Research Project

Czech Technical University in Prague

F4

Faculty of Nuclear Sciences and Physical Engineering
Department of Physics

Simulation of Excitation Dynamics in Networks Using Quantum Computers

Jiří Viták

Supervisor: Iskender Yalcinkaya, Ph.D.
August 2025

Acknowledgements

I would like to thank Dr. Iskender Yalcinkaya for the professional help and time he gave me during the writing of this thesis.

Declaration

I declare that the presented work is solely mine and that I cited all the literature used.

In Prague, 5. August 2025

Abstract

The study of excitation dynamics in various networks using continuous-time quantum walks (CTQWs) has garnered significant attention in recent years, particularly in the context of quantum computing and quantum information science. This project focuses on simulating CTQWs on quantum computers, with a particular emphasis on excitation transfer and Anderson localization. We explore strategies to enhance excitation transfer in graphs that do not naturally exhibit perfect state transfer (PST) by adding internal connections or introducing disorder into the system's Hamiltonian. Additionally, we investigate the disrupting effects of disorder on excitation transfer and the phenomenon of Anderson localization in CTQWs. Using variational quantum algorithms (VQAs) or internal connections in graphs, we were able to simulate and observe enhanced excitation transfer in graphs.

Keywords: Continuous-time quantum walk, excitation transfer, Anderson localization, Variational Quantum Algorithms

Supervisor: Iskender Yalcinkaya, Ph.D.
Katedra fyziky,
Fakulta jaderná a fyzikálně inženýrská
ČVUT v Praze

Abstrakt

Studium excitační dynamiky v různých sítích pomocí kvantových procházek se spojitým časem (CTQW) si v posledních letech získalo značnou pozornost, zejména v kontextu kvantových počítačů a kvantové informatiky. Tento projekt se zaměřuje na simulaci CTQW na kvantových počítačích se zvláštním důrazem na přenos excitace a Andersonovu lokalizaci. Zkoumáme strategie pro zlepšení přenosu excitace v grafech, které přirozeně nevykazují dokonalý přenos stavu (PST), přidáním vnitřních spojení nebo zavedením neuspořádanosti do hamiltoniánu systému. Kromě toho zkoumáme rušivé účinky neuspořádanosti na přenos excitace a fenomén Andersonovy lokalizace v CTQW. Pomocí variačních kvantových algoritmů (VQA) nebo vnitřních spojení v grafech jsme byli schopni simulovat a pozorovat zlepšený přenos excitace v grafech.

Klíčová slova: Kvantová procházka v spojitém čase, přenos excitace, Andersonova lokalizace, variabilní kvantové algoritmy

Překlad názvu: Simulace dynamiky excitací v sítích pomocí kvantových počítačů

Contents

Figures

Tables

Chapter 1

Introduction

Continuous-time quantum walks (CTQW) are the quantum analogue of classical random walks and have emerged as a powerful framework for simulating complex quantum dynamics [1, 2]. They have been applied to a wide range of problems in quantum computing, including quantum algorithm design and Hamiltonian simulation. For example, CTQWs have been used to design quantum search algorithms [2]. In fact, certain quantum walk models are known to implement universal quantum computation [2], indicating their broad applicability in quantum computing. Beyond purely computational tasks, CTQWs have also been applied in modeling quantum processes in chemistry and biology. For instance, quantum walk models have been used to simulate excitation energy transfer in photosynthetic light-harvesting complexes [3]. Similarly, CTQWs provide a natural framework for studying and simulating excitation dynamics in networks via usage of quantum computers. Furthermore, when disorder is introduced into the system, one can observe Anderson localization, an important phenomenon from condensed matter physics. The ability to efficiently implement and simulate CTQWs on quantum computers thus opens new possibilities for exploring and controlling such quantum transport phenomena, as well as for developing novel quantum algorithms.

In this project, we will focus on simulating CTQWs on quantum computers, with a particular emphasis on excitation transfer and Anderson localization. We will explore strategies to enhance excitation (state) transfer in graphs that do not naturally exhibit perfect state transfer (PST) by adding internal connections or introducing disorder into the system's Hamiltonian. Additionally, we will investigate the phenomenon of Anderson localization in CTQWs and its implications for excitation transfer. The project is structured as follows. In Chapter 2, we present the relevant theoretical background, covering topics such as quantum computing, graph structures, CTQWs, excitation transfer, and the role of Anderson localization in quantum walks. We then describe the methods used to simulate CTQWs on quantum computers, including Trotterization techniques for time-evolution of the Hamiltonian and the use of variational quantum algorithms (VQAs) for optimizing excitation transfer. Additionally, we discuss the implementation of quantum error mitigation and suppression techniques to improve the reliability of our simulations. Chap-

ter 3 presents the results of our simulations, including the effects of internal connections and disorder on excitation transfer, as well as the observation of Anderson localization in CTQWs.

Chapter 2

Theoretical Background

2.1 Quantum computing

Quantum computing is an emerging scientific field that leverages the principles of quantum mechanics to perform computations far more efficiently than classical computers for certain tasks. Since Richard Feynman proposed the idea in 1982, both quantum computers and quantum algorithms have been subjects of intense research [4].

Most notably, Grover's algorithm demonstrated that quantum computers can search an unsorted database in $O(\sqrt{N})$ time, compared to the $O(N)$ time required by classical computers. Later, Shor's algorithm showed that quantum computers can factor large numbers exponentially faster than any known classical algorithm [5, 6].

Quantum computing makes use of Dirac notation, in which a quantum state $|\psi\rangle$ is represented as a vector in a complex Hilbert space $\mathcal{H}$. The time evolution of a quantum state is governed by a unitary evolution operator U, derived from the Schrödinger equation. When the Hamiltonian H is time-independent, this operator is defined as $U(t) = e^{-\frac{i}{\hbar}Ht}$ where $\hbar$ is the Planck's constant and t denotes time [7]. In quantum information theory, $\hbar$ is often set to 1 for simplicity, leading to the expression $U(t) = e^{-iHt}$.

2.2 Graph structures

Graph structures are mathematical representations of networks, consisting of vertices (nodes) and edges (connections). They are widely used in various fields, including computer science, physics, and biology, to model complex systems. In quantum computing, quantum walks on graphs are widely used to model dynamics of quantum systems.

Formally, a graph is a pair $G = (V, E)$, where V is a non-empty finite set of vertices and E is a finite set of edges. A graph is called *simple* if it is undirected and contains neither self-loops nor multiple edges. In an undirected graph, edges have no orientation, meaning that the connection between two vertices does not have a direction. Conversely, in a directed graph (or digraph), each edge has a direction, indicating a one-way relationship

from one vertex to another [8].

Each graph has its own matrix representation. For a graph G with vertices labeled $\{1, 2, \ldots, n\}$, the *adjacency matrix* A is a square $n \times n$ matrix containing information about links between the vertices. The element A_{ij} is equal to 1 if there is an edge between vertex i and vertex j, and 0 otherwise. The *degree matrix* D is a diagonal matrix where each diagonal element D_{ii} represents the degree of vertex i, which is the number of edges connected to it. The *Laplacian matrix* L is defined as $L = D - A$, and it encodes information about the connectivity of the graph [9].

Notable graph structures that we will be using in our simulations include complete graphs and cycle graphs. A complete graph is a simple graph in which each pair of distinct vertices is connected by a unique edge [8]. A cycle graph consists of a single closed loop, where each vertex is connected to exactly two others. In this thesis, we will also be utilizing cycle graphs with a few additional internal connections. These connections alter the graph structure and can be used to enhance excitation transfer.

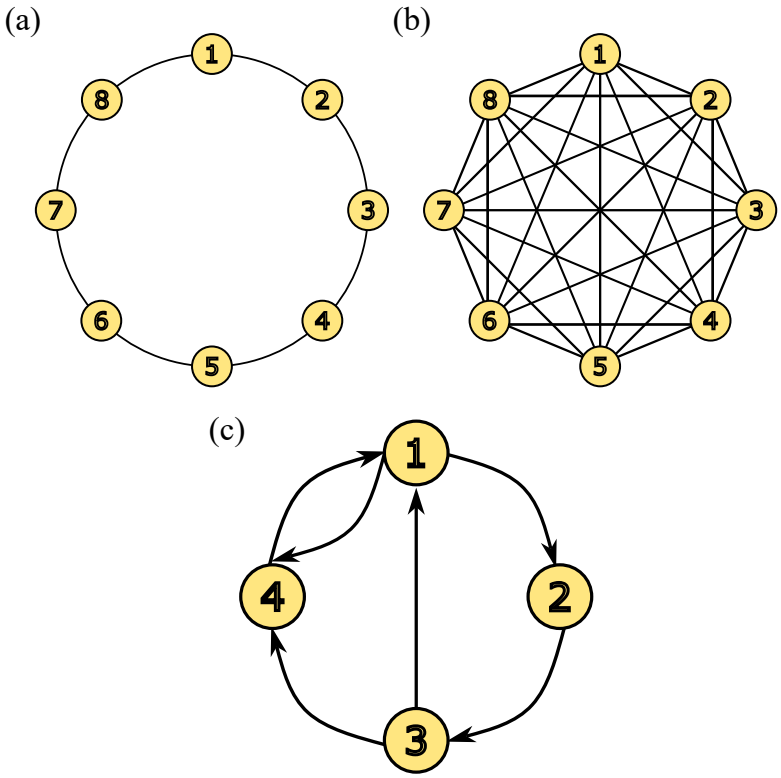


Figure 2.1: Examples of graph structures: (a) a cycle graph with 8 vertices, denoted C_8; (b) a complete graph with 8 vertices, denoted K_8; (c) a directed graph.

2.3 Continuous-time quantum walk

Quantum walks are quantum analogues of classical random walks, where a particle (or walker) propagates through a graph according to quantum

mechanical principles. The use of quantum mechanics allows quantum walks to exhibit properties, such as interference and superposition, which can lead to faster more efficient exploration of the graph, and exponential speedups in certain computational tasks compared to classical random walks. Based on the time evolution of the quantum state, quantum walks can be classified into two categories: discrete-time quantum walks (DTQW) [10] and continuous-time quantum walks (CTQW) [11]. In this thesis, we will focus solely on CTQW, as it allows us to omit the coin operator—a key component of DTQW that determines the walk's direction and requires an additional qubit for implementation in simulations.

