UNIVERSIDAD DE CONCEPCIÓN

FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS

DEPARTAMENTO DE FÍSICA

NUMERICAL METHODS FOR VARIATIONAL QUANTUM AND QUANTUM-INSPIRED COMPUTING

Thesis presented in partial fulfillment of the requirements to receive the degree of Doctor of Philosophy in Physics

Author: Jorge Abraham Gidi Chomalí

Advisors: Dr. Aldo Delgado Hidalgo

Dr. Juan José García Ripoll

June 5, 2025

Concepción, Chile

Abstract

Quantum computation is a promising approach to solving problems in physics, optimization, and numerical simulation. Quantum algorithms leverage superposition and entanglement in an exponentially large Hilbert space to encode and solve problems that are intractable for classical computers. However, these algorithms require error-corrected quantum hardware, which is not yet available.

This thesis explores two alternative quantum-based approaches that promise to achieve better scalability than traditional algorithms in the near-term: Hybrid variational quantum algorithms (VQAs) and fully classical, quantum-inspired algorithms.

In the first part, we review and benchmark optimization methods well suited for VQAs, specifically the Simultaneous Perturbation Stochastic Approximation (SPSA). We introduce complex-variable extensions and improvements to the second-order and quantum natural implementations. Finally, we assess their performance across quantum control, quantum tomography, and the variational quantum eigensolver, providing practical guidelines for common VQA scenarios.

In the second part, we apply Matrix Product States (MPS) to solve time-dependent partial differential equations (PDEs). By integrating Hermite Distributed Approximating Functionals (HDAF) into the MPS framework, we develop a novel encoding to efficiently approximate differential operators with exponential precision. We benchmark this approach against traditional schemes for simulating the expansion of a levitated nanoparticle. The quantum-inspired numerical integration schemes deliver large memory savings and adequate runtimes even in presence of moderate chirping.

Resumen (Spanish)

La computación cuántica promete resolver problemas en física, optimización y simulación numérica. Los algoritmos cuánticos aprovechan la superposición y el entrelazamiento en un espacio de Hilbert exponencialmente grande para codificar y resolver problemas que serían intratables para los computadores clásicos. Sin embargo, estos algoritmos requieren de dispositivos cuánticos con corrección de errores, los cuales aún no están disponibles.

Esta tesis explora dos enfoques alternativos, basados en principios cuánticos, que prometen alcanzar un mejor escalamiento que los enfoques computacionales tradicionales en el corto plazo: los algoritmos híbridos variacionales cuánticos (VQAs) y los algoritmos clásicos de inspiración cuántica.

En la primera parte, se revisan y comparan métodos de optimización adecuados para VQAs, en particular el *Simultaneous Perturbation Stochastic Approximation* (SPSA). Se introducen extensiones en variable compleja y mejoras en sus variantes de segundo orden y de gradiente natural cuántico. Finalmente, se evalúa su desempeño en tareas como control cuántico, tomografía cuántica y el *variational quantum eigensolver*, entregando recomendaciones prácticas para escenarios típicos de optimización en VQAs.

En la segunda parte, se utilizan estados producto de matrices (MPS) para resolver ecuaciones diferenciales parciales (EDP) dependientes del tiempo. Al integrar los *Hermite Distributed Approximating Functionals* (HDAF) en el marco de MPS, se desarrolla una nueva codificación eficiente de operadores diferenciales con precisión

exponencial. Este enfoque se compara con métodos tradicionales para simular la expansión de una nanopartícula levitada. Los esquemas de integración de inspiración cuántica logran un considerable ahorro de memoria y tiempos de ejecución adecuados, incluso en presencia de *chirping* moderado.

Agradecimientos (Spanish)

Debo admitir que, a lo largo de mi vida, rara vez he sabido dónde estoy o hacia dónde voy. Mi camino en la física no ha sido una excepción, con giros y no linealidades. Si hoy llego a culminar un doctorado en física, es en gran medida gracias a quienes me han acompañado y sostenido en este recorrido.

Por eso, quiero comenzar agradeciendo a mi director de tesis, Aldo Delgado, por recibirme en su grupo de investigación y por apoyarme en cada etapa del proceso. También agradezco a mi amiga Bárbara, quien tiene buena parte de la culpa de que hoy trabaje en computación cuántica. Siempre agradeceré tu guía constante y tu amistad.

Una parte fundamental de este proceso fue mi estadía en el Instituto de Física Fundamental en Madrid. Allí tuve la suerte de conocer a personas extraordinarias, entre quienes quiero destacar especialmente a Juan José García Ripoll, cuya buena voluntad y vasto conocimiento marcaron mi formación. También quiero destacar a Paula García, mi mejor amiga y colega en España. Se suponía que trabajáramos a la par, pero siempre sentí que era yo quien aprendía de ti. Gracias por todas nuestras conversaciones, tanto de física como de anime y Taylor Swift. También agradezco a Luciano y Nicole, por hacer de mi paso por España una experiencia mucho más grata. No se imaginan cuánto significó su compañía, cariño y consideración.

Agradezco particularmente a quienes son coautores del trabajo que presento en esta tesis. Cada uno de ustedes ha sido una fuente de aprendizaje y ha contribuido directamente a construir el camino que hoy celebro. Asimismo, este documento no

habría sido posible sin la dedicación de mi comisión evaluadora, conformada, además de mis dos guías, por Dardo Goyeneche y Pablo Solano. Gracias por el tiempo y la experiencia prestada.

A lo largo de estos años, he tenido el privilegio de formar parte de varias comunidades académicas. Me siento afortunado de haber sido parte de los grupos Quantum Legion y QUINFOG. A quienes los conforman, gracias por su apoyo y por la constante red de aprendizaje que han significado. Además, soy un miembro orgulloso del Millenium Institute for Research in Optics (MIRO) y me alegra profundamente haber trabajado junto a otros integrantes. Gracias Pablo, Paulina, Agustín y Calu por su labor, y por permitirnos desarrollar la ciencia y la divulgación en Chile.

En esta misma línea, no puedo dejar de agradecer a todo el equipo que ha sido parte de las Escuelas de Computación Cuántica. Trabajar con ustedes cada verano ha sido desafiante, pero también una de las experiencias más gratificantes de mi vida.

La experiencia del doctorado fue la suma de pequeños momentos. A mis compañeros de oficina: gracias por hacer del trabajo mucho más ameno. Desde la oficina 305 en la CFM, pasando por el despacho 113 del IFF, hasta la 603 nuevamente en la CFM. Gracias en especial a Constanza, Francisco y Katherine por su compañía durante la última parte de este proceso. Y a Marco, que sin compartir oficina, hizo que mi trabajo en la CFM fuera mucho más agradable y humano. Agradezco también al personal de la Universidad de Concepción. En especial, a Julia Herrera, por su apoyo y gratas conversaciones, y a Nilsa García, por su infinita ayuda en los procesos del doctorado. También agradezco a Juan Carlos, Fidel y Marco por su amabilidad cotidiana.

Gracias a todas las personas que me brindaron su amistad y su tiempo a lo largo de estos años, sin excepción. Muchas alegraron mi día a día con gestos breves, pero significativos. Me resulta imposible nombrarlas a todas, pero quiero hacer una mención especial a Hugo, Alejandro, Danilo y Nicol, por su compañía en lo que ha sido este viaje fuera del hogar, desde mucho antes de entrar al doctorado. También

quiero mencionar a quienes me acogieron durante el caótico periodo que fue el retorno desde la pandemia: gracias Scarlett, Guillermo, Mariana, Camilo, Ayleen, Marcel y Aníbal. En este mismo espiritu, agradezco también a Pedro, Jorge, Nelson, Cote, Adheris, Tomás y Lucas. Cada uno de ustedes estuvo ahí cuando lo necesité.

A mi pana Arelly: gracias por ser como has sido conmigo. Por compartir risas y lágrimas, incluso con miles de kilómetros de por medio. Gracias también a Joaquín y Freddy por ir juntos en el pase VIP al infierno. Gracias Carla por acompañarme en la escritura de esta tesis, y por las risas infinitas.

No habría podido mantenerme cuerdo durante el doctorado si no fuera por los viajes a la montaña y quienes los compartieron conmigo. Gracias a mi cordada Chillaneja, y especialmente a Andrea: tu amistad y apañe han sido muy importantes para mí estos años.

Finalmente, a mi familia: gracias a mis padres, porque ambos han sido siempre un ejemplo de amor y compromiso. Si hoy hago un doctorado, es porque ustedes me enseñaron el valor del conocimiento y del trabajo. Gracias también a Vicky, Leyla, Cata y Nacho, porque desde sus formas únicas me enseñan y me inspiran constantemente a ser mejor. Finalmente, agradezco a todo el grupo de la familia unida, por su apoyo incondicional y por ser el espacio amoroso que son.

Funding

This work has been funded by ANID Chile through ANID-PFCHA/DOCTORADO-NACIONAL/2020-21202616. This work was also supported by the Millenium Institute for Research in Optics (MIRO), by the Ministry for Digital Transformation and of Civil Service of the Spanish Government through the QUANTUM ENIA project call - Quantum Spain project, and by the European Union through the Recovery, Transformation and Resilience Plan - NextGenerationEU within the framework of the Digital Spain 2026 Agenda. This research benefited from the Scientific Computing Area (AIC), SGAI-CSIC, and their assistance while using the DRAGO Supercomputer to perform simulations.

Contents

List of Figures

List of Tables

Part I

Introduction and Background

Chapter 1

Introduction

"One never notices what has been done; one can only see what remains to be done."

Marie Curie

1.1 Motivation

In the last century, quantum mechanics has fundamentally changed our understanding of microscopic systems. Unlike classical mechanics, which provides deterministic equations of motion, quantum mechanics introduces probabilistic principles that govern the behavior of small-scale phenomena. Two key features of quantum mechanics, superposition and entanglement, not only challenge our classical intuition but also enable new approaches to information processing and computation.

Quantum computing has become a promising approach for solving problems in physics, optimization, and numerical simulation. Unlike classical computing, which processes information using bits with values that are either 0 or 1, quantum computing uses quantum bits (qubits) capable of existing in superposition, allowing richer computational possibilities. Quantum entanglement further enables stronger correlations than those of classical systems, making some quantum algorithms more

efficient than their classical counterparts. However, the practical implementation of these algorithms requires large-scale, fault-tolerant quantum computers that still remain beyond our technological capabilities. These devices are subject to noise and decoherence, which hinder the quality of the results, making general practical quantum calculations unfeasible today.

Given these limitations, alternative strategies have been explored to take advantage of our knowledge of quantum mechanics and translate them into computational advantages in the near term. This thesis focuses on two approaches: Variational quantum algorithms (VQA) and quantum-inspired numerical methods.

VQAs are hybrid quantum-classical algorithms in which a quantum circuit evaluates an objective function, which is then optimized by a classical algorithm. These methods are practical for noisy intermediate-scale quantum (NISQ) devices since they require relatively shallow circuits, reducing the impact of noise. However, the optimization process depends on the ability of a classical optimizer to train a large number of parameters based on the values of objective functions that are inherently noisy due to the statistical nature of quantum measurements.

Quantum-inspired methods, on the other hand, use insights from quantum systems to develop new algorithms while running entirely on classical hardware. Among these methods, tensor networks such as matrix product states (MPS) have proven useful for compressing and efficiently simulating some large quantum systems. These techniques have also been adapted to tackle classical problems, such as solving partial differential equations (PDEs), where they may offer significant memory savings compared to traditional numerical methods.

This thesis explores these two approaches, focusing on (i) stochastic optimization for variational quantum algorithms and (ii) matrix product state methods for solving PDEs.

1.2 Research problem and proposed solutions

In VQAs, objective functions inherently contain noise due to hardware imperfections and the statistical nature of quantum measurements. Therefore, classical optimizers must be robust against noise and scale efficiently to high-dimensional parameter spaces. The Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm meets these requirements but strongly depends on hyperparameters that are challenging to fine-tune. Second-order and quantum natural gradient variants of SPSA mitigate this issue by adaptively scaling parameter updates using curvature or geometry-aware information, making them easier to tune. This thesis explores first-order, second-order, and quantum natural SPSA methods, introducing novel improvements and complex-variable extensions, and benchmarks their performance to provide practical guidelines for common VQA scenarios.

Similarly, accurately solving PDEs with limited computational resources is a ubiquitous challenge in physics and engineering. Matrix Product States (MPS) are a promising tool in this context, as they have the potential to provide exponential compression and efficiently represent large amounts of data. However, conventional discretization techniques for differential operators have limitations within the MPS framework: local finite differences are computationally efficient but introduce numerical dissipation, whereas global spectral methods offer higher accuracy but can be computationally expensive. To address this gap, we integrate Hermite Distributed Approximating Functionals (HDAF), which are identity resolutions based on Hermite polynomials modulated by a Gaussian envelope, into the MPS methodology. This approach yields highly accurate, yet locally confined differential operators that effectively improve the balance between computational cost and accuracy in MPS-based numerical PDE solvers.

1.3 Objectives and methodology

The main goal of this research is to improve classical numerical methods to help leverage quantum-based strategies that outperform state-of-the-art computational methods in the near term. Specifically, this work focuses on two goals: (i) to provide guidelines for the optimization of variational quantum algorithms (VQAs) by developing and benchmarking methods based on the Simultaneous Perturbation Stochastic Approximation (SPSA) and (ii) to provide new matrix product state (MPS)-based numerical solvers for partial differential equations (PDEs), with an improved cost-accuracy relation than current solvers, by integrating Hermite Distributed Approximating Functionals (HDAF) into their framework.

The specific objectives and their corresponding methodologies are described in the following.

1. Optimization methods for VQAs

1.1. Review existing SPSA methods for quantum applications

Review the literature on existing SPSA-based methods and how they have been used in the context of quantum applications, specifically VQAs. Consider common limitations and the standard techniques used to mitigate them.

1.2. Introduce complex-variable extensions to the SPSA methods

Employ Wirtinger calculus to reformulate gradient and Hessian approximations in the second-order SPSA (2SPSA) and quantum natural SPSA (QN-SPSA) to use complex-variable perturbations.

1.3. Benchmark the methods on representative VQAs problems

Use the Julia programming language [1] to develop an optimization library [2] and implement instances of canonical VQAs. These instances will be the variational quantum eigensolver (VQE, using Yao [3]), quantum control, and quantum state estimation, which are relevant and well documented in the

literature. Then, analyze the performance of SPSA methods based on the behavior of statistical indicators, particularly the mean, variance, median, and interquartile range. Explore standard selections of hyperparameters and strategies for mitigating problems.

1.4. Provide practical guidelines for applying SPSA methods in the context of VQAs

Classify the results for each application and separate them into two groups: (ii) vanilla methods, which include standard SPSA implementations, and (ii) improved methods, which incorporate enhancements such as blocking, resampling, alternative formulas for post-processing of Hessian matrices, and scalar preconditioning. Identify the best methods in each group, in terms of convergence, and characterize them in terms of quantum resources (number of evaluations), classical computational requirements, and reliability. Summarize the findings into practical recommendations for selecting and tuning SPSA variants.

2. Quantum-inspired numerical methods for solving PDEs

2.1. Review the usage of Hermite Distributed Approximating Functionals (HDAF) to approximate differential operators

Review the literature on HDAF theory and their numerical metaheuristics to approximate different kinds of differential operators.

2.2. Extend the MPS framework to encode operators within the HDAF formulation

Find analytical expressions to construct different HDAF operators in terms of efficient MPO representations.

2.3. Develop numerical integrators based on the operators provided by the HDAF-MPS encoding

Use and extend the Python [4] library SeeMPS [5] as a framework to nu-

merically implement analytical expressions for HDAF MPOs. Then, use the operators and MPS/MPO algebra to develop adequate numerical integration schemes based on these operators.

2.4. Benchmark the new techniques in physically relevant scenarios

Assess the efficiency and accuracy of these new methods, as well as traditional methods, against a one-step short-time evolution of the benchmark PDE problem: A levitated nanoparticle that expands on harmonic and anharmonic potentials. These problems are selected because they have features that challenge both traditional and MPS-based methods. For the harmonic case, in particular, an analytical solution exists that allows to validate the accuracy of the methods.

2.5. Compare the performance of the proposed methods with traditional numerical approaches

Select the methods that perform the best in the previous step, based on accuracy and efficiency, and analyze their behavior when solving a long-term evolution of the benchmark problems. Analyze the asymptotic behavior of the integrators by identifying scaling laws in memory and run-time.

1.4 Outline of the document

This thesis is structured as follows. Part I provides an overview of the research and defines the basic concepts used throughout the document: Chapter 2 presents a gentle introduction to quantum mechanics and its fundamental features. Chapter 3 covers the principles of quantum computing, introducing qubits, quantum gates, and quantum circuits in near- and long-term scenarios. Chapter 4 discusses quantum-inspired computing, focusing on matrix product states as an efficient classical representation of quantum systems, exploring their mathematical properties and the regimes where they can outperform current computational methods based on dense vectors.

The core contributions of this thesis are presented in Parts II and III, where advanced concepts are introduced as needed.

Part II of the thesis is dedicated to the study of stochastic optimization for variational quantum algorithms: Chapter 5 motivates the study of stochastic optimization algorithms in this context, emphasizing the challenges of optimizing quantum circuits. Chapter 6 formulates the optimization problems that arise in VQAs and the role of the Simultaneous Perturbation Stochastic Approximation (SPSA) in this context. Chapter 7 reviews existing SPSA-based methods and develops new extensions. Chapter 8 benchmarks the performance of these methods against representative VQAs. Finally, Chapter 9 summarizes the findings and presents practical guidelines for applying SPSA-based techniques in VQAs.

Part III focuses on the solution of partial differential equations (PDEs) using matrix product states (MPS): Chapter 10 motivates the use of quantum-inspired tools to solve PDEs, particularly matrix product states. Chapter 11 describes the benchmark problem of simulating a levitated nanoparticle expanding in harmonic and anharmonic potentials. Chapter 12 reviews the HDAF representation of differential operators, integrates them into the MPS framework, and applies them to develop numerical methods for the time evolution of PDEs. Chapter 13 benchmarks the performance of these new methods, comparing them with traditional approaches. Finally, Chapter 14 summarizes the behavior of our numerical methods.

Part IV concludes the thesis by summarizing the key findings and presenting potential directions for future research.

Chapter 2

Quantum Mechanics

"No phenomenon is a phenomenon until it is an observed phenomenon"

John Archibald Wheeler

2.1 The History of Quantum Mechanics

With just over a century of history, quantum mechanics is a broad and very relevant topic in modern physics. It was born in an attempt to address fundamental inconsistencies in classical mechanics, especially for systems at the atomic and subatomic levels. While classical mechanics is effective for large-scale objects, it fails to describe the behavior of very small systems.

2.1.1 The Old Quantum Theory

At the end of the 1800s, the laws of physics were thought to be almost fully understood. Newton's mechanics explained motion, Maxwell's equations described electricity and magnetism, and thermodynamics provided rules for heat and energy. However, there were still a few problems in which the predictions from classical me-

chanics were inconsistent with the experimental data. One such case is the ultraviolet catastrophe: Classical theory predicted that a perfect black body should emit more and more energy at higher frequencies, diverging to infinity.

In 1900, Max Planck solved this problem by assuming that energy could only be emitted or absorbed in discrete amounts, which he called quanta. He found that the energy of light had to be proportional to its frequency ν,

$$E = h\nu,$$

where h is Planck's constant [6]. At first, Planck just took this assumption as a mathematical trick to fit the experimental data.

In 1905, Albert Einstein used this idea to explain the photoelectric effect, where light shining on a metal would induce an electrical current. Classical theory predicted that a light of any frequency would free electrons from the metal, depending primarily on the light intensity. However, experimental data showed that the kinetic energy of the electrons had a lower threshold, depending on the frequency of light, and not its intensity [7]. To explain this phenomenon, Einstein suggested that light itself is made up of particles[1], each carrying energy $h\nu$ [8], and that electrons were only emitted if this energy exceeded a material-specific threshold. This idea introduced the concept of wave-particle duality, where light could behave both as a wave and as a particle depending on the situation. Einstein was awarded the Nobel Prize in Physics in 1921 "for his services to Theoretical physics, and especially for his discovery of the law of the photoelectric effect" [9].

Another major problem was the unexplained stability of atoms. Classical physics predicted that electrons orbiting a nucleus should constantly emit radiation, lose energy, and eventually fall to the nucleus, thus making atoms unstable. However, this is not the case. In 1913, Niels Bohr introduced a model of the hydrogen atom in which electrons could only occupy specific orbits with quantized energy levels [10].

[1] Later denominated *photons* by Gilbert N. Lewis.

This explained the spectral lines of hydrogen but did not have a justification.

In 1924, Louis de Broglie suggested a new idea: If light could behave as both a wave and a particle, maybe matter could also have wave-like properties, with a wavelength given by the inverse of its momentum [11],

$$\lambda = \frac{h}{p}.$$

This idea was later confirmed in electron diffraction experiments [12, 13], showing that particles could behave like waves.

One key consequence was the realization that quantum systems do not always have well-defined states. Instead, they can exist in superpositions of multiple states at once. For example, an electron in an atom is not simply in one specific orbit but in a superposition of different energy levels, and its behavior is only described probabilistically. This superposition principle would later become one of the cornerstones of quantum mechanics.

Although these ideas helped solve many problems, they were just a mix of quantum patches over classical mechanics with no clear underlying framework. A more complete theory was needed.

2.1.2 The New Quantum Theory

In the mid-1920s, scientists finally developed a consistent mathematical framework for quantum mechanics. Two different but equivalent formulations appeared: matrix mechanics and wave mechanics.

In 1925, Werner Heisenberg introduced what later developed into matrix mechanics [14]. In his approach, observable quantities were organized into arrays with an algebra that replaced the usual differential equations of classical mechanics. Max Born and Pascual Jordan soon recognized that these arrays obeyed the rules of matrix multiplication, and together with Heisenberg presented the full formalism in 1926 [15, 16]. The same year, inspired by de Broglie's wave idea, Erwin Schrödinger

introduced wave mechanics, which described quantum systems using the Schrödinger equation [17],

$$i\hbar\frac{\partial}{\partial t}\psi = H\psi,$$

where $\hbar = h/2\pi$ is the reduced Planck's constant and H is the energy operator of the system.

Here, the wave function ψ is the quantum state, and its squared magnitude $|\psi|^2$ gives the probability of finding a system in a particular configuration. This probabilistic interpretation, introduced by Max Born in 1926 [18], marked a huge difference with deterministic classical physics. In 1927, Heisenberg also formulated the uncertainty principle, which showed that certain pairs of properties, such as position and momentum, cannot be precisely determined at the same time [19].

In this context, another strange feature of quantum mechanics began to raise questions: the theory predicted that when two particles interact, their quantum states could become entangled. Although the term *entanglement* was introduced later by Schrödinger, who identified it as *the* characteristic trait of quantum mechanics [20], the concept was already explored in the thought experiment by Einstein, Podolsky and Rosen (EPR) in 1935 [21]. They considered a pair of particles prepared in such a state that measuring one particle would instantly influence the outcome of measuring the other, regardless of the distance between them[2]. Troubled by what Einstein called *Spukhaften Fernwirkung*[3], they proposed the existence of *hidden variables*, which would restore a deterministic and local description of reality. Here, *locality* refers to the principle that an object can only be influenced by its immediate surroundings, which means that physical effects cannot propagate arbitrarily fast. EPR argued that quantum mechanics was incomplete and that its probabilistic nature was due to a lack of knowledge about these hidden variables.

[2]Such an example is given in Section 2.2.5.

[3]German for "spooky action at a distance".

Three decades later, in 1964, John Bell formulated what is now known as Bell's theorem [22], showing that any theory based on local hidden variables should obey certain statistical inequalities, which quantum mechanical predictions could violate. Later experiments violated these constraints [23, 24], consistent with the predictions of quantum mechanics. These experiments confirmed that entanglement is a real phenomenon incompatible with any theory based on local hidden-variable theories. The Nobel Prize in Physics in 2022 was awarded "for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum information science" [25].

What started as a series of modifications to classical mechanics turned into a fundamental and mathematically rigorous theory. However, even today there are ongoing efforts to further understand quantum foundations and explain quantum mechanics from reasonable principles [26, 27].

2.2 Postulates of Quantum Mechanics

As experimental evidence accumulated and ideas were tested, some great minds of the century contributed to build a consistent formulation of quantum mechanics. In 1930, Paul Dirac published *The Principles of Quantum Mechanics* [28]. In this book, Dirac introduced the concept of states as elements of a complex vector space, developed the now standard bra-ket notation, and framed quantum mechanics in terms of a set of postulates designed from intuition and pragmatism. Two years later, John von Neumann provided the first rigorous mathematical foundation for quantum mechanics in his book *Mathematical Foundations of Quantum Mechanics* [29]. Building on Dirac's insights, von Neumann formalized the theory within the structure of Hilbert spaces and introduced a precise treatment of observables, measurement, and state evolution.

In the following, we introduce a customized modern formulation of the postulates

of quantum mechanics that evolved from the Dirac–von Neumann axioms. For each postulate, we provide brief explanations in order to clarify the reasoning behind its particular formulation. We also draw parallels to highlight features unique to quantum mechanics, in contrast to classical theory. For simplicity, we will restrict our attention to finite-dimensional Hilbert spaces. Also, notice that the following postulates hold for an *isolated* quantum system.

2.2.1 Description of the state of a system

Postulate *The state of an isolated quantum physical system is represented by a state vector $|\psi\rangle$, belonging to a Hilbert space $\mathcal{H}$ called "the state space". All physically realizable states are normalized,*

$$\| \, |\psi\rangle \, \| = \sqrt{\langle\psi|\psi\rangle} = 1, \tag{2.1}$$

In classical mechanics, the state of a particle is specified by its position and momentum alone, completely determining how the system evolves. However, in quantum mechanics, specifying both the position and momentum simultaneously is not possible, and experimental outcomes cannot be predicted in general. A new mathematical object is introduced to encapsulate all probabilistic aspects of measurement outcomes; vectors in a Hilbert space[4]. These probabilities are computed as absolute squares of inner products (see Section 2.2.3), making them invariant under a global phase transformation. Consequently, if $|\psi\rangle$ is a valid quantum state, then $e^{i\phi}\,|\psi\rangle$ represents the same physical state for any real ϕ. Since $\mathcal{H}$ is a linear vector space, the superposition principle holds: a linear combination of physical states is also a valid physical state; a feature that fundamentally differs from classical mechanics. Finally, the normalization condition is required to ensure that the probability of all possible outcomes adds up to 1.

[4]Named after the Prussian mathematician David Hilbert (1862-1943).

2.2.2 Observables as Hermitian operators

Postulate *Every observable quantity in quantum mechanics corresponds to a Hermitian operator O acting on the state space. The possible results of a measurement of O are given by its eigenvalues.*

In classical mechanics, physical observables such as energy, position, and momentum are described by continuous real-valued functions defined on the phase space, and measuring an observable corresponds to evaluating that function at the current state. By contrast, quantum mechanical measurements yield real outcomes that may be discrete (*e.g.* a photon's polarization or the energy levels of an atom) or continuous (*e.g.* position or relative phase), consistent with the eigenvalues[5] of Hermitian operators acting on finite- or infinite-dimensional Hilbert spaces, respectively. However, in this thesis, we restrict our attention to finite-dimensional Hilbert spaces, where observables are represented by Hermitian matrices.

An operator O is Hermitian if and only if $O = O^\dagger$, where the dagger sign $\dagger$ stands for complex transposition. All Hermitian operators are normal, $O^\dagger O = OO^\dagger$, and all normal operators have a spectral decomposition.

The spectral decomposition of a Hermitian operator O acting on a finite-dimensional Hilbert space $\mathcal{H}$ reads

$$O = \sum_i \lambda_i \left|\lambda_i\right\rangle \left\langle\lambda_i\right|,$$

where $\lambda_i \in \mathbb{R}$ are the eigenvalues and $\left|\lambda_i\right\rangle$ are the corresponding eigenvectors. The eigenvectors form a basis of $\mathcal{H}$. Thus, any state can be described as a superposition

$$\left|\psi\right\rangle = \sum_i \left\langle\lambda_i\right| \left.\psi\right\rangle \left|\lambda_i\right\rangle,$$

where the coefficients $\left\langle\lambda_i\right| \left.\psi\right\rangle$ are related to the probability of measuring $\left|\psi\right\rangle$ in the state $\left|\lambda_i\right\rangle$ as formalized in the following postulate 2.2.3.

[5] The word *eigen* stands for *self* or *characteristic* in German.

Finally, the operator is required to be Hermitian because under this condition the eigenvalues (*i.e.* the measurement results) are real.

2.2.3 Measurements and the Born rule for probability

Postulate *If the system is in a state* $|\psi\rangle$*, measuring an observable* $O = \sum_i \lambda_i |\lambda_i\rangle\langle\lambda_i|$ *will project the state into one of the eigenvectors* $|\lambda_i\rangle$*, with measurement result* λ_i *and a probability given by the Born rule*[6]*,*

$$P(\lambda_i) = |\langle\lambda_i|\psi\rangle|^2. \tag{2.2}$$

In classical mechanics, a measurement simply reads off the value of the observable function in the current state and does not inherently change the state of the system. In quantum mechanics, however, a measurement *fundamentally* perturbs the state of the system, which becomes one of the eigenstates $|\lambda_i\rangle$ of the observable after the measurement. Then, an immediate repetition of the measurement will keep the state fixed at $|\lambda_i\rangle$ with the repeating result λ_i. Experiments show that, unless the state is one of the eigenstates of the observable (*i. e.* the property measured is well defined), quantum measurements are inherently probabilistic and we cannot predict their exact result. This is reflected by the Born rule (2.2); even if we have all the information about the system (we know its quantum state), we can only predict the probability of obtaining each of a set of outcomes.

Note that practical physical measurements may include the effects of noise, detector inefficiencies, and indirect measurements. One accounts for these effects by generalizing the notion of observable to a collection of measurement operators $\{M_i\}$ satisfying the completeness relation $\sum_i M_i^\dagger M_i = I$. Then, the probability of measuring the outcome labeled as i is

$$P(i) = \|M_i|\psi\rangle\|^2,$$

[6]Named after the Prussian physicist Max Born (1882-1970).

and, after the measurement, the system is projected into the state $M_i |\psi\rangle / \| M_i |\psi\rangle \|$.

These are known as positive operator-valued measures (POVM), where each M_i acts as a generalization of the projector $|\lambda_i\rangle\langle\lambda_i|$ with the distinction that, unlike the projector, M_i is not generally idempotent, $M^2 \neq M$. A direct consequence is that repeated measurements will not necessarily yield the same repeated result.

We can fully recover the projective measurements stated in the postulate as a special case of POVMs by requiring the measurement operators to be projectors, $M_i = |\lambda_i\rangle \langle\lambda_i|$.

Finally, notice that the sum of all probabilities is guaranteed to be 1 by the first postulate, 2.2.1, since

$$\sum_i P(i) = \sum_i \| M_i |\psi\rangle \|^2 = \langle\psi| \left(\sum_i M_i^\dagger M_i \right) |\psi\rangle = \langle\psi|\psi\rangle = 1.$$

2.2.4 Time evolution and the Schrödinger equation

Postulate *The time evolution of a quantum state $|\psi\rangle$ is governed by the Schrödinger equation,*

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle , \tag{2.3}$$

where H is the quantum Hamiltonian of the system.

In the absence of measurements, the time evolution is smooth and deterministic. This is similar to classical mechanics, where the evolution of a system is described by Hamilton's equations. In this sense, the Schrödinger equation is a quantum analog of the classical equations of motion, with the Hamiltonian[7] operator playing the same central role.

Many parallels can be drawn between the Schrödinger equation and the classical motion equations and, in fact, Equation (2.3) can be obtained as a generalization of the classical Hamilton-Jacobi equations [30].

[7] Named after the Irish astronomer, mathematician and physicist Sir William Rowan Hamilton.

The time evolution of a quantum state is customarily understood through the action of a time evolution operator $U(t, t_0)$,

$$|\psi(t)\rangle = U(t, t_0)\,|\psi(t_0)\rangle\,,$$

subject to the initial value problem

$$\frac{d}{dt}U = \frac{-i}{\hbar}HU, \tag{2.4}$$

$$U(t_0, t_0) = I. \tag{2.5}$$

If the Hamiltonian is time-independent, the solution to the previous problem is analytical and given by

$$U(t, t_0) = \exp\left[-\frac{i}{\hbar}(t - t_0)H\right].$$

However, if the Hamiltonian depends on time, the expression for U is more complicated. A formal solution is the Dyson[8] series,

$$U(t, t_0) = I + \sum_{n=1}^{\infty}\left(-\frac{i}{\hbar}\right)^n \int_{t_0}^{t} dt_1 \int_{t_0}^{t_1} dt_2 \cdots \int_{t_0}^{t_{n-1}} dt_n H(t_1)H(t_2)\cdots H(t_n),$$

under the assumption that the infinite series converges.

In practice, the sum can be truncated to yield approximate expressions for $U(t, t_0)$, although other approaches are preferred for typical numerical computations. For instance, a first-order Taylor series expansion in Δt shows that

$$U(t_0 + \Delta t, t_0) = \exp(-iH(t_0)\Delta t/\hbar) + O((\Delta t)^2).$$

This leads to a piecewise constant approximation of the time evolution operator with N steps,

$$U_N(t, t_0) = \prod_{k=N-1}^{0} \exp\left[\frac{-i}{\hbar}\frac{(t - t_0)}{N}H\left(t_0 + k\frac{(t - t_0)}{N}\right)\right],$$

[8]Named after the British-American physicist Freeman Dyson (1923-2020).

such that $U(t, t_0) = U_N(t, t_0) + O((t - t_0)^2/N)$. This yields

$$U(t, t_0) = \lim_{N \to \infty} U_N(t, t_0),$$

if the infinite operator composition converges.

Notice that each exponential in U_N is a unitary operator since it is the composition of complex exponentials with Hermitian argument. Therefore, U is unitary itself $U^\dagger U = I$. This means that the norm of the initial state is preserved along the time evolution,

$$\langle \psi(t) | \, \psi(t) \rangle = \langle \psi(0) | \, U^\dagger U \, | \psi(0) \rangle = \langle \psi(0) | \, \psi(0) \rangle = 1.$$

2.2.5 Composite systems

Postulate *For a composite system made of subsystems with Hilbert spaces $\mathcal{H}_1, \mathcal{H}_2, \ldots$, the Hilbert space of the whole system is the tensor product*

$$\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \ldots . \tag{2.6}$$

In classical mechanics, the state space of a composite system is given by the Cartesian[9] product of the state spaces of its parts. Two independent systems with u and v degrees of freedom combine into a system with $u + v$ degrees of freedom. In quantum mechanics, states are represented by vectors in a Hilbert space, and the superposition principle must hold: any linear combination of products of states from the subsystems must be a valid state of the composite system. Therefore, the product of individual Hilbert spaces must yield a Hilbert space whose dimension is the product of the dimensions of the subsystems. The tensor product satisfies this requirement.

Moreover, the use of the tensor product to compose individual Hilbert spaces is not an arbitrary choice. It is the only product ensuring that independent systems

[9] Named after the French philosopher, scientist, and mathematician René Descartes (1596-1650).

have independent probability distributions [31]. From postulate 2.2.3, if a system is in state $|\psi\rangle$, then the probability of obtaining the outcome a when measuring an observable with associated eigenstate $|a\rangle$ is

$$P(a) = |\langle a|\psi\rangle|^2.$$

Assume that a system is composed by two independent subsystems. Then, the composite system is in a product state, which we write as $|\psi_1\rangle \otimes |\psi_2\rangle$ for the first system in state $|\psi_1\rangle$ and the second system in state $|\psi_2\rangle$. The probability of obtaining outcomes a_1 and a_2 from measurements on each respective subsystem is,

$$P(a_1, a_2) = |(\langle a_1| \otimes \langle a_2|)(|\psi_1\rangle \otimes |\psi_2\rangle)|^2 = |\langle a_1|\psi_1\rangle|^2 |\langle a_2|\psi_2\rangle|^2 = P_1(a_1)P_2(a_2),$$

where $P_1(a_1)$ and $P_2(a_2)$ are the probabilities for outcomes a_1 and a_2 measured independently on each subsystem. The tensor product is the only product ensuring that probabilities factorize for all product states and measurements, preserving the statistical independence of independent subsystems.

Consider the following example: two spin-1/2 particles, each with two possible states that we will denote "up" ($|\uparrow\rangle$) and "down" ($|\downarrow\rangle$). In the classical case, the composite system could be in one of the following states: The two particles in the state up, the first up and the second down, the first down and the second up, or both particles in the state down. Respectively,

$$|\uparrow\rangle \otimes |\uparrow\rangle, |\uparrow\rangle \otimes |\downarrow\rangle, |\downarrow\rangle \otimes |\uparrow\rangle, |\downarrow\rangle \otimes |\downarrow\rangle.$$

In quantum mechanics, however, the superposition principle states that any linear combination of valid states is also a valid state, and this principle must hold for composite systems. The state space of the combined system must contain any linear combination

$$|\psi\rangle = \alpha |\uparrow\rangle \otimes |\uparrow\rangle + \beta |\uparrow\rangle \otimes |\downarrow\rangle + \gamma |\downarrow\rangle \otimes |\uparrow\rangle + \delta |\downarrow\rangle \otimes |\downarrow\rangle,$$

with normalization $|\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$. The tensor product $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$ is precisely the space that contains all such states.

One can notice that many states in this space cannot be written as simple products like $|\psi_1\rangle \otimes |\psi_2\rangle$. Such states are said to be entangled. An example is the Bell state

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}\Big(|\uparrow\rangle \otimes |\uparrow\rangle + |\downarrow\rangle \otimes |\downarrow\rangle \Big).$$

In this state, measuring any of the two particles would yield an outcome up or down with the same probability. However, after measuring the spin of one particle, the spin of the other is immediately known: both particles have the same spin. These quantum correlations have been confirmed experimentally and cannot be explained by local theories based on hidden variables.

A final important implication of the tensor product structure of composite spaces is that the dimensionality of the state space grows exponentially with the number of subsystems. For example, a system composed of N spin-1/2 particles will have a Hilbert space of dimension 2^N. This greatly complicates the classical simulation of composite quantum mechanical systems and, at the same time, gives extraordinary theoretical power to computers with quantum mechanical parts, as discussed in Chapter 3.

2.3 Mixed states

Until this point, we have discussed pure states, which are described by state vectors in a Hilbert space. However, pure states assume perfect knowledge of the quantum system. In real-world applications, our knowledge is often incomplete due to statistical uncertainty or environmental interactions, requiring a more general description of states known as mixed states.

To describe both pure and mixed states, we introduce the density matrix formalism. Instead of working with vectors pertaining to the Hilbert space $\mathcal{H}$, we promote

the pure quantum states to projector operators acting on $\mathcal{H}$,

$$|\psi\rangle \longrightarrow \rho = |\psi\rangle \langle\psi| ,$$

which have the advantage of naturally ignoring global phases,

$$e^{i\phi} |\psi\rangle \longrightarrow \left(e^{i\phi} |\psi\rangle\right) \left(e^{-i\phi} \langle\psi|\right) = |\psi\rangle \langle\psi| .$$

For a statistical mixture of different quantum states $\{|\psi_i\rangle\}$ with associated probabilities p_i, the system is in a mixed state,

$$\rho = \sum_i p_i |\psi_i\rangle \langle\psi_i| .$$

Unlike a pure state, a mixed state cannot be expressed as a single state vector. Instead, it represents an ensemble of quantum states with classical uncertainty.

In the following, we restate the postulates of quantum mechanics for mixed states.

2.3.1 Description of the state of a system

Postulate *The state of a quantum system is represented by a density operator ρ acting on a Hilbert space $\mathcal{H}$. This operator is positive semidefinite and has trace* $\mathrm{Tr}(\rho) = 1$.

The trace of an operator A belonging to a Hilbert space $\mathcal{H}$ is defined such that, for any given orthonormal basis $\{|\phi_i\rangle\}$ of $\mathcal{H}$,

$$\mathrm{Tr}(A) = \sum_i \langle\phi_i| A |\phi_i\rangle .$$

One can verify that the value of the trace of A is independent on the choice of basis. An important particular case is the choice of basis given by the eigenvectors of A. Then, it follows that the trace of A is equal to the sum of its eigenvalues.

Given the definition of the trace operation, all physically valid density matrices ρ must satisfy the following properties,

- Hermiticity, $\rho^\dagger = \rho$. This ensures that the eigenvalues (probabilities) are real.
- Positive semidefiniteness, $\langle\phi|\,\rho\,|\phi\rangle \geq 0, \forall\,|\phi\rangle$. This condition includes the previous one, and further constrains the eigenvalues of ρ to be nonnegative.
- Unit trace, $\mathrm{Tr}(\rho) = 1$. Ensure that the total probability adds up to 1.
- Purity condition, $1/d \leq \mathrm{Tr}(\rho^2) \leq 1$ for a state in a d-dimensional Hilbert space. The purity of a density matrix is defined as $\mathrm{Tr}(\rho^2)$. A state is pure if and only if $\mathrm{Tr}(\rho^2) = 1$.

2.3.2 Observables as Hermitian operators (unchanged)

Postulate *Every observable quantity in quantum mechanics corresponds to a Hermitian operator O acting on the state space. The possible results of a measurement of O are given by its eigenvalues.*

2.3.3 Measurement and the Born rule for probability

Postulate *A measurement on a system in a mixed state ρ, described by an observable $O = \sum_i |\lambda_i\rangle\langle\lambda_i|$ yields the outcome λ_i with probability*

$$P(\lambda_i) = Tr(|\lambda_i\rangle\langle\lambda_i|\,\rho) = \langle\lambda_i|\,\rho\,|\lambda_i\rangle\,. \tag{2.7}$$

The state after the measurement becomes

$$\rho' = \frac{|\lambda_i\rangle\langle\lambda_i|\,\rho\,|\lambda_i\rangle\langle\lambda_i|}{\mathrm{Tr}(|\lambda_i\rangle\langle\lambda_i|\,\rho)} = |\lambda_i\rangle\langle\lambda_i|$$

This formulation also generalizes to positive operator-valued measures (POVMs), where the measurement is described by a set of operators $\{M_i\}$ satisfying $\sum_i M_i^\dagger M_i = I$. The probability of obtaining the result labeled i is $P(i) = Tr(M_i^\dagger M_i \rho)$, and the post-measurement state is $\rho' = \frac{M_i \rho M_i^\dagger}{Tr(M_i^\dagger M_i \rho)}$.

2.3.4 Time evolution and the von Neumann equation

Postulate *For an isolated quantum system, the time evolution of the density matrix is governed by the von Neumann equation,*

$$i\hbar\frac{d\rho}{dt} = [H, \rho],$$

where H is the Hamiltonian of the system, and $[H, \rho] = H\rho - \rho H$ is the commutator.

The von Neumann equation[10] is the generalization of the Schrödinger equation (2.3) for density matrices, and has a formal solution

$$\rho(t) = U(t)\rho(t=0)U^\dagger(t), \tag{2.8}$$

where $U(t)$ is the unitary time evolution operator defined by Equations (2.4) and (2.5). This implies that, for a closed system, the purity of the state is preserved during time evolution.

Considering that a density matrix must remain positive semidefinite and trace-normalized, the most general physically admissible evolution is not necessarily unitary. For example, if the system is not isolated but is allowed to interact with an environment, the evolution is no longer unitary. Instead, assuming Markovian dynamics[11], it follows a Lindblad[12] master equation [32, 33],

$$\frac{d\rho}{dt} = -\frac{i}{\hbar}[H, \rho] + \sum_k \left(L_k \rho L_k^\dagger - \frac{1}{2}\left[L_k^\dagger L_k \rho + \rho L_k^\dagger L_k \right] \right),$$

where L_k are the Lindblad operators that describe dissipative interactions.

[10] Named after the Austro-Hungarian mathematician John von Neumann (1903-1957).

[11] Named after the Russian mathematician Andrey Markov (1856-1922). A Markovian process is one such that the probability of transitioning to a new state depends only on the current state and not on the past history of the system.

[12] Named after the Swiss physicist Göran Lindblad (1940-2022).

2.3.5 Composite systems

Postulate *For a system composed of multiple subsystems with multiple Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$, the total system lives in the tensor product space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. The density matrix of the composite system is $\rho_{AB} \in \mathcal{H}$, and the state of each subsystem can be obtained via the partial trace,*

$$\rho_A = \mathrm{Tr}_B(\rho_{AB}),$$
$$\rho_B = \mathrm{Tr}_A(\rho_{AB}).$$

The partial trace over one of the subsystems, say $\mathcal{H}_B$, is the unique linear map satisfying, for all operators $A \in \mathcal{H}_A$, $B \in \mathcal{H}_B$,

$$\mathrm{Tr}_B(A \otimes B) = A\mathrm{Tr}(B).$$

Operationally, if $\{|\phi_i\rangle_B\}$ is an orthonormal basis of $\mathcal{H}_B$, then for any state ρ_{AB} the partial trace over subsystem B is given by

$$\mathrm{Tr}_B(\rho_{AB}) = \sum_i \left(I_A \otimes \langle\phi_i|_B\right) \rho_{AB} \left(I_A \otimes |\phi_i\rangle_B\right).$$

The partial trace of a pure composite state can be mixed, which occurs when the subsystems are entangled. Moreover, any mixed state of a system A can be obtained as the partial trace of a pure state in a larger Hilbert space.

To characterize correlations between subsystems, we introduce the notion of separability. A bipartite state ρ_{AB} is said to be separable if it can be written as a convex combination of product states [34],

$$\rho_{AB} = \sum_{k=1}^{N} p_k \rho_A^{[k]} \otimes \rho_B^{[k]}, \tag{2.9}$$

where $p_k \geq 0$ are probabilities such that $\sum_k^N p_k = 1$, and $\rho_A^{[k]}$ and $\rho_B^{[k]}$ are density operators acting on $\mathcal{H}_A$ and $\mathcal{H}_B$, respectively.

More specifically, if $N = 1$ the state is a product state, with uncorrelated subsystems. For $N > 1$, the state is separable, with classical correlations. If decomposition (2.9) does not exist, the state is said to be entangled.

