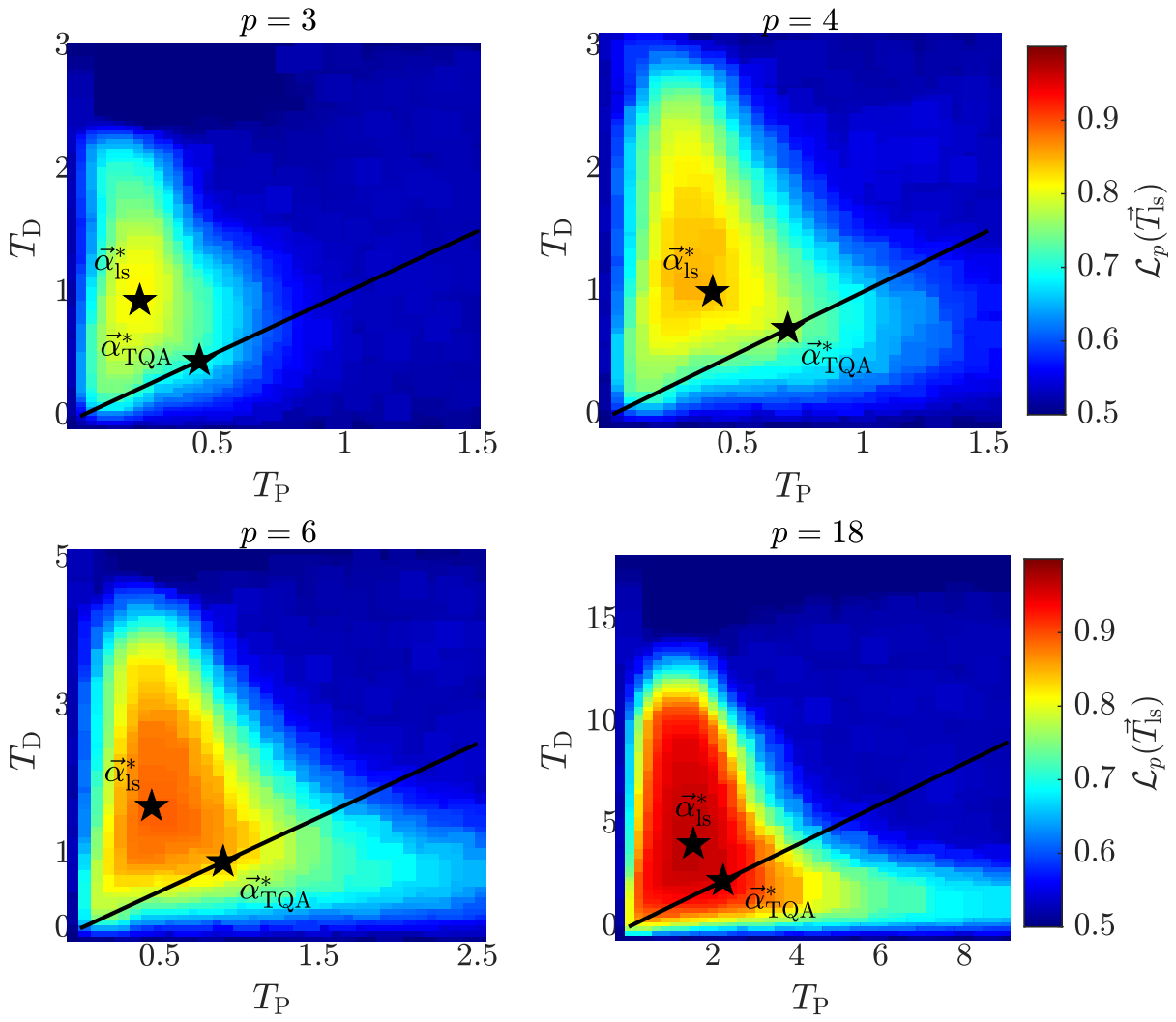


Figure 3.5: Plotting T_D against T_P for $p = 3, 4, 6, 18$. The color indicates $\mathcal{L}_p(\vec{T}_\mathrm{ls})$. Black stars indicate the optimal solution for $\vec{\alpha}^*_\mathrm{ls}$ and $\vec{\alpha}^*_\mathrm{TQA}$. The black line indicates the solution space for TQA. For every time point, we average over 1000 SK instances.

evolution, optimal parameters for TQA result in a total time of $T^{\mathrm{TQA}^*}_\mathrm{tot} = 0.9$, whereas linear schedules necessitate $T^{\mathrm{ls}^*}_\mathrm{tot} = 1.2$. The maximum times T_D and T_P for linear schedules are

$$T^\mathrm{max}_\mathrm{D} = 2\pi p \quad \text{and} \quad T^\mathrm{max}_\mathrm{P} = \frac{\pi p(p+1)}{p-1}. \tag{3.2.12}$$

$T^\mathrm{max}_\mathrm{D}$ corresponds to a constant schedule by setting $\Delta t^\mathrm{D}_1 = \Delta t^\mathrm{D}_2 = \ldots = \beta_p = 2\pi$. For T_P we achieve $T^\mathrm{max}_\mathrm{P}$, due to the constraint that we fixed $\gamma_0 = 0$ and the parameter range to $[0, 2\pi]$.

In the top row of Figure 3.5, we observe an increase of local minima in the parameter space for linear schedules from $p = 3$ to $p = 4$ in the vicinity of the global minimum with respect to $\mathcal{L}_4(\vec{T}_\mathrm{ls})$. Additionally, the region of local minima in the parameter space for linear schedules is close to the global minimum of TQA.

In the lower-left section of Figure 3.5, we can see a noticeable rise in the presence of local minima close to the global minimum when using linear schedules. These additional

local minima are situated at greater distances within the parameter space compared to those observed for $p = 3$ and $p = 4$. The expansion of the local minima in the parameter space of linear schedules leads to an overlap between the region of the global minimum for TQA protocol and the local minima region of linear schedules. The local minima are now below the solid black line compared to the cases when $p = 3$ and $p = 4$. This suggests that linear schedules also with T_{D}-time shorter than T_{P}-time are suitable for higher p.

For $p = 18$ in Figure 3.5, it is evident that the region of local minima in the parameter space expands significantly below and above the solid black line. This expansion is particularly pronounced within the range of short T_{P} and long T_{D} times.

Notably, both methods reach a circuit depth where the global minimum of each method is encircled by neighboring local minima in the parameter space. Essentially, this implies that both approaches have the potential to discover a well-suited schedule for the SK Model that could be applicable across various instances. Particularly for short circuit depths ($p < 10$), we present evidence suggesting that identifying a single schedule suitable for most instances will be more attainable with linear schedules. Furthermore, with increasing p, a variety of linear schedules $\vec{\alpha}_{\mathrm{ls}}$ emerges, demonstrating a mean approximation ratio that closely approaches the global optimum.

3.2.5 System size dependence on the optimal linear schedules

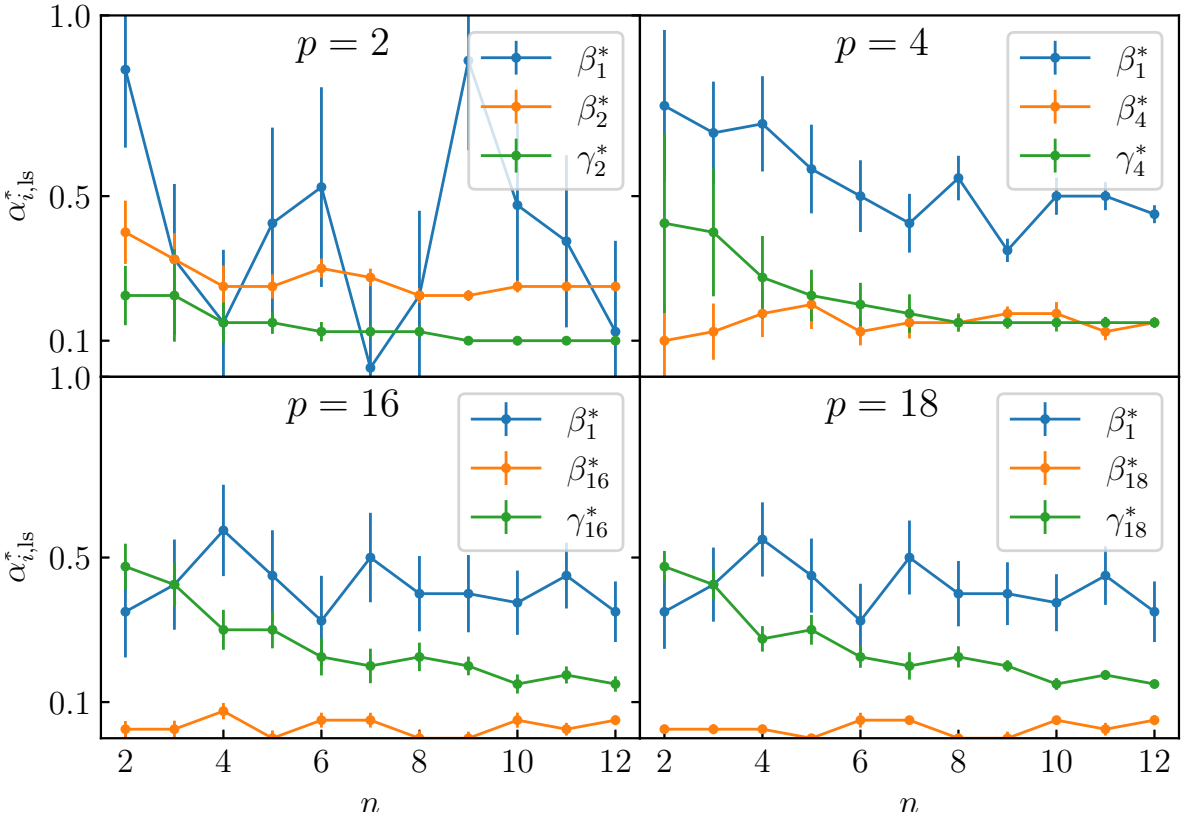


Figure 3.6: Plotting the schedule parameters $\vec{\alpha}_{i,\mathrm{ls}}^*$ against the system size n for the SK model. For linear schedules and fixed γ_1 we have $\vec{\alpha}^* = (\beta_1^*, \beta_p^*, \gamma_p^*)^T$ (blue, orange, green). Depicted are the parameters for $p = 2, 4, 16, 18$. We average over 1000 random SK instances per p and n.

To facilitate further investigations, we present the dependence of the optimal param-

eters $\vec{\alpha}^*_{\text{ls}}$ on the system size in Figure 3.6. For $p = 2$, β^*_1 does not converge for increasing system size. β^*_1 exhibits fluctuations across the entire range of $[0, 2\pi]$. Whereas for γ^*_2, β^*_2 we can note a convergence for $n > 4$ to $\gamma^*_2 = 0.2$, $\beta^*_2 = 0.45$. For $p = 4$ we observe a tendency for β^*_1 to become constant for $n > 7$ with $\beta^*_1 = 0.5$. On top of that, γ^*_4 and β^*_4 also get constant for $n > 7$ with γ^*_4, $\beta^*_4 = 0.15$. This aligns with our observation in Figure 3.4 for $p = 3$ for linear schedules.

In the case of small values of p, the optimal schedule remains independent of the system size as n increases, with $\gamma^*_p = \beta^*_p = 0.15$ converging to the same parameter values. As n grows and p remains small, a non-TQA schedule becomes more suitable. For $p = 16$ and $p = 18$ the parameters β^*_1, β^*_p, γ^*_p are identical.

On top of that, for $p \to \infty, n \to \infty$, the linear schedule $\vec{\alpha}^*_{\text{ls}}$ becomes independent of depth and system size. Since we reach convergence for $\mathcal{L}_p(\vec{\alpha}^*_{\text{ls}})$, $\vec{\alpha}^*_{\text{ls}}$ remains the same from $p = 16$ to $p = 18$. As p increases, the standard deviation of each parameter in $\vec{\alpha}^*_{\text{ls}}$ decreases. Therefore, we discover a universal linear schedule in the thermodynamic limit when p is sufficiently large.

3.3 Applying sigmoid schedule to QAOA

This section will compare the TQA protocol $\vec{\alpha}_{\text{TQA}}$ and the symmetric sigmoid schedules $\vec{\alpha}_{\text{sg}}$. The parameters for the sigmoid schedule are defined by

$$\gamma_p(\vec{\alpha}) = \sum_{p'=1}^{p} \frac{\beta_1}{1 + e^{(p'-p/2)c_{\text{sg}}}}, \quad \beta_p(\vec{\alpha}) = \sum_{p'=1}^{p} \frac{\beta_1}{1 + e^{(p'-p/2)c_{\text{sg}}}} \quad . \tag{3.3.1}$$

The sigmoid function allows us to change the schedule's shape with one parameter $c_{\text{sg}} > 0$. A linear schedule is related to $c_{\text{sg}} \to 0$. The schedule becomes a unit step function for high c_{sg}. In Figure 3.7, we depict the relationship between $1 - \mathcal{L}_p$ and p for both optimized sigmoid and TQA schedules. In the case of the sigmoid schedule, we consider four distinct sigmoid values c_{sg}, for which we optimized the parameter β^*_1 per p. To make it comparable to the previous results, we utilize again the SK model. In a general context, we observe a diminishing trend in $1-\mathcal{L}_p$ as the parameter p increases up to a certain threshold value for all sigmoid schedules depicted. Beyond this specific threshold value of p, $1 - \mathcal{L}_p$ remains approximately constant. Consequently, we cannot identify a suitable value for β^*_1 for each sigmoid schedule, causing $1 - \mathcal{L}_p \to 0$ for $p \to \infty$. However, for $c_{\text{sg}} = 1.0$, we identify a performance improvement over TQA of $\mathcal{L}_4(\vec{\alpha}^*_{\text{TQA}}) - \mathcal{L}_4(\beta^*_1, c_{\text{sg}} = 1.0) = 0.1$ for $p = 4$. This improvement decreases for the best sigmoid schedules as p increases. Specifically, for $p = 5$ and $p = 10$, the improvement over TQA is $\mathcal{L}_5(\vec{\alpha}^*_{\text{TQA}}) - \mathcal{L}_5(\beta^*_1, c_{\text{sg}} = 1.0) = 0.07$ and $\mathcal{L}_{10}(\vec{\alpha}^*_{\text{TQA}}) - \mathcal{L}_{10}(\beta^*_1, c_{\text{sg}} = 1.0) = 0.05$, respectively.

Figure 3.8 shows the normalized shapes of the parameter schedules corresponding to the sigmoid values in Figure 3.7. For small p, the preferred shape is closer to a step function. As the parameter p increases, the curve progressively takes on a form that increasingly resembles the TQA schedule. This explains the decrease in the improvement in $\mathcal{L}_p$ between sigmoid and TQA for increasing p. For large p, the sigmoid step function shapes are no longer advantageous over TQA. Furthermore, the sigmoid shapes become more damped as β^*_1 decreases with increasing p. In conclusion, we recommend using a sigmoid schedule for low p rather than a TQA schedule.

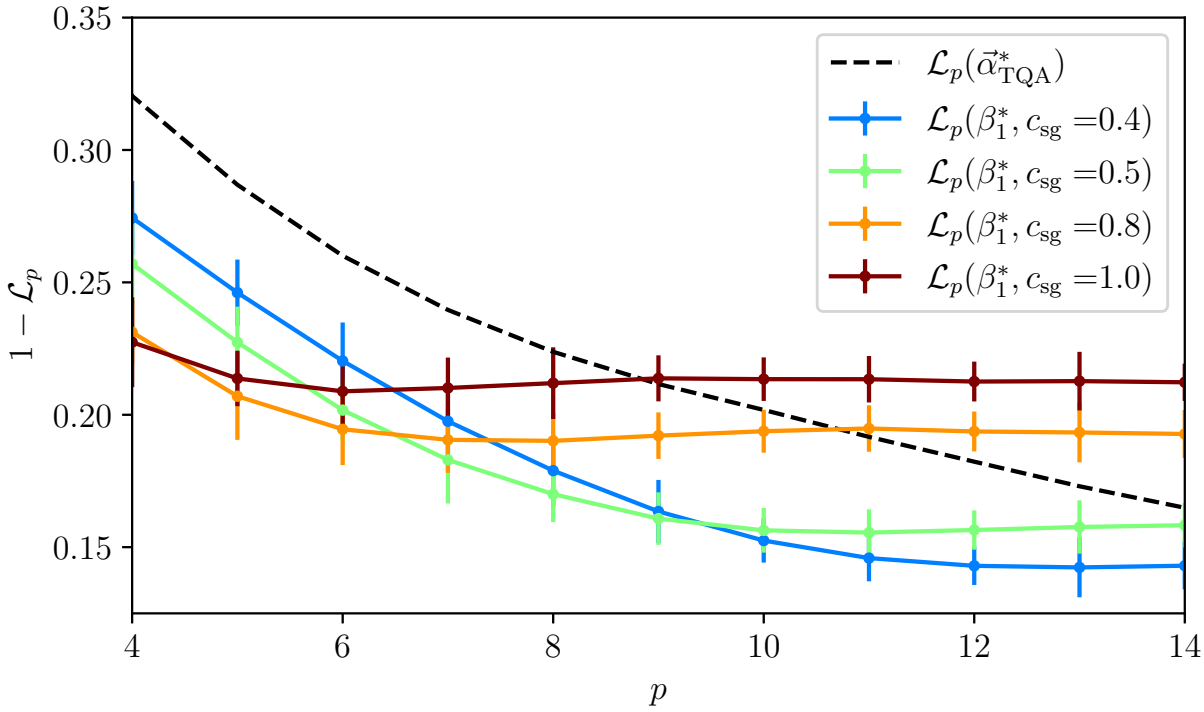


Figure 3.7: Plotting mean approximation ratio of the optimized schedule parameters $1-\mathcal{L}_p(\vec{\alpha})$ for QAOA against p, for 4 different sigmoid schedules. Plotted are $1-\mathcal{L}_p$ for sigmoid values $c_{\mathrm{sg}} = 0.4, 0.5, 0.8, 1.0$, which performs best on QAOA for $p = 4, 6, 8, 10$, respectively. Figure 3.8 shows the shapes of these sigmoid functions. The black dashed line indicates TQA performance. For every schedule, we average over 1000 random SK instances.

3.4 Linear schedules versus Fourier initialization

This section will examine the linear schedules and the Fourier method developed by Zhou et al. [45] regarding performance and optimization efforts. This section presents a comparison between two approaches that seek to determine the optimal angles $\vec{\gamma}^*$ and $\vec{\beta}^*$. We assess both approaches based on their impact on the $\mathcal{M}_p$ value of QAOA, as well as the level of optimization efforts required. The term "optimization efforts" refers to the number of times the optimization process has to be invoked.

The first approach involves employing linear schedules, as mentioned earlier, which are based on the annealing schedule technique. The second approach, proposed by Zhou et al. [45], utilizes the Fourier method, which involves reusing the parameters p for QAOA parameters $p+1$. Thus, requiring only $O[\mathrm{poly}(p)]$ optimizer calls, compared to the $2^{O(p)}$ required for the random initialization strategy. The method of Zhou et al. is based on spectral analysis and involves transforming the $(\vec{\gamma}, \vec{\beta})$ according to

$$\gamma_i = \sum_{k=1}^{q} u_k \sin\left[\left(k-\frac{1}{2}\right)\left(i-\frac{1}{2}\right)\frac{\pi}{2}\right],\ \beta_i = \sum_{k=1}^{q} v_k \cos\left[\left(k-\frac{1}{2}\right)\left(i-\frac{1}{2}\right)\frac{\pi}{2}\right]. \tag{3.4.1}$$

The parameters u_k and v_k are chosen freely and can be thought of as the k^{th} amplitude of either γ_i or β_i. $\vec{u} = (u_1, \ldots, u_q)$ and $\vec{v} = (v_1, \ldots, v_q)$ represent the sets of parameters

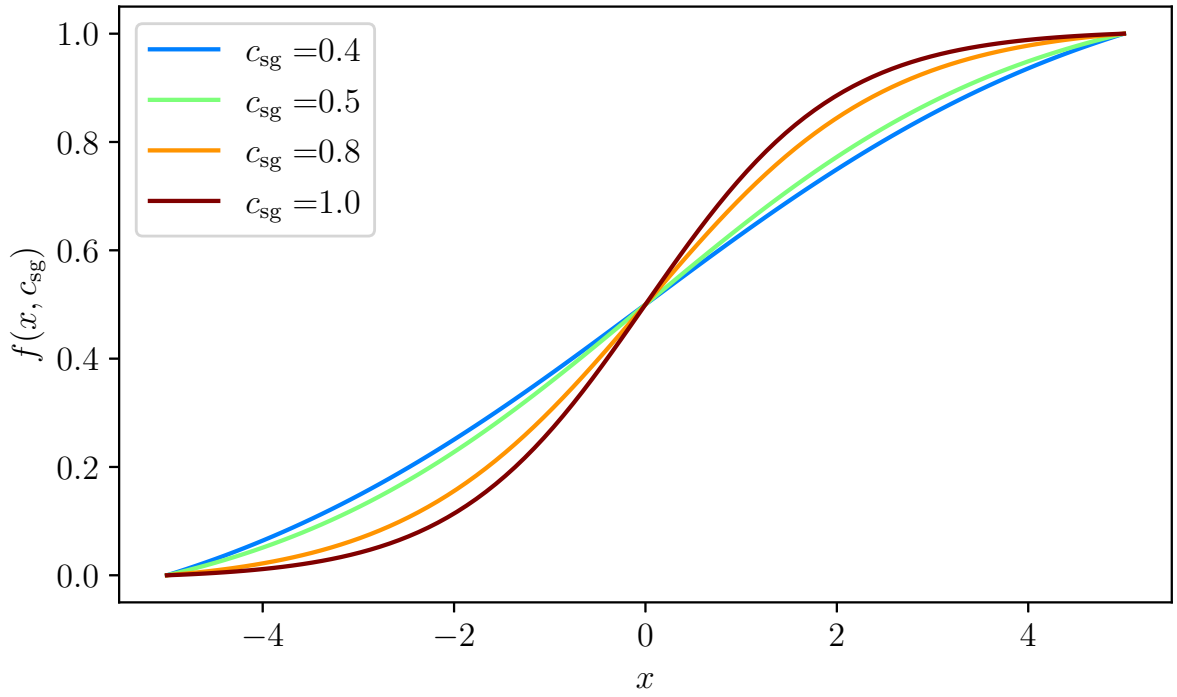


Figure 3.8: Sigmoid shapes: Plotting schedule parameters $f(x, c_{sg})$ against x. x is reflecting the p values for γ_i. Depicted are the sigmoid schedules, which show the best QAOA performance for $p = 4, 6, 8, 10$ from Figure 3.7. These sigmoid plots are normalized to $p = 10$. Because we have a symmetric sigmoid schedule, the β_i values are γ_i mirrored at the y-axis.

corresponding to u_k and v_k. The value of q can be selected within the range $1 \leqslant q \leqslant p$. Starting with $p = 1$, we then insert the parameters u_1 and v_1 to obtain u_2 and v_2 for $p = 2$. As we expect a smooth transition for γ_i, β_i from p to $p + 1$, only the lower frequencies of $(\vec{u}, \vec{v})$ will be relevant. Therefore, trying lower values of q first is advisable since higher values will be cut off. The different Fourier methods, represented as Fourier$[q, R]$, depend on q and another parameter R. If q is set to p, the case is denoted by Fourier$[\infty, R]$. R describes the number of perturbations we apply to the initial point. For instance, if $R = 10$, we will set 10 initial points for p. We will optimize each and select the best option. To compare linear schedules against the Fourier method, we will investigate the same problem as in [45]. We are dealing with a weighted Max-cut on a three-regular graph

$$\hat{H} = \sum_{<i,j>} \frac{w_{ij}}{2}(\mathbb{1} - Z_i Z_j). \tag{3.4.2}$$

The weighting function w_{ij} assigns a randomly selected value from a Gaussian distribution to each edge i, j. The Gaussian distribution has zero mean and unit variance. Like in the original paper, we set $R = 10$ and $q = p$.

In Figure 3.9, we compare the Fourier approach, linear schedules, and random initialization methods. The red curve, which indicates the mean approximation ratio for linear schedules $\mathcal{M}_p(\vec{\alpha}^*_{\text{ls}})$, achieves comparable mean approximation ratios as $\mathcal{M}_p(\text{Fourier}[\infty, R = 10])$ (blue curve) method for $p \leqslant 10$. Both mean approximation ratios are on the same scale and reach a maximal difference of $\mathcal{M}_p(\vec{\alpha}^*_{\text{ls}}) - \mathcal{M}_p(\text{Fourier}[\infty, R = 10]) \approx 0.01$ for

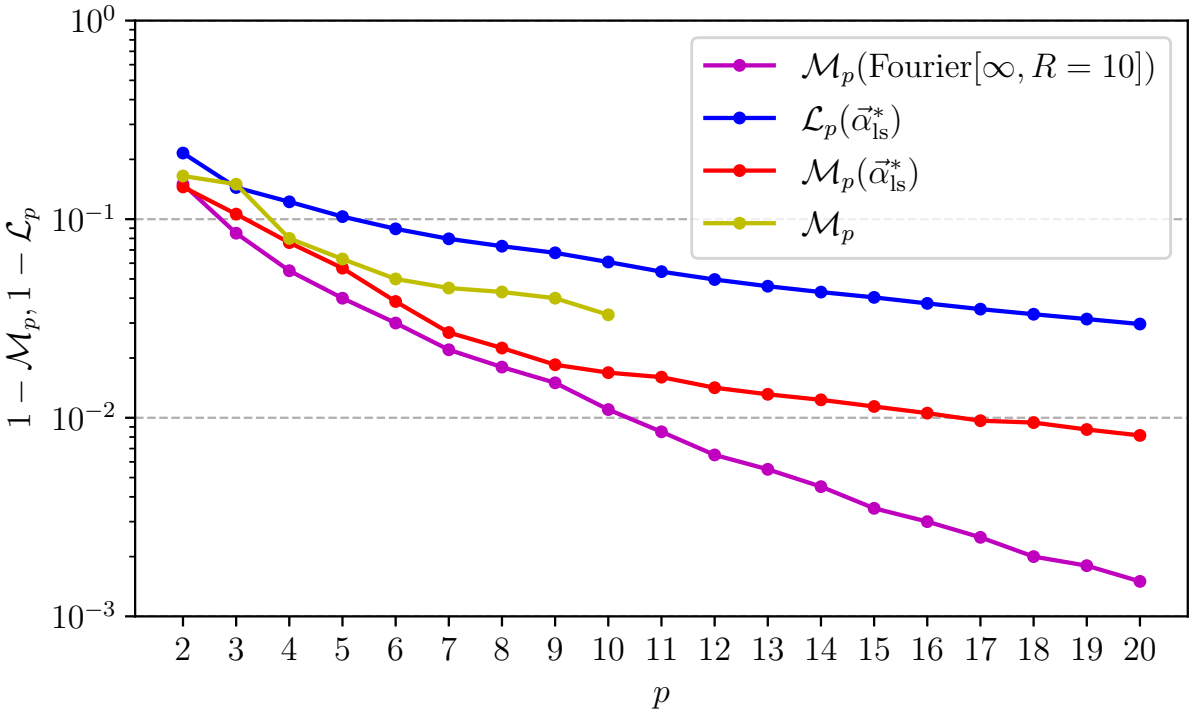


Figure 3.9: Plotting $1-\mathcal{M}_p$ against p. The magenta-colored curve represents the Fourier method with $q=p$. The blue curve represents the initial values for linear schedules, and the red curve represents the QAOA execution with linear schedule initialization. Yellow indicates the results for random initial starting points for $p \leqslant 10$. For $p \leqslant 10$ and random initialization, we conducted more than 10000 optimizer executions. For higher values of p, it is not possible to use the random initialization technique, therefore. We averaged the results over 1000 instances.