Let $G = (V, E)$ be a graph with N vertices. The CTQW on the graph is described by the Hamiltonian $H = -\gamma A$, where γ is a hopping parameter and the Hamiltonian is defined in the basis of the vertices of the graph $\{1, 2, \ldots, n\}$ [1] The time evolution of the quantum state $|\psi(t)\rangle$ is given by the Schrödinger equation

$$i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle . \tag{2.1}$$

The solution to equation (2.1) is given by

$$|\psi(t)\rangle = U(t)|\psi(0)\rangle = e^{-iHt}|\psi(0)\rangle , \tag{2.2}$$

where $U(t)$ describes the time-evolution of CTQW $|\psi(t)\rangle$ from the initial state $|\psi(0)\rangle$ at time $t = 0$ [12]. N discrete components of $|\psi(t)\rangle$ are the probability amplitudes corresponding to each vertex j of the graph. A particle localized at any vertex j of the graph is described by the state $|j\rangle$, where all the states form a complete and orthonormal set over all sites, i.e. $\sum_{j\in N}|j\rangle\langle j| = I$. Therefore the probability of finding the particle at vertex m at time t is given by the expression

$$P_{n,m}(t) = |\langle m|\exp(-iHt)|n\rangle|^2, \tag{2.3}$$

where n is the initial position of the particle [13].

2.3.1 Trotterization

As dictated by principles of quantum mechanics, the time evolution is continuous. However, most quantum hardware operates using discrete gate sets and cannot directly implement this continuous evolution. To bridge this gap, we can utilize Trotterization (also referred to as Suzuki-Trotter decomposition scheme or splitting method), a technique that approximates the continuous-time evolution operator by decomposing it into a sequence of simpler unitary operations that can be implemented with discrete quantum gates.

Mathematically, the Suzuki-Trotter decomposition is an approximation of the operator exponentials, expressed as follows:

$$e^{(A+B+\ldots)h} = e^{Ah}e^{Bh}\cdots + \mathcal{O}(h^2), \tag{2.4}$$

[1] In this thesis, we will use the hopping parameter $\gamma = 1$ for simplicity without losing any generality; parameter γ only rescales the time evolution of the quantum state, which can be absorbed into the time variable t.

where A, B are operators and $h \to 0$ [14]. The error arrises due to non-commutativity and has been studied in detail in recent years [15, 16]. In the case of simulating CTQW evolution, it is a trade-off between the number of Trotter steps and the error in the approximation. The more Trotter steps we take, the smaller the error, but at the same time we need to implement more gates on the quantum computer, which is a limitation in the noisy intermediate-scale quantum (NISQ) era, where the number of gates is limited by the decoherence time of the qubits.

Assuming time-independent composite Hamiltonian $\mathcal{H}$, which can be decomposed into K terms as follows:

$$\mathcal{H} = \sum_{k=1}^{K} \mathcal{H}_k, \tag{2.5}$$

where terms $\mathcal{H}_k$ do not generally commute (i.e., $[\mathcal{H}_i, \mathcal{H}_j] \neq 0$ for some i, j) [17], the evolution operator (2.2) can be discretized into r steps of size $\Delta t = t/r$ as follows:

$$U(t) \approx \left(\prod_{k=1}^{K} e^{-iH_k \Delta t} \right)^r . \tag{2.6}$$

The operator-norm error ϵ for a single step arising from the non-commutativity of the H_k can be bounded by the following expression:

$$\epsilon \leq \frac{(\Delta t)^2}{2} \sum_{i \neq j} \|[H_i, H_j]\| + \mathcal{O}\left((\Delta t)^3\right), \tag{2.7}$$

where $\|.\|$ stands for the spectral norm and the global error over the whole simulation time t scales as [17]:

$$\left\| e^{-iHt} - \left(\prod_{k=1}^{K} e^{-iH_k \Delta t} \right)^N \right\| \leq \frac{t^2}{2r} \sum_{i<j} \|[H_i, H_j]\| . \tag{2.8}$$

The accuracy can be improved by increasing the order of the Suzuki-Trotter decomposition. One possible improvement to the Lie-Trotter formula (2.6) is symmetrization, which leads to the second-order formula:

$$U(t) \approx \left(\prod_{k=1}^{K} e^{-iH_k \Delta t/2} \right)^r \left(\prod_{k=K}^{1} e^{-iH_k \Delta t/2} \right)^r , \tag{2.9}$$

where the error scales as $\mathcal{O}\left((\Delta t)^2 t\right)$ [17]. Higher $2m$-th order approximations can then be constructed recursively using Suzuki's symmetrized product formula method [18, 19].

Both shown formulas for the Trotterized evolution operator (2.6), (2.9) require a time-independent Hamiltonian, and each Trotter step is of the same length. These assumptions are sufficient for our purposes. However, more advanced Trotterization methods have recently been developed that allow for time-dependent Hamiltonians and adaptive Trotter step lengths [20, 21], which may be useful in the future for constructing smaller quantum circuits.

2.4 Circuit implementation of CTQWs

The implementation of continuous-time quantum walks (CTQWs) on quantum computers involves constructing a quantum circuit that simulates the time evolution of a quantum state according to the system's Hamiltonian. First, the Hamiltonian matrix is expressed as a sum of tensor products of Pauli operators, where each term corresponds to a local interaction. For example, if we consider the Hamiltonian of a 4-cycle graph C_4, its matrix and Pauli decomposition is given by:

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \rightarrow H = I \otimes X + X \otimes X, \tag{2.10}$$

where I is the identity operator and X is the Pauli-X operator. If the total time evolution t is discretized into r small time steps $\Delta t = t/r \ll 1$ as described in Section 2.3.1, the corresponding time evolution then can be expressed as:

$$U(t) = e^{-iHt} \approx e^{-i(I\otimes X)\Delta t} e^{-i(X\otimes X)\Delta t}. \tag{2.11}$$

Therefore, given an initial state, final evolution time t, and a specific number of Trotter steps r, the corresponding quantum circuit can be constructed by considering the type of interaction appears in the substeps of the Trotterization such as $I \otimes X$ and $X \otimes X$. As an illustrative example, consider a 4-cycle graph C_4 with initial state $|00\rangle$, final time $t = 1$ and 10 Trotter steps. A single Trotter step in this case may be implemented as shown in Fig. 2.2.

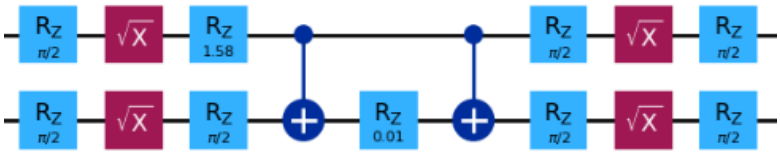


Figure 2.2: Example of a single Trotter step in a quantum circuit for a 4-cycle graph C_4, with an initial state $|00\rangle$, evolution time $t = 1$, and 10 Trotter steps. The circuit includes gates such as CNOT, $\sqrt{X}$, and R_Z, used to implement the Hamiltonian dynamics

2.5 Excitation transfer

In CTQWs, excitation transfer refers to the propagation of a quantum excitation (a.k.a. walker) through a graph, moving from one vertex to another. This process is governed by the Hamiltonian of the system and can be described by the probability distribution for finding the excitation at different vertices over time. As mentioned earlier, a key motivation for studying excitation transfer comes from photosynthetic energy transfer, where the efficiency of energy transport is crucial for the overall performance of the system [3].

A particularly interesting phenomenon in excitation transfer is perfect state transfer (PST), where the excitation initially localized at vertex u is transferred to a target vertex v with unit probability at a specific time τ [22], i.e.,

$$|\langle v| e^{-iH\tau} |u\rangle|^2 = 1. \tag{2.12}$$

Two notable special cases of PST are: when $u = v$, the graph G is said to be *periodic*; and when PST occurs between all distinct pairs of vertices u and v, the graph exhibits *universal* PST. In simulations on quantum computers there will be no experiment completely satisfying the condition (2.12) due to the presence of noise. However, we can still observe the phenomenon of excitation transfer, where the probability of finding the walker at vertex v is approaches 1 at time τ. Thus we will be using condition for *high amplitude transfer* (HAT) as a measure of excitation transfer, which is defined as follows:

$$|\langle v| e^{-iH\tau} |u\rangle|^2 \geq \eta, \tag{2.13}$$

where η is a threshold value [23]. In this project, we use $\eta = 0.95$ as our threshold, as it captures a substantial portion of the excitation transfer dynamics while staying within practical limits for quantum simulations. The choice of η represents a trade-off between accuracy and noise: higher thresholds may yield more precise results but often require greater resources and longer simulation times. Depending on the graph or application, different values of η may be more suitable.

PST is closely related to the structure and size of the graph. Some examples of small simple graphs where PST occurs under CTQW are the complete graph with two nodes K_2 and the cycle graph with four nodes C_4. Furthermore, interestingly as shown by [24], the n-cube has PST for any $n \in \mathbb{N}$. On the other hand certain graphs like K_n do not exhibit PST for $n \geq 3$ [22]. This raises the question of whether it is possible to enhance excitation transfer in graphs that do not naturally exhibit PST, and whether, from a practical standpoint, the time required for excitation transfer can be reduced in graphs that already PST exhibit. One possible solution was shown for K_n graphs by [25], where the authors showed that PST is enabled by simply removing the edge between initial and target vertex. We propose two possible solutions to enhance excitation transfer in graphs. The first is to add additional edges to the graph by introducing internal connections between vertices. This approach is based on the idea that increasing the graph's connectivity can facilitate excitation transfer. The second approach is to modify the system's Hamiltonian by introducing disorder, which can also enhance excitation transfer. This is a common technique in quantum mechanics, where the Hamiltonian is often modified to include external potentials or interactions that influence the system's dynamics.