Chapter 3

Quantum Computing

> *"Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy"*
>
> Richard Feynman

Quantum computing concerns the processing of information using systems that obey the laws of quantum mechanics. These systems can exhibit phenomena such as superposition and entanglement, enabling new modes of computation with no classical analogs, which could be leveraged to perform calculations more efficiently than classical computers. Although quantum mechanics originated in the early 1900s, the idea of applying its principles to computing only took strength in the 1980s.

One of the earliest motivations for quantum computing came from Richard Feynman, who observed that simulating quantum systems on classical computers becomes inefficient as system size grows, due to the exponential scaling of the Hilbert space; a consequence from postulate 2.2.5. Feynman suggested that quantum systems might be better simulated using quantum mechanical devices, thus laying the foundation for quantum computation [35].

All physical systems are constrained by the laws of physics, which dictate the fundamental limits of computation. An important result in this context is Landauer's principle [36], which states that erasing a single bit of information in a system at temperature T incurs a minimum energy cost of $k_B T \ln 2$, where k_B is the Boltzmann constant.

In classical devices, information is typically represented using electrical signals that switch transistors on and off, encoding binary states. These signals are processed through logic operations implemented in electronic circuits. While many classical logic gates are irreversible and therefore subject to Landauer's limit, this is not an inherent requirement of classical computation. Reversible computation, where no information is discarded and thus avoids this thermodynamic cost, has been theoretically explored since the 1970s. For example, Charles Bennett showed that any computation could, in principle, be performed reversibly [37]. However, in practice, implementing fully reversible computation in classical hardware is very challenging [38]. Quantum computing, in contrast, can be naturally posed reversibly since quantum mechanics dictates that the evolution of closed systems must be unitary (see Section 2.2.4).

The foundations of quantum computation were developed in the 1980s with contributions from people like Paul Benioff, who showed in 1980 that a Turing machine could be reproduced by following evolution dictated by the Schrödinger equation [39], and David Deutsch, who introduced in 1985 the concept of the quantum Turing machine [40], formalizing the notion of quantum algorithms. Later, in 1989, Deutsch developed the quantum circuit model [41], which remains the dominant framework for quantum computation today. This model represents quantum computations as sequences of unitary transformations applied to quantum bits (qubits), enabling operations that have no classical counterpart.

Building on these theoretical foundations, the development of quantum algorithms provided a major motivation to further explore quantum computing. In

1992, David Deutsch and Richard Jozsa proposed a class of problems that quantum computation could solve in exponentially less time than any classical deterministic computation [42]. However, the solution to their problem had no practical use. Similarly, in 1993, Daniel Simon conceived a problem that a quantum computer would solve exponentially faster than a classical computer [43]. Inspired by the quantum algorithm that Simon proposed to solve that problem, Peter Shor demonstrated in 1994 that a quantum computer could solve integer factoring and discrete logarithm problems with an algorithm almost exponentially faster than the best-known classical methods [44]. This posed a significant threat to cryptographic systems, such as RSA encryption, which relies on the difficulty of prime factorization for security [45]. Later, in 1996, Lov Grover developed a quantum search algorithm that could speed up database search quadratically, compared to classical approaches [46]. These discoveries demonstrated the potential of quantum computers to outperform classical systems in specific tasks, a concept now known as "quantum advantage" or "quantum supremacy"[1].

Realizing these theoretical ideas in practice requires a series of conditions. In 1996, David DiVincenzo provided a list of minimal requirements for creating a quantum computer, which he later summarized as follows [47, 48]: (1) a scalable physical system with well-characterized qubits, (2) the ability to initialize the state of the qubits to a simple fiducial state, (3) long relevant decoherence times, much longer than the gate operation time, (4) a universal set of quantum gates, and (5) a qubit-specific measurement capability.

Researchers have explored various physical platforms to implement and manipulate qubits that meet DiVincenzo's criteria. These platforms include trapped ions [49], photonic systems [50], nuclear magnetic resonance (NMR) machines [51] and superconducting circuits [52]. Many experimental milestones demonstrate the

[1] Although the term "quantum supremacy" has been largely abandoned in the community due to its unintended connotations.

fast-paced development of quantum hardware. To name a few: in 1995, researchers from NIST Boulder implemented the first entangling 2-qubit logic gate: a CNOT gate on a single trapped ion [53], following the scheme by Cirac and Zoller [54]. Soon, the first execution of a quantum algorithm followed: in 1998, Oxford researchers used a 2-qubit NMR processor to solve Deutsch's problem [51]. Three years later, in 2001, IBM implemented Shor's factoring algorithm on a 7-qubit NMR device and successfully factored $15 = 3 \times 5$ [55]. In 2003, the first implementation of a CNOT quantum gate entangling two distinct trapped ions was reported at the University of Innsbruck [56].

In the following decade, there was already commercial interest and public access to quantum computers: in 2010, D-Wave Systems released the first commercially available quantum computer, based on quantum annealing, and the University of Bristol provided free access to a photonic quantum processor over the internet in 2013 [57]. Later in 2016, IBM launched the IBM Q Experience: a cloud accessible, gate-based superconducting quantum computer, offering public access to design circuits and run them on real superconducting hardware [58].

In 2018, John Preskill introduced the concept of Noisy Intermediate-Scale Quantum (NISQ) era [59], characterizing devices with 50-100 qubits, limited coherence times, and no error correction. He emphasized that these devices could still offer practical value until the development of fault-tolerant quantum computers. Since then, quantum hardware has continued to improve. For instance, in 2024, Google presented a 105-qubit superconducting quantum chip that achieved a significant milestone in quantum error correction: the system implemented quantum operations with error rates low enough that grouping physical qubits into logical qubits would allow the reduction of overall errors [60]. Nevertheless, the available number of qubits is still far from enough to implement quantum error correction schemes and run practical quantum algorithms reliably.

Although large-scale, fault-tolerant quantum computers still remain theoretical,

their successful realization could revolutionize computational science. They could solve certain problems exponentially faster than classical computers, or with exponentially smaller storage requirements. However, scaling up quantum hardware while bounding error rates remains a difficult challenge, which motivates the design of algorithms and numerical schemes that can extract value from the quantum devices available today.

To provide a foundation for these techniques, this chapter introduces the fundamental principles of quantum computing. We begin with a basic review of classical computing and transition to quantum systems by analogy. We present the concept of qubits, quantum logic gates and quantum circuits, and introduce the paradigms of fault-tolerant quantum computing, noisy intermediate-scale quantum devices, and variational quantum algorithms (VQAs).

3.1 Gate-based classical computing

At its core, computation is about processing and transmitting information. A fundamental result proved by Claude Shannon is that logical operations can be systematically represented using binary switches [61]. He established that information processing could be physically implemented with electrical circuits, providing the basis for digital computers.

The fundamental unit of classical information is the bit (binary digit), which represents a choice between two alternatives. A bit takes values from the set $\{0, 1\}$ to encode information, and collections of bits can encode variables containing larger amounts of information. For instance, computers store integers in binary representation, using a fixed amount of bits. The modern standard for integer representation in most architectures is 64-bit signed integers. In this format, the first bit is the sign bit (0 for positive, 1 for negative), leaving 63 bits for the number itself in binary representation. This allows for the representation of values from -2^{63} to $2^{63} - 1$.

Letters are encoded as integers in the ASCII standard, and real numbers are stored in the more convoluted IEEE 754 Floating Point standard, which allows to operate with a number of significant digits and exponent, but the essence is the same: we encode the information as strings of 0s and 1s.

3.1.1 Logic operations and classical circuits

To manipulate bits, we use Boolean[2] algebra, where logic operations act on binary values. Typical Boolean operations are

- NOT (negation)

  A single bit operation, that flips a 0 into 1 and vice-versa,

$$\mathrm{NOT}(0) = 1, \quad \mathrm{NOT}(1) = 0,$$

- AND

  A two-bit operation, which only results in a 1 if both inputs are 1,

$$\mathrm{AND}(A, B) = A \cdot B,$$

- OR

  A two-bit operation, resulting in 1 if at least one input is 1,

$$\mathrm{OR}(A, B) = A + B - A \cdot B,$$

- XOR (Exclusive OR)

  A two-bit operation, resulting in 1 if the inputs differ, and 0 otherwise

$$\begin{aligned} \mathrm{XOR}(A, B) &= A + B \mod 2 \\ &= \mathrm{OR}(\mathrm{AND}(A, \mathrm{NOT}(B)), \mathrm{AND}(\mathrm{NOT}(A), B)) \end{aligned}$$

[2]Named after the British mathematician George Boole (1815-1864), considered as an early founder of computer science.

- SWAP

  A two bit operation which also returns two bits, with their values swapped,

$$\text{SWAP}(A, B) = (B, A).$$

Notice how all these two-bit gates, with the exception of the SWAP, take 2 bits as inputs and return only 1 bit. This loss of a bit incurs in the theoretical inefficiency lower bound set by Landauer [36].

In practice, most classical computational devices encode bits as cables in a circuit, and the value of the bit is whether the cable has current (1) or not (0). Logic gates are implemented with a number of interconnected transistors, and more complex operations such as the addition of two integers are performed as a fixed composition of logic operations over the bits that make up the inputs. This composition is called a logic circuit.

Thus, any physical system that can exist in two distinguishable states can represent a bit. A collection of such systems (a register) can encode more complex information, and logic circuits process the information in these registers by applying sequences of logic operations. That is the basis for classical information processing, allowing us to perform computations in our cellphones, supercomputers, an abacus, to name a few classical computing devices.

3.2 Quantum bits

When working with quantum information, the fundamental unit is the quantum bit (qubit). Consider, for instance, a two-state quantum system, such as an electron in a two-level atom. When measured, the electron can be found in either the first energy level (state $|0\rangle$) or the second energy level (state $|1\rangle$). However, the superposition principle states that any linear combination of these states is also a valid quantum

state. Therefore, the qubit can be in an arbitrary state:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{3.1}$$

where the normalization condition for quantum states requires $|\alpha|^2 + |\beta|^2 = 1$.

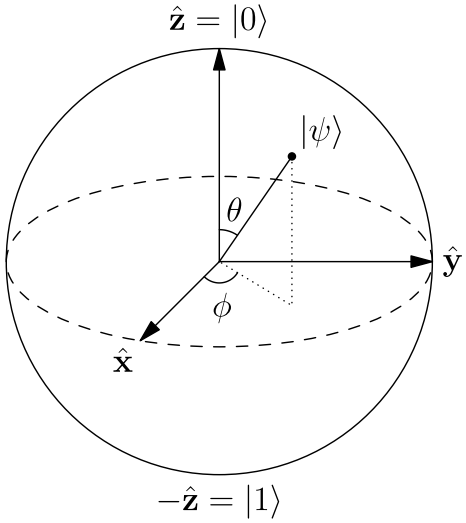


Figure 3.1: Representation of a pure state $|\psi\rangle$ on the Bloch sphere [62].

A useful way to visualize the state of a single qubit is through the Bloch sphere[3], as depicted on Figure 3.1. In this model, each possible state $|\psi\rangle$ corresponds to a point on the surface of a sphere of unit radius. The standard basis states $|0\rangle$ and $|1\rangle$ correspond to the north and south pole of the sphere, respectively, and any other point on the sphere is in a superposition of these basis states, with the equator representing states with equal probability of being measured in $|0\rangle$ or $|1\rangle$. The angles θ and ϕ in the image allow to rewrite $|\psi\rangle$ from (3.1) as

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\phi}\sin(\theta/2)\,|1\rangle\,,$$

where $\theta = 2\cos^{-1}(|\alpha|) \in [0, \pi]$, $\phi = \arg(\beta/\alpha) \in [0, 2\pi[$, and the global phase $\alpha/|\alpha|$ was ignored.

[3]Named after the Swiss physicist Felix Bloch (1905-1983).

We represent the Hilbert space of a single qubit as $\mathcal{H}_2 = \mathbb{C}^2$, where quantum states are 2-dimensional complex vectors, and we associate $|0\rangle$ and $|1\rangle$ to the canonical basis of $\mathbb{C}^2$,

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{3.2}$$

To describe systems composed of multiple qubits, we rely on principle 2.2.5 and construct larger Hilbert spaces as tensor products of the spaces of individual qubits. In particular, to encode a vector from a Hilbert space of dimension m, we require at least k qubits, spanning a space $\mathcal{H}_{2^k} \supseteq \mathcal{H}_m$, with

$$k = \lceil \log_2(m) \rceil.$$

For example, the state $|0\rangle \otimes |1\rangle$ represents a two-qubit system where the first qubit is in $|0\rangle$ and the second qubit in $|1\rangle$, written explicitly as

$$|0\rangle \otimes |1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

In the literature, there are multiple equivalent ways to denote a multi-qubit state. For instance, consider a system with 5 qubits, each in the states $|0\rangle$, $|1\rangle$, $|1\rangle$, $|0\rangle$, and $|1\rangle$, respectively. This state can be equivalently expressed as

$$\begin{aligned} |0\rangle \otimes |1\rangle \otimes |1\rangle \otimes |0\rangle \otimes |1\rangle &= |0\rangle_1 |1\rangle_2 |1\rangle_3 |0\rangle_4 |1\rangle_5 \\ &= |0\rangle |1\rangle |1\rangle |0\rangle |1\rangle \\ &= |01101\rangle . \end{aligned}$$

Another notation typically found in the literature would represent the state as $|13\rangle$, following from the fact that 01101 is the binary representation of the decimal number 13. However, this notation is informal and should only be used when the number of qubits is clear from the context.

3.3 Quantum gates

There are multiple models for quantum computation, including adiabatic quantum computing, measurement-based quantum computing and topological quantum computing, each with a different way to describe transformations of quantum information. In this thesis, we will focus on the gate-based model, which is the most common model and is analogous to classical digital circuits.

In the gate-based model, a quantum computation is performed by applying sequences of transformations to qubits, where these transformations are realized as quantum gates. Quantum gates are typically unitary operations, which preserve the norm of the quantum state and thus are reversible operations. The basic building blocks in this model are the quantum logic gates, which are elementary unitary operations. By composing these gates, one can approximate arbitrary unitaries, similar to how classical logic gates (such as AND, OR, and NOT) are combined to perform complex classical computations.

A fundamental result supporting the gate-based model for quantum computing is the Solovay-Kitaev theorem [63, 64], stated as follows.

Solovay-Kitaev theorem *Let $\mathcal{G}$ be a finite set of elements in $\mathrm{SU}(d)$ containing its own inverses, and such that the group generated by $\mathcal{G}$ is dense in $\mathrm{SU}(d)$. There is a constant c such that, for any unitary $U \in \mathrm{SU}(d)$ and any desired accuracy $\epsilon > 0$, there exists a finite sequence $S = g_1 g_2 \dots g_l$ of elements $g_i \in \mathcal{G}$, with $l = O(\log^c(1/\epsilon))$, satisfying*

$$\|U - S\| < \epsilon,$$

where $\|\cdot\|$ denotes the operator norm.

Here, $\mathrm{SU}(d)$ denotes the *special unitary group of degree d*, consisting of $d \times d$ complex unitary matrices with determinant 1. For $d = 2$, the Solovay-Kitaev theorem provides that any single-qubit unitary operation can be efficiently approximated, to

arbitrary precision, by a finite sequence of gates from a universal gate set[4]. Furthermore, the Solovay-Kitaev algorithm outputs the approximating sequence of gates in time $O(\log^k(1/\epsilon))$, where the constants c and k have values typically ranging from 1 to 4 [64].

To approximate general multi-qubit unitaries, one requires entangling gates. More specifically, any unitary operation acting on an arbitrary number of qubits can be approximated using a combination of single-qubit gates with a two-qubit entangling gate, such as the CNOT gate [65, 66]. A gate set that allows to perform such approximation is called universal for quantum computing. An example of a universal set in this context is given by the gates $\{\mathrm{H}, \mathrm{T}, \mathrm{CNOT}\}$, defined in Sections 3.3.1 and 3.3.2.

In practice, quantum logic gates are implemented by engineering quantum evolution under specific Hamiltonians and time intervals, as discussed in Section 2.2.4. Naturally, some given quantum hardware will support a limited set of physical operations, which translates into a set of native gates. To execute quantum algorithms, quantum compilers translate arbitrary gates into sequences of native gates that can be physically implemented on the target device.

In the following, we will introduce some relevant single-qubit and two-qubit logic gates.

3.3.1 Single qubit gates

3.3.1.1 Pauli gates

The most basic single-qubit gates are the Pauli gates[5], which represent half-full rotations around each of the three Cartesian axes of the Bloch sphere 3.1. These gates serve as conceptual building blocks for more general quantum operations.

[4] In this context, universal refers to the ability to approximate any single-qubit gate. Do not confuse with universal for quantum computing, which we introduce in the next paragraph.

[5] Named after the Austro-Hungarian physicist Wolfgang Pauli (1900-1958).

The Pauli gates are defined as

$$\mathrm{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathrm{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad \mathrm{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

- The X gate is the quantum analog of the NOT gate. It flips the elements of the computational basis,

$$\mathrm{X}\,|0\rangle = |1\rangle\,,$$
$$\mathrm{X}\,|1\rangle = |0\rangle\,.$$

- The Y gate is similar to the X gate, but also introduces a phase factor,

$$\mathrm{Y}\,|0\rangle = i\,|1\rangle\,,$$
$$\mathrm{Y}\,|1\rangle = -i\,|0\rangle\,.$$

- The Z gate leaves $|0\rangle$ unchanged, while applying a π phase to $|1\rangle$,

$$\mathrm{Z}\,|0\rangle = |0\rangle\,,$$
$$\mathrm{Z}\,|1\rangle = -\,|1\rangle\,.$$

Note that $\mathrm{X}^2 = \mathrm{Y}^2 = \mathrm{Z}^2 = I$, since each represents a half-full rotation.

3.3.1.2 Rotation gates

Rotation gates are a more general class of single-qubit transformations, which are parameterized by an angle, and allow for arbitrary rotations about the Bloch sphere axes. The rotation gates around axes x, y and z, respectively, are

$$\mathrm{R}_x(\theta) = \exp\left(-i\frac{\theta}{2}\mathrm{X}\right) = \begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix},$$
$$\mathrm{R}_y(\theta) = \exp\left(-i\frac{\theta}{2}\mathrm{Y}\right) = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix},$$
$$\mathrm{R}_z(\theta) = \exp\left(-i\frac{\theta}{2}\mathrm{Z}\right) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}.$$

Note that, since Pauli gates are 180-degree rotations, they are special cases of the rotation gates, up to a global phase,

$$X = -i\mathrm{R}_x(\pi), \quad Y = -i\mathrm{R}_y(\pi), \quad Z = -i\mathrm{R}_z(\pi).$$

From Euler's rotation theorem[6], it follows that any single-qubit unitary operation, which corresponds to a rotation in the Bloch sphere, can be expressed as a composition of at most three rotations around two different fixed axes. This also implies that the three rotation gates are not independent: A rotation around any of the three principal axes can be achieved from rotations around the other two.

3.3.1.3 Hadamard gate

The Hadamard gate[7] is particularly important to create uniform superpositions of computational basis states. Its matrix representation is given by

$$\mathrm{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

and it maps a computational basis state $|0\rangle$ or $|1\rangle$ to an equally probable superposition of both,

$$\mathrm{H}\,|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right),$$
$$\mathrm{H}\,|1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right).$$

3.3.1.4 Phase shift and special cases

The phase shift gate is a general operation that leaves the state $|0\rangle$ unchanged while adding a phase factor to the state $|1\rangle$. It is equivalent to a R_z rotation, up to a global phase,

$$\mathrm{P}(\phi) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix} = e^{i\phi/2}\mathrm{R}_z(\phi).$$

[6] Named after the Swiss mathematician and physicist Leonhard Euler (1707-1783).

[7] Named after the French mathematician Jacques Hadamard (1865-1963).

Its action on the basis states is given by

$$\mathrm{P}(\phi)\,|0\rangle = |0\rangle\,,$$
$$\mathrm{P}(\phi)\,|1\rangle = e^{i\phi}\,|1\rangle\,,$$

and relevant special cases of this gate are:

- The S gate, $\mathrm{S} = \sqrt{\mathrm{Z}} = \mathrm{P}(\pi/2)$.
- The T gate, $\mathrm{T} = \sqrt{\mathrm{S}} = \mathrm{P}(\pi/4)$.

3.3.2 Two-qubit gates

Two-qubit gates are essential to represent interactions between qubits. Moreover, when combined with a set of single-qubit gates that is universal for SU(2), entangling two-qubit gates allow the formation of a universal gate set [65, 66]. This makes it possible to approximate any multi-qubit unitary operation, asymptotically covering the entire Hilbert space $\mathcal{H}_{2^n}$ for n qubits.

3.3.2.1 Controlled gates

Controlled gates are a class of operations that require a control and a target qubit. They apply a conditional transformation whenever the control qubit is on the state $|1\rangle$, and leave the target unchanged when the control qubit is on the state $|0\rangle$.

Any single-qubit operation can be controlled by another qubit, but in this section we will only show the controlled-X operation, as an example and due to its importance. Other controlled operations work in an analog way.

The CNOT, CX, or controlled-X gate flips a target qubit if and only if the control

qubit is in the state $|1\rangle$. Its matrix representation is

$$\mathrm{CX}_{12} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes \mathrm{X} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

and its effect on the computational basis states is

$$\begin{aligned} \mathrm{CX}_{12}\,|00\rangle &= |00\rangle\,, \\ \mathrm{CX}_{12}\,|01\rangle &= |01\rangle\,, \\ \mathrm{CX}_{12}\,|10\rangle &= |11\rangle\,, \\ \mathrm{CX}_{12}\,|11\rangle &= |10\rangle\,. \end{aligned}$$

Another relevant example is the controlled-Z or CZ gate, which applies a Z gate to the target qubit when the control is in state $|1\rangle$, and does nothing otherwise. The CZ gate is specially relevant in practice, since it is the native entangling gate in many superconducting quantum processors.

3.3.2.2 SWAP gate

The SWAP gate interchanges the state of two qubits. Its matrix representation is

$$\mathrm{SWAP}_{12} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and its action on the computational basis states is

$$\begin{aligned} \mathrm{SWAP}_{12}\,|00\rangle &= |00\rangle\,, \\ \mathrm{SWAP}_{12}\,|01\rangle &= |10\rangle\,, \\ \mathrm{SWAP}_{12}\,|10\rangle &= |01\rangle\,, \\ \mathrm{SWAP}_{12}\,|11\rangle &= |11\rangle\,. \end{aligned}$$

Note that a SWAP gate can be constructed from 3 controlled-X gates as follows,

$$\mathrm{SWAP}_{12} = \mathrm{CX}_{12}\mathrm{CX}_{21}\mathrm{CX}_{12}.$$

It is important to recall that the SWAP gate only exchanges the states of two qubits. Therefore it does not change the degree of entanglement of a bipartite quantum state.

Since the SWAP gate is not entangling, it cannot be used to create a universal set of gates. However, it is still useful in quantum hardware with limited connectivity, where not all pairs of qubits can interact directly. In these architectures, SWAP operations can bring the states of distant qubits into proximity for interaction.

3.3.2.3 iSWAP gate

The iSWAP gate is an interesting example that is closely related to the SWAP gate but is able to generate entanglement. Therefore, it can be used to form a universal gate set. Moreover, the iSWAP is a maximally entangling gate native to some quantum processors based on superconducting circuits or trapped ions.

Its matrix representation is,

$$\mathrm{iSWAP}_{12} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and its action on the computational basis states is

$$\begin{aligned} \mathrm{iSWAP}_{12}\,|00\rangle &= |00\rangle\,, \\ \mathrm{iSWAP}_{12}\,|01\rangle &= i\,|10\rangle\,, \\ \mathrm{iSWAP}_{12}\,|10\rangle &= i\,|01\rangle\,, \\ \mathrm{iSWAP}_{12}\,|11\rangle &= |11\rangle\,. \end{aligned}$$

3.4 Quantum circuits

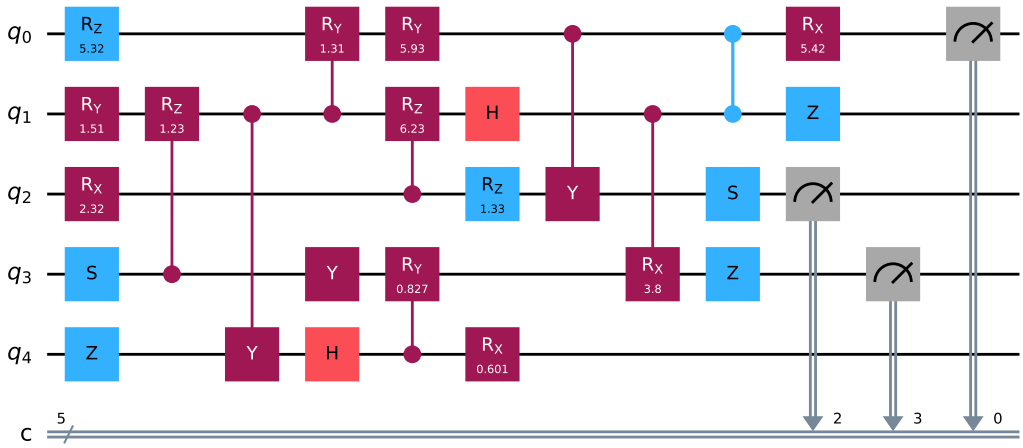


Figure 3.2: Example of a 5-qubit quantum circuit. The circuit was generated using Qiskit [67].

Just as in the classical case, we arrange a series of logic operations on qubits to construct quantum circuits, which implement algorithms. An example of a quantum circuit is shown in Figure 3.2. In this representation,

- Each qubit is represented by a horizontal line, with time progressing from left to right. It is usually assumed that each qubit starts in the state $|0\rangle$.
- Quantum gates are depicted as labeled boxes placed along these lines, indicating the operations applied to the qubits in a specific order.
- Single qubit gates (*e.g.* R_z, H, Z) apply individual transformations to qubits.
- Controlled operations (*e.g.* CR_y) involve interactions between two qubits. These are typically represented with a circle marker on the control qubit and a corresponding gate on the target qubit.

- Measurements, shown at the end with a gauge symbol, project the qubit state onto the computational basis ($|0\rangle$ or $|1\rangle$). The measurement outcomes are stored in classical registers, denoted at the bottom of the figure as c.

For example, in the first qubit, q_0, the circuit first applies a R_z rotation, followed by a CR_y gate controlled by qubit q_1, and so forth.

The complexity of a quantum circuit is typically characterized by three quantities: depth, width, and gate count.

Visual inspection from Figure 3.2 reveals that some gates can be executed in parallel. The depth of a quantum circuit quantifies the minimum number of sequential layers of operations required to run the entire computation. The depth of the circuit is particularly important because it correlates with the execution time of the algorithm. For instance, the Grover search algorithm performs a search on an unstructured database with a circuit depth proportional to $\sqrt{N}$, where N is the number of elements in the database. In contrast, a classical brute-force search requires the order of N sequential operations. This means that the time classical methods take grows quadratically with the size of the database, compared to the Grover algorithm.

The width of a circuit, also referred to as its memory size, is simply the number of qubits it requires.

Finally, the gate count is the number of quantum gates applied. However, not all gates contribute equally to the overall error of a computation. For instance, two-qubit gates introduce noise levels that are approximately two to three orders of magnitude higher than those of single-qubit gates. For this reason, while the gate count usually includes both single-qubit and two-qubit gates, they are reported separately.

The ability of quantum hardware to successfully execute an algorithm depends on how well it can tolerate noise and errors, which are directly influenced by the circuit depth and gate count. Different algorithms have varying levels of noise tolerance, which makes these complexity measures crucial in assessing the feasibility of quantum

computations.

3.5 Fault-tolerant quantum computing

Fault-tolerant quantum computing (FTQC) is a paradigm in which quantum operations can be performed reliably even in the presence of gate errors and decoherence [68]. The collection of techniques used to protect quantum information and achieve this reliability in the presence of noise is called quantum error correction (QEC).

In QEC, a single logical qubit is encoded on multiple physical qubits. This redundancy allows errors to be detected and corrected without destroying the coherence of the quantum states encoded [69, 70]. However, for a quantum system to be fault-tolerant, errors must remain below a specific threshold. Once this threshold is reached, increasing the number of physical qubits per logical qubit allows arbitrarily reducing the effective error rate [71–73].

One of the most promising approaches to QEC are surface codes [74], due to their high error threshold and relatively simple implementation. However, resource requirements blow up, possibly requiring thousands of physical qubits for each logical qubit, depending on the application and physical error rates [75].

Recent developments have shown encouraging progress towards FTQC. In 2024, Google presented the 105-qubit Willow processor, with error rates low enough such that increasing the size of surface codes can extend the lifespan of a logical qubit rather than diminishing it [60]. This is the first experimental verification that the error threshold for fault-tolerance can be surpassed in a real quantum processor. In spite of that, the number of available qubits remains insufficient to implement error-corrected quantum algorithms.

The transition from current devices to FTQC requires:

1. Improved qubit coherence times to reduce the rate of environmental decoher-

ence

2. Enhanced gate fidelities to minimize operational errors

3. More efficient error correction codes to reduce the physical-to-logical qubit ratio

4. Scalable qubit architectures that maintain error rates as system size increases

Achieving FTQC would enable the implementation of large-scale quantum algorithms, such as Shor's factoring algorithm [44], potentially demonstrating quantum advantage for problems of practical significance.

3.6 NISQ computing

While fault-tolerant quantum computers remain a long-term goal, current quantum devices operate in what John Preskill termed the Noisy Intermediate-Scale Quantum (NISQ) era [76]. NISQ devices typically feature 50-100 qubits that suffer from noise and lack error correction. Despite these limitations, NISQ devices may still offer computational advantages for specific problems.

The boundary between NISQ and FTQC computing is not clearly defined. We are entering a transitional stage, sometimes called early fault-tolerant quantum computing (EFTQC) or intermediate-scale quantum computing (ISQ). This stage is characterized by moderate error rates and small logical encodings that allow partial error correction but with limited resources to perform useful calculations [77].

Currently, the utility of quantum processors depends on the execution of relatively shallow quantum circuits with sufficient fidelity to produce meaningful results. To this end, researchers have developed various approaches to extract value from these imperfect devices [78], including: (i) Quantum error mitigation techniques that reduce noise effects without implementing full error correction [79, 80], (ii) custom circuit designs that minimize depth and complexity for specific tasks [81], and (iii)

hybrid quantum-classical approaches that divide computational tasks between quantum and classical processors [82]. In this thesis, we are especially interested in the third approach, customarily named variational quantum algorithms (VQAs).

3.6.1 Variational quantum algorithms

One promising approach to overcoming the limitations of quantum hardware in the short term is variational quantum algorithms (VQAs). These are hybrid computational schemes that combine the capabilities of classical and quantum computers. These algorithms encode the solution to computational problems as optimization tasks, where a parameterized quantum circuit evaluates an objective function, and a classical optimizer iteratively adjusts the parameters to find optimal solutions. To remain compatible with NISQ hardware, the quantum circuit is typically shallow, since deeper circuits accumulate more errors and become harder to train. Nevertheless, the quantum stage of the algorithm operates within an exponentially large Hilbert space, which is generally intractable for classical computers. This opens a window for potential computational advantage whenever the objective function can be more efficiently evaluated on a quantum processor than on a classical machine.

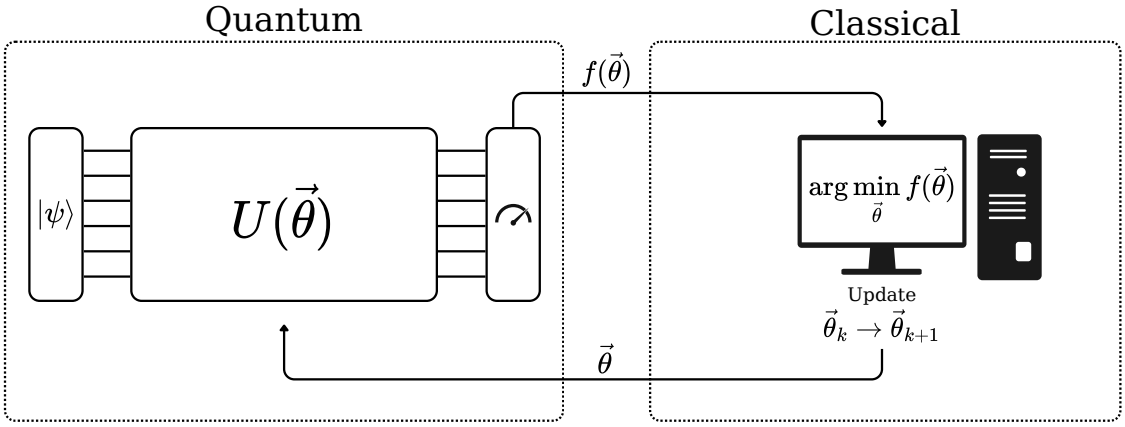


Figure 3.3: Schematization of a generic variational quantum algorithm.

A schematization of a generic VQA is shown on Figure 3.3. The workflow of these algorithms consists of three main components:

(i) A parameterized quantum circuit (PQC):

The quantum part of the algorithm is a shallow quantum circuit represented by a unitary $U(\boldsymbol{\theta})$ with tunable parameters, $\boldsymbol{\theta}$. Depending on the problem to be solved, the circuit has a fixed structure known as *Ansatz*[8], usually composed of a number of layers of parameterized rotations interspersed with entangling gates. The measurements in this circuit produce classical data that can be post-processed to obtain a cost function $f(\boldsymbol{\theta})$. The quantum circuit thus acts as a black box that evaluates the parameters $\boldsymbol{\theta}$ and outputs a value of the corresponding cost function $f(\boldsymbol{\theta})$.

(ii) A Classical optimization loop:

A classical computer processes the measurement data from the PQC, treating the computed cost function $f(\boldsymbol{\theta})$ as an objective function to be minimized.

(iii) An iterative parameter update:

Running on the classical computer, the optimization algorithm ajdusts the parameters based on previous evaluations of $f(\boldsymbol{\theta})$. The optimization strategy may consist on gradient-based (*e.g.* stochastic gradient descent) or gradient-free (*e.g.* Nelder-Mead, SPSA) methods, with the goal of optimizing the cost function over multiple iterations.

By iterating this procedure, the algorithm progressively improves the quantum circuit parameters until reaching a near-optimal solution. The process terminates after a predefined number of iterations or when the solution fulfills some criteria, such as reaching a threshold value, achieving a small gradient magnitude, or satifying a desired level of approximation quality.

[8]This word is German for "an educated guess".

Applications where VQAs could provide computational advantage against classical methods include quantum chemistry [83–85], quantum control [86–88], quantum simulation [89, 90], entanglement detection [91–93], state estimation [94–98], quantum machine learning [99–103], error correction [104], graph theory [105–107], differential equations [108–110], and finances [111].

Despite their promising features, VQAs are not exempt from difficulties. The number of parameters to optimize is typically large, and the estimation of quantities is noisy due to both hardware imperfections and the finite number of measurements. Moreover, the optimization landscape typically presents local minima that further complicate gradient-based optimizers. Among these problems, a particularly problematic phenomenon that limits the scalability of VQAs is the occurrence of barren plateaus: Regions in the parameter space where the objective function gradient becomes exponentially small, in variance and mean, with respect to the number of qubits [112, 113].

Barren plateaus may originate from different sources, and mitigation techniques have focused on several corresponding strategies: Careful initialization [114], problem-specific ansatz design [115], layer-wise training [116], and the use of local cost functions [113]. The development of robust optimization methods for VQAs, particularly those resilient to noise and barren plateaus, is an active area of research.

Although it remains unclear whether VQAs will provide demonstrable quantum advantages in the near term [117], they still represent one of the most practical approaches to extract computational utility from NISQ-era quantum processors. The optimization challenges faced by VQAs, particularly in the presence of noise and high-dimensional parameter spaces, motivate the exploration of stochastic optimization algorithms in Part II of this thesis.

Chapter 4

Quantum-Inspired Computing

"Simplicity is the ultimate sophistication."

Leonardo da Vinci

Although quantum computing is expected to enhance our computational capabilities significantly, we cannot yet build practical quantum computers. As quantum hardware continues to develop, an alternative approach is to design algorithms that take advantage of concepts from quantum mechanics while running on classical computers. This approach, known as quantum-inspired computing, is a class of computational techniques that adopt ideas, mathematical structures, and optimization strategies from quantum theory but operate entirely on classical hardware. These methods are valuable because they can provide exponential or polynomial speed-ups in certain problems or enable more compact data representations [118–122].

This chapter introduces tensor networks, particularly matrix product states (MPS), as tools for efficiently representing high-dimensional data with specific correlation patterns. We explain how tensor decompositions can help overcome exponential scaling in certain problems, develop a physical intuition about the structure and properties of MPS, and introduce an algebraic framework that allows the translation of conventional vector-based numerical methods to the language of MPS. These

methodologies are the foundation of the numerical methods presented in Chapter 12.

4.1 Tensor Networks

4.1.1 Motivation

A core concept in quantum-inspired computing is tensor networks (TN), which are structured arrangements of contracted tensors that enable the efficient representation of high-dimensional data.

Tensor networks originated from the need to simulate quantum many-body systems, where the exponential growth of Hilbert spaces renders the calculations intractable even for moderate system sizes.

Early research on tensor networks began in 1971 with Roger Penrose's work on graphical notation for tensor contractions [123]. In the context of quantum many-body physics, a significant milestone was the development of the Density Matrix Renormalization Group (DMRG) algorithm by Steven White in 1992 [124]. The DMRG approach consisted of systematically truncating the Hilbert space while retaining the most relevant states, according to the eigenvalues of their reduced density matrix. While initially formulated in a language different from that of tensor networks, DMRG was later recognized to be equivalent to a variational optimization in the class of matrix product states (MPS) [125, 126].

The formal connection between DMRG and MPS generated interest in tensor network representations. The MPS formalism provided both a clear mathematical structure and an intuitive graphical representation that facilitated the development of new algorithms.

The theoretical foundations of tensor networks were significantly strengthened by insights from quantum information theory, when Vidal [127] and Verstraete and Cirac [128] explored the relationship between entanglement structure and the efficiency of tensor network representations. This work established a deep connection between the

physical properties of quantum systems and the computational resources required to simulate them effectively. As a result, new algorithms were developed for quantum applications, including the estimation of ground states [129], the time evolution of open and closed quantum systems [130, 131], and quantum computing [132]

Although tensor networks originated in quantum many-body physics, they have since been applied to a wide range of fields, bringing quantum-inspired computational advantages to classical problems. Successful applications include machine learning [133, 134], signal processing [135–137], and solving differential equations [120, 138].

4.1.2 Intuition

Before going into formal definitions, we can develop an intuition on how tensor networks achieve their computational advantages. Consider the simple example of storing a large matrix. A general $M \times N$ matrix requires storing MN elements, which could become prohibitive for large dimensions. However, if the matrix has some underlying structure, we can factorize it into a product of smaller matrices.

For example, some given matrix W could be written as UV, where U is $M \times \chi$ and V is $\chi \times N$. Instead of storing MN elements, we now store only $(M + N)\chi$ elements. When the matrix is large, $M, N \gg 1$, and χ remains bounded, the factorized representation is significantly more memory efficient than storing the entire matrix.

This concept extends naturally to higher-dimensional data. Consider a 3-dimensional tensor T of size $M \times N \times P$, representing, for example, a time-evolving $2D$ image or a three-particle quantum state. Storing this tensor directly requires MNP elements. However, if we can factorize it into a network of smaller tensors, we could achieve some compression.

For example, a simple factorization can be written as

$$T_{ijk} = \sum_{\alpha}^{\chi_\alpha} \sum_{\beta}^{\chi_\beta} A_{i\alpha} B_{\alpha j \beta} C_{\beta k},$$

where A, B and C are smaller tensors connected through shared indices α and β.

The values of χ, χ_α, and χ_β, known as *bond dimensions*, determine the efficiency of the factorized representations and depend on the correlations present in T. When the bond dimensions are much smaller than the original tensor dimensions, we achieve a large compression. In particular, the bond dimensions depend on the amount of entanglement between different partitions of the data. This is formalized in Section 4.2.

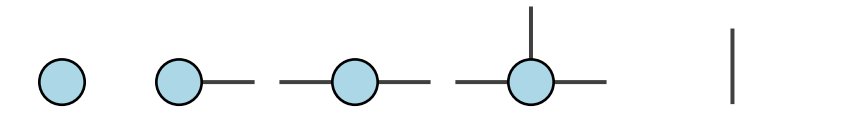
These examples illustrate how tensor contractions enable more efficient data storage than direct representations, and also how quantum mechanical concepts, such as entanglement, emerge in the study of tensor networks.

This is the core insight of tensor networks: Complex, high-dimensional data can be efficiently represented through networks of interconnected lower-dimensional tensors, when correlations allow it.

4.1.3 Penrose diagrams and tensor network algebra

Working with tensor networks algebraically quickly becomes cumbersome as multiple indices are involved. In this context, Penrose's graphical notation [123] provides a powerful visual language that makes tensor operations intuitive. In this notation, each tensor is represented with a node or closed shape, with lines (legs) representing its indices. Tensors with different ranks are depicted as in Figure 4.1: (a) A scalar (rank-0 tensor) has no free legs, (b) a vector (rank-1 tensor) has one free leg, (c) a matrix (rank-2 tensor) has two free legs, (d) a rank-3 tensor has 3 free legs, and (e) the identity matrix (Kronecker[1] delta) has two free legs without a node.

[1]Named after the Prussian mathematician Leopold Kronecker (1823-1891).



(a) scalar (b) vector (c) matrix (d) 3-tensor (e) identity matrix

Figure 4.1: Diagrams for tensors with different dimensionality.

4.1.3.1 Basic operations

Contractions This notation is particularly helpful for representing operations. The operational basis is that connecting two lines corresponds to a *contraction*; the summation over a shared index. Then, two tensors with a contracted leg can be combined into a single tensor. Since indices may have different lengths, legs are sometimes given names, which also serves as a disambiguation in some contexts. An example is shown in Figure 4.2.

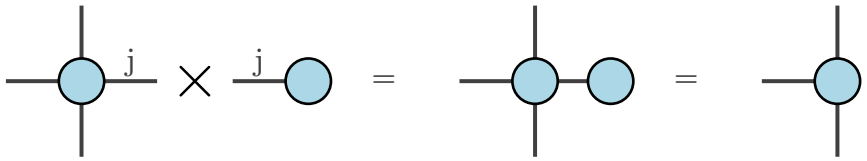


Figure 4.2: Example of the contraction of a 4−tensor with a vector, resulting in a 3−tensor.

Trace of a matrix An important operation is the trace of a matrix, consisting in the sum of the elements in its diagonal,

$$\mathrm{Tr}(M) = \sum_i M_{ii}.$$

The computation of the trace corresponds to contracting both matrix indices together, which is represented in Figure 4.3. This can also be interpreted as contracting both legs of M with both legs of the identity matrix, $\mathrm{Tr}(M) = \sum_{ij} M_{ij}\delta_{ij}$.

Figure 4.3: Trace of a matrix.

Index permutation and transposition The permutation of two indices corresponds to rearranging the position of these legs. For a rank-2 tensor, swapping the two legs equals taking the transpose. This is illustrated in Figure 4.4.

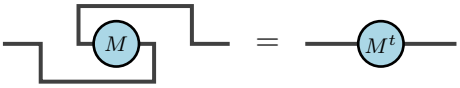


Figure 4.4: Permutation of the two indices of a matrix.

This transposition can also be visualized as a 180° rotation of the diagram that flips the position of the legs. However, this rotation does not include complex conjugation, which must be indicated explicitly.

For example, the squared L^2-norm of a complex vector v involves both transposition and complex conjugation, $\|v\|_2^2 = v^\dagger v$. In diagrammatic form, this corresponds to the connection of v with its Hermitian conjugate $v^\dagger$, as shown in Figure 4.5.

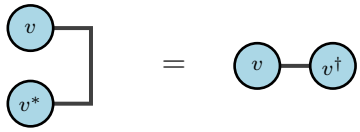


Figure 4.5: Squared L^2-norm of a vector v.

Reshaping Reshaping is an operation that groups or splits indices. Therefore, it changes the rank of a tensor. The reshaping operation is feasible whenever the joint axis has a size that is the product of the split indices. The operation can be understood as a reorganization of the tensor elements, such as stacking the columns of a matrix into a single vector. The reshaping of a vector into a matrix is depicted on Figure 4.6

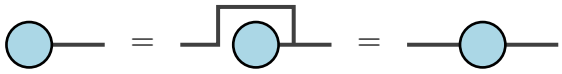


Figure 4.6: Reshaping of a vector into a matrix.

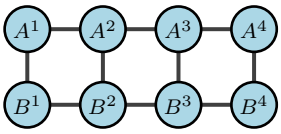


Figure 4.7: Contraction of a tensor network into a scalar.

Derivatives An important advantage of the Penrose notation is that it simplifies computing derivatives and other complex operations. For example, consider the tensor network depicted in Figure 4.7, which is equivalent to a scalar. Computing its derivative with respect to the elements of the rank-3 tensor B^2, in typical notation would be

$$\begin{aligned}\frac{\partial}{\partial_{B^2_{abc}}} \sum_{\{i\}} & A^1_{i_7 i_1} A^2_{i_1 i_8 i_2} A^3_{i_2 i_9 i_3} A^4_{i_3 i_{10}} B^1_{i_7 i_4} B^2_{i_4 i_8 i_5} B^3_{i_5 i_9 i_6} B^4_{i_6 i_{10}} \\ &= \sum_{i_1 i_2 i_3 i_6 i_7 i_9 i_{10}} A^1_{i_7 i_1} A^2_{i_1 b i_2} A^3_{i_2 i_9 i_3} A^4_{i_3 i_{10}} B^1_{i_7 a} B^3_{c i_9 i_6} B^4_{i_6 i_{10}},\end{aligned}$$

which is cumbersome even for a small tensor network. In contrast, in diagramatic notation, the derivative with respect to a tensor element is much easier. Since all contractions are linear, the derivative is achieved simply by removing that tensor and reversing the direction of its connecting legs[2]. The result is shown in Figure 4.8, which is consistent with the algebraic expression, where B^2 does not appear and the free indices are a, b and c.