$p \leqslant 10$. However, the infidelity $1-\mathcal{M}_p$ for linear schedules is always greater. For $p>10$, $\mathcal{M}_p(\vec{\alpha}^*_{\text{ls}})$ exhibits only a marginal improvement from p to $p+1$ compared to $p<10$. This suggests that the initialization method may not effectively leverage the additional dimensionality brought by the increase in p. For $p=20$, the mean approximation ratio attains an infidelity value of $1-\mathcal{M}_{20}(\vec{\alpha}^*_{\text{ls}})=0.01$. In contrast, $\mathcal{M}_{20}(\text{Fourier}[\infty, R=10])$ achieves infidelity of approximately $1-\mathcal{M}_{20}(\text{Fourier}[\infty, R=10]) \approx 0.001$ for $p=20$. To achieve this, more optimization runs are needed for the Fourier method. The Fourier method depends on the previous p value, which requires more optimization effort for high p, such as $p=20$. Particularly in this case, we have 10 optimizer executions per p, resulting in 190 optimizer executions for $p=20$. In contrast, linear schedules only require 2 optimizer executions for $p=20$. The random initialization method is only effective for $p \leqslant 6$. To achieve results close to Fourier or linear schedules with the random initialization method, 10000 optimization executions are necessary for $4<p<6$.

3.5 Conclusion

In this chapter, we showed that linear schedules outperform TQA in two categories for $p \leqslant 10$. First, when considering a specific parameter configuration p, linear schedules yield higher mean approximation ratios for the optimized parameters $\mathcal{L}_p(\vec{\alpha}^*)$ compared

to TQA, as expressed by $\mathcal{L}_p(\vec{\alpha}^*_{\text{ls}}) > \mathcal{L}_p(\vec{\alpha}^*_{\text{TQA}})$. However, the difference in $\mathcal{L}_p(\vec{\alpha}^*)$ between $\vec{\alpha}^*_{\text{TQA}}$ and $\vec{\alpha}^*_{\text{ls}}$ diminishes for $p > 10$. Second, linear schedules outperform TQA in terms of the mean approximation ratio $\mathcal{M}_p(\vec{\alpha}^*)$, as indicated by $\mathcal{M}_p(\vec{\alpha}^*_{\text{ls}}) > \mathcal{M}_p(\vec{\alpha}^*_{\text{TQA}})$. The results demonstrate that the optimal parameters obtained from linear schedules guide QAOA to a region in the parameter space where more nearby local minima align closely with the global minimum of the QAOA objective, compared to TQA. This assertion is substantiated by the average separation between $\vec{\gamma}(\vec{\alpha}^*)$ and $\vec{\beta}(\vec{\alpha}^*)$ compared to $\vec{\gamma}^*(\vec{\alpha}^*)$ and $\vec{\beta}^*(\vec{\alpha}^*)$. Notably, the average separation for TQA is larger, especially for short circuit depth.

Optimizing linear schedules requires three parameters, while TQA optimization only requires one. However, even though the parameter space is larger, we need only one optimization run for the subroutine to get a greater $\mathcal{L}_p$ for linear schedules than for TQA. Therefore, we observed no difference in optimization efforts between linear schedules and TQA in low-dimension parameter space.

Moreover, for low p, a sigmoid schedule offers an advantage over TQA in the mean approximation ratio $\mathcal{L}_p$. Thus, linear schedules outperform TQA with another one-parameter method. In general, the difference between linear schedules and TQA becomes insignificant as p increases. This conclusion is supported by the time comparison plot in Figure 3.5.

Chapter 4

Coherent and non-unitary errors in ZZ-generated gates

The foundation of every quantum computer is built upon two-qubit gates, which are more prone to errors than one-qubit gates. For the NISQ algorithm, the ZZ-generated gates ($R_{ZZ}(\gamma)$) play a crucial role. The gate's accuracy and impact on the algorithm's performance rely on the hardware capabilities for two-qubit gates, specifically how the decomposition into the native gate set affects the circuit depth. This chapter focuses on decomposing $R_{ZZ}(\gamma)$ into CZ, CP, and iSWAP gates, considering both incoherent and coherent errors in the gate fidelity. We show that CZ and CP decomposition achieve comparable gate fidelities for low coherent and incoherent noise.

This chapter is structured as follows: In Section 4.2, we explain the calculation of gate fidelity, drawing upon the research conducted by Cabrera et al. [101]. Furthermore, we emphasize the key findings of their work. Section 4.3 discusses incorporating coherent errors in CZ and CP circuits, providing analytical and numerical results for the gate fidelities. Next, in Section 4.4, we analyze the gate decompositions for CP and CZ gates under depolarizing error. Finally, in Section 4.5, we compare the gate fidelities of CZ and CP decompositions and discuss which gate might be more suitable for a superconducting platform considering specific noise conditions. Since iSWAP belongs to the same equivalence class as CZ, we anticipate that CZ and iSWAP yield comparable gate fidelities. Thus, we only present the less known decomposition of $R_{ZZ}(\gamma)$ into iSWAP in Section 4.6.

4.1 Depolarizing channel

In the introduction chapter, we investigated the general concept of quantum channels. This section will clarify the specific depolarizing channel, which models different standard errors within a single model. The general form of the d-dimensional depolarizing error is given by

$$\mathcal{E}(\rho) = \frac{\mathbb{1}}{2^d} p + (1-p)\rho. \tag{4.1.1}$$

Here, p denotes the probability that the density matrix becomes a completely mixed state $\mathbb{1}/2^d$ after the channel is applied, while $1-p$ represents the probability that the initial density matrix ρ remains unchanged. The depolarizing channel can be decomposed into bit-flip, phase-flip, and bit-phase-flip errors, which occur in all permutations with equal probability. Table 4.1 provides an overview of the Kraus operators for the error under consideration. A bit-flip and phase-flip can be achieved by applying X and Z operators,

M	$\hat{M}\|\psi\rangle$	Quantum Operation
Z	$\hat{M}\|0\rangle = -\|0\rangle$	Phase-flip
X	$\hat{M}\|0\rangle = \|1\rangle$	Bit-flip
Y	$\hat{M}\|0\rangle = -\|1\rangle$	Bit- and Phase-flip

Table 4.1: Quantum error operators on the example state $|0\rangle$

respectively. The combination of both errors applied simultaneously leads to $Y = iXZ$. The three types of errors result in different contractions of the Bloch sphere (see Figure 4.1). The contraction of the Bloch sphere differs for each type of error: the bit-flip channel contracts along the Y-Z plane, the phase-flip contracts along the X-Y plane, and the phase-bit-flip channel contracts along the X-Z plane. The depolarizing channel uniformly contracts the Bloch sphere in each direction. The contraction effect depends on the probability p. Regarding two-qubit gates, we have two options for the depolarizing channel: $d = 2$ and $d = 1$. The latter corresponds to applying two single quantum channels to each qubit. First, we will consider the Kraus operator sum representation of Equation (4.1.1) for $d = 2$ the density matrix of a mixed state can be decomposed using the operators $\hat{L} \in [X, Y, Z, \mathbb{1}]$ and $\hat{M} \in [X, Y, Z]$, resulting in $\mathbb{1}/2^2 = (p/16)\rho + (p/16)\sum_{i=1}^{4}\sum_{j=1}^{3}\hat{M}_j\hat{L}_i\rho\hat{L}_i^\dagger\hat{M}_j^\dagger$. By substituting $p \to (16/15)p$, this decomposition yields the operator sum representation of the $d = 2$ depolarizing channel

$$\mathcal{E}(\rho) = (1-p)\rho + \frac{p}{15}\sum_{i=1}^{4}\sum_{j=1}^{3}\hat{M}_j\hat{L}_i\rho\hat{L}_i^\dagger\hat{M}_j^\dagger. \tag{4.1.2}$$

Another possibility is to describe the depolarizing error using two one-qubit quantum channels, resulting in

$$\begin{aligned}\mathcal{E}(\rho) = (1-p)\rho + (1-p)\frac{p}{3}\textstyle\sum_i \hat{M}_i\rho\hat{M}_i^\dagger + (1-p)\frac{p}{3}\sum_i \hat{M}_i\rho\hat{M}_i^\dagger \\ + \frac{p^2}{9}\textstyle\sum_{i,j}\hat{M}_j\hat{M}_i\rho\hat{M}_i^\dagger\hat{M}_j^\dagger.\end{aligned} \tag{4.1.3}$$

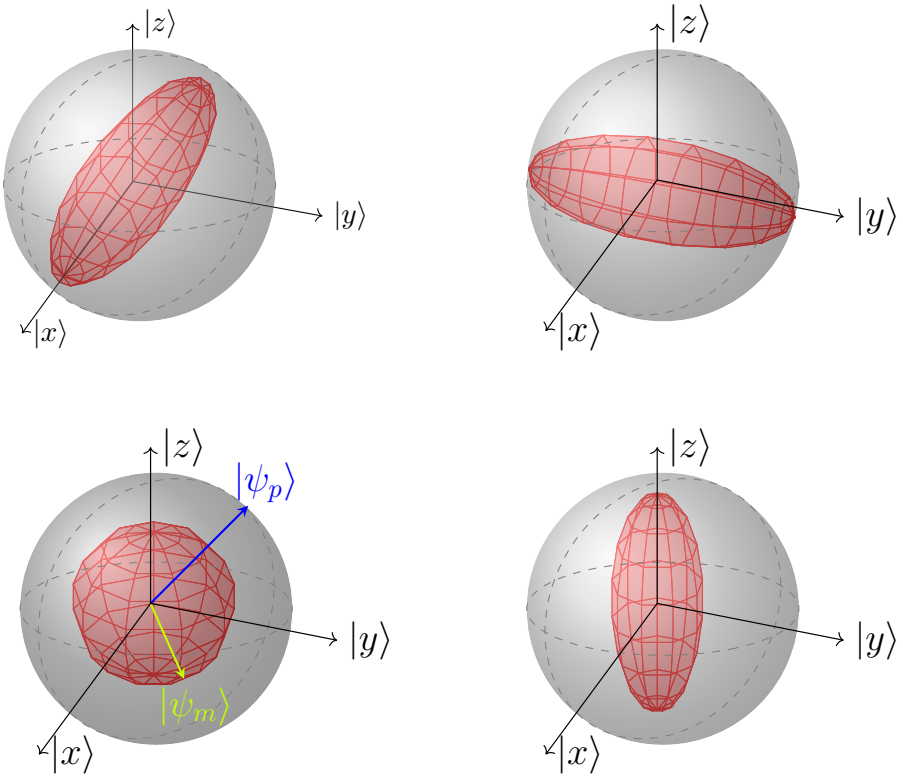


Figure 4.1: Sketching incoherent errors: In gray, we plot the Bloch sphere for single qubits, which is the error-free state. The blue error indicates that all the pure states $|\psi_p\rangle$ live on the Bloch sphere. The mixed states live on the red sphere like $|\psi_m\rangle$. The red spheres illustrate the contraction of the Bloch sphere after a related error. Top left: bit-flip error, Top right: phase-flip and bit-flip error, bottom right: phase-flip error, bottom left: depolarizing error.

The equation displayed above contains two probability weights. One corresponds to the likelihood of encountering two errors on both qubits, which is $p^2/9$. In contrast, the other probability weight corresponds to the occurrence of an error on a single qubit, which is $((p - p^2)/3)$. For a two-qubit gate with $d = 2$, the error occurrence grows linearly with p, while for $d = 1$, we have a polynomial dependence of p for error occurrence. For the case of a two-qubit gate, the $d = 2$ channel is more accurate. From a general hardware perspective, there is no reason to unweight different error types for a two-qubit gate. We will not investigate correlated two-qubit quantum channels where only the same type of error on each qubit is allowed, for instance, $X_1X_2\rho X_1X_2$.

4.2 Gate fidelity

Gate fidelity is a vital measure tool to classify the quality of the gates. To calculate the gate fidelity, we first examine the state fidelity, defined by Richard Josza [102], which is the square root of the original Uhlmann equation [103]

$$F(\rho, \rho') = \left(\mathrm{Tr}\left(\sqrt{\sqrt{\rho}\rho'\sqrt{\rho}}\right)\right)^2, \tag{4.2.1}$$

with ρ and ρ' being general density matrices. Let ρ and ρ' be representatives for pure states $|\psi\rangle$ and $|\psi'\rangle$. In this case, Equation (4.2.1) reduces to the square overlap $|\langle\psi|\psi'\rangle|^2$ because $\rho = \rho^{1/2}$ for pure states.

For gate fidelities, we always want to compare a gate operation $\hat{U}$ under error-free and erroneous conditions. The unitary gate operation takes any pure state to another pure state $\rho = \hat{U}\rho_0\hat{U}^\dagger$. The erroneous gate operation $\rho' = \mathcal{E}(\rho_0)$ can be non-unitary, and thus Equation (4.2.1) reduces to $F(\rho, \rho') = \mathrm{Tr}(\rho\rho')$. The gate fidelity is the average of the integral over all possible pure states as input states. For the one-qubit gate, this implies integrating over the Bloch sphere, and for the two-qubit case, integrating over two times the Bloch sphere

$$\mathcal{F} = \langle F \rangle = \frac{1}{16\pi^2} \int \mathrm{Tr}(\rho\rho')d\Omega^2, \tag{4.2.2}$$

with $d\Omega^2$ being the volume integral. The Pauli matrices and the identity matrix span the whole space of density matrices over $\mathbb{C}^4 \times \mathbb{C}^4$. Thus ρ can be decomposed into $\rho = (\mathbb{1} + \sum_{i=1}^{15} q_j S_j)$. Where S_j is an element of $S_j \in [X, Y, Z, \mathbb{1}] \otimes [X, Y, Z]$. Next, we insert the Pauli expansion into Equation (4.2.2) and integrate over

$$\mathcal{F} = \frac{1}{16}\left[4 + \frac{1}{5}\sum_{j=1}^{15} f_j\right] = \frac{1}{16}\left[4 + \frac{1}{5}\sum_{i,j}\mathrm{Tr}[\hat{U}S_j\hat{U}^\dagger\mathcal{E}(S_j)]\right]. \tag{4.2.3}$$

Only the diagonal elements are non-zero for the previous equation. Bowdrey et al. [104] showed this for the one-qubit case. Cabrera examined [105, 101] the n-qubit case.

Zoller et al. [106, 107] demonstrated that gate fidelity can also be assessed using 16 initial pure states and using Equation (4.2.1). Frequently, the 16 initial states

$$|\Psi_a\rangle\,|\Psi_b\rangle\,, \tag{4.2.4}$$

with a,b=1,..,4 and

$$\begin{aligned} \Psi_1 &= |0\rangle\,, \quad \Psi_3 = (1/\sqrt{2})(|0\rangle + |1\rangle), \\ \Psi_2 &= |1\rangle\,, \quad \Psi_4 = (1/\sqrt{2})(|0\rangle + i\,|1\rangle), \end{aligned} \tag{4.2.5}$$

are used [106, 107, 108, 109]. We vectorize the two indices of $|\Psi_a\rangle\,|\Psi_b\rangle$ to achieve the the 16 state fidelities f_j. The value of the gate fidelity is always between 0 and 1 because of the normalized initial states.

4.3 Decomposing $R_{ZZ}(\gamma)$ gate into CP and CZ gates with coherent noise

In this section, we examine the decomposition of $R_{ZZ}(\gamma)$ into CZ and CP gates. We demonstrate the incorporation of coherent error and present the results for the gate fidelity in the presence of coherent noise.

4.3.1 CZ and CP circuits

The $R_{ZZ}(\gamma)$ gate plays an essential role in implementing quantum algorithms. Typical combinatorial optimization problems like travel salesman will be mapped onto an Ising

chain and incorporated in a QAOA ansatz or an annealing process.

Depending on the hardware, the $R_{ZZ}(\gamma)$ gate will be decomposed into the native gate set. For example, trapped-ion-based quantum computers have the $R_{ZZ}(\gamma)$ gate or other gates within the same equivalence class, such as $R_{XX}(\gamma)$ [110, 111], natively available. This means that $R_{ZZ}(\gamma)$ gate can be created efficiently by applying single-qubit rotations to their native two-qubit gate, as shown by $R_{ZZ}(\gamma) = \mathrm{H}_1\mathrm{H}_2 R_{XX}(\gamma)\mathrm{H}_1\mathrm{H}_2$. In this context, we are investigating the decomposition of $R_{ZZ}(\gamma)$ into the fixed controlled-Z gate (CZ) and the parametric controlled-phase gate (CP). However, since the controlled-Z gate is not in the same equivalence class of non-local two-qubit gates ($SU(4) \setminus (SU(2) \otimes SU(2))$) as $R_{ZZ}(\gamma)$, at least two two-qubit gates are required for the decomposition [25, 112, 113]. The group of Andreas Wallraff has been investigating the hardware implementation of CP and CZ gates [114]. We decompose $R_{ZZ}(\gamma)$ into a controlled-phase gate (CP) [114] according to

$$R_{ZZ}(\gamma) = \quad i \;\; \boxed{R_Z(\gamma)} \;\; \bullet \qquad j \;\; \boxed{R_Z(\gamma)} \;\; \boxed{\begin{pmatrix} 1 & 0 \\ 0 & e^{(-2i\gamma)} \end{pmatrix}}, \quad \text{with CP}(\gamma) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\gamma} \end{pmatrix} \tag{4.3.1}$$

being the CP gate. The $R_{ZZ}(\gamma)$ gate and Z-gate are defined as

$$R_{ZZ}(\gamma) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{i\gamma} & 0 & 0 \\ 0 & 0 & e^{i\gamma} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad R_Z(\gamma) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\gamma} \end{pmatrix}. \tag{4.3.2}$$

CP and $R_{ZZ}(\gamma)$ gate belong to the same equivalence class, meaning we only need single-qubit rotations $R_Z(\gamma)$ to construct the decomposition. The decomposition of $R_{ZZ}(\gamma)$ into controlled-Z gate CZ is according to

$$R_{ZZ}(\gamma) = \quad i \;\; \bullet \;\; \mathrm{H} \;\; \bullet \;\; \mathrm{H} \;\; \boxed{R_Z(\gamma)} \;\; \mathrm{H} \;\; \bullet \;\; \mathrm{H} \qquad j \;\; \mathrm{H} \;\; \boxed{\mathrm{X}} \;\; \mathrm{H} \;\; \boxed{\mathrm{X}} \;\; \mathrm{H}, \tag{4.3.3}$$

$$\text{with} \quad \mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}.$$

To decompose CNOT into CZ, we require five single-qubit gates - four Hadamard gates for basis transformation and one $R_Z(\gamma)$ for the rotation. Unlike the CP decomposition, which involves two-qubit and single-qubit gates containing γ, the CZ decomposition contains γ only in the $R_Z(\gamma)$ gate. This could provide an advantage on certain hardware, as single-qubit gates are generally less error-prone than two-qubit gates, and a fixed two-qubit gate like CZ only needs to be calibrated once. The error rate for single-qubit gates is negligible, with a gate fidelity of 10^{-5} [115]. However, the error rate for two-qubit gates

is higher than for single-qubit gates. In the supremacy experiment [19], the two-qubit gate infidelity was approximately 0.36%, which makes it impossible to run an algorithm with quantum error correction code. An error rate below 0.1% is necessary to build a fault-tolerant quantum computer, as demonstrated by various threshold theorems [21, 22, 23].

4.3.2 CZ gate fidelity under coherent noise

To begin, we explore the CZ decomposition and the impact of coherent errors on gate fidelity. For this purpose, we consider the circuit represented by (4.3.3) and include the CP gate as the erroneous gate operation

$$\hat{U}_{\mathrm{cz}}^{\mathrm{co}} = \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{CP}(\theta) \cdot \mathrm{H}_2 \cdot R_{Z_2}(\gamma) \cdot \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{CP}(\zeta) \cdot \mathrm{H}_2. \tag{4.3.4}$$

ζ and θ are coherent over-rotations applied after the two CZ gates. The error-free CZ decomposition ($\mathrm{CP}(\theta, \zeta = 0)$) reads

$$\hat{U}_{\mathrm{cz}} = \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{H}_2 \cdot R_{Z_2}(\gamma) \cdot \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{H}_2. \tag{4.3.5}$$

The gate fidelity $\mathcal{F}_{\mathrm{cz}}^{\mathrm{co}}$, with the superscript for the error type and the subscript for the decomposition, is calculated according to Zoller et al. [106, 107] and Equation (4.2.1)

$$\mathcal{F}_{\mathrm{cz}}^{\mathrm{co}} = \frac{1}{16} \sum_{j=1}^{16} |\, \langle \psi_j | (\hat{U}_{\mathrm{cz}}^{\mathrm{co}})^\dagger \cdot \hat{U}_{\mathrm{cz}} | \psi_j \rangle \,|^2. \tag{4.3.6}$$

Because we are still in the coherent Schrödinger picture, Equation (4.2.1) reduces to the square overlap. The decomposition of $R_{ZZ}(\gamma)$ into CZ gates is not diagonal, which means that the quantum operators $\mathrm{CP}(\theta)$ and $\mathrm{CP}(\zeta)$ cannot commute through due to the presence of Hadamard gates. Therefore, the state fidelity equation is

$$\begin{aligned} f_{\mathrm{cz},j}^{\mathrm{co}} &= |\, \langle \psi_j | \, \mathrm{H}_2^\dagger \cdot \mathrm{CZ}^\dagger \cdot \mathrm{CP}^\dagger(\theta) \cdot \mathrm{H}_2^\dagger \cdot R_{Z_2}^\dagger(\gamma) \cdot \mathrm{H}_2^\dagger \cdot \mathrm{CZ}^\dagger \cdot \mathrm{CP}^\dagger(\zeta) \cdot \mathrm{H}_2^\dagger \cdot \mathrm{H}_2 \cdot \mathrm{CZ} \\ &\quad \times \mathrm{H}_2 \cdot R_{Z_2}(\gamma) \cdot \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{H}_2 \, | \psi_j \rangle \,|^2 \\ &= |\langle \psi_j | \mathrm{H}_2 \cdot \mathrm{CZ} \cdot \mathrm{CP}(\theta) \cdot R_{X_2}(\gamma) \cdot \mathrm{CZ} \cdot \mathrm{CP}(\theta) \cdot \mathrm{CZ} \cdot R_{X_2}(\gamma) \cdot \mathrm{CZ} \cdot \mathrm{H}_2 | \psi_j \rangle|^2. \end{aligned} \tag{4.3.7}$$

Now, we can summarize the results for all 16 state fidelities. The error only influences the third and fourth entries of the state vector $|\psi_j\rangle$, which results in no dependence on the coherent error for $\psi_3, \psi_6, \psi_{10}, \psi_{16}$, leading to a state fidelity of $f_{3,6,10,16} = 1$. For further investigation, we presume the coherent error to be small and impose a standard deviation of the error of $\sigma(\theta) \leqslant 0.06\pi$, equivalent to an error rate of 3%. This is a realistic error rate for superconducting hardware. We expand each state fidelity to the second order to obtain a closed equation for gate fidelity in the CZ decomposition that does not violate space constraints. We will disregard any terms that are of order three $\mathcal{O}(h^3)$ or higher. The variable h^3 encompasses all terms of order three, namely ζ^3, θ^3, $\zeta^2\theta^1$, and $\zeta^1\theta^2$. To determine the general form of h^m without coefficients, we can apply the binomial equation

$$h^m = \sum_{s=1}^{m} \zeta^s \theta^{m-s}. \tag{4.3.8}$$

9 of 16 states exhibit a quadratic dependence on the errors and demonstrate a strong relationship between the error and the rotation angle

$$f_{\mathrm{cz},j} = 1 - h^2(\Xi + \varpi\sin(\gamma) + \Upsilon\cos(\gamma)) + \mathcal{O}(h^3). \tag{4.3.9}$$

Not every element of h^2 contributes to a given state fidelity equation. The values of $\Xi, \varpi, \Upsilon \in \mathbb{R}$ depend on the considered state. We distinguish 4 different state fidelities among the 9

$$\begin{aligned}
f^{\mathrm{co}}_{\mathrm{cz},5}, f^{\mathrm{co}}_{\mathrm{cz},7}, f^{\mathrm{co}}_{\mathrm{cz},9}, f^{\mathrm{co}}_{\mathrm{cz},11} &= 1 - 0.19\,\theta^2 - 0.25\,\theta\zeta\cos(\gamma) - 0.125\,\theta\zeta - 0.19\,\zeta^2 + \mathcal{O}(h^3),\\
f^{\mathrm{co}}_{\mathrm{cz},8} &= 1 - 0.25\,\theta^2 - 0.5\,\theta\zeta\cos(\gamma) + \zeta^2\,(-0.125\cos(2\gamma) - 0.125) + \mathcal{O}(h^3),\\
f^{\mathrm{co}}_{\mathrm{cz},12}, f^{\mathrm{co}}_{\mathrm{cz},14} &= 1 - 0.12\,\theta^2 + \zeta^2\,(-0.13\sin(\gamma) - 0.03\cos(2\gamma) - 0.16) + \mathcal{O}(h^3),\\
&\quad + \zeta\,(-0.125\,\theta\sin(\gamma) - 0.25\,\theta\cos(\gamma) - 0.125\,\theta) + \mathcal{O}(h^3)\\
f^{\mathrm{co}}_{\mathrm{cz},1}, f^{\mathrm{co}}_{\mathrm{cz},2} &= 1 - 0.25\,\theta^2 - 0.5\,\theta\zeta\cos(\gamma) - 0.25\,\zeta^2 + \mathcal{O}(h^3),
\end{aligned} \tag{4.3.10}$$

In addition, the last three remaining state fidelities show a quadratic relationship between θ and the rotation angle

$$\begin{aligned}
f^{\mathrm{co}}_{\mathrm{cz},4} &= 1 + \zeta^2\,(0.125\cos(2\gamma) - 0.125) + \mathcal{O}(h^3)\\
f^{\mathrm{co}}_{\mathrm{cz},13}, f^{\mathrm{co}}_{\mathrm{cz},15} &= 1 + (-0.16 + \zeta^2\,(0.125\cos(\gamma) + 0.03\cos(2\gamma)) + \mathcal{O}(h^3).
\end{aligned} \tag{4.3.11}$$

The analytical equation for gate fidelity in CZ decomposition up to the second order, obtained by summing up all state fidelities, is

$$\begin{aligned}
\mathcal{F}^{\mathrm{co}}_{\mathrm{cz}} &= \frac{1}{16}\sum_{j=1}^{16} |\,\langle\psi_j|(\hat{U}^{\mathrm{co}}_{\mathrm{cz}})^\dagger\hat{U}_{\mathrm{cz}}|\psi_j\rangle\,|^2\\
&= 1 - 0.12\,\theta^2 - 0.02\,\theta\zeta\sin(\gamma) - 0.19\,\theta\zeta\cos(\gamma) - 0.05\,\theta\zeta\\
&\quad - 0.02\,\zeta^2\sin(\gamma) + 0.02\,\zeta^2\cos(\gamma) - 0.13\,\zeta^2 + \mathcal{O}(h^3).
\end{aligned} \tag{4.3.12}$$

The numerical results are displayed in Figure 4.2, which shows the gate fidelity $\mathcal{F}^{\mathrm{co}}_{\mathrm{cz}}$ as a function of the rotation angle γ. This fidelity exhibits a quadratic dependence on the parameters θ, ζ and a damped cosine function dependence on γ. The most significant infidelity, we would expect at $\cos(\gamma = \pi/2)$. However, due to the small contribution of $\sin(\gamma)$, the minimum fidelity $\min_\gamma \mathcal{F}^{\mathrm{co}}_{\mathrm{cz}}(\gamma, \theta = \text{const.})$ is shifted. The minimum gate fidelity is achieved at $\gamma = 0.72\pi$ and $\sigma(\zeta), \sigma(\theta) = 0.062\pi$. Since the gate fidelity depends on γ, we obtain the same gate fidelity for different standard deviations $\sigma(\zeta), \sigma(\theta)$. For instance, at $\gamma = 0.72$ and standard deviations of $\sigma(\zeta), \sigma(\theta) = 0.051\pi(2.55\ \%)$, the gate infidelity is $1 - \mathcal{F}^{\mathrm{co}}_{\mathrm{cz}}(\gamma = 0.72\pi, \sigma(\zeta)\sigma(\theta) = 0.051\pi) = 0.0068$. The same infidelity is achieved at $\gamma = 0$ with standard deviations of $\sigma(\zeta), \sigma(\theta) = 0.054\pi$ (2.7%). Conversely, for $\sigma(\zeta), \sigma(\theta) = 0.054\pi$ we achieve $1 - \mathcal{F}^{\mathrm{co}}_{\mathrm{cz}}(\gamma = 0.72\pi, \sigma(\zeta)\sigma(\theta) = 0.051\pi) = 0.0068$ as we know, and $1 - \mathcal{F}^{\mathrm{co}}_{\mathrm{cz}}(\gamma = 0.72\pi, \sigma(\zeta)\sigma(\theta) = 0.054\pi) = 0.0085$ at the minimum $\gamma = 0.72\pi$. Hence, the rotation angles close to $\gamma \approx 0$ and $\gamma \approx \pi$ are more error-robust against coherent noise than those close to $\gamma \approx 0.72\pi$. For $\sigma(\zeta), \sigma(\theta) < 0.04\pi$, the gate fidelity difference between distinct values of γ is negligible. The gate fidelity decreases below 99% for standard deviations$\sigma(\zeta), \sigma(\theta) > 0.06\pi$. For simplicity, we assume $\theta = \zeta$.