2.6 Anderson localization

First stated in P. W. Anderson's seminal paper in 1958 [26], *Anderson localization* is a phenomenon in condensed matter physics where the wave

function of a particle becomes localized in a disordered medium, leading to the absence of diffusion. This phenomenon arises in systems with disorder, where the interference of multiple scattering paths suppresses wave propagation.

The approach to simulating Anderson localization on quantum computers is to use a disordered Hamiltonian, which can be represented as a sum of two terms: the hopping term and the disorder term. The hopping term H_h describes the movement of the walker between adjacent vertices, which is identical to the adjacency matrix of the graph. The disorder term H_d introduces random on-site energies ϵ_i at each vertex, which can be represented as a real diagonal matrix with random values along the diagonal. The total Hamiltonian can be expressed as follows:

$$H = H_h + H_d = A + \sum_{i=1}^{N} \epsilon_i \left|i\right\rangle \left\langle i\right|, \tag{2.14}$$

with $i \in \{1, 2, \ldots, N\}$ and N being the number of vertices in the graph [27].

By implementing the disordered Hamiltonian (2.14) on a quantum computer, one can observe how disorder affects excitation transfer, with the walker's probability distribution exhibiting exponential localization under the effect of disorder as illustrated in Fig. 2.3. Another measure used to quantify the localization is the *Inverse Participation Ratio* (IPR), which is defined as follows:

$$\text{IPR} = \sum_{i=1}^{N} |\psi_i|^4, \tag{2.15}$$

where $\psi_i = \langle i|\psi\rangle$ is the amplitude of the wave function at site i, and N is the total number of sites in the system [28]. The IPR provides a measure of how spread out the wave function is over the lattice. For a completely delocalized state, where the wave function is uniformly distributed over all N sites, the IPR approaches $1/N$, indicating low localization. Conversely, for a perfectly localized state concentrated at a single site, the IPR approaches 1, signifying maximal localization. Thus, higher IPR values indicate stronger localization, making it a useful diagnostic tool for detecting Anderson localization in quantum systems.

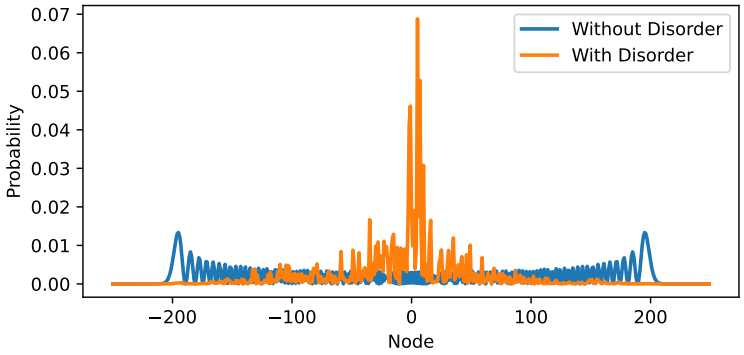


Figure 2.3: Numerical simulation of the difference in probability distributions of a CTQW on a line after $t = 100$, comparing the case without disorder to the case with disorder, where the disorder was generated using a uniform distribution with $\epsilon \in [-0.5, 0.5]$, where only one disorder profile was used.

2.7 Variational Quantum Algorithms

Variational quantum algorithms (VQAs) are a class of quantum algorithms that leverage the principles of quantum mechanics to solve optimization problems. They are particularly well-suited for near-term quantum computers, which are characterized by limited qubit counts and noise. VQAs typically defined by three main components: a cost function, an ansatz, and an optimizer. The cost function encodes the solution to the problem we want to solve, while the ansatz is a parameterized quantum circuit that generates a trial state. Lastly, classical optimizers are used to adjust the parameters $\boldsymbol{\theta}$ of the ansatz in order to minimize the cost function $C(\boldsymbol{\theta})$ in the optimization tasks

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} C(\boldsymbol{\theta}), \tag{2.16}$$

where $\boldsymbol{\theta}^*$ is the optimal set of parameters [29].

2.7.1 Cost function

The cost function, also referred to as a loss function, is a mathematical expression that maps values of trainable parameters $\boldsymbol{\theta}$ to a real number. The cost then defines a hypersurface in the parameter space, where the goal of the optimizer is to find the global minima. The cost function can be defined as:

$$C(\boldsymbol{\theta}) = f(\{\rho_k\}, \{O_k\}, U(\boldsymbol{\theta})), \tag{2.17}$$

where f is a function, $U(\boldsymbol{\theta})$ is a prameterized unitary, $\boldsymbol{\theta}$ is a vector of both discrete and continuous parameters, $\{\rho_k\}$ are input states from training set and $\{O_k\}$ refer to a set of observables. The cost function can then be written in a form as:

$$C(\theta) = \sum_k f_k \left(Tr \left[O_k U(\boldsymbol{\theta}) \rho_k U^\dagger(\boldsymbol{\theta}) \right] \right), \tag{2.18}$$

for some set of functions $\{f_k\}$ [29].

As given by [29], the cost function must satisfy desirable criteria, such as being 'faithful', meaning the minimum of $C(\boldsymbol{\theta})$ corresponds to the solution of the problem, and being able to 'efficiently estimate' $C(\boldsymbol{\theta})$ by performing measurements on a quantum computer. The second criteria, nowadays, means that in the era of NISQ devices, the quantum circuits must have small circuit depth.

2.7.2 Ansatz

In VQAs, the ansatz is essentially a parameterized quantum circuit designed to approximate the solution to a specific problem. Generally speaking, the form of the ansatz is problem dependent, meaning that the ansatz must be tailored to the specific problem being solved. For a cost function $C(\boldsymbol{\theta})$ given in a form in Eq. (2.18), the ansatz is a unitary operator $U(\boldsymbol{\theta})$ that can be expressed as a product of N sequentially applied unitaries:

$$U(\boldsymbol{\theta}) = U_N(\boldsymbol{\theta}_N)U_{N-1}(\boldsymbol{\theta}_{N-1})\dots U_1(\boldsymbol{\theta}_1), \tag{2.19}$$

with

$$U_l(\theta_l) = \prod_m e^{-i\theta_m H_m} W_m, \tag{2.20}$$

where H_m is a Hermitian operator, W_m is an unparameterized unitary operator and θ_l is the l-th element in $\boldsymbol{\theta}$ [29].

2.7.3 Optimizers

The final component of a variational quantum algorithm is the classical optimizer, which iteratively updates the parameters $\boldsymbol{\theta}$ of the ansatz to minimize the cost function $C(\boldsymbol{\theta})$. Optimizers play a crucial role in navigating the complex, often non-convex landscape of quantum cost functions. They can be broadly categorized into gradient-based and gradient-free methods, depending on whether they require the computation of derivatives. The choice of optimizer can significantly affect the performance, convergence speed, and resource requirements of the algorithm, especially given the noise and resource constraints of near-term quantum devices [29]. Below, we briefly describe three commonly used optimizers in the VQA context.

ADAM

The ADAM optimizer (Adaptive Moment Estimation) is a gradient-based method that uses momentum and adaptive learning rates. It maintains estimates of first and second moments of the gradient, which makes it robust to noisy or sparse gradient information. ADAM is widely used in machine learning and VQAs due to its versatility and low memory overhead [30]. In a VQA the gradient must be estimated (e.g. via the parameter-shift rule), so ADAM typically requires two or more circuit evaluations per parameter update.

It gradually descends the cost landscape by accumulating "velocity," which helps escape shallow local minima [30]. However, ADAM can struggle on very flat or highly non-convex landscapes (e.g. quantum "barren plateaus") where gradients vanish. It may take many iterations (and circuit calls) to converge, and its performance depends sensitively on hyperparameters (learning rate, decay factors, etc.).

COBYLA

The COBYLA (Constrained Optimization BY Linear Approximations) is a derivative-free optimizer that builds a linear model of the objective in a small "trust region" around the current point. It is designed for problems where gradients are unavailable or costly. At each iteration, COBYLA evaluates the cost at one new point and updates parameters by solving a linear approximation within a shrinking region. This means each step uses only one quantum-circuit evaluation, which can greatly reduce the total query count. In practice, COBYLA is recommended for smooth, noise-free VQA problems when the number of evaluations must be minimized [31]. It converges quickly in many cases, but it is fundamentally a local search. COBYLA tends to follow the nearest downhill direction and may converge to the closest local minimum. Its performance depends on the initial trust-region size and it can behave erratically if the cost landscape is noisy or highly irregular.

Bayesian optimization

Bayesian optimization (BO) is a global, sample-efficient method for optimizing black-box functions $f(x)$. BO is typically applied to problems of the form

$$\max_{x \in \mathbb{X}} f(x), \tag{2.21}$$

where $\mathbb{X}$ is the set of all possible parameters x, and is most effective in problems with 20 or fewer dimensions.

BO operates by fitting a statistical model, typically a Gaussian Process (GP), to past evaluations of the cost function. This model predicts both the mean and uncertainty of the objective at any point in the parameter space. An acquisition function is then used to balance exploration (sampling uncertain regions) and exploitation (sampling promising areas). Each BO iteration typically involves a single new circuit evaluation, along with the classical overhead of updating the GP and optimizing the acquisition function [32].

As a result, BO can often identify high-performing variational parameters with significantly fewer quantum-circuit calls than naive search or strictly local methods. Moreover, BO has proven effective in tuning hyperparameters in machine learning algorithms and deep neural networks [33].

2.8 Quantum error mitigation and suppression techniques

One of the current hardest tasks for quantum computing is to achieve reliable results predicted by theory on current quantum computers due to the ever-occuring noise. The noise can be caused by various sources, such as decoherence, gate errors, and measurement errors. These errors can significantly affect the performance of quantum algorithms and lead to incorrect results. Therefore, it is crucial to develop techniques for mitigating the effects of noise in quantum computations.