4.1.4 Common tensor network families

Tensor networks are available in various configurations, each designed to capture specific entanglement patterns. This makes different architectures suitable for par-

[2]We invert the direction since we want the legs to point in the same direction as the tensor in the derivative.

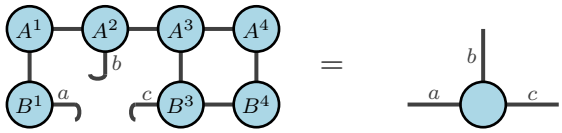


Figure 4.8: Derivative of the diagram in Figure 4.7 with respect to B^2_{abc}. The result is a 3-tensor.

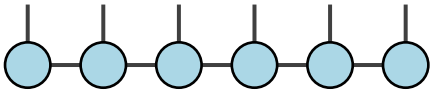


Figure 4.9: Matrix Product State (MPS) structure.

ticular computational tasks. A critical parameter in all tensor network designs is the *bond dimension* χ, which represents the size of the virtual indices connecting different tensors. This parameter controls both the expressiveness of the network and its computational complexity. In the following, we give a short review of some relevant tensor network families, the purpose of their design, and their typical computational complexity in terms of the bond dimension.

Matrix product states (MPS), formalized in 1992 as *finitely correlated states* [139], are one of the simplest and most widely studied families of tensor networks, favored by their linear geometry and computational simplicity. An MPS consists of tensors arranged in a one-dimensional chain, sequentially connected through bonds, as shown in Figure 4.9. This linear geometry makes MPS particularly effective for simulating linearly arranged quantum systems dominated by short-range entanglement, such as spin chains and one-dimensional lattice models with nearest-neighbor interactions. The computational complexity of common MPS operations typically scales as $O(\chi^3)$, making it one of the most efficient families of tensor networks.

Projected entangled pair states (PEPS) generalize MPS to a two-dimensional geometry. A graphical representation of PEPS is provided in Figure 4.10. PEPS form a regular lattice structure that naturally captures two-dimensional entanglement patterns, making them suitable for quantum lattice models. Although PEPS

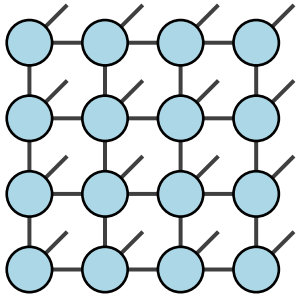


Figure 4.10: Projected Entangled Pair State (PEPS) structure.

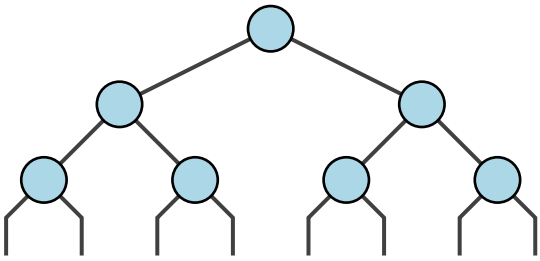


Figure 4.11: Tree tensor network (TTN) structure.

can represent two-dimensional quantum systems more efficiently than MPS, their computational cost is exponential in the number of tensors [140]. Exact contractions of PEPS networks are generally intractable, and approximate contraction schemes must be used with operational complexities around $O(\chi^{10})$ [141].

Tree tensor networks (TTN) organize tensors in a hierarchical tree structure, depicted in Figure 4.11. This architecture efficiently represents states with multiscale entanglement, where correlations exist at different distance scales. TTN offer a middle ground between the efficiency of MPS and the expressive power of more complicated networks. In the field of numerical analysis, they have been found useful for compressing multivariate functions [142]. The computational complexity for typical TTN operations scales as $O(\chi^4)$, making them more expensive than MPS but still tractable for moderate bond dimensions [143].

Multi-scale entanglement renormalization ansatz (MERA) combines a tree structure with additional "disentangler" tensors, designed to handle critical systems where

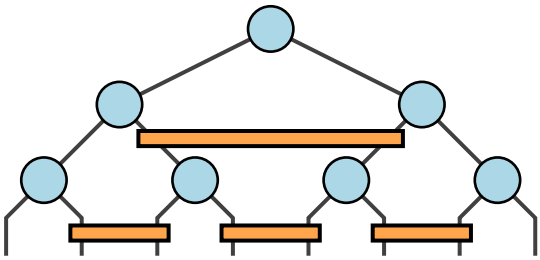


Figure 4.12: MERA structure

correlations decay as power laws. This is represented in Figure 4.12. Disentangler tensors are unitary transformations chosen to minimize entanglement across adjacent blocks, making hierarchical contraction more efficient. This hierarchical structure naturally encodes the logarithmic scaling of entanglement entropy characteristic of critical systems, allowing MERA to efficiently represent states that would require exponentially large bond dimensions in simpler tensor networks. While computationally expensive, with operations typically scaling as $O(\chi^8)$ or higher, MERA has proven particularly valuable for scale-invariant systems [144].

The computational complexity of operations on these networks varies significantly with their geometry. MPS operations typically scale polynomially with the system size, whereas operations on more complex networks like PEPS can be prohibitive. The selection of an appropriate tensor network architecture ultimately depends on the structure of the target problem and the available computational resources. For many practical applications, including the quantum-inspired numerical methods developed in this thesis, MPS provide sufficient expressivity and computational tractability.

4.2 Singular Value Decomposition

4.2.1 Definition

A fundamental numerical tool for linear algebra is the *Singular Value Decomposition* (SVD). It is very relevant in the construction and computation of tensor networks, as it provides a systematic way to factorize an arbitrary matrix $M \in \mathbb{C}^{m\times n}$ as

$$M = U\Sigma V^\dagger, \tag{4.1}$$

where $U \in \mathbb{C}^{m\times m}$ and $V \in \mathbb{C}^{n\times n}$ are unitary matrices satisfying $U^\dagger U = I_m$ and $V^\dagger V = I_n$. The diagonal matrix $\Sigma \in \mathbb{R}^{m\times n}$ contains the singular values of M, which are nonnegative and non-increasingly ordered such that $\Sigma_{ii} \geq \Sigma_{jj} \geq 0$ for $j > i$.

Note that the SVD is not unique. While the diagonal matrix Σ is uniquely determined up to the ordering of its singular values, the unitary matrices are not. For each singular value, the corresponding columns of U and V may incorporate a common phase $e^{i\theta}$, leaving the product $U\Sigma V^\dagger$ unchanged. When the singular values are degenerate, the corresponding columns of U and V are only determined up to a unitary transformation within the degenerate subspace. Moreover, in the case of vanishing singular values, the associated columns of U and V can be chosen freely, as long as the overall matrices are unitary.

Unlike eigenvalues, which are defined only for squared matrices, any matrix M has singular values. They are given by the square roots of the eigenvalues of the positive semidefinite Hermitian matrix $M^\dagger M$ (or $MM^\dagger$). The number of non-zero singular values gives the number of linearly independent columns of M.

The computational complexity of a full SVD is $O(mn\min(m,n))$. For large matrices, $M, N \gg 1$, a truncated SVD is commonly used, which retains only the first k singular values and vectors. This yields the best rank-k approximation to M in the spectral and Frobenius norms [145]. The complexity of computing an exact truncated SVD on a dense matrix is $O(mnk)$, while approximate randomized

algorithms can further reduce it to $O(mn \log k)$ [146].

4.2.2 Connection to the Schmidt decomposition

The *Schmidt decomposition*, which is very relevant in quantum information theory, is stated as follows.

Schmidt decomposition *Let $\mathcal{H}_A$ and $\mathcal{H}_B$ be finite-dimensional Hilbert spaces. Given any bipartite pure state $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, there exist orthonormal bases $\{|u_i\rangle_A\}$ for $\mathcal{H}_A$ and $\{|v_i\rangle_B\}$ for $\mathcal{H}_B$, and nonnegative numbers $\{\sqrt{\lambda_i}\}$, called Schmidt coefficients, such that*

$$|\psi\rangle_{AB} = \sum_{i=1}^{r} \sqrt{\lambda_i} |u_i\rangle_A |v_i\rangle_B. \tag{4.2}$$

The number of Schmidt coefficients, $r \leq \min(\dim \mathcal{H}_A, \dim \mathcal{H}_B)$, is the *Schmidt rank*, which detects entanglement across the bipartition: for $r = 1$, the state is separable, and $r > 1$ implies that the state is entangled. Assuming the state $|\psi\rangle_{AB}$ is normalized, the Schmidt coefficients satisfy $\sum_{i=1}^{r} \lambda_i = \| |\psi\rangle_{AB} \|^2 = 1$.

The Schmidt decomposition can be found using the SVD. A bipartite quantum state $|\psi\rangle_{AB}$ can always be reshaped into a matrix M,

$$M_{ij} = \langle i|_A \langle j|_B |\psi\rangle_{AB} ,$$

such that

$$|\psi\rangle_{AB} = \sum_{ij} M_{ij} |i\rangle_A |j\rangle_B .$$

Performing an SVD (4.1), $M_{ij} = \sum_k S_{kk} U_{ik} V^*_{jk}$, the state can be rewritten in the form of a Schmidt decomposition (4.2),

$$|\psi\rangle_{AB} = \sum_k S_{kk} U |k\rangle_A \otimes V^* |k\rangle_B .$$

That is, the Schmidt decomposition is obtained by performing an SVD on the reshaped coefficient matrix of $|\psi\rangle_{AB}$. Its singular values are the Schmidt coefficients of $|\psi\rangle_{AB}$, a basis of the subsystem A is given by the columns of U and the corresponding basis for the subsystem B is given by the rows of $V^\dagger$.

The SVD can also be used to factorize a tensor with more than two indices by reshaping it into a matrix, performing the SVD in the usual manner, and reshaping again to split the indices back. Consider, for instance, a tensor M_{ijkl} that we want to split into two tensors with indices ijm and mkl. To accomplish this goal, one could join the indices in two groups, $[ij]$ and $[kl]$, and perform an SVD to obtain

$$M_{[ij][kl]} = \sum_m U_{[ij]m} S_{mm} V^\dagger_{m[kl]},$$

where the indices can be reshaped back and S can be absorbed into $V^\dagger$. This is illustrated in Figure 4.13.

In this manner, the SVD can be adapted to factorize any tensor with more than one index. This decomposition allows for the construction of tensor networks and forms a basis to study matrix product states.

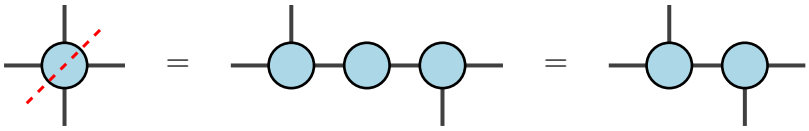


Figure 4.13: Factorization of a tensor with 4 indices via SVD.

4.2.3 Entanglement quantification

Given a bipartite pure state $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, the von Neumann entanglement entropy measures the amount of quantum correlations between subsystems A and

B. The von Neumann entanglement entropy is defined as

$$S_{AB}(|\psi\rangle) = -\mathrm{Tr}\left[\rho_A \log_2 \rho_A\right] \tag{4.3}$$

$$= -\sum_{i=1}^{r} \lambda_i \log_2(\lambda_i), \tag{4.4}$$

where $\rho_A = \mathrm{Tr}_B(|\psi\rangle_{AB} \langle\psi|_{AB})$ is the reduced density matrix for subsystem A, λ_i are its eigenvalues in descending order, and $\sqrt{\lambda_i}$ are the Schmidt coefficients obtained from the decomposition (4.2).

Some notable properties of the entanglement entropy are: (i) It vanishes if and only if the state is separable, $r = 1$ and $\lambda_1 = 1$, (ii) it reaches a maximum value $\log_2(d)$ for maximally entangled states, where all the Schmidt coefficients are equal $\sqrt{\lambda_i} = \sqrt{1/d}$ and $d = r = \min(\dim \mathcal{H}_A, \dim \mathcal{H}_B)$, and (iii) it is invariant under local unitary operations.

In tensor network methods, the entanglement entropy provides insight into the compressibility of the states. For instance, the number of significant Schmidt coefficients across any bipartition sets the minimum bond dimension required for an accurate tensor network representation.

4.3 Matrix Product States

Matrix product states (MPS) provide a compact representation of quantum states by decomposing high-dimensional tensors into a chain of lower-rank tensors. In applied mathematics, they are also known as quantized tensor trains (QTTs) [147], which are a subclass of the broader class of tensor trains (TTs) [148].

In the following, we will introduce how MPS are constructed from the SVD and how an efficient algebra system can be defined to operate with them. We will present our deductions for MPS with physical dimensions of 2 to emulate the encoding quantum bits, but they are easily generalized to encodings with a larger dimension.

4.3.1 Construction from the SVD

Consider an arbitrary quantum state $|\psi\rangle$ belonging to an m-dimensional Hilbert space $\mathcal{H}_m$. We can always embed $\mathcal{H}_m$ into a larger Hilbert space $\mathcal{H}_{2^n} = \mathcal{H}_2^{\otimes n}$ with dimension 2^n, for some value of n, and write

$$|\psi\rangle = \sum_{\alpha=0}^{2^n-1} \psi_\alpha |\alpha\rangle, \tag{4.5}$$

where $|0\rangle, |1\rangle, \ldots, |2^n-1\rangle$ form the computational basis for $\mathcal{H}_{2^n}$. The index α can be split using its binary decomposition, $\alpha = [\alpha_n \ldots \alpha_1]_2 = \sum_i 2^{i-1}\alpha_i$, allowing us to express the state in a basis of n qubits,

$$|\psi\rangle = \sum_{\alpha_1,\alpha_2,\ldots,\alpha_n=0}^{1} \psi_{\alpha_1\alpha_2\ldots\alpha_n} |\alpha_1\rangle|\alpha_2\rangle\cdots|\alpha_n\rangle. \tag{4.6}$$

All information about the state is contained in the tensor $\psi_{\alpha_1\alpha_2\ldots\alpha_n}$, which may be reorganized in different ways. One particularly useful factorization may be obtained by applying the SVD to all consecutive bipartitions of the system. This procedure systematically factorizes the tensor $\psi_{\alpha_1\alpha_2\ldots\alpha_n}$ into a product of smaller tensors, each inheriting a single index from the original tensor. This is the *Matrix Product State* (MPS) representation,

$$\psi_{\alpha_1\alpha_2\cdots\alpha_n} = \sum_{\gamma_1,\ldots,\gamma_n=0}^{\chi_1-1,\cdots,\chi_n-1} A^1_{\gamma_1\alpha_1\gamma_2} A^2_{\gamma_2\alpha_2\gamma_3} \cdots A^n_{\gamma_n\alpha_n\gamma_1}, \tag{4.7}$$

where we imposed open boundary conditions by setting $\gamma_{k+1} = \gamma_1$ with $\chi_1 = 1$. Here, the tensor A^k is referred to as the k-th *site* of the MPS, where $\{\alpha_i\}$ are the *physical indices*, $\{\gamma_i\}$ are the *virtual* or *internal indices*, and $\{\chi_i\}$ are the *bond dimensions*. Both sides of Eq. (4.7) are depicted in Figure 4.14.

Analyzing the storage complexity of the tensor ψ provides insight into the efficiency of the MPS representation. In its naive form (L.H.S. of Eq. (4.7)), ψ is a single tensor with n indices, each of size 2, leading to 2^n total elements. However, in MPS form (R.H.S. of Eq. (4.7)), ψ is factorized into n smaller tensors, where the k-th site

(a) (b)

Figure 4.14: (a) Tensor $\psi_{\alpha_1,\alpha_2,..,\alpha_n}$ at the L.H.S. of Eq. (4.7) and (b) the MPS factorization at the R.H.S. Red lines represent contracted virtual indices. The dashed line is a false sum with $\chi = 1$, from the open boundary conditions.

has three indices $(\gamma_k, \alpha_k, \gamma_{k+1})$. The total number of elements in this representation is $\sum_{k=1}^{n} 2\chi_k\chi_{k+1}$, but since each bond dimension satisfies $\chi_k \leq \chi_{\max}$, the number of elements is upper bounded by $2n\chi_{\max}^2$.

It may seem paradoxical that the storage complexity of ψ is exponential in n in its original form but only linear in its MPS representation, but the explanation is that the bond dimension $\chi_{\max}$ could, in principle, grow exponentially with n. However, the MPS factorization is derived from successive Schmidt decompositions, and χ_k can be as small the Schmidt rank of the corresponding bipartition. This implies that the efficiency of the MPS representation depends directly on the amount of entanglement among partitions of the state. If the entanglement entropy (4.4) is limited for all contiguous bipartitions $A : B$, with A consisting of qubits $\{1, \ldots, k\}$ and B qubits $\{k+1, \ldots, n\}$, then the MPS form is exponentially more efficient than the full tensor representation.

4.3.2 Area and volume entanglement laws

For many physical systems, such as ground states of gapped one-dimensional Hamiltonians, the entanglement entropy obeys an area law [149–151]. This means that the entanglement entropy (4.4) scales as the boundary separating the subsystems. In one dimension, this boundary is just a point, implying that the entanglement

entropy does not scale with the size of the subsystems but remains bounded,

$$S \sim O(1).$$

The physical intuition behind this scaling is that correlations are short-ranged, and that most of the entanglement between the subsystems is localized near the interface.

In contrast, some states such as highly excited, thermal or random states, follow a volume law scaling of entanglement. This means that the entanglement entropy scales linearly with the number of degrees of freedom in the system,

$$S \sim O(n),$$

implying that correlations extend thoughout the entire system.

The bond dimension χ of an MPS dictates the number of Schmidt coefficients retained at each bipartition and, consequently, the amount of entanglement entropy that can be represented. Given the Schmidt decomposition (4.2), the entanglement entropy is given by,

$$S = -\sum_{i=1}^{\chi} \lambda_i \log_2(\lambda_i) \leq \log_2 \chi.$$

This inequality directly implies that the bond dimension χ must grow exponentially with the entanglement entropy,

$$\chi \geq 2^S.$$

For area-law states, where S remains bounded, or critical systems where $S \sim \log_2(n)$, the required bond dimension scales at most linearly with the system size. This ensures that MPS can efficiently approximate those systems with moderate computational resources. However, for volume-law states, where $S \sim O(n)$, the required bond dimension grows exponentially,

$$\chi \sim O(2^n).$$

This makes MPS representations impractical for volume-law states.

4.3.2.1 Examples

Area law Consider the Greenberger-Horne-Zeilinger (GHZ) state [152] in n qubits,

$$|\text{GHZ}\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle^{\otimes n} + |1\rangle^{\otimes n}\right),$$

and let us examine the entanglement across a bipartition defining subsystems A and B for the groups of qubits $\{1,\ldots,k\}$ and $\{k+1,\ldots,n\}$, respectively. In this decomposition, the state can be written

$$|\text{GHZ}\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|0\rangle_A^{\otimes k}|0\rangle_B^{\otimes n-k} + |1\rangle_A^{\otimes k}|1\rangle_B^{\otimes n-k}\right),$$

leading to a reduced density matrix for subsystem A

$$\begin{aligned}\rho_A &= \text{Tr}_B(|\text{GHZ}\rangle_{AB}\langle\text{GHZ}|_{AB}) \\ &= \frac{1}{2}\left(|0\rangle_A^{\otimes k}\langle 0|_A^{\otimes k} + |1\rangle_A^{\otimes k}\langle 1|_A^{\otimes k}\right).\end{aligned}$$

This matrix has two nonzero eigenvalues, each equal to $1/2$. Then, the entropy of entanglement (4.3) yields a constant value

$$S_{AB}(|\text{GHZ}\rangle) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1.$$

Notice that this result is independent of the bipartition chosen. Furthermore, since the S_{AB} does not grow with the system size, the GHZ state admits an exact MPS representation with constant bond dimension $\chi = 2$, independent of the number of qubits.

Volume law Consider a Haar-random pure state in n qubits,

$$|\psi\rangle = \sum_{i_1,\ldots,i_n=0}^{1} c_{[i_n\ldots i_1]_2}|i_1\rangle\cdots|i_n\rangle,$$

where the coefficients c_i follow a complex normal distribution, and the state is normalized such that $\sum_i |c_i|^2 = 1$.

Let us define a bipartition by separating the qubits in two subsystems, $A = \{1, \ldots, k\}$ and $B = \{k+1, \ldots, n\}$. Averaging over Haar-random states, for large subsystem dimensions, $1 \ll 2^k$ and $1 \ll 2^{n-k}$, the entanglement entropy across the bipartition satisfies [153]

$$\mathbb{E}[S_{AB}] \approx k - \frac{2^{2k-n}}{2 \ln 2}. \tag{4.8}$$

Let's consider two regimes. First, if the subsystem A is much smaller than B, $k \ll n$, the entropy scales linearly with the number of qubits in A,

$$\mathbb{E}[S_{AB}] \approx k.$$

Second, for an equal bipartition, $k = n/2$, the entanglement entropy yields

$$\mathbb{E}[S_{AB}] \approx \frac{n}{2} - \frac{1}{2 \ln 2},$$

which is close to the maximal entanglement entropy $S_{\max} = \log_2(2^{n/2}) = n/2$. Then, on average, Haar-random states exhibit a near-maximal entanglement across a balanced bipartition, and require a bond dimension that grows exponentially with the size of the smaller subsystem,

$$\chi_k \sim 2^k,$$
$$\chi_{\max} \sim 2^{n/2},$$

consistent with a volume law scaling. Consequently, they are not amenable to an efficient MPS representation.

4.3.3 Canonical form

An important property of MPS representations is that they have a gauge freedom: Different sets of tensors can represent the same quantum state. In principle, local tensors can be re-parameterized in infinte ways while keeping the overall wavefunction

fixed. For instance, inserting a matrix U and its inverse U^{-1} between two neighboring tensors on the MPS,

$$A^k \to A^k U, \tag{4.9}$$

$$A^{k+1} \to U^{-1} A^{k+1}, \tag{4.10}$$

leaves the global contraction unchanged,

$$\begin{aligned}
\sum_{\gamma_k} A^{k-1}_{\gamma_{k-1}\alpha_k\gamma_k} A^{k}_{\gamma_k\alpha_{k+1}\gamma_{k+1}} &= \sum_{\gamma_k,\gamma'_k} \left[A^{k-1}_{\gamma_{k-1}\alpha_k\gamma_k}\right] \left[A^{k}_{\gamma'_k\alpha_{k+1}\gamma_{k+1}}\right] \delta_{\gamma_k\gamma'_k} \\
&= \sum_{\gamma_k,\gamma'_k,\ell} \left[A^{k-1}_{\gamma_{k-1}\alpha_k\gamma_k} U_{\gamma_k\ell}\right] \left[U^{-1}_{\ell\gamma'_k} A^{k}_{\gamma'_k,\alpha_{k+1}\gamma_{k+1}}\right] \\
&= \sum_{\ell} \left[\sum_{\gamma_k} A^{k-1}_{\gamma_{k-1}\alpha_k,\gamma_k} U_{\gamma_k\ell}\right] \left[\sum_{\gamma'_k} U^{-1}_{\ell\gamma'_k} A^{k}_{\gamma'_k\alpha_{k+1}\gamma_{k+1}}\right],
\end{aligned}$$

as demonstrated in Figure 4.15.

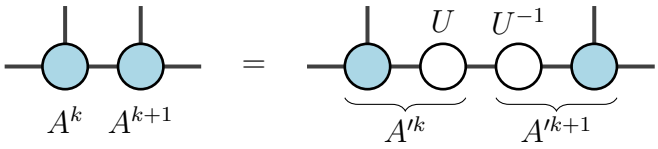


Figure 4.15: Gauge freedom demonstration for the contraction of two sites in an MPS.

Since U and U^{-1} cancel out, the global state remains unchanged, which means that different choices of U between consecutive sites lead to different MPS representations of the same tensor. Canonical forms [154] can be established by imposing specific constraints on the tensors. Specifically, the MPS is in canonical form with respect to a site if its left and right tensors define partial isometries.

Canonical forms, to be formalized in the following, provide several practical advantages that make tensor network methods numerically stable and efficient. For instance, when an MPS needs to be compressed to lower its bond dimension, the SVD-based truncation in canonical form leads to an optimal approximation in the

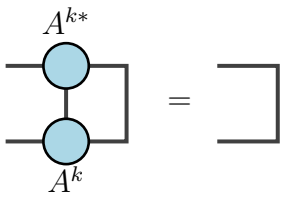


Figure 4.16: Right-canonical condition in diagramatic notation, required for $k > 1$.

Frobenius norm. They also simplify the calculation of norms, expectation values, and the implementation of variational optimization procedures such as DMRG, ensuring that each optimization step focuses on a small subset of tensors, instead of the whole chain.

4.3.3.1 Right-canonical form

For an MPS to be right-canonical, all sites A^k but the first on the left, $k = 1$, are required to satisfy the following orthogonality condition,

$$\sum_{\alpha_k,\gamma_{k+1}} A^{k*}_{\gamma_k\alpha_k\gamma_{k+1}} A^k_{\gamma'_k\alpha_k\gamma_{k+1}} = \delta_{\gamma_k\gamma'_k}. \tag{4.11}$$

This condition, depicted in Figure 4.16, is helpful because it improves the stability of numerical algorithms and simplifies operations. One such operation is to compute the norm of an MPS with open boundary conditions. In this case, every contraction becomes an identity with the exception of the contraction of the leftmost sites, which is the only one that has to be explicitly computed. This is depicted on Figure 4.17.

4.3.3.2 Left- and mixed-canonical form

The left-canonical form imposes an analogous condition to the right-canonical case, but in the opposite direction. An MPS is said to be left-canonical if the tensors at sites $k < n$ satisfy the orthogonality condition,

$$\sum_{\alpha_k\gamma_k} A^{k*}_{\gamma_k\alpha_k\gamma_{k+1}} A^k_{\gamma_k\alpha_k\gamma'_{k+1}} = \delta_{\gamma_{k+1}\gamma'_{k+1}}. \tag{4.12}$$

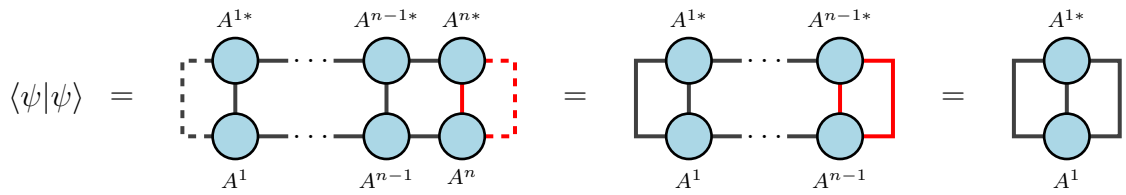


Figure 4.17: Calculation of the norm of a right-canonical MPS. Dashed lines represent false summation indices, possible due to the open boundary conditions, with $\chi_1 = 1$. Contractions in red return an identity, due to the condition in Figure 4.16.

The diagrammatic representation of the left-canonical condition is similar to the right-canonical case but with the direction flipped, as shown in Figure 4.18.

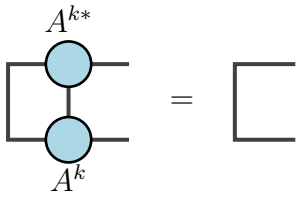


Figure 4.18: Left-canonical condition in diagrammatic notation, required for $k < n$.

In practice, it is often useful to combine both left- and right-canonical forms into a *mixed-canonical* form. A mixed-canonical MPS has a designated orthogonality center at a specific site Γ, meaning that the tensors to the left, $k < \Gamma$ are left-canonical, and the sites to the right, $k > \Gamma$ are right-canonical. This form is widely used in variational approaches like the Density Matrix Renormalization Algorithm (DMRG), where local updates on a few sites are performed while keeping the rest of the MPS in canonical form.

4.3.3.3 Canonicalization by SVD

One of the most common methods for transforming an arbitrary MPS into a canonical form is through the Singular Value Decomposition (SVD). The process relies on the fundamental property of the SVD that any tensor can be decomposed into a partial

isometry, a diagonal matrix of singular values, and another partial isometry.

Right-canonicalization Starting from the rightmost site, each tensor A^k of size $(\chi_k, 2, \chi_{k+1})$ is reshaped into a matrix $(2\chi_k, \chi_{k+1})$ and decomposed using SVD,

$$A^{k,\text{ reshaped}} = U\Sigma V^\dagger.$$

Here U is a unitary matrix, Σ contains the singular values and $V^\dagger$ is another unitary matrix. Then, the matrix $V^\dagger$ of size $(\chi_{SVD}, 2\chi_{k+1})$ is reshaped to $(\chi_{SVD}, 2, \chi_{k+1})$ and replaces the tensor site A^k, while $U\Sigma$ of size (χ_k, χ_{SVD}) is absorbed from the right into A^{k-1},

$$A^k \to V^{\dagger,\text{ reshaped}},$$
$$A^{k-1} \to A^{k-1}U\Sigma.$$

The process is repeated for $k = n-1, \ldots, 1$ until the last site accumulates the nonunitary factor $U\Sigma$.

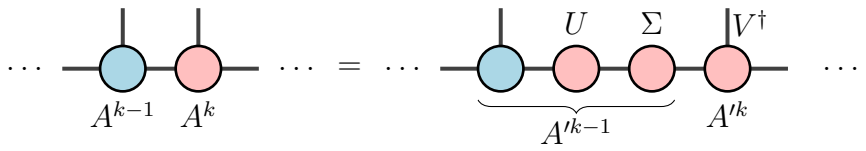


Figure 4.19: Right-canonicalization iteration repeated for each $k > 1$.

Left- and mixed-canonicalization To transform an MPS into the left-canonical form, the analogous procedure to right-canonicalization is performed. Starting from the leftmost site, A^k is decomposed by SVD and replaced with U, while the factor $\Sigma V^\dagger$ is absorbed by A^{k+1}.

In the case of mixed-canonicalization, a left-canonicalization is performed from $k = 1, \ldots, \Gamma - 1$ and a right canonicalization from $k = n, n-1, \ldots, \Gamma + 1$. Then, the orthogonality center A^Γ absorbs the non-unitary factors from both sides. This form is especially useful for computing the expectation values of local operators.

4.3.4 Advantages and limitations of MPS

Matrix product states can emulate the quantum bit encoding of quantum computers, achieving exponential compression for states with limited entanglement [127, 128], particularly for one-dimensional systems that follow an area law scaling of entanglement. This compression transforms what would be an exponentially large representation into one that grows only linearly with the system size.

The memory efficiency of MPS is evident when comparing storage requirements. An arbitrary quantum state of n qubits requires 2^n complex numbers for exact storage. In contrast, an MPS representation with bond dimension χ requires only $O(n\chi^2)$ parameters, offering exponential compression when χ remains bounded. However, this compression is not guaranteed.

The MPS storage efficiency depends on the entanglement structure of the target state. For states obeying an area law of entanglement, where the entanglement between two subsystems scales with the boundary rather than its volume, the required bond dimension χ remains bounded or grows polynomially with the system size n. This includes ground states of gapped local Hamiltonians and many physically relevant states in one dimension.

Conversely, for states with a volume law scaling of entanglement, such as random states, the bond dimension must grow exponentially with the system size to maintain accuracy, negating the advantage of the MPS representation. This establishes a clear domain of applicability: MPS methods are efficient for states with limited entanglement but lose their advantage for highly entangled systems.

Similarly, MPS enable the efficient computation of observables and other physical quantities with operations that scale polynomially rather than exponentially with the system size. The computational complexity of common MPS operations scales polynomially in χ, meaning that computations can become expensive, even if they remain tractable. However, MPS representations allow adaptive bond dimensions that can be dynamically adjusted to balance accuracy and computational complexity,

providing an estimate of the error incurred.

Regarding geometry, MPS adapt naturally and efficiently to one-dimensional systems, but are challenged when applied to higher-dimensional problems where entanglement structures become more complex. Higher dimensional problems can be addressed by flattening the structure into 1D via a snake pattern, but this often leads to artificially high entanglement across the resulting 1D path, requiring larger bond dimensions. Also, choosing different paths lead to different bond dimensions [120]. While extensions like PEPS address some of these issues, they introduce computational difficulties of their own.

It is important to note that MPS provide approximate representations for all quantum states, with accuracy and efficiency determined by truncation parameters. This approximation introduces systematic errors that must be carefully handled. The balance between compression and approximation accuracy is the central trade-off when employing MPS techniques, with a feasibility window provided by the properties of the system that is represented.

4.4 Matrix Product Operators

Matrix Product Operators (MPO) [131, 155] extend the concept of MPS to operators rather than states. Analogously to Equation (4.7), we use the SVD to partition an operator,

$$O = \sum_{\alpha,\beta} O_{\alpha\beta} |\alpha\rangle\langle\beta| \tag{4.13}$$

$$= \sum_{\alpha_1,\ldots,\alpha_n=0}^{1} \sum_{\beta_1,\ldots,\beta_n=0}^{1} O_{\alpha_1\ldots\alpha_n\beta_1\ldots\beta_n} |\alpha_1\rangle \ldots |\alpha_n\rangle\langle\beta_1| \ldots \langle\beta_n|, \tag{4.14}$$

to find the MPO form

$$O_{\alpha_1\ldots\alpha_n\beta_1\ldots\beta_n} = \sum_{\gamma_1,\ldots,\gamma_n=0}^{\chi_1-1,\ldots,\chi_n-1} A^1_{\gamma_1\alpha_1\beta_1\gamma_2} A^2_{\gamma_2\alpha_2\beta_2\gamma_1} \ldots A^n_{\gamma_n\alpha_n\beta_n\gamma_1}. \tag{4.15}$$

(a) (b)

Figure 4.20: (a) Tensor $O_{\alpha_1 \ldots \alpha_n \beta_1 \ldots \beta_n}$ at the LHS of Equation (4.15) and (b) its MPO factorization at the RHS.

Figure 4.20 illustrates this factorization.

MPOs provide an efficient representation of many physically relevant operators, including Hamiltonians with local interactions, allowing for the efficient computation of ground states, expectation values, and time evolution.

4.5 MPS/MPO and algebra

One way of building MPS/MPO is to begin with the states or operators in their full form and perform sequential SVDs to construct their matrix product form. This procedure is useful and typically used in some contexts, but having to form the full tensors may defeat the usefulness of tensor networks. Instead, we would like to construct the basic pieces of our problems directly in their MPS/MPO form and use some algebra to solve our problems and extract information without ever forming large tensors. Here, we describe the fundamental operations of this algebra and their computational costs.

4.5.1 Scalar product

The scalar product between two MPS $|\psi\rangle$ and $|\phi\rangle$ is one of the most fundamental operations. Let,

$$|\psi\rangle = \sum_{\{\alpha\}} \sum_{\{\gamma\}} A^1_{\gamma_1\alpha_1\gamma_2} A^2_{\gamma_2\alpha_2\gamma_3} \dots A^n_{\gamma_n\alpha_n\gamma_2} |\alpha_1, \dots, \alpha_n\rangle$$
$$|\phi\rangle = \sum_{\{\alpha\}} \sum_{\{\mu\}} B^1_{\mu_1\alpha_1\mu_2} B^2_{\mu_2\alpha_2\mu_3} \dots B^n_{\mu_n\alpha_n\mu_2} |\alpha_1, \dots, \alpha_n\rangle .$$

Their scalar product is

$$\langle\phi|\psi\rangle = \sum_{\{\gamma,\mu\}} E^1_{\mu_1\gamma_1\mu_2\gamma_2} E^2_{\mu_2\gamma_2\mu_3\gamma_3} \dots E^n_{\mu_n\gamma_n\mu_1\gamma_1},$$

where we have defined

$$E^k_{\mu_k\gamma_k\mu_{k+1}\gamma_{k+1}} = \sum_{\alpha_k} B^{k*}_{\mu_k\alpha_k\mu_{k+1}} A^k_{\gamma_k\alpha_k\gamma_{k+1}}.$$

At first glance, the product would seem to have computational complexity $O(n\chi^4)$, but following the contraction order depicted in Figure 4.21 it is reduced to $O(n\chi^3)$.

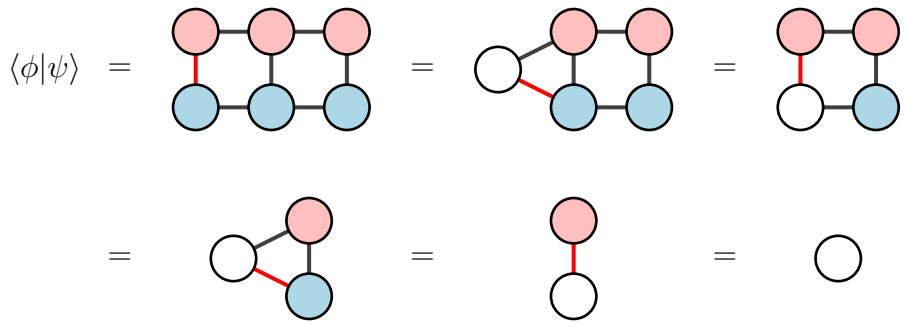


Figure 4.21: Efficient contraction scheme for the inner product of two MPS. Red lines indicate the contraction to be performed at each step.

4.5.2 MPS-MPS addition

The sum of two MPS with tensors $A^k_{\gamma_k\alpha_k\gamma_{k+1}}$ and $B^k_{\mu_k\alpha_k\mu_{k+1}}$ can be performed analytically. The result is an MPS with sites

$$C^k_{\alpha_k} = A^k_{\alpha_k} \oplus B^k_{\alpha_k} = \begin{bmatrix} A^k_{\alpha_k} & 0 \\ 0 & B^k_{\alpha_k} \end{bmatrix},$$

where the matrix representation stands for the two virtual indices, and the physical index is fixed. Therefore, the bond indices on C are $\nu_{k-1} = 0, \ldots, \chi^A_{k-1} + \chi^B_{k-1} - 1$ and $\nu_k = 0, \ldots, \chi^A_k + \chi^B_k - 1$, and the maximum bond dimension grows, at worst, to $\chi_{\max} = \chi^A_{\max} + \chi^B_{\max}$.

Usually, the bond dimension required to express the resulting MPS is much lower, and additional compression can be applied. Alternative variational approaches are described in Section 4.5.5

4.5.3 MPO-MPS application

The application of an MPO to an MPS is another fundamental operation. Given an MPS with sites A^k and an MPO with sites W^k, the result is an MPS $|\phi\rangle = O|\psi\rangle$ with tensors

$$B^k_{[\mu_k\gamma_k]\alpha'_k[\mu_{k+1}\gamma_{k+1}]} = \sum_{\alpha_k} W^k_{\mu_k\alpha'_k\alpha_k\mu_{k+1}} A^k_{\gamma_k\alpha_k\gamma_{k+1}}. \tag{4.16}$$

The resulting MPS has a bond dimension that is a product of the bond dimensions of the original MPS and the MPO, $\chi = \chi^A\chi^O$. This leads to a polynomial scaling of the bond dimensions in repeated applications.

Similarly to the case of the sum of MPS, this can be implemented using an alternative approach detailed in Section 4.5.5.

4.5.4 Simplification or compression

Since every operation on an MPS tends to increase its bond dimensions, an efficient compression algorithm is essential to maintain computational feasibility. The objective is to approximate a given MPS $|\psi\rangle$ by another MPS with the smallest possible bond dimension, which is generally accomplished by two main approaches.

Singular value truncation One widely used approach is sequential singular value truncation [127, 130], which iteratively reduces bond dimensions while maintaining a controlled approximation error. The method proceeds by selecting two neighboring site tensors, say A^1 and A^2, and contracting them into a single tensor. A singular value decomposition (SVD) is then performed on the resulting tensor, and small singular values are truncated based on a predefined numerical tolerance. The tensor is subsequently factorized back into two site tensors, and the same procedure is then applied to the next pair of neighboring tensors, moving sequentially along the chain until reaching the last site. Once the end is reached, the process is reversed, sweeping back in the opposite direction. Each complete forward or backward pass is referred to as a chain sweep, and multiple sweeps are typically performed until the bond dimensions converge.

For each truncation, if the last r singular values s_i are removed, one incurs in an absolute error $\epsilon = \sqrt{\sum_{i=\chi-r}^{\chi} s_i^2}$ in the L^2-norm for the resulting state. Thus, one can keep track of the numerical errors accumulated.

Variational approach The second approach consists of iteratively solving an optimization problem, site by site [126]. The goal is to find an MPS $|\phi\rangle$ with a smaller bond dimension χ that closely approximates the original MPS,

$$\begin{aligned} |\phi\rangle &= \arg\min_{\phi\in\mathcal{M}_\chi} \| |\phi\rangle - |\psi\rangle \|^2 , \\ &= \arg\min_{\phi\in\mathcal{M}_\chi} \langle\psi|\psi\rangle + \langle\phi|\phi\rangle - 2\,\mathrm{Re}\,\langle\psi|\phi\rangle , \end{aligned} \tag{4.17}$$

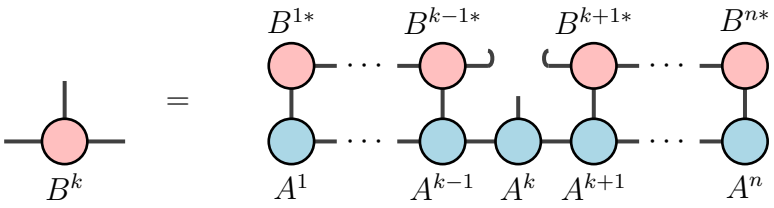


Figure 4.22: Solution to the optimization problem (4.17) for a single site B^k.

where $\mathcal{M}_\chi$ is the manifold of the MPS with prescribed bond dimension χ.

Assuming $|\phi\rangle$ to be in canonical form with orthogonality center at site k, the solution to Equation (4.17) when optimizing with respect to the site B^k is found as

$$\frac{\partial}{\partial B^k}\langle\phi|\phi\rangle = \frac{\partial}{\partial B^k}\langle\psi|\phi\rangle\,. \tag{4.18}$$

This solution, depicted in Figure 4.22, is equivalent to

$$B^k_{\gamma_k\alpha_k\gamma_{k+1}} = U(\phi,\psi)_{\gamma_k\alpha_k\gamma_{k+1}}, \tag{4.19}$$

where U is the contraction of the whole network for $\langle\phi|\psi\rangle$ with the exception of B^{k*}.

The last relation can be used iteratively, sweeping across sites, until convergence is achieved or until some tolerance criteria are met. If the method converges to an MPS that is not close enough to the target MPS, then the algorithm must be run again with a larger bond dimension.

A problem with this method is that it requires one to know the target bond dimension beforehand, but it can be improved by combining two sites and then performing the optimization. Then, the resulting tensor is split via SVD and truncated accordingly, replacing the old tensors. In this manner, the bond dimension is dynamically adapted.

4.5.5 MPS sum and MPO application as optimization problems

In practice, the linear combination of MPS as well as the application of an MPO to an MPS can be solved as extensions to the method for variational compression explained above.

Consider the sum of many MPS $\{|\psi_i\rangle\}$ with coefficients $\{c_i\}$. The resulting MPS can be approximated as

$$|\phi\rangle = \arg\min_{\phi\in\mathcal{M}_\chi} \left\| |\phi\rangle - \sum_i c_i |\psi_i\rangle \right\|^2, \tag{4.20}$$

leading to the iterative, site by site solution

$$B^k_{\gamma_k\alpha_k\gamma_{k+1}} = \sum_i c_i U(\phi,\psi_i)_{\gamma_k\alpha_k\gamma_{k+1}}. \tag{4.21}$$

Similarly, for the application of an MPO to an MPS, one may solve

$$|\phi\rangle = \arg\min_{\phi\in\mathcal{M}_\chi} \| |\phi\rangle - O|\psi\rangle \|^2, \tag{4.22}$$

4.5.6 Other operations

Other operations, such as MPO-MPO addition or composition, can be generalized from the cases shown previously. Moreover, more complex operations can also be defined using these basic operations as building blocks, recasting an MPS as an MPO or vice versa when needed. In the following, we demonstrate this procedure for two particularly useful operations: MPO simplification and elemenwise MPS product.

MPO simplification The simplification of an MPO may be accomplished by recasting it as an MPS with squared physical dimension. This transformation is performed by reshaping each site tensor,

$$O_{\gamma_k\alpha'_k\alpha_k\gamma_{k+1}} \to O_{\gamma_k[\alpha'_k\alpha_k]\gamma_{k+1}}.$$

Then, we can apply standard MPS simplification methods. Once simplified, the tensors can be reshaped back to their original MPO form, resulting in an equivalent but more compact representation of the original operator.

Element-wise MPS product The element-wise product of two MPS can be efficiently computed by recasting one MPS as a diagonal MPO and then applying it to the second MPS. To implement this approach, we define the diagonal MPO with sites B^k that are related to the sites A^k of the original MPS as

$$B^k_{\gamma_k \alpha'_k \alpha_k \gamma_{k+1}} = A^k_{\gamma_k \alpha_k \gamma_{k+1}} \delta_{\alpha'_k \alpha_k}.$$

4.5.7 Summary of computational costs

The efficiency of quantum-inspired algorithms depends on the computational cost of fundamental operations in this tensor network algebra. Table 4.1 summarizes the time complexity of these operations. In these estimates, the cubic scaling with χ arises from tensor contractions and SVD truncation steps.

Operation	Complexity	Result bond dim.
MPS norm / inner product	$O(n\chi^3)$	-
MPO-MPS contraction (exact)	$O(n\chi_O^2\chi^2)$	$\chi_O\chi$
Sum of m MPS (exact)	$O(nm\chi^2)$	$m\chi$
Compression via SVD (per sweep)	$O(n\chi^3)$	$\leq \chi$
Variational sum of m MPS (per sweep)	$O(nm\chi^3)$	$\leq m\chi$
Variational MPO application (per sweep)	$O(n\chi_O^3\chi^3)$	$\leq \chi_O\chi$

Table 4.1: Summary of computational complexities for key MPO-MPS algebra operations. n is the number of sites, χ_O the maximum bond dimension of an MPO, and χ the maximum bond dimension of MPS.