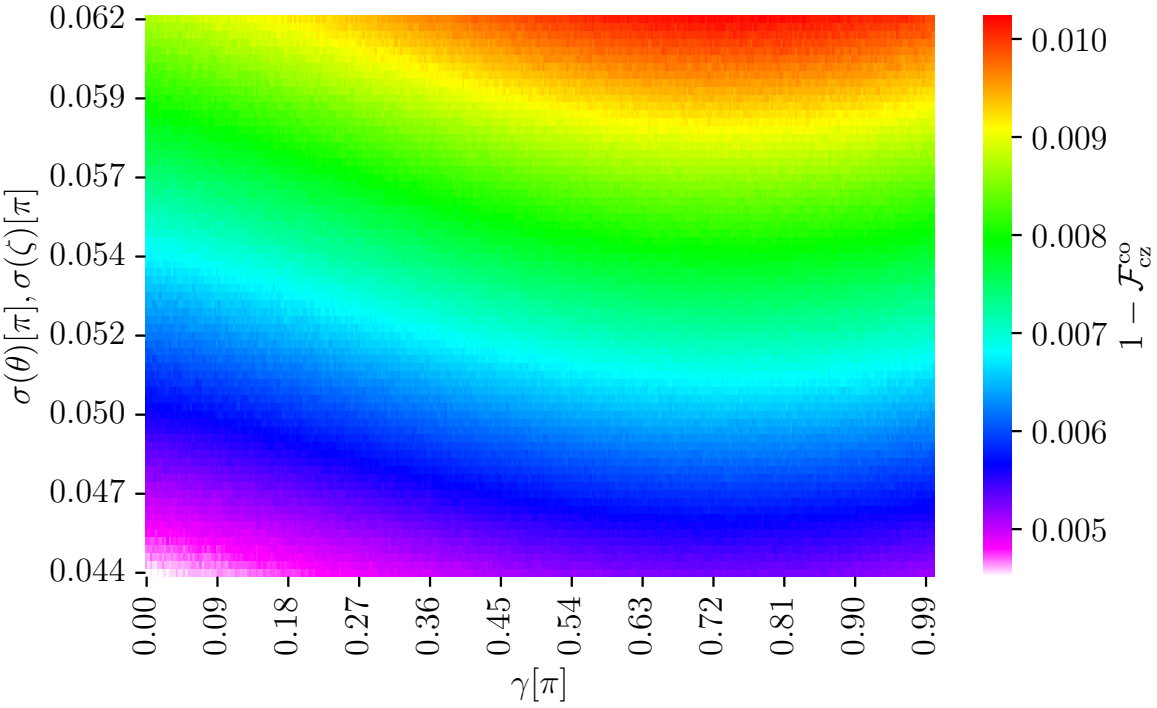


Figure 4.2: $R_{ZZ}(\gamma)$ gate fidelity in CZ gate decomposition plotted against γ rotation and coherent errors θ, ζ. The color map indicates the gate infidelity $1-\mathcal{F}_{cz}$. For θ and ζ, we independently picked 10000 times from a random Gaussian distribution with mean = 0 and std=$\sigma(\theta), \sigma(\zeta)$.

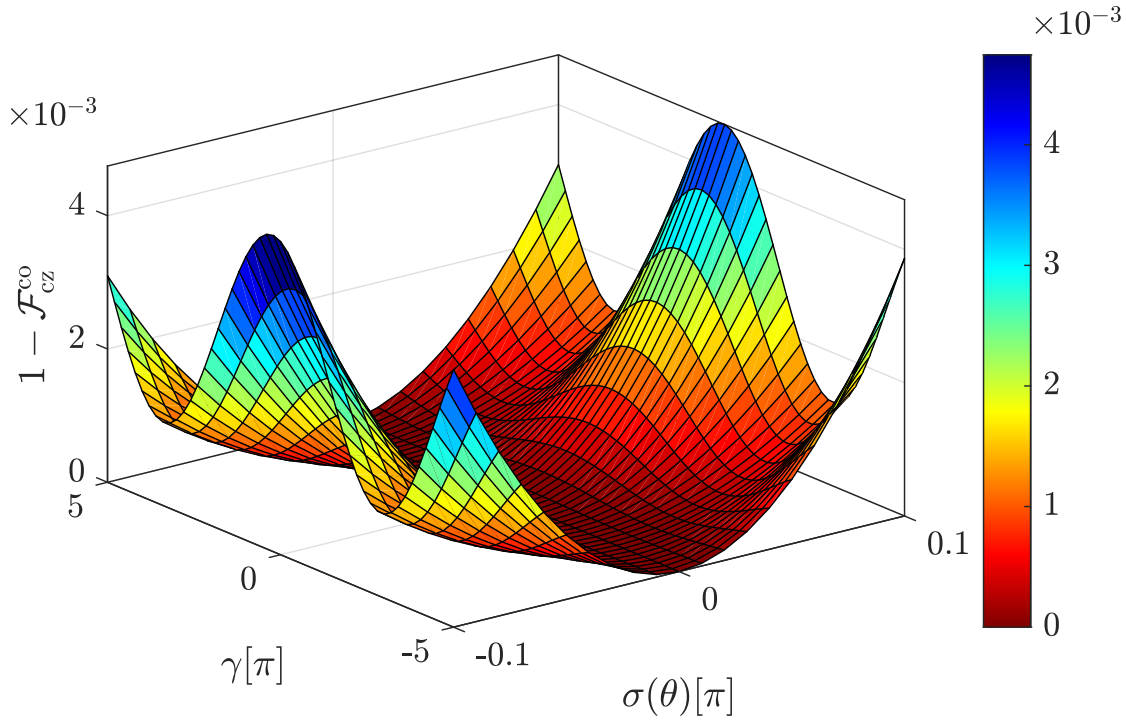


Figure 4.3: Plotting the gate fidelity of Equation (4.3.13) for γ and $\sigma(\theta)$ as the x-y-axis and the z-direction displays $1 - \mathcal{F}_{cz}^{co}$.

Using this assumption, we can deduce the following coherent error gate fidelity for the

CZ decomposition

$$\mathcal{F}_{\text{cz}}^{\text{co}} = 1 - \theta^2(0.3 + 0.04\sin(\gamma) + 0.17\cos(\gamma)) + \mathcal{O}(\theta^3). \tag{4.3.13}$$

Figure 4.3 illustrates the relationship between the gate infidelity, as defined in Equation (4.3.13), and the rotation angle. It is evident that the rotation angle significantly influences the gate infidelity. The difference between the minimum and maximum infidelity increases as the standard deviation $\sigma(\theta)$ increases.

4.3.3 CP gate fidelity under coherent noise

The error-free decomposition of the $R_{ZZ}(\gamma)$ gate into a CP gate is given by

$$\hat{U}_{\text{cp}}^{\text{co}} = R_{Z_1}(\gamma) \cdot R_{Z_2} \cdot \text{CP}(-2\gamma), \tag{4.3.14}$$

and the erroneous CP gate decomposition is according to

$$\hat{U}_{\text{cp}}^{\text{co}} = R_{Z_1}(\gamma) \cdot R_{Z_2}(\gamma) \cdot \text{CP}(-2\gamma + \theta). \tag{4.3.15}$$

This results in the following gate fidelity

$$\begin{aligned}
\mathcal{F}_{\text{cp}}^{\text{co}} &= \frac{1}{16}\sum_{j=1}^{16} |\, \langle\psi_j|(\hat{U}_{\text{cp}}^{\text{co}})^\dagger \cdot \hat{U}_{\text{cp}}|\psi_j\rangle \,|^2 \\
&= \sum_{j=1}^{16} |\, \langle\psi_j|R_{Z_1}^\dagger(-2\gamma) \cdot R_{Z_2}^\dagger(\gamma) \cdot \text{CP}^\dagger(-2\gamma + \theta) \cdot R_{Z_1}(\gamma) \cdot R_{Z_2} \cdot \text{CP}(-2\gamma)|\psi_j\rangle \,|^2 \\
&= \sum_{j=1}^{16} |\, \langle\psi_j|\text{CP}^\dagger(\theta)|\psi_j\rangle \,|^2.
\end{aligned} \tag{4.3.16}$$

The decomposition of $R_{ZZ}(\gamma)$ into CP gates for error-prone $\hat{U}_{\text{cp}}^{\text{co}}$ and error-free $\hat{U}_{\text{cp}}$ unitary is diagonal. This implies that all gate operations commute for the gate fidelity $\mathcal{F}_{\text{cp}}^{\text{co}}$, allowing the right side $\text{CP}^\dagger(\gamma+\theta)$ to cancel out with the Hermitian conjugate counterpart on the left side. As a result, the only remaining gate operation is the gate error unitary $\text{CP}^\dagger(\theta)$. Notably, the gate fidelity in CP decomposition under the coherent error is independent of the rotation angle γ. Three different state fidelities $f_{\text{cp},j}^{\text{co}}$ can be observed among all 16 states. The first set of states, $f_{\text{cp};1,2,3,6,7,10,11,16}^{\text{co}} = 1$, remain unaffected by the coherent error. The other two types of state fidelities are

$$\begin{aligned}
f_{\text{cp},4}^{\text{co}}, f_{\text{cp},5}^{\text{co}}, f_{\text{cp},8}^{\text{co}}, f_{\text{cp},9}^{\text{co}} &= \frac{1}{2}(1 + \cos(\theta)), \\
f_{\text{cp},12}^{\text{co}}, f_{\text{cp},13}^{\text{co}}, f_{\text{cp},14}^{\text{co}}, f_{\text{cp},15}^{\text{co}} &= \frac{1}{16}(10 + 6\cos(\theta)).
\end{aligned} \tag{4.3.17}$$

Therefore, the gate fidelity follows a cosine in dependence of θ, and we can deduce

$$\mathcal{F}_{\text{cp}}^{\text{co}} = \frac{1}{32}(25 + 7\cos(\theta)). \tag{4.3.18}$$

Figure 4.4 displays the gate fidelity and state fidelities for $\theta \in [0, 0.06\pi]$. At the maximum error bound of 0.06π, the gate fidelity decreases below 99.2%. $\mathcal{F}_{\text{cp}}^{\text{co}}(\theta = 0.06\pi)$ is 0.7% higher than $\mathcal{F}_{\text{cz}}^{\text{co}}(\gamma = 0.72, 0.06\pi) = 99.0\%$ at the minimum gate fidelity. For $\theta \leqslant 0.02\pi$, the gate fidelity is $\mathcal{F}_{\text{cz}}^{\text{co}} \leqslant 99.9\%$.

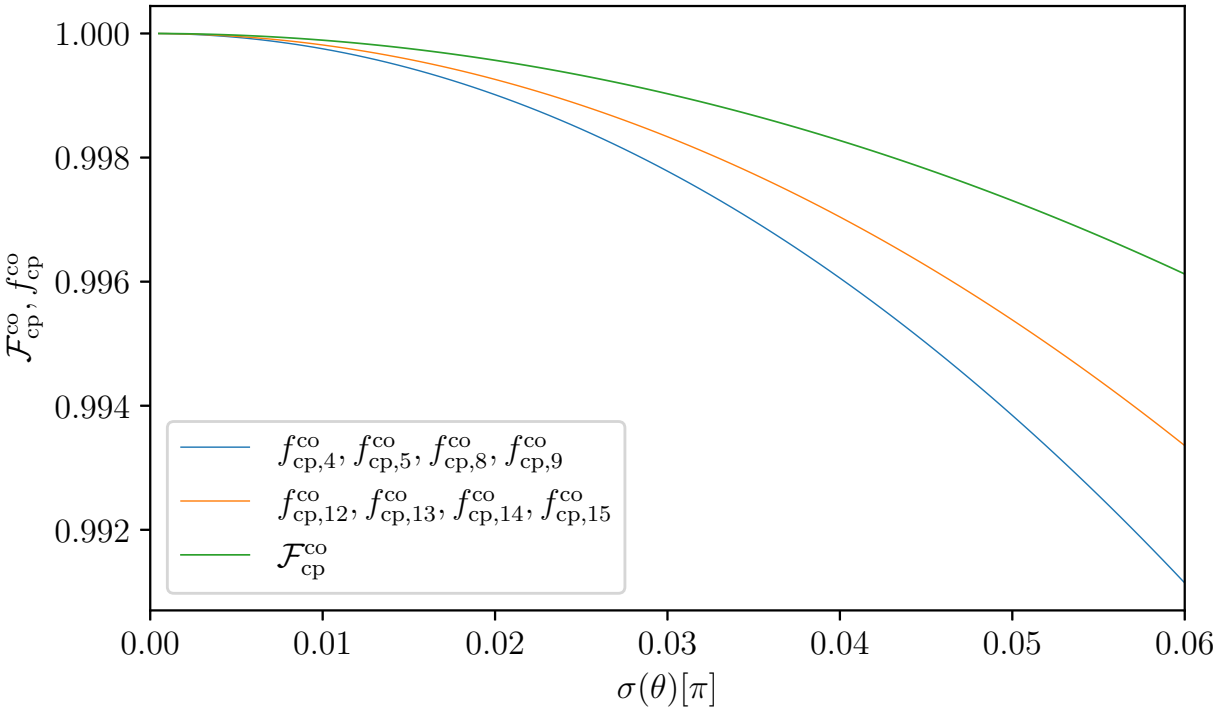


Figure 4.4: Gate fidelity of $R_{ZZ}(\gamma)$ in CP decomposition plotted against standard deviation $\sigma(\theta)$.

4.3.4 Comparing state fidelities

Although $\mathcal{F}_{\text{cp}}^{\text{co}}$ has a slight advantage over $\mathcal{F}_{\text{cz}}^{\text{co}}$, there are certain state fidelities f_j where $\mathcal{F}_{\text{cz}}^{\text{co}}$ is advantageous over $\mathcal{F}_{\text{cp}}^{\text{co}}$. To compare the state fidelities between CZ and CP decompositions, we define

$$\Delta f_j^{\text{co}} = f_{\text{cz},j}^{\text{co}} - f_{\text{cp},j}^{\text{co}}. \tag{4.3.19}$$

Depending on the rotation angle γ, CZ decomposition is advantageous over CP decomposition for $f_{4,8,13,15}$. Figure 4.5 exhibits the result of the state fidelities for f_4(left plot) and $f_{13,15}$(right plot). First, we investigate the left plot. When the standard deviations are high, $\sigma(\zeta), \sigma(\theta) \geqslant 0.05\pi$, and the rotation angle is close to $\gamma \approx \pi$ or $\gamma \approx 0$, the advantage of CZ decomposition over CP is 0.9%. On the other hand, for a rotation angle close to $\gamma = 0.5$ and high standard deviations of $\zeta(\theta), \sigma(\theta) \geqslant 0.05\pi$, Δf_4 shows an advantage for CP over CZ decomposition. We will further examine the state fidelities

$$f_{\text{cz},4}^{\text{co}} \propto (\theta^2(\cos(2\gamma) - 1)) \quad \text{and} \quad f_{\text{cp},4}^{\text{co}} \propto (1 - \theta^2/2), \tag{4.3.20}$$

to comprehend the change of Δf_4 in dependence of $\sigma(\zeta), \sigma(\theta)$ and γ. The cosine in $f_{\text{cz},4}^{\text{co}}$ leads to the highest deficit in gate fidelity for $\gamma \approx 0.5\pi$. On the other hand, $f_{\text{cz},4}^{\text{co}}$ is increased for high and low angles $\gamma \in \{0, \pi\}$, even with increased $\sigma(\zeta), \sigma(\theta)$. Since f_4^{cp} does not depend on γ, increasing $\sigma(\theta)$ results in the lowest state fidelity. Hence, Δf_4^{co}

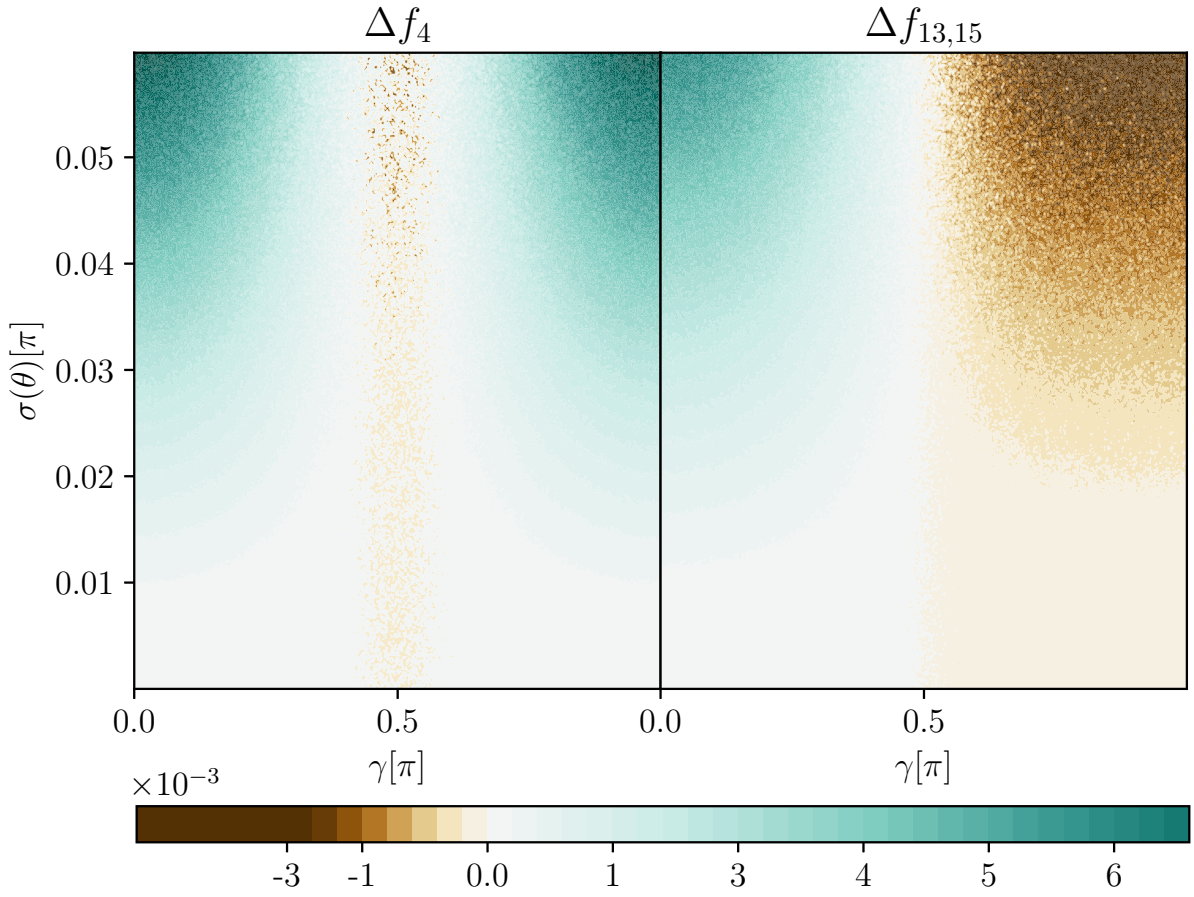


Figure 4.5: Plotting Δf_j against coherent error ($\sigma(\theta)$ on the y-axis) and the rotation angle (γ on the x-axis). The dark green color indicates an advantage of CZ decomposition over CP decomposition. White color indicates no advantage of one gate over the other. The dark brown color indicates an advantage of CP over CZ decomposition. Left plot: State fidelity difference for $f_4 = \frac{1}{\sqrt{2}}(0,0,1,1)^T$. Right plot: State fidelity difference for $f_{13} = \frac{1}{2}(i,i,1,1)^T$ and $f_{15} = \frac{1}{2}(1,1,1,1)^T$.

increases in combinations with high or low γ and increasing $\sigma(\theta)$. The right plot presents $\Delta f_{13,15}$. Compared to Δf_4^{co}, the state fidelity for CZ decomposition has a shift in the cosine, and the lowest state fidelity value is smaller than for Δf_4^{co}. For this reason, CP is advantageous over CZ when $\gamma \approx \pi$ for $\Delta f_{13,15}$.

4.4 CZ and CP gate under depolarizing noise

This section examines how the depolarizing channel $\mathcal{E}$ affects the gate fidelity of $R_{ZZ}(\gamma)$. Therefore, we decompose $R_{ZZ}(\gamma)$ into CZ and CP gates and compare their respective gate

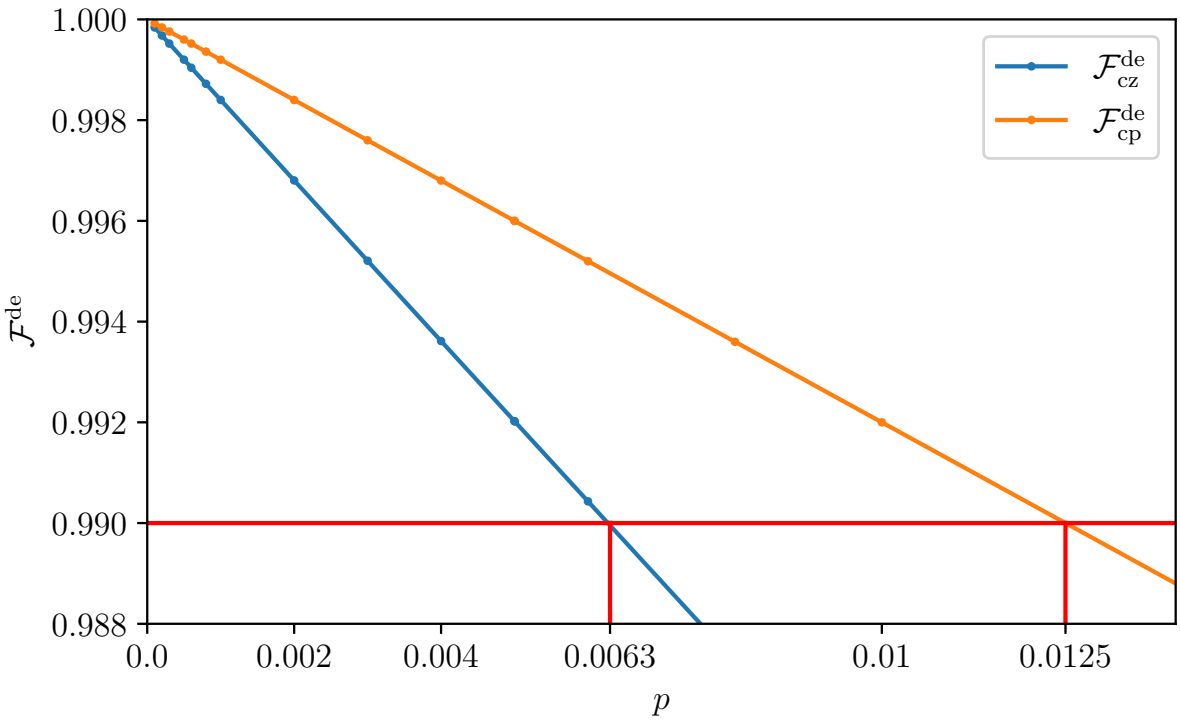


Figure 4.6: Gate fidelities for $R_{ZZ}(\gamma)$ plotted against the probability for depolarizing error p for CP (orange) and CZ (blue) decomposition. The red line indicates the 99% gate fidelity barrier.

fidelities. To calculate the gate fidelity, we will use

$$\begin{aligned}\mathcal{F}^{\mathrm{de}} &= \frac{1}{16}\sum_{j=1}^{16}\left(\mathrm{Tr}\left(\sqrt{\sqrt{\rho_j}\rho'_j\sqrt{\rho_j}}\right)\right) = \frac{1}{16}\sum_{j=1}^{16}\left(\mathrm{Tr}\left(\sqrt{\rho_j\rho'_j\rho_j}\right)\right) \\ &= \frac{1}{16}\sum_{j=1}^{16}\left(\mathrm{Tr}\left(\sqrt{\hat{U}\left|\psi_j\right\rangle\left\langle\psi_j\right|\hat{U}^\dagger\rho'_j\hat{U}\left|\psi_j\right\rangle\left\langle\psi_j\right|\hat{U}^\dagger}\right)\right) = \frac{1}{16}\sum_{j=1}^{16}\left\langle\psi_j\right|\hat{U}^\dagger\rho'_j\hat{U}\left|\psi_j\right\rangle .\end{aligned} \tag{4.4.1}$$

$|\psi_j\rangle$ represents the initial pure state and $\hat{U}$ a general unitary transformation, while ρ'_j denotes the density matrix achieved after applying the gate operation charged with an incoherent error on the state $|\psi_j\rangle$. Since ρ_j is a pure state, we have $\sqrt{\rho_j} = \sqrt{(\rho_j)^2} = \rho_j$. The state fidelity with respect to the depolarizing error and CP decomposition is

$$f^{\mathrm{de}}_{\mathrm{cp},j} = \left\langle\psi_j\right|U^\dagger_{\mathrm{cp}}\mathcal{E}(\hat{U}_{\mathrm{cp}}\left|\psi_j\right\rangle\left\langle\psi_j\right|\hat{U}^\dagger_{\mathrm{cp}})\hat{U}_{\mathrm{cp}}\left|\psi_j\right\rangle , \tag{4.4.2}$$

while for CZ decomposition, it is

$$f^{\mathrm{de}}_{\mathrm{cz},j} = \left\langle\psi_j\right|\hat{U}^\dagger_{\mathrm{cz}}\mathcal{E}(\mathcal{E}(\hat{U}_{\mathrm{cz}}\left|\psi_j\right\rangle\left\langle\psi_j\right|\hat{U}^\dagger_{\mathrm{cz}}))\hat{U}_{\mathrm{cz}}\left|\psi_j\right\rangle . \tag{4.4.3}$$

We apply the incoherent channel twice in the CZ decomposition due to the two CZ gates. Although the channels can be applied after each gate, we can use both after the unitary gate operation $\hat{U}_{\mathrm{cz}}$ because the depolarizing channels commute. Figure 4.6 shows the results for $p \leqslant 1.4\%$. $\mathcal{F}^{\mathrm{de}}_{\mathrm{cp}}$ reaches a gate fidelity of 99% for $p = 0.0125$. The same gate fidelity is attained for $\mathcal{F}^{\mathrm{de}}_{\mathrm{cz}}$ at $p = 0.0063$. Since we apply two depolarizing channels to the CZ decomposition, the fidelity is reached at approximately half the error rate of

CP decomposition. By applying the definition (4.1.1) of $\mathcal{E}$ on CP, we obtain the linear equation for the gate fidelity $\mathcal{F}_{\text{cp}}^{\text{de}} = 1 - 0.8p$. For the CZ gate fidelity, we deduce

$$\mathcal{F}_{\text{cz}}^{\text{de}} = (1-p)^2 + \frac{1}{4}(1-p)p + \frac{1}{16}p^2. \tag{4.4.4}$$

Expanding Equation (4.4.4) up to the first order results in $\mathcal{F}_{\text{cz}}^{\text{de}} = 1 - 1.55\,p$. Single-state fidelities are identical since the depolarizing error contracts the Bloch sphere uniformly. We demonstrated that for small standard deviations $\sigma(\zeta), \sigma(\theta)$ and low and high angles γ, the gate fidelities $\mathcal{F}_{\text{cz}}^{\text{co}}$ and $\mathcal{F}_{\text{cp}}^{\text{co}}$ yield similar results in the coherent case. However, the gate fidelity for CZ decomposition decreases twice as fast as p increases when exposed to the depolarizing error.