There are two main approaches to deal with noise in quantum computing: *quantum error correction* (QEC) on one hand and *quantum error mitigation* (QEM) and *quantum error suppression* (QES) on the other hand. QEC involves encoding logical qubits into a larger number of physical qubits, which allows for the detection and correction of errors without the requirement of direct measurement of the qubits. This approach is powerful but requires a large overhead in terms of qubit resources, giving it the ability to overcome the issue of noise in quantum computers in a long run. However, in this thesis we will focus on QEM and mainly on QES, which are more practical for current NISQ devices. QEM techniques aim to reduce the impact of noise by improving the final results of a quantum computation typically by executing the computation multiple times to cancel the noise on average or to extrapolate to a zero-noise limit [34]. QES techniques, on the other hand, aim to suppress the noise during quantum computations by modifying the quantum circuit or the measurement process [35]. These techniques are often simpler to implement and require fewer resources than QEC. Just to illustrate the difference between QEM and QES, we will describe one common QEM technique, called *zero-noise extrapolation* (ZNE) and one common QES technique, called *dynamical decoupling* (DD) and *Pauli twirling* (PT).

2.8.1 Zero-noise extrapolation

First introduced simultaneously by [36] and [37], ZNE is a quantum error mitigation technique that aims to reduce the impact of noise in quantum computations by extrapolating the results to a zero-noise limit. The basic idea behind ZNE is to run the same quantum circuit multiple times with different effective levels of processor noise and then use the results to extrapolate to the ideal case of zero noise [38].

Formally, the noise level of the circuit is parameterized by a fault rate λ, where for $\lambda = 0$ the noise is absent and for $\lambda = 1$ the true noise of the physical device is present. The expectation value is then a function of λ and can be expressed as:

$$E(\lambda) = \mathrm{Tr}\left[O\rho_\lambda\right], \tag{2.22}$$

where O is the observable and ρ_λ is the density matrix of the system at noise level λ. The ZNE extrapolation then consists of two steps: first, noise

scaling is run, where $E(\lambda)$ is measured at $m \in \mathcal{N}$ different values of $\lambda \geq 1$. The second step is the extrapolation, where $E(0)$ is estimated from the m previously measured values [38].

2.8.2 Dynamical decoupling

DD is a quantum error suppression technique used to protect qubits from decoherence caused by environmental noise. As the name suggests, DD suppresses the errors by the application of a sequence of pulses, which compose identity in the ideal case, that effectively decouple the system from the environment. Since its development in 1988 by [39], DD has been widely studied and applied in various quantum computing platforms, including superconducting qubits [40].

There are several types of DD sequences, each of which is most efficient when applied to a specific type of noise. The most common DD sequences are *Carr-Purcell-Meiboom-Gill* (CPMG) DD sequence, also known as XX DD sequence, and XY4 DD sequence. The CPMG pulse sequence is given as follows:

$$CPMG \equiv \frac{\tau}{2} - X - \tau - X - \frac{\tau}{2}, \tag{2.23}$$

where τ is the duration between the pulses and X represents the Pauli operator X. The CPMG sequence can suppress homogeneous dephasing along one axis, but is ineffective against noise coming from generic system-environment interactions. This is where the XY4 sequence comes into play, as a universal DD sequence. The XY4 sequence is defined as:

$$XY4 \equiv Y - \tau - X - \tau - Y - \tau - X - \tau, \tag{2.24}$$

where it utilizes both X and Y Pauli operators, basically applying π-rotations around $\hat{x}$ and $\hat{y}$ axes [35].

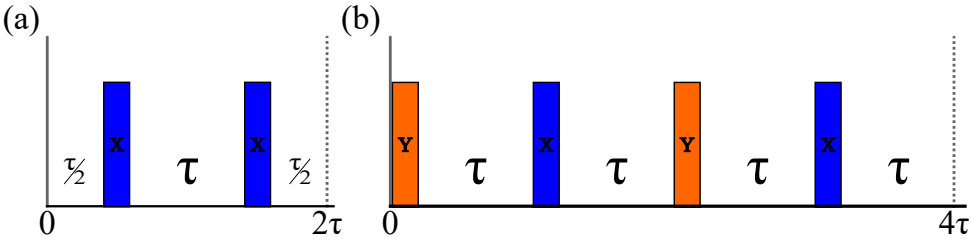


Figure 2.4: Examples of dynamical decoupling sequences: (a) CPMG sequence; (b) XY4 sequence.

2.8.3 Pauli twirling

PT is one specific quantum noise tailoring method of a quantum errors suppression techniques that are referred to as *twirling* or *randomized compiling* (RC). The basic idea behind PT is to transform the noise channel into a more manageable stochastic Pauli channel, where n-qubit Pauli channel corresponds to a quantum channel that applies random n-qubit Pauli operators on a

quantums state ρ according to some probability distribution. Mathematically formulated, the action of an n-qubit Pauli channel $\mathcal{P}^{(n)}$ on the state ρ can be expressed as:

$$P_{\vec{p}}^{(n)}(\rho) = \sum_{\vec{j},\vec{k}} p_{\vec{l},\vec{k}} X^{\vec{j}} Z^{\vec{k}} \rho \left(X^{\vec{j}} Z^{\vec{k}} \right)^{\dagger}, \tag{2.25}$$

where $0 \leq p_{\vec{j},\vec{k}} \leq 1$ and $\sum_{\vec{j},\vec{k}} p_{\vec{j},\vec{k}} = 1$ are the probabilities of applying the Pauli operators $X^{\vec{j}} Z^{\vec{k}}$ on the state ρ. The n-qubit Pauli operators are defined as:

$$X^{\vec{j}} := \sigma_x^{j_1} \otimes \sigma_x^{j_2} \otimes \cdots \otimes \sigma_x^{j_n}, \quad Z^{\vec{k}} := \sigma_z^{k_1} \otimes \sigma_z^{k_2} \otimes \cdots \otimes \sigma_z^{k_n}, \tag{2.26}$$

where $\vec{j} = (j_1, j_2, \ldots, j_n)$ and $\vec{k} = (k_1, k_2, \ldots, k_n)$ are binary vectors of length n that indicate the positions of the Pauli operators in the tensor product and σ_x and σ_z are the Pauli matrices [41].

The implementation of PT on a quantum computer involves inserting random Pauli gates before and after specific quantum operations, typically the gates that are most affected by noise, such that the net effect on the ideal unitary operation is preserved, but the noise channel is transformed into a Pauli channel. Specifically, for a given quantum gate U, PT introduces a pair of Pauli operators P and $P^{\dagger}$ such that $P'UP = U$, which implies that $PU = UP'$, where P is a Pauli operator randomly chosen from the set op Pauli operators $\{I, X, Y, Z\}$ and P' is computed accordingly. In practical settings, the quantum circuit is divided into cycles as shown on Fig. 2.5, basically segments of quantum gates, on which the PT is applied [42, 43].

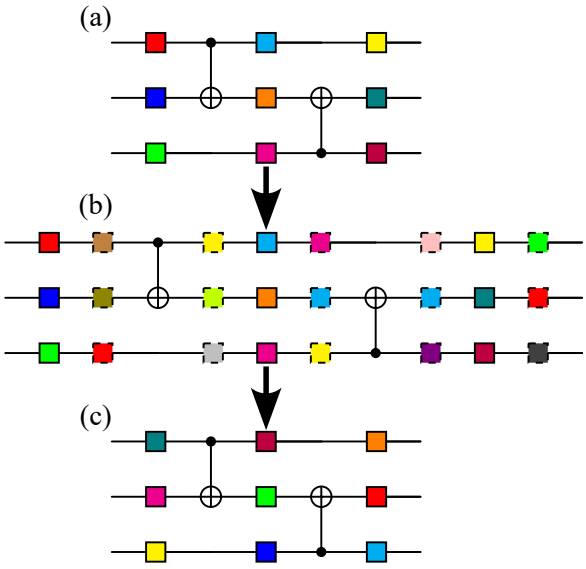


Figure 2.5: (a) The original circuit is divided into cycles, where each cycle consists of single-qubit gates and a CNOT gate. (b) Pauli twirling is applied, with Pauli operators inserted before and after the single-qubit gates. (c) A randomized circuit is obtained, in which the Pauli operators have been compiled into the single-qubit gates.

Chapter 3

Results

Having described the theoretical background of the excitation transfer and method to simulate it effectively on the quantum computer in the previous chapter, we will now present the results of our simulations. The simulations were performed on currently available IBM quantum computers, specifically on the *IBM Aachen*.

3.1 Enabling efficient excitation transfer on graphs

Simulating quantum walk processes, such as the continuous-time quantum walk (CTQW), on a quantum computer is a challenging task, particularly due to the pervasive presence of noise. When simulating CTQW on a quantum computer, the precision of the simulation depends on two key factors: the approximation error introduced by Trotterization and the *circuit depth* of the quantum circuit. The *circuit depth* refers to the number of sequential layers of quantum gates that must be applied, which directly impacts the overall fidelity of the computation due to accumulated noise and decoherence. Increasing the number of Trotter steps or using higher-order Trotterization methods reduces the approximation error but simultaneously increases the circuit depth, making the simulation more susceptible to noise.

To estimate the error due to this approximation, we numerically calculate the state ρ_{exact} obtained from the exact solution given by the evolution operator (2.2), and compare it with the state ρ_{Trotter} derived from the Trotterized evolution operator using the Lie-Trotter formula (2.6) or the Suzuki-Trotter formula (2.9). The error is then calculated as:

$$\epsilon = 1 - F\left(\rho_{exact}, \rho_{Trotter}\right), \tag{3.1}$$

where $F(\rho_{exact}, \rho_{Trotter})$ is the fidelity between the two states, which is defined as:

$$F(\rho_1, \rho_2) = Tr\left[\sqrt{\sqrt{\rho_1}\rho_2\sqrt{\rho_1}}\right]^2. \tag{3.2}$$

Lets consider the case of simulating CTQW on a cycle graph with 8 vertices C_8, we know that HAT appears on C_8 at time $t = 15.63 \approx 5\pi$[1]with probability at target vertex $P_{target} \geq 0.99$. However, simulating this specific graph-time combination on a quantum computer is not feasible due to the circuit depth

required. To maintain an approximation error from Trotterization around $\epsilon \approx 0.1$, we need to employ at least 97 Lie–Trotter steps or approximately 78 second-order Suzuki–Trotter steps. This results in circuit depths of approximately 3395 and 2900, respectively. Such deep circuits are well beyond the capabilities of current NISQ devices.