Part II

Stochastic Optimization for Variational Quantum Algorithms

This part of the thesis corresponds to our publication, J. Gidi *et al.* "Stochastic optimization algorithms for quantum applications". In: *Phys. Rev. A* 108 (3 Sept. 2023), p. 032409. DOI: 10.1103/PhysRevA.108.032409. URL: https://link.aps.org/doi/10.1103/PhysRevA.108.032409

Chapter 5

Motivation

"The sea does not reward those who are too anxious, too greedy, or too impatient."

Anne Morrow Lindbergh

As mentioned in Section 3.6.1, the choice of optimization algorithm heavily affects the performance of variational quantum algorithms (VQA). Several studies have contributed to the establishment of general guidelines for choosing the optimization algorithm with the best behavior, according to some predefined metric, for certain classes of problems. These consider methods such as Stochastic Gradient Descent [157], Adaptive Gradient Algorithm [158], Root Mean Square Propagation [159], Adam and variations [160–162], Nelder-Mead [163], Powell method [164], and Newton Conjugate Gradient [165], among many others [166–172].

In the growing list of optimization methods used in hybrid optimization, stochastic optimization algorithms [173–177] play an essential role. State initialization, quantum gates, and measurements are noisy processes that lead to a noisy evaluation of the objective function. This intrinsically stochastic behavior of the objective function negates the mathematical convergence guarantees of commonly used classical optimization methods [178]. However, certain stochastic optimization methods

have convergence proofs that admit the presence of noise. In this scenario, a method that achieves good performance in various applications of hybrid optimization is the simultaneous perturbation stochastic approximation (SPSA) method [179]. The main advantages of SPSA are its robustness to unbiased noise, which is ubiquitous in quantum mechanics, and the fact that it can approximate the gradient of an objective function with only two measurements. In particular, this approximation does not require knowledge of the operational form of the objective function. SPSA has been successfully implemented on several experimental platforms and is one of the standard methods for optimizing variational quantum eigensolvers (VQEs) [180–184], quantum neural networks (QNNs) [185–187], and quantum tomography settings [94, 95, 188].

Given the success of stochastic optimization algorithms within quantum computing, efforts have been made to improve their performance in solving certain tasks. One proposal is the second-order SPSA (2SPSA), which improves the convergence rate of SPSA by preconditioning the gradient with the inverse of a simultaneous perturbation estimate of the Hessian of the objective function [189, 190]. This method is inspired by the deterministic Newton-Raphson algorithm and requires four evaluations of the objective function per iteration to estimate both gradient and Hessian. It has been shown that this method achieves a nearly optimal asymptotic error for well-conditioned problems. However, the error is several orders of magnitude larger for a poorly conditioned Hessian [191].

Another proposal focused on quantum computing is quantum natural gradient optimization [192]. The SPSA algorithm explores the parameter space within a flat geometry, which can lead to an unfavorable update of the parameters. In contrast, quantum natural gradient uses information about the geometry of the parametric quantum state to update the parameters appropriately. The Fubini-Study metric tensor represents this information. Natural gradient optimization provides several advantages over vanilla methods (that is, methods in their standard, unmodified form).

This is because the natural gradient is invariant under re-parametrization [193] and approximately invariant under over-parametrization [194]. SPSA has been adapted to approximate the quantum natural gradient (QN-SPSA) using a simultaneous perturbation estimate of the Fubiny-Study metric tensor [195]. This estimation requires four fidelity evaluations per iteration and the usual two function evaluations required to estimate the gradient. The fidelity evaluations can be performed efficiently using the swap-test [196], among other alternatives [197]. This method is appropriate in contexts where the evaluation of the objective function is too expensive in contrast to the evaluation of the fidelity. An example is to estimate the fundamental energy of molecules [182, 183, 198, 199]. However, similarly to 2SPSA, ill-conditioned metrics can reduce the performance of QN-SPSA [200–202].

Optimization methods can be extended to work in the field of complex numbers using the Wirtinger calculus [203]. Some examples are the complex Newton-Raphson algorithm [204] and the complex quantum natural gradient [205]. These methods optimize the objective function without resorting to the separation of complex variables into real and imaginary parts. In the literature, it has been argued that optimization methods formulated within complex numbers could achieve better performance, which has been observed in a small set of examples [206–208]. This seems to be a more natural approach to optimization in quantum mechanics, where most functions have complex arguments. For example, continuous variable quantum computing employs displacement and squeezing operators, which depend on complex parameters [209, 210].

Recently, the complex simultaneous perturbation stochastic approximation (CSPSA) method [96] has been introduced. This is a generalization of SPSA that optimizes within the field of complex numbers. It has been shown that CSPSA can deliver better results in the estimation of pure states [96] and is robust against noise [98]. It has been applied to entanglement estimation [93], quantum state discrimination [103], and violation of the Claus-Horne-Shimony-Holt inequality [211].

Here, we present a comparative analysis of several stochastic optimization methods applied to real-valued functions of complex variables. We first review the basic principles of the SPSA algorithm. Subsequently, we review the 2SPSA and QN-SPSA algorithms using SPSA as a guideline. We also review the CSPSA algorithm and develop two new optimization algorithms based on the CSPSA algorithm: 2CSPSA and QN-CSPSA. These are the complex field formulations of their real counterparts 2SPSA and QN-SPSA, respectively.

We study the performance of the introduced methods by comparing their convergence rate as a function of the number of iterations with respect to SPSA, 2SPSA, and QN-SPSA. This comparison is carried out in three contemporary applications: variational quantum eigensolver, quantum control, and quantum state estimation.

We use a variational quantum eigensolver to obtain the ground state energy of the Heisenberg Hamiltonian for a 10-qubit ring configuration, which is an ubiquitous and relatively simple model that describes the interactions within a chain of spins [180]. For quantum control, we implement the GRadient Ascent Pulse Engineering (GRAPE) method [212], which is used to engineer quantum gates and states. This method approximates a control pulse by a sequence of pulses of constant intensity. The control parameters of these pulses are optimized to find the best implementation of a given gate or state, even in the presence of noise [86]. In particular, we apply GRAPE to the generation of 5-qubit pure states. Finally, in quantum state estimation, we implement Self-Guided Quantum Tomography (SGQT) [94], which is based on the minimization of the infidelity between an unknown state and a known parametrized state, to characterize 6-qubit pure states. Since the optimization methods studied are stochastic, we use numerical simulations and sampling to estimate the mean, variance (or standard deviation), median, and interquartile range of the relevant figures of merit. Measurements are simulated using a finite sample of various sizes.

Chapter 6

The optimization problem

"A good mathematical problem should be simple to state but difficult to solve."

David Hilbert

Consider the problem of optimizing a real function f of p complex variables, $f : \mathbb{C}^p \to \mathbb{R}$. That is, finding an argument $\boldsymbol{z}^{\triangle} \in \mathbb{C}^p$ such that $f(\boldsymbol{z}^{\triangle})$ is a local minimum of the function f. This problem can be solved by mapping the complex variables to the field of real numbers through the relation $\boldsymbol{z} = \boldsymbol{x} + i\boldsymbol{y}$, in which case f becomes $f(\boldsymbol{\theta})$ with $\boldsymbol{\theta} = (\boldsymbol{x}, \boldsymbol{y})^T \in \mathbb{R}^{2p}$. Then, one can use real variable optimization algorithms to find $\boldsymbol{\theta}^{\triangle} = (\boldsymbol{x}^{\triangle}, \boldsymbol{y}^{\triangle})^T$ such that $f(\boldsymbol{\theta}^{\triangle})$ is a minimum of f, and retrieve the solution for the original complex variable problem as $\boldsymbol{z}^{\triangle} = \boldsymbol{x}^{\triangle} + i\boldsymbol{y}^{\triangle}$. However, it is possible to solve the optimization problem using the Wirtinger calculus [203, 213], which does not resort to mapping complex variables to real ones.

Although both approaches are mathematically equivalent, their practical implications differ when solved. It has been conjectured that a complex variable reformulation of real variable optimization algorithms may lead to increased performance [206–208], which has been observed when working on pure-state quantum tomography [96]. Furthermore, for applications in quantum theory, which are natively stated in terms

of complex variables, the transformation to real variables adds an extra step in the optimization process. For this reason, here we review some real variable optimization methods relevant to quantum applications and present their complex variable analogs.

A particularly suitable class of methods for optimizing multivariate functions in the presence of noisy measurements are the stochastic approximation methods [173–177]. This family of methods originates from the Robbins-Monro algorithm [214] designed to find a root θ of a function $M(\boldsymbol{x})$ given by the expectation of a random variable $Y(x)$. Here, M is unknown, just like the probability function of Y, and the Robbins-Monro algorithm gives an estimate of θ by making successive observations on Y. From the Robbins-Monro algorithm, it is possible to consider M as a regression function [215] and propose a scheme to estimate the maximum of M. Then, the use of stochastic approximations arises to deliver an algorithm that converges to an optimal value of a function f using the Kiefer and Wolfowitz procedure when $M = \nabla f$.

A widely used family of stochastic approximation (SA) methods is based on the iterative rule

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - a_k \boldsymbol{g}_k(\boldsymbol{\theta}_k), \tag{6.1}$$

where the descent step series $a_k = a/(k+A)^s$ is fixed by the externally selected gain parameters a, A, and s. The quantity $\boldsymbol{g}_k$ is a stochastic approximation of the gradient of the objective function at $\boldsymbol{\theta}_k$, which depends on the gain coefficient $b_k = b/k^t$, with b and t externally fixed gain parameters.

In the following subsections, we review the SPSA algorithm and its extension to the second-order and quantum natural gradient algorithms, 2SPSA and QN-SPSA, respectively. Subsequently, we review the CSPSA algorithm for complex variables and develop two extensions to it; the second-order algorithm 2CSPSA and the quantum natural gradient algorithm QN-CSPSA. This work is conducted in a similar way to the SPSA algorithm, by considering an iterative rule as Eq. (6.1), but generalized to the case of complex variables. Finally, at the end of this section, we present

typical modifications to improve the performance of the optimization algorithms, namely blocking and resampling, and introduce two further variations: an alternative Hessian post-processing procedure and a scalar approximation to second-order and quantum natural algorithms that reduce their classical computational cost from cubic to linear in the number of variables.

Chapter 7

Numerical Methods

"Not all those who wander are lost"

J. R. R. Tolkien

7.1 Real-variable methods

7.1.1 SPSA

The simultaneous perturbation stochastic approximation (SPSA) is a multivariate optimization method for real functions of real variables. Although the SPSA denomination came later, the method was first presented by Spall [179] and corresponded to an improvement over the finite difference stochastic approximation (FDSA) of Kiefer and Wolfowitz [215]. Both the FDSA and SPSA algorithms optimize the function $f(\boldsymbol{\theta})$ with $\boldsymbol{\theta} \in \mathbb{R}^p$ by following the recursive stochastic approximation rule Eq. (6.1). However, the main feature of SPSA is that, instead of estimating each of the p components of the gradient as a stochastic finite difference approximation, it

defines the estimator $\boldsymbol{g}_k$ as

$$\boldsymbol{g}_k(\boldsymbol{\theta}) = \frac{f(\boldsymbol{\theta} + b_k\boldsymbol{\Delta}_k) - f(\boldsymbol{\theta} - b_k\boldsymbol{\Delta}_k)}{2b_k} \begin{pmatrix} 1/\Delta_{k,1} \\ \vdots \\ 1/\Delta_{k,p} \end{pmatrix}, \tag{7.1}$$

where $\boldsymbol{\Delta}_k$ is a random perturbation vector with p components typically drawn uniformly from the set $\{\pm 1\}$, and the finite-difference approximation step $b_k = b/k^t$ is controlled by the externally selected gain parameters b and t. It should be noted that, while $\boldsymbol{g}_k(\boldsymbol{\theta}_k)$ does not necessarily have the direction of the gradient at each iteration, it is an asymptotically unbiased estimator of the gradient; it converges to the gradient when averaged in the statistical limit. Furthermore, Eq. (7.1) makes the SPSA algorithm especially suitable for high-dimensional problems, since it always requires 2 function evaluations per iteration, regardless of the number p of variables, in contrast to the FDSA algorithm, which requires $2p$ function evaluations per iteration.

An iteration of the SPSA algorithm is given by Equations (6.1) and (7.1) and requires a total of 2 objective function evaluations.

7.1.2 2SPSA

The iterative rule used in the SPSA algorithm is derived from a first-order gradient descent approximation. Then, the rate of convergence of the algorithm could be accelerated using a second-order iterative rule coming from the Newton-Raphson method, given by

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \left[\mathcal{H}(\boldsymbol{\theta}_k)\right]^{-1} \left(\frac{\partial f}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}_k)\right)^T, \tag{7.2}$$

where $\eta \in \mathbb{R}^+$ is the learning rate and $\mathcal{H}$ is the Hessian of f. A stochastic approximation based on Equation (7.2) is proposed by Spall [189], deriving the so-called adaptive or second-order SPSA (2SPSA) algorithm. The iterative rule now yields

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \overline{a}_k \overline{\mathcal{H}}_k^{-1} \boldsymbol{g}_k(\boldsymbol{\theta}_k), \tag{7.3}$$

where $\overline{a}_k = 1/(k+A)^s$ no longer depends on a. The gradient estimator $\boldsymbol{g}_k$ is defined by Eq. (7.1), as in the first-order case, and $\overline{\mathcal{H}}_k$ is a modified version of the simultaneous perturbations stochastic approximation of the Hessian matrix. In particular, we compute $\overline{\mathcal{H}}_k$ by [189]

$$\mathcal{H}'_k = \frac{\mathcal{H}_k + [\mathcal{H}_k]^T}{2}, \tag{7.4a}$$

$$\mathcal{H}''_k = \frac{k}{k+1}\mathcal{H}''_{k-1} + \frac{1}{k+1}\mathcal{H}'_k, \tag{7.4b}$$

$$\overline{\mathcal{H}}_k = \sqrt{\mathcal{H}''^2_k} + \varepsilon I, \tag{7.4c}$$

where, in execution order, Eq. (7.4a) ensures that the Hessian approximation is symmetric as the analytical Hessian, then Eq. (7.4b) stabilizes the estimator by introducing inertia from previous iterations, starting from an identity at the zeroth iteration, $\mathcal{H}''_0 = I$, and finally, Eq. (7.4c) with $0 < \varepsilon \ll 1$ guarantees positive-definiteness.

A one-sided simultaneous perturbation stochastic approximation to the Hessian matrix is taken as

$$\mathcal{H}_k(\boldsymbol{\theta}) = \frac{\boldsymbol{g}_k(\boldsymbol{\theta} + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) - \boldsymbol{g}_k(\boldsymbol{\theta})}{\tilde{b}_k} \begin{pmatrix} 1/\tilde{\Delta}_{k,1} \\ \vdots \\ 1/\tilde{\Delta}_{k,p} \end{pmatrix}^T, \tag{7.5}$$

which allows reusing the function evaluations from the centered gradient estimator. By inserting the definition of the gradient approximation Eq. (7.1), then Eq. (7.5) can be rewritten by components as

$$[\mathcal{H}_k]_{ij} = \frac{\delta^2 f_k(\boldsymbol{\theta})}{2b_k\tilde{b}_k\Delta_{k,i}\tilde{\Delta}_{k,j}}, \tag{7.6}$$

where

$$\begin{aligned}\delta^2 f_k(\boldsymbol{\theta}) = {} & f(\boldsymbol{\theta} + b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) - f(\boldsymbol{\theta} + b_k\boldsymbol{\Delta}_k) \\ & - f(\boldsymbol{\theta} - b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) + f(\boldsymbol{\theta} - b_k\boldsymbol{\Delta}_k),\end{aligned} \tag{7.7}$$

$\tilde{b}_k = \tilde{b}/k^t$ is a gain series similar in nature to b_k, and $\tilde{\boldsymbol{\Delta}}_k$ is a random vector formed by p components uniformly generated from the set $\{\pm 1\}$ analogous to $\boldsymbol{\Delta}_k$.

Thereby, an iteration of this method is given by Equations (7.7), (7.6), (7.4), (7.1), and (7.3) and requires a total of 4 objective function evaluations.

7.1.3 QN-SPSA

The Gradient Descent method reaches a local minimum by moving, at each iteration, along the direction of the steepest descent of the objective function in the Euclidian parameter space, $-(\partial f/\partial \boldsymbol{\theta})^T$, limiting the magnitude of the update step, $\Delta\boldsymbol{\theta}$. The steepest descent rule can be obtained by choosing the increment as

$$\Delta\boldsymbol{\theta} = \underset{\Delta\boldsymbol{\theta}\in\mathbb{R}^{2p}}{\arg\min} \left\{ \left\langle \left(\frac{\partial f}{\partial \boldsymbol{\theta}}\right)^T, \Delta\boldsymbol{\theta} \right\rangle + \frac{1}{2\eta}\left\|\Delta\boldsymbol{\theta}\right\|_2^2 \right\}, \tag{7.8}$$

where $\eta \in \mathbb{R}^+$ is the learning rate, $\langle \boldsymbol{\theta}, \boldsymbol{\theta}' \rangle = \boldsymbol{\theta}^T\boldsymbol{\theta}'$ is the inner product for two vectors $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, respectively, and $\|\cdot\|_2 = \sqrt{\langle \cdot, \cdot \rangle}$ is the l^2 norm. Differentiating the argument at the right-hand side of Eq. (7.8) with respect to $\Delta\boldsymbol{\theta}$ and setting it to $\mathbf{0}$, provides the well-known gradient descent step

$$\Delta\boldsymbol{\theta} = -\eta \left(\frac{\partial f}{\partial \boldsymbol{\theta}}\right)^T. \tag{7.9}$$

This result is based on the l^2 geometry, where a shift in any direction in the parameter space is equally weighted. However, the objective function may not be equally sensitive to changes in different parameters and, therefore, a more adequate notion of distance would measure the step length $\Delta\boldsymbol{\theta}$ by weighting the changes on each parameter. This is addressed by a method called natural gradient descent [216], which endows the parameter space with a suitable metric $\mathcal{G}$ that induces the norm $\|\cdot\|_{\mathcal{G}} = \sqrt{\langle \cdot, \mathcal{G}\, \cdot \rangle}$. Then, the increment is stated as

$$\Delta\boldsymbol{\theta} = \underset{\Delta\boldsymbol{\theta}\in\mathbb{R}^{2p}}{\arg\min} \left\{ \left\langle \left(\frac{\partial f}{\partial \boldsymbol{\theta}}\right)^T, \Delta\boldsymbol{\theta} \right\rangle + \frac{1}{2\eta}\left\|\Delta\boldsymbol{\theta}\right\|_{\mathcal{G}}^2 \right\}, \tag{7.10}$$

which leads to the natural gradient descent rule

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \left[\mathcal{G}(\boldsymbol{\theta}_k)\right]^{-1} \left(\frac{\partial f}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}_k)\right)^T. \tag{7.11}$$

The quantum natural method, which takes $\mathcal{G}$ as the Fubini-Study metric tensor, is particularly useful for improving convergence rates for optimization problems in quantum applications [192]. The Fubini-Study metric tensor is proportional to the Quantum Fisher information matrix, so its calculation can be very expensive when many variables are involved. This problem was addressed by Gacon *et al.* [195] by taking advantage of the similarity between the Equations (7.2) and (7.11), along with the possibility of writing the Fubini-Study metric tensor as

$$\mathcal{G}(\boldsymbol{\theta}) = -\frac{1}{2}\left[\frac{\partial}{\partial \boldsymbol{\theta}}\left(\frac{\partial F(\boldsymbol{\theta}', \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^T\right]\Bigg|_{\boldsymbol{\theta}'=\boldsymbol{\theta}}, \tag{7.12}$$

where $F(\boldsymbol{\theta}', \boldsymbol{\theta})$ is the fidelity between two quantum states parameterized with the variables $\boldsymbol{\theta}'$ and $\boldsymbol{\theta}$, respectively. In particular, the Fubini-Study metric tensor was approximated according to Eq. (7.12) using the stochastic approximation of the Hessian employed by the 2SPSA algorithm. In this manner, they proposed the quantum natural SPSA (QN-SPSA) algorithm, which avoids the curse of dimensionality.

In order to reuse the equations already presented for the 2SPSA method, we will abuse notation and denote $\mathcal{H}$ the Hessian estimate of the Fubini-Study metric, yielding

$$[\mathcal{H}_k]_{ij} = -\frac{\delta^2 F_k(\boldsymbol{\theta}_k)}{4 b_k \tilde{b}_k \Delta_{k,i} \tilde{\Delta}_{k,j}}, \tag{7.13}$$

where

$$\begin{aligned}\delta^2 F_k(\boldsymbol{\theta}) = {} & F(\boldsymbol{\theta}, \boldsymbol{\theta} + b_k \boldsymbol{\Delta}_k + \tilde{b}_k \tilde{\boldsymbol{\Delta}}_k) \\ & - F(\boldsymbol{\theta}, \boldsymbol{\theta} + b_k \boldsymbol{\Delta}_k) \\ & - F(\boldsymbol{\theta}, \boldsymbol{\theta} - b_k \boldsymbol{\Delta}_k + \tilde{b}_k \tilde{\boldsymbol{\Delta}}_k) \\ & + F(\boldsymbol{\theta}, \boldsymbol{\theta} - b_k \boldsymbol{\Delta}_k),\end{aligned} \tag{7.14}$$

and $\boldsymbol{\Delta}$ and $\tilde{\boldsymbol{\Delta}}$ are two vectors of p components randomly sampled from the set $\{\pm 1\}$.

Following the same logic as in the 2SPSA algorithm, the simultaneous perturbation stochastic approximation of the Hessian Eq. (7.13) must be conditioned by the

procedure on the system of Equations (7.4). Let us note that while we are using the 2SPSA discretization scheme and update rule, this is a first-order method, as the conditioner $\mathcal{H}_k$ comes not from a second-order expansion on the target function but only from a different metric in the parameter space.

Requiring only two measurements of the objective function makes this algorithm especially suitable for problems where the metric tensor can be efficiently approximated. That is when evaluating the fidelity F between two known pure quantum states requires marginal resources compared to the potentially expensive target function f.

An iteration of the QN-SPSA method is given by Equations (7.14), (7.13), (7.4), (7.1), and (7.3) and requires a total of 2 objective function evaluations and 4 fidelity evaluations.

7.2 Complex-variable methods

Now we formulate the problem of optimizing real-valued functions of complex variables. In the case of quantum mechanics, most of the functions that interest us depend on complex variables and their complex conjugates. Consequently, these functions do not satisfy the Cauchy-Riemann conditions and lack a Taylor series expansion. This can be solved by resorting to the real and imaginary parts of the complex variables. Wirtinger calculus [203], however, allows us to define a derivative, the Wirtinger derivative, that exists even for non-holomorphic functions. We consider a function $f : \boldsymbol{\mu} \in \mathbb{C}^{2p} \to \mathbb{R}$ with $\boldsymbol{\mu} = (\boldsymbol{z}, \boldsymbol{z^*})^T$, which can be expressed in a power series for a complex increment $\Delta\boldsymbol{\mu} = (\Delta\boldsymbol{z}, \Delta\boldsymbol{z}^*)^T$,

$$f(\boldsymbol{\mu} + \Delta\boldsymbol{\mu}) = f(\boldsymbol{\mu}) + \frac{\partial f}{\partial \boldsymbol{\mu}}\Delta\boldsymbol{\mu} + \frac{1}{2}\Delta\boldsymbol{\mu}^\dagger \mathcal{H} \Delta\boldsymbol{\mu} + \dots, \tag{7.15}$$

where

$$\mathcal{H} = \frac{\partial}{\partial \boldsymbol{\mu}} \left(\frac{\partial f}{\partial \boldsymbol{\mu}} \right)^\dagger \tag{7.16}$$

is the complex Hessian of the function f [204], the symbol (†) denotes the conjugate transpose, and differentiation with respect to $\boldsymbol{\mu}$ is defined by

$$\frac{\partial f}{\partial \boldsymbol{\mu}} = \left(\frac{\partial f}{\partial \boldsymbol{z}} \, , \, \frac{\partial f}{\partial \boldsymbol{z}^*} \right), \tag{7.17}$$

where the complex variables $\boldsymbol{z}$ and $\boldsymbol{z}^*$ are considered to be independent. Let us note that the inner product between any column two vectors $\boldsymbol{\mu} = (\boldsymbol{z} \quad \boldsymbol{z}^*)^T$ and $\boldsymbol{\mu}' = (\boldsymbol{z}' \quad \boldsymbol{z}'^*)^T$, with $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{C}^p$, is a real number,

$$\boldsymbol{\mu}^\dagger \boldsymbol{\mu}' = (\boldsymbol{z}^* \quad \boldsymbol{z}) \begin{pmatrix} \boldsymbol{z}' \\ \boldsymbol{z}'^* \end{pmatrix} = 2 \, \mathrm{Re}\{\boldsymbol{z}^\dagger \boldsymbol{z}'\}. \tag{7.18}$$

7.2.1 CSPSA

Performing a first-order approximation on $|\Delta\boldsymbol{\mu}|$ from Eq. (7.15), that is,

$$f(\boldsymbol{\mu} + \Delta\boldsymbol{\mu}) - f(\boldsymbol{\mu}) \approx \frac{\partial f}{\partial \boldsymbol{\mu}} \Delta\boldsymbol{\mu}, \tag{7.19}$$

we obtain that the largest decrease of the function f is achieved by a perturbation $\Delta\boldsymbol{\mu}$ in the direction of $-(\partial f/\partial \boldsymbol{\mu})^\dagger$. This provides the complex equivalent to the gradient descent update rule, which is given by the expression

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \eta \left(\frac{\partial f}{\partial \boldsymbol{\mu}} \right)^\dagger, \tag{7.20}$$

where $\eta \in \mathbb{R}^+$ is the learning rate. The above equation yields a stochastic approximation [96] used to introduce the CSPSA algorithm given by the iterative rule

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k - a_k \boldsymbol{g}_k(\boldsymbol{z}_k), \tag{7.21}$$

where $a_k = a/(k + A)^s$. The gradient estimator is now given by

$$\boldsymbol{g}_k(\boldsymbol{z}) = \frac{f(\boldsymbol{z} + b_k \boldsymbol{\Delta}_k) - f(\boldsymbol{z} - b_k \boldsymbol{\Delta}_k)}{2 b_k} \begin{pmatrix} 1/\Delta^*_{k,1} \\ \vdots \\ 1/\Delta^*_{k,p} \end{pmatrix}, \tag{7.22}$$

where $b_k = b/k^t$ and $\boldsymbol{\Delta}_k$ is a random vector with p components uniformly generated from the set $\{\pm 1, \pm i\}$, with i the imaginary unit. To keep the notation simple, we have omitted the dependency of $\boldsymbol{g}_k$ on $\boldsymbol{z}^*$. Consequently, we write $\boldsymbol{g}_k(\boldsymbol{z}, \boldsymbol{z}^*)$ as $\boldsymbol{g}_k(\boldsymbol{z})$ and similarly for other functions.

An iteration of the CSPSA method is given by Equations (7.22) and (7.21) and requires a total of 2 objective function evaluations.

7.2.2 2CSPSA

To obtain a second-order iterative rule, we add up to second-order terms on $|\Delta\boldsymbol{\mu}|$ from expansion Eq. (7.15) and consider the problem of finding the perturbation $\Delta\boldsymbol{\mu}$ that minimizes $f(\boldsymbol{\mu} + \Delta\boldsymbol{\mu})$. This is done by taking $\partial f(\boldsymbol{\mu} + \Delta\boldsymbol{\mu})/\partial\Delta\boldsymbol{\mu} = 0$, which reduces to the equation

$$\left[\frac{\partial}{\partial\boldsymbol{\mu}}\left(\frac{\partial f}{\partial\boldsymbol{\mu}}\right)^\dagger\right]\Delta\boldsymbol{\mu} = -\left(\frac{\partial f}{\partial\boldsymbol{\mu}}\right)^\dagger. \tag{7.23}$$

This can be rewritten in terms of $\boldsymbol{z}$ and $\boldsymbol{z}^*$ as

$$\begin{pmatrix} \mathcal{H}_{zz} & \mathcal{H}_{zz^*} \\ \mathcal{H}_{z^*z} & \mathcal{H}_{z^*z^*} \end{pmatrix}\begin{pmatrix} \Delta\boldsymbol{z} \\ \Delta\boldsymbol{z}^* \end{pmatrix} = -\begin{pmatrix} [\partial f/\partial\boldsymbol{z}]^\dagger \\ [\partial f/\partial\boldsymbol{z}^*]^\dagger \end{pmatrix}, \tag{7.24}$$

where the elements of the block matrix are

$$\mathcal{H}_{zz} = \frac{\partial}{\partial\boldsymbol{z}}\left(\frac{\partial f}{\partial\boldsymbol{z}}\right)^\dagger, \tag{7.25}$$

$$\mathcal{H}_{zz^*} = \frac{\partial}{\partial\boldsymbol{z}}\left(\frac{\partial f}{\partial\boldsymbol{z}^*}\right)^\dagger, \tag{7.26}$$

$$\mathcal{H}_{z^*z} = \mathcal{H}_{zz^*}^\dagger, \text{ and} \tag{7.27}$$

$$\mathcal{H}_{z^*z^*} = \mathcal{H}_{zz}^*. \tag{7.28}$$

The system of Equations (7.24) has the solution

$$\Delta\boldsymbol{z} = \left(\mathcal{H}_{zz} - \mathcal{H}_{z^*z}\mathcal{H}_{z^*z^*}^{-1}\mathcal{H}_{zz^*}\right)^{-1}\left\{\mathcal{H}_{z^*z}\mathcal{H}_{z^*z^*}^{-1}\left(\frac{\partial f}{\partial\boldsymbol{z}^*}\right)^\dagger - \left(\frac{\partial f}{\partial\boldsymbol{z}}\right)^\dagger\right\}, \tag{7.29}$$

which is the update step corresponding to a Newton algorithm [204]. While this solution requires a large number of operations, it is customary to use a block-diagonal approximation, $\mathcal{H}_{zz^*} \approx 0$, yielding a pseudo-Newton method [217] with

$$\Delta \boldsymbol{z} = -\mathcal{H}_{zz}^{-1} \left(\frac{\partial f}{\partial \boldsymbol{z}} \right)^{\dagger}, \tag{7.30}$$

which also has the advantage of being operationally independent of $\boldsymbol{z}^*$ in practice.

Analog to the 2SPSA method, in the stochastic approximation, we take the descent direction given by Eq. (7.30). Thereby, we define the 2CSPSA algorithm by means of the update rule

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k - \overline{a}_k \left[\overline{\mathcal{H}}_k(\boldsymbol{z}_k) \right]^{-1} \boldsymbol{g}_k(\boldsymbol{z}_k), \tag{7.31}$$

where $\overline{a}_k = 1/(k+A)^s$, $\boldsymbol{g}_k(\boldsymbol{z})$ is given by Eq. (7.22), and $\overline{\mathcal{H}}_k$ is a modified version of the simultaneous perturbation stochastic approximation for the partial complex Hessian $\mathcal{H}_{zz}$ at the $k-$th iteration. Similar to the system of Equations (7.4) for the real-variable case, $\overline{\mathcal{H}}_k$ is computed through the sequence

$$\mathcal{H}'_k = \frac{\mathcal{H}_k + [\mathcal{H}_k]^{\dagger}}{2}, \tag{7.32a}$$

$$\mathcal{H}''_k = \frac{k}{k+1} \mathcal{H}''_{k-1} + \frac{1}{k+1} \mathcal{H}'_k, \tag{7.32b}$$

$$\overline{\mathcal{H}}_k = \sqrt{\mathcal{H}''^2_k} + \varepsilon I, \tag{7.32c}$$

where, in execution order, Eq. (7.32a) makes the Hessian approximation hermitian as the exact Hessian, then Eq. (7.32b) stabilizes the estimator by introducing inertia from previous iterations, starting from an identity at the zeroth iteration, that is, $\mathcal{H}''_0 = I$, and finally Eq. (7.32c) with $0 < \varepsilon \ll 1$ ensures positive-definiteness. Note that the regularization Eq. (7.32c) is still valid in the complex-variable case since its input, $\mathcal{H}''_k$, has real eigenvalues due to the previous hermitization Eq. (7.32a).

In this case, the components of the simultaneous perturbation stochastic approximation of the partial complex Hessian $\mathcal{H}_{zz}$ are given by

$$[\mathcal{H}_k(\boldsymbol{z})]_{ij} = \frac{\delta^2 f_k(\boldsymbol{z})}{2 b_k \tilde{b}_k \Delta^*_{k,i} \tilde{\Delta}_{k,j}}, \tag{7.33}$$

where

$$\delta^2 f_k(\boldsymbol{z}) = f(\boldsymbol{z} + b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) - f(\boldsymbol{z} + b_k\boldsymbol{\Delta}_k) \\ - f(\boldsymbol{z} - b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) + f(\boldsymbol{z} - b_k\boldsymbol{\Delta}_k), \tag{7.34}$$

and $\boldsymbol{\Delta}$ and $\tilde{\boldsymbol{\Delta}}$ are two random vectors, each composed by p elements uniformly generated from the set $\{\pm 1, \pm i\}$.

The 2CSPSA method requires the inversion and regularization of a $p \times p$ hermitian complex matrix. In contrast, the analog 2SPSA optimization of an equivalent problem would require the inversion and regularization of a $2p \times 2p$ symmetric real matrix.

An iteration of this method is given by Equations (7.34), (7.33), (7.32), (7.22), and (7.31) and requires a total of 4 objective function evaluations.

7.2.3 QN-CSPSA

The natural gradient method has been adapted for a complex parameter space [205] by posing the usual natural gradient update rule Eq. (7.11) with the relevant metric $\mathcal{G}$ and using an invertible linear transformation W to move back and forth between the real and complex parametrizations such that

$$W \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x} + i\boldsymbol{y} \\ \boldsymbol{x} - i\boldsymbol{y} \end{pmatrix} := \begin{pmatrix} \boldsymbol{z} \\ \boldsymbol{z}^* \end{pmatrix}, \tag{7.35}$$

where $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^p$. However, continuously moving between parameterizations is undesirable, and therefore here we present a natively complex implementation of the natural gradient method for quantum applications, which proceeds analogously to the QN-SPSA method.

The complex gradient descent rule Eq. (7.20) can be obtained as a solution to the optimization problem

$$\Delta\boldsymbol{\mu} = \underset{\Delta\boldsymbol{\mu}\in\mathbb{C}^{2p}}{\arg\min} \left\{ \left\langle \left(\frac{\partial f}{\partial \boldsymbol{\mu}}\right)^{\dagger}, \Delta\boldsymbol{\mu} \right\rangle + \frac{1}{2\eta} \left\| \Delta\boldsymbol{\mu} \right\|_2^2 \right\}, \tag{7.36}$$

where $\eta \in \mathbb{R}^+$ is the learning rate, $\langle \boldsymbol{\mu}, \boldsymbol{\mu}' \rangle = \boldsymbol{\mu}^\dagger \boldsymbol{\mu}'$ is the inner product for two complex vectors $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$, respectively, and $\| \cdot \|_2 = \sqrt{\langle \cdot, \cdot \rangle}$ is the l^2 norm. As in the real case, to require the parameter update to remain small in the space endowed with metric $\mathcal{G}$, the l^2 norm is replaced in Eq. (7.36) by $\| \cdot \|_{\mathcal{G}} = \sqrt{\langle \cdot, \mathcal{G} \cdot \rangle}$. This leads to the optimization problem

$$\Delta\boldsymbol{\mu} = \underset{\Delta\boldsymbol{\mu} \in \mathbb{C}^{2p}}{\arg\min} \left\{ \left\langle \left(\frac{\partial f}{\partial \boldsymbol{\mu}} \right)^\dagger, \Delta\boldsymbol{\mu} \right\rangle + \frac{1}{2\eta} \| \Delta\boldsymbol{\mu} \|_{\mathcal{G}}^2 \right\}, \tag{7.37}$$

which has the solution

$$\Delta\boldsymbol{\mu} = -\eta \mathcal{G}^{-1} \left(\frac{\partial f}{\partial \boldsymbol{\mu}} \right)^\dagger, \tag{7.38}$$

where $\mathcal{G}$ is an hermitian matrix.

In the case that the optimization space is the set of pure quantum states, the metric $\mathcal{G}$ can be chosen proportional to the Quantum Fisher complex information matrix [218], that is,

$$\mathcal{G} = -\frac{1}{2} \left[\frac{\partial}{\partial \boldsymbol{\mu}} \left(\frac{\partial F(\boldsymbol{\mu}', \boldsymbol{\mu})}{\partial \boldsymbol{\mu}} \right)^\dagger \right] \Bigg|_{\boldsymbol{\mu}' = \boldsymbol{\mu}}, \tag{7.39}$$

where $F(\boldsymbol{\mu}', \boldsymbol{\mu})$ is the fidelity between two states parameterized with variables $\boldsymbol{\mu}'$ and $\boldsymbol{\mu}$, respectively.

Considering, as in the 2CSPSA case, a block-diagonal approximation of $\mathcal{G}$, the first row of Eq. (7.38) yields

$$\Delta\boldsymbol{z} = -\eta \mathcal{G}_{zz}^{-1} \left(\frac{\partial f}{\partial \boldsymbol{z}} \right)^\dagger, \tag{7.40}$$

where $\mathcal{G}_{zz}$ is the top left block of $\mathcal{G}$.

Given the Hessian form of $\mathcal{G}_{zz}$ and considering the similarity of Equations (7.40) and (7.30), we can borrow the discretization scheme from 2CSPSA to approximate $\mathcal{G}_{zz}$. Denoting $\mathcal{H}_k$ as the simultaneous perturbation stochastic approximation of $\mathcal{G}_{zz}$ at iteration k, allows us to reuse the equations already presented for 2CSPSA giving

$$[\mathcal{H}_k]_{ij} = -\frac{\delta^2 F_k(\boldsymbol{z}_k)}{4 b_k \tilde{b}_k \Delta_{k,i}^* \tilde{\Delta}_{k,j}}, \tag{7.41}$$

with

$$\delta^2 F_k(\boldsymbol{z}) = F(\boldsymbol{z}, \boldsymbol{z} + b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) - F(\boldsymbol{z}, \boldsymbol{z} + b_k\boldsymbol{\Delta}_k) \\ - F(\boldsymbol{z}, \boldsymbol{z} - b_k\boldsymbol{\Delta}_k + \tilde{b}_k\tilde{\boldsymbol{\Delta}}_k) + F(\boldsymbol{z}, \boldsymbol{z} - b_k\boldsymbol{\Delta}_k), \tag{7.42}$$

and conditioning as in the system of Equations (7.32). As before, $\boldsymbol{\Delta}$ and $\tilde{\boldsymbol{\Delta}}$ are two random vectors, each composed by p elements uniformly generated from the set $\{\pm 1, \pm i\}$.

An iteration of this method, which we call quantum natural CSPSA (QN-CSPSA), is given by Equations (7.42), (7.41), (7.32), (7.22), and (7.31) and requires a total of 2 objective function evaluations and 4 fidelity evaluations.

7.3 Method Improvements

The previous sections presented the optimization methods in their vanilla form. However, it is possible to introduce further modifications that can improve their convergence properties. In particular, we will address two typical modifications, blocking and resampling, and two extra variations we propose: an alternative Hessian postprocessing procedure and a scalar approximation for the preconditioned methods.

7.3.1 Blocking

This technique consists in blocking the progression of the method if the updated parameters $\boldsymbol{z}_{k+1}$ do not meet a given criterion. Conventionally, the updated variable is required to improve the value of the objective function with respect to the previous iteration plus some fixed non-negative tolerance,

$$f(\boldsymbol{z}_{k+1}) < f(\boldsymbol{z}_k) + \delta. \tag{7.43}$$

The tolerance δ is usually set as twice the approximate standard deviation of the noise in the evaluation of the objective function, which can be estimated by collecting

several evaluations at the initial value of the parameters. Regardless of whether the step is accepted, the Hessian estimate $\mathcal{H}_k''$ from Equations (7.4b) and (7.32b) is typically updated at every iteration, to allow the estimator to incorporate information from all collected data [189].

7.3.2 Resampling

This technique is also known as gradient or Hessian averaging. It consists in computing the random estimators for the gradient and Hessian N_R times per iteration to perform the update of the corresponding variable using the average of these estimators. This practice is recommended in noisy environments [189].

Note that the authors of QN-SPSA [195] implement resampling only for the Hessian estimator with the premise that evaluating the metric is cheaper than evaluating the objective function, which could lead to a better convergence rate with little increment on the experimental resources. However, here we stick to the convention stated by [189], which is also implemented on Qiskit [67].

7.3.3 Post-processing

Several post-processing procedures have been proposed to improve the stability of the 2SPSA algorithm [176]. We consider two alternatives; the original proposal given by Equations (7.4), and the procedure given by

$$\mathcal{H}_k' = \frac{\mathcal{H}_k + [\mathcal{H}_k]^\dagger}{2}, \tag{7.44a}$$

$$\mathcal{H}_k'' = \sqrt{\mathcal{H}_k'^2 + \varepsilon I}, \tag{7.44b}$$

$$\overline{\mathcal{H}}_k = \frac{k}{k+1}\overline{\mathcal{H}}_{k-1} + \frac{1}{k+1}\mathcal{H}_k''. \tag{7.44c}$$

The main differences are that our proposal applies regularization before incorporating inertia, and the regularization process is more aggressive for the same value

of the regularization parameter ε. We expect it to perform better in the worst-case scenario of highly noisy or poorly conditioned problems.

7.3.4 Scalar Preconditioning Approximation

Preconditioned methods, such as 2SPSA, 2CSPSA, QN-SPSA, and QN-CSPSA, adaptively adjust the descent direction and magnitude by adding a preconditioner to the stochastic approximation. However, these methods can exhibit numerical instabilities due to the inversion of a possibly ill-conditioned Hessian estimation. Postprocessing procedures can partially mitigate these issues, but these methods still lack consistency in numerical simulations compared to first-order methods. Most likely, these problems are caused by an inadequate adjustment of the descent direction. We consider these problems most likely induced by an inadequate adjustment of the descent direction.

It has been suggested [191] to replace the Hessian estimation with a scalar function of its eigenvalues. Thereby, the descent direction is chosen according to the first-order gradient estimator while retaining the descent magnitude adaptivity from the preconditioner. Following these considerations, we propose a scalar approximation to the Hessian estimates (7.6), (7.13), (7.33), and (7.41). Specifically, we omit the stochastic perturbations $\boldsymbol{\Delta}_k$ and $\tilde{\boldsymbol{\Delta}}_k$ presented in the Hessian estimates to only adjust the descent magnitude and preserve the first-order descent direction. Namely, we approximate the Hessian estimate of the second-order methods by

$$\mathcal{H}'_k = \frac{\delta^2 f_k(\boldsymbol{z})}{2b_k\tilde{b}_k}, \tag{7.45}$$

and the Hessian estimates for quantum natural optimizers by

$$\mathcal{H}' = -\frac{\delta^2 F_k}{4b_k\tilde{b}_k}. \tag{7.46}$$

From this procedure, we consider a new set of second-order and quantum natural methods where the computational complexity is reduced. Namely, the number of

classical operations on each iteration is reduced from $O(p^3)$ to $O(p)$ where p is the number of variables.

Chapter 8

Applications and results

"An algorithm must be seen to be believed."

Donald Knuth

We study the performance of the above optimization methods by comparing the rate of convergence of the objective function towards the minimum as a function of the number of iterations. We consider three important applications: variational quantum eigensolver, quantum control, and quantum state estimation. We use the variational quantum eigensolver to obtain the ground state energy of the Heisenberg Hamiltonian, which is an ubiquitous and relatively simple model that describes the interactions within a chain of spins. For quantum control, we implement the GRadient Ascent Pulse Engineering (GRAPE) method [212], which approximates a control pulse by a sequence of constant-intensity pulses. The control parameters of this pulse are optimized to find the best implementation of a given state, even in the presence of noise [86]. Finally, for quantum state estimation, we implement Self-Guided Quantum Tomography (SGQT) [94], based on minimizing the infidelity between an unknown state and a known parameterized state.

The studied optimization methods are stochastic. We use ensembles of numerical simulations to estimate the mean, standard deviation, median, and interquartile

range of the objective function. Measurements are simulated by sampling a multinomial distribution with various numbers of trials. In the figures below, only the upper half of the standard deviation is shown.

We test the optimization methods considering different configurations and look for the ones that offer the best performance. The configurations we tested are all possible combinations of the following alternatives: with or without blocking, resampling with $N_R = 1, 2, 5$, the two basic post-processing procedures, Equations (7.32) or Equations (7.44), and standard, asymptotic or static set of gain coefficients. The standard set is given by $a = 3$, $b = 0.1$, $A = 0$, $s = 0.602$ and $t = 0.101$, the asymptotic set by $a = 3$, $b = 0.1$, $A = 0$, $s = 1$ and $t = 1/6$, and the static set by $a = 0.01$, $b = 0.01$, $A = 0$, $s = 0$ and $t = 0$.

For clarity, we consider simulations with two groups of methods: (i) vanilla methods and (ii) improved methods, that is, the vanilla methods implemented with the improvements proposed in Section 7.3. The reason behind this separation lies in the drastic increase in resources required to perform blocking and resampling, and it could be useful to be able to discriminate when it is really worth swapping resources for better results.

We have created a freely available library [2] that contains the codes in the Julia programming language [1] that implements all the optimization methods.

8.1 Variational Quantum Eigensolver

The search for the ground state and its energy E_0 of a Hamiltonian is a problem of great interest in areas such as computational chemistry and condensed matter physics. This is because much of the phenomenology and properties of quantum systems can be studied from the ground state and its energy. However, finding this eigenstate in large systems is not a trivial task. It is often infeasible due to the exponential growth of the dimension of the Hilbert space with respect to the number

of subsystems. For large systems, the Rayleigh-Ritz method [219, 220] is a useful tool since it is limited to searching a parameterized subset of the original Hilbert space to reduce the computational cost of optimization. A further reduction in computational cost is achieved using the variational quantum eigensolver (VQE) method [221]. This consists of performing the Rayleigh-Ritz method with the help of a classical and a quantum computer, which makes it a promising tool for the current generation of quantum technologies.