4.5 Performance of CZ and CP gate

In this section, we visualize coherent and incoherent errors for $R_{ZZ}(\gamma)$ into CZ and CP decomposition into one result. We summarize the results from both previous sections and recommend which gate should be used under which conditions. On top of that, we show the scaling behavior of both errors and how they are connected.

4.5.1 Advantage of CP over CZ gate

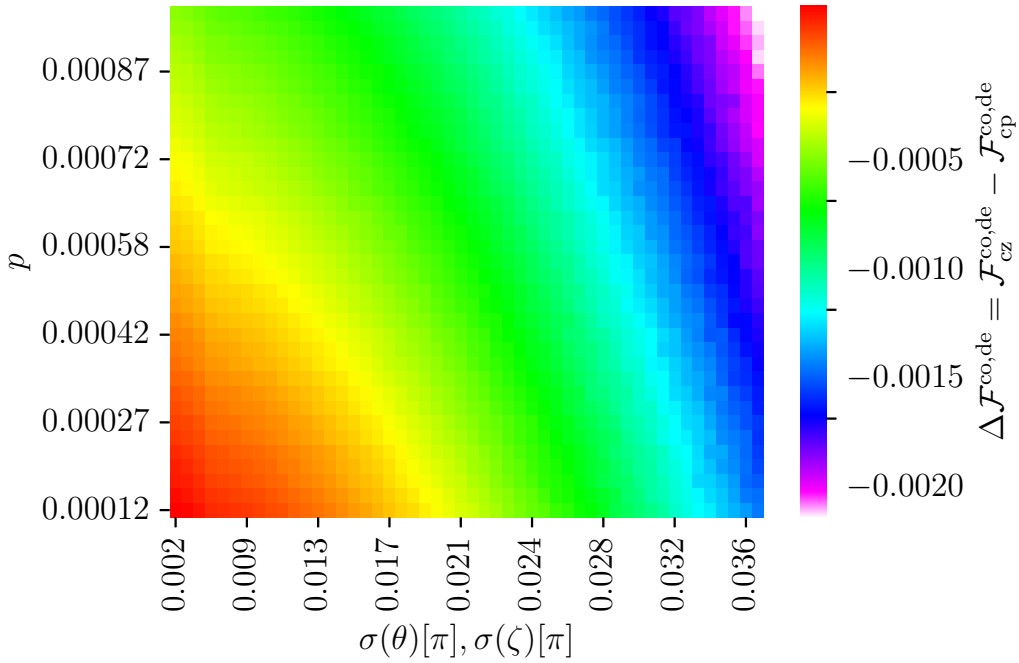


Figure 4.7: Plotting the fidelity difference $\mathcal{F}_{\text{cp}} - \mathcal{F}_{\text{cz}}$ in dependency of the coherent error θ (x-axis) and the depolarizing error p (y-axis). Purple or white corresponds to a high advantage of F_{cp} over F_{cz}, whereas the red color corresponds to comparable gate fidelity of both decompositions.

In Figure 4.7, we simultaneously apply the depolarizing and coherent error. For the simulation in Figure 4.7, we set the rotation angle to $\gamma = 0.01\pi$ to neglect the $\gamma \sin(\theta)$

effects in $\mathcal{F}_{\text{cz}}^{\text{de,co}}$. The plot depicts gate fidelities above 99%. If $\sigma(\theta) < 0.016(0.8\%)$ and $p < 0.032\%$, both decompositions yield similar gate fidelities, with a difference of at most 0.02% between $\mathcal{F}_{\text{cp}}^{\text{de,co}}$ and $\mathcal{F}_{\text{cz}}^{\text{de,co}}$. For $\sigma(\zeta), \sigma(\theta) \geqslant 0.016(0.8\%)$ and $p \geqslant 0.032\%$, gate fidelities for both decompositions drop below 99.8%.

Consequently, if we can suppress the incoherent error and allow for small coherent over-rotations, there would be no advantage of CP over CZ. If $\sigma(\zeta), \sigma(\theta) \geqslant 0.036(0.8\%)$ and $p \geqslant 0.0087\%$, we achieve a fidelity difference of about 0.3% (Figure 4.7; purple/white region). In Figure 4.7, $\mathcal{F}_{\text{cp}}^{\text{de,co}} \approx 99.8\% \pm 0.2\%$ for all error probabilities, while $\mathcal{F}_{\text{cz}}^{\text{de,co}} \leqslant 99.5\%$ in the lowest case (top-right corner). The polynomial scaling of the depolarizing error would be demonstrated by increasing the depolarizing error ($p > 0.1\%$). Thus, the heatmap would change from a radial color trend caused by the influence of $\sigma(\zeta), \sigma(\theta)$ and p on the gate fidelities to a polynomial trend in dependence on p for higher error rates. Of course, the circuit depth duplicates for CZ over CP decomposition. However, the pulse calibration for different angles for the parametric CP gate would also introduce another source of error. If both gates have small depolarizing error rates of $p \leqslant 0.032\%$ and small coherent over-rotations of $\sigma(\theta), \sigma(\zeta) < 0.016\pi\ (0.8\%)$, we recommend using the CZ gate over the CP gate. CP decomposition is advantageous over CZ decomposition for depolarizing error rates $p \geqslant 0.03\%$.

4.5.2 Scaling of coherent and incoherent error

This section examines the scaling behavior of coherent and incoherent errors concerning CZ and CP decomposition for $R_{ZZ}(\gamma)$. Given that coherent and incoherent errors commute, we can derive the following equation for the resulting density matrix ρ after applying CP decomposition $\hat{U}_{\text{cp}}^{\text{co}}$ and the incoherent channel $\mathcal{E}$

$$\mathcal{E}(\hat{U}_{\text{cp}}^{\text{co},\dagger}\rho\hat{U}_{\text{cp}}^{\text{co}}) = (1-p)\hat{U}_{\text{cp}}^{\text{co},\dagger}\rho\hat{U}_{\text{cp}}^{\text{co}} + p\mathbb{1}. \tag{4.5.1}$$

The above equation tells us the proportion of the unitary and incoherent error. We obtain the scaling quantities

$$\Omega_{\text{de,cp}} = p \quad \text{and} \quad \Omega_{\text{co,cp}} = (1-p). \tag{4.5.2}$$

The coherent error is associated with the input density matrix, represented by the factor $(1-p)$, denoting the probability of no incoherent error occurring. As decoherence increases, the proportion of coherent error decreases linearly.

However, Equation (4.5.2) only informs us about the scaling behavior of the errors without providing insight into the gate fidelity. It is important to note that, at the same error probability, the incoherent error will result in a larger reduction in gate fidelity than the coherent error. This is due to the exponential decay of the off-diagonal elements of the density matrix ρ, caused by the incoherent error.

After applying CZ error-prone gate decomposition $\hat{U}_{\text{cz}}^{\text{co}}$ and $\mathcal{E}$ we achieve

$$\mathcal{E}(\mathcal{E}(\hat{U}_{\text{cz}}^{\text{co},\dagger}\hat{U}_{\text{cz}}^{\text{co},\dagger}\rho\hat{U}_{\text{cz}}^{\text{co}}\hat{U}_{\text{cz}}^{\text{co}})) = (1-p)^2\hat{U}_{\text{cz}}^{\text{co},\dagger}\hat{U}_{\text{cz}}^{\text{co},\dagger}\rho\hat{U}_{\text{cz}}^{\text{co}}\hat{U}_{\text{cz}}^{\text{co}} + ((1-p)p+p)\mathbb{1}. \tag{4.5.3}$$

Thus, the proportions of the incoherent and coherent error in CZ decomposition are

$$\Omega_{\text{de,cz}} = (1-p)p + p. \quad \text{and} \quad \Omega_{\text{co,cz}} = (1-p)^2. \tag{4.5.4}$$

Figure 4.8 illustrates the proportions outlined in Equation (4.5.4) and (4.5.4). For $p \ll 0.1$, both CZ and CP decompositions exhibit identical proportions for coherent and incoher-

ent errors. At these low error probabilities, the gate fidelities $\mathcal{F}_{\text{cz}}^{\text{de,co}}$ and $\mathcal{F}_{\text{cp}}^{\text{de,co}}$ achieve comparable levels of accuracy. However, as $p \gg 0.1$, we observe linear scaling, denoted by $\Omega_{\text{de,cp}}$ and $\Omega_{\text{co,cp}}$, for CP decomposition, and polynomial scaling, denoted by $\Omega_{\text{de,cz}}$ and $\Omega_{\text{co,cz}}$, for CZ decomposition. Because the coherent error is linked to the input density matrix, and as the depolarizing error p increases, the proportion $\Omega_{\text{co,cz}}$ decreases. Due to the polynomial scaling of Ω in CZ decomposition, the gate fidelity $\mathcal{F}_{\text{cz}}^{\text{de,co}}$ decreases at a faster rate compared to $\mathcal{F}_{\text{cp}}^{\text{de,co}}$. In Figure 4.8, two dashed red lines represent the proportions of incoherent and coherent errors, with $\Omega_{\text{co}} = \Omega_{\text{de}} = 0.5$. The gate fidelities for CZ and CP decomposition coincide at $p = 0.29$ for CZ and $p = 0.5$ for CP.

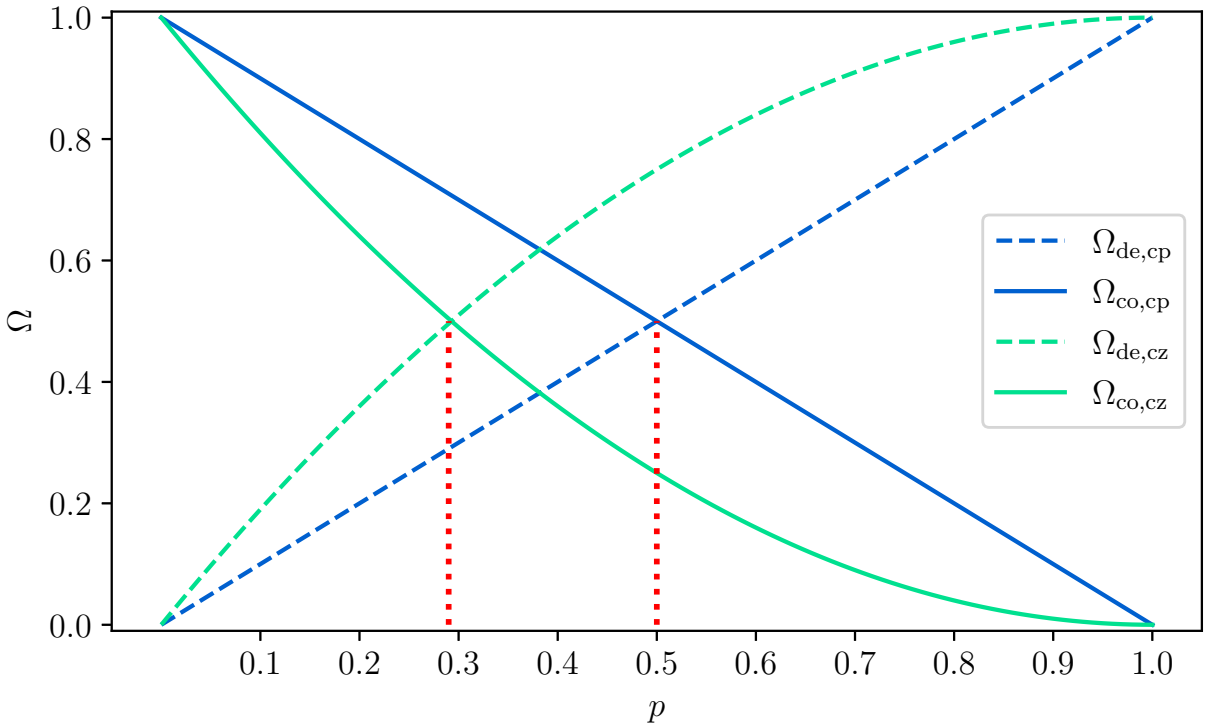


Figure 4.8: Plotting scale Ω (y-axis) of coherent and incoherent errors for $R_{ZZ}(\gamma)$ in CP and CZ decomposition against the depolarizing error p (x-axis). Dashed lines indicate the incoherent error proportion, and solid lines are the coherent error proportion. Green-colored curves are in CZ decomposition, and blue-colored curves are in CP decomposition.

4.6 Decomposing $R_{ZZ}(\gamma)$ into iSWAP gate

In hardware implementations, the iSWAP gate is also widely used as the non-local two-qubit gate. It belongs to the same equivalence class as CZ. CZ and iSWAP are members of the XY family. The general XY-gate is defined as

$$XY_{lk}(\alpha) = \exp\left(-\frac{i}{2}\alpha\left(X_l X_k + Y_l Y_k\right)\right), \tag{4.6.1}$$

with α being the rotation angle. The two-qubit gates CZ and iSWAP are achieved by tuning the angle α

$$\text{iSWAP} = XY(\alpha = 1.5) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.6.2}$$

In order to decompose $R_{ZZ}(\gamma)$ into iSWAP gates, we need to check if both gates belong to the same equivalence class. If so, both gates only differ in single-qubit rotations. To test this, we illustrate the results by Yuriy Makhlin [113] and use them to prove that iSWAP and $R_{ZZ}(\gamma)$ do not belong to the same equivalence class. First, Mahklin [113] showed that local two-qubit gates $\hat{U}_1 \otimes \hat{U}_2$ with $\hat{U}_1$, $\hat{U}_2 \in SU(2)$ and unit determinant can be transformed into subgroup $SO(4)$, which consists of real and orthogonal matrices in Bell basis explicit. Moreover, he proved that for every two-qubit gate $\hat{U}$ with $\det(\hat{U}) = 1$, the local invariants are determined by the eigenvalues of the characteristic polynomial of $m(\hat{U}) = \hat{U}_B^T \hat{U}_B$ with

$$\hat{U}_B = \hat{Q}^\dagger \hat{U} \hat{Q}, \tag{4.6.3}$$

being the transformation from standard basis $|00\rangle, |11\rangle, |01\rangle, |10\rangle$ into Bell basis $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, $\frac{i}{\sqrt{2}}(|01\rangle + |10\rangle)$, $\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$, $\frac{i}{\sqrt{2}}(|00\rangle - |11\rangle)$ with

$$\hat{Q} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & i \\ 0 & i & 1 & 0 \\ 0 & i & -1 & 0 \\ 1 & 0 & 0 & -i \end{pmatrix}. \tag{4.6.4}$$

We need to follow a specific procedure to decide if two randomly selected unitary matrices $\hat{L}$ and $\hat{U}$ belong to the same equivalence class in $SU(4)$. Firstly, we must convert both matrices from the standard basis to the Bell basis using $\hat{L}_B^T \hat{L}_B$ and $\hat{U}_B^T \hat{U}_B$. Secondly, we need to compute the characteristic polynomial, which is expressed in the following form

$$\begin{aligned} \chi(m(\hat{U})) = \det(\lambda I - m(\hat{U})) = \lambda^4 - \text{Tr}(m(\hat{U}))\lambda^3 + \frac{1}{2}(\text{Tr}^2(m(\hat{U})) - \text{Tr}(m(\hat{U})^2))\lambda^2 \\ + \overline{\text{Tr}(m(\hat{U}))} s + 1. \end{aligned} \tag{4.6.5}$$

Hence, we have to calculate the two values

$$\text{Tr}(m) \text{ and } \text{Tr}^2(m) - \text{Tr}(m^2). \tag{4.6.6}$$

If both values are identical, we are in the same equivalence class. So Makhlin [113] restricts his results to two-qubit gates with $\det(\hat{U}) = 1$ first. Makhlin [113] also investigates the general two-qubit gates $\hat{N} \in SU(4)$. We focus on $SU(4)$ because every gate $\hat{K} \in U(4)$ can be decomposed into $\hat{K} = e^{i\alpha}\hat{K}_1$ with $\hat{K}_1 \in SU(4)$. General two-qubit gates are defined up to a global phase factor $e^{i\alpha}$. To account for this factor, we need to divide the two invariants by the determinant and the corresponding normalization constant, which leads us to obtain

$$G_1 = \frac{\text{Tr}(m(\hat{U}))}{16 \det(\hat{U})}, \tag{4.6.7}$$

$$G_2 = \frac{\mathrm{Tr}^2(m(\hat{U})) - \mathrm{Tr}(m(\hat{U})^2)}{4\det(\hat{U})}. \tag{4.6.8}$$

The values of the invariants for both gates and additional ones are noted in the following table. We use the notation [Gate] and select a representative gate from that class to

Gate	G_1	G_2
iSWAP	i/4	0
$R_{ZZ}(\gamma)$	$\frac{1}{2} + \frac{1}{2}\cos(2\gamma)$	$2 + \cos(2\gamma)$
CP	$\frac{1}{2} + \frac{1}{2}\cos(2\gamma)$	$2 + \cos(2\gamma)$
CZ	0	1

Table 4.2: local invariants G_1, G_2 for different gates

designate a specific equivalence class. The SWAP class, denoted as [SWAP], encompasses iSWAP and has invariant values of $G_1 = i/4$ and $G_2 = 0$. The [CP] class, which includes $R_{ZZ}(\gamma)$, exhibits invariant values of $G_1 = \frac{1}{2} + \frac{1}{2}\cos(2\gamma)$ and $G_2 = 2 + \cos(2\gamma)$. Hence, it can be concluded that iSWAP and $R_{ZZ}(\gamma)$ are not equivalent. According to [112], we can transform between two equivalence classes by applying the original Gate (iSWAP) twice to obtain the target gate $R_{ZZ}(\gamma)$. Furthermore, as per [112], we require local rotation gates $\hat{U}_1, \hat{U}_2, \hat{U}_3 \in SU(2) \otimes SU(2)$ between the two-qubit gates. Thus, the general form for decomposing $R_{ZZ}(\gamma)$ into iSWAP gates is

$$R_{ZZ}(\gamma) = \hat{U}_1(\vec{\omega}) \cdot \mathrm{iSWAP} \cdot \hat{U}_2(\gamma, \vec{\varsigma}) \cdot \mathrm{iSWAP} \cdot \hat{U}_3(\vec{\eta}). \tag{4.6.9}$$

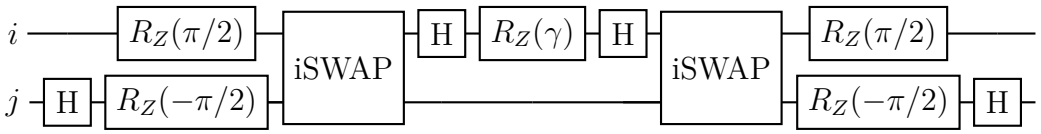
We restrict our rotation angle γ to be in $\hat{U}_2$. Every single-qubit gate can be described by a unitary of $SU(2)$ with

$$\hat{U}(\beta, \kappa, \mu) = e^{i\alpha} R_Z(\beta) \cdot R_Y(\kappa) \cdot R_Z(\mu), \tag{4.6.10}$$

and three parameters contained in R_X and R_Y rotations. α is the global phase. Our local two-qubit gates $\hat{U}_i \in SU(2) \otimes SU(2)$ can be described by six parameters. To optimize our circuit, we need to determine the optimal values for the 18 variables representing the single-qubit rotations. We utilize the BFGS optimizer to identify these values. Once we have found the optimal parameters, we can decompose the local gates into Hadamard and R_Z rotation gates, which will provide us with the final decomposition

$$R_{ZZ}(\gamma) = \hat{U}_1(\vec{\omega}) \cdot \mathrm{iSWAP} \cdot \hat{U}_2(\gamma, \vec{\varsigma}) \cdot \mathrm{iSWAP} \cdot \hat{U}_3(\vec{\eta})$$

$=$



. (4.6.11)

If quantum hardware has an iSWAP gate available, we only need to calibrate the R_Z gate and the Hadamard gate on the machine to obtain the [CP] class.

4.7 Conclusion

In this chapter, we compared $R_{ZZ}(\gamma)$ in CZ and CP gate decomposition. Considering both small coherent and incoherent errors, we have discovered that the gate fidelities $\mathcal{F}_{\text{cz}}^{\text{de,co}}$ and $\mathcal{F}_{\text{cp}}^{\text{de,co}}$ exhibit comparable results. Depending on the state, CZ gate even provides an advantage in state fidelities for pure incoherent errors. This is important because, for variational quantum algorithms, the parameter space tends to accumulate numerous local minima as a result of coherent errors. This accumulation of minima introduces complexity to the optimization process. Consequently, using a QAOA on hardware that utilizes CZ gates and has a short circuit depth can provide advantages compared to hardware that relies on CP gates. Regarding coherent errors, we discovered that the gate fidelity of $R_{ZZ}(\gamma)$ in CZ decomposition depends on the rotation angle γ. In the case of small coherent errors and the absence of incoherent noise, the gate fidelity is higher for small rotation angles than for larger ones. Additionally, we derived an analytical equation that expresses the gate fidelity for CZ decomposition as a function of the coherent error.

Furthermore, we have demonstrated a gate decomposition method for $R_{ZZ}(\gamma)$ that employs the iSWAP gate, assuming the availability of a Hadamard gate and Pauli-Z rotations. The iSWAP decomposition consists of two fixed two-qubit gates like CZ, and we expect similar gate fidelities between the two decompositions. Since our quantum channels do not affect the single-qubit gates, only the state fidelities will change. Therefore, we have not conducted numerical and analytical studies for iSWAP decomposition and instead refer to CZ decomposition.

Chapter 5

QAOA in the digital-analog scheme

This chapter will discuss the publication "Approximating the quantum approximate optimization algorithm with digital-analog interactions" released in Phys. Rev. A 106, 042446 (2022). This project was a joint effort between David Headley from Daimler-Benz AG and Ana Martin from IKERBASQUE, Bilbao, Spain. We aimed to combine David Headley and my knowledge of quantum algorithms with Ana Martin's expertise in the digital-analog scheme. My primary contribution to this project was the numerical implementation of the QAOA digital-analog approach, while David Headley was responsible for the analytic bound estimation. This chapter aims to integrate the Quantum Approximate Optimization Algorithm (QAOA) into the digital-analog paradigm. We demonstrate that when employing fast single-qubit gates, the digital-analog approach of QAOA achieves comparable results to the digital implementation.

This chapter is structured as follows: We provide a concise overview of the history of digital-analog quantum computing in Section 5.1. Following that, in Section 5.2, we study the mathematical concepts that enable us to translate an algorithm into a digital-analog device. The primary focus of Section 5.3 is to clarify the specific conditions under which the QAOA implemented in the digital-analog scheme produces comparable results to its digital counterpart.

5.1 Digital Analog Quantum Computing

Analog quantum simulators such as [116, 117] have limitations when executing various algorithms, unlike a universal quantum computer that can simulate arbitrary Hamiltonians using one- and two-qubit gates, as demonstrated in [18]. Due to high circuit depth, the gate-based technique is typically vulnerable to numerous coherent and incoherent errors. Apart from depolarizing errors, the system can also encounter cross-talk issues. To illustrate, constructing a two-qubit gate in a superconducting platform requires bringing the two qubits that are intended to interact into resonance. Unfortunately, the qubit pair sometimes transmits an electric signal to uninvolved qubits, leading to disruptions. Capacitively or inductively coupled qubits typically cause this phenomenon, which can also occur when the qubits are decoupled and go off-resonance. As a result, quantum algorithms like Shor can encounter cross-talk issues when a large number of qubits are required. In contrast, analog quantum computers, such as quantum annealers, are less susceptible to these types of errors compared to gate-based machines [118]. In theory, every analog device has the potential to perform any algorithm that a gate-based machine can [119, 120]. Nonetheless, this concept is mainly theoretical, as in practice, the evolution times required are often too long and would exceed the coherence time. In the case of an analog quantum computer, all qubits are brought into resonance, and the system's Hamiltonian evolves in time without any external disturbances. Digital-analog quantum computing integrates the gate-based and the analog approach. The purpose is to merge all non-local two-qubit gates into a single Hamiltonian, which is then simulated in time. The system is controlled by fast single-qubit gates, which are considerably faster than two-qubit gates by a factor of roughly two to three orders of magnitude [121, 122], leading to minimal disruptions.