To reduce circuit depth, we consider alternative graphs where excitation transfer occurs at shorter times. We began with the 8-vertex cycle graph and introduced additional edges by connecting internal vertices. This approach is motivated by the idea that increasing a graph's connectivity can facilitate faster excitation transfer. Among all possible edge additions, we illustrate our findings with two representative examples:

- The first graph includes four additional edges.
- The second graph includes twelve additional edges.

These are shown in Fig. 3.1. From numerical simulations, we observe that HAT occurs at time $t \approx \pi/2$ with $P_{\text{target}} \geq 0.99$ for both graphs. The critical distinction lies in circuit depth: to achieve approximation error $\epsilon \leq 0.1$, we require 8 Suzuki–Trotter steps for both graphs, but the resulting circuit depths are 205 and 424, respectively.

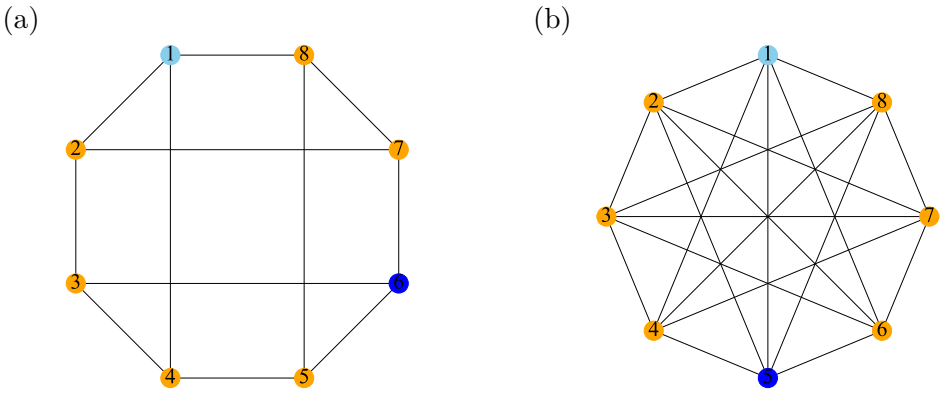


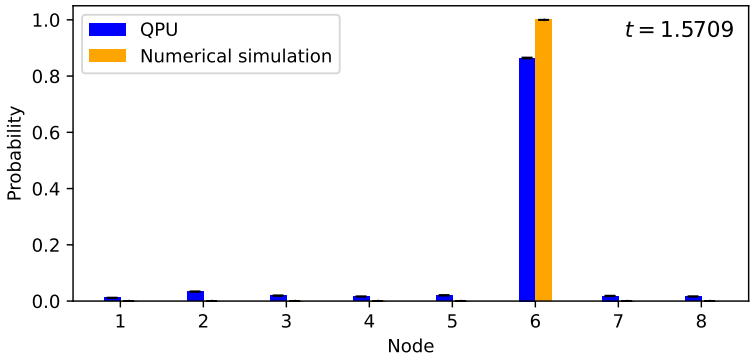
Figure 3.1: Graphs with added edges to improve excitation transfer. (a) Graph with four additional edges. (b) Graph with twelve additional edges. The initial excitation state is localized at the vertex highlighted in light blue, and the target vertex is shown in dark blue

The impact of reduced circuit depth is illustrated in Fig. 3.2, where the simulation results for the first graph clearly exhibit reduced noise effects compared to the second graph. The standard deviations are calculated from 20 different runs of the same circuit, and the error bars represent the standard deviation of the probability distribution.

The simulations employed a combination of quantum error mitigation and suppression techniques. The agreement between experimentally obtained and

[1]Note that the time t is given in natural units where $\hbar = 1$. Since our simulation uses a dimensionless evolution operator $U(t) = e^{-iHt}$ without the factor of $1/\hbar$, this time does not correspond to physical time but rather to a rescaled, unitless evolution parameter.

(a)



(b)

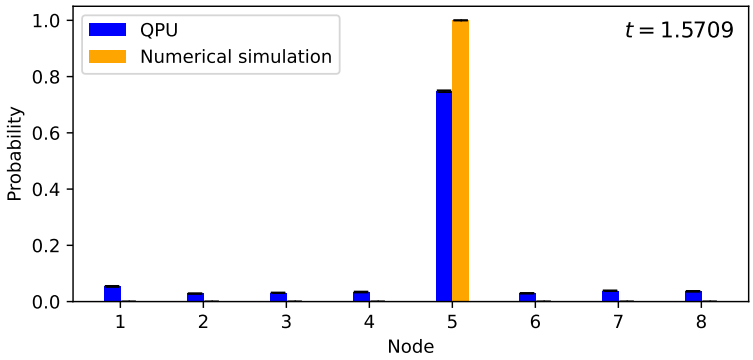


Figure 3.2: Probability distributions for the graphs from Fig. 3.1. (a) Graph with four additional edges. (b) Graph with twelve additional edges. Probability distribution (a) was simulated with using PT and (b) was simulated with using DD (XX) quantum error suppression technique.

expected probability distributions was evaluated using the Total Variation Distance (TVD), defined as:

$$D_{TV}\left(P_{num}, P_{exp}\right) = \frac{1}{2}\sum_{i} |P_{num}(i) - P_{exp}(i)|, \tag{3.3}$$

where $P_{num}(i)$ is the probability of finding the walker at vertex i from numerical simulation and $P_{exp}(i)$ is the expected probability distribution from numerical simulation [44]. The TVD values range from 0 to 1, where $D_{TV} = 0$ means that these distributions are identical and $D_{TV} = 1$ means that these distributions are completely different. We compared results using no mitigation, PT, and DD with XX and XY4 sequences. From Tab. 3.1, we observe that error mitigation techniques did yield some improvements, but the results are still far from the ideal case. The TVD values are significantly still in range of 0.1 to 0.3, indicating that the experimental results deviate from the expected probability distributions. This suggests that while the error mitigation techniques have some effect, they are not sufficient to fully correct the errors introduced by noise in the quantum circuit.

Graph	No Mitigation	PT	DD (XX)	DD (XY4)
1	0.157 ± 0.002	0.136 ± 0.001	0.234 ± 0.002	0.331 ± 0.002
2	0.319 ± 0.003	0.271 ± 0.002	0.252 ± 0.002	0.342 ± 0.003

Table 3.1: TVD between numerical and experimental probability distributions for the two graphs. The results are shown for three different QEM techniques: no mitigation, PT, and DD with XX and XY4 sequences.

3.2 Absence of excitation transfer on 1D lattice

Due to the presence of random disorder in the Hamiltonian, Anderson localization occurs, resulting in the suppression of excitation transfer, as theoretically discussed in Section 2.6. We simulated these dynamics on an eight-cycle graph C_8, where disorder was introduced by adding random on-site energies with strengths in the range $[-20, 20]$ to the Hamiltonian. The simulation time was set to $t = 1.2$ as it is the shortest time required for a CTQW on C_8 without disorder to reach the farthest vertex with a probability of $P_{farthest} \geq 0.1$. The simulation was implemented using Trotterization, and the associated approximation error was estimated using the same approach as described in the previous section via Eq. (3.1). The results are presented in Tab. 3.2, while the corresponding circuit depths are shown in Tab. 3.3.

The results demonstrate that the approximation error is significantly reduced when using the Suzuki-Trotter formula, particularly at the fourth order. However, this improvement comes at the cost of increased circuit depth, which becomes impractical for current quantum hardware. Consequently, we chose to use the Lie-Trotter formula with seven Trotter steps, yielding a reasonable approximation error of $\epsilon \approx 0.1$ and a circuit depth of 342.

The simulation results are depicted in Fig. 3.3, where the probability distribution exhibits the expected peak at the initial vertex and an exponential decay in both directions. We also attempted to apply error mitigation techniques; however, they did not lead to any significant improvement, consistent with observations in the previous section. The TVD values for simulations with and without error mitigation are shown in Tab. 3.4.

Even the best TVD value of $D_{TV} = 0.24$ indicates a notable discrepancy between the numerical and experimental probability distributions, highlighting the substantial impact of noise on the results. Potential solutions include employing more advanced error mitigation techniques or reducing circuit depth through more efficient Trotterization strategies, such as adaptive Trotter steps. Nonetheless, these results remain promising, demonstrating that CTQWs can still capture the effects of Anderson localization and excitation transfer on the C_8 graph, even under current hardware limitations, unlike discrete-time quantum walks (DTQWs), where we were previously unable to observe such effects on the C_8 graph [45], this limitation is primarily due to the additional circuit depth introduced by the extra dimension required to implement the coin operator.

steps	L-T	S-T (2nd order)	S-T (4th order)
1	0.96 ± 0.04	0.72 ± 0.20	0.55 ± 0.30
2	0.59 ± 0.23	0.51 ± 0.26	0.31 ± 0.23
3	0.46 ± 0.25	0.27 ± 0.27	0.12 ± 0.13
4	0.33 ± 0.25	0.23 ± 0.29	0.05 ± 0.07
5	0.28 ± 0.24	0.15 ± 0.22	0.02 ± 0.04
6	0.22 ± 0.30	0.01 ± 0.18	0.01 ± 0.01
7	0.10 ± 0.12	0.05 ± 0.11	0.01 ± 0.01

Table 3.2: Approximation error for the Trotterized evolution operator. The errors are calculated for Lie-Trotter (L-T) and Suzuki-Trotter (S-T) formulas of 2nd and 4th order for 1 to 6 Trotter steps. The errors are calculated as the average over 20 runs of the simulation.

steps	L-T	S-T(2nd order)	S-T (4th order)
1	60	111	495
2	113	202	988
3	169	304	1465
4	223	399	1974
5	284	502	2402
6	342	599	2898
7	388	691	3405

Table 3.3: Circuit depth on the *IBM Aachen* quantum computer for the Trotterized evolution operator. The depth is calculated for Lie-Trotter (L-T) and Suzuki-Trotter (S-T) formulas of 2nd and 4th order for 1 to 6 Trotter steps.