The goal is to find the eigenstate $|\psi_0\rangle$ associated with the lowest eigenvalue E_0 of a Hamiltonian. This ground state can be characterized as the solution to the optimization problem

$$E_0 = \min_{|\psi\rangle} \frac{\langle\psi|H|\psi\rangle}{\langle\psi|\psi\rangle}. \tag{8.1}$$

The Rayleigh-Ritz method provides an estimate of E_0 by parameterizing the trial states as $|\psi(\boldsymbol{\theta})\rangle$ and optimizing over the vector $\boldsymbol{\theta}$ of parameters. The underlying idea is that the subset defined by the parameterization must have a smaller dimension than the total Hilbert space to reduce the computational cost.

The VQE method considers the generic hermitian Hamiltonian operator $H = \sum_{i=1}^{n} h_i\sigma_i$ and the trial states parameterization

$$|\psi(\boldsymbol{\theta})\rangle = U_N(\theta_N)\cdots U_1(\theta_1)|\mathbf{0}\rangle, \tag{8.2}$$

where $U_i(\theta_i)$ are quantum gates parameterized by θ_i and applied one after the other to the initial state $|\mathbf{0}\rangle$. This parameterization corresponds to a variational quantum circuit. The average energy $\langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle$ can then be computed by individually measuring each term $\langle\psi(\boldsymbol{\theta})|\sigma_i|\psi(\boldsymbol{\theta})\rangle$ on a quantum computer and adding the results weighted with their respective coefficients h_i. Thereafter, the values of $\langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle$ are used by a suitable optimization method running on a classic computer.

In general, the VQE method uses SPSA as the optimization algorithm due to its robustness against noise, which suggests that the optimization methods presented

here can also be used. In order to evaluate the performance of these various methods, we use as a testing ground the problem of finding the ground state energy of the Heisenberg Hamiltonian, which models the magnetic interaction of a ferromagnetic lattice. This is given by the expression

$$H_H = j \sum_{\langle m,n \rangle} \sum_{k=x,y,z} \sigma_m^k \sigma_n^k + h \sum_m \sigma_m^z, \tag{8.3}$$

where j and h are dimensionless coupling constants between neighboring sites and with an external magnetic field, respectively, $\sigma_m^x, \sigma_m^y, \sigma_m^z$ are the single-qubit Pauli operators acting on the m-th lattice site, and $\langle m, n \rangle$ indicates that the sum is performed on the nearest neighbors in the lattice. To parameterize the trial states we use the complex parametric single-qubit gate

$$W(z) = e^{-i(z\sigma_+ + z^*\sigma_-)}, \tag{8.4}$$

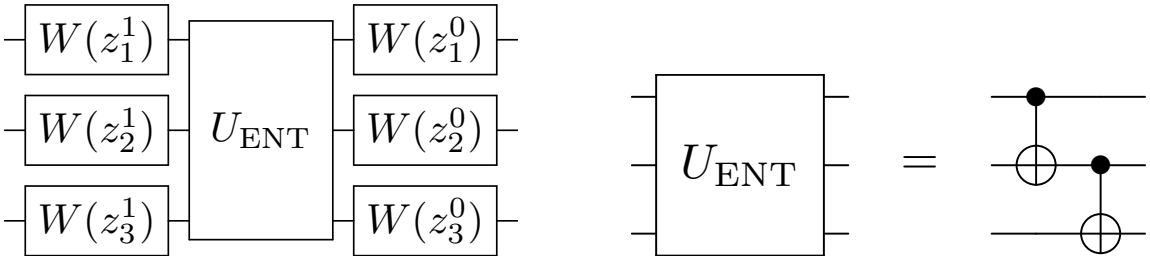
where z is a complex parameter and $\sigma_\pm = \sigma^x \pm i\sigma^y$. This gate can be implemented experimentally by a sequence of three real parameter gates.

The parameterization used for the trial states is given by

$$|\psi(\mathbf{z})\rangle = \prod_{q=1}^{N} W_q^d(z_q^d) U_{\text{ENT}} \cdots \prod_{q=1}^{N} W_q^1(z_q^1) U_{\text{ENT}} \prod_{q=1}^{N} W_q^0(z_q^0) |\mathbf{0}\rangle, \tag{8.5}$$

where W_q^l corresponds to applying W on the q-th qubit, l indicates the layer of the circuit, and U_{ENT} is the three-qubit entangling gate depicted in Figure 8.1b. Similarly for z_q^l.

To evaluate the performance of the different algorithms we consider the Heisenberg Hamiltonian Eq. (8.3) with $h = 0.3$ and $j = 1$ for a ring of 10 qubits with periodic boundary conditions $q_{i+10} = q_i$. The trial states are parametrized by Eq. (8.5) with $d = 1$ entangling layers, as depicted by the circuit in Figure 8.1a. Each algorithm is simulated considering 10^2 randomly selected initial states according to a Haar distribution, which allows estimating statistical indicators such as mean, median, standard deviation, and interquartile range. The measurements required



(a) Parametric circuit (b) Entangling gate

Figure 8.1: Implementation of the VQE circuit for the Heisenberg Hamiltonian with three qubits.

by each optimization method are simulated with an ensemble size of 2×10^4. The standard gain coefficients used in first-order algorithms are $b = 0.1$, and a follows a calibration based on [180].

Figure 8.2 displays the best performance of the vanilla algorithms as a function of the number of iterations. All methods delivered the best results using the post-processing Eq. (7.44). This figure shows that the best performers are the real and complex first-order and the quantum natural complex algorithms, which exhibit an almost indistinguishable behavior in mean, median, standard deviation, and interquartile range. These algorithms converge to a minimum at approximately 2×10^2 iterations, after which become approximately constant. The second-order algorithms exhibit a slower convergence, reaching a similar value only after 7×10^2 iterations.

Figure 8.3 displays the best performance of the improved methods. In this case, the best performers are the first-order methods, the scalar version of second-order methods, and the scalar version of quantum natural methods. These exhibit an almost indistinguishable convergence in mean and median as well as similar dispersion. In particular, a minimum is achieved at approximately 10^2 iterations after which the energy becomes nearly constant. 2CSPSA and 2SPSA scalar methods use post-processing of Equations (7.32) and (7.44), respectively, and standard gains. QN-CSPSA and QN-SPSA scalar methods use post-processing of Equations (7.32)

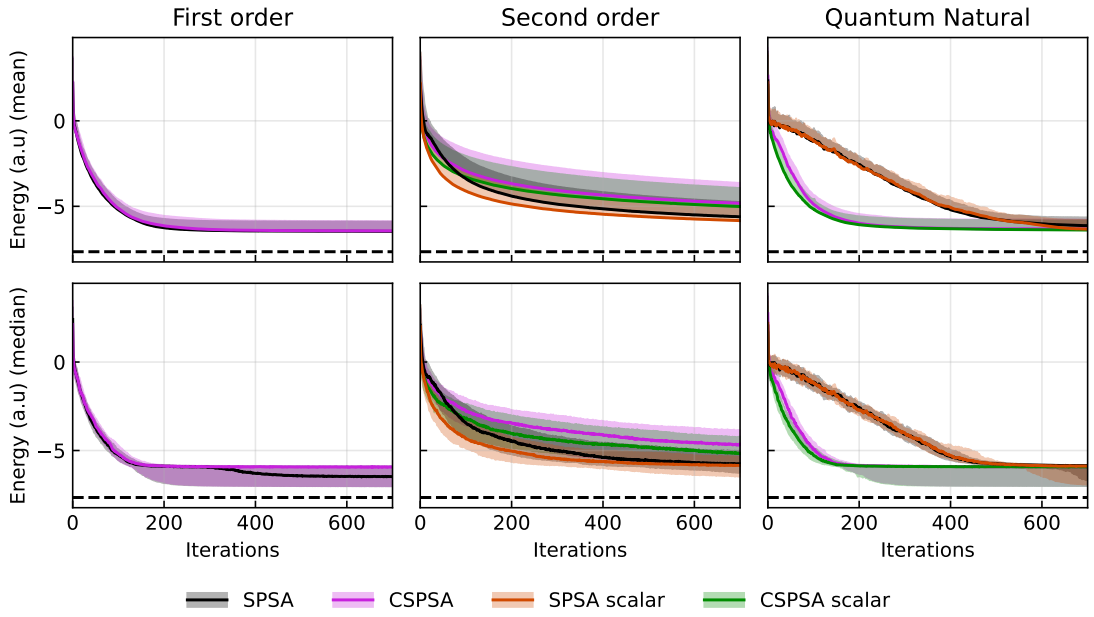


Figure 8.2: The mean (top row) and median (bottom row) of the energy (in arbitrary units) as a function of the number of iterations obtained through the VQE for the Heisenberg Hamiltonian in a 10-qubit ring configuration using vanilla optimization algorithms. The shaded areas represent the variance (top row) and the interquartile range (bottom row). The dashed line indicates the exact minimum. The statistics are obtained from a sample of 10^2 randomly generated states to estimate the minimum energy. The measurements in each circuit were estimated with 2×10^4 shots. The values of the gain coefficients and post-processing class can be found in Table A.1 of Appendix A.

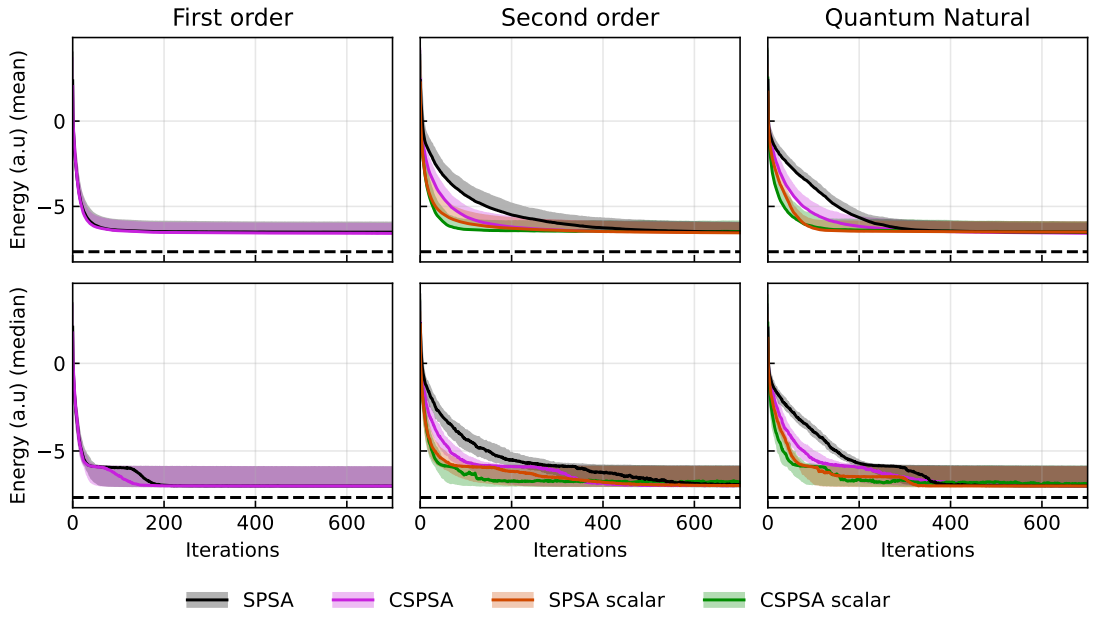


Figure 8.3: The mean (top row) and median (bottom row) of the energy as a function of the number of iterations obtained through the VQE for the Heisenberg Hamiltonian in a 10-qubit ring configuration using improved optimization algorithms. The shaded areas represent the variance (top row) and the interquartile range (bottom row). The dashed line indicates the exact minimum. The statistics are obtained from a sample of 10^2 randomly generated states to estimate the minimum energy. The measurements in each circuit were estimated with 2×10^4 shots. The values of the gain coefficients, post-processing class, and the setting of resampling and blocking can be found in Table A.2 of Appendix A.

and (7.44), respectively, and asymptotic gains. All best performers use resampling with $N_R = 5$.

From Figures 8.2 and 8.3 we conclude that the best performance in the variational quantum eigensolver is achieved using SPSA, CSPSA, QN-CSPSA, and QN-CSPSA scalar methods, which does not significantly differ in their vanilla or improved versions. Blocking and resampling lead to a clear improvement of the second-order methods, delivering results similar to the best performers.

First-order algorithms provide the best performance for this particular problem. Nonetheless, it's crucial to note that this level of performance was attained through a resource-intensive search for gain coefficients. In the absence of such a search, the first-order algorithms performed poorly. To bypass the calibration of the gain coefficients, quantum natural algorithms can be applied while achieving a performance close to the calibrated CSPSA. In particular, the scalar quantum natural CSPSA algorithm also reduces the classical computational cost.

8.2 Quantum Control

Quantum control theory lays a firm theoretical foundation for developing a series of systematic methods that allow the manipulation and control of quantum systems. In particular, the search for an optimized time evolution that allows guiding the system from an initial state to a desired final state is of great interest. Quantum control theory has already achieved significant successes in physical chemistry [222], atomic and molecular physics [223], quantum optics [224], and has also contributed to understanding fundamental aspects of quantum mechanics [225]. In recent years, the development of the general principles of quantum control theory has been recognized as an essential requirement for the current and future applications of quantum technologies.

A particular problem in quantum control is the precise engineering of quantum

states, that is, whether a quantum system can be driven into a given state. This problem has practical importance since it is closely related to the universality of quantum computing and the possibility of achieving transformations at the atomic or molecular scale. An important research problem is that of finite-dimensional quantum systems, for which the controllability criteria can be expressed in terms of parameters included in the Hamiltonian of the system.

The quantum state control problem [226] consists in identifying an appropriate set of time-dependent control parameters $u_k(t)$ in such a way that its controlled change in time guides the evolution of the system from an initial state $|\psi_0\rangle$ to a predetermined objective state $|\psi_f\rangle$. The control parameters enter in the Hamiltonian as coefficients in a linear combination of operators, that is,

$$H(t) = H_0 + \frac{1}{2}\sum_k \left(u_k(t)C_k + u_k^*(t)C_k^\dagger\right), \tag{8.6}$$

where the set $\{C_k\}$ are a basis of operators, and we allow the possibility of complex control parameters. In order to obtain this set of parameters, it is necessary to solve the time-dependent Schrödinger equation. Unfortunately, solutions of the Schrödinger equation for a time-dependent Hamiltonian cannot generally be obtained analytically. However, it is possible in certain cases to use techniques developed in the area of adiabatic control [227–230].

To overcome this problem, we use the GRadient Ascent Pulse Engineering (GRAPE) method [212], originally introduced in nuclear magnetic resonance spectroscopy and proposed to design a pulse sequence that drives the evolution toward the optimum of a predefined objective function. This method allows us to compute the evolution of a time-dependent Hamiltonian through a sequence H_m of time-independent Hamiltonians. The total evolution time T is divided into a number M of time intervals $\Delta t_m = t_{m+1} - t_m$ $(m = 0, \ldots, M-1)$, which are normally of equal length so that in each interval the control parameters $u_k(t)$ are approximately

constant. In each time interval, the evolution is given by

$$U_m = e^{-i\Delta t_m H_m}, \tag{8.7}$$

where $H_m = H(t_m^*)$ with $t_m^* \in [t_m, t_{m+1}]$. A classical optimization algorithm is used to obtain the values of the control parameters that lead to the optimum of the objective function. The evolution of the system at time T is thus approximated by the sequence

$$U_{\text{GRAPE}} = \prod_{m=0}^{M-1} U_m \tag{8.8}$$

and the state of the system at time T is

$$|\tilde{\psi}_f\rangle = U_{\text{GRAPE}} \, |\psi_0\rangle \, . \tag{8.9}$$

Once a propagator has been computed for a set of control parameters, all that remains is to choose an objective function to compare the target state with the state given by the evolution for a given set of control parameter values. In our case, we use the infidelity that is given by

$$I(|\tilde{\psi}_f\rangle, |\psi_f\rangle) = 1 - |\, \langle \tilde{\psi}_f | \psi_f \rangle \, |^2, \tag{8.10}$$

which is minimized with an optimization algorithm. The original GRAPE proposal uses the descending gradient algorithm. The dimension of the search space is given by $N_p M$, where N_p is the number of parameters, and therefore can be very large.

To test the optimization methods introduced here, we turn to the quantum control of a five-qubit system, where we aim at preparing the target state $|\psi_f\rangle = |0\rangle^{\otimes 5}$ by controlling the evolution generated by the Heisenberg Hamiltonian given by

$$H_H(t) = -\frac{1}{2} \sum_{k=x,y,z} J_k(t) \sum_{\langle m,n \rangle} \sigma_m^k \sigma_n^k, \tag{8.11}$$

which depends on the three complex coupling constants $J_x(t)$, $J_y(t)$ and $J_z(t)$. These play the role of control parameters whose values are driven by the quantum control method to approach the desired target state.

After applying the GRAPE method for the evolution of the system, the final state is

$$|\tilde{\psi}_f\rangle = \prod_{i=1}^{M-1} e^{-i\Delta t_m H_H(t_m^*)} |\psi_0\rangle \,, \tag{8.12}$$

where $|\psi_0\rangle$ is an initial five-qubit state and H_H contains the control parameters.

Our performance study is based on numerical simulations where we implement GRAPE with each of the methods reviewed or proposed here. For a given optimization method, we start by choosing an initial state $|\psi_0\rangle$ from a Haar-uniform distribution on which we apply the GRAPE method with $M = 25$ and 10^3 iterations. Therefore, the dimension of the complex search space is 75, with the real search space being twice as large. The measurements required by the optimization method are simulated with an ensemble of size 2^{13}. This procedure is repeated 10^4 times to obtain estimates of relevant statistical indicators such as mean, median, standard deviation, and interquartile range, as functions of the number of iterations. The gain parameters used in the numerical simulations are shown in Tables A.3 and A.4 of Appendix A.

The results of the numerical simulations of the GRAPE method with the different optimization algorithms in the five-qubit case are depicted in Figures 8.4 and 8.5, which show the best results among the vanilla methods and the improved methods, respectively. Each figure shows the value of the mean (upper row) and median (lower row) infidelity as a function of the number of iterations together with the variance (upper row) and the interquartile range (lower row) as shaded areas.

Figure 8.4 shows the comparison between methods without using blocking and resampling (see Section 7.3), that is, the vanilla methods. Second-order methods exhibit the best mean performance, particularly 2CSPSA and scalar 2CSPSA. These are closely followed by their quantum natural counterparts. First-order methods initially offer a better convergence rate, but stagnate after a certain number of iterations. Note that this is the only case among all applications where first-order SPSA and CSPSA achieve their best performance using the static gain coefficients. In the

median, second-order methods exhibit a higher convergence rate, closely followed by the quantum natural methods. Although first-order methods require a larger number of iterations, a similar value of infidelity is reached in all cases. However, they behave differently in mean and median, in contrast to the complex second-order and quantum natural methods exhibiting consistent statistical indicators.

Figure 8.5 shows the comparison between methods when we allow the usage of blocking and resampling, that is, the improved methods. The best performance in the mean and the median is attained by the first-order methods seconded by the quantum natural methods, which exhibit a slightly slower rate of convergence with similar standard deviation and interquartile range. Second-order methods are the worst performers. These are characterized by a lower precision in mean, large standard deviation, and a slower rate of convergence, with the exception of scalar methods. Figure 8.5 also indicates that complex methods perform better than their real counterparts.

Generally, the first-order CSPSA method with resampling and blocking obtains the best result, using Equations (7.44) for post-processing, closely followed by the QN-CSPSA method.

8.3 Quantum state estimation

Born's rule endows quantum mechanics with predictive power. According to this rule, the probability p_k of obtaining a result k in an experiment described by a POVM $\{E_k\}$ when the quantum system is described by a quantum state ρ is given by the Hilbert-Schmidt inner product $p_k = \mathrm{Tr}(\rho E_k)$. Therefore, a comparison between theoretical predictions and experimental results requires an accurate characterization of the quantum state ρ and of the experiment through POVM $\{E_k\}$. This leads to the problem of estimating quantum states and processes. To do this, several quantum state estimation methods have been designed, most of them based on the

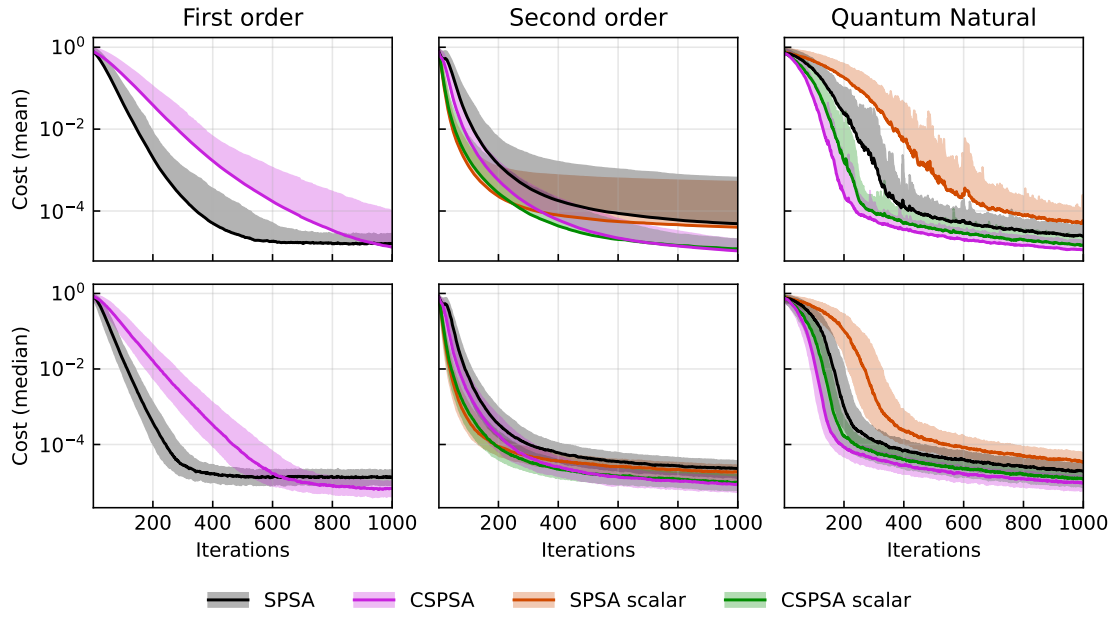


Figure 8.4: The mean (top row) and median (bottom row) of the infidelity as a function of the number of iterations obtained through the GRAPE method applied to the quantum control of a 5-qubit pure state and vanilla optimization algorithms. The shaded areas represent the variance (top row) and the interquartile range (bottom row). The values of the infidelity are obtained by simulating a measurement process with a sample size of 2^{13} and 25 iterations of GRAPE, 10^4 shots per measurement, and 10^3 iterations, which are generated through uniformly distributed initial states $|\psi_0\rangle$. The values of the gain coefficients and post-processing class can be found in Table A.3 of Appendix A.

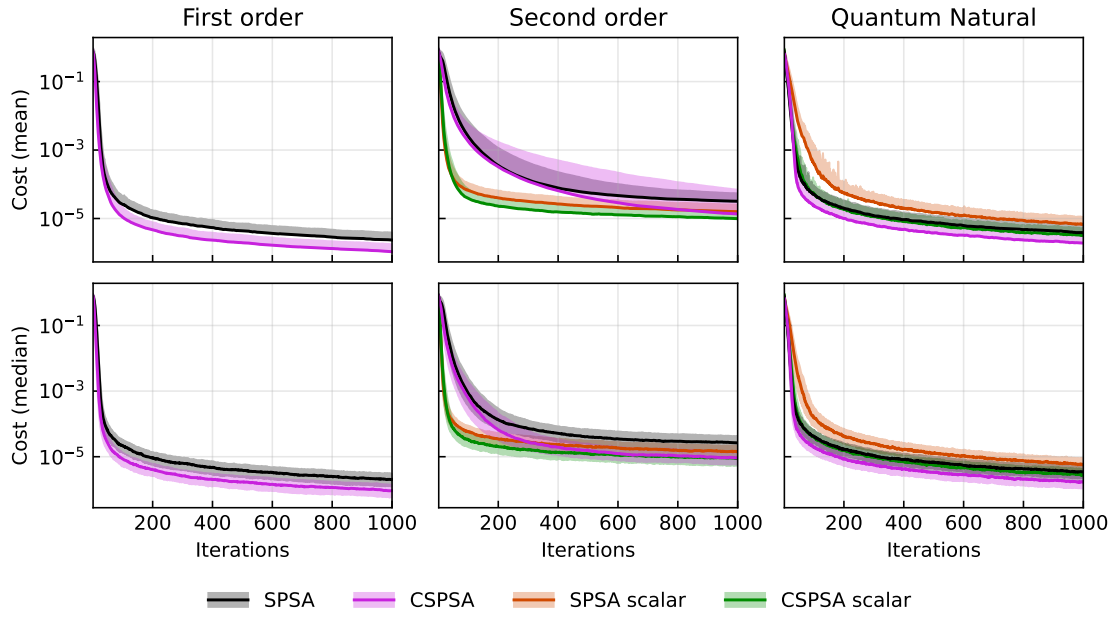


Figure 8.5: The mean (top row) and median (bottom row) of the infidelity as a function of the number of iterations obtained through the GRAPE method applied to the quantum control of a 5-qubit pure state and improved optimization algorithms. The shaded areas represent the variance (top row) and the interquartile range (bottom row). The values of the infidelity are obtained by simulating a measurement process with a sample size of 2^{13} and 25 iterations of GRAPE, 10^4 shots per measurement, and 10^3 iterations, which are generated through uniformly distributed initial states $|\psi_0\rangle$. The values of the gain coefficients, post-processing class, and the setting of resampling and blocking can be found in Table A.4 of Appendix A.

post-processing of experimental data acquired through the measurement of a fixed informationally complete POVM. Adaptive measurements have also been used to design quantum state estimation methods. Today, methods for estimating quantum states are an important tool for both quantum communications and quantum computing and have been used for the characterization of single-photon and continuous variable states [231–235], cavity fields [236], atomic ensembles [237–239], trapped ions [240, 241], optical detectors [242–244], and for quantum key distribution [245].

Recently, the estimation of finite-dimensional pure unknown states has been formulated as an optimization problem [94]. According to this, the unknown state is characterized as the minimizer of infidelity $I(|\psi\rangle, |\phi\rangle) = 1 - |\langle\psi|\phi\rangle|^2$, that is,

$$|\psi\rangle = \arg\left(\min_{|\phi\rangle\in\mathcal{H}} I(|\psi\rangle, |\phi\rangle)\right). \tag{8.13}$$

This suggests using optimization algorithms to minimize fidelity and estimate the unknown state $|\psi\rangle$, which has been called self-guided quantum tomography (SGQT). Gradient-based optimization is ruled out since it is not known how to measure the gradient of the infidelity with respect to the parameters entering the $|\phi\rangle$ state. However, infidelity can be measured by projecting the unknown state onto any basis containing the state $|\phi\rangle$. In this scenario, the optimization methods presented previously can be used to experimentally implement the infidelity minimization according to SGQT. Initially, SGQT was based on the SPSA algorithm. Subsequently, CSPSA was introduced in SGQT, obtaining an improvement in the rate of convergence and a lower dispersion in the sample of estimates. More recently, CSPSA was combined with maximum likelihood estimation to achieve precision close to the lower limit of Gill-Massar, which is the best achievable estimation accuracy for pure states [246]. Estimating pure states through SPSA and CSPSA has already been experimentally demonstrated [95].

We use pure state estimation through SGQT to test the performance of the optimization methods proposed. After selecting a particular optimization method,

we generate an unknown 6-qubit pure state and an initial guess state from a Haar-uniform distribution. The optimization method is iterated 5×10^3 times and the fidelity values are obtained by simulating a measurement with binomial distribution on an ensemble size $N = 2 \times 10^4$. This procedure is repeated 10^2 times to generate estimates of relevant statistic indicators. The gain parameters used in the numerical simulations of each method are shown in Appendix A.

The results of the numerical simulations of SGQT with the different optimization algorithms are depicted in Figures 8.6 and 8.7 that show the best results among the vanilla and improved methods, respectively. Each figure shows the value of the mean (upper row) and median (lower row) infidelity as a function of the number of iterations together with the variance (upper row) and the interquartile range (lower row) as shaded areas. In every figure, the first column contains the results of SPSA and CSPSA. The second column contains the results obtained by the second-order algorithms, that is, 2SPSA, 2CSPSA, scalar 2SPSA, and scalar 2CSPSA. The third column contains the results obtained by the algorithms based on the quantum natural method, that is, QN-SPSA, QN-CSPSA, scalar QN-SPSA, and scalar QN-CSPSA.

In Figures 8.6 and 8.7 mean and median values for each algorithm are very close. Furthermore, the variance and interquartile range are very narrow, which shows the absence of outliers in the generated samples. Typically, all optimization algorithms are characterized by a sharp decrease in infidelity followed by an approximately linear asymptotic regime.

Figure 8.6 shows the comparison between each method without using the blocking and resampling improvements in Equations 7.44, as these methods largely increase the number of resources. The first-order methods offer the best performance, with an improvement of about an order of magnitude over the other methods. In contrast, the second-order methods perform slightly better than their QN counterpart. The scalar approximation shows no improvements for the second-order and quantum natural methods. The complex methods show better convergence than their real

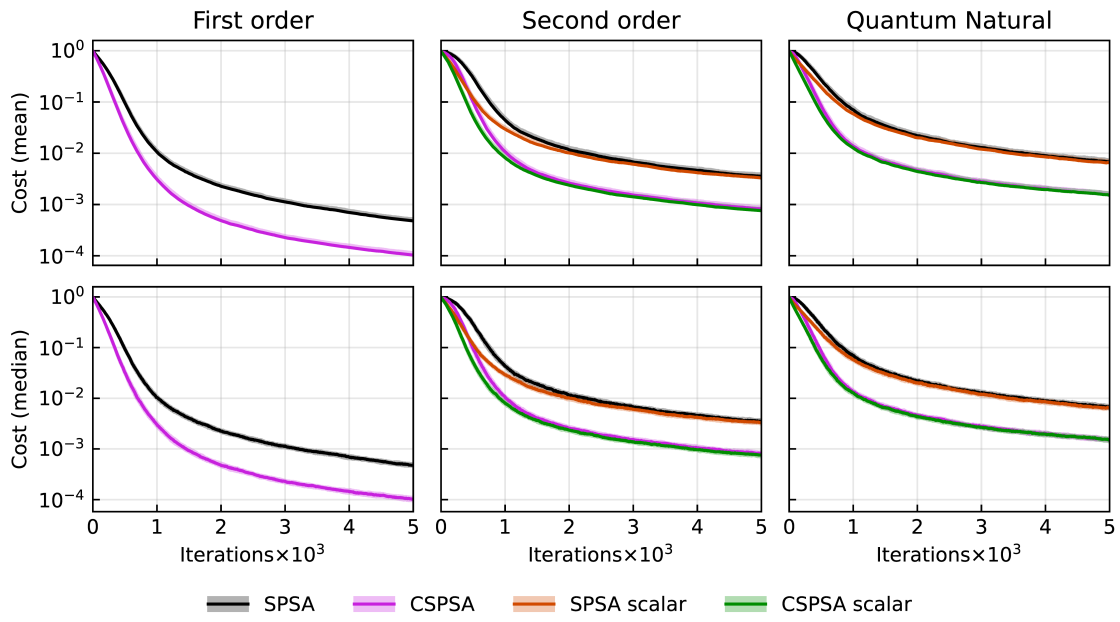


Figure 8.6: The mean (top row) and median (bottom row) of the infidelity as a function of the number of iterations obtained by using SGQT to estimate six-qubit states and vanilla optimization algorithms. Shaded areas represent variance (top row) and interquartile range (bottom row). Statistical indicators are calculated from a sample of 10^2 Haar-uniform distributed pairs of unknown and initial guess states. Measurements of the infidelity are simulated with a binomial distribution with $N = 2 \times 10^4$ shots. The values of the gain coefficients and post-processing class can be found in Table A.5 of Appendix A.

counterparts by about an order of magnitude.

Figure 8.7 shows the comparison between methods when we allow the use of blocking and resampling. For first-order algorithms, gradient blocking and resampling show no improvement, while second-order and QN methods improve when the Hessian approximation is averaged 5 times per iteration. This improvement decreases the performance difference between the first-order and the other methods. Second-order methods still perform slightly better than QN methods. We also note that the improvement obtained by resampling is smaller for the scalar approximation.

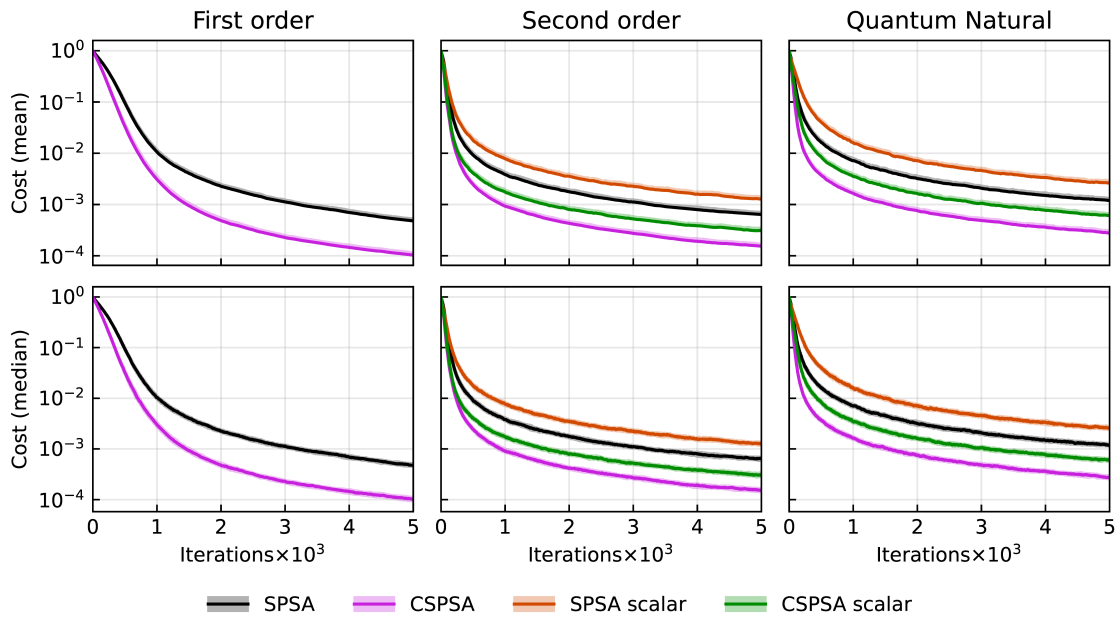


Figure 8.7: The mean (top row) and median (bottom row) of the infidelity as a function of the number of iterations obtained by using SGQT to estimate six-qubit states and improved optimization algorithms. Shaded areas represent variance (top row) and interquartile range (bottom row). Statistical indicators are calculated from a sample of 10^2 Haar-uniform distributed pairs of unknown and initial guess states. Measurements of the infidelity are simulated with a binomial distribution with $N = 2 \times 10^4$ shots. The values of the gain coefficients, post-processing class, and the setting of resampling and blocking can be found in Table A.6 of Appendix A.

In our simulations, blocking does not improve our results when considering the Hessian post-processing Equations (7.44). On the other hand, when considering the post-processing Equations (7.4), the blocking show great improvement which matches our median results, but with worse mean performance and with wider data variability (See Appendix A).

From Figures 8.6 and 8.7 we conclude that the first-order methods show better mean and median performance in the estimation of pure 6-qubit states via SGQT, even without considering gradient resampling. In this scenario, second-order methods

are not expected to work properly since the fidelity Hessian vanishes for pure states. This issue could lower both precision and convergence speed. However, the Hessian post-processing allows us to mitigate this problem by perturbing the Hessian matrix with a weighted identity matrix. In this way, the best result achieved by the second-order methods uses the post-processing Equations (7.44). Quantum natural-based methods show similar behavior, albeit with slightly slower convergence.

First-order methods perform the best even without considering gradient resampling. In contrast, second-order and quantum natural methods need resampling improvement to stay competitive but require a much higher number of resources.

Chapter 9

Conclusions

We have exhaustively compared different stochastic optimization methods applied to real-valued functions of complex variables. We started by reviewing the theory of the SPSA algorithm and two of its variants: 2SPSA and QN-SPSA. These three methods use a simultaneous perturbation stochastic approximation of the gradient of the objective function to optimize it. SPSA is a first-order algorithm, while 2SPSA is a second-order algorithm. QN-SPSA is a quantum natural algorithm: a first-order algorithm that uses the Fubini-Study metric as a preconditioner. We also reviewed the CSPSA algorithm, which optimizes real functions of complex variables without resorting to the real and imaginary parts of complex variables. This is a more natural approach in quantum mechanics, where most functions have complex arguments. Using CSPSA as starting point, we proposed two new optimization methods: 2CSPSA and QN-CSPSA, which are the complex field formulations of their real counterparts.

All the optimization methods presented here share the property that the number of evaluations (or measurements) of the objective function does not depend on the dimension of the optimization problem. This is an important advantage when the number of parameters on which the objective function depends is large. The number of objective function evaluations is constant at each iteration but different for each

method. SPSA and CSPSA use 2 evaluations of the objective function per iteration. 2SPSA and 2CSPSA use 4 evaluations of the objective function since they are second-order methods. Finally, QN-SPSA and QN-CSPSA use 2 evaluations of the objective function plus the calculation of an approximation of a metric. If the metric is the Fubiny-Study metric tensor, then the approximation is calculated by evaluating the fidelity with respect to 4 different pure states.

To assess the performance of the optimization methods, we have compared them in three important applications in quantum computing: variational quantum eigensolver applied to the Heisenberg Hamiltonian of a 10-qubit ring, quantum control applied to a 5-qubit pure quantum state, and quantum state estimation to reconstruct a 6-qubit pure quantum state. These three applications have different objective functions that need to be measured in a quantum device and iteratively optimized to obtain a solution. In particular, we have compared the convergence rate as a function of the number of iterations. To do this, we have considered vanilla and improved algorithms versions.

Our simulations show several interesting results. The best performance is systematically achieved by the first-order CSPSA algorithm. In the case of the variational quantum eigensolver, improved first-order CSPSA and SPSA algorithms provide the best performance, exhibiting identical mean and median and similar standard deviation and interquartile range. In quantum control, improved CSPSA achieves better convergence in mean and median than all other algorithms, exhibiting a narrow standard deviation and interquartile range. This is also the case for state estimation, although in this case, the vanilla version of the CSPSA algorithm is almost indistinguishable from its improved version.

The second-best overall performance has mixed results. In the variational quantum eigensolver, the improved second-order and improved quantum natural algorithms lead to an almost indistinguishable performance, while in quantum control the improved quantum natural algorithms, particularly improved QN-CSPSA, are

clearly second best. In this case, scalar second-order algorithms perform, in mean and median, similarly to quantum natural algorithms. In contrast, their non-scalar counterparts show much lower mean performance, indicating the presence of a large number of outliers. In the case of quantum state estimation, improved second-order algorithms provide better performance than their quantum natural counterparts. In particular, second-order CSPSA achieves the second-best performance.

Generally, vanilla second-order algorithms lead to lower performance than vanilla quantum natural algorithms. This is mitigated by blocking and resampling in the improved versions of second-order algorithms, which offer performance close to that of improved quantum natural algorithms. Furthermore, complex algorithms perform better than their real counterparts, although the difference may be statistically insignificant in certain cases.

During optimization of a function, it may be possible that no information about the Hessian matrix is available a priori, either because of its high complexity or because it cannot be easily obtained analytically or numerically. For such cases, it would be desirable that second-order methods, which are based on the Hessian matrix approximation, would still be useful in the event that the Hessian matrix exhibits singularities. This is the case of quantum state estimation, where the Hessian vanishes identically. Nevertheless, second-order methods display a performance similar to that of first-order methods. Hessian post-processing Equations (7.44) ensures that the preconditioning matrix is proportional to the identity for a vanishing Hessian matrix. This leads to second-order methods working like first-order methods, albeit possibly with suboptimal gain coefficients.

The stochastic optimization methods studied here are defined through a set of gain parameters whose values specify the gain coefficients. These in turn control the step size and magnitude of the approximation of the gradient. In this way, the gain parameters are hyperparameters that allow us to control the algorithms' convergence rate. In principle, it is conceivable to find gain parameters that lead to

the best convergence rate. However, this is an expensive optimization problem whose solution might even depend on the optimizer of the objective function. Therefore, it is usual to resort to gain parameters that have proven to be good enough in practice. We have resorted to the standard gain parameters, which lead to a fast convergence in the regime of a small number of iterations, and to the asymptotic gain parameters, which lead to a fast convergence in the regime of a large number of iterations. Note that a change in the gain parameters affects not only the mean and median convergence but also the variance and interquartile range. We have also performed our simulations considering static gain coefficients, which only led to a significant improvement in the case of vanilla first-order methods applied to quantum control.

From numerical simulations with fewer qubits, we observed that the performance difference between quantum natural and first-order algorithms tends to narrow as the number of qubits increases. For the simulations reported here, the performance difference among these algorithms is small. This may indicate that quantum natural methods may outperform first-order methods for a larger number of qubits. However, this advantage of quantum natural methods is obtained by increasing the number of measurements and the classical computational cost. In this scene, the scalar quantum natural methods proposed here might be a good alternative since, according to our results, they offer comparable performance at a reduced classical cost.

According to the applications considered here, vanilla first-order algorithms are efficient and reliable options for the most general case. If higher accuracy is needed, improved first-order algorithms are the straight choice. First-order methods may require careful calibration of the gain parameters, in which case the quantum natural algorithms are a suitable alternative. In addition, quantum natural algorithms show promising results for many qubits, while second-order algorithms do not exhibit a comparative advantage.

In our study of first- and second-order algorithms, we have considered a single

source of noise, namely the statistical character of quantum measurements. It is possible to consider other error sources, such as those affecting NISQ processors. However, first-order algorithms, real or complex, have convergence proofs that allow certain types of errors that affect the evaluation of the target function. Thereby, it is expected that these algorithms will converge even in the presence of moderate noise, albeit with an increased number of iterations. The scenario in the case of the preconditioned algorithms is less clear because of the inversion of the approximated Hessian matrix. Therefore, a natural extension of this work would be to consider realistic noise sources and their impact on the convergence rate. Also, we have considered the performance as a function of the number of iterations. It is possible, however, to consider other valuable resources such as the number of measurements, evaluations, and circuits. These should also be considered in further studies of the real performance of optimization algorithms.

Appendix A

Best settings

In this Chapter, we provide the value of the statistical indicators; median, interquartile range (IQR), mean, and standard deviation (STD), obtained through numerical simulations for the best configuration of each optimization method in each application. These values were used to determine the algorithm with the best performance after 700, 1000, and 5000 iterations for the variational quantum eigensolver, quantum control of quantum states, and self-guided quantum tomography, respectively. We indicate the gain coefficients and equations used for post-processing for each vanilla method. In the case of improved methods, we also indicate the amount of resampling and the use of blocking.

Method	Gains	Post-processing	Median	IQR	Mean	STD
SPSA	Standard	-	−6.48	1.11	−6.46	0.56
CSPSA	Standard	-	−5.93	1.12	−6.44	0.57
2SPSA	Standard	Eqs. (7.44)	−5.76	1.17	−5.61	0.85
2CSPSA	Standard	Eqs. (7.44)	−4.64	1.96	−4.82	1.19
scalar 2SPSA	Standard	Eqs. (7.44)	−5.84	1.08	−5.84	0.80
scalar 2CSPSA	Standard	Eqs. (7.44)	−5.17	1.49	−5.01	1.08
QN-SPSA	Asymptotic	Eqs. (7.44)	−5.86	0.85	−6.13	0.48
QN-CSPSA	Standard	Eqs. (7.44)	−5.92	1.10	−6.33	0.53
scalar QN-SPSA	Asymptotic	Eqs. (7.44)	−5.90	1.07	−6.34	0.53
scalar QN-CSPSA	Standard	Eqs. (7.44)	−5.93	1.10	−6.39	0.55

Table A.1: Best configuration and statistical indicators for each vanilla method applied to variational quantum eigensolver.

Method	Gains	Post-processing	Resampling	Blocking	Median	IQR	Mean	STD
SPSA	Standard	-	5	No	−7.00	1.12	−6.58	0.56
CSPSA	Standard	-	5	No	−7.00	1.12	−6.51	0.57
2SPSA	Standard	Eqs. (7.4)	2	Yes	−6.94	1.11	−6.51	0.56
2CSPSA	Standard	Eqs. (7.32)	5	Yes	−6.98	1.13	−6.52	0.56
scalar 2SPSA	Standard	Eqs. (7.44)	5	Yes	−6.97	1.09	−6.55	0.30
scalar 2CSPSA	Standard	Eqs. (7.32)	5	Yes	−6.78	1.15	−6.47	0.58
QN-SPSA	Asymptotic	Eqs. (7.44)	2	Yes	−7.00	1.12	−6.50	0.56
QN-CSPSA	Asymptotic	Eqs. (7.32)	5	Yes	−6.98	1.11	−6.57	0.54
scalar QN-SPSA	Asymptotic	Eqs. (7.44)	5	Yes	−7.00	1.25	−6.50	0.57
scalar QN-CSPSA	Asymptotic	Eqs. (7.32)	5	Yes	−6.88	1.13	−6.51	0.57

Table A.2: Best configuration and statistical indicators for each method with improvements applied to variational quantum eigensolver.