The origins of digital-analog quantum computing can be traced back to nuclear magnetic resonance [123] and quantum optimal control [124]. For example, assume a Hamiltonian $\hat{H}$ needs to be simulated for time t using a quantum simulator. However, the simulator can only access $\hat{H}'$. So the question that arises is, can we decompose the simulator $\hat{H}'$ into

$$e^{-i\hat{H}t} = e^{-i\hat{H}'t_1}U_1 e^{-i\hat{H}'t_2}U_2 \dots e^{-i\hat{H}'t_n}U_n, \tag{5.1.1}$$

with control unitaries U_i to construct the entire time evolution $e^{-i\hat{H}t}$. The first theoretical concepts for digital-analog quantum computing were proposed by Dodd et al. [125] and Parra-Rodriguez et al. [126], among others [127, 128, 129]. Parra-Rodriguez et al. [126] demonstrated the digital-analog procedure for an Ising-like problem, which naturally appears in many algorithms. Meanwhile, Dodd et al. [125] presented a different decomposition technique that has been proven to be universal. The first digital-analog quantum simulators for chemistry [130] and other models [131, 132] for superconducting qubits have already been proposed. The first quantum algorithms, such as the quantum Fourier transformation, have also been translated into the digital-analog approach [133]. The work of Ana Martin et al. [133] is a significant cornerstone in the field. QFT is a subroutine in numerous algorithms, including quantum phase estimation and the HHL algorithm, as introduced by Harrow et al. in [10]. Ana Martin and her colleagues provide evidence demonstrating the noise resilience of the QFT against coherent errors. In addition, Garcia-Molina et al. [134] showed that the QFT is more robust against incoherent errors than its digital counterpart.

5.2 Digital-analog paradigm

This section will discuss the digital-analog paradigm's theory and how to implement it on a physical machine, as outlined in the work by Parra-Rodriguez et al. [126]. As abovementioned, we want to merge the error resilience of the analog computer with the freedom of choice in the algorithms that a gate-based quantum computer offers.

5.2.1 Calculating digital-analog blocks

This section provides a summary of the research conducted by Parra-Rodriguez et al. as documented in their work [126]. We assume that we have access to a device Hamiltonian, an analog quantum computer capable of bringing all qubits into resonance and allows us to control the device Hamiltonian by turning it on and off. We further demand access to single-qubit gates we can apply to the system as required.

We now demonstrate how to map every algorithm's two-qubit gates onto the analog device according to [126]. For this purpose, we consider an arbitrary two-body Ising Hamiltonian,

$$\hat{H}_{\text{arb}} = \sum_{i<j} g_{ij} Z_i Z_j, \tag{5.2.1}$$

with an all-to-all connection (ATA), with the aim of determining the ground state, for example. To prevent errors, we aim to simulate this Hamiltonian in time instead of decomposing every ZZ coupling into a two-qubit gate. The objective is to achieve the desired transformation. To accomplish this, we will utilize a homogenous Ising Hamiltonian

$$\hat{H}_{\text{device}} = g \sum_{i<j} Z_i Z_j, \tag{5.2.2}$$

as our device. For the sake of simplicity, we set $g = 1$ in the device Hamiltonian. The Ising Hamiltonian can construct universal Z_iZ_j gates [135]. As previously mentioned, our goal is to simulate $\hat{H}_{\text{arb}}$ on the device $\hat{H}_{\text{device}}$, within a time of t_F, through the use of the time evolution operator $\hat{U}_{\text{arb}}(t) = e^{-i\hat{H}_{\text{arb}}t_F}$. Therefore, we must determine the appropriate mapping

$$t_F \sum_{i,j}^{n} g_{i,j} Z_i Z_j = \sum_{i<j}^{n} \sum_{l<m}^{n} t_{lm} X_l X_m Z_i Z_j X_l X_m. \tag{5.2.3}$$

Equation (5.2.3) illustrates the transformation from one Ising Hamiltonian to another. We can solve this equation for the time intervals t_{lm} and combine the resulting values with Pauli-X rotations around the ZZ rotation gates on the device Hamiltonian with n qubits. By doing so, we can map the problem Hamiltonian onto the device. We replace the digital gates with at most $n(n-1)/2$ time slices. The total simulation time of the system is given by the sum $\sum_{l<m} t_{lm}$. To compute t_{lm}, we commute the Pauli-X gates from the left side of the Z_iZ_j to the right site. For single-qubit indices $l, m \neq i \neq j$, we can easily commute through, and the Pauli-X gates cancel out. However, for the case of $Z_iZ_jX_j = -X_jZ_iZ_j$, a minus sign arises, and we can rewrite the right side of Equation (5.2.3) to

$$t_F \sum_{i,j}^{n} g_{i,j} Z_i Z_j = \sum_{i<j}^{n} \sum_{l<m}^{n} t_{lm} (-1)^{\delta_{li} + \delta_{lj} + \delta_{mi} + \delta_{mj}} Z_i Z_j. \tag{5.2.4}$$

In the next step, we vectorize the index pairs $(l, m) \to \omega$ and $(i, j) \to \zeta$ with

$$\omega = n(l-1) + \frac{l(l+1)}{2} + m \quad \text{and} \tag{5.2.5}$$

$$\zeta = n(i-1) + \frac{i(i+1)}{2} + j. \tag{5.2.6}$$

By using the vectorization, we can express the Kronecker sign in Equation (5.2.4) as a two-dimensional matrix with

$$M_{\omega\zeta} = (-1)^{\delta_{li} + \delta_{lj} + \delta_{mi} + \delta_{mj}}. \tag{5.2.7}$$

The Matrix inversion of $M_{\omega\zeta}$ will provide us the time intervals t_ω with

$$g_\zeta = (1/t_F) M_{\omega\zeta} t_\omega \to t_\omega = M^{-1}_{\omega\zeta} g_\zeta t_F, \tag{5.2.8}$$

with $g_{ij} \to g_\zeta$ and $t_{lm} \to t_\omega$.

We have demonstrated that we can compute the time intervals between the various qubit interactions by matrix inversion $M^{-1}_{\omega\zeta}$. With this method, we can now simulate

$$\hat{U}_{\text{arb}}(t) = e^{-it_F \hat{H}_{\text{arb}}} \quad \text{with} \quad \hat{U}_{\text{device}}(t_F) = \prod_{l<m}^{n} X_l X_m e^{t_{lm} \sum_{i<j}^{n} Z_i Z_j} X_l X_m. \tag{5.2.9}$$

To simplify the above expression, we apply the X rotations as NOT gates when using them on a digital-analog device. This can be achieved using the rule

$$\begin{aligned} e^{it\hat{U}\hat{W}\hat{U}^\dagger} &= \sum_{k=0}^{\infty} \frac{(it)^k}{k!} (\hat{U}\hat{W}\hat{U}^\dagger)^k = \ldots + \frac{(it)^2}{2!} (\hat{U}\hat{W}\cancel{\hat{U}^\dagger\hat{U}}\hat{W}\hat{U}^\dagger) + \ldots \\ &= \hat{U}\left(\ldots + \frac{(it)^2}{2!}\hat{W}^2 + \ldots \hat{U}^\dagger\right) = \hat{U} e^{it\hat{W}} \hat{U}^\dagger, \end{aligned} \tag{5.2.10}$$

for any unitary operation $\hat{U}$ and Pauli-X gates. This allows us to simplify the right-hand side of Equation (5.2.9). We can use Equation (5.2.9) to simulate entangling two-qubit gates on an analog device by decomposing every algorithm's two-qubit gates into an inhomogeneous Ising chain. This allows us to perform the simulation on the device instead of a digital quantum computer. Although the sign matrix cannot be inverted directly for $n = 4$, a solution is presented in [126].

All the concepts presented in this chapter can be extended to inhomogeneous and non-all-to-all connected device Hamiltonians, as discussed in [136]. Galicia et al. [136] address the challenge of dealing with practical hardware limitations by constraining the device Hamiltonian to graphs with limited connectivity. The only difference in their approach would be the compiled circuits, as constraints in the device Hamiltonian's connectivity may require more SWAP gates. The matrix inversion provides us with the time intervals, as shown in Equation (5.2.8). Still, it does not indicate how to implement it on an actual device due to the possibility of negative time intervals.

5.2.2 Negative times in the digital-analog approach

To simulate in the DAQC approach, we must keep the problem Hamiltonian unchanged because we are simulating the Hamiltonian throughout $\sum_{l<m}^{n} t_{lm}$. Hence, we have to ensure that all times t_{lm} are positive. We will elaborate on how to achieve this for both homogenous and inhomogeneous device Hamiltonians. Furthermore, there is a general concept that is applicable regardless of the device Hamiltonian.

We begin by considering the homogenous ATA device Hamiltonian. In this case, we can take advantage of the periodicity of the unitary and add 2π to every time block. However, with the always-on resource Hamiltonian, the required time would be too long. Instead of gaining an error advantage by reducing cross-talk, we would exceed the coherence time. Another approach involves utilizing the eigenvalues [126]. To start, we can express the matrix M multiplied by the time vector $\vec{t}$ as

$$M\vec{t} = M(\vec{t} - t_{\min}\vec{1} + t_{\min}\vec{1}), \tag{5.2.11}$$

with $t_{\min}$ being the minimal time of the time vector with $t_{\min} < t_\omega$ for all ω. $\vec{1}$ is a column vector filled with ones. Intuitively, $\vec{1}$ is an eigenvector of matrix M. Following from the fact that M is a symmetric matrix built up on Pauli-X commutation relations. Furthermore, the plus state $|+\rangle$, which is an eigenstate of the Pauli-X gate, is proportional to $\vec{1}$. Therefore, we deduce

$$M\vec{t} = M(\vec{t} - t_{\min}\vec{1}) + \lambda t_{\min}\vec{1}. \tag{5.2.12}$$

λ is the eigenvalue of M for eigenvector $\vec{1}$. Consequently, the time vector $\vec{t^*} = \vec{t} - t_{\min}\vec{1}$ for the sign matrix is

$$M\vec{t} = M\vec{t^*} + \lambda t_{\min}\vec{1}. \tag{5.2.13}$$

We have an additional analog block for simulating $\lambda\vec{1}$. However, it is crucial to ensure that the eigenvalue λ is negative so that this block evolves in positive time. The eigenvalue turns out to be negative for $n < 7$ and for $n = 3, 5, 6$, while the matrix becomes singular at $n = 4$. For $n > 6$, the eigenvalue is always positive, which could lead to the extra block $\lambda t_{\min}$ being negative if $t_{\min} < 0$. To circumvent this issue, one possible solution is to add 2π to the exponent, thereby ensuring that the homogenous resource Hamiltonians remain positive. By refraining from adding 2π to every analog block, we can prevent the coherence lifetime of the qubit from being exceeded. The periodicity argument we lose in the case of inhomogeneous device Hamiltonians. In such a case, we can convert $\hat{H}_{\text{arb}} \to -\hat{H}_{\text{arb}}$ and use the problem Hamiltonian's periodicity if possible. In this study, we focus primarily on Max-cut problems, which have multiple integer values of 1/2 as couplings ($g_{ij} \in N/2$ with $N \in \mathbb{Z}+$). To account for this, we introduce a minus sign to the problem Hamiltonian and time, which we then calculate in modulo 2π. By doing so, we can absorb the negative sign of the problem Hamiltonian into the matrix M and ensure that the eigenvalue λ is negative. Alternatively, we can achieve $-M$ by inverting the protocol. In this case, instead of sandwiching the ZZ couplings with $X1X_2ZZX_1X_2$, we would sandwich them with $X_3..X_nZZX_3..X_n$. The time evolution of the device unitary, including the extra block, is given by

$$\hat{U}_{\text{device}}(t_F) = \exp\left(i\sum_{\zeta,\omega} M_{\omega\zeta}t^*_\omega Z_iZ_j\right)\exp\left(\lambda t_{\min}\sum_{\zeta} Z_iZ_j\right). \tag{5.2.14}$$

5.2.3 stepwise Digital Analog Quantum Computing

The Digital Analog Quantum Computing (DAQC) paradigm encompasses two distinct procedures. The first is the stepwise Digital Analog Quantum Computing (sDAQC) scheme, as depicted in Figure 5.1. This scheme consists of analog Hamiltonian simulation components interspersed with single-qubit X rotations, each requiring a duration of Δt, corresponding to the time required for a full rotation of a Pauli-X gate. Notably, this duration is approximately two to three orders of magnitude shorter [121, 122] than the typically required time for a two-qubit gate. To implement this procedure, a device must be capable of activating and deactivating the device Hamiltonian and enabling access to single-qubit rotations. When all sources of error are disregarded, sDAQC is identical to the pure digital approach.

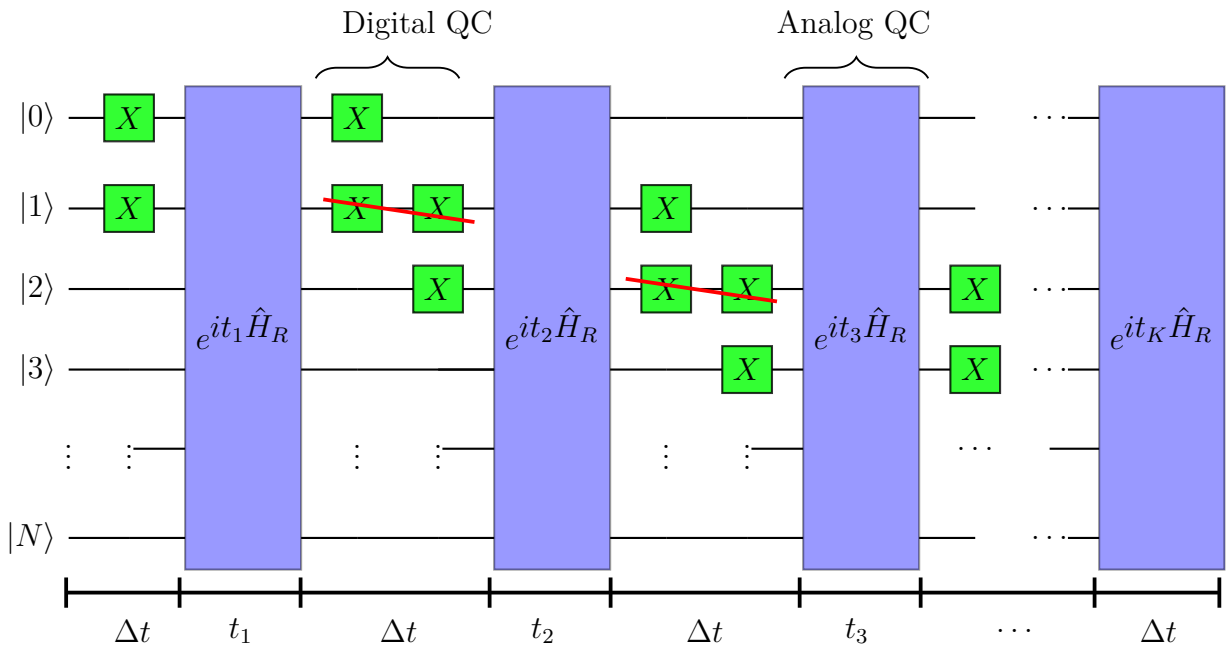


Figure 5.1: General stepwise Digital Analog Quantum Computing(sDAQC): Blue gates: depicted are the analog blocks for the times t_{lm} for a general resource Hamiltonian of size n and analog blocks of depth $t_K = n(n-1)/2$. Green gates: single-qubit X rotation gates that are applied in time Δt in the digital part. The bottom black line indicates the timeline. Red lines indicate cancellation of single-qubit rotations $X_i X_i = 1$ of adjacent time slices for example t_{12} and t_{23}.

5.2.4 banged Digital Analog Quantum Computing

The alternative procedure to sDAQC is the banged Digital Analog Quantum Computing (bDAQC) approach. In the bDAQC scheme, we activate the device Hamiltonian at the beginning and deactivate it at the end of the time evolution. Furthermore, we disrupt the system with fast single-qubit gates in the interim.

The bDAQC scheme is illustrated in Figure 5.2, which entails incorporating X rotation gates into the device Hamiltonian. These gates can be applied simultaneously during a time interval of Δt. To transform Pauli-X gates into X rotation gates, we use $X = e^{iXt}$

for $t = \pi/2$. As a result, the time evolution unitary for bDAQC can be expressed as

$$e^{i(\hat{H}+X)\Delta t} = e^{i\hat{H}\Delta t}e^{iX\Delta t} + \mathcal{O}(\Delta t^2). \tag{5.2.15}$$

The subsequent inquiry pertains to the Trotter error's size, which is $\mathcal{O}(\Delta t^2)$, and how it compares to that of the digital approach. This is dependent on the speed of the single-qubit X gates. In the case of infinitely fast single-qubit gates, with $\Delta t \to 0$, bDAQC is identical to sDAQC. Suppose the Trotter error is insignificant for the chosen device. In

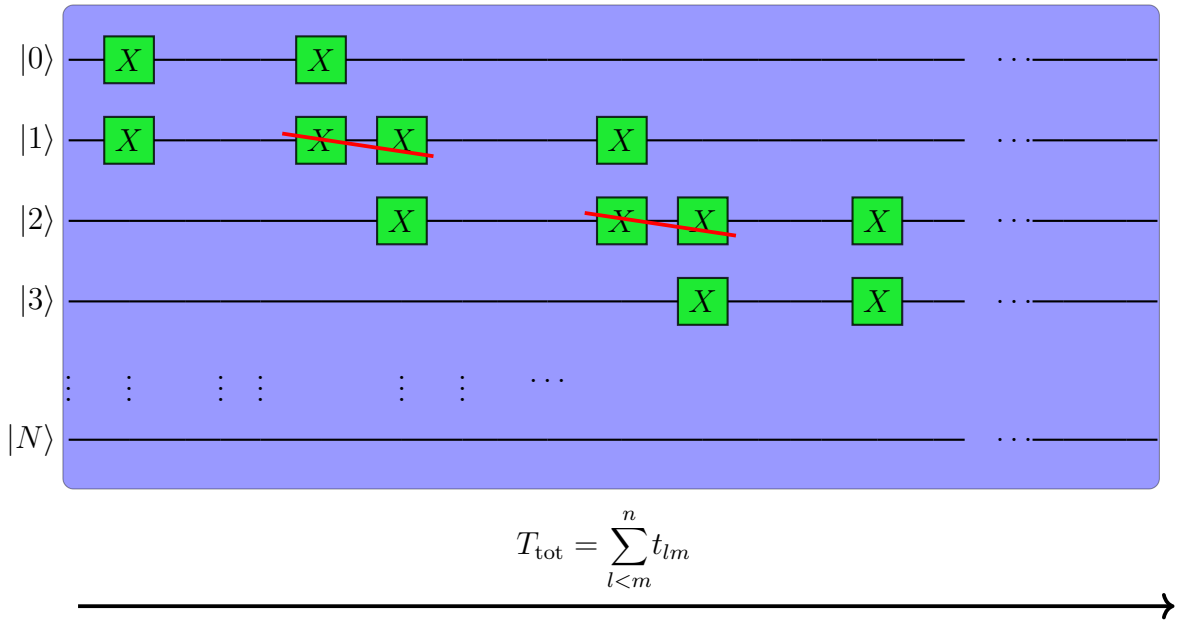


Figure 5.2: General banged Digital Analog Quantum Computing(bDAQC): The blue gate is the resource Hamiltonian which is turned on for the total time T_{tot} (bDAQC). The single-qubit X rotations are applied simultaneously at certain time points.

that case, the bDAQC scheme is more error robust than sDAQC concerning cross-talk due to the less needed control ability of the device Hamiltonian. In the following, we designate the pure analog bDQAC unitary as $\hat{U}_{ZZ}$, and $\hat{U}_{ZZ+X}$ represents the unitary that incorporates digital single-qubit X-gates alongside its analog component.

5.3 QAOA in the DAQC scheme

This section demonstrates the integration of QAOA into the digital-analog approach as a primary focus. Subsequently, we evaluate the performance of QAOA in the bDAQC scheme compared to its pure digital counterpart. Additionally, we provide an analysis of the computational time required for both the digital and digital-analog approach of QAOA.

5.3.1 Apply DAQC to QAOA

This subsection focuses on the workings of a potential QAOA device in the digital-analog scheme. To simplify matters, we concentrate on problem Hamiltonians

$$\hat{H}_{\text{arb}} = \sum_{ij}(\mathbb{1} - Z_i Z_j). \tag{5.3.1}$$

The problem graphs we investigate are random non-regular graphs. However, it is worth noting that other problems involving Z_iZ_j interactions, such as 2SAT, can also be explored. Theoretically, problems like Max-kXOR with higher degrees than 2 for k can be applied to the device Hamiltonian. After, we map to the device Hamiltonian. A standard device Hamiltonian for this type of problem may resemble

$$\hat{H}_{\text{device}} = \theta(t)\sum_{i<j}^{n} Z_i Z_j + \alpha \sum_{i=0}^{n} \phi_i(t) X_i. \tag{5.3.2}$$

The parameter α corresponds to the coupling strength and determines the gate duration of a Pauli-X rotation. By summing up the two-qubit ZZ interaction terms, we can construct the entangling device Hamiltonian, which features a homogeneous coupling. Additionally, we use a step function θ with $\theta \in \{0, 1\}$ to enable sudden activation and deactivation of the coupling.

In addition, we require single-qubit Pauli-X gates with a step-function ϕ_i for each qubit, with $\phi_i \in \{0, 1\}$. With digital and analog control components, we aim to construct a QAOA device tailored for solving the Max-cut problem. While we could adjust the number of one-qubit gates available to make the device more versatile for various problems, this is not the primary goal of our research. Rather, we aim to create NISQ algorithms, such as QAOA, using a digital-analog device suitable for this purpose. To employ a machine that meets these requirements, we must activate and deactivate the device Hamiltonian using θ in the sDAQC scheme. However, in the bDAQC scheme, we can relinquish even more control by enabling the device Hamiltonian at the initial time $t = 0$ and switching it off at the final time t_F using $\theta(t_F - t)$. Compared to other algorithms like QFT in the digital-analog scheme [133], where the $\exp(-i\gamma ZZ)$ incorporation must be performed, the QAOA is already formulated in terms of these rotations for the two-qubit gates, giving it an advantage. As a result, we can employ the mapping from Equation (5.2.8) to determine the total time required in the bDAQC scheme. To calculate the overall time in the bDAQC scheme, we only need to multiply the times by the variational parameter γ, leading to

$$\sum_{d=1}^{p} \gamma_d t_\omega = \sum_{d=1}^{p} \gamma_d M^{-1}_{\omega\zeta} g_\zeta t_F. \tag{5.3.3}$$

We only need to perform the matrix inversion for $p = 1$ and can use it for higher orders of p. Therefore, we will focus on $p = 1$ in our upcoming simulations. For the single-qubit rotations of the device Hamiltonian $\hat{H}_{\text{device}}$, we have to set all $\phi_i = 1$ instantaneously and let the device Hamiltonian to be on all the time. The gate rotation time for the driver Hamiltonian is β/α. We need to include an extra block $\lambda t_{\min}\vec{1}$ between the γt_ω and the driver block. Ultimately, we need to demonstrate that the Trotter error in the middle

analog block of the γ decomposition

$$\hat{U}_{ZZ}(t_\omega)\hat{U}_{X+ZZ}(1/\alpha)\hat{U}_{ZZ}(t_{\omega+1}) = \exp\left(i\left(t_\omega\gamma - \frac{1}{\alpha}\right)\sum_{i<j} Z_iZ_j\right) \times \exp\left(\frac{i}{\alpha}\left(\alpha(X_l + X_m + X_{l+1} + X_{m+1}) + \sum_{i<j} Z_iZ_j\right)\right)\exp\left(i\left(t_{\omega+1}\gamma - \frac{1}{\alpha}\right)\sum_{i<j} Z_iZ_j\right), \tag{5.3.4}$$

does not lead to a high difference in $\mathcal{M}_1$ compared to the QAOA in the DQC scheme. Of course, this depends on the coupling strength α. Equation (5.3.4) shows the implementation of a random bDAQC block. Generally speaking, we subtract $1/\alpha$ of the device Hamiltonian before and after the single-qubit rotations and add $1/\alpha$ to the single-qubit X rotations. Subtracting $3/(2\alpha)$ of the device Hamiltonian for the analog blocks at the beginning and the end, and $1/\alpha$ for the analog blocks in between, is also necessary. To simulate this process, we need to Trotterize the block $\hat{U}_{X+ZZ}(1/\alpha)$. We distinguish here between the computational Trotter error due to the simulations on the computer and the Trotter error due to the bDAQC scheme.

For the driver block, we also have to subtract infinitesimal time from the device Hamiltonian

$$\begin{aligned}&\hat{U}_{ZZ}(\lambda t_{\min})\hat{U}(\hat{B} + \hat{H}_{\text{device}}, \beta/\alpha) \\ &= \exp\left(i\left(\lambda t_{\min}\gamma - \frac{\beta}{\alpha}\right)\sum_{k<j} Z_iZ_j\right)\exp\left(\frac{i\beta}{\alpha}\left(\sum_{k<j} Z_iZ_j + \alpha\sum_i X_i\right)\right).\end{aligned} \tag{5.3.5}$$

We need to combine all the device's Hamiltonian ingredients to describe the system's evolution. This includes turning on the single-qubit rotations of the drive for a duration of β/α. However, we also need to subtract the time taken by the extra analog block, which is represented by $\lambda t_{\min}\vec{1}$. After this step, we arrive at the final unitary operator that fully characterizes the system's evolution

$$\hat{U}_{\text{device}}(t) = \mathcal{T}\exp\left(\int_0^{t_F} \hat{H}_{\text{device}}(t)dt\right), \tag{5.3.6}$$

with $\mathcal{T}$ being the time ordering operator, we must include because of the time dependence of the device Hamiltonian.

As we have already mentioned, implementing a device with the digital-analog scheme can reduce errors caused by cross-talk. Moreover, the variational nature of the QAOA, compared to non-variational algorithms, provides coherent error reduction, which is an added advantage. This combination of NISQ and DAQC concepts could pave the way for a new era of devices before fault-tolerant quantum computers can be built.

5.3.2 Performance of bDAQC-QAOA versus QAOA

This section focuses on examining how well the bDAQC scheme performs with respect to the QAOA. In the preceding section, we integrated all components necessary to implement the bDAQC scheme for QAOA on a single device. As a result, we formulate the QAOA

ansatz for the bDAQC scheme with $p = 1$ as follows

$$\begin{aligned}\hat{U}_{\mathrm{DA}}(\alpha)\,|s\rangle =& \exp\left(\left(\lambda t_{\min}\gamma - \frac{\beta_1}{\alpha}\right)\hat{H}_{\mathrm{res}}\right)\exp\left(\frac{\beta_1}{\alpha}\left(\hat{H}_{\mathrm{res}} + \alpha\sum_i X_i\right)\right)\\ &\times \prod_{l<m}^{n}\exp\left(\frac{1}{2\alpha}(X_l + X_m)\hat{H}_{\mathrm{res}}\right)\\ &\times \exp\left(\left(\gamma_1 t_{lm} - \frac{1}{\alpha}\right)\hat{H}_{\mathrm{res}}\right)\\ &\times \exp\left(\frac{1}{2\alpha}(X_l + X_m)\hat{H}_{\mathrm{res}}\right)|s\rangle\end{aligned} \tag{5.3.7}$$

and calculate the loss function. After the optimization, we first calculate the expectation value with the optimal parameters

$$\langle\hat{H}\rangle^{\mathrm{DA}}_{\vec{\gamma}^*_{\mathrm{DA}},\vec{\beta}^*_{\mathrm{DA}}} = \langle s|\,\hat{U}_{\mathrm{DA}}(\alpha)\hat{H}\hat{U}_{\mathrm{DA}}(\alpha)^\dagger\,|s\rangle\,. \tag{5.3.8}$$

After normalization with the maximum value of the problem Hamiltonian $E_{\max}$, we obtain the mean approximation ratio

$$\mathcal{M}_1^{\mathrm{DA}} = \frac{\langle\hat{H}\rangle^{\mathrm{DA}}_{\vec{\gamma}^*_{\mathrm{DA}},\vec{\beta}^*_{\mathrm{DA}}} - E_{\min}}{E_{\max} - E_{\min}}. \tag{5.3.9}$$

Subsequently, we compare the mean approximation ratio of the bDAQC-QAOA scheme with the pure DQC-QAOA mean approximation ratio. The difference in the mean approximation ratios is

$$\Delta\mathcal{M}_1 = \mathcal{M}_1 - \mathcal{M}_1^{\mathrm{DA}}. \tag{5.3.10}$$

As mentioned in the prior subsection, we study the speed ratio α and examine the limit when bdaqc-QAOA delivers comparable results to DQC-QAOA mean approximation ratio. As the problem Hamiltonian $\hat{H}_{\mathrm{arb}}$, we use non-regular Max-cut instances of the Erdős–Rényi model for random graphs, where the probability of having a connection between two vertices is 70%.