No Mitigation	PT	DD (XX)	DD (XY4)
0.24 ± 0.02	0.26 ± 0.02	0.26 ± 0.02	0.29 ± 0.02

Table 3.4: TVD between numerical and experimental probability distributions for the Anderson localization simulation. The results are shown for three different QEM techniques: no mitigation, PT, and DD with XX and XY4 sequences.

3.3 Noise-enhanced excitation transfer by using VQAs

We have previously demonstrated that excitation transfer can be enhanced by increasing the connectivity of the graph. In this section, we show that excitation transfer can also be improved by introducing disorder into the Hamiltonian. Specifically, we employ a VQA to optimize the parameters of a quantum circuit simulating CTQWs on a graph.

In this project, we propose a VQA that optimizes the disorder strengths (on-site energies) of the Hamiltonian of a graph to enhance excitation transfer. The algorithm, detailed in Algorithm 1, is designed to identify the optimal parameters that minimize the time t at which excitation transfer occurs with a probability satisfying HAT condition (2.13), from an initial vertex v_{start} to

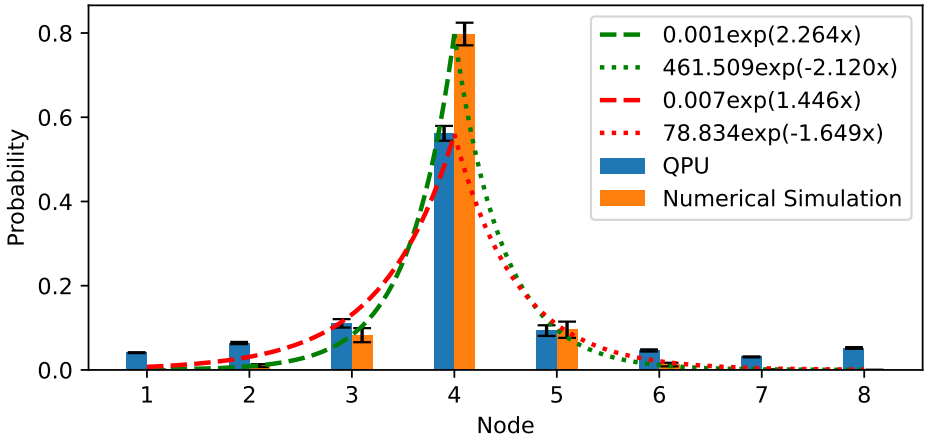


Figure 3.3: Simulation of Anderson localization on a C_8 graph with random on-site energies. The simulation was performed with 7 Trotter steps using the Lie-Trotter formula and without any error mitigation techniques. The initial state was localized at vertex 5 and the probability distribution is shown for time $t = 1.2$ and is averaged over 100 runs.

a target vertex v_{target}.

The convergence of the algorithm and the number of iterations required are heavily influenced by the choice of optimization strategy. In this work, we employ Bayesian Optimization (BO), which is particularly suitable for this task due to its effectiveness in exploring complex, high-dimensional parameter spaces that may contain multiple local minima.

To further reduce the number of iterations and overall runtime, we apply a secondary BO loop to optimize the hyperparameters of the VQA itself. These hyperparameters include the initial values for time and disorder strengths, the total number of VQA iterations, and the number of initial points sampled in the parameter space. This hyperparameter tuning procedure, illustrated in Algorithm 2, is performed using a simulator to avoid the high cost of running inefficient configurations on real quantum hardware. Once the optimal hyperparameters are found, the VQA is executed on a quantum computer.

We now describe in more detail the construction of the quantum circuits used to simulate the CTQW during each VQA iteration. As in previous sections, we employ Trotterization to approximate the time evolution operator $U(t)$, defined in Eq. (2.6). The number of Trotter steps $N_{Trotter}$ is determined based on a target approximation error ϵ, which depends on the selected Trotter scheme (Lie-Trotter or Suzuki-Trotter). A trade-off exists between approximation accuracy and circuit depth: reducing the number of Trotter steps lowers the circuit depth but increases the approximation error. We choose ϵ accordingly to balance fidelity and feasibility.

To further mitigate noise, we PT, a quantum error suppression technique. PT is applied by inserting random Pauli gates before and after each Trotter

Algorithm 1 VQA for excitation transfer

Input Hamiltonian of graph H, initial vertex v_{start}, target vertex v_{target}
Randomly initialize parameters: time t and disorders $\boldsymbol{\theta}$, where $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$ are the disorder strengths
while convergence criteria not met **do**
  Compute the minimum number of Trotter steps $N_{Trotter}$ required to achieve approximation error ϵ using either Lie-Trotter or Suzuki-Trotter formula.
  Construct and execute the quantum circuit with $N_{Trotter}$ Trotter steps and the current parameters
  Evaluate the total variation distance (TVD) between the resulting and expected probability distributions
  if TVD exceeds the threshold **then**
    Add a penalty to the cost function
  end if
  Update parameters using BO
end while
Output: Minimal time and corresponding disorders $\boldsymbol{\theta}$

Algorithm 2 Hyperparameter Tuning for VQA

Input: Search ranges for hyperparameters (initial time, disorder strengths, number of VQA iterations, number of initial points)
for each tuning iteration **do**
  Execute Alg. 1 using sampled hyperparameters
  Evaluate performance via the TVD, the minimal achieved transfer time t, and the number of iterations until convergence
  Update hyperparameters using BO
end for
Output: Optimal hyperparameters that yield the best VQA performance

step, as described in Section 2.8. The resulting circuit is then executed on a quantum processor to evaluate the excitation transfer probability.

Having described the VQA and the construction of the quantum circuit, we now present the results of our simulations. We demonstrate both the creation of HAT in scenarios where it was previously not possible, and the acceleration of excitation transfer by introducing disorder into the Hamiltonian.

In the first case, we considered the 4-cycle graph C_4, with the target node located adjacent to the initial node. By tuning the hyperparameters using Algorithm 2, we identified the optimal initial parameters for the VQA: 13 initial points[2], 60 iterations, a time bound of $t \in [3.31, 3.99]$, and disorder strengths in the range of $[-0.94, 2.66]$. The VQA converged to optimal parameters after 60 iterations, yielding an excitation transfer time of $t_{\text{optimal}} = 1.83$ and disorder strengths $\boldsymbol{\theta} = (-0.05, -0.11, 2.90, 2.80)$. The corresponding probability distribution, obtained using the evolution operator (2.2) with the Hamiltonian's diagonal elements given by $\boldsymbol{\theta}$, is shown in Fig. 3.4. The

probability of finding walker at the target vertex 2 is $P_{target} = 0.97$, which satisfies the HAT condition (2.13). This result confirms that excitation transfer can be enhanced by introducing disorder into the Hamiltonian, even between nodes where such transfer is not possible without disorder.

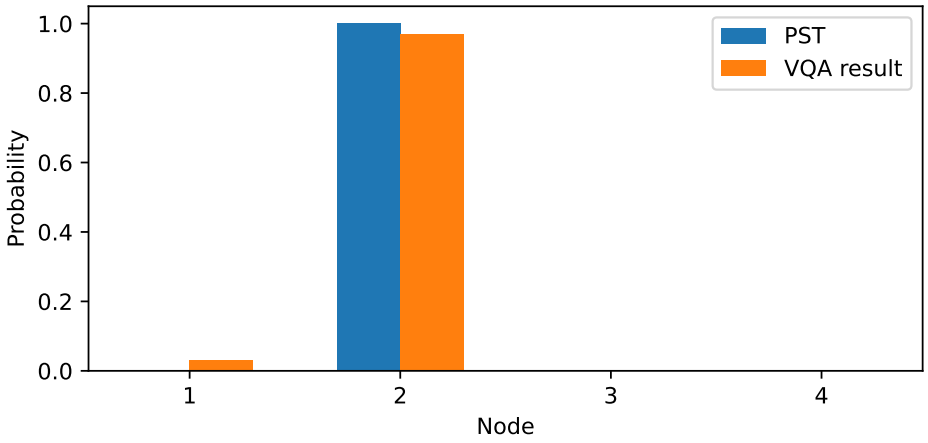


Figure 3.4: Probability distribution for a CTQW on a 4-cycle graph C_4 with disorder strengths $\boldsymbol{\theta} = (-0.05, -0.11, 2.90, 2.80)$ at time $t = 1.83$. The initial state is localized at vertex 1 and the target vertex is vertex 2.

In the second case, we examined the 8-cycle graph C_8, with the target node located directly opposite the initial node. Numerical simulations using the evolution operator (2.2) indicate that the shortest excitation transfer with $P_{target} \geq 0.99$ occurs at approximately $t \approx 12\pi$. Using the same procedure as in the previous case, we tuned the VQA hyperparameters via Alg. 2 and found optimal initial values: 12 initial points, 103 iterations, a time bound of $t \in [3.1, 6.9]$, and the disorder strengths within the range $[0.11, 2.09]$.

The VQA converged after 103 iterations, yielding $t_{optimal} = 6.02$ and disorder strengths $\boldsymbol{\theta} = (1.67, 0.67, 0.34, 1.53, 0.11, 0.93, 1.65, 0.12)$. The resulting probability distribution, shown in Fig. 3.5, achieved a target vertex probability of $P_{target} = 0.87$. For comparison, running the VQA on a simulator with the same hyperparameters produced optimal parameters $t_{optimal} = 6.31$ and $\boldsymbol{\theta} = (1.03, 0.87, 0.43, 1.02, 0.72, 0.60, 0.93, 0.43)$ with a significantly improved $P_{target} = 0.99$. This discrepancy highlights the effect of noise on quantum hardware, which degrades the excitation transfer performance compared to the simulator.