Method	Gains	Post-processing	Median	IQR	Mean	STD
SPSA	Static	-	1.35×10^{-5}	1.21×10^{-5}	1.60×10^{-5}	1.11×10^{-5}
CSPSA	Static	-	6.84×10^{-6}	6.01×10^{-6}	1.31×10^{-5}	8.23×10^{-5}
2SPSA	Standard	Eqs. (7.44)	2.31×10^{-5}	2.34×10^{-5}	4.90×10^{-5}	6.02×10^{-4}
2CSPSA	Standard	Eqs. (7.44)	8.79×10^{-6}	8.12×10^{-6}	1.07×10^{-5}	9.76×10^{-6}
scalar 2SPSA	Standard	Eqs. (7.44)	1.89×10^{-5}	1.71×10^{-5}	4.03×10^{-5}	4.75×10^{-4}
scalar 2CSPSA	Standard	Eqs. (7.44)	9.73×10^{-6}	8.93×10^{-6}	1.18×10^{-5}	8.32×10^{-6}
QN-SPSA	Asymptotic	Eqs. (7.44)	1.93×10^{-5}	1.88×10^{-5}	2.36×10^{-5}	1.82×10^{-5}
QN-CSPSA	Asymptotic	Eqs. (7.44)	9.44×10^{-6}	8.49×10^{-6}	1.12×10^{-5}	7.50×10^{-6}
scalar QN-SPSA	Asymptotic	Eqs. (7.44)	3.44×10^{-5}	3.73×10^{-5}	5.68×10^{-5}	1.22×10^{-4}
scalar QN-CSPSA	Asymptotic	Eqs. (7.44)	1.24×10^{-5}	1.11×10^{-5}	1.47×10^{-5}	1.04×10^{-5}

Table A.3: Best configuration and statistical indicators for each vanilla method applied to quantum control of quantum states.

Method	Gains	Post-processing	Resampling	Blocking	Median	IQR	Mean	STD
SPSA	Asymptotic	-	5	No	2.02×10^{-6}	1.75×10^{-6}	2.35×10^{-6}	1.49×10^{-6}
CSPSA	Asymptotic	-	5	No	9.24×10^{-7}	8.16×10^{-7}	1.07×10^{-6}	6.95×10^{-7}
2SPSA	Standard	Eqs. (7.44)	2	No	2.64×10^{-5}	2.64×10^{-5}	3.19×10^{-5}	2.24×10^{-5}
2CSPSA	Standard	Eqs. (7.44)	2	No	9.15×10^{-6}	8.45×10^{-6}	1.36×10^{-5}	5.57×10^{-5}
scalar 2SPSA	Standard	Eqs. (7.44)	5	No	1.43×10^{-5}	1.28×10^{-5}	1.58×10^{-5}	9.12×10^{-6}
scalar 2CSPSA	Standard	Eqs. (7.44)	5	No	8.84×10^{-6}	7.68×10^{-6}	9.85×10^{-6}	5.84×10^{-6}
QN-SPSA	Asymptotic	Eqs. (7.44)	5	Yes	3.46×10^{-6}	2.61×10^{-6}	3.85×10^{-6}	2.22×10^{-6}
QN-CSPSA	Asymptotic	Eqs. (7.44)	5	No	1.69×10^{-6}	1.43×10^{-6}	1.92×10^{-6}	1.15×10^{-6}
scalar QN-SPSA	Asymptotic	Eqs. (7.44)	5	Yes	5.92×10^{-6}	5.01×10^{-6}	6.85×10^{-6}	4.44×10^{-6}
scalar QN-CSPSA	Asymptotic	Eqs. (7.44)	5	Yes	2.80×10^{-6}	2.24×10^{-6}	3.20×10^{-6}	1.95×10^{-6}

Table A.4: Best configuration and statistical indicators for each method with improvements applied to quantum control of quantum states.

Method	Gains	Post-processing	Median	IQR	Mean	STD
SPSA	Asymptotic	-	4.76×10^{-4}	6.48×10^{-5}	4.79×10^{-4}	5.50×10^{-5}
CSPSA	Asymptotic	-	1.01×10^{-4}	2.00×10^{-5}	1.03×10^{-4}	1.40×10^{-5}
2SPSA	Standard	Eqs. (7.44)	6.17×10^{-4}	3.52×10^{-3}	3.55×10^{-3}	4.75×10^{-4}
2CSPSA	Standard	Eqs. (7.44)	1.43×10^{-4}	8.15×10^{-4}	8.15×10^{-4}	1.02×10^{-4}
scalar 2SPSA	Standard	Eqs. (7.44)	5.22×10^{-4}	3.26×10^{-3}	3.29×10^{-3}	3.70×10^{-4}
scalar 2CSPSA	Standard	Eqs. (7.44)	1.38×10^{-4}	7.60×10^{-4}	7.58×10^{-4}	9.87×10^{-5}
QN-SPSA	Standard	Eqs. (7.44)	6.68×10^{-3}	4.04×10^{-3}	6.72×10^{-3}	8.42×10^{-4}
QN-CSPSA	Standard	Eqs. (7.44)	1.52×10^{-3}	9.43×10^{-4}	1.53×10^{-3}	1.93×10^{-4}
scalar QN-SPSA	Standard	Eqs. (7.44)	6.58×10^{-3}	4.00×10^{-3}	6.55×10^{-3}	8.29×10^{-4}
scalar QN-CSPSA	Standard	Eqs. (7.44)	1.49×10^{-3}	9.69×10^{-4}	1.51×10^{-3}	1.94×10^{-4}

Table A.5: Best configuration and statistical indicators for each vanilla method applied to self-guided quantum tomography.

Method	Gains	Post-processing	Resampling	Blocking	Median	IQR	Mean	STD
SPSA	Asymptotic	-	1	No	4.76×10^{-4}	6.48×10^{-5}	4.79×10^{-4}	5.50×10^{-5}
CSPSA	Asymptotic	-	1	No	1.01×10^{-4}	2.00×10^{-5}	1.03×10^{-4}	1.40×10^{-5}
2SPSA	Standard	Eqs. (7.44)	5	No	6.45×10^{-4}	1.01×10^{-4}	6.39×10^{-4}	7.26×10^{-5}
2CSPSA	Standard	Eqs. (7.44)	5	No	1.52×10^{-4}	3.12×10^{-5}	1.54×10^{-4}	1.90×10^{-5}
scalar 2SPSA	Standard	Eqs. (7.44)	5	No	1.27×10^{-3}	2.21×10^{-4}	1.27×10^{-3}	1.67×10^{-4}
scalar 2CSPSA	Standard	Eqs. (7.44)	5	No	3.05×10^{-4}	5.05×10^{-5}	3.06×10^{-4}	4.04×10^{-5}
QN-SPSA	Standard	Eqs. (7.44)	5	No	1.17×10^{-3}	2.06×10^{-4}	1.19×10^{-3}	1.45×10^{-4}
QN-CSPSA	Standard	Eqs. (7.44)	5	No	2.74×10^{-4}	4.53×10^{-5}	2.76×10^{-4}	3.33×10^{-5}
scalar QN-SPSA	Standard	Eqs. (7.44)	5	No	2.57×10^{-3}	4.68×10^{-4}	2.55×10^{-3}	3.45×10^{-4}
scalar QN-CSPSA	Standard	Eqs. (7.44)	5	No	6.06×10^{-4}	1.15×10^{-4}	6.09×10^{-4}	7.60×10^{-5}

Table A.6: Best configuration and statistical indicators for each method with improvements applied to self-guided quantum tomography.

Part III

Solving PDEs with Matrix Product States

This part of the thesis corresponds to the work developed during my doctoral stay at the Institute of Fundamental Physics (IFF-CSIC) in Madrid, Spain. The work was done in collaboration, where we worked equally with Paula García-Molina, under the guidance of Luca Tagliacozzo and Juan José García-Ripoll. The research is available on arXiv [247], and is currently under review in the Journal of Computational Physics.

Chapter 10

Motivation

"The aim of science is not to open the door to infinite wisdom, but to set a limit to infinite error."

Bertolt Brecht

Solving time-dependent partial differential equations (PDEs) is fundamental in many areas of science and engineering. In quantum physics, this task is particularly challenging due to exponential computational costs [35], which stem from both the high dimensionality of quantum systems and the unbounded nature of certain domains, such as the expansion of a particle in a potential well [248, 249].

Traditional numerical methods struggle with large quantum systems because of this exponential scaling. Quantum computing has been proposed as a potential solution [250], taking advantage of exponential compression from amplitude encoding [120, 251, 252] and tools such as the quantum Fourier transform (QFT) [253]. However, its practical implementation remains limited by the current lack of scalable and fault-tolerant quantum hardware [254].

As a more accessible alternative, variational quantum algorithms (VQAs) have also been explored to solve PDEs. These include the variational quantum eigensolver (VQE) for static problems and algorithms for linear [255, 256] and nonlinear equa-

tions [257, 258]. However, VQAs face challenges such as noise, barren plateaus, and scalability, which limit their practical utility and raise doubts about their potential usefulness in the near term. It is still uncertain whether they can solve problems that cannot be simulated efficiently by classical computers [117].

Although quantum algorithms avoid some limitations of classical solvers, they are currently constrained by hardware limitations and optimization difficulties. Moreover, it is not clear whether quantum computers are even necessary for problems involving bandwidth-limited functions with limited entanglement [120, 121].

A current challenge is to develop quantum-inspired algorithms that mimic quantum data compression while remaining entirely classical. Quantum-inspired methods have been applied to quantum dynamics [259, 260] and time-dependent PDEs [261], even reaching specific domains such as kinetic plasma simulations [262].

The development of such algorithms for tackling PDE evolution requires three components: (i) an efficient representation of the solution, (ii) a compatible encoding of the differential operator, and (iii) a time evolution method.

We adopt matrix product states (MPS), known as quantized tensor trains (QTT) in applied mathematics [263], to represent bandwidth-limited functions [121].

A persistent difficulty is the discretization of differential operators, which often relies on accurate but costly spectral methods like Fourier techniques with Trotter expansions [264, 265] or Chebyshev propagators [266], Finite difference schemes offer lower computational cost but are typically less flexible and less accurate.

We propose a novel differential operator encoding by incorporating Hermite Distributed Approximating Functionals (HDAF) [267–272] into the MPS framework. This approach yields accurate matrix product operators (MPOs) with low bond dimension. Based on these operators, we introduce three families of time evolution algorithms: explicit and implicit Runge-Kutta methods, Arnoldi iteration, and split-step schemes. These methods are compared to each other, against quantum-inspired finite difference methods [120, 260], and against spectral FFT split-step methods

using standard vector representations [273].

To evaluate these techniques, we consider the expansion of a particle in a potential well, a computationally intensive scenario relevant to optomechanics [248, 249, 274]. Traditional methods are constrained by the rapidly expanding spatial domain, but the MPS/QTT representation may overcome this limitation by enabling exponential compression. However, wavefunction chirping due to acceleration can increase entanglement in the MPS representation, requiring larger bond dimensions, and potentially negating these advantages. In cases where analytical solutions are available [275], this setting offers a valuable testbed to assess numerical methods.

Part III of the thesis is structured as follows. Chapter 11 defines the benchmark problems, Chapter 12 presents the numerical methods, and Chapter 13 the corresponding results. In particular, Sections 12.1 and 12.2 detail the MPS finite-precision algebra framework and introduce HDAFs for constructing MPOs, including the metaheuristics to fine-tune the approximations. Section 12.3 reviews quantum time evolution and presents the proposed numerical schemes. Section 13.1 compares the performance of these methods in a one-step evolution test, and Section 13.2 applies the best techniques to simulate full harmonic and anharmonic expansion problems. Finally, Chapter 14 summarizes the key findings of the study.

Chapter 11

Benchmark problem: Particle expansion

"The river is everywhere at once."

Hermann Hesse

A quantum quench refers to the sudden change in the Hamiltonian of a system that drives it out of equilibrium [276, 277]. In this section, we introduce the particle expansion problem which arises from the abrupt relaxation of a confining harmonic potential.

This process exhibits several features relevant to our study. First, it is of broad interest in fields such as many-body physics [278–281] and quantum optomechanics [249]. Second, it is computationally demanding for both traditional and MPS-based solvers. Standard vector-based methods become inefficient because of the need to represent large spatial domains during expansion. This makes the problem a suitable test case for MPS simulations. However, wavefunction acceleration leads to chirping, increasing the bond dimension required by MPS methods. Third, when the system quenches from one harmonic potential to another, there is an analytical solution [275], providing a benchmark to assess both speed and accuracy.

The particle's time evolution is governed by the Schrödinger equation,

$$i\partial_t \psi(x,t) = \left(-\frac{\hbar^2}{2m}\partial_x^2 + V(x,t)\right)\psi(x,t). \tag{11.1}$$

The choice of potential $V(x,t)$ determines the dynamics. Our benchmark problem consists of a sudden quench of a harmonic potential from frequency ω_0 to ω_H at $t = 0$,

$$V(x,t) = \begin{cases} \frac{1}{2}\omega_0^2 x^2, & t \leq 0, \\ \frac{1}{2}\omega_H^2 x^2, & t > 0. \end{cases} \tag{11.2}$$

For the rest of the thesis, we will adopt the natural units of the pre-quench Harmonic oscillator. That is, we express mass, time and length in units of m, ω_0^{-1} and $\sqrt{\hbar/m\omega_0}$, respectively. This effectively sets $\hbar = m = \omega_0 = 1$. However, we will retain ω_0 in the equations for clarity.

Assuming the wavefunction is initially in the ground state of the pre-quench Hamiltonian,

$$\psi(x, t=0) = \left(\frac{\omega_0}{\pi}\right)^{1/4} \exp\left(-\frac{1}{2}\omega_0 x^2\right), \tag{11.3}$$

its evolution is given by the exact solution:

$$\psi(x,t) = \left(\frac{\omega(t)}{\pi}\right)^{1/4} \exp\left(-\left[\frac{\omega(t)}{2} + i\beta(t)\right]x^2\right), \tag{11.4}$$

$$\omega(t) = \omega_H \left(\frac{\omega_H}{\omega_0}\cos^2(\omega_H t) + \frac{\omega_0}{\omega_H}\sin^2(\omega_H t)\right)^{-1}, \tag{11.5}$$

$$\beta(t) = \frac{\omega(t)}{4}\left(\frac{\omega_H}{\omega_0} - \frac{\omega_0}{\omega_H}\right)\sin(2\omega_H t). \tag{11.6}$$

This solution is a time-dependent complex Gaussian. Its width, $\sigma(t) = 1/\sqrt{\omega(t)}$, oscillates with period π/ω_H. For $\omega_H < \omega_0$, the system expands during the first half-period. The minimum and maximum widths are $\sigma_{\min} = 1/\sqrt{\omega_0}$ at $t = 0$ and $\sigma_{\max} = \sqrt{\omega_0}/\omega_H$ at $t = 0.5\pi/\omega_H$.

The expansion factor is $\sigma_{\max}/\sigma_{\min} = \omega_0/\omega_H$, which means that the frequency ratio controls the spatial amplification. Higher expansion ratios increase the computational cost of accurate simulations.

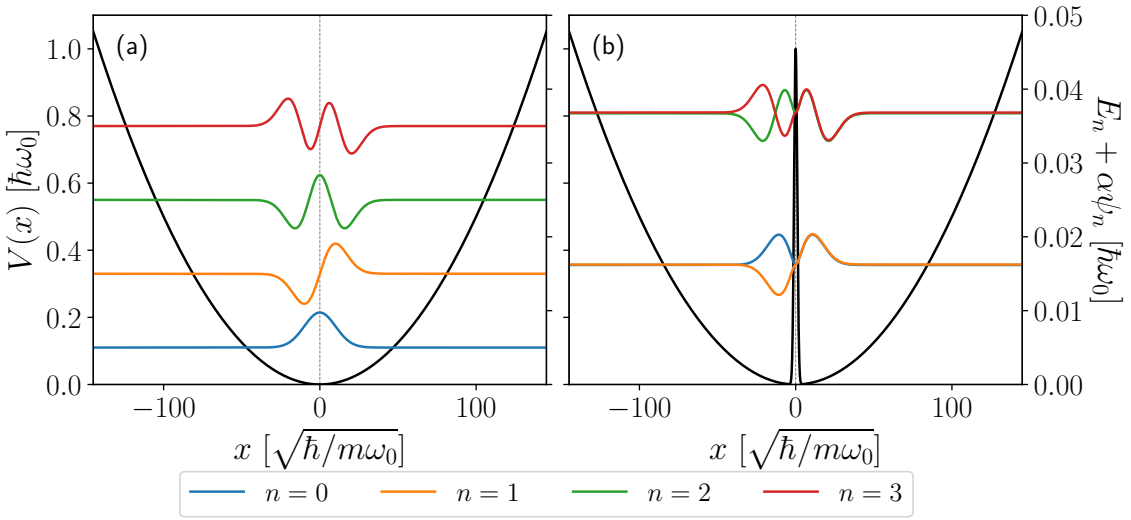


Figure 11.1: First 4 eigenstates of the Hamiltonian, for $t > 0$, using (a) the harmonic potential (11.2) and (b) the double-well potential (11.7) with $\omega_H/\omega_0 = 0.01$ and $u = \sigma = 1$. For visualization, the wavefunctions have been displaced and scaled to $E_n + \alpha\psi_n(x)$, where $\psi_n(x)$ is the n-th eigenstate, E_n its corresponding energy, and $\alpha = 0.02$.

A related and more complex scenario involves a transition to a wider trap with an anharmonic term, of interest in experimental optomechanics [274, 282–284]. Analytical solutions are generally unavailable in such cases, while numerical challenges remain. One example is the double-well potential,

$$V(x,t) = \begin{cases} \frac{1}{2}\omega_0^2 x^2, & t \le 0, \\ \frac{1}{2}\omega_H^2 x^2 + u\exp(-x^2/2\sigma^2), & t > 0. \end{cases} \tag{11.7}$$

Figure 11.1 shows the harmonic and double-well potentials, and the first 4 eigenstates of the corresponding Hamiltonians. In the harmonic case (a), the energy spectrum is uniform and the eigenstates are either symmetric (n even) or antisymmetric (n odd). In the double-well (b), the parity of the eigenstates is preserved, but the energies have been displaced, becoming anharmonic. The antisymmetric eigenfunctions differ slightly from the harmonic case, only around $x = 0$, while the first

symmetric eigenfunctions are notably suppressed in the center due to the repulsive Gaussian barrier.

The eigenstates in the double-well potential form near-degenerate symmetric/antisymmetric pairs for $n = 2m$ and $n = 2m + 1$, respectively, with the pair labeled by m. The interference of the paired wavefunctions produce distributions that are localized in either the left or right well, while the near-degeneracy of their energies indicates a slow tunneling effect.

In our case, the initial state (11.3) is symmetric. Therefore, the evolution must also remain symmetric, without tunneling. The wavefunction will be composed of symmetric eigenfunctions of the double-well Hamilonian at all times. Moreover, the anharmonicity in the energy levels leads to dephasing over time. In contrast to the harmonic case, no exact revival of the initial wavefunction is expected.

For small values of the barrier amplitude u, the dynamics are dominated by harmonic expansion, with the central Gaussian bump acting only as a perturbation. In this regime, the analytic solution (11.4) serves as a reference for the expansion. However, the Gaussian perturbation introduces a symmetric separation in the particle's probability density, deviating from a single Gaussian to a two-peaked structure. We include this case as a benchmark to evaluate the performance of our proposed methods under realistic and experimentally relevant conditions.

Chapter 12

Numerical Methods

"The bamboo that bends is stronger than the oak that resists."

Japanese proverb

12.1 Quantum-Inspired Numerical Analysis

The MPS/QTT format may provide exponential memory compression and mitigate the curse of dimensionality [120, 259]. In particular, functions with rapidly decaying Fourier coefficients have an efficient representation with limited entanglement [121].

MPS and tensor train methods have been successfully applied across various numerical analysis tasks, including high-dimensional nonlinear PDEs [285], Hamilton-Jacobi-Bellman equations [286–290], the Schrödinger equation [291], and stochastic equations [292–294]. Simulations of quantum dynamics usually rely on spectral methods such as Fourier transforms with Trotter decompositions [264, 265] or Chebyshev time propagators [266]. Other strategies include implicit time integration via alternating least squares (ALS) solvers and global space-time formulations for parabolic PDEs [147]. Quantum-inspired methods have been successfully applied to a range of problems, including the Schrödinger equation [259], fluid turbulence [138, 295],

Hamiltonian PDEs [251], and the Vlasov-Poisson system [262].

The MPS/MPO-based representations, along with the operations described in this chapter, define a finite-precision algebra [260] that mirrors conventional matrix-vector arithmetic. This algebra forms the foundation of the quantum-inspired algorithms developed in this chapter.

12.1.1 Function and Operator Encoding

The amplitude encoding of functions [120, 251, 252] discretizes a function $f(x)$ over the interval $[a, b[$ to an n-qubit quantum register, resulting in a normalized quantum state:

$$|f\rangle = \frac{1}{C} \sum_{\alpha=0}^{2^n-1} f(x_\alpha) |\alpha\rangle, \quad x_\alpha = a + \alpha\Delta x, \tag{12.1}$$

where C is a normalization constant. This encoding corresponds to a many-body wavefunction that, for bandwidth-limited functions, can be efficiently represented as an MPS [120].

12.1.1.1 Analytic examples

Some functions, such as ax, bx^2 and $\exp(cx)$, and some operators admit an exact representation as MPS/MPO. In this Chapter, we make use of several MPS/MPO with analytical construction. For instance, we require an MPS to represent our numerical grid: the position MPS. To construct Hamiltonians and functions thereof, we also used the corresponding position MPO, together with differential operators built on the basis of shift operators. In this section, we will show how these exact MPS/MPO are defined in terms of their site tensors. More details on these constructions can be found in García-Ripoll [120].

Position MPS The goal is to represent a vector x with entries

$$x_i = a + i\Delta x,$$

where i is an integer index. By expressing i in its binary form,

$$i = [i_1 i_2 \dots i_n]_2 = \sum_{k=1}^{n} 2^{n-k} i_k$$

each binary digit i_k is associated with a site tensor of the MPS. The overall MPS factorization is written as

$$x_i = L A^1(i_1) A^2(i_2) \dots A^n(i_n) R,$$

where the matrices $A^k(i_k)$ are the site tensors evaluated at the corresponding binary digits, and the boundaries L and R are a row and a column vector, respectively.

To motivate the structure of the site tensors, note that

$$\begin{bmatrix} 1 & x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & y \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & x+y \\ 0 & 1 \end{bmatrix},$$

which inspires the choice

$$A^k(i_k) = \begin{bmatrix} 1 & \Delta x 2^{n-k} i_k \\ 0 & 1 \end{bmatrix}$$

such that the product of all sites yields

$$A^1(i_1) A^2(i_2) \dots A^n(i_n) = \begin{bmatrix} 1 & \Delta x \sum_{k=1}^{n} 2^{n-k} i_k \\ 0 & 1 \end{bmatrix}.$$

Then, choosing the boundary vectors as

$$L = \begin{bmatrix} 1 & a \end{bmatrix}$$
$$R = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

the MPS factorization reduces to the vector element desired,

$$L A^1(i_1) A^2(i_2) \dots A^n(i_n) R = a + \Delta x \sum_{k=1}^{n} 2^{n-k} i_k = x_i.$$

We can absorb the boundaries on the adjacent sites,

$$LA^1 = \begin{bmatrix} 1 & a + \Delta x 2^{n-1} i_1 \end{bmatrix},$$
$$A^n R = \begin{bmatrix} \Delta x i_n \\ 1 \end{bmatrix},$$

finally prescribing the MPS sites, which are nonzero only at positions

$$A^1_{0i_10} = 1, \tag{12.2a}$$
$$A^1_{0i_11} = a + \Delta x 2^{n-1} i_1, \tag{12.2b}$$
$$A^k_{0i_k0} = 1, \tag{12.2c}$$
$$A^k_{1i_k1} = 1, \tag{12.2d}$$
$$A^k_{0i_k1} = \Delta x 2^{n-k} i_k, \tag{12.2e}$$
$$A^n_{0i_n0} = \Delta x i_n, \tag{12.2f}$$
$$A^n_{1i_n0} = 1. \tag{12.2g}$$

Notice that the position MPS has a fixed bond dimension $\chi = 2$.

Position operator The position operator can be formed by recasting the position MPS discussed above as a diagonal MPO, with the sites B^k of the MPO defined in terms of the sites A^k of the MPS as

$$B^k_{\gamma_k \alpha'_k \alpha_k \gamma_{k+1}} = A^k_{\gamma_k \alpha_k \gamma_{k+1}} \delta_{\alpha'_k, \alpha_k}.$$

Shift operators Given the encoding Equation (12.1), a shift operator acts on the basis according to

$$\hat{\Sigma}^{\pm} |\alpha\rangle = |\alpha \pm 1 \mod 2^n\rangle.$$

We will introduce the analytical deduction for $\hat{\Sigma}^{+m} = \left(\hat{\Sigma}^{+}\right)^m$. Notice that

$$\hat{\Sigma}^{-} = \left[\hat{\Sigma}^{+}\right]^{-1},$$

and that the procedures below are also valid for negative m, with $\hat{\Sigma}^{-m} = \hat{\Sigma}^{2^n - m}$.

A shift operator $\hat{\Sigma}^{+m}$ works in a similar way to classical ripple adder circuits. See Figure 12.1 for reference. Each MPO site has four indices which we will consider as two inputs and two outputs: the right bond index γ_{k+1} is a carry coming from less significant digits, the left bond index γ_k is the carry passed on to the next site, the physical input index α_k is the original digit on the MPS and the physical output index α'_k is the new digit after the contribution of the carry.

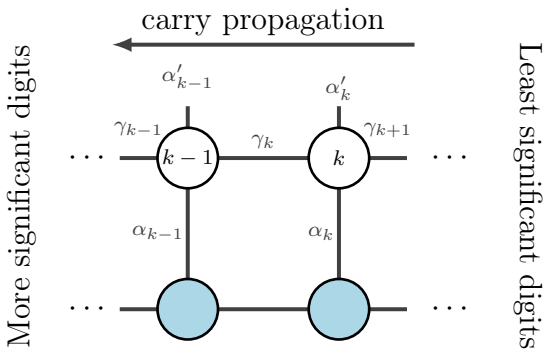


Figure 12.1: Diagram of the carry propagation on shift MPO.

The outputs depend on the value of the inputs. At site i, the combined value from the physical digit and the incoming carry is $\alpha_k + \gamma_{k+1}$. This sum must be split into the new physical digit α'_k that remains at the site and a carry γ_k that propagates to the next site on the left. Since we are working in base 2 (emulating qubits), we decompose the sum as

$$\alpha_k + \gamma_{k+1} = 2\gamma_k + \alpha'_k,$$

leading to the update rules

$$\begin{aligned} \alpha'_k &= (\alpha_k + \gamma_{k+1}) \mod 2, \\ \gamma_k &= \left\lfloor \frac{\alpha_k + \gamma_{k+1}}{2} \right\rfloor. \end{aligned}$$

Using these relations, we define the tensor elements at each site of $\hat{\Sigma}^{+m}$ by

$$A^k_{\gamma_k \alpha'_k \alpha_k \gamma_{k+1}} = \begin{cases} 1, & \text{if } \alpha'_k = (\alpha_k + \gamma_{k+1}) \bmod 2 \text{ and } \gamma_k = \left\lfloor \frac{\alpha_k + \gamma_{k+1}}{2} \right\rfloor, \\ 0, & \text{otherwise,} \end{cases} \tag{12.3}$$

where each tensor depends on the carry propagated from the adjacent, less significant site, and the sequence is initialized with the carry at the original shift m.

However, following the last definition, the bond dimension increases with the magnitude of m, while most of the tensor entries remain zero. To optimize this representation, we consider the set of all possible carry values arising from the combinations of α_k and γ_k,

$$\Gamma = \left\{ \gamma_k : \gamma_k = \left\lfloor \frac{\alpha_k + \gamma_{k+1}}{2} \right\rfloor, \alpha_k \in \{0, 1\}, \gamma_{k+1} \in \Gamma^{\text{in}} \right\}.$$

Since Γ may have repeated, and non-contiguous values, we compress it into a sorted, contiguous collection of unique values

$$\Gamma^{\text{sorted}} = \{c_0, c_1, \ldots, c_{\ell-1}\},$$

with $c_i < c_{i+1}$. Then, we introduce the mapping

$$f : \Gamma \to \{0, 1, \ldots, \ell - 1\}, \qquad \text{with } f(c_k) = k,$$

which reindexes the carry values into a compact sequential set. This mapping ensures that the MPO tensors use the smallest bond dimension necessary to propagate the carry.

With this compression, the tensor elements are more efficiently defined as

$$A^k_{\gamma_k \alpha'_k \alpha_k \gamma_{k+1}} = \begin{cases} 1, & \text{if } \alpha'_k = (\alpha_k + \gamma_{k+1}) \bmod 2 \text{ and } \gamma_k = f\left(\left\lfloor \frac{\alpha_k + \gamma_{k+1}}{2} \right\rfloor\right), \\ 0, & \text{otherwise.} \end{cases} \tag{12.4}$$

In this formulation, the set of carry values contains only one or two distinct, adjacent values. Then, the bond dimension is at most 2, regardless of the shift magnitude.

After propagating through all sites, the final tensor still carries a free bond index corresponding to the remaining carry. To impose periodic boundary conditions and recover a left-most bond dimension of 1, this index is contracted

$$A^1_{\alpha'_1\alpha_1\gamma_1} = \sum_{\gamma_0} A^1_{\gamma_0\alpha'_1\alpha_1\gamma_1}, \tag{12.5}$$

and reshaped accordingly.

MPO for circulant matrices Circulant matrices can be formed as a linear combination of shift operators,

$$C = \begin{bmatrix} w_0 & w_1 & w_2 & \cdots & w_M \\ w_M & w_0 & w_1 & \cdots & w_{M-1} \\ w_{M-1} & w_M & w_0 & \cdots & w_{M-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_1 & w_2 & w_3 & \cdots & w_0 \end{bmatrix} = \sum_{i=0}^{M} w_i \hat{\Sigma}^{+i}.$$

Although one can construct the corresponding MPO by explicitly summing the weighted MPO representations of each shift operator, there is a more efficient strategy that reduces the intermediate bond dimensions. The analysis presented for a single shift can be extended to the case of multiple simultaneous shifts. In this framework, the chain of carry indices is initialized from an array representing multiple shifts. Since each index encodes a distinct shift, the rightmost MPO will have a bond dimension greater than 1 at the end of the chain, which can be contracted with the array of weights to obtain the MPO of the circulant matrix,

$$O^n_{\gamma_n\alpha'_n\alpha_n} = \sum_{\gamma_{n+1}} O^n_{\gamma_n\alpha'_n\alpha_n\gamma_{n+1}} w_{\gamma_{n+1}}, \tag{12.6}$$

where this last site must be reshaped to have a right bond dimension of size 1.

The bond dimension of the MPO can be analyzed as follows. Consider using a contiguous block of shifts, namely $\{s, s+1, \ldots, s+M-1\}$ with M different shifts. After one propagation, the carry rule produces a new set whose size is $\left\lfloor \frac{M}{2} \right\rfloor + 1$.

Iterating this process, and ignoring the floor operation, we obtain an approximate rule for the $(n-k)$-th site with a right bond dimension

$$\chi^{n-k} \lesssim 2 + \frac{M-2}{2^k}.$$

This result can be generalized to a set of discontiguous shifts by separation into p different contiguous sets of size M_i each. Then, the bond dimension of the k-th site fulfills

$$\chi^k \lesssim \sum_{i=1}^{p} \left(2 + \frac{M_i - 2}{2^{n-k}}\right),$$

with the maximum right bond dimension present on the second to last site, $(n-1)$.

Note that this is the bond dimension for the analytical construction, but the resulting MPO can be further compressed by the methods presented earlier.

12.1.1.2 Loading MPS/MPO via Interpolation

Loading functions It is often required to encode a complicated function, or even a function without known functional form as an MPS. One possible approach is to encode the function in a standard vector and to find an approximate MPS by repeated application of SVDs. However, in general, this approach is limited by the size of the vector. A more appropriate approach is to use interpolation techniques to approximate the MPS [296–298].

For completeness, we will briefly review the Chebyshev approximation of functions via MPS from Rodríguez-Aldavero *et al.* [298]. This method is based on Chebyshev interpolation, which achieves exponential convergence rates for analytical functions, and algebraic convergence rates for functions with limited differentiability. Moreover, the method is suitable in the MPS context: Although the bond dimension for an MPS that encodes a polynomial of degree d is bounded by $\chi_{\max} \leq d+1$, this method shows an improved linear scaling, $\chi_{\max} = O(0.28d)$.

The first step is to compute the Chebyshev-Gauss nodes, mapped to the domain $]a, b[$,

$$r_k = \frac{a+b}{2} - \frac{b-a}{2} \cos\left(\frac{\pi(2k-1)}{2d}\right),$$

for $k \in \{1, .., d\}$. Then, the function of interest is evaluated at these nodes.

The Chebyshev expansion of $f(x)$ is, up to degree d,

$$f(x) \approx \sum_{k=0}^{d} c_k T_k(x),$$

where $T_k(x)$ is the k-th Chebyshev polynomial of the first kind. The Chebyshev coefficients c_k are calculated using the discrete cosine transform (DCT),

$$c_k = \mathrm{DCT}\{f(r)\}_k.$$

Then, one can reconstruct $f(x)$: Based on the recurrence relation of the Chebyshev polynomials,

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x),$$

the Clenshaw evaluation provides an efficient and numerically stable algorithm. Intermediate MPS are calculated according to the relation

$$|y_k\rangle = c_k|I\rangle - |y_{k+1}\rangle + 2|x\rangle \odot |y_{k+1}\rangle,$$

for $k = d, \ldots, 0$, where $|y_{d+1}\rangle = |y_d\rangle$ are vectors of only zeros, $|I\rangle$ is a vector of only ones, and $|x\rangle$ encodes the spatial grid. Here, $\langle\odot\rangle$ denotes a point-wise multiplication. Finally, the MPS encoding $|f(x)\rangle$ is obtained as $|f(x)\rangle = |y_0\rangle - |x\rangle$.

Loading diagonal operators In Section 12.3.3, we require to load diagonal operators as MPO, which act as functions of the coordinate basis,

$$G|f(x)\rangle = |g(x)f(x)\rangle.$$

Although interpolative techniques can be used to construct a diagonal MPO from the two-dimensional function $h(x, y) = \delta_{xy} g(x)$, a more accurate and efficient approach is to first encode the function $g(x)$ as an MPS g, and then convert it into a diagonal MPO G. The procedure for recasting an MPS as a diagonal MPO was explained on Section 4.5.6.

12.2 MPO Encoding for Differential Operators

Given the efficient MPS representations for functions, a relevant challenge is finding suitable representations for operators that act on these functions. This includes potentials, derivative operators, and evolution operators that govern the system dynamics. Since potentials are diagonal and can often be derived directly in the MPS representation—either exactly or through interpolation techniques such as Chebyshev approximation [298] or TT-cross interpolation [296, 297]—the more significant challenge lies in constructing efficient representations for derivatives and related evolution operators.

Finite difference methods are widely used to approximate derivatives. These are based on Taylor expansions of order p, with errors of $O(\Delta x^m)$, where m depends on the specific combination of terms. Hence, the accuracy of the method is fundamentally limited by the grid resolution.

The most common schemes are the centered finite difference formulas,

$$\frac{\partial f(x)}{\partial x} = \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x} + O(\Delta x^2), \tag{12.7}$$

$$\frac{\partial^2 f(x)}{\partial x^2} = \frac{f(x + \Delta x) - 2f(x) + f(x - \Delta x)}{\Delta x^2} + O(\Delta x^2), \tag{12.8}$$

with truncation errors scaling quadratically with the grid spacing. Noise amplification in these schemes can be mitigated to construct smoother differentiators [299].

In the MPS formalism, finite difference operators can be implemented using a

linear combination of displacement operators $\Sigma^{\pm}$ [120],

$$|\partial_x f\rangle \simeq \frac{1}{2\Delta x}\left(\hat{\Sigma}^{+} - \hat{\Sigma}^{-}\right)|f\rangle + O(\Delta x^2), \tag{12.9}$$

$$\left|\partial_x^2 f\right\rangle \simeq \frac{1}{\Delta x^2}\left(\hat{\Sigma}^{+} - 2\mathbb{I} + \hat{\Sigma}^{-}\right)|f\rangle + O(\Delta x^2), \tag{12.10}$$

with a constant bond dimension $\chi = 3$, independent of the system size.

Besides truncation error, finite difference schemes also suffer from round-off error. This error becomes significant when the function differences $|f(x) - f(x \pm \Delta x)|$ approach machine precision δ. For second derivatives, round-off errors scale as $\delta/\Delta x^2$, potentially becoming large for small Δx. A practical way to reduce this error is to increase Δx while keeping the number of grid points fixed.

This work seeks to overcome the precision and flexibility limitations of finite-difference formulas. Spectral methods, particularly Fourier-based techniques, are known to offer exponential convergence and high accuracy for sufficiently smooth functions. These methods have been adapted to the MPS/QTT framework [120, 121, 136, 137], often using ad-hoc heuristics for operator construction. We present an alternative spectral approach involving Hermite Distributed Approximating Functionals (HDAFs), a powerful yet lesser-known tool in numerical analysis.

In the following sections, we describe how HDAFs can be adapted to construct MPOs for arbitrary differential operators, offering high accuracy with relatively low computational cost. Our method differs from earlier works where HDAFs were applied in matrix form to each site in tensor trains [300]. In contrast, we build global MPOs compatible with the MPS algebra.

12.2.1 Hermite Distributed Approximating Functionals

This section presents an extension of the Hermite Distributed Approximating Functionals (HDAF) to reconstruct functions of differential operators within a finite-precision MPS-MPO framework. The HDAF performs this reconstruction as a linear combination of Hermite polynomials weighted by a Gaussian filter, approximating

these operators with tunable pseudospectral precision at a limited cost. This section also covers a general review of the HDAF and the metaheuristics behind the use of this technique.

The Distributed Approximating Functionals (DAF) are well-tempered approximations to the Dirac delta distribution. The first DAF, developed before the name was coined, were the Hermite Distributed Approximating Functionals (HDAF) [267],

$$\delta_M(x;\sigma) = \frac{\exp\left(\frac{-x^2}{2\sigma^2}\right)}{\sqrt{2\pi}\sigma} \sum_{m=0}^{M/2} \left(-\frac{1}{4}\right)^m \frac{H_{2m}\left(\frac{x}{\sqrt{2}\sigma}\right)}{m!}, \tag{12.11}$$

where $H_n(x)$ is the n-th Hermite polynomial. It has two free parameters: The even integer M and the positive real σ are the order of the highest polynomial and the width of the approximation to the delta distribution, respectively.

The kernel defined in Equation (12.11) is a nascent delta function that operates as the identity for polynomial functions of degree M or lower,

$$f(x) \approx \int dx'\, \delta_M(x - x';\sigma) f(x'), \tag{12.12}$$

approaching the Dirac distribution in the limit $\sigma/M \to 0$. However, unlike the exact delta distribution, $\delta_M(x;\sigma)$ is generally a bandwidth-limited, infinitely smooth function that is compatible with quadrature methods and differentiation.

The well-tempered property of the method arises from the absence of special points in the reconstruction. Exact reproduction at grid points is not required; therefore, it does not constitute an interpolation scheme. Moreover, a fundamental property of Eq. (12.12) is that the approximation converges uniformly to $f(x)$ [301], and the approximation error usually resembles the function $f(x)$, although it is several orders of magnitude smaller [270]. This implies that the error is smaller when the function approaches zero, which is relevant and desirable when representing wavefunctions.

Equation (12.12) is customarily discretized on a uniform grid with spacing Δx, using midpoint integration to render it as a matrix-vector product, $f(x_i) =$

$\sum_j K_{ij} f(x_j)$, where K is a symmetric Toeplitz matrix with components

$$K_{ij} = \Delta x \delta_M(\Delta x|i-j|;\sigma). \tag{12.13}$$

Since $\delta_M(x;\sigma)$ has an exponentially decaying envelope, two essential properties arise: (i) the reconstruction matrix K can be highly sparse and concentrated around its main diagonal, with the number of diagonals controlled by $\sigma/\Delta x$, and (ii) a minimal number of diagonals (that is, quadrature nodes) is required to accurately discretize the integral in Eq. (12.12). The composite midpoint rule converges especially fast for periodic or peaked functions with vanishing derivatives at integration limits [302]. Moreover, in this context, the midpoint rule surpasses the accuracy of higher-order Newton-Cotes quadrature rules using the same number of grid points [303, 304].

The MPO corresponding to the K matrix can be constructed as a weighted combination of the displacement operators $\Sigma^{\pm}$ as

$$\hat{K} = \Delta x \delta_M(0;\sigma)\mathbb{I} + \sum_{i=1}^{2^n-1} \Delta x \delta_M(i\Delta x;\sigma)\left(\hat{\Sigma}^{+i} + \hat{\Sigma}^{-i}\right), \tag{12.14}$$

where the symmetry of δ_M has been used. While in principle the sum ranges over the entire grid, in practice, one only needs to sum until some integer W such that δ_M has vanished according to a prescribed tolerance, as detailed in Section 12.2.4.4.

12.2.2 HDAF differentiation

The HDAF formalism opens the way to estimate the derivative of a function of any order, as well as functions $D[\partial/\partial x]$ of such derivatives. From Eq. (12.12) it follows that

$$D\left[\frac{\partial}{\partial x}\right] f(x) \approx \int dx'\, D\left[\frac{\partial}{\partial x}\right] \delta_M(x-x') f(x'). \tag{12.15}$$

An analytical expression for $D[\partial/\partial x]\delta_M(x-x')$ is usually easy to find. The typical procedure involves using the Rodrigues formula for the Hermite polynomials,

$$H_n(x) = (-1)^n \exp\left(x^2\right) \frac{\partial^n}{\partial x^n} \exp\left(-x^2\right), \tag{12.16}$$

to rewrite Eq. (12.11) as

$$\delta_M(x;\sigma) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{m=0}^{M/2} \frac{1}{m!} \left(-\frac{\sigma^2}{4}\right)^m \frac{\partial^{2m}}{\partial x^{2m}} \exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{12.17}$$

Then, the differential operator is applied to $\delta_M(x-x';\sigma)$ acting on the exponential, and the Rodrigues' formula (12.16) is used to recover an expression in terms of Hermite polynomials, without explicit derivatives. For instance, the l-th derivative leads to

$$\delta_M^{(l)}(x;\sigma) = \left(\frac{-1}{\sqrt{2}\sigma}\right)^l \frac{\exp\left(\frac{-x^2}{2\sigma^2}\right)}{\sqrt{2\pi}\sigma} \sum_{m=0}^{M/2} \left(-\frac{1}{4}\right)^m \frac{H_{2m+l}\left(\frac{x}{\sqrt{2}\sigma}\right)}{m!}. \tag{12.18}$$

Analog to the reconstruction (12.14), the differentiating MPO for the l-th derivative in the HDAF formalism is

$$\hat{K}^{(l)} = \Delta x \delta_M^{(l)}(0;\sigma)\mathbb{I} + \sum_{i=1}^{2^n-1} \Delta x \delta_M^{(l)}(i\Delta x;\sigma) \left(\hat{\Sigma}^{+i} + (-1)^l \hat{\Sigma}^{-i}\right), \tag{12.19}$$

where the symmetry or antisymmetry of $\delta_M^{(l)}$ has been used for l even or odd, respectively.

Attention must be paid to the fact that the $l-$th derivative has a prefactor $\sigma^{-(l+1)}$ with $\sigma = O(\Delta x)$. Since these operators will be used within a finite-precision numerical framework, round-off errors can significantly impact the accuracy of the differentiation. These errors arise from the limited significant digits that computers may represent, inducing small approximation errors amplified by large weighting prefactors. In particular, these deviations dominate when Δx is too small, which limits how dense the numerical grid can be.

The round-off error amplification problem is expected in numerical differentiation techniques, but can be mitigated. In the particular MPS-MPO framework, differentiating operators can be discretized to accommodate a certain number of qubits and

then be adapted to a denser grid, where identities are added as new sites to account for each additional qubit. This approach is equivalent to nearest-neighbor interpolation. It keeps the round-off errors constant and does not introduce additional complexity to the MPO.

Figure 12.2 accounts for the differentiation accuracy of HDAF and finite differences in approximating the second derivative of a Gaussian function. In the case of the HDAF, the convergence is faster in the number of qubits for larger values of M, as expected. However, HDAF and finite differences suffer from relevant round-off errors for many qubits. Once optimal accuracy is achieved for a certain number of qubits, the accuracy of the operators can be retained while acting on a finer grid. Round-off errors are effectively kept constant by following the procedure above.

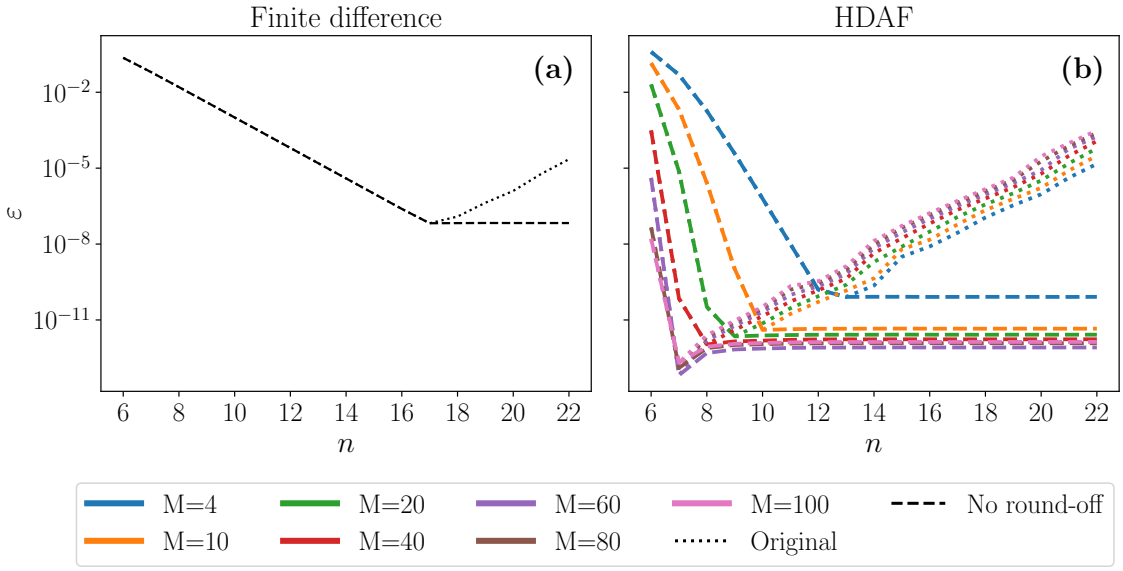


Figure 12.2: Errors in the second derivative approximation of a Gaussian function for a varying number of qubits. Dotted lines correspond to the direct implementation of the differentiating operators. Dashed lines implement the procedure specified at the end of Section 12.2.2 to limit round-off errors. (a) Finite differences, (b) HDAF.