Figure 5.3 shows the results for qubit numbers ranging from 8 to 25 and speed ratios ranging from 1 to 10000. The green color indicates that the DQC-QAOA outperforms the bDAQC approach, while the brown color indicates the opposite. The white color represents a neutral regime. We observe that in the low-speed ratio regime (1-10) and low qubit numbers (8-10), the DQC-QAOA outperforms the bDAQC-QAOA by approximately 7.5%. This result is caused by a high Trotter error, which is worst when the coupling strength is $\alpha = 1$ in the bDAQC-QAOA scheme. In scenarios involving low-speed ratios and low qubit numbers, the bDAQC scheme can no longer encode the problem Hamiltonian, resulting in an error equivalent to the result of a depolarizing error regarding the parameter space. This error affects the parameter space. As a result, the parameter space becomes flat with unstable regions. The mean approximation ratio calculation ensures that the percentage difference in the worst-case scenario of low-speed ratio α does not exceed 12.5%. As the Trotter error increases, local minima and maxima values will decrease or increase until the parameter space becomes flat. The flat space's expectation value is close to the global minimum. Whereas the overlap of both approaches goes down to zero. Furthermore, current quantum devices can execute two and one-qubit gates at

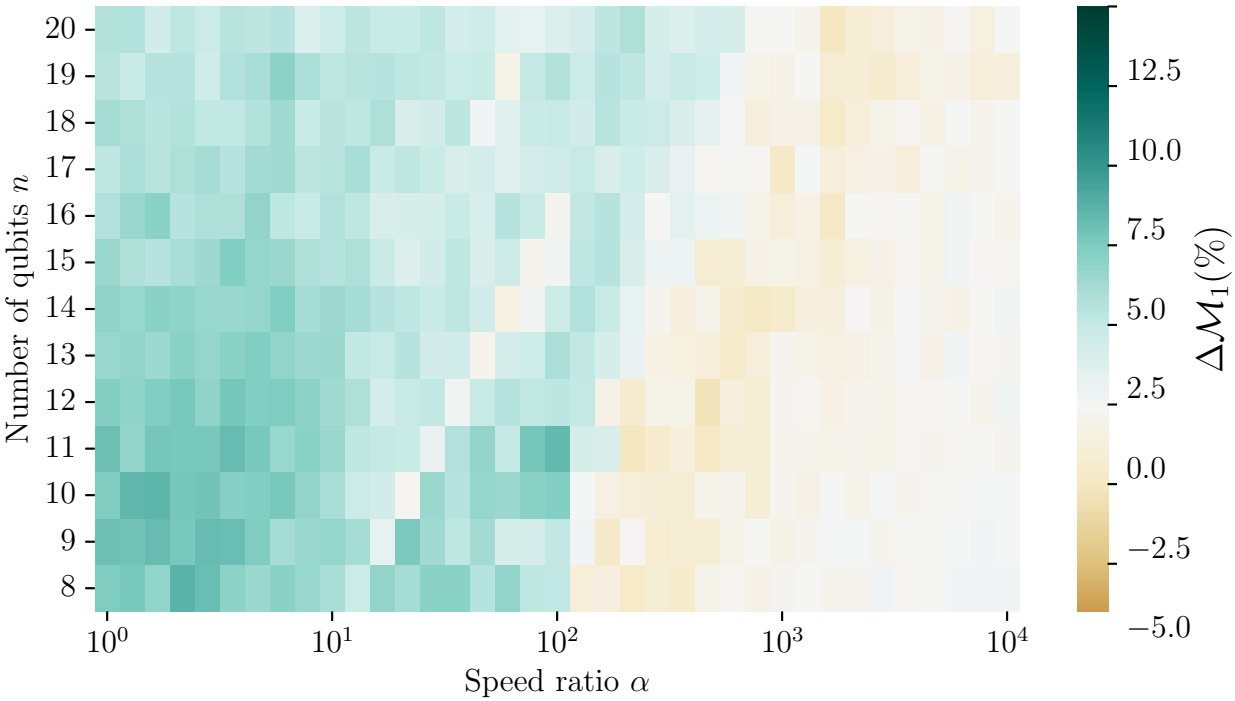


Figure 5.3: The color map indicates the percentage difference in mean approximation ratio between pure digital QAOA and bDAQC-QAOA. According to Equation (5.3.10), the difference is $\Delta\mathcal{M}_1$. The x-axis indicates the speed ratio α of the single-qubit rotations. The y-axis displays the number of qubits per instance. We take random non-regular Max-cut instances with a filling factor of 70%. We average over 50 instances per speed ratio α and per qubit number n.

different speeds. However, we notice a slight improvement for bDAQC-QAOA at higher qubit numbers ($n > 14$) when the speed ratio is the same. The QAOA approach fails to optimize due to a more complex parameter space.

Increasing α to 10^2 reduces the Trotter error in the bDAQC ansatz, resulting in a decrease in the mean approximation ratio from 7.5% to 2.5%. This reflects the current hardware's order of magnitude for α. In the range of $\alpha \in [10^2, 10^3]$, we see even an advantage of the bDAQC-QAOA over the DQC-QAOA scheme. As the qubit number increases, the advantage of bDAQC-QAOA over the digital QAOA appears to shift towards higher α values. This cannot be explained purely by the translation from DQC to bDAQC. In literature, there are examples of how noise could even be beneficial for variational algorithms [48]. In such cases, errors can be interpreted as Pauli errors and could potentially lead to an improvement, as demonstrated in [48]. Thus, a detailed error analysis comparing DQC and bDAQC devices would be necessary, but this is not the subject of this research. If we increase α to an order of 10^4, we reach the limit where the Trotter error in the bDAQC scheme becomes negligible. In this limit, DQC and bDAQC are identical under error-free devices.

Suppose current hardware providers like Google and IBM could bring all qubits into resonance. In that case, we could easily measure if we have an advantage of DAQC over DQC due to fewer cross-talk errors. This we should do for low time decomposition t_{nm}, to stay in coherence time.

5.3.3 Parameter space of bDAQC-QAOA

In this subsection, we investigate the parameter space for bDAQC-QAOA discussed in the previous subsection.

Figure 5.4 shows one instance's expectation values for two different values of $\alpha \in \{10, 1000\}$ in the parameter space. The expectation value F of the bDAQC approach changes to the Trotter error. As expected, the left plot indicates that at a high-speed ratio of $\alpha = 1000$, the values of F_1 for both approaches are almost identical. However, for $\alpha = 10$, we notice that the Trotter error introduces a reduction in the global minimum. Additionally, the solution space presents more small local minima than the DQC-QAOA approach, making the optimization process even more challenging. Although the optimal parameter distances, γ^* and β^*, between bDAQC and DQC increase from 0.1 a.u. for $\alpha = 1000$ to 0.15 a.u. for $\alpha = 10$, we cannot generalize that the distance increases as α decreases. The coherent error appears random among the instances, making it difficult to predict.

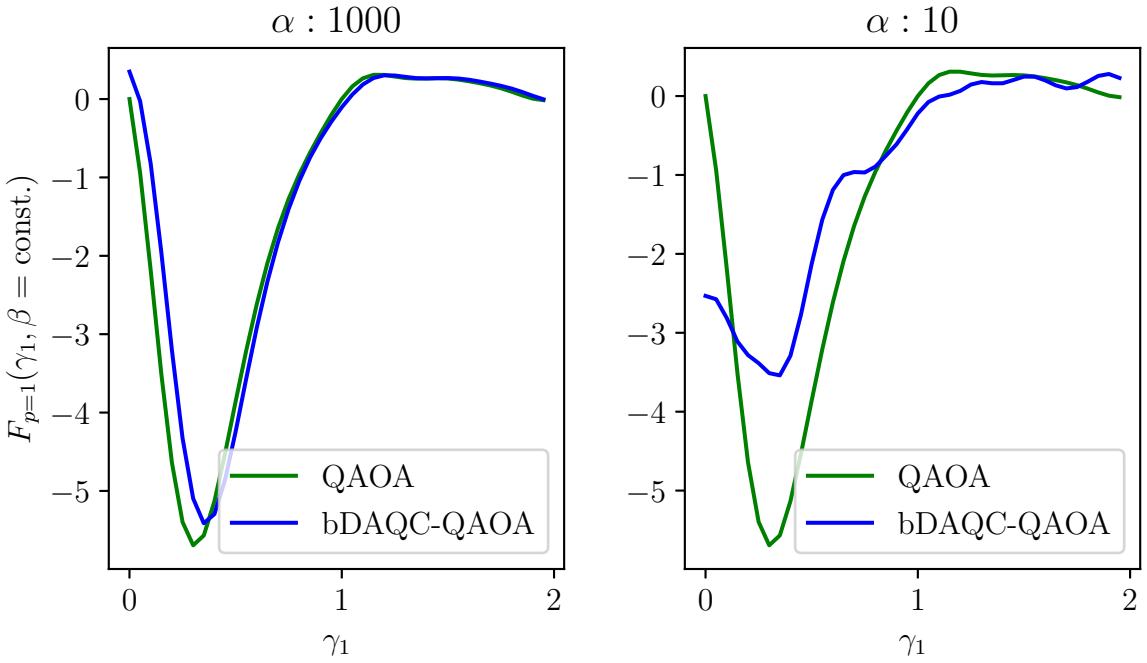


Figure 5.4: Plotting expectation value F_1 for fixed β_1 against γ_1 for $p = 1$ and for DQC and bDAQC. For different values of α. For both plots, we take the same problem of a random problem graph of size $N = 8$. The green curve is the expectation value of QAOA in DQC, and the blue curve is the expectation value of QAOA in bDAQC. Left plot: we take a speed ratio of $\alpha = 10$. We fixed $\beta_1^{\text{DQC}} = 0.65$ and $\beta_1^{\text{bDAQC}} = 0.5$ at the global minimum of both approaches. Therefore, the distance due to the coherent error of the global minimum is $\Delta(\gamma_1^*, \beta_1^*)^{\text{DQC,bDAQC}} = 0.15$ a.u. Right plot: we take speed ratio of $\alpha = 1000$. β_1 is fixed to $\beta_1^{\text{DQC}} = 0.65$ and $\beta_1^{\text{bDAQC}} = 1.65$. Hence the distance is $\Delta(\gamma_1^*, \beta_1^*)^{\text{DQC,bDAQC}} = 0.1$ a.u. due to the coherent error.

5.3.4 Variational resilience

Unlike the HHL algorithm [137] or the QFT [133] in the DAQC scheme, the QAOA has an additional feature, the optimizer, which reduces the coherent error in the approach. This makes the QAOA, in combination with the DAQC scheme, particularly appealing for the NISQ era. In this section, we investigate the role of the optimizer in the bDAQC-QAOA scheme. The QAOA leverages the expectation value as the loss function, allowing the optimizer to determine the optimal angles $\vec{\gamma}^*, \vec{\beta}^*$. We utilize the identical approach to find the optimal angles $\vec{\gamma}^*_{\mathrm{DA}}, \vec{\beta}^*_{\mathrm{DA}}$ of $\langle \hat{H} \rangle^{\mathrm{DA}}_{\vec{\gamma}^*_{\mathrm{DA}}, \vec{\beta}^*_{\mathrm{DA}}}$ regarding the expectation value of Equation (5.3.8). An open question remains regarding at which α the optimal parameters $\vec{\gamma}^*, \vec{\beta}^*$ of the QAOA match those of the bDAQC-QAOA ansatz. Furthermore, if there are differences in the optimal angles between the two approaches, how can the optimizer mitigate the Trotter error and find new parameters for the DA approach?

We analyze Figure 5.5 to explore this question. In particular, we utilize the optimal

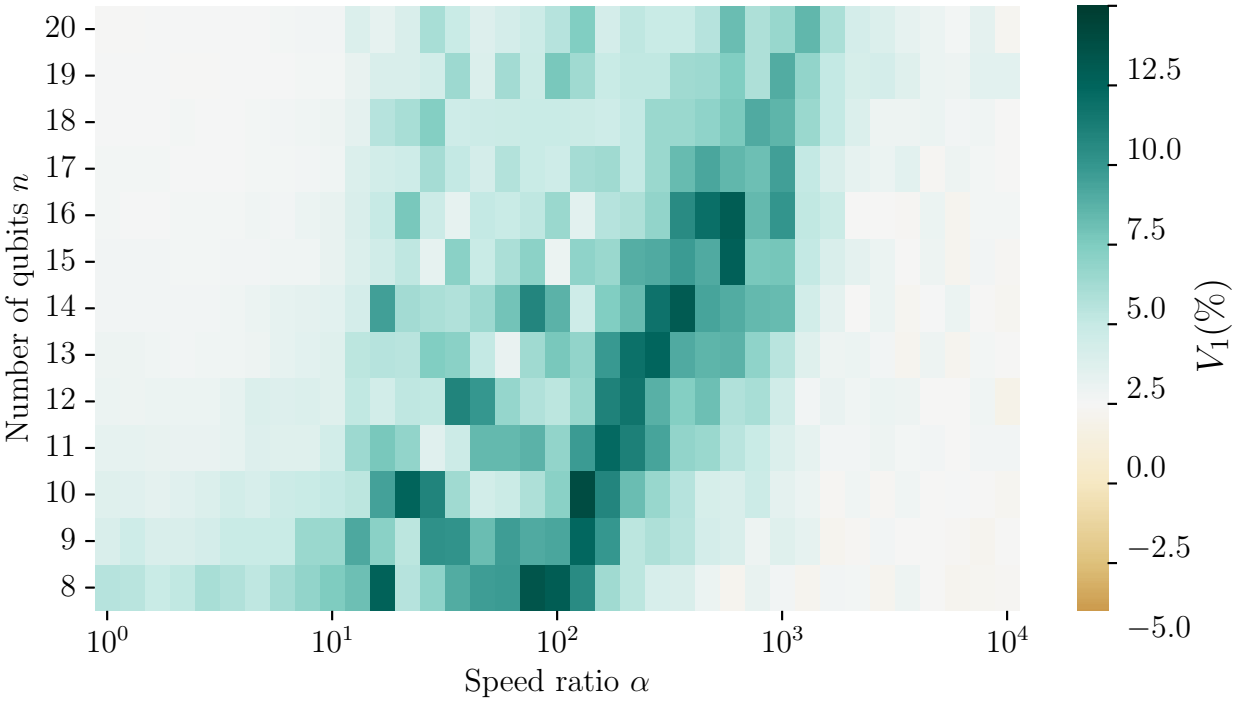


Figure 5.5: The color map indicates the percentage difference in mean approximation ratio between bDAQC-QAOA with optimized parameters $\vec{\gamma}^*_{\mathrm{DA}}, \vec{\beta}^*_{\mathrm{DA}}$ and unoptimized bDAQC-QAOA using the parameters $\vec{\gamma}^*, \vec{\beta}^*$ from QAOA. According to Equation (5.3.11), the difference is V_1. The x-axis indicates the speed ratio α of the single-qubit rotations. The y-axis indicates the number of qubits per instance. We take random non-regular Max-cut instances with a filling factor of 70%. We average over 50 instances per speed ratio α and per qubit number n.

parameters $\vec{\gamma}^*, \vec{\beta}^*$ obtained from QAOA and substitute them into the non-variational form of the bDAQC-QAOA expectation value in Equation (5.3.8). We then calculate the mean approximation ratio and subtract this value from the optimized bDAQC-QAOA

mean approximation ratio in variational form, leading to

$$V_1 = \frac{\langle \hat{H} \rangle^{\mathrm{DA}}_{\vec{\gamma}^*_{\mathrm{DA}},\vec{\beta}^*_{\mathrm{DA}}} - \langle \hat{H} \rangle^{\mathrm{DA}}_{\vec{\gamma}^*,\vec{\beta}^*}}{E_{\max} - E_{\min}}. \tag{5.3.11}$$

For high α values, there is no difference because both approaches of DQC and bDAQC are identical. Consequently, when the optimal angles are the same, the variational and non-variational forms of bDAQC-QAOA yield no distinction.

However, at lower speed ratios, ranging from $\alpha \in [10^1, 10^2]$, the difference between the two approaches achieves the maximum with $\Delta V_1 \geqslant 10\%$. In this range, the variational bDAQC-QAOA scheme outperforms the non-variational form regarding the mean approximation ratio, and the optimal angles differ. The introduced Trotter error α in the bDAQC approach makes the non-variational form of bDAQC-QAOA ineffective. Nevertheless, the optimizer can mitigate the error in the bDAQC approach.

When α is in the range of $\alpha \in [1, 10]$, as mentioned in the preceding section, the bDAQC-QAOA approach fails to encode the problem. The surface is almost flat in this range, and the variational form cannot absorb the Trotter error. In this specific range, the bDAQC-QAOA circuit encounters a phenomenon known as a barren plateau, where the gradient tends to vanish [138]. The occurrence of significant noise in the system [139] can contribute to the emergence of this phenomenon. In such cases, the optimizer will not help, and if we insert the optimal parameters of the QAOA in the bDQAC scheme, we will get two different optimal points with the same value. Figure 5.5 defines when to use the variational form or the fixed number of the gate sequence.

5.3.5 Time duration of DQC and DAQC

In order to compare the QAOA on a DQC and a DAQC device, we must determine the time required for both algorithms. Therefore, we conducted a time analysis for each approach without incorporating any assumptions about errors in the two-qubit gates or any factors that may limit coherence time.

However, to conduct a more detailed analysis, we would need to examine specific hardware components, such as Pauli errors, cross-talk, and the latency of a two-qubit gate. To ensure a fair comparison, we assume the digital approach possesses an all-to-all connected hardware graph, enabling ZZ interactions without needing SWAP gates. However, the digital approach cannot apply all two-qubit gates associated with a single qubit on the hardware graph simultaneously. Therefore, we make a straightforward approach that the time required for a problem Hamiltonian is

$$T_{\mathrm{tot,DQC}} = \max_{v \in V}(\deg(v)) + 1. \tag{5.3.12}$$

The graph $G(V, E)$ represents the problem Hamiltonian with E the set of all edges and V the set of all vertices. deg is the degree of a vertex v, so the number of edges connected to the vertex. The graph's max degree is the vertex with the most connections. This is the maximum time interval we have for a digital version. We set the time for a single interval to one without considering any quantum compiling improvements that could potentially reduce circuit depth as demonstrated in [140, 141]. To obtain the total time for DAQC, we sum up all time intervals t_{nm} and add the extra analog block of $\lambda t_{\min}$.

Figure 5.7 depicts an example problem graph, including its decomposition into DQC

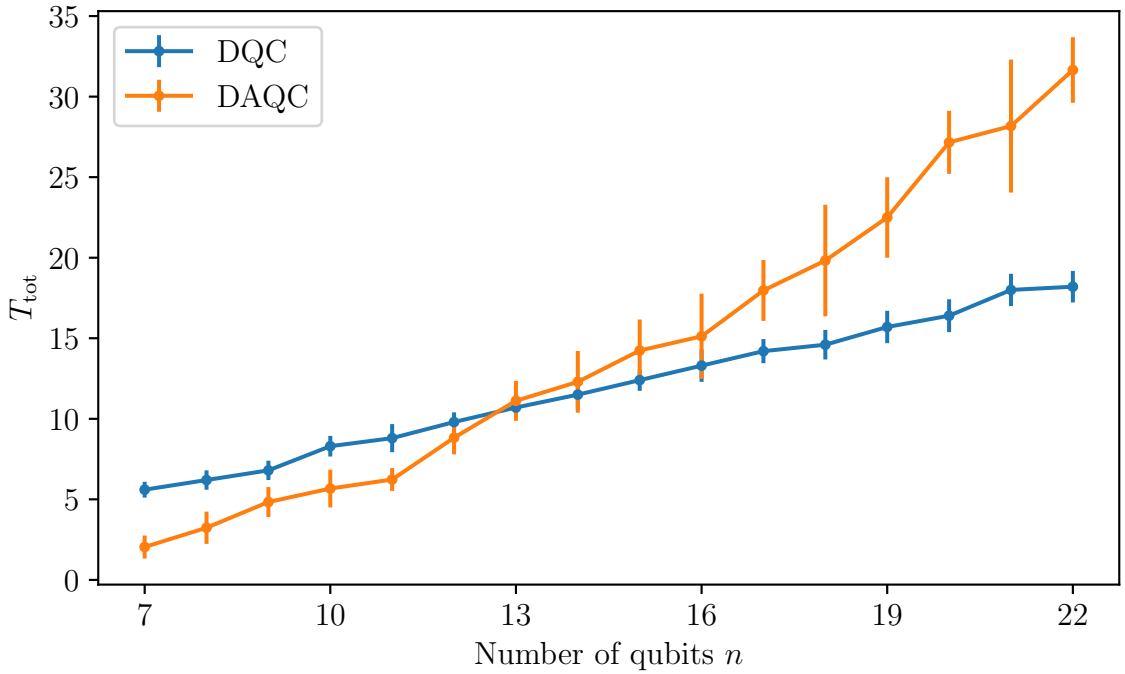


Figure 5.6: Plotting the number of qubits against the total time required for digital quantum computing (orange curve) and digital-analog quantum computing (blue curve). The x-axis indicates the number of qubits for a random Max-cut graph with a filling factor of 70% for the Erdős–Rényi graph. The y-axis indicates the time the protocols need. We average over 100 random instances per number of qubits. The resource Hamiltonian for DAQC is set to be homogeneous.

and DAQC. In this example, the max degree of the graph is equal to the total time in the DQC, expressed as $T_{\text{tot,DQC}} = \max_{v\in V}(\deg(v)) \neq \max_{v\in V}(\deg(v)) + 1$. The distinction is due to the non-regular problem graph. In regular graphs, the time interval is primarily according to Equation (5.3.12).

Because we are looking for bDAQC decomposition, we include all single-qubit gates, which take time Δt into the analog blocks. Thus, Δt does not account for the time calculation. Figure 5.6 shows the results of this study. The DQC protocol demonstrates a linear trend in dependence on the qubit number. The DAQC protocol outperforms DQC in total time T_{tot} for $n \leqslant 13$. For $n > 13$, the DAQC scheme requires more time T_{tot} than DQC. Estimating the total time for the two QAOA protocols helps us to determine the appropriate circuit depth for running QAOA using the DAQC scheme. This is particularly important for short circuit depths compatible with the available qubits on hardware platforms in the NISQ era.

It should be noted that we have not accounted for any inhomogeneous coupling in our simulations. Introducing inhomogeneous coupling to the resource Hamiltonian would result in a robust exponential increase in the total time for the DAQC protocol shown in Figure 5.6. Consequently, the advantage of DAQC over DQC would shift to a lower qubit number. An even more significant challenge is not having an all-to-all connected resource Hamiltonian. In these cases, we must incorporate the swap gates into the analog time evolution. This would significantly increase the total time required for the DAQC protocol compared to the DQC protocol, which requires only swap gates. The total time calculation is not an algorithm-independent comparison between DQC and DAQC.

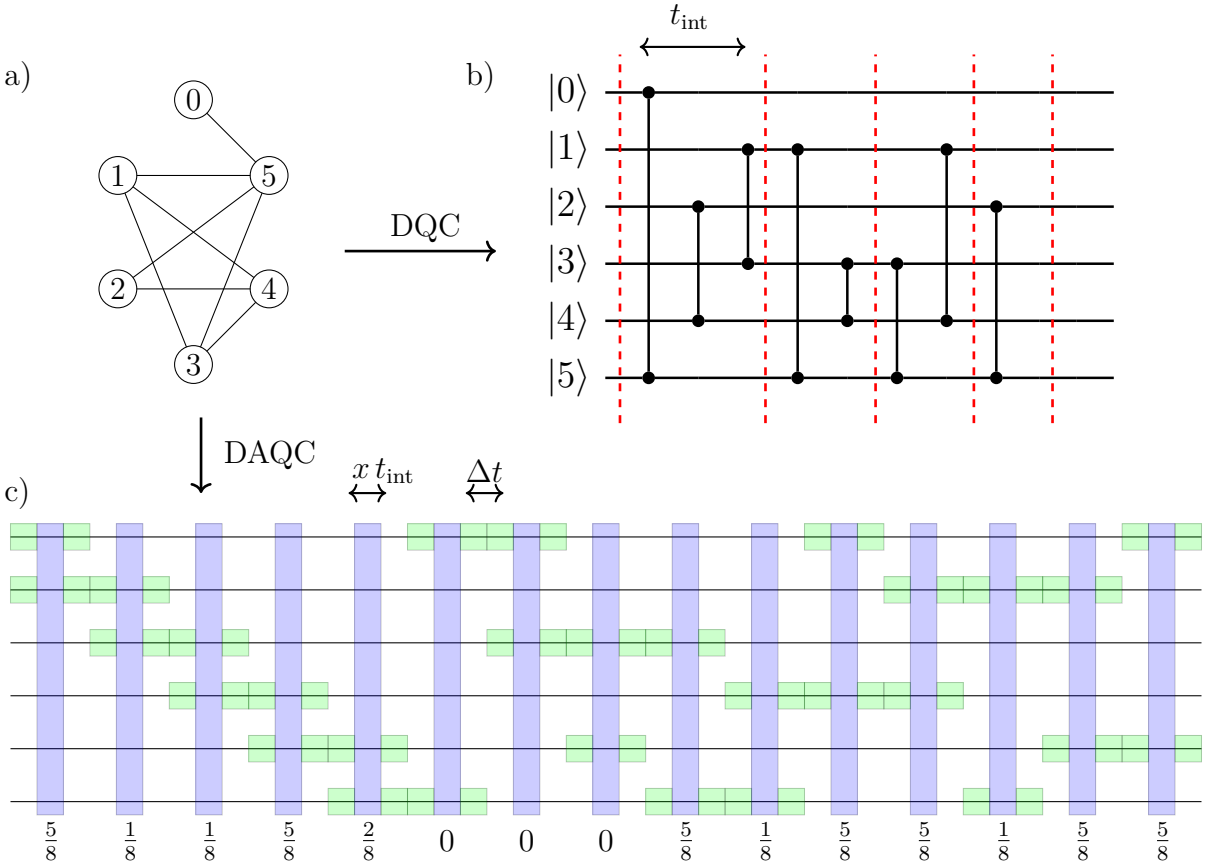


Figure 5.7: Compiling DAQC and DQC for an arbitrary problem Hamiltonian. a) depicted is a non-regular problem graph of qubit size 6 with equal weights per edge with a number of edges $|E| = 8$ (Erdős–Rényi graph with filling factor 60 %). b) shows the compiling of a) into DQC into 4 t_{int} time intervals with $\max(\deg(v) = 4)$. c) shows the decomposition of a) into DAQC into 15 time intervals of length xt_{int}. Blue are the analog blocks, and green are the single-qubit X rotation blocks. The value X of each time interval is written below the respected blue analog block. Δt is the time for the X gates. The extra analog block without single-qubit X gates $\lambda t_{\min}$ is not depicted. Overall, we have as total time for bDAQC: $T_{\text{tot}} = (41/8)t_{\text{int}}$ (after X gate cancellation + 3 zero time intervals + extra analog block). We will include the Δt in the analog time block due to the bDAQC scheme. In this specific example, DQC is faster than bDAQC by 1.125 t_{int}.