Although both results demonstrate the viability of disorder-assisted excitation transfer, the accuracy of the resulting probability distributions remains limited compared to scenarios without disorder. This limitation primarily stems from the constraints imposed on the hyperparameter tuning. Our primary goal was to identify optimal hyperparameters for the VQA using

[2]In Bayesian optimization, initial points refer to the set of parameter configurations evaluated at the beginning of the optimization process, typically chosen randomly or via space-filling designs, to initialize the surrogate model before iterative optimization begins.

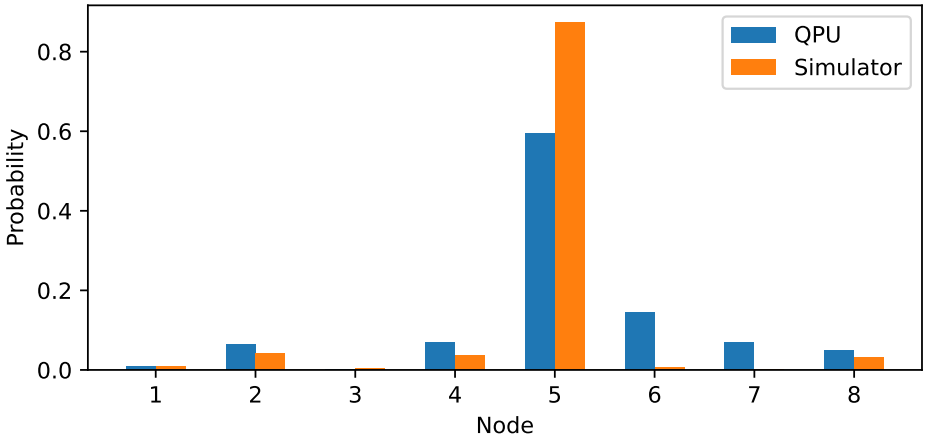


Figure 3.5: Comparison of probability distributions for a CTQW on an 8-cycle graph C_8 with parameters obtained from a VQA run on a quantum computer and a simulator. The initial state is localized at vertex 1 and the target vertex is vertex 5.

a limited number of iterations, due to restricted quantum computer runtime. However, extending the tuning process with more iterations could yield parameters that significantly improve excitation transfer probability.

To illustrate this, we conducted an additional simulation using a VQA with 20 initial points, 300 iterations, a time bound of $t \in [3.0, 3.5]$, and disorder strengths in the range of $[-2.0, 2.0]$. This simulation, executed on a simulator, produced optimal parameters $t_{\text{optimal}} = 5.99$ and $\boldsymbol{\theta} = (0.52, -1.10, -1.11, -0.96, 0.49, 1.82, 2.0, 1.78)$, yielding $P_{target} = 0.94$ and a TVD $D_{TV} = 0.06$ compared to the perfect state transfer distribution. As shown in Fig. 3.6, this result is significantly better than the one obtained on the quantum computer, indicating that longer hyperparameter tuning runs can indeed lead to improved excitation transfer.

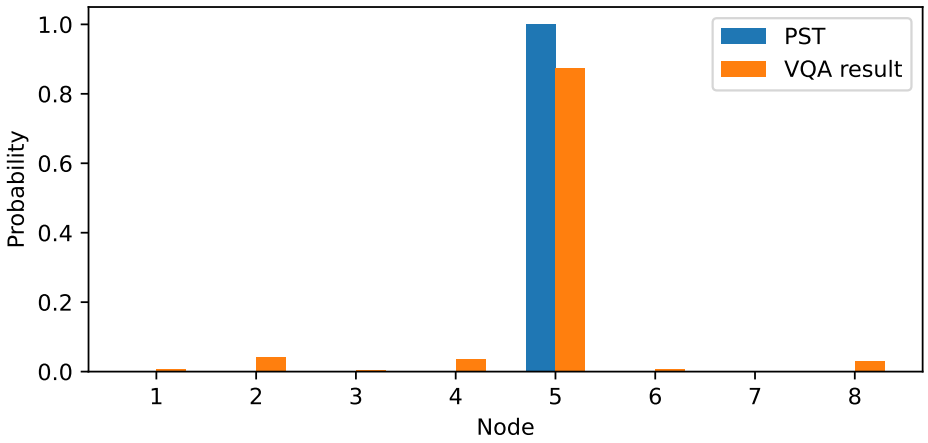


Figure 3.6: Resulting probability distribution for a VQA on an 8-cycle graph C_8 with disorder strengths $\boldsymbol{\theta} = (0.52, -1.10, -1.11, -0.96, 0.49, 1.82, 2.0, 1.78)$ at time $t = 5.99$. The initial state is localized at vertex 1 and the target vertex is vertex 5. This result was obtained using a simulator with more VQA iterations than in Fig. 3.5.

Chapter 4

Conclusion

The main goal of this project was to investigate efficient practical implementation of a CTQWs and utilize it to simulate excitation transfer and Anderson localization on quantum computers. For this purpose, we employed the Trotterization method to approximate the evolution operator and constructed quantum circuits that simulate CTQWs on graphs. We also explored various quantum error mitigation and suppression techniques to reduce the noise effects on the quantum circuits. Finally VQA was developed to optimize the parameters of the quantum circuit to enhance excitation transfer by introducing disorder to the Hamiltonian.

Building on the theoretical foundations presented in Chapter 2, Chapter 3 presents the results of our simulations. First, we demonstrate that excitation transfer can be enhanced by increasing the connectivity of the graph. Notably, we show that the transfer time, time at which the excitation transfer occurs, can be significantly reduced by adding edges to the graph. This not only improves transfer efficiency but also reduces circuit depth, enabling more practical simulations on existing quantum devices.

Second, we simulate Anderson localization on a cycle graph with random on-site energies and demonstrate that the excitation transfer is suppressed due to the presence of disorder in the Hamiltonian. Furthermore, we show clear evidence of localization on a probability distribution.

Third, we develop a VQA that optimizes the parameters of the quantum circuit to enhance excitation transfer by introducing disorder into the Hamiltonian. Remarkably, we find that this approach can enable high-amplitude transfer (HAT) between nodes where no excitation transfer is possible without disorder. In some cases, it can also significantly accelerate the excitation transfer by leveraging this disorder.

However, it is important to note that the scalability of our simulations remains limited by the effects of quantum noise. While our results show that the phenomena of excitation transfer and Anderson localization can indeed be observed on current quantum hardware, noise still poses a significant challenge. Even with error mitigation techniques, simulations involving larger graphs or longer evolution times are currently not feasible.

Moreover, in the case of VQA, the number of diagonal elements in the Hamiltonian corresponds to the number of parameters to be optimized. For

larger graphs, this increased parameter count raises the risk of encountering a barren plateau, a known issue where the optimization landscape becomes flat and gradients vanish, making parameter optimization extremely difficult. This remains an active area of research in the field of variational quantum algorithms.

These findings highlight the potential of CTQWs as a powerful tool for simulating excitation dynamics on quantum computers, while also emphasizing the challenges posed by noise and optimization in practical implementations. Future work could focus on utilizing more advanced Trotterization methods, such as adaptive Trotter steps, to further reduce circuit depth and improve the efficiency of simulations. Additionally, creating hardware-aware designs of quantum circuits could help mitigate the effects of noise and enhance the performance of quantum algorithms.

Bibliography

[1] Zhaoyang Chen, Guanzhong Li, and Lvzhou Li. "Implementation of a continuous-time quantum walk on a sparse graph". In: *Physical Review A* 110.5 (Nov. 2024). ISSN: 2469-9934. DOI: 10.1103/physreva.110.052215. URL: http://dx.doi.org/10.1103/PhysRevA.110.052215.

[2] Xiaogang Qiang, Shixin Ma, and Haijing Song. "Quantum Walk Computing: Theory, Implementation, and Application". In: *Intelligent Computing* 3 (Jan. 2024). ISSN: 2771-5892. DOI: 10.34133/icomputing.0097. URL: http://dx.doi.org/10.34133/icomputing.0097.

[3] Masoud Mohseni et al. "Environment-assisted quantum walks in photosynthetic energy transfer". In: *The Journal of Chemical Physics* 129.17 (Nov. 2008). ISSN: 1089-7690. DOI: 10.1063/1.3002335. URL: http://dx.doi.org/10.1063/1.3002335.

[4] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6 (1982), pp. 467–488. ISSN: 1572-9575. DOI: 10.1007/BF02650179. URL: https://doi.org/10.1007/BF02650179.

[5] Lov K. Grover. *A fast quantum mechanical algorithm for database search*. 1996. arXiv: quant-ph/9605043 [quant-ph]. URL: https://arxiv.org/abs/quant-ph/9605043.

[6] Peter W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science* (1994), pp. 124–134. URL: https://api.semanticscholar.org/CorpusID:15291489.

[7] Julian Berberich and Daniel Fink. "Quantum Computing Through the Lens of Control: A Tutorial Introduction". In: *IEEE Control Systems* 44.6 (Dec. 2024), pp. 24–49. ISSN: 1941-000X. DOI: 10.1109/mcs.2024.3466448. URL: http://dx.doi.org/10.1109/MCS.2024.3466448.

[8] Robin J. Wilson. *Introduction to Graph Theory*. New York: Prentice Hall/Pearson, 2010. ISBN: 027372889X 9780273728894.

[9] Luca Razzoli, Paolo Bordone, and Matteo G A Paris. "Universality of the fully connected vertex in Laplacian continuous-time quantum walk problems". In: *Journal of Physics A: Mathematical and Theoretical* 55.26 (June 2022), p. 265303. ISSN: 1751-8121. DOI: 10.1088/1751-8121/ac72d5. URL: http://dx.doi.org/10.1088/1751-8121/ac72d5.

[10] Y. Aharonov, L. Davidovich, and N. Zagury. "Quantum random walks". In: *Phys. Rev. A* 48 (2 Aug. 1993), pp. 1687–1690. DOI: 10.1103/PhysRevA.48.1687. URL: https://link.aps.org/doi/10.1103/PhysRevA.48.1687.