12.2.3 HDAF free propagator

The HDAF scheme is very powerful, as it cannot only approximate derivatives but also functions of those derivatives. The first and one of the most relevant applications of this idea, posed in [267], is the banded approximation of the free propagator.

From Eq. (12.15), taking the differential operator $D[\partial/\partial x]$ to be the free propagator,

$$T(\tau) = \exp\left(-\frac{i\tau}{2}\frac{\partial^2}{\partial x^2}\right), \tag{12.20}$$

the kernel for the approximation is $T(\tau)\delta_M(x-x';\sigma)$. This quantity is readily computed from Eq. (12.17) since $T(\tau)$ commutes with the derivatives and it spreads a Gaussian function to another,

$$T(\tau)\exp\left(\frac{-(x-x')^2}{2\sigma^2}\right) = \left[\frac{\sigma}{\sigma_\tau}\right]\exp\left(\frac{-(x-x')^2}{2\sigma_\tau^2}\right), \tag{12.21}$$

mapping the original variance σ^2 to $\sigma_\tau^2 = \sigma^2 + i\tau$. Then, using Eq. (12.16) yields the free propagator kernel

$$\begin{aligned}\delta_M(x-x';\sigma,\tau) &= T(\tau)\delta_M(x-x';\sigma)\\ &= \frac{\exp\left(\frac{-(x-x')^2}{2\sigma_\tau^2}\right)}{\sqrt{2\pi}\sigma_\tau}\sum_{m=0}^{M/2}\left(\frac{-\sigma^2}{4\sigma_\tau^2}\right)^m\frac{H_{2m}\left(\frac{x-x'}{\sqrt{2}\sigma_\tau}\right)}{m!}.\end{aligned} \tag{12.22}$$

The MPO for the free propagator in the HDAF formalism is

$$\hat{K}_\tau = \Delta x\delta_M(0;\sigma,\tau)\mathbb{I} + \sum_{i=1}^{2^n-1}\Delta x\delta_M(i\Delta x;\sigma,\tau)\left(\hat{\Sigma}^{+i}+\hat{\Sigma}^{-i}\right). \tag{12.23}$$

The kernel (12.22) becomes complex and highly oscillating as time increases. In addition, its width increases as a fundamental consequence of the free propagator. However, the spreading in the HDAF formalism is the minimum possible, since it is inherited from the Gaussian generator of the Hermite polynomials [268].

The HDAF approximation for the propagator has been used in many applications of split-step integration methods within the traditional vector framework. In that

framework, $\hat{K}_\tau$ is represented as a matrix acting on a discretized function. A central contribution in this work is to realize that the same matrix can be more efficiently represented as an MPO using the displacement operators $\hat{\Sigma}^{\pm}$ and additional simplification steps that significantly reduce the effective bond dimension of the operator. In this scenario, the MPO HDAF propagator is a competitive alternative to using MPO Fourier-based techniques [120], directly representing the evolution operator in the coordinate representation.

12.2.4 HDAF metaheuristics

12.2.4.1 Free parameter election

Identical reconstruction. The formulation of the HDAF operator (12.12) as a discrete matrix has 2 sources of error: (i) the assumption that the function $f(x)$ can be expressed as a polynomial of degree M within the extent of the Gaussian envelope of the HDAF, and (ii) the discretization of the convolution integral to a finite sum employing the midpoint rule. While the error (i) vanishes in the limit $M/\sigma \to \infty$, it is clear from (ii) that it is not possible to increase M or decrease σ indefinitely. A more oscillatory integrand will require a larger value of $\sigma/\Delta x$ for the Gaussian envelope to cover enough nodes and achieve satisfactory integration accuracy.

In general, for a fixed M, there is a value of $\sigma/\Delta x$ that makes the reconstruction optimal, and the larger is M, the better the maximum accuracy that can be achieved. The rationale behind this optimal relationship between M and $\sigma/\Delta x$ is that a perfect reconstruction occurs when the M zeros of the HDAF match the zeros on the grid, and only the term in the origin contributes [269]. One possible approach, therefore, is to set the discrete HDAF to be 1 at the origin [305],

$$K_{ii} = \Delta x \delta_M(0;\sigma) = 1,$$

yielding,

$$\sigma_M = \frac{\Delta x}{\sqrt{2\pi}} \sum_{m=0}^{M/2} \left(\frac{-1}{4}\right)^m \frac{H_{2m}(0)}{m!}, \tag{12.24}$$

which makes the HDAF approximately vanish at integer multiples of Δx [306].

In practice, a lower bound to $\sigma/\Delta x$ is prescribed to ensure convergence of the midpoint rule when M is small. For the context of double floating-point precision, we heuristically set

$$\sigma/\Delta x \geq 3 \Rightarrow \sigma_{\min} = 3\Delta x, \tag{12.25}$$

and choose the value of σ according to

$$\sigma = \max(\sigma_M, \sigma_{\min}). \tag{12.26}$$

Differentiation. Assuming that the $l-$th derivative of the function $f(x)$ is accurately described with an HDAF of order M, i.e., also pertains to the DAF-class [270], then the approximation to the derivative can be thought as a reconstruction of $f^{(l)}(x)$ instead of $f(x)$,

$$\begin{aligned} f^{(l)}(x) &\approx \int dx'\, \delta_M^{(l)}(x - x'; \sigma) f(x') \\ &= \int dx'\, \delta_M(x - x'; \sigma) f^{(l)}(x'). \end{aligned}$$

In this spirit, the optimal value of σ is computed again using Eq. (12.26), which does not depend on the function to reconstruct, provided that M is fixed.

Free evolution. Since the action of the free propagator is to spread the original wavefunction, the width of the HDAF will not be a problem for midpoint integration. For efficiency purposes, the election of σ is made so that the new width of the freely propagated HDAF is the smallest possible [267]. This value follows from equation (12.22). The leading Gaussian,

$$\exp\left(\frac{-x^2}{2(\sigma^2 + i\tau)}\right) = \exp\left(\frac{-x^2}{2w^2}\right) \exp\left(i \frac{x^2}{2w^2} \frac{\tau}{\sigma^2}\right),$$

has an effective variance $w^2 = (\sigma^2 + \tau^2/\sigma^2)$ with an optimal value $\sigma = \sqrt{\tau}$ that minimizes its spatial extent. Then, the value of σ is chosen

$$\sigma = \max(\sigma_M, \sigma_{\min}, \sqrt{\tau}), \tag{12.27}$$

where σ_M and $\sigma_{\min}$ and are the same values (12.24) and (12.25) used for identical HDAF reconstruction.

12.2.4.2 Self-consistent error estimation

The HDAF filter (12.12) can be analyzed in Fourier space. From equation (12.17), the kernel spectrum has the analytical form

$$\widehat{\delta}_M(k;\sigma) = \exp\left(\frac{-k^2\sigma^2}{2}\right) \sum_{m=0}^{M/2} \frac{1}{m!} \left(\frac{k^2\sigma^2}{2}\right)^m. \tag{12.28}$$

The summation is a truncated series expansion of $\exp(k^2\sigma^2/2)$ to the order $M/2$. This expression is the basis to prove that the HDAF filter approaches a true Dirac delta distribution in the limits of an infinitely broad filter or an infinitely large polynomial basis,

$$\lim_{\sigma\to 0} \widehat{\delta}_M(k;\sigma) = \lim_{M\to\infty} \widehat{\delta}_M(k;\sigma) = 1 \quad \forall k \in \mathbb{R},$$

thus identically preserving the function to reconstruct.

From equation (12.28), one can also note that the Fourier expression of the HDAF is symmetric, bounded, and monotonically decreasing in $k \in \mathbb{R}^+$, with

$$1 = \widehat{\delta}_M(0;\sigma) \geq \widehat{\delta}_M(k;\sigma) \geq \lim_{k\to\infty} \widehat{\delta}_M(k;\sigma) = 0,$$

therefore acting as a low-pass filter. Moreover, it has been shown that $\widehat{\delta}_M(k,\sigma)$ is an almost-ideal low-pass filter with transition frequency

$$k^\star = \sqrt{M+1}/\sigma, \tag{12.29}$$

and a transition region width that scales as $O(M^{-1/2}\sigma^{-1})$ [272].

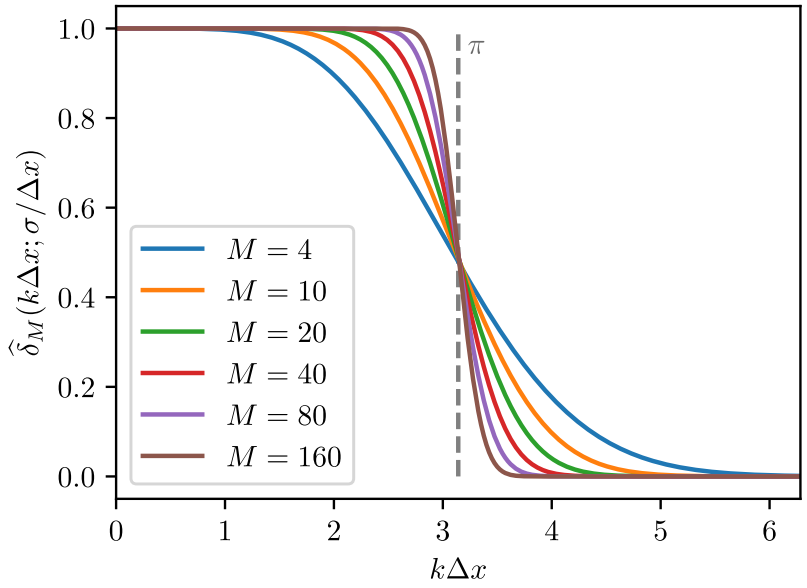


Figure 12.3: Fourier spectrum of $\delta_M(x;\sigma)$. Frequencies and widths are normalized to the grid spacing Δx. The width σ is computed for each M according to equation (12.24).

This behavior is depicted in Figure 12.3 for varying M. There is a region of frequencies below $k^\star$, the so-called DAF plateau, such that $\widehat{\delta}_M(k;\sigma) \approx 1$. There follows a transition region centered around $k^\star$ where the value of the filter smoothly vanishes, and then it indefinitely approximates to zero. As M increases, the DAF plateau extends closer to $k^\star$. Note that choosing σ from equation (12.24) fixes the transition frequency $k^\star \approx \pi/\Delta x$, which is the maximum frequency that can be represented on a discrete uniform grid.

The HDAF reconstruction will be accurate for bandwidth-limited functions whose spectra lie within the DAF plateau, and any function with higher frequency contributions will be smoothed. This suggests that this formalism is a good fit to use together with matrix product states, since both techniques are especially suitable to represent bandwidth-limited functions [121].

12.2.4.3 Evaluation of the HDAF coefficients

All the HDAF operators presented here are generated as a combination of displacements and coefficients

$$K_{\tau}^{(l)} = \sum_{k=-2^n+1}^{2^n-1} \Delta x \delta_M^{(l)}(k\Delta x; \sigma, \tau) \left(\Sigma^+\right)^k . \tag{12.30}$$

Moreover, the coefficients fulfill the general form

$$\Delta x \delta_M^{(l)}(x; \sigma, \tau) = d_l \sum_{m=0}^{M/2} h_{m,l}(x/\sqrt{2(\sigma^2 + i\tau)}), \tag{12.31}$$

with the definitions

$$\begin{aligned} d_l &= \frac{(-1)^l \Delta x}{\sqrt{2(\sigma^2 + i\tau)}^{l+1}\sqrt{\pi}}, \\ h_{n,l}(x) &= H_{2n+l}(x) \exp\left(-x^2\right) \frac{c^n}{n!}, \\ c &= -\frac{1}{4}\frac{\sigma^2}{(\sigma^2 + i\tau)}. \end{aligned}$$

From the properties of Hermite polynomials, it follows that $h_{n,l}(x)$ obeys the double recurrence relation

$$\begin{aligned} h_{n+1,l}(x) &= \frac{2c}{n+1}\left[x h_{n,l+1}(x) - (2n + l + 1) h_{n,l}(x)\right], \\ h_{n+1,l+1}(x) &= 2x h_{n+1,l}(x) - 2c(2 + l/(n+1)) h_{n,l+1}(x), \end{aligned}$$

that makes the calculation of (12.31) efficient and accurate, starting from the initial values $h_{0,l}(x) = H_l(x)\exp(-x^2)$ and $h_{0,l+1}(x) = H_{l+1}(x)\exp(-x^2)$.

12.2.4.4 Effective summation bounds

It was previously mentioned that equations (12.14), (12.19), (12.23) and (12.30) formally sum over all grid points of x. However, in practice, the exponential decay of the HDAF narrows the sum to a small subset of points around the origin. This subset can be further restricted since the filters are either symmetric or antisymmetric.

Only the highest power in the argument of $\delta_M^{(l)}(x;\sigma,\tau)$ will contribute significantly to the value of the HDAF for $x \gg 0$. Let W be the smallest positive integer such that the coefficients contribute at most a predefined error tolerance ε,

$$\left|\Delta x \delta_M^{(l)}(W\Delta x;\sigma,\tau)\right| \leq \varepsilon.$$

This integer can be estimated tightly by replacing the sum in (12.31) with the highest power term in $W\Delta x$ from the polynomial in $h_{M/2,l}(W\Delta x/\sqrt{2(\sigma^2+i\tau)})$, but it leads to a transcendental equation for W. Instead, we approximate the sum by its last term as a whole and use the following upper bound for the Hermite polynomials with complex argument [307],

$$|H_n(z)| \leq \sqrt{2}^n \sqrt{n!}\exp\left(\sqrt{2n}|z|\right), \quad z \in \mathbb{C}, n \in \mathbb{N}.$$

This bound tends to overestimate the polynomial, but the Gaussian envelope quickly dominates, preventing this from becoming a significant problem. Then, we find W by setting

$$\begin{aligned}|\Delta x \delta_M^{(l)}| &\lesssim \frac{\Delta x\sqrt{(M+l)!}}{\sqrt{2\pi}\sqrt{|\sigma^2+i\tau|}^{l+1}(M/2)!}\left|\frac{\sigma^2}{2(\sigma^2+i\tau)}\right|^{M/2} \\ &\quad \times \exp\left(W\Delta x\sqrt{\frac{M+l}{|\sigma^2+i\tau|}} - \frac{W^2\Delta x^2}{2(\sigma^2+\tau^2/\sigma^2)}\right) \\ &= \varepsilon,\end{aligned}$$

which reduces to the quadratic equation,

$$W^2\frac{\Delta x^2}{2(\sigma^2+\tau^2/\sigma^2)} - W\Delta x\sqrt{\frac{M+l}{|\sigma^2+i\tau|}} + \ln\frac{\varepsilon}{\eta} = 0, \tag{12.32}$$

with

$$\eta = \frac{\Delta x\sqrt{(M+l)!}}{\sqrt{2\pi}\sqrt{|\sigma^2+i\tau|}^{l+1}(M/2)!}\left|\frac{\sigma^2}{2(\sigma^2+i\tau)}\right|^{M/2}.$$

HDAF MPOs (12.30) are obtained by summing over indices $-W$ to W, where W is the closest integer from above to the solution of (12.32). This sum contains $2W+1$ weighted displacement operators $\hat{\Sigma}^{\pm k}$, where only $W+1$ coefficients must be explicitly computed due to the symmetry $\delta_M^{(l)}(-x) = (-1)^l \delta_M^{(l)}(x)$. Despite the number of summands, in practice the resulting MPO will be relatively simple, with a small bond dimension, for a reasonable choice of the HDAF parameters.

12.3 Time evolution algorithms

Our goal is to solve the time-dependent Schrödinger equation (11.1). The formal solution can be written as the repeated action of a time-dependent unitary operator $U(t)$ on an initial state $\psi(x, t=0)$:

$$\psi(x,t) = U(t)\psi(x,0) = e^{-it\hat{H}}\psi(x,0), \tag{12.33}$$

where $\hat{H} = D(-\partial_x^2) + V(x)$, with D a linear function and $V(x)$ a potential.

In our applications, the state $\psi(x,t)$ is encoded using MPS/QTT, and the time evolution is approximated using MPOs and finite-precision MPS algebra. Specifically, we either encode the Hamiltonian $\hat{H}$ as an MPO and construct the time evolution operator $U(t)$ numerically, or approximate $U(t)$ directly as an MPO. In both cases, the QTT/MPS representation allows us to operate on exponentially large grids with 2^n spatial points for n qubits, offering a potential advantage over standard vector-based methods.

To solve the time evolution problem, the PDE operators from Section 12.2 require global evolution schemes that are not limited to local interactions. Several MPS algorithms are available for this purpose, including the time-dependent variational principle (TDVP) [261, 308, 309], as well as Taylor, Padé, and Arnoldi approximations of the evolution operator [310].

This section presents a selection of time evolution methods implemented using MPO-MPS. These include explicit Runge-Kutta schemes (Euler, Improved Eu-

ler, and standard fourth-order Runge-Kutta), implicit methods (Crank-Nicolson), restarted Arnoldi iteration, and the split-step method. The use of HDAF-based operator approximations enables the application of the split-step method directly in the coordinate representation. All other schemes are compatible with both finite difference and HDAF representations of the differential operator.

12.3.1 Runge-Kutta Methods

Runge-Kutta methods approximate time evolution using a Taylor expansion of the state. The local (*i.e.* one-step) error scales algebraically with the expansion order m as $O(\Delta t^{m+1})$. Below, we summarize some representative variants.

12.3.1.1 Euler method

The Euler method is the simplest first-order scheme,

$$\begin{aligned} \psi_0 &= \psi(x, t_0), \\ \psi_{k+1} &= \psi_k - i\Delta t H \psi_k. \end{aligned} \tag{12.34}$$

12.3.1.2 Improved Euler or Heun method

The Improved Euler method, also known as Heun's method, is a second-order scheme that refines the Euler step using a predictor-corrector scheme, resulting in

$$\psi_{k+1} = \psi_k - i\frac{\Delta t}{2}\left[v_1 + v_2\right], \tag{12.35}$$

where $v_1 = H\psi_k$, and $v_2 = H(\psi_k - i\Delta t v_1)$.

12.3.1.3 Fourth-order Runge-Kutta method

The well-known fourth-order Runge-Kutta method is widely used for solving PDEs due to its balance between accuracy, stability and simplicity. It is given by

$$\begin{aligned}
\psi_{k+1} &= \psi_k - i\frac{\Delta t}{6}(v_1 + 2v_2 + 2v_3 + v_4), \text{ with} \\
v_1 &= H\psi_k, \\
v_2 &= H\left(\psi_k - i\frac{\Delta t}{2}v_1\right), \\
v_3 &= H\left(\psi_k - i\frac{\Delta t}{2}v_2\right), \\
v_4 &= H\left(\psi_k - i\Delta t v_3\right).
\end{aligned} \tag{12.36}$$

12.3.1.4 Crank–Nicolson Method

Implicit methods can offer enhanced numerical stability, particularly for stiff systems. The Crank–Nicolson scheme is a second-order implicit integrator based on the trapezoidal rule. It combines the forward (explicit) and backward (implicit) Euler steps. The update rule for the state at time step $k+1$ is given by

$$\left(\mathbb{I} + \frac{i\Delta t}{2}H\right)\psi_{k+1} = \left(\mathbb{I} - \frac{i\Delta t}{2}H\right)\psi_k. \tag{12.37}$$

This equation defines a linear system that must be solved at each step. In standard matrix-vector implementations, it can be handled via direct solvers or iterative methods. In the MPS/MPO framework, iterative solvers such as conjugate gradient descent can be adapted to perform the inversion.

12.3.2 Restarted Arnoldi Iteration

The restarted Arnoldi iteration is particularly well suited for large-scale problems. The idea is to project the evolution operator onto a Krylov subspace spanned by the basis

$$\{|v_i\rangle = H^{i-1}|\psi_k\rangle\}_{i=1}^{n_v},$$

constructed from repeated applications of the Hamiltonian.

With this basis, the method builds the matrices

$$A_{ij} = \langle v_i | H | v_j \rangle,$$
$$N_{ij} = \langle v_i | v_j \rangle,$$
$$V_{ij} = \langle j | v_j \rangle,$$

and approximates the time evolution as

$$|\psi_{k+1}\rangle \approx V^\dagger e^{-i\Delta t N^{-1} A} V |\psi_k\rangle.$$

The accuracy of the method improves with the number of Krylov vectors, scaling as $O(\Delta t^{n_v})$. In practice, even a small number of vectors ($n_v = 5$–10) may yield accurate results. A key advantage of this approach is that the projected matrices A and N are of fixed size, independent of the system dimension. This makes the method computationally efficient and well suited for quantum-inspired representations where explicit use of the full Hamiltonian is impractical.

12.3.3 Split-Step Method

Split-step methods approximate the exponential of the Hamiltonian by decomposing it into a product of exponentials that can be applied more efficiently. These methods exploit the separability of the Hamiltonian into kinetic and potential components.

The first-order approximation is based on the Lie–Trotter product formula,

$$U(\Delta t) \approx e^{-i\Delta t D(-\partial_x^2)} e^{-i\Delta t V(x)} + O(\Delta t^2), \tag{12.38}$$

where $D(-\partial_x^2)$ represents the kinetic term and $V(x)$ the potential.

Higher-order approximations, such as the second-order symmetric decomposition,

$$e^{-i\Delta t(D(-\partial_x^2)+V(x))} \approx e^{-i\Delta t V(x)/2} e^{-i\Delta t D(-\partial_x^2)} e^{-i\Delta t V(x)/2} + O(\Delta t^3), \tag{12.39}$$

reduce the error scaling and improve accuracy. This specific form corresponds to the Störmer–Verlet method, a widely used second-order symplectic integrator [311], known for its ability to conserve energy in long-time simulations of conservative systems. Higher-order generalizations can be derived using Suzuki–Trotter expansions [312, 313], although they require additional exponential evaluations [314].

Traditionally, the application of the free propagator $\exp(-i\Delta t D(-\partial_x^2))$ requires two Fourier transforms to diagonalize the operator in momentum space. In the MPS formalism, this is implemented using the quantum Fourier transform (QFT) or some approximation thereof [137, 253]. While these transforms are more efficient than the classical FFT, they still add computational overhead. In contrast, the Hermite Distributed Approximating Functional (HDAF) framework avoids that problem entirely. We apply this operator directly in the coordinate space using the free-propagator approximation (12.22), thereby replacing the aplication of three MPOs with a single MPO.

The potential propagator $\exp\left(-i\frac{\Delta t}{2}V(x)\right)$, being diagonal in the coordinate basis, is efficiently approximated using interpolation as implemented in Rodríguez-Aldavero *et al.* [298]. While simple potentials, such as the harmonic oscillator, admit exact MPO representations, more complex cases, like the double-well potential in Equation (11.7) require the numerical construction of the MPO.

Chapter 13

Application and results

"A journey of a thousand miles begins with a single step"

Lao Tzu

13.1 One-step study

All methods described in Section 12.3 must be applied iteratively in small time steps Δt to simulate the full evolution of a quantum state $\psi(x,t)$. Studying and comparing their performance over a single integration step provides a useful starting point to understand their long-term accuracy and efficiency.

As a benchmark, we consider the evolution of a quenched state under the Hamiltonian (11.2), which admits an analytical solution. This allows us to estimate the error in the wavefunction across different algorithms, grid sizes, and time steps. Specifically, we simulate the expansion of a quantum state following a sudden reduction in the trapping frequency of a harmonic potential, with $\omega_H/\omega_0 = 0.01$. This corresponds to a 100-fold increase in the width of the wavefunction, from an initial standard deviation σ_0 to a final value $\sigma_{\max} = 100\sigma_0$.

The spatial simulation domain is chosen to accommodate the full expansion of

the wavepacket, covering the interval $x \in [-L/2, L/2)$ with $L = 16\sigma_{\max}$. Since the initial state is highly localized around $x = 0$, an accurate representation of both the initial and final states imposes a lower bound on the grid resolution and the number of qubits required.

Representing such a sharply peaked initial function poses a challenge for MPS-based methods. In particular, loading a function that is nearly zero over most of the domain requires careful padding with zeros to mitigate potential sampling or approximation errors introduced by TT-cross or Chebyshev interpolation methods in the exponentially decaying tails.

The accuracy and performance of the algorithms are gauged using three figures of merit: (i) the function norm-2 difference

$$\varepsilon = \sqrt{\Delta x \sum_i \left|\psi(x_i, \Delta t) - \tilde{\psi}(x_i, \Delta t)\right|^2}, \tag{13.1}$$

measures the accuracy of the methods, where $\psi(x_i, \Delta t)$ is the analytic solution for $t = \Delta t$, and $\tilde{\psi}(x_i, \Delta t)$ is the one-step approximation. The other figures of merit, (ii) the run time, and (iii) the maximum bond dimension $\chi_{\max}$ determine their cost.

The one-step benchmark study must separately assess the effects of time and spatial discretizations to isolate the error contributions that come from numerical integration and PDE operator representation. As discussed in Section 12.2, the HDAF approach offers an exponential improvement over finite differences in the approximation of derivatives. Based on this observation, the first set of simulations employs a fixed spatial discretization with $n = 18$ qubits (262,144 grid points) to compare different time integration schemes. The finite difference method uses filter nine from Reference [299] to enhance noise robustness, while the HDAF approximation is fixed with $M = 40$. Both methods assume periodic boundary conditions.

Figure 13.1 compares the function error ε (13.1) and the run time as functions of the time step Δt for the finite difference and HDAF approximations in all integration methods. In Figure 13.1(b), the HDAF-based simulations exhibit the expected error

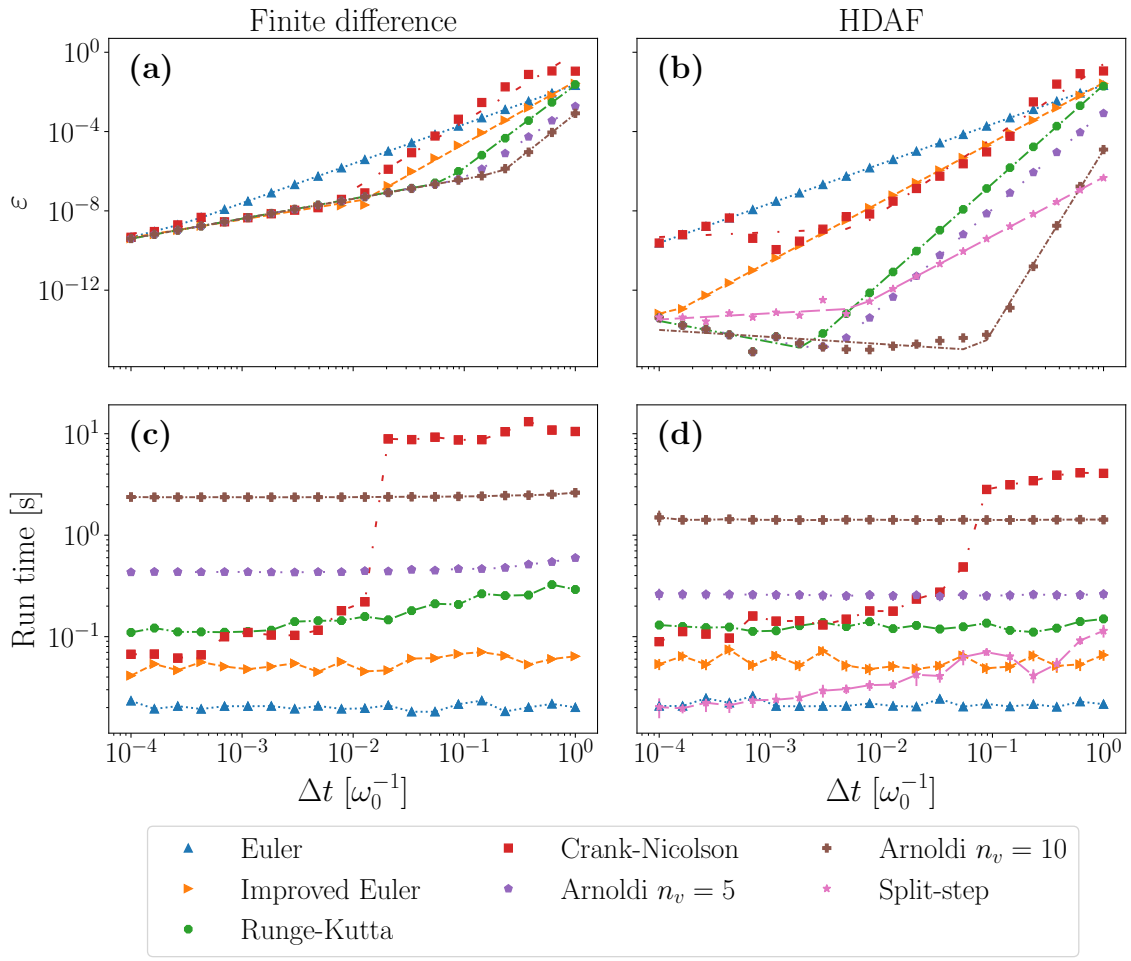


Figure 13.1: One-step evolution for a range of Δt and a fixed number of qubits $n = 18$. (a) Error ε (finite difference), (b) Error ε (HDAF), (c) run time (finite difference), (d) run time (HDAF). The run time is averaged over ten runs.

scaling for each integration method, reaching a numerical precision limited only by MPS truncation. In contrast, the finite difference results in Figure 13.1(a) show a plateau in accuracy due to the truncation error, which dominates for small Δt and prevents further error reduction. Appendix B provides linear fits that account for the scaling of ε with Δt.

As shown in Figures 13.1(c)-(d), the computational cost of HDAF and finite difference methods is comparable. This is because the dominant cost arises from the

bond dimension of the MPS states, which remains similar between both approaches. Given the higher accuracy of HDAF-based operators and their similar computational overhead, we will from now on adopt HDAF spectral methods exclusively and discard finite difference approximations from the remainder of this study.

All evolution algorithms involve a trade-off between accuracy (*i.e.*, the approximation order) and computational cost, as illustrated in Figures 13.1(b)–(d). In the HDAF-based implementation, all the methods converge in terms of the function error ε with respect to the number of qubits. For intermediate and large time steps, the error exhibits an algebraic dependence on Δt, in agreement with theoretical predictions. However, for very small Δt, the convergence is limited by the numerical accuracy of the MPS representation.

Among the schemes tested, the Arnoldi method with $n_v = 10$ Krylov vectors achieves the best accuracy, allowing for larger time steps due to its higher-order convergence. The fourth-order Runge-Kutta, Arnoldi with $n_v = 5$, and split-step methods yield comparable accuracy over a wide range of Δt values. In contrast, the lower-order Euler method and the Crank–Nicolson implementation (using conjugate gradient inversion) are limited by their lower order and convergence behavior, preventing them from reaching beyond the intrinsic truncation error of the MPS representation.

To determine the best balance in cost and accuracy, the error ε must be evaluated along with the run time, which is shown in Figure 13.1(d). Although the Arnoldi method with $n_v = 10$ provides the best accuracy, its run time is approximately two orders of magnitude higher than that of the split-step method. As a result, even when using smaller time steps, the split-step method offers a more favorable cost-accuracy trade-off. Furthermore, in many practical applications, extremely high accuracy is not required, and larger values of Δt are acceptable. For these reasons, the split-step method using the HDAF-based approximation of the free propagator is selected as the optimal approach to simulate the expansion problem.

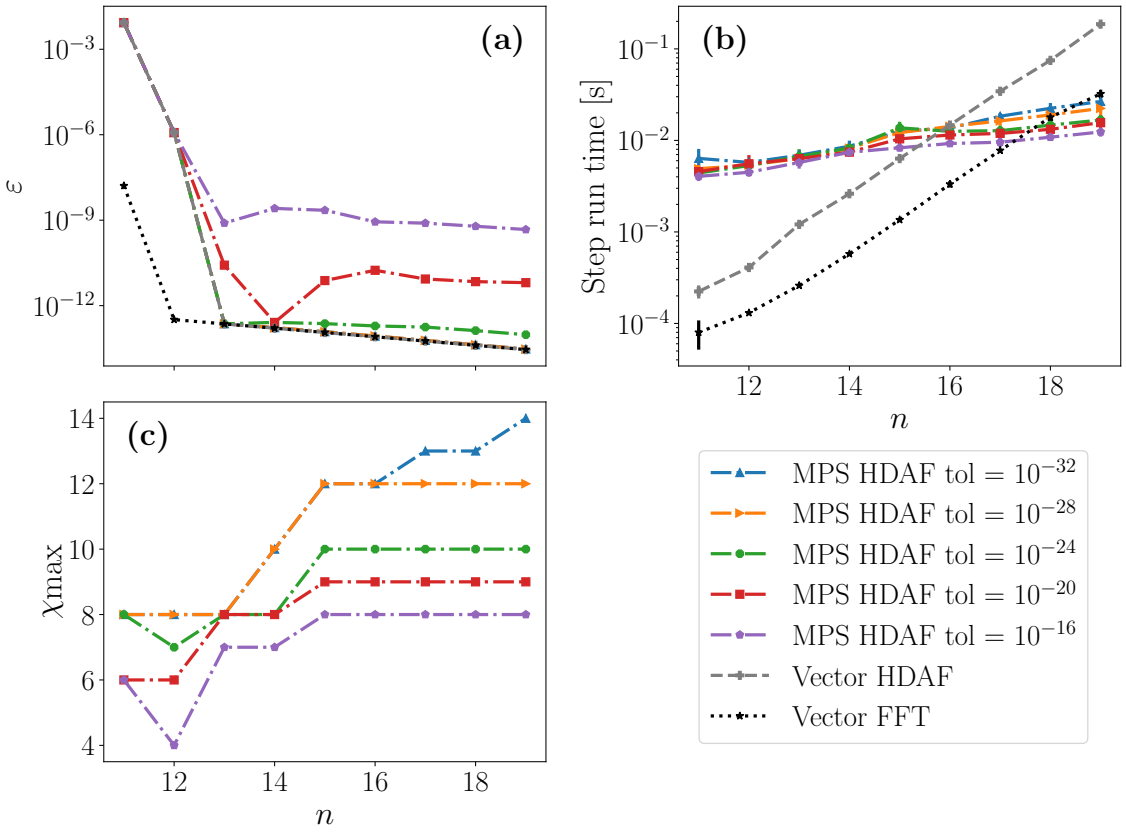


Figure 13.2: Number of qubits n scaling of the split-step one-step evolution, comparing vector-based (HDAF and FFT) and different tolerance MPS-based (HDAF) methods. The time step is fixed, $\Delta t = 10^{-4}$. (a) Error ε. (b) Run time. (c) Maximum bond dimension $\chi_{\max}$. The run time is averaged over ten runs.

To assess the efficiency of MPS methods, we compare the performance of the HDAF split-step method against state-of-the-art vector-based implementations of the split-step method using the fast Fourier transform (FFT). Figure 13.2 analyzes the computational scaling of our three figures of merit for MPS and vector representations with increasing spatial resolution. The MPS implementation is evaluated for various truncation tolerances in the SVD and simplifications introduced by the finite-precision algebra. The simplification accuracy is measured by the 2-norm difference $\|\psi - \phi\|_2^2$, where $|\psi\rangle$ is the original MPS with bond dimension χ_ψ, and $|\phi\rangle \in \mathrm{MPS}_{\chi_\phi}$

is its projection onto a reduced subspace with bond dimension $\chi_\phi < \chi_\psi$.

Figure 13.2(a) shows that the SVD truncation tolerance in the MPS algorithm is the dominant factor in the total error of the split-step method. To achieve a numerical precision comparable to that of the vector-based implementation, tolerances on the order of $O(10^{-28})$ or lower are required.

In terms of computational performance, Figure 13.2(b) shows that FFT-based methods are faster than our vector HDAF implementation. However, the MPS implementations demonstrate asymptotically better scaling than both, as it avoids the exponential memory requirements inherent to vector representations. Although the number of spatial points increases exponentially with the number of qubits, the MPS approach maintains a manageable computational cost. Additionally, the MPS run time is largely insensitive to the chosen SVD tolerance because the resulting bond dimensions remain similar across truncations. For tolerances below 10^{-28}, the bond dimension is typically overestimated, which leads us to adopt this value as a reasonable balance in our simulations. For internal MPS simplifications, we continue to use a tolerance near machine precision.

Figure 13.3 further supports this choice by illustrating how the error ε scales with the time step Δt. When using SVD tolerances below 10^{-28}, the MPS-based method exhibits similar behavior to the vector implementation. As the SVD tolerance increases, however, the MPS truncation error begins to dominate for small Δt, limiting the achievable accuracy regardless of the integration method. For larger time steps, the split-step error dominates, which permits the use of more relaxed tolerances without significant loss of precision.

13.2 Quantum Quench Evolution

Having introduced the MPS-based algorithms for solving time-dependent PDEs, we now illustrate their applicability to a physically motivated problem. As presented

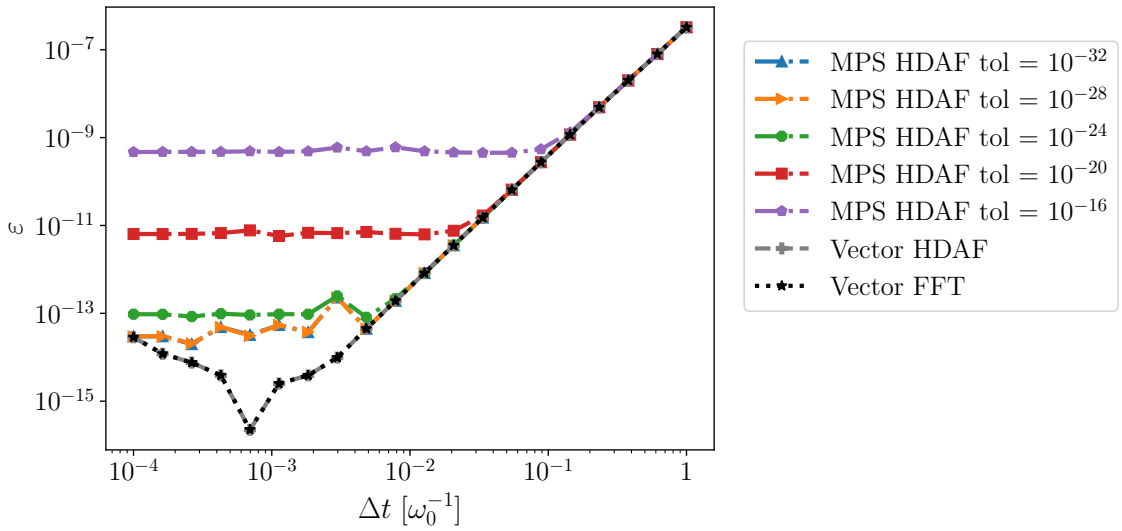


Figure 13.3: Error ε scaling with time step Δt for the split-step one-step evolution, comparing vector-based (HDAF and FFT) and different tolerance MPS-based (HDAF) methods. Spatial discretization $n = 20$.

in Chapter 11, the scenario involves the expansion of a quantum particle initially confined in a harmonic trap. Two versions of the potential are considered: a purely harmonic potential, as in Section 13.1, and a double-well configuration discussed in Section 13.2.2. This section focuses on the numerical methods that exhibit the best trade-off between performance and accuracy, namely the HDAF split-step method implemented in the MPS/QTT framework and the FFT split-step in the vector-based case.

13.2.1 Harmonic Expansion

The first set of simulations revisits the harmonic quench scenario from Section 13.1, in which a particle expands under a harmonic potential (11.2) whose frequency is suddenly reduced by a factor of 100. This results in a 100-fold expansion of the particle's wavefunction. To capture this evolution accurately, we use a spatial discretization with $2^{20} = 1,048,576$ points, corresponding to $n = 20$ qubits, over the

interval $x \in [-L/2, L/2)$ with $L = 16\sigma_{\max}$.

The simulation begins with the ground state of a harmonic oscillator with frequency $\omega_0 = 1$; a real-valued Gaussian wavefunction. At time $t = 0$, the potential is suddenly quenched into a much weaker trap with frequency $\omega_H = 0.01$. This induces a rapid expansion of the wavepacket, which continues until $0.5\pi/\omega_H$, at which point the solution (11.4) reaches its maximum width.

To study the methods in a range of temporal resolutions, we consider three different time step values: $\Delta t = 0.01$, 0.1, and 1. The simulation is run up to a final time $t_f = 158$, which is slightly above the expansion period and aligns with multiples of all time step values. The spatial domain remains fixed at $x \in [-L/2, L/2)$ with $L = 16\sigma_{\max}$, as in Section 13.1.

Figures 13.4(a)–(b) show the evolution of the function error ε (13.1) over time for the split-step method, comparing the MPS-HDAF and vector-FFT implementations, respectively. In both cases, the error exhibits algebraic growth with time, with nearly identical coefficients (see Appendix C). This behavior is attributed to the Störmer–Verlet integration scheme employed by the split-step method. The observed error scaling is consistent with the linear accumulation typical of symplectic integrators applied to periodic Hamiltonian systems [315]. This contrasts with the quadratic error growth often seen in non-symplectic methods, such as Runge–Kutta schemes [316].

The saturation of the global error may be explained by the energy-conserving properties of symplectic integrators. These methods are known to bound energy errors over arbitrarily long simulation times [311], which in turn suggests bounded phase-space errors for conservative systems exhibiting periodic motion. Such properties make the split-step method particularly well suited for long-time simulations of isolated quantum systems.

Figures 13.4(c)–(d) present the run time as a function of the simulated time. In the case of the vector-based FFT implementation, the cost per time step remains

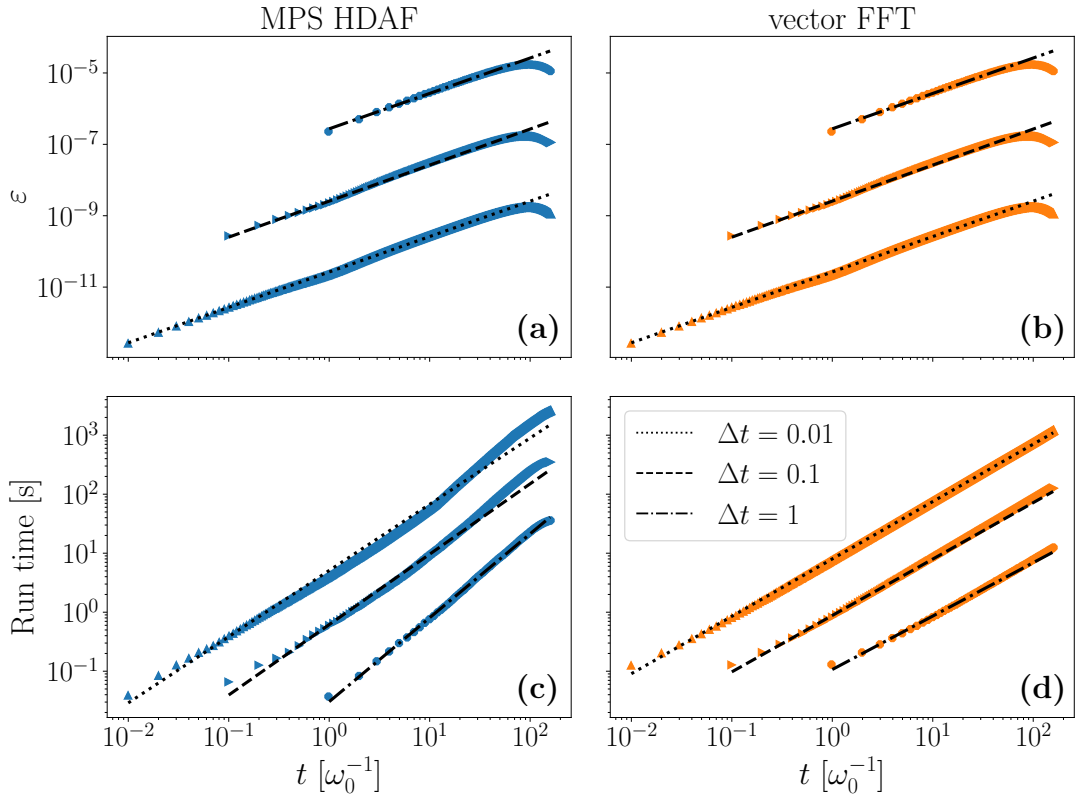


Figure 13.4: Particle expansion in a harmonic potential (11.2) with $\omega_H/\omega_0 = 10^{-2}$, $t_f = 158$ and $n = 20$ for the MPS split-step methods with HDAF differentiation and a state-of-the-art FFT split-step. Figures (a)-(c) show the error scaling with the evolution time, and figures (d)-(f) show the run time scaling with the evolution time.

constant, leading to a linear run time growth. For the MPS simulation, the run time scales slightly superlinearly. This is primarily due to the gradual increase in bond dimension as the wavefunction expands, which increases the memory requirements and the computational cost of MPS and MPO-MPS operations over time.