When comparing DQC and DAQC protocols of QFT [133], the time period for the DAQC protocols is shorter for high qubit numbers.

5.3.6 Minimum state fidelity

So far, we have presented numerical results for the Trotter error between QAOA in bDAQC and the DQC scheme. In this section, we will examine an analytical approach to establishing the minimum state fidelity between the two approaches. Therefore, we first estimate the Trotter error between both unitaries and combine this with the state fidelity. David

Headley proved this in Phys. Rev. A 106, 042446 (2022) . I will rephrase the main results in this section.

The structure of our problem is reflected in the second-order Trotterization [142, 143]

$$\Big\|\exp(iA/2)\exp(iB)\exp(iA/2)\Big\| \leqslant \frac{1}{12}\Big\|[[A,B],B]\Big\| + \frac{1}{24}\Big\|[[A,B],B]\Big\|. \tag{5.3.13}$$

We will now shift our attention to the device Hamiltonian, specifically when we activate the single-qubit X rotations

$$\hat{H} = \alpha \sum_{m\in M_\mu} X_m + \sum_{j<k} Z_j Z_k. \tag{5.3.14}$$

We will apply this Hamiltonian for time Δt between the two analog blocks. Assuming a homogeneous resource Hamiltonian and M_μ representing a different set of single-qubit rotations, we have three options for M_μ. Generally speaking, we can apply this bound at most $n(n-1)/2+2$ times. Specifically, we have $n(n-1)/2$ analog blocks following matrix inversion, one extra block to avoid negative times, and the X-driver field from the QAOA ansatz. Neglecting cancellation, we have $(n(n-1)/2)-1$ bounds with $|S_\mu| = 4$ elements, two bounds with $|S_\mu| = 2$ elements, and one bound, which is the driver block, where we apply a total of $|S_\mu| = n$ single-qubit rotations. Additionally, for the driver bound, we use $t = \beta/\alpha$ for Δt with

$$\hat{A} = t\sum_{j<k} Z_j Z_k,\ \hat{B} = t\alpha \sum_{m\in M_\mu} X_m. \tag{5.3.15}$$

We can now express the Trotter error bound for Equation (5.3.13)

$$\begin{aligned}
&\Bigg\|\exp\left(it\sum_{j<k} Z_jZ_k\right)\exp\left(it\alpha\sum_{m\in M_\mu} X_m\right)\exp\left(it\sum_{j<k} Z_jZ_k\right)\\
&-\exp\left(it\left(\alpha\sum_{m\in M_\mu} X_m + \sum_{j<k} Z_jZ_k\right)\right)\Bigg\| \leqslant \frac{\alpha^2t^3}{12}\left\|\left[\left[\sum_{j<k} Z_jZ_k, \sum_{m\in M_\mu} X_m\right], \sum_{m\in M_\mu} X_m\right]\right\|\\
&+\frac{\alpha t^3}{24}\left\|\left[\left[\sum_{j<k} Z_jZ_k, \sum_{m\in M_\mu} X_m\right], \sum_{j<k} Z_jZ_k\right]\right\|.
\end{aligned} \tag{5.3.16}$$

We now calculate the first commutator on the RHS

$$\begin{aligned}
&\sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j<k}[[Z_jZ_k, X_m], X_{m'}]\\
&= \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j<k}(2(\delta_{jm}-1)+2(\delta_{km}-1))[Z_jZ_kX_m, X_{m'}]\\
&= \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{k>m} 2[Z_mZ_kX_m, X_{m'}] + \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j<m} 2[Z_jZ_mX_m, X_{m'}]\\
&= \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j\neq m} 2[Z_jZ_mX_m, X_{m'}] = \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j\neq m} 2(\delta_{jm}+\delta_{jm'}+\delta_{mm'})[Z_jZ_mX_m, X_{m'}]\\
&= \sum_{m\in M_\mu}\sum_{m'\in M_\mu}\sum_{j\neq m} 2\Big([Z_jZ_mX_m, X_{m'}]\delta_{m'm} + [Z_jZ_mX_m, X_{m'}]\delta_{jm}(1-\delta_{m'm})\Big)
\end{aligned}$$

$$= \sum_{m \in M_\mu} \Big(\sum_{j \neq m} 4 Z_m Z_j + \sum_{m' \in M_\mu | m' \neq m} 4 i Y_m Y_{m'} \Big). \tag{5.3.17}$$

To begin, we extract the commutator sums and then evaluate the inner commutator $[Z_j Z_k, X_m]$. When $j = m$ or $k = m$, the commutator vanishes. For the non-commuting terms, we utilize $XZ = -ZX$ to obtain $2 Z_j Z_m X_m$ per bracket. We then split both Kronecker deltas into two sums, the first for higher values of k concerning m and the second for lower values of j with respect to m. This is possible because $j \neq k$. Next, we eliminate one of the two indices j, k. Since both terms yield the same value, we can rewrite it as $\sum_{k>m} a + \sum_{j<m} a = \sum_{j \neq m} a$. In the following equation, we discard every commutator for which all three indices j, m, and m' differ. We need this to understand the subsequent line. We then distinguish the indices and count how many are identical. We have one term where all three indices are identical and another term where we have two pairs of indices. The final equation in (5.3.17) is derived by commuting through the elements. In the first sum, we commute X through, cancel it out, and then obtain a minus sign, which multiplies the commutator by two. Through $\delta m' m$, the sum over m' vanishes as well. The second sum follows the same procedure, but instead of canceling out, we multiply to obtain $ZX = iY$. Additionally, through δ_{jm}, we cancel out the sum over j and must exclude m from m' in the sum.

In the next step, we estimate the operator norm of this commutator. The operator norm estimates how much the unitary matrices differ from each other, which we determine through the greatest eigenvalue. For commuting matrices, we can use the following theorem.

Theorem 5.3.1. *Let A be a matrix with spectrum $\sigma(A)$ and B a matrix with spectrum $\sigma(B)$. If and only if the matrices are commuting $[A, B] = 0$ then we can say about the spectrum of $A + B$*

$$\sigma(A + B) := \lambda_k \in \{\lambda_i + \lambda_j, \lambda_i \in \sigma(A), \lambda_j \in \sigma(B)\}, \quad \forall k. \tag{5.3.18}$$

Then we can say that the Matrices A and B are simultaneously diagonalizable with

$$P(A + B)P^{-1} = PAP^{-1} + PBP^{-1}, \tag{5.3.19}$$

with P being the transformation matrix of the map and P^{-1} the inverse of P.

The last line of Equation (5.3.17) presents two sums, one for $Z_m Z_j$ and another for $i Y_m Y_{m'}$. In the worst-case scenario, these sums add linearly as an upper bound for the greatest eigenvalue. This is valid for terms of the form $(Z_m Z_j + i Y_m Y_{m'}) \delta_{jm'}$, which commute and fulfill the previous theorem. If non-commuting terms are added to the sum, such as $(Z_m Z_j + i Y_m Y_{m'})(1 - \delta_{jm'})$, the eigenvalues will be lower. The linear addition of the eigenvalues provides the best upper bound for the operator norm. We must now collect all eigenvalues and determine their length. For a given S_μ, there are s possible indices, leading to $s(s-1)YY$ and $s(n-1)ZZ$ contributions to the greatest eigenvalue. We can rewrite Equation (5.3.17) using the norm to $s\sqrt{(s-1)^2 + (n-1)^2}$. Consequently, we estimate the bound of the commutator to

$$\frac{\alpha^2 t^3 s \sqrt{(s-1)^2 + (n-1)^2}}{3}. \tag{5.3.20}$$

We now investigate the second commutator on the RHS of Equation (5.3.16)

$$\begin{aligned}
&\sum_{m\in M_\mu}\sum_{j<k}\sum_{j'<k'}[[Z_jZ_k,X_m],Z_{j'}Z_{k'}] = \sum_{m\in M_\mu}\sum_{j<k}\sum_{j'<k'}[(Z_jZ_kX_m - X_mZ_jZ_k),Z_{j'}Z_{k'}]\\
&= \sum_{m\in M_\mu}\sum_{j<k}\sum_{j'<k'}2(\delta_{jm}+\delta_{km})[Z_jZ_kX_m,Z_{j'}Z_{k'}] = \sum_{m\in M_\mu}\sum_{j'<k'}\sum_{j\neq m}2Z_j[Z_mX_m,Z_{j'}Z_{k'}]\\
&= \sum_{m\in M_\mu}\sum_{j'<k'}\sum_{j\neq m}2(\delta_{j'm}+\delta_{k'm})Z_j[Z_mX_m,Z_{j'}Z_{k'}] = \sum_{m\in M_\mu}\sum_{j'\neq m}\sum_{j\neq m}-4Z_jZ_{j'}Z_mX_mZ_m\\
&= \sum_{m\in M_\mu}\sum_{j'\neq m}\sum_{j\neq m}4Z_jZ_{j'}X_m.
\end{aligned} \tag{5.3.21}$$

We begin by extracting the commutator sums and then examine the inner commutator. If the index pairs, (j,m) or (k,m), do not commute, we use -$XZ = ZX$, which expands the commutator. If all three indices differ, the inner commutator and the entire term cancel out. Next, we sum over the Kronecker deltas and apply the previous fact. When $\delta_{km}\delta_{jm}$ is true, the term cancels out, resulting in $\sum_{j<k}\to\sum_{j\neq m}$. We can then extract Z_j from the commutator, as the outer commutator consists of a ZZ-string $Z_{j'}Z_{k'}$ and commutes. We repeat this step in the following equation, as we did in the second line. If all three indices differ, everything cancels out in the commutator. As a result, we can rewrite the sum over prime indices as $\sum_{j'<k'}\to\sum_{j'\neq m}$. Finally, we can commute Z_m, which cancels out with the other. Let us now consider the upper bound of this commutator. The sums over j and j' produce a $(n-1)^2$ for each single-qubit rotation s in the set of S_μ. Therefore, we have $s(n-1)^2$ terms per S_μ.

We can estimate the upper bound using the sub-multiplicative norm

$$||\hat{A}\hat{B}|| \leqslant ||\hat{A}||\,||\hat{B}||, \tag{5.3.22}$$

which is valid for square matrices. Therefore, in the worst possible case, the greatest eigenvalue is equal to the number of Pauli strings, leading to a total bound for the Trotter error of

$$\Delta_\mu = \frac{\alpha s t^3}{3}\left(\frac{(n-1)^2}{2} + \alpha\sqrt{(s-1)^2+(n-1)^2}\right). \tag{5.3.23}$$

To establish the relationship between the Trotter bound and the minimum state fidelity between error-free DQC and bDAQC with Trotter error, we need to calculate the minimum state fidelity using the following formula

$$f_{\alpha-\text{DA-QAOA}} = \min_{\psi}\left|\left|\langle\psi|U^\dagger_{\text{QAOA}}\hat{U}_{\text{DA}}|\psi\rangle\right|\right|^2. \tag{5.3.24}$$

First, we go back to the Trotter error. We can bound this error by Δ, and subsequently, for two unitaries, we obtain

$$||\hat{U}^\dagger_{\text{QAOA}} - \hat{U}_{\text{DA}}|| \leqslant \Delta. \tag{5.3.25}$$

Using the sub-multiplicative norm, Equation (5.3.25) can be rewritten as

$$\begin{aligned}
&||\mathbb{1} - \hat{U}_{\text{QAOA}}\hat{U}_{\text{DA}}|| = ||U_{\text{QAOA}}\hat{U}^\dagger_{\text{QAOA}} - \hat{U}_{\text{QAOA}}\hat{U}_{\text{DA}}|| \leqslant\\
&||\hat{U}_{\text{QAOA}}||\,||\hat{U}^\dagger_{\text{QAOA}} - \hat{U}_{\text{DA}}|| \leqslant \Delta.
\end{aligned} \tag{5.3.26}$$

We can use the fact that the norm of a unitary operator is always $||\hat{U}_{\text{QAOA}}|| = 1$ to establish the validity of this inequality. Furthermore, the difference in norm between a unitary and the identity operator can be bounded by the maximum value of the unitary, which is given by

$$||e^{i|\theta|_{\max}} - 1|| \leqslant \Delta. \tag{5.3.27}$$

$|\theta|_{\max}$ is the greatest phase for an eigenvalue $e^{i|\theta|_{\max}}$ of $\hat{U}_{\text{QAOA}}\hat{U}_{\text{DA}}$. We now assume that all angles are small and lie in the interval $[-\pi/2, \pi/2]$. We define the greatest eigenvalue as the one that results in the most significant decrease in the Trotter error. Worth noting that the error is confined to the interval $0 < \Delta < 1$. The upcoming question is, what leads to the largest deficit in the phase causing the greatest deficit Δ. The real part of the eigenvalue in this interval is always positive. Consequently, as we approach the boundaries, this reduces Δ. Besides, the sine of the imaginary part is increased at the borders. We finally find the equation relating the Trotter borders and the greatest eigenvalue

$$\sqrt{\sin(|\theta|_{\max})^2 + (1 - \cos(|\theta|_{\max})^2)} = 2\sin\left(\frac{|\theta|_{\max}}{2}\right) = \Delta. \tag{5.3.28}$$

Through $\sin^{-1}$, we can deduce the angle

$$\theta_{\max} = 2\sin^{-1}\left(\frac{\Delta}{2}\right). \tag{5.3.29}$$

Consider the state fidelity of a single step μ in one analog block for one unitary operation

$$\left|\left|\langle\psi|\hat{O}|\psi\rangle\right|\right|^2, \tag{5.3.30}$$

with $\hat{O} = \hat{U}^{\dagger}_{\text{QAOA}}\hat{U}_{\text{DA}}$. We can further diagonalize Equation (5.3.30) to

$$\left|\left|\langle\psi'|\text{diag}\left(e^{i\vec{\theta}}\right)|\psi'\rangle\right|\right|^2, \tag{5.3.31}$$

The vector $\vec{\theta}$ includes all phases θ_i to the eigenvalues $e^{i\theta_i}$, ranging from $\theta_{\min}$ to $\theta_{\max}$. Our goal is to minimize the state fidelity, which can be achieved by preparing the state in an equal superposition of the most negative and most positive eigenvalues of the operator $\hat{O}$. To be precise, if we have the eigenvalues θ_i in an n-dimensional Hilbert space ordered from maximum to minimum value, such that $\theta_i \in \theta_{\max}, \theta_{\max}/2, \theta_{\max}/3, \ldots, \theta_{\min}$, then we can find the smallest square overlap for a state by defining

$$|\psi'\rangle = \frac{1}{\sqrt{2}}(|\theta_{\max}\rangle + |\theta_{\min}\rangle). \tag{5.3.32}$$

We can determine the minimum state fidelity by

$$\begin{aligned}\min_{\psi'}\left|\left|\langle\psi'|\mathrm{diag}\left(e^{i\vec{\theta}}\right)|\psi'\rangle\right|\right|^2 &= \frac{1}{4}\left|\left|(\langle\theta_{\max}|+\langle\theta_{\min}|)\hat{O}(|\theta_{\max}\rangle+|\theta_{\min}\rangle)\right|\right|^2 \\ &= \frac{1}{4}\left|\left|\langle\theta_{\max}|e^{i\theta_{\max}}|\theta_{\max}\rangle+\langle\theta_{\min}|e^{i\theta_{\min}}|\theta_{\min}\rangle\right|\right|^2 \\ &\geqslant \frac{1}{4}\left|e^{i|\theta|_{\max}}+e^{-i|\theta|_{\max}}\right|^2 = \cos(|\theta|_{\max})^2.\end{aligned} \tag{5.3.33}$$

Because we are in orthogonal basis, terms like $\langle\theta_{\min}|\theta_{\min}\rangle = 0$ cancel out. Therefore we can bound the fidelity of one analog block μ to

$$f_\mu \geqslant \cos(|\theta|_{\max})^2. \tag{5.3.34}$$

We can now insert the Trotter bound Δ_μ into Equation (5.3.29) and insert Equation (5.3.29) into the previous equation

$$f_\mu \geqslant \cos\left(2\sin^{-1}\left(\frac{\Delta}{2}\right)\right)^2 \geqslant 1-\Delta_\mu^2, \tag{5.3.35}$$

for small Δ. We now sum up all analog blocks and all single-qubit rotations combinations μ to accomplish

$$f_{\alpha-\text{DA-QAOA}} \geqslant 1-\sum_{\mu=1}^{n(n-1)/2+2}(1-f_\mu) = 1-\sum_{\mu=1}^{n(n-1)/2+2}\Delta_\mu^2. \tag{5.3.36}$$

5.4 Conclusion

We have demonstrated the implementation of QAOA in the digital-analog bDAQC scheme. We showed that the mean approximation ratios between the digital and bDAQC approaches are identical for high-speed ratios α for the single-qubit gates. However, for low-speed ratio α, the gate duration of two and one-qubit gates becomes equal. In this case, we observed an increase in the Trotter error in the bDAQC scheme compared to high $\alpha \gg 1$, which results in a decrease in the max and min values of the loss function $\langle\hat{H}\rangle^{\text{DA}}_{\vec{\gamma},\vec{\beta}}$. Low α is unrealistic for a real device. Single-qubit gates are always faster than two-qubit ones. Furthermore, we showed that in the mid-range of $\alpha \in [10^2, 10^3]$, we could take advantage of the Trotter error and see a benefit over the digital QAOA. Moreover, we have demonstrated that the variational approach helps in mitigating the Trotter error associated with the bDAQC-QAOA approach. Additionally, we derived an analytical bound for the minimum state fidelity of the erroneous bDAQC-QAOA ansatz in dependence of the Trotter error. We also established that the digital-analog approach takes less time for low qubit numbers than the digital approach. However, for $n \geqslant 13$, DQC-QAOA has an advantage over bDAQC-QAOA in time duration. The time analysis neglects circuit optimization and is only the first quantifier for the time comparison between both schemes.

Conclusions and Outlook

This thesis has shown different improvements made to the QAOA and provided insights into its capabilities for solving specific problems. We delved into various combinatorial optimization problems and explored different initialization techniques for QAOA's parameters. Furthermore, we examined the influence of gate errors on the QAOA's most important gate. Additionally, we demonstrated the efficacy of QAOA in an algorithm-hardware co-design context.

In Chapter 2, we first investigated the QAOA performance for constraint optimization problems. We showed that for applying QAOA to random Max-kXOR problems, the performance of the algorithm decreases for increasing k for hard-to-solve instances. We further provide evidence that the relative QAOA improvement from p to $p+1$ decreases with increasing k and p. This indicates that for high k, Max-kXOR problems need a high circuit depth, and therefore, running QAOA on a real device is unlikely. Our results emphasize the analytic study of Basso et al. [86], which shows that QAOA for even k is limited in performance. We further demonstrated the correlation of the clause-to-variable ratio r on the QAOA performance for Max-kXOR. Increasing r leads to harder problems and, therefore, harder to solve for QAOA. We notice a fast shift in performance when $r < 1$. In such cases, the problem instances are under-constrained, making them easier to solve. Conversely, when $r > 1$, the instances are in the over-constrained domain, with fewer solutions compared to the under-constrained domain. Here, the performance change in QAOA exhibits a linear trend with a slight negative slope as r increases. For Max-kSAT problems, we also provide evidence that the decrease in performance of QAOA for increasing k for hard-to-solve instances, meaning r is beyond the respected phase transition of the related decision problem.

In Chapter 3, we analyzed annealing-inspired initialization strategies for the QAOA. We showed that linear schedules outperform the Trotterized Quantum Annealing (TQA) protocol suggested by Sack et al. [92]. First, when examining a particular parameter setting p, linear schedules exhibit superior mean approximation ratios $\mathcal{L}_p(\vec{\alpha}^*)$ for the optimized schedule parameters as compared to TQA. Second, when employing QAOA with optimized schedule parameters, linear schedules also surpass TQA in terms of the mean approximation ratio $\mathcal{M}_p(\vec{\alpha}^*)$. We illustrated that the advantage is more pronounced for shallow circuit depth ($p < 10$). Moreover, we demonstrated that the optimal QAOA parameters $\vec{\gamma}^*$ and $\vec{\beta}^*$ are in closer proximity to the initial parameters $\vec{\gamma}(\vec{\alpha}^*_{\text{ls}})$ and $\vec{\beta}(\vec{\alpha}^*_{\text{ls}})$ of the linear schedules, as compared to TQA. This implies that employing linear schedules reduces the level of optimization effort required.

In Chapter 4, we examined the difference in the gate fidelity of the $R_{ZZ}(\gamma)$ gate in controlled-phase gate (CP) and controlled-Z gate (CZ) decomposition. We showed that $R_{ZZ}(\gamma)$, which is a cornerstone for the implementation of the QAOA, in CZ decomposition provides comparable gate fidelities as CP decomposition for small coherent and incoherent errors, despite its need for two two-qubit gates. Moreover, we showed that CZ is advantageous over CP for the pure incoherent error case for some states. This holds significance because the parameter space of a variational quantum algorithm experiences an increase of small local minima due to coherent errors, adding complexity to the optimization process. Thus, employing a QAOA on hardware featuring CZ gates for short circuit depth can offer advantages compared to hardware equipped with CP gates. In contrast to CP, the gate fidelity in CZ decomposition depends on the rotation angle γ.

In Chapter 5, we investigated the QAOA in the digital-analog paradigm. We showed that under realistic gate duration for single-qubit rotations, the QAOA in the bDAQC paradigm becomes identical to the pure digital version. In fact, the mean approximation ratios are nearly the same. If single-qubit gates are as fast as two-qubit gates, the QAOA in the bDAQC paradigm gets worse. Furthermore, we've shown that employing the variational approach aids in reducing the Trotter error linked to the bDAQC-QAOA method. Additionally, we demonstrated that the QAOA implemented in the bDAQC requires less time on the device compared to its digital counterpart for shallow circuit depth. Overall, the QAOA in the bDAQC scheme can be implemented on a real device and could lead to higher accuracy than the digital version due to being less error-prone to cross-talk errors, for instance.

This thesis sheds light on our expectations for QAOA concerning combinatorial optimization problems and the search for optimal parameters. It also highlights the vast potential for future exploration in this field. It remains uncertain whether QAOA can outperform classical algorithms, and identifying the problem domains that are particularly well-suited for accelerated performance with QAOA is an ongoing pursuit. Furthermore, introducing QAOA within the digital-analog scheme paves the way for developing a novel quantum machine that offers enhanced error robustness compared to its purely digital counterpart, tailored explicitly for QAOA. However, it is important to note that such a quantum machine has not yet been realized, and experimental physicists may need to conceive new concepts and approaches before we witness the emergence of a digital-analog device designed specifically for QAOA.

Bibliography

[1] Paul Benioff. "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines". In: *Journal of Statistical Physics* 22 (1980), pp. 563–591. URL: https://doi.org/10.1007/BF01011339.

[2] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21 (1982), pp. 467–488. URL: https://doi.org/10.1007/BF02650179.

[3] Isaac L. Chuang, Neil Gershenfeld, and Mark Kubinec. "Experimental Implementation of Fast Quantum Searching". In: *Phys. Rev. Lett.* 80 (1998), pp. 3408–3411. URL: https://link.aps.org/doi/10.1103/PhysRevLett.80.3408.

[4] J. A. Jones and M. Mosca. "Approximate Quantum Counting on an NMR Ensemble Quantum Computer". In: *Phys. Rev. Lett.* 83 (1999), pp. 1050–1053. URL: https://link.aps.org/doi/10.1103/PhysRevLett.83.1050.

[5] H. Häffner et al. "Scalable multiparticle entanglement of trapped ions". In: *Nature* 438 (2005), pp. 643–646. URL: https://doi.org/10.1038/nature04279.

[6] Lieven M. K. Vandersypen et al. "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance". In: *Nature* 414 (2001), pp. 883–887. URL: https://doi.org/10.1038/414883a.

[7] C. Monroe et al. "Demonstration of a Fundamental Quantum Logic Gate". In: *Phys. Rev. Lett.* 75 (1995), pp. 4714–4717. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.4714.

[8] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM J. Comput.* 26 (1997), pp. 1484–1509. URL: https://doi.org/10.1137/S0097539795293172.

[9] Lov K. Grover. "A Fast Quantum Mechanical Algorithm for Database Search". In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*. 1996, pp. 212–219. URL: https://doi.org/10.1145/237814.237866.

[10] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum Algorithm for Linear Systems of Equations". In: *Phys. Rev. Lett.* 103 (2009), p. 150502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.103.150502.

[11] William K. Wootters and Wojciech Zurek. “A single quantum cannot be cloned”. In: *Nature* 299 (1982), pp. 802–803. URL: https://www.nature.com/articles/299802a0.

[12] James L. Park. “The concept of transition in quantum mechanics”. In: *Foundations of Physics* 1 (1970), pp. 23–33. URL: https://doi.org/10.1007/BF00708652.

[13] Peter W. Shor. “Scheme for reducing decoherence in quantum computer memory”. In: *Phys. Rev. A* 52 (1995), R2493–R2496. URL: https://link.aps.org/doi/10.1103/PhysRevA.52.R2493.

[14] A. M. Steane. “Error Correcting Codes in Quantum Theory”. In: *Phys. Rev. Lett.* 77 (1996), pp. 793–797. URL: https://link.aps.org/doi/10.1103/PhysRevLett.77.793.

[15] Daniel Gottesman, Alexei Kitaev, and John Preskill. “Encoding a qubit in an oscillator”. In: *Phys. Rev. A* 64 (2001), p. 012310. URL: https://link.aps.org/doi/10.1103/PhysRevA.64.012310.

[16] Emanuel Knill and Raymond Laflamme. *Concatenated Quantum Codes*. 1996. URL: https://arxiv.org/abs/quant-ph/9608012.

[17] Michael A Nielsen and Isaac Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000. DOI: https://doi.org/10.1017/CBO9780511976667.

[18] Adriano Barenco et al. “Elementary gates for quantum computation”. In: *Phys. Rev. A* 52 (1995), pp. 3457–3467. URL: https://link.aps.org/doi/10.1103/PhysRevA.52.3457.

[19] Frank Arute et al. “Quantum supremacy using a programmable superconducting processor”. In: *Nature* 574 (2019), pp. 505–510. URL: https://www.nature.com/articles/s41586-019-1666-5.