[11] Edward Farhi and Sam Gutmann. "Quantum computation and decision trees". In: *Phys. Rev. A* 58 (2 Aug. 1998), pp. 915–928. DOI: 10.1103/PhysRevA.58.915. URL: https://link.aps.org/doi/10.1103/PhysRevA.58.915.

[12] Salvador Elías Venegas-Andraca. "Quantum walks: a comprehensive review". In: *Quantum Information Processing* 11.5 (July 2012), pp. 1015–1106. ISSN: 1573-1332. DOI: 10.1007/s11128-012-0432-5. URL: http://dx.doi.org/10.1007/s11128-012-0432-5.

[13] İskender Yalçınkaya. "Spreading and transport properties of quantum walks". PhD thesis. Sabanci University, Istanbul, 2016.

[14] Johann Ostmeyer. "Optimised Trotter decompositions for classical and quantum computing". In: *Journal of Physics A: Mathematical and Theoretical* 56.28 (June 2023), p. 285303. ISSN: 1751-8121. DOI: 10.1088/1751-8121/acde7a. URL: http://dx.doi.org/10.1088/1751-8121/acde7a.

[15] Andrew M. Childs et al. "Theory of Trotter Error with Commutator Scaling". In: *Physical Review X* 11.1 (Feb. 2021). ISSN: 2160-3308. DOI: 10.1103/physrevx.11.011020. URL: http://dx.doi.org/10.1103/PhysRevX.11.011020.

[16] Tatsuhiko N. Ikeda, Hideki Kono, and Keisuke Fujii. "Measuring Trotter error and its application to precision-guaranteed Hamiltonian simulations". In: *Physical Review Research* 6.3 (Sept. 2024). ISSN: 2643-1564. DOI: 10.1103/physrevresearch.6.033285. URL: http://dx.doi.org/10.1103/PhysRevResearch.6.033285.

[17] Tilock Sadhukhan. *Trotterization and Digital Quantum Simulation*. Jan. 2025. DOI: 10.13140/RG.2.2.12968.07689.

[18] Masuo SUZUKI. "General Decomposition Theory of Ordered Exponentials". In: *Proceedings of the Japan Academy, Series B* 69.7 (1993), pp. 161–166. DOI: 10.2183/pjab.69.161.

[19] Nathan Wiebe et al. "Higher order decompositions of ordered operator exponentials". In: *Journal of Physics A: Mathematical and Theoretical* 43.6 (Jan. 2010), p. 065203. ISSN: 1751-8121. DOI: 10.1088/1751-8113/43/6/065203. URL: http://dx.doi.org/10.1088/1751-8113/43/6/065203.

[20] Tatsuhiko N. Ikeda et al. "Minimum Trotterization Formulas for a Time-Dependent Hamiltonian". In: *Quantum* 7 (Nov. 2023), p. 1168. ISSN: 2521-327X. DOI: 10.22331/q-2023-11-06-1168. URL: https://doi.org/10.22331/q-2023-11-06-1168.

[21] Hongzheng Zhao et al. *Making Trotterization adaptive and energy-self-correcting for NISQ devices and beyond*. 2023. arXiv: 2209.12653 [quant-ph]. URL: https://arxiv.org/abs/2209.12653.

[22] Vivien M. Kendon and Christino Tamon. "Perfect state transfer in quantum walks on graphs". English. In: *Journal of Computational and Theoretical Nanoscience* 8.3 (Mar. 2011). Copyright © 2011 American Scientific Publishers., pp. 422–433. ISSN: 1546-1955. DOI: 10.1166/jctn.2011.1706.

[23] K. Barr et al. *Periodicity and perfect state transfer in quantum walks on variants of cycles*. 2013. arXiv: 1204.5937 [quant-ph]. URL: https://arxiv.org/abs/1204.5937.

[24] Matthias Christandl et al. "Perfect State Transfer in Quantum Spin Networks". In: *Physical Review Letters* 92.18 (May 2004). ISSN: 1079-7114. DOI: 10.1103/physrevlett.92.187902. URL: http://dx.doi.org/10.1103/PhysRevLett.92.187902.

[25] Sougato Bose et al. *Communication in XYZ All-to-All Quantum Networks with a Missing Link*. 2008. arXiv: 0808.0748 [quant-ph]. URL: https://arxiv.org/abs/0808.0748.

[26] P. W. Anderson. "Absence of Diffusion in Certain Random Lattices". In: *Phys. Rev.* 109 (5 Mar. 1958), pp. 1492–1505. DOI: 10.1103/PhysRev.109.1492. URL: https://link.aps.org/doi/10.1103/PhysRev.109.1492.

[27] J. A. Izaac, J. B. Wang, and Z. J. Li. "Continuous-time quantum walks with defects and disorder". In: *Phys. Rev. A* 88 (4 Oct. 2013), p. 042334. DOI: 10.1103/PhysRevA.88.042334. URL: https://link.aps.org/doi/10.1103/PhysRevA.88.042334.

[28] Yingjian Liu et al. *Quantum Algorithms for Inverse Participation Ratio Estimation in multi-qubit and multi-qudit systems*. 2024. arXiv: 2405.03338 [quant-ph]. URL: https://arxiv.org/abs/2405.03338.

[29] M. Cerezo et al. "Variational quantum algorithms". In: *Nature Reviews Physics* 3.9 (Aug. 2021), pp. 625–644. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00348-9. URL: http://dx.doi.org/10.1038/s42254-021-00348-9.

[30] David Fitzek et al. "Optimizing Variational Quantum Algorithms with qBang: Efficiently Interweaving Metric and Momentum to Navigate Flat Energy Landscapes". In: *Quantum* 8 (Apr. 2024), p. 1313. ISSN: 2521-327X. DOI: 10.22331/q-2024-04-09-1313. URL: http://dx.doi.org/10.22331/q-2024-04-09-1313.

[31] T. Powers and R. M. Rajapakse. *Using Variational Eigensolvers on Low-End Hardware to Find the Ground State Energy of Simple Molecules*. 2023. arXiv: 2310.19104 [quant-ph]. URL: https://arxiv.org/abs/2310.19104.

[32] Peter I. Frazier. *A Tutorial on Bayesian Optimization*. 2018. arXiv: 1807.02811 [stat.ML]. URL: https://arxiv.org/abs/1807.02811.

[33] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. *Practical Bayesian Optimization of Machine Learning Algorithms*. 2012. arXiv: 1206.2944 [stat.ML]. URL: https://arxiv.org/abs/1206.2944.

[34] Zhenyu Cai et al. "Quantum error mitigation". In: *Rev. Mod. Phys.* 95 (4 Dec. 2023), p. 045005. DOI: 10.1103/RevModPhys.95.045005. URL: https://link.aps.org/doi/10.1103/RevModPhys.95.045005.

[35] Arefur Rahman, Daniel J. Egger, and Christian Arenz. "Learning how to dynamically decouple by optimizing rotational gates". In: *Physical Review Applied* 22.5 (Nov. 2024). ISSN: 2331-7019. DOI: 10.1103/physrevapplied.22.054074. URL: http://dx.doi.org/10.1103/PhysRevApplied.22.054074.

[36] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. "Error Mitigation for Short-Depth Quantum Circuits". In: *Phys. Rev. Lett.* 119 (18 Nov. 2017), p. 180509. DOI: 10.1103/PhysRevLett.119.180509. URL: https://link.aps.org/doi/10.1103/PhysRevLett.119.180509.

[37] Ying Li and Simon C. Benjamin. "Efficient Variational Quantum Simulator Incorporating Active Error Minimization". In: *Phys. Rev. X* 7 (2 June 2017), p. 021050. DOI: 10.1103/PhysRevX.7.021050. URL: https://link.aps.org/doi/10.1103/PhysRevX.7.021050.

[38] Tudor Giurgica-Tiron et al. "Digital zero noise extrapolation for quantum error mitigation". In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, Oct. 2020, pp. 306–316. DOI: 10.1109/qce49297.2020.00045. URL: http://dx.doi.org/10.1109/QCE49297.2020.00045.

[39] Lorenza Viola and Seth Lloyd. "Dynamical suppression of decoherence in two-state quantum systems". In: *Phys. Rev. A* 58 (4 Oct. 1998), pp. 2733–2744. DOI: 10.1103/PhysRevA.58.2733. URL: https://link.aps.org/doi/10.1103/PhysRevA.58.2733.

[40] Vinay Tripathi et al. "Suppression of Crosstalk in Superconducting Qubits Using Dynamical Decoupling". In: *Phys. Rev. Appl.* 18 (2 Aug. 2022), p. 024068. DOI: 10.1103/PhysRevApplied.18.024068. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.18.024068.

[41] Sumeet Khatri. *Pauli Channels*. Accessed: 2025-04-30. 2020. URL: https://sumeetkhatri.com/wp-content/uploads/2020/03/pauli.pdf.

[42] Joel J. Wallman and Joseph Emerson. "Noise tailoring for scalable quantum computation via randomized compiling". In: *Phys. Rev. A* 94 (5 Nov. 2016), p. 052325. DOI: 10.1103/PhysRevA.94.052325. URL: https://link.aps.org/doi/10.1103/PhysRevA.94.052325.

[43] Neelay Fruitwala et al. *Hardware-Efficient Randomized Compiling*. 2024. arXiv: 2406.13967 [quant-ph]. URL: https://arxiv.org/abs/2406.13967.

[44] Arnab Bhattacharyya et al. "On Approximating Total Variation Distance". In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*. IJCAI-2023. International Joint Conferences on Artificial Intelligence Organization, Aug. 2023, pp. 3479–3487. DOI: 10.24963/ijcai.2023/387. URL: http://dx.doi.org/10.24963/ijcai.2023/387.

[45] Jiří Viták. *Programování a simulace na kvantových počítačích*. Bachelor's thesis. Prague, Czech Republic, 2024. URL: http://hdl.handle.net/10467/116735.