The run time of MPS-based algorithms is closely related to the bond dimension χ of the quantum state at each time step. Figure 13.5(b) shows the maximum bond dimension $\chi_{\max}$ throughout evolution. The three simulations exhibit a similar trend:

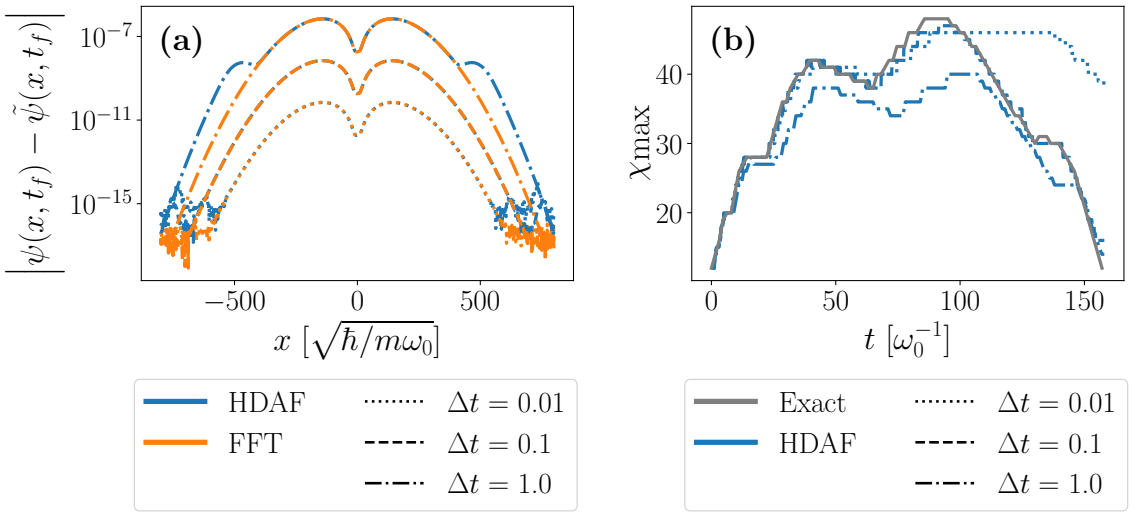


Figure 13.5: Particle expansion in a harmonic potential (11.2) with $\omega_H/\omega_0 = 10^{-2}$, $t_f = 158$ and $n = 20$ for the split-step methods with HDAF differentiation and a state-of-the-art FFT split-step. (a) Pointwise error $|\psi(x,t) - \tilde{\psi}(x,t)|$ of the maximum width solution approximated by the methods $\tilde{\psi}(x,t)$ with respect to the analytic solution $\psi(x,t)$ (11.4). (b) Maximum bond dimension $\chi_{\max}$ for each time step.

the bond dimension increases during the simulation and reaches its peak when the absolute value of the phase $\beta(t)$ in the analytic solution (11.4) is largest. Lower bond dimensions are observed at the beginning and end of the simulation, where the wavefunction corresponds to a real-valued Gaussian.

This growth in χ can be attributed to the chirping of the wavefunction, a result of the wavepacket's acceleration during expansion, which introduces complex phases and effectively increases the entanglement encoded in the MPS. This behavior is inherent to the physical setting and does not depend on the numerical scheme used, besides truncation tolerance.

Interestingly, we observe that the bond dimension of the exact solution serves as an approximate upper bound for the bond dimensions obtained in the numerical simulations. This suggests that the errors introduced by the numerical methods tend to deviate the solution not in a random direction but towards a more efficient

MPS representation. This observation is consistent with the nature of HDAF-based approximations which act as low-pass filters, attenuating the high-frequency response of differential operators. In this sense, HDAF operators appear to be well suited to the MPS/QTT framework, relying on approximations that are a good fit for their formalism.

Figure 13.5(a) displays the pointwise error of the numerically computed wavefunction at its maximum width, defined as $|\psi(x,t) - \tilde{\psi}(x,t)|$, where $\psi(x,t)$ is the analytic solution (11.4) and $\tilde{\psi}(x,t)$ is the result of the numerical method. Both the MPS-HDAF and vector-FFT implementations of the split-step method exhibit similar error profiles. The most significant discrepancies appear at the edges of the spatial domain and for the larger time step, possibly due to errors associated with the MPS representation.

To examine the evolution of the particle, let us focus on one simulation. Figure 13.6 illustrates the time evolution of the wavefunction computed with the MPS-HDAF split-step method for a time step $\Delta t = 0.1$. As predicted by the analytic solution (11.4), the weaker harmonic potential (Figure 13.6(a)) induces an expansion of the particle, as observed in Figure 13.6(b). This expansion continues until the final time t_f, where the wavefunction is close to its maximum spatial width.

13.2.2 Double well potential

The calibration of the simulation parameters in Section 13.2.1 provides a reference to select the appropriate time steps, spatial discretization, and truncation tolerances for the split-step algorithm. We now consider a more complex and physically relevant scenario: the expansion of a nanoparticle in an anharmonic double-well potential. This problem serves as an analogue to a "double-slit" experiment, where the particle evolves into a coherent superposition at both halves of the trapping potential. Such configurations are commonly studied in levitodynamics [274, 282–284].

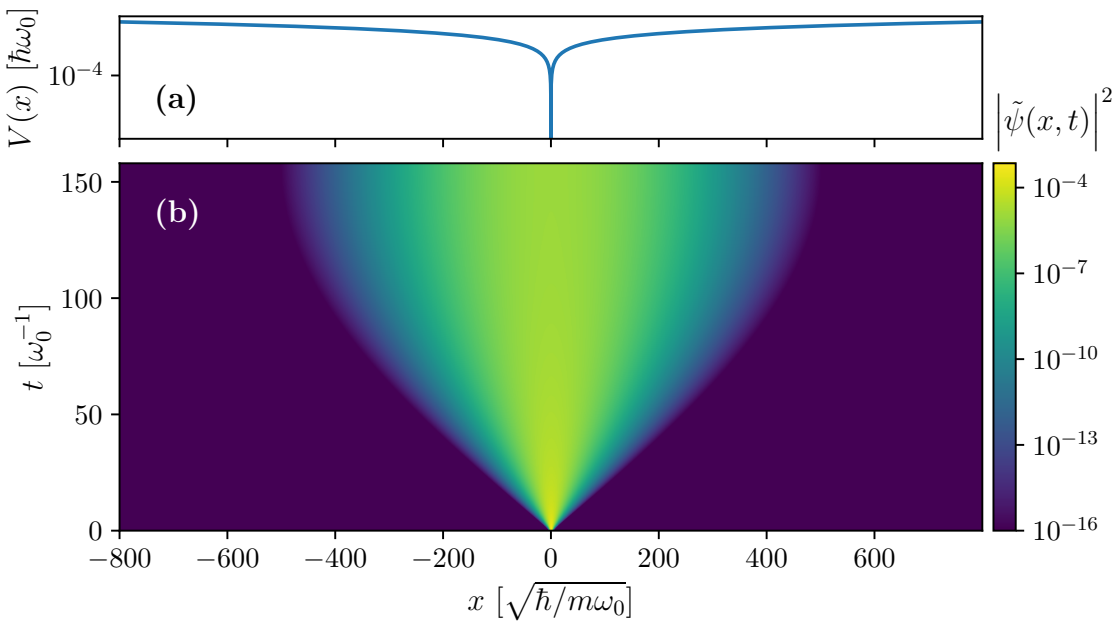


Figure 13.6: Particle expansion in a harmonic oscillator potential (11.2) with $\omega_H/\omega_0 = 10^{-2}$, with $t_f = 158$, $\Delta t = 0.1$ and $n = 20$ for the MPS split-step method with HDAF differentiation. (a) Potential $V(x)$ (11.2). (b) Wavefunction density $|\tilde{\psi}(x,t)|^2$.

In this simulation, the double-well potential is modeled as an open harmonic trap perturbed by a repulsive Gaussian term, as defined in (11.7), with parameters $u = 1$, $\sigma = 1$, and a harmonic frequency ω_H that is 100 times smaller than the initial confinement frequency ω_0, as in the previous simulations. The resulting potential is illustrated in Figure 13.7(a). The harmonic component weakens the overall confinement, promoting expansion, while the Gaussian barrier at $x = 0$ splits the trap into two wells.

The simulation runs until the final time $t_f = 1000$, exceeding the characteristic period $T = \pi/\omega_H$ of the unperturbed harmonic expansion. This extended timescale enables the observation of multiple cycles of expansion and contraction, revealing wavefunction collapse and revival. Figure 13.7 presents the simulation results.

The evolution of the probability density $|\tilde{\psi}(x,t)|^2$, shown in Figure 13.7(b), re-

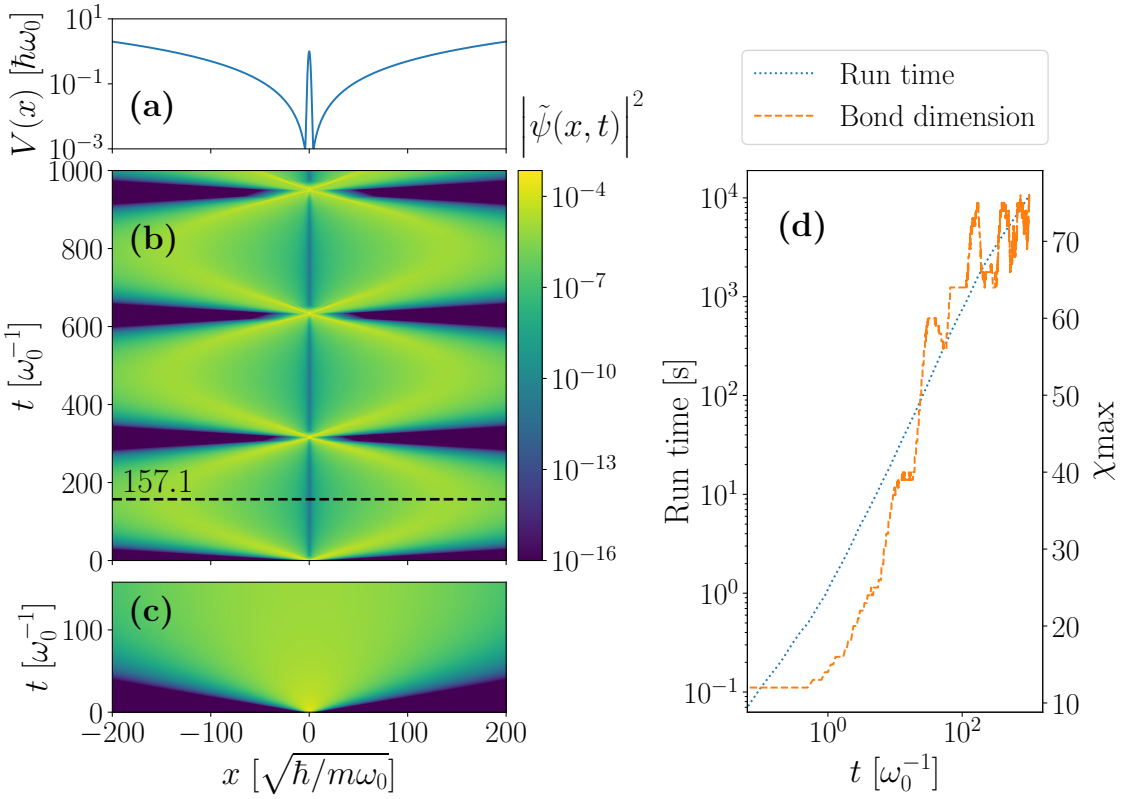


Figure 13.7: Particle expansion in a double well potential (11.7) with $\omega_H/\omega_0 = 10^{-2}$, $u = 1$ and $\sigma = 1$, with $t_f = 1000$, $\Delta t = 0.1$ and $n = 20$ for the MPS split-step method with HDAF differentiation. (a) Potential $V(x)$ 11.7. (b) Wavefunction density $|\tilde{\psi}(x,t)|^2$. (c) Wavefunction density $|\tilde{\psi}(x,t)|^2$ for the harmonic potential (11.2). (d) Run time and maximum bond dimension $\chi_{\max}$.

flects the interplay between the harmonic and Gaussian components in the potential. The harmonic potential drives the periodic spreading and refocusing of the wavepacket, with maximal expansion occurring around $0.5\pi/\omega_H \approx 157.1$. Meanwhile, the repulsive Gaussian barrier induces spatial separation in the density, effectively dividing the wavefunction into two peaks propagating in opposite directions.

This anharmonicity also impacts the complexity of the MPS representation. Unlike the purely harmonic case, where the bond dimension exhibits cyclic growth and reduction, the presence of the Gaussian term causes the bond dimension to saturate

over time, as shown in Figure 13.7(d). This saturation leads to a decrease in the scaling of the run time as the system evolves.

Chapter 14

Conclusions

This work has introduced a novel encoding of differential operators using Hermite Distributed Approximating Functionals (HDAFs) within the MPS/QTT framework. This representation achieves exponential accuracy while maintaining favorable scaling in computational resources.

The proposed HDAF-based encoding enables the development of quantum-inspired time evolution algorithms to solve time-dependent PDEs. We implemented and evaluated explicit and implicit Runge–Kutta methods, a restarted Arnoldi iteration scheme, and a split-step integrator. Notably, the split-step method benefits from the efficient approximation of the free propagator in the coordinate space, which standard finite differences cannot approximate.

The implementation of these time evolution operators with HDAF discretization outperforms their finite-difference counterparts in accuracy, while maintaining comparable computational cost. Among the methods considered, the split-step scheme provides the best trade-off in accuracy and efficiency. Furthermore, HDAF-based time evolution algorithms are competitive with state-of-the-art vector implementations, offering exponential compression of function representations with moderate computational overhead.

The particle expansion scenario serves as a demanding benchmark for the meth-

ods. It presents challenges for MPS-based solvers because of the emergence of chirping and rapid phase oscillations. However, MPO-MPS algorithms yield accurate results with manageable bond dimensions and adequate run times.

Note that while the performance of MPS and FFT-based implementations is similar for one-dimensional problems, MPS methods offer an advantage in higher-dimensional settings because of their efficient scaling with grid size. Future work will focus on extending these algorithms to multidimensional domains and improving their performance through C/C++ backends and low-level optimizations, thereby reducing the runtime overheads of the current Python implementations.

The present implementation is based on the SElf-Explaining Matrix Product State (SeeMPS) library for Python [5].

Appendix B

One-step scaling

Figures 13.1(a)-(b) show a fit for the error ε with Δt for the methods in Section 12.3, for the finite difference and HDAF derivative approximation of the derivative, respectively. Tables B.1 and B.2 contain the concrete numerical data of the fit $\varepsilon = C\Delta t^m$. We use a piecewise linear fit and show the data for larger Δt, which are not limited by the MPS accuracy.

Method	C	m
Euler	2.29×10^{-2}	1.99
Improved Euler	3.37×10^{-2}	3.14
Runge-Kutta	2.20×10^{-2}	4.18
Crank-Nicolson	8.87×10^{-1}	3.44
Arnoldi $n_v = 5$	2.04×10^{-3}	3.79
Arnoldi $n_v = 10$	7.88×10^{-4}	4.45

Table B.1: Function error ε (13.1) fit, $\varepsilon = C\Delta t^m$, for each method for a $n = 18$ discretization and finite difference approximation of the derivative.

Method	C	m
Euler	2.13×10^{-2}	2.00
Improved Euler	2.88×10^{-2}	3.00
Runge-Kutta	2.21×10^{-2}	4.97
Crank-Nicolson	2.52×10^{-1}	3.73
Arnoldi $n_v = 5$	1.02×10^{-3}	4.92
Arnoldi $n_v = 10$	1.16×10^{-5}	9.11
Split-step	4.94×10^{-7}	2.96

Table B.2: Function error ε (13.1) fit, $\varepsilon = C\Delta t^m$, for each method for a $n = 18$ discretization and HDAF approximation of the derivative.

Appendix C

Harmonic quantum quench evolution scaling

Figure 13.4 shows the error ε scaling and run time of the harmonic quantum quench evolution with time. Tables C.1-C.4 contain the concrete numerical data of the fits $\varepsilon = Ct^m$ and $T = Ct^m$, where T is the run time.

Δt	C	m
0.01	2.65×10^{-11}	0.99
0.1	2.56×10^{-9}	1.01
1.0	2.72×10^{-7}	0.99

Table C.1: Function error ε (13.1) fit, $\varepsilon = Ct^m$, for the split-step HDAF MPS method, using different step-sizes Δt.

Δt	C	m
0.01	2.65×10^{-11}	0.99
0.1	2.56×10^{-9}	1.01
1.0	2.72×10^{-7}	0.99

Table C.2: Function error ε (13.1) fit, $\varepsilon = Ct^m$, for the split-step FFT method, using different step-sizes Δt.

Δt	C	m
0.01	5.10	1.22
0.1	0.62	1.19
1.0	0.03	1.42

Table C.3: Run time fit, $T = Ct^m$, for the split-step HDAF MPS method, using different step-sizes Δt.

Δt	C	m
0.01	7.99	0.97
0.1	0.88	0.95
1.0	0.11	0.91

Table C.4: Run time fit, $T = Ct^m$, for the split-step FFT method, using different step-sizes Δt.

Part IV

Closing

Chapter 15

Conclusions

"We shall not cease from exploration, and the end of all our exploring will be to arrive where we started and know the place for the first time."

T. S. Eliot

This thesis has explored two main research directions: stochastic optimization for variational quantum algorithms (Part II) and solving partial differential equations (PDEs) using matrix product states (MPS) (Part III). In this chapter, we summarize the key findings, reflecting on the specific objectives outlined in Section 1.3: Objectives and methodology .

1. Optimization methods for VQAs

The first part of this work focused on optimizing variational quantum algorithms (VQAs), which are among the most promising candidates for quantum advantage in the current noisy intermediate-scale quantum (NISQ) era. These algorithms rely on a combination of classical optimization and shallow quantum circuits, making them robust against certain types of noise. The effectiveness of VQAs is significantly impacted by the election of the classical optimizer, which must perform in complex, high-dimensional landscapes with a high probability of bar-

ren plateaus. For this reason, we benchmark the family of Simultaneous Perturbation Stochastic Approximation optimizers, which are especially suitable for high-dimensional problems and have convergence proofs even in the presence of noise.

1.1. Review existing SPSA methods for quantum applications

In Chapter 7, we present a review on the theory of the first-order SPSA, second-order 2SPSA, and quantum natural QN-SPSA methods, as well as the first-order complex-variable CSPSA, and improvements typically used in the context of VQAs.

1.2. Introduce complex-variable extensions to SPSA methods

In Chapter 7 we also proposed complex-variable variants of the second-order 2CSPSA, and quantum natural QN-CSPSA methods. Additionally, we propose new improvements, including (i) a new post-processing formula for the Hessian and (ii) scalar versions of the second-order and quantum natural methods that avoid matrix inversion, effectively reducing the classical complexity from $O(p^3)$ to $O(p)$, for problems with p parameters.

In all of these methods, the number of evaluations is constant, varying for each method, and is independent of the objective function.

1.3. Benchmark the methods on representative VQAs problems

In Chapter 8 we benchmarked the methods against 3 representative VQAs: (i) Variational quantum eigensolver for a 10-qubit Heisenberg Hamiltonian, (ii) quantum control for 5-qubit pure states and (iii) quantum state estimation for 6-qubit pure states.

Although the election of gain coefficients is central to achieving good behavior of the optimizers, it involves a complicated optimization problem that is generally unfeasible to solve. Therefore, one usually resorts to coefficient sets that have proven useful in various applications. We have tested 3 standard

sets here: the standard, asymptotic, and static sets described in Chapter 8.

We compared the optimization algorithms in two groups: (a) vanilla and (b) improved versions, where the improvements consist of blocking, resampling, post-processing, and scalar preconditioning. Blocking and resampling increase the number of quantum evaluations required per iteration.

The best-performing algorithm across all tasks was the first-order CSPSA. In general, SPSA and CSPSA benefited from improvements (b), except for quantum state estimation, where the performance was indistinguishable from the vanilla methods.

Generally, vanilla second-order algorithms lead to lower performance than vanilla quantum natural algorithms. This is mitigated by blocking and resampling, making improved second-order algorithms perform similarly to improved quantum natural algorithms. Furthermore, complex-variable algorithms perform better than their real counterparts, although the difference may be statistically insignificant in some cases.

For some applications, it may be possible that the inversion of the Hessian matrix exhibits singularities. This is the case of quantum state estimation, where the analytical Hessian vanishes exactly. In this case, Hessian postprocessing (7.44) ensures that the preconditioning matrix is proportional to the identity. This leads to second-order methods falling back to the behavior of first-order methods, albeit with possibly suboptimal gain coefficients.

We observed that the performance difference between quantum natural and first-order algorithms narrows as the number of qubits increases. This may indicate that quantum natural methods could outperform first-order methods for problems in a larger number of qubits. Moreover, quantum natural methods require a less exhaustive search for adequate gain parameters. However, this advantage is obtained at the cost of increasing the number of measurements and the classical computational complexity. In this context, our scalar

formulation for quantum natural methods might be a good alternative since, according to our results, they offer comparable performance at a reduced classical cost.

1.4. Provide practical guidelines for applying SPSA methods in the context of VQAs

We give these practical guidelines in Chapter 9. In summary, first-order vanilla algorithms are efficient and reliable options for the most general case. If higher accuracy is needed, improved first-order algorithms are the straight choice. However, first-order methods may require careful calibration of the gain parameters, in which case quantum natural algorithms are a suitable alternative at the cost of extra fidelity measurements. In addition, scalar quantum natural algorithms show promising results for many qubits, while second-order algorithms do not exhibit a comparative advantage.

2. Solving PDEs with matrix product states

The second part of this thesis concerns the development of quantum-inspired numerical methods to efficiently solve time-dependent partial differential equations. The objectives and associated results are as follows.

2.1. Review the usage of Hermite Distributed Approximating Functionals (HDAF) to approximate differential operators

In Chapter 12, we provided a review on Hermite Distributed Approximating Functionals, covering the theoretical foundations and metaheuristics to construct the identity, the n-th derivative, and the free propagator HDAFs.

2.2. Extend the MPS framework to encode operators within the HDAF formulation

By integrating Hermite Distributed Approximating Functionals (HDAF) within the MPS framework as a weighted combination of shift operators, we developed an efficient encoding to approximate differential operators with

exponential accuracy, where the maximum bond dimension saturates with the number of qubits in the discretization. The resulting operators exhibit a Gaussian envelope, ensuring their spatial localization up to numerical resolution. In practice, these operators act as bandwidth-limited approximations to their global counterparts, making them a natural fit for MPS techniques, which efficiently represent bandwidth-limited functions.

2.3. Develop numerical integrators based on the operators provided by the HDAF-MPS encoding

The HDAF encoding of differential operators enabled the design of quantum-inspired time evolution algorithms, including explicit and implicit Runge-Kutta methods, restarted Arnoldi iteration, and split-step methods. These HDAF-based methods consistently outperformed finite-difference implementations in accuracy while maintaining comparable or better computational costs.

2.4. Benchmark the new techniques in physically relevant scenarios

The methods were validated against the simulation of particle expansion in two quantum quench scenarios: (i) a harmonic potential, where an analytic solution allowed measuring the accuracy of the methods, and (ii) a double-well potential, which induced wave function separation, a relevant phenomenon in experimental optomechanics.

MPS-HDAF methods provided accurate results, even in the presence of wavefunction chirping, which is a known challenge for MPS-based methods.

Among all the quantum-inspired approaches tested, the HDAF-based split-step method displayed the best balance in cost and accuracy. A key advantage of this method is the direct space HDAF representation of the free-propagator, which eliminates the need for Fourier transforms that traditional implementations require.

2.5. Compare the performance of the proposed methods with traditional numerical approaches

The MPS-HDAF algorithms achieved accuracy levels comparable to traditional FFT-based implementations. In particular, the MPS-HDAF split-step also achieved runtimes comparable to those of FFT split-step implementations while offering an advantage in scalability, especially with respect to storage resources.

15.1 Future directions

The work presented in this thesis can be extended in several directions. In the context of stochastic optimization for VQAs, we only considered the effects of statistical noise from quantum measurements. However, real devices introduce additional errors due to decoherence, gate imperfections, and readout noise. A natural next step would be to test the performance of stochastic optimization in practical scenarios, either by using realistic noise models or by running the algorithms on real quantum hardware.

Further improvements to the SPSA algorithm are also worth exploring. One possibility is to use adaptive learning rate techniques, where hyperparameters are adjusted during the optimization based on the observed behavior of the cost function. Another direction is to include momentum in the update rules, following the successful example of optimizers like Adam in machine learning. These ideas could help reduce the need for manual tuning and improve convergence in flat or noisy regions.

We are currently working on variational algorithms for (i) synchronizing two quantum computers, and (ii) self-guided quantum tomography of sparse states. In the first case, we are minimizing cross-correlations between measurement outcomes transmitted over a classical network. In the second case, we minimize the direct fidelity estimation to reconstruct pure or mixed sparse quantum states. Both algo-

rithms are driven by the CSPSA optimization method.

With regard to the second part of this thesis, MPS-HDAF methods could be extended to more general problems. For example, we have observed good stability when solving the nonlinear Burgers' equation, which develops shocks and requires high spatial resolution with good control of the Gibbs phenomenon. These techniques should also adapt efficiently to high-dimensional problems, or to quantum systems with dissipation. We are particularly interested in the latter case, where the MPS-HDAF split-step method can be adapted to solve the Lindblad equation. We are also interested in using the MPS-HDAF framework to study quantum chemistry problems in first quantization.

Finally, in the intersection of VQA and MPS methods, we are developing a self-guided quantum tomography method for MPS. The goal is to improve convergence and avoid barren plateaus in quantum state tomography, while enabling scalable methods that remain efficient for systems with a large number of qubits. Preliminary results suggest that adaptive learning rates for CSPSA can be especially helpful in this setting.

Chapter 16

List of activities

This last chapter is a binnacle of academic activities performed during the Ph.D. program, including publications, presentations, outreach, and other relevant activities.

Accepted publications

2023

- J. Gidi *et al.* "Stochastic optimization algorithms for quantum applications". In: *Phys. Rev. A* 108 (3 Sept. 2023), p. 032409. DOI: 10.1103/PhysRevA.108.032409. URL: https://link.aps.org/doi/10.1103/PhysRevA.108.032409

- Hugo A. Carril *et al.* "Formation of multiple BGK-like structures in the time-asymptotic state of collisionless Vlasov-Poisson plasmas". In: *Phys. Rev. E* 107 (6 June 2023), p. 065203. DOI: 10.1103/PhysRevE.107.065203. URL: https://link.aps.org/doi/10.1103/PhysRevE.107.065203

Submitted publications

2024

- Jorge Gidi *et al. Pseudospectral method for solving PDEs using Matrix Product States*. 2024. DOI: 10.48550/ARXIV.2409.02916. URL: https://arxiv.org/abs/2409.02916

Presentations

2020

- **Talk** presented online at te **XXII Simposio Chileno de Física**, from november 24 to 26. Title: "Inestabilidad transversal de modos tipo BGK formados por distribuciones tipo *bump-on-tail*".

2021

- **Poster** presented online at the **XII Conferencia Latinoamericana de Geofísica Espacial, COLAGE 2021**, from November 22 to 26. Title: "Transverse Instability of Self-consistently Formed BGK modes".

2022

- **Talk** presented to the "Information Coding Group" on 23 May in Linköping, Sweden. Title: "Stochastic Optimization Algorithms for Quantum Applications".

- **Talk** presented at the **Workshop MIRO 2022**, from october 5 to 7 in Santiago, Chile. Title: "Algoritmos de optimización estocástica para aplicaciones cuánticas".

- **Poster** presented at the **XXIII Simposio Chileno de Física**, from november 22 to 24 in Valparaíso, Chile. Title: "Stochastic Optimization Methods for Quantum Applications".

2023

- **Member of the organizing team** of the **Primera Escuela de Computación Cuántica MIRO-UdeC**, during January 9 to 13 in Concepción, Chile.
- Introductory **class** to programming in the Python language, at the **Primera Escuela de Computación Cuántica MIRO-UdeC**.
- **Poster** presented at the **Superconducting Qubits and Algorithms Conference**, held from August 29 to September 1 in Munich, Germany. Title: "Stochastic Optimization on Complex Variables for Variational Quantum Algorithms".
- **Talk** presented at the **Workshop MIRO 2023**, held from October 10 to 13 in Santiago, Chile. Title: "Estimation of Mixed Quantum States using Relative Entropy Measurements".

2024

- **Staff member** in the **Primer Congreso Chileno de Física de Plasmas**, held from November 20 to 23 in Concepción, Chile.
- **Member of the organizing team** of the **Escuela de Computación Cuántica MIRO-PUC 2024**, held from January 8 to 12 in Santiago, Chile. My participation was online.
- **Talk** presented online at the London Research Seminars, October 17th. Title: "Pseudospectral method for solving PDE's with Matrix Product States".

- **Talk** presented at the **Workshop MIRO 2024**, held from November 11 to 15 in Santiago, Chile. Title: "Pseudospectral method for solving PDEs with Matrix Product states".

- **Talk** presented at the **XXIV Simposio Chileno de Física**, held from November 20 to 22 in Temuco, Chile. Title: "Operadores diferenciales con precision pseudoespectral para MPS".

- **Poster** presented at the **Quantum Optics X**, held from December 9 to 13 in Puerto Varas, Chile. Title: "Pseudospectral method for solving Partial Differential Equations using Matrix Product States".

2025

- **Member of the organizing team** of the **Tercera Escuela de Computación Cuántica 2025**, held from January 6 to 10 in Santiago, Chile.

- **Talk** presented at the **Tercera Escuela de Computación Cuantica 2025**, held from January 6 to 10 in Santiago, Chile. Title: "RSA and Shor's algorithm".

- **Talk** presented to the Quantum Development Information and Technology (QuDIT) group of the Pontificia Universidad Católica de Chile. January 13 in Santiago, Chile. Title: "Matrix Product States".

Other relevant Activities and Outreach

2021

- Part of the first-place winning team at the **Quantum Hackathon CIC-IPN Mexico**, held online from October 24 to 27. Project: "Threerra" https://github.com/jgidi/threerra.

2022

- Part of the second-place winning team at the **Open Hackathon - QHACK 2022**, held online from February 14 to 25. As a consequence, the team was invited to Helsinki to meet at the facilities of the VTT Technical Research Centre of Finland, and IQM. Project: "VarQuS: Time Evolution of Quantum Systems using Variational Quantum Algorithms" https://github.com/lezav/variational_quantum_simulation.

2023

- Part of the fifth-place winning team in the **Coding Challenges - QHACK 2023**, held online from February 16 to 21.
- **Programmer** in the **QuantumBuilder** project for the Millenium Institute for Research in Optics (MIRO). Project consisting in an extension of the Qiskit Blocks game to teach quantum computing in a minecraft-like world.
- **Instructor** in the **Taller de Fenómenos Ópticos**, on August 2 and 4, to teach basic optics phenomema to children in Concepción,Chile.
- **Instructor** at the **Festival de las Ciencias 2023**, on October 3 in Los Ángeles, Chile.
- **Talk** presented at the **Olimpiadas de Física 2023**, on October 14 in Concepción, Chile. Title: "Computación Cuántica en la era actual".
- **Doctoral stay** in Madrid, from December 3 2023 to August 24 2024, at the Fundamental Physics Institute (IFF-CSIC) under the guidance of Juan José García Ripoll.
- **Instructor** in the activity **Aprendiendo computación cuántica jugando con QiskitBlocks**, on December 20 in Madrid, Spain.

2024

- Part of the first-place winning team in the **Open Hackathon - QHACK 2024**, category "The sound of Silence", held online from February 16 to 22. Project: "Variational Quantum Algorithms in First Quantization" https://github.com/jgidi/VFA-Schrodinger-like-equations.
- Part of the fifth-place winning team in the **Coding Challenges - QHACK 2023**, held online from February 12 to 16.
- **Monitor** at the **Puertas Abiertas UdeC**, October 9 and 10 in Concepción, Chile.
- **Instructor** in the activity **QuantumBuilder**, October 16 in Coronel, Chile.
- **Instructor** in the activity **QuantumBuilder**, November 6 and 8 in Santiago, Chile.

2025

- **Instructor** in the activity **Día de la Astronomía UdeC**, March 20 in Concepción, Chile.

Bibliography

[1] Jeff Bezanson *et al.* "Julia: A Fresh Approach to Numerical Computing". In: *SIAM Review* 59.1 (2017), pp. 65–98. DOI: 10.1137/141000671. URL: https://doi.org/10.1137/141000671 (cit. on pp. 5, 106).

[2] Jorge Gidi. *jgidi/ComplexSPSA.jl: v0.2.1*. Zenodo. https://doi.org/10.5281/zenodo.7613310. 2023. DOI: 10.5281/ZENODO.7229868. URL: https://zenodo.org/record/7229868 (cit. on pp. 5, 106).

[3] Xiu-Zhe Luo *et al.* "Yao.jl: Extensible, Efficient Framework for Quantum Algorithm Design". In: *Quantum* 4 (Oct. 2020), p. 341. DOI: 10.22331/q-2020-10-11-341. URL: https://doi.org/10.22331/q-2020-10-11-341 (cit. on p. 5).

[4] Guido Van Rossum and Fred L Drake Jr. *Python tutorial*. Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995 (cit. on p. 6).

[5] Juan José García Ripoll *et al. SeeMPS: SElf-Explaining Matrix-Product-State library*. https://github.com/juanjosegarciaripoll/seemps2/tree/aa187705aa96d1569d479700b6bfe29fdc7078c6. 2025 (cit. on pp. 6, 184).

[6] Max Planck. "Ueber das Gesetz der Energieverteilung im Normalspectrum". In: *Annalen der Physik* 309.3 (1901), pp. 553–563. DOI: https://doi.org/10.1002/andp.19013090310. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19013090310. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19013090310 (cit. on p. 10).

[7] P. Lenard. "Ueber die lichtelektrische Wirkung". In: *Annalen der Physik* 313.5 (1902), pp. 149–198. DOI: https://doi.org/10.1002/andp.19023130510. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19023130510. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19023130510 (cit. on p. 10).

[8] A. Einstein. "Über einen die Erzeugung und Verwandlung des Lichtes betreffenden heuristischen Gesichtspunkt". In: *Annalen der Physik* 322.6 (1905), pp. 132–148. DOI: https://doi.org/10.1002/andp.19053220607. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19053220607. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19053220607 (cit. on p. 10).

[9] *The Nobel Prize in Physics 1921 - NobelPrize.org — nobelprize.org*. https://www.nobelprize.org/prizes/physics/1921/summary/. [Accessed 27-05-2025] (cit. on p. 10).

[10] N. Bohr. "I. On the constitution of atoms and molecules". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 26.151 (July 1913), pp. 1–25. ISSN: 1941-5990. DOI: 10.1080/14786441308634955. URL: http://dx.doi.org/10.1080/14786441308634955 (cit. on p. 10).

[11] Louis De Broglie. "Recherches sur la théorie des Quanta". In: *Annales de Physique* 10.3 (1925), pp. 22–128. ISSN: 1286-4838. DOI: 10.1051/anphys/192510030022. URL: http://dx.doi.org/10.1051/anphys/192510030022 (cit. on p. 11).

[12] C. Davisson and L. H. Germer. "Diffraction of Electrons by a Crystal of Nickel". In: *Phys. Rev.* 30 (6 Dec. 1927), pp. 705–740. DOI: 10.1103/PhysRev.30.705. URL: https://link.aps.org/doi/10.1103/PhysRev.30.705 (cit. on p. 11).

[13] G. P. Thomson and A. Reid. “Diffraction of Cathode Rays by a Thin Film”. In: *Nature* 119.3007 (June 1927), pp. 890–890. ISSN: 1476-4687. DOI: 10.1038/119890a0. URL: http://dx.doi.org/10.1038/119890a0 (cit. on p. 11).

[14] W. Heisenberg. “Über quantentheoretische Umdeutung kinematischer und mechanischer Beziehungen.” In: *Zeitschrift für Physik* 33.1 (Dec. 1925), pp. 879–893. ISSN: 1434-601X. DOI: 10.1007/bf01328377. URL: http://dx.doi.org/10.1007/BF01328377 (cit. on p. 11).

[15] M. Born and P. Jordan. “Zur Quantenmechanik”. In: *Zeitschrift für Physik* 34.1 (Dec. 1925), pp. 858–888. ISSN: 1434-601X. DOI: 10.1007/bf01328531. URL: http://dx.doi.org/10.1007/BF01328531 (cit. on p. 11).

[16] M. Born, W. Heisenberg, and P. Jordan. “Zur Quantenmechanik. II.” In: *Zeitschrift für Physik* 35.8–9 (Aug. 1926), pp. 557–615. ISSN: 1434-601X. DOI: 10.1007/bf01379806. URL: http://dx.doi.org/10.1007/BF01379806 (cit. on p. 11).

[17] E. Schrödinger. “Quantisierung als Eigenwertproblem”. In: *Annalen der Physik* 384.4 (Jan. 1926), pp. 361–376. ISSN: 1521-3889. DOI: 10.1002/andp.19263840404. URL: http://dx.doi.org/10.1002/andp.19263840404 (cit. on p. 12).

[18] Max Born. “Zur Quantenmechanik der Stoßvorgänge”. In: *Zeitschrift für Physik* 37.12 (Dec. 1926), pp. 863–867. ISSN: 1434-601X. DOI: 10.1007/bf01397477. URL: http://dx.doi.org/10.1007/BF01397477 (cit. on p. 12).

[19] W. Heisenberg. “Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik”. In: *Zeitschrift für Physik* 43.3–4 (Mar. 1927), pp. 172–198. ISSN: 1434-601X. DOI: 10.1007/bf01397280. URL: http://dx.doi.org/10.1007/BF01397280 (cit. on p. 12).

[20] E. Schrödinger. “Discussion of Probability Relations between Separated Systems”. In: *Mathematical Proceedings of the Cambridge Philosophical Society* 31.4 (1935), pp. 555–563. DOI: 10.1017/S0305004100013554 (cit. on p. 12).

[21] A. Einstein, B. Podolsky, and N. Rosen. “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?” In: *Physical Review* 47.10 (May 1935), pp. 777–780. ISSN: 0031-899X. DOI: 10.1103/physrev.47.777. URL: http://dx.doi.org/10.1103/PhysRev.47.777 (cit. on p. 12).

[22] J. S. Bell. “On the Einstein Podolsky Rosen paradox”. In: *Physics Physique физика* 1.3 (Nov. 1964), pp. 195–200. ISSN: 0554-128X. DOI: 10.1103/physicsphysiquefizika.1.195. URL: http://dx.doi.org/10.1103/PhysicsPhysiqueFizika.1.195 (cit. on p. 13).

[23] Stuart J. Freedman and John F. Clauser. “Experimental Test of Local Hidden-Variable Theories”. In: *Phys. Rev. Lett.* 28 (14 Apr. 1972), pp. 938–941. DOI: 10.1103/PhysRevLett.28.938. URL: https://link.aps.org/doi/10.1103/PhysRevLett.28.938 (cit. on p. 13).

[24] Alain Aspect, Philippe Grangier, and Gérard Roger. “Experimental Tests of Realistic Local Theories via Bell’s Theorem”. In: *Physical Review Letters* 47.7 (Aug. 1981), pp. 460–463. ISSN: 0031-9007. DOI: 10.1103/physrevlett.47.460. URL: http://dx.doi.org/10.1103/PhysRevLett.47.460 (cit. on p. 13).

[25] *The Nobel Prize in Physics 2022 - NobelPrize.org — nobelprize.org*. https://www.nobelprize.org/prizes/physics/2022/summary/. [Accessed May 10, 2025] (cit. on p. 13).

[26] Lucien Hardy. “Quantum Theory From Five Reasonable Axioms”. In: arXiv:quant-ph/0101012 (Sept. 2001). arXiv:quant-ph/0101012. DOI: 10.48550/arXiv.quant-ph/0101012. URL: http://arxiv.org/abs/quant-ph/0101012 (cit. on p. 13).

[27] Giulio Chiribella, Giacomo Mauro D'Ariano, and Paolo Perinotti. "Quantum from principles". In: arXiv:1506.00398 (Mar. 2016). arXiv:1506.00398. DOI: 10.48550/arXiv.1506.00398. URL: http://arxiv.org/abs/1506.00398 (cit. on p. 13).

[28] Paul Dirac. *The principles of quantum mechanics*. en. Oxford, England: Oxford University Press, 1930 (cit. on p. 13).

[29] John VonNeumann. *Mathematische Grundlagen der Quantenmechanik*. ger. Berlin [u.a.]: Springer, 1932. URL: http://eudml.org/doc/203794 (cit. on p. 13).

[30] Lloyd Motz. "Quantization and the Classical Hamilton-Jacobi Equation". In: *Phys. Rev.* 126 (1 Apr. 1962), pp. 378–382. DOI: 10.1103/PhysRev.126.378. URL: https://link.aps.org/doi/10.1103/PhysRev.126.378 (cit. on p. 17).

[31] Gabriele Carcassi, Lorenzo Maccone, and Christine A. Aidala. "Four Postulates of Quantum Mechanics Are Three". In: *Physical Review Letters* 126.11 (Mar. 2021). ISSN: 1079-7114. DOI: 10.1103/physrevlett.126.110402. URL: http://dx.doi.org/10.1103/PhysRevLett.126.110402 (cit. on p. 20).

[32] G. Lindblad. "On the generators of quantum dynamical semigroups". In: *Communications in Mathematical Physics* 48.2 (June 1976), pp. 119–130. ISSN: 1432-0916. DOI: 10.1007/bf01608499. URL: http://dx.doi.org/10.1007/BF01608499 (cit. on p. 24).

[33] Vittorio Gorini, Andrzej Kossakowski, and E. C. G. Sudarshan. "Completely positive dynamical semigroups of N-level systems". In: *Journal of Mathematical Physics* 17.5 (May 1976), pp. 821–825. ISSN: 0022-2488. DOI: 10.1063/1.522979. eprint: https://pubs.aip.org/aip/jmp/article-pdf/17/5/821/19090720/821_1_online.pdf. URL: https://doi.org/10.1063/1.522979 (cit. on p. 24).

[34] Reinhard F. Werner. "Quantum states with Einstein-Podolsky-Rosen correlations admitting a hidden-variable model". In: *Phys. Rev. A* 40 (8 Oct. 1989), pp. 4277–4281. DOI: 10.1103/PhysRevA.40.4277. URL: https://link.aps.org/doi/10.1103/PhysRevA.40.4277 (cit. on p. 25).

[35] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6–7 (June 1982), pp. 467–488. ISSN: 1572-9575. DOI: 10.1007/bf02650179. URL: http://dx.doi.org/10.1007/BF02650179 (cit. on pp. 26, 135).

[36] R. Landauer. "Irreversibility and Heat Generation in the Computing Process". In: *IBM Journal of Research and Development* 5.3 (1961), pp. 183–191. DOI: 10.1147/rd.53.0183 (cit. on pp. 27, 32).

[37] C. H. Bennett. "Logical Reversibility of Computation". In: *IBM Journal of Research and Development* 17.6 (1973), pp. 525–532. DOI: 10.1147/rd.176.0525 (cit. on p. 27).

[38] Michael P. Frank. "Throwing computing into reverse". In: *IEEE Spectrum* 54.9 (2017), pp. 32–37. DOI: 10.1109/MSPEC.2017.8012237 (cit. on p. 27).

[39] 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* 22.5 (May 1980), pp. 563–591. ISSN: 1572-9613. DOI: 10.1007/bf01011339. URL: http://dx.doi.org/10.1007/BF01011339 (cit. on p. 27).

[40] David Deutsch. "Quantum theory, the Church–Turing principle and the universal quantum computer". In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400 (1985), pp. 117–97. URL: https://api.semanticscholar.org/CorpusID:1438116 (cit. on p. 27).

[41] David Deutsch. en. In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 425.1868 (Sept. 1989), pp. 73–90. ISSN: 0080-4630. DOI: 10.1098/rspa.1989.0099 (cit. on p. 27).

[42] D Deutsch and R Jozsa. "Rapid solution of problems by quantum computation". In: *Proc., Math. Phys. Sci.* 439.1907 (Dec. 1992), pp. 553–558 (cit. on p. 28).

[43] D.R. Simon. "On the power of quantum computation". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 116–123. DOI: 10.1109/SFCS.1994.365701 (cit. on p. 28).

[44] P.W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134. DOI: 10.1109/SFCS.1994.365700 (cit. on pp. 28, 45).

[45] R. L. Rivest, A. Shamir, and L. Adleman. "A method for obtaining digital signatures and public-key cryptosystems". In: *Commun. ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 0001-0782. DOI: 10.1145/359340.359342. URL: https://doi.org/10.1145/359340.359342 (cit. on p. 28).

[46] Lov K. Grover. "A fast quantum mechanical algorithm for database search". In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*. STOC '96. Philadelphia, Pennsylvania, USA: Association for Computing Machinery, 1996, pp. 212–219. ISBN: 0897917855. DOI: 10.1145/237814.237866. URL: https://doi.org/10.1145/237814.237866 (cit. on p. 28).

[47] David P. DiVincenzo. *Topics in Quantum Computers*. 1996. DOI: 10.48550/ARXIV.COND-MAT/9612126. URL: https://arxiv.org/abs/cond-mat/9612126 (cit. on p. 28).

[48] David P. DiVincenzo. "The Physical Implementation of Quantum Computation". In: *Fortschritte der Physik* 48.9-11 (2000), pp. 771–783. DOI: https://doi.org/10.1002/1521-3978(200009)48:9/11<771::AID-PROP771>3.0.CO;2-E. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/1521-3978%28200009%2948%3A9/11%3C771%3A%3AAID-PROP771%3E3.0.CO%3B2-E. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/1521-3978%28200009%2948%3A9/11%3C771%3A%3AAID-PROP771%3E3.0.CO%3B2-E (cit. on p. 28).

[49] I. Waki *et al.* "Observation of ordered structures of laser-cooled ions in a quadrupole storage ring". In: *Phys. Rev. Lett.* 68 (13 Mar. 1992), pp. 2007–2010. DOI: 10.1103/PhysRevLett.68.2007. URL: https://link.aps.org/doi/10.1103/PhysRevLett.68.2007 (cit. on p. 28).

[50] Paul G. Kwiat *et al.* "New High-Intensity Source of Polarization-Entangled Photon Pairs". In: *Phys. Rev. Lett.* 75 (24 Dec. 1995), pp. 4337–4341. DOI: 10.1103/PhysRevLett.75.4337. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.4337 (cit. on p. 28).

[51] Jonathan A. Jones, Michele Mosca, and Rasmus H. Hansen. "Implementation of a quantum search algorithm on a quantum computer". In: *Nature* 393.6683 (May 1998), pp. 344–346. ISSN: 1476-4687. DOI: 10.1038/30687. URL: http://dx.doi.org/10.1038/30687 (cit. on pp. 28, 29).

[52] Y. Nakamura, Yu. A. Pashkin, and J. S. Tsai. "Coherent control of macroscopic quantum states in a single-Cooper