[20] R. Barends et al. “Superconducting quantum circuits at the surface code threshold for fault tolerance”. In: *Nature* 508 (2014), pp. 500–503. URL: https://www.nature.com/articles/nature13171.

[21] Austin G. Fowler et al. “Surface codes: Towards practical large-scale quantum computation”. In: *Phys. Rev. A* 86 (2012), p. 032324. URL: https://link.aps.org/doi/10.1103/PhysRevA.86.032324.

[22] David S. Wang, Austin G. Fowler, and Lloyd C. L. Hollenberg. “Surface code quantum computing with error rates over 1%”. In: *Phys. Rev. A* 83 (2011), p. 020302. URL: https://link.aps.org/doi/10.1103/PhysRevA.83.020302.

[23] Austin G. Fowler, Ashley M. Stephens, and Peter Groszkowski. “High-threshold universal quantum computation on the surface code”. In: *Phys. Rev. A* 80 (2009), p. 052312. URL: https://link.aps.org/doi/10.1103/PhysRevA.80.052312.

[24] F. Bloch. “Nuclear Induction”. In: *Phys. Rev.* 70 (1946), pp. 460–474. URL: https://link.aps.org/doi/10.1103/PhysRev.70.460.

[25] Jun Zhang et al. “Geometric theory of nonlocal two-qubit operations”. In: *Phys. Rev. A* 67 (2003), p. 042313. URL: https://link.aps.org/doi/10.1103/PhysRevA.67.042313.

[26] A.Yu. Kitaev. "Fault-tolerant quantum computation by anyons". In: *Annals of Physics* 303.1 (2003), pp. 2–30. ISSN: 0003-4916. DOI: https://doi.org/10.1016/S0003-4916(02)00018-0. URL: https://www.sciencedirect.com/science/article/pii/S0003491602000180.

[27] Feng Pan and Pan Zhang. *Simulating the Sycamore quantum supremacy circuits*. 2021. URL: https://arxiv.org/abs/2103.03074.

[28] Laura Gentini et al. "Noise-resilient variational hybrid quantum-classical optimization". In: *Phys. Rev. A* 102 (2020), p. 052414. URL: https://link.aps.org/doi/10.1103/PhysRevA.102.052414.

[29] Lukasz Cincio et al. "Machine Learning of Noise-Resilient Quantum Circuits". In: *PRX Quantum* 2 (2021), p. 010324. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.010324.

[30] Jarrod R McClean et al. "The theory of variational hybrid quantum-classical algorithms". In: *New Journal of Physics* 18 (2016), p. 023023. URL: https://doi.org/10.1088/1367-2630/18/2/023023.

[31] Jarrod R. McClean et al. "Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states". In: *Phys. Rev. A* 95 (2017), p. 042308. URL: https://link.aps.org/doi/10.1103/PhysRevA.95.042308.

[32] Kunal Sharma et al. "Noise resilience of variational quantum compiling". In: *New Journal of Physics* 22 (2020), p. 043006. URL: https://doi.org/10.1088/1367-2630/ab784c.

[33] Yuri Alexeev et al. "Quantum Computer Systems for Scientific Discovery". In: *PRX Quantum* 2 (2021), p. 017001. DOI: 10.1103/PRXQuantum.2.017001. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.017001.

[34] Hasan Mustafa et al. "Variational Quantum Algorithms for Chemical Simulation and Drug Discovery". In: 2022, pp. 1–8. DOI: 10.1109/TQCEBT54229.2022.10041453.

[35] R.P. Feynman and L.M. Brown. *Selected Papers of Richard Feynman: With Commentary*. World Scientific, 2000. URL: https://books.google.de/books?id=qnwkqcVixucC.

[36] Ivan Kassal et al. "Simulating Chemistry Using Quantum Computers". In: *Annual Review of Physical Chemistry* 62 (2011), pp. 185–207. URL: https://doi.org/10.1146/annurev-physchem-032210-103512.

[37] Alán Aspuru-Guzik et al. "Simulated quantum computation of molecular energies". In: *Science* 309 (2005), pp. 1704–1707. DOI: DOI:10.1126/science.1113479.

[38] Joonsuk Huh et al. "Boson sampling for molecular vibronic spectra". In: *Nature Photonics* 9 (2015), pp. 615–620. URL: https://doi.org/10.1038/nphoton.2015.153.

[39] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. "Progress towards practical quantum variational algorithms". In: *Phys. Rev. A* 92 (2015), p. 042303. URL: https://link.aps.org/doi/10.1103/PhysRevA.92.042303.

[40] Leonie Mueck. "Quantum reform". In: *Nature Chemistry* (2015), pp. 361–363. URL: https://doi.org/10.1038/nchem.2248.

[41] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *A Quantum Approximate Optimization Algorithm*. 2014. URL: https://arxiv.org/abs/1411.4028.

[42] Alberto Peruzzo et al. "A variational eigenvalue solver on a photonic quantum processor". In: *Nature Communications* 5 (2014), p. 4213. URL: https://doi.org/10.1038/ncomms5213.

[43] Jarrod R McClean et al. "The theory of variational hybrid quantum-classical algorithms". In: *New Journal of Physics* 18 (2016), p. 023023. URL: https://doi.org/10.1088/1367-2630/18/2/023023.

[44] Jonathan Romero et al. "Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz". In: *Quantum Science and Technology* 4 (2018), p. 014008. URL: https://doi.org/10.1088/2058-9565/aad3e4.

[45] Leo Zhou et al. "Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices". In: *Phys. Rev. X* 10 (2020), p. 021067. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.021067.

[46] Andrea Skolik et al. "Layerwise learning for quantum neural networks". In: *Quantum Machine Intelligence* 3 (2021), p. 5. URL: https://doi.org/10.1007/s42484-020-00036-4.

[47] Ernesto Campos, Aly Nasrallah, and Jacob Biamonte. "Abrupt transitions in variational quantum circuit training". In: *Phys. Rev. A* 103 (2021), p. 032607. URL: https://link.aps.org/doi/10.1103/PhysRevA.103.032607.

[48] Chenfeng Cao and Xin Wang. "Noise-Assisted Quantum Autoencoder". In: *Phys. Rev. Applied* 15 (2021), p. 054012. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.15.054012.

[49] Panagiotis Kl. Barkoutsos et al. "Improving Variational Quantum Optimization using CVaR". In: *Quantum* 4 (2020), p. 256. URL: https://doi.org/10.22331/q-2020-04-20-256.

[50] Ioannis Kolotouros and Petros Wallden. "Evolving objective function for improved variational quantum optimization". In: *Phys. Rev. Res.* 4 (2022), p. 023225. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.4.023225.

[51] Matteo M. Wauters et al. "Reinforcement-learning-assisted quantum optimization". In: *Phys. Rev. Research* 2 (2020), p. 033446. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.033446.

[52] Zhihui Wang et al. "Quantum approximate optimization algorithm for MaxCut: A fermionic view". In: *Phys. Rev. A* 97 (2018), p. 022304. DOI: 10.1103/PhysRevA.97.022304.

[53] Edward Farhi et al. "The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size". In: *Quantum* 6 (2022), p. 759. URL: https://doi.org/10.22331/q-2022-07-07-759.

[54] Joao Basso et al. "The Quantum Approximate Optimization Algorithm at High Depth for MaxCut on Large-Girth Regular Graphs and the Sherrington-Kirkpatrick Model". In: Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. URL: https://drops.dagstuhl.de/opus/volltexte/2022/16514/.

[55] V. Akshay et al. “Reachability Deficits in Quantum Approximate Optimization”. In: *Phys. Rev. Lett.* 124 (2020), p. 090504. DOI: 10.1103/PhysRevLett.124.090504.

[56] M. B. Hastings. *Classical and Quantum Bounded Depth Approximation Algorithms*. 2019. URL: https://arxiv.org/abs/1905.07047.

[57] V. Akshay et al. “Parameter concentrations in quantum approximate optimization”. In: *Phys. Rev. A* 104 (2021), p. L010401. URL: https://link.aps.org/doi/10.1103/PhysRevA.104.L010401.

[58] Michael Streif and Martin Leib. “Training the quantum approximate optimization algorithm without access to a quantum processing unit”. In: *Quantum Science and Technology* 5 (2020), p. 034008. URL: https://doi.org/10.1088/2058-9565/ab8c2b.

[59] Jahan Claes and Wim van Dam. “Instance Independence of Single Layer Quantum Approximate Optimization Algorithm on Mixed-Spin Models at Infinite Size”. In: *Quantum* 5 (2021), p. 542. URL: https://doi.org/10.22331/q-2021-09-15-542.

[60] Ruslan Shaydulin and Stefan M. Wild. “Exploiting Symmetry Reduces the Cost of Training QAOA”. In: *IEEE Transactions on Quantum Engineering* 2 (2021). URL: https://www.osti.gov/biblio/1824244.

[61] John Preskill. *Quantum computing and the entanglement frontier*. 2012. URL: https://arxiv.org/abs/1203.5813.

[62] Scott Aaronson and Alex Arkhipov. “The Computational Complexity of Linear Optics”. In: *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*. 2011, pp. 333–342. URL: https://doi.org/10.1145/1993636.1993682.

[63] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. “Average-Case Complexity Versus Approximate Simulation of Commuting Quantum Computations”. In: *Phys. Rev. Lett.* 117 (2016), p. 080501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.117.080501.

[64] Sergio Boixo et al. “Characterizing quantum supremacy in near-term devices”. In: *Nature Physics* 14 (2018), pp. 595–600. URL: https://doi.org/10.1038/s41567-018-0124-x.

[65] Adam Bouland et al. “On the complexity and verification of quantum random circuit sampling”. In: *Nature Physics* 15 (2019), pp. 159–163. URL: https://doi.org/10.1038/s41567-018-0318-2.

[66] Edward Farhi and Aram W Harrow. *Quantum Supremacy through the Quantum Approximate Optimization Algorithm*. 2016. URL: https://arxiv.org/abs/1602.07674.

[67] Stuart Hadfield et al. “From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz”. In: *Algorithms* 12 (2019). URL: https://www.mdpi.com/1999-4893/12/2/34.

[68] Zhihui Wang et al. “XY mixers: Analytical and numerical results for the quantum alternating operator ansatz”. In: *Phys. Rev. A* 101 (2020), p. 012320. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.012320.

[69] Ryan LaRose, Eleanor Rieffel, and Davide Venturelli. "Mixer-Phaser Ansätze for Quantum Optimization with Hard Constraints". In: 2021. arXiv: 2107.06651 [quant-ph].

[70] Tobias Stollenwerk, Elisabeth Lobe, and Martin Jung. "Flight Gate Assignment with a Quantum Annealer". In: *Quantum Technology and Optimization Problems*. Springer International Publishing, 2019, pp. 99–110. URL: https://arxiv.org/abs/1811.09465.

[71] M. S. Sarandy, L. -A. Wu, and D. A. Lidar. *Consistency of the Adiabatic Theorem*. 2005. arXiv: quant-ph/0405059 [quant-ph].

[72] H. D. Zeh. "On the interpretation of measurement in quantum theory". In: *Foundations of Physics* 1 (1970), pp. 69–76. URL: https://doi.org/10.1007/BF00708656.

[73] O Kuebler and H.D Zeh. "Dynamics of quantum correlations". In: *Annals of Physics* 76 (1973), pp. 405–418. URL: https://www.sciencedirect.com/science/article/pii/0003491673900407.

[74] W. H. Zurek. "Pointer basis of quantum apparatus: Into what mixture does the wave packet collapse?" In: *Phys. Rev. D* 24 (1981), pp. 1516–1525. URL: https://link.aps.org/doi/10.1103/PhysRevD.24.1516.

[75] W. H. Zurek. "Environment-induced superselection rules". In: *Phys. Rev. D* 26 (1982), pp. 1862–1880. URL: https://link.aps.org/doi/10.1103/PhysRevD.26.1862.

[76] E. Joos and H. D. Zeh. "The emergence of classical properties through interaction with the environment". In: *Zeitschrift für Physik B Condensed Matter* 59 (1985), pp. 223–243. URL: https://doi.org/10.1007/BF01725541.

[77] W. H. Zurek. "Reduction of the Wavepacket: How Long Does it Take?" In: Springer US, 1986, pp. 145–149. URL: https://doi.org/10.1007/978-1-4613-2181-1_10.

[78] M.A. Schlosshauer. *Decoherence: and the Quantum-To-Classical Transition*. The Frontiers Collection. Springer Berlin Heidelberg, 2010. URL: https://books.google.de/books?id=agmycQAACAAJ.

[79] H. P. Breuer and F. Petruccione. *The theory of open quantum systems*. Great Clarendon Street: Oxford University Press, 2002.

[80] Juan Pablo Paz, Salman Habib, and Wojciech H. Zurek. "Reduction of the wave packet: Preferred observable and decoherence time scale". In: *Phys. Rev. D* 47 (1993), pp. 488–501. URL: https://link.aps.org/doi/10.1103/PhysRevD.47.488.

[81] Todd A. Brun Daniel A. Lidar, ed. *Quantum Error Correction*. Cambridge University Press, 2013. DOI: 10.1017/CBO9781139034807.

[82] S. Banerjee. *Open Quantum Systems: Dynamics of Nonclassical Evolution*. Texts and Readings in Physical Sciences. Springer Nature Singapore, 2018. ISBN: 9789811348150. URL: https://books.google.de/books?id=kHpDwAEACAAJ.

[83] K. Kraus et al. *States, Effects, and Operations: Fundamental Notions of Quantum Theory*. Lecture Notes in Physics. Springer Berlin Heidelberg, 1983. ISBN: 9780387127323. URL: https://books.google.de/books?id=fRBBAQAAIAAJ.

[84] Kunal Marwaha. "Local classical MAX-CUT algorithm outperforms $p = 2$ QAOA on high-girth regular graphs". In: *Quantum* 5 (2021), p. 437. DOI: 10.22331/q-2021-04-20-437. URL: https://doi.org/10.22331/q-2021-04-20-437.

[85] Kunal Marwaha and Stuart Hadfield. *Bounds on approximating Max kXOR with quantum and classical local algorithms*. 2021. DOI: 10.48550/ARXIV.2109.10833. URL: https://arxiv.org/abs/2109.10833.

[86] Joao Basso et al. "Performance and limitations of the QAOA at constant levels on large sparse hypergraphs and spin glass models". In: *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. 2022, pp. 335–343. DOI: 10.1109/FOCS54457.2022.00039.

[87] Johan Håstad. "Some Optimal Inapproximability Results". In: *J. ACM* 48 (2001), pp. 798–859. URL: https://doi.org/10.1145/502090.502098.

[88] Richard M. Karp. "Reducibility among Combinatorial Problems". In: Springer US, 1972, pp. 85–103. URL: https://doi.org/10.1007/978-1-4684-2001-2_9.

[89] Nadia Creignou, Hervé Daudé, and Oliver Dubois. "Approximating the Satisfiability Threshold for Random k-XOR-formulas". In: *Combinatorics, Probability and Computing* 12 (2003), pp. 113–126. DOI: https://doi.org/10.1017/S0963548302005552.

[90] Amin Coja-Oghlan, Amir Haqshenas, and Samuel Hetterich. "Walksat Stalls Well Below Satisfiability". In: *SIAM J. Discret. Math.* 31 (2017), pp. 1160–1173. URL: https://doi.org/10.1137/16M1084158.

[91] Samuel Hetterich. "Analysing Survey Propagation Guided Decimationon Random Formulas". In: vol. 55. 2016. URL: http://drops.dagstuhl.de/opus/volltexte/2016/6219.

[92] Stefan H. Sack and Maksym Serbyn. "Quantum annealing initialization of the quantum approximate optimization algorithm". In: *Quantum* 5 (2021), p. 491. URL: https://doi.org/10.22331%2Fq-2021-07-01-491.

[93] Jonathan Wurtz and Peter J. Love. "Counterdiabaticity and the quantum approximate optimization algorithm". In: *Quantum* 6 (2022), p. 635. DOI: https://doi.org/10.22331/q-2022-01-27-635.

[94] Tameem Albash and Daniel A. Lidar. "Adiabatic quantum computation". In: *Rev. Mod. Phys.* 90 (2018), p. 015002. URL: https://link.aps.org/doi/10.1103/RevModPhys.90.015002.

[95] David Sherrington and Scott Kirkpatrick. "Solvable Model of a Spin-Glass". In: *Phys. Rev. Lett.* 35 (1975), pp. 1792–1796. URL: https://link.aps.org/doi/10.1103/PhysRevLett.35.1792.

[96] F Barahona. "On the computational complexity of Ising spin glass models". In: *Journal of Physics A: Mathematical and General* 15 (1982), pp. 3241–3253. URL: https://doi.org/10.1088/0305-4470/15/10/028.

[97] M. Cerezo et al. "Cost function dependent barren plateaus in shallow parametrized quantum circuits". In: *Nature Communications* 12.1 (2021). URL: https://doi.org/10.1038/s41467-021-21728-w.

[98] G. G. Guerreschi and A. Y. Matsuura. “QAOA for Max-Cut requires hundreds of qubits for quantum speed-up”. In: *Scientific Reports* 9 (2019), p. 6903. URL: https://doi.org/10.1038/s41598-019-43176-9.

[99] Daniel Liang, Li Li, and Stefan Leichenauer. “Investigating quantum approximate optimization algorithms under bang-bang protocols”. In: *Phys. Rev. Research* 2 (2020), p. 033402. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.033402.

[100] Zhi-Cheng Yang et al. “Optimizing Variational Quantum Algorithms Using Pontryagin’s Minimum Principle”. In: *Phys. Rev. X* 7 (2017), p. 021027. URL: https://link.aps.org/doi/10.1103/PhysRevX.7.021027.

[101] R. Cabrera and W.E. Baylis. “Average fidelity in n-qubit systems”. In: *Physics Letters A* 368 (2007), pp. 25–28. URL: https://www.sciencedirect.com/science/article/pii/S0375960107004859.

[102] Richard Josza. “Fidelity for Mixed Quantum States”. In: *Journal of Modern Optics* 41 (1994).

[103] A. Uhlmann. “The “transition probability” in the state space of a $\star$-algebra”. In: *Reports on Mathematical Physics* 9 (1976), pp. 273–279. URL: https://www.sciencedirect.com/science/article/pii/0034487776900604.

[104] Mark D. Bowdrey et al. “Fidelity of single qubit maps”. In: *Physics Letters A* 294 (2002), pp. 258–260. URL: https://www.sciencedirect.com/science/article/pii/S0375960102000695.

[105] Michael A Nielsen. “A simple formula for the average gate fidelity of a quantum dynamical operation”. In: *Physics Letters A* 303 (2002), pp. 249–252. URL: https://www.sciencedirect.com/science/article/pii/S0375960102012720.

[106] J. F. Poyatos, J. I. Cirac, and P. Zoller. “Complete Characterization of a Quantum Process: The Two-Bit Quantum Gate”. In: *Phys. Rev. Lett.* 78 (1997), pp. 390–393. URL: https://link.aps.org/doi/10.1103/PhysRevLett.78.390.

[107] J. I. Cirac and P. Zoller. “Quantum Computations with Cold Trapped Ions”. In: *Phys. Rev. Lett.* 74 (20 1995), pp. 4091–4094. URL: https://link.aps.org/doi/10.1103/PhysRevLett.74.4091.

[108] Markus J. Storcz and Frank K. Wilhelm. “Decoherence and gate performance of coupled solid-state qubits”. In: *Phys. Rev. A* 67 (2003), p. 042319. URL: https://link.aps.org/doi/10.1103/PhysRevA.67.042319.

[109] Michael Thorwart and Peter Hänggi. “Decoherence and dissipation during a quantum XOR gate operation”. In: *Phys. Rev. A* 65 (2001), p. 012309. URL: https://link.aps.org/doi/10.1103/PhysRevA.65.012309.

[110] Klaus Mølmer and Anders Sørensen. “Multiparticle Entanglement of Hot Trapped Ions”. In: *Phys. Rev. Lett.* 82 (1999), pp. 1835–1838. URL: https://link.aps.org/doi/10.1103/PhysRevLett.82.1835.

[111] Anupam Mitra et al. “Robust Mølmer-Sørensen gate for neutral atoms using rapid adiabatic Rydberg dressing”. In: *Phys. Rev. A* 101 (2020), p. 030301. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.030301.

[112] Jun Zhang et al. "Exact Two-Qubit Universal Quantum Circuit". In: *Phys. Rev. Lett.* 91 (2003), p. 027903. URL: https://link.aps.org/doi/10.1103/PhysRevLett.91.027903.

[113] Yuriy Makhlin. "Nonlocal Properties of Two-Qubit Gates and Mixed States, and the Optimization of Quantum Computations". In: *Quantum Information Processing* 1 (2002), pp. 243–252. URL: https://doi.org/10.1023/A:1022144002391.

[114] Nathan Lacroix et al. "Improving the Performance of Deep Quantum Optimization Algorithms with Continuous Gate Sets". In: *PRX Quantum* 1 (2020), p. 110304. URL: https://link.aps.org/doi/10.1103/PRXQuantum.1.020304.

[115] K. R. Brown et al. "Single-qubit-gate error below $\mathbf{10^{-4}}$ in a trapped ion". In: *Phys. Rev. A* 84 (2011), p. 030303. URL: https://link.aps.org/doi/10.1103/PhysRevA.84.030303.

[116] Oleksandr Kyriienko and Anders S. Sørensen. "Floquet Quantum Simulation with Superconducting Qubits". In: *Phys. Rev. Applied* 9 (2018), p. 064029. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.9.064029.

[117] N. Goldman and J. Dalibard. "Periodically Driven Quantum Systems: Effective Hamiltonians and Engineered Gauge Fields". In: *Phys. Rev. X* 4 (2014), p. 031027. URL: https://link.aps.org/doi/10.1103/PhysRevX.4.031027.

[118] Tadashi Kadowaki and Hidetoshi Nishimori. "Quantum annealing in the transverse Ising model". In: *Phys. Rev. E* 58 (5 1998), pp. 5355–5363. URL: https://link.aps.org/doi/10.1103/PhysRevE.58.5355.

[119] Dorit Aharonov et al. "Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation". In: (2004). DOI: 10.48550/ARXIV.QUANT-PH/0405098.

[120] Seth Lloyd. "Universal Quantum Simulators". In: *Science* 273 (1996), pp. 1073–1078. URL: https://www.science.org/doi/abs/10.1126/science.273.5278.1073.

[121] C. J. Ballance et al. "High-Fidelity Quantum Logic Gates Using Trapped-Ion Hyperfine Qubits". In: *Phys. Rev. Lett.* 117 (2016), p. 060504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.117.060504.

[122] Norbert M. Linke et al. "Experimental comparison of two quantum computing architectures". In: *Proceedings of the National Academy of Sciences of the United States of America* 114 (), p. 3305. DOI: https://doi.org/10.1073/pnas.1618020114.

[123] C.P. Slichter. *Principles of Magnetic Resonance*. Springer Series in Solid-State Sciences. Springer Berlin Heidelberg, 1996. URL: https://books.google.de/books?id=zgnrRkaIhFoC.

[124] Herschel Rabitz et al. "Whither the Future of Controlling Quantum Phenomena?" In: *Science* 288 (2000), pp. 824–828. URL: https://www.science.org/doi/abs/10.1126/science.288.5467.824.

[125] Jennifer L. Dodd et al. "Universal quantum computation and simulation using any entangling Hamiltonian and local unitaries". In: *Phys. Rev. A* 65 (2002), p. 040301. URL: https://link.aps.org/doi/10.1103/PhysRevA.65.040301.

[126] Adrian Parra-Rodriguez et al. “Digital-analog quantum computation”. In: *Phys. Rev. A* 101 (2020), p. 022305. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.022305.

[127] Ll. Masanes, G. Vidal, and J. I. Latorre. “Time-Optimal Hamiltonian Simulation and Gate Synthesis Using Homogeneous Local Unitaries”. In: *Quantum Info. Comput.* 2 (2002), pp. 285–296.

[128] C. H. Bennett et al. “Optimal simulation of two-qubit Hamiltonians using general local operations”. In: *Phys. Rev. A* 66 (2002), p. 012305. URL: https://link.aps.org/doi/10.1103/PhysRevA.66.012305.

[129] E. Jané et al. “Simulation of Quantum Dynamics with Quantum Optical Systems”. In: *Quantum Info. Comput.* 3 (2003), pp. 15–37.

[130] M.-H. Yung et al. “From transistor to trapped-ion computers for quantum chemistry”. In: *Scientific Reports* 4.1 (2014), p. 3589. URL: https://doi.org/10.1038/srep03589.

[131] A. Mezzacapo et al. “Digital Quantum Rabi and Dicke Models in Superconducting Circuits”. In: *Scientific Reports* 4 (2014), p. 7482. URL: https://doi.org/10.1038/srep07482.

[132] Lucas Lamata. “Basic protocols in quantum reinforcement learning with superconducting circuits”. In: *Scientific Reports* 7 (2017), p. 1609. URL: https://doi.org/10.1038/s41598-017-01711-6.

[133] Ana Martin et al. “Digital-analog quantum algorithm for the quantum Fourier transform”. In: *Phys. Rev. Research* 2 (2020), p. 013012. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.013012.

[134] Paula García-Molina, Ana Martin, and Mikel Sanz. *Noise in Digital and Digital-Analog Quantum Computation*. 2021. URL: https://arxiv.org/abs/2107.12969.

[135] M. Van den Nest, W. Dür, and H. J. Briegel. “Completeness of the Classical 2D Ising Model and Universal Quantum Computation”. In: *Phys. Rev. Lett.* 100 (2008), p. 110501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.100.110501.

[136] Asier Galicia et al. “Enhanced connectivity of quantum hardware with digital-analog control”. In: *Phys. Rev. Research* 2 (2020), p. 033103. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.033103.

[137] Ana Martin, Ruben Ibarrondo, and Mikel Sanz. *Digital-analog co-design of the Harrow-Hassidim-Lloyd algorithm*. 2022. URL: https://arxiv.org/abs/2207.13528.

[138] Jarrod R. McClean et al. “Barren plateaus in quantum neural network training landscapes”. In: *Nature Communications* 9 (2018), p. 4812. URL: https://doi.org/10.1038/s41467-018-07090-4.

[139] Samson Wang et al. “Noise-induced plateaus in variational quantum algorithms”. In: *Nature Communications* 12 (2021), p. 6961. URL: https://doi.org/10.1038/s41467-021-27045-6.

[140] Sumeet Khatri et al. “Quantum-assisted quantum compiling”. In: *Quantum* 3 (2019), p. 140. URL: https://doi.org/10.22331/q-2019-05-13-140.

[141] Marco Maronese et al. *Quantum Compiling*. 2021. URL: https://arxiv.org/abs/2112.00187.

[142] Ian D. Kivlichan et al. "Improved Fault-Tolerant Quantum Simulation of Condensed-Phase Correlated Electrons via Trotterization". In: *Quantum* 4 (2020), p. 296. URL: https://doi.org/10.22331/q-2020-07-16-296.

[143] Dave Wecker et al. "Gate-count estimates for performing quantum chemistry on small quantum computers". In: *Phys. Rev. A* 90 (2014), p. 022305. URL: https://link.aps.org/doi/10.1103/PhysRevA.90.022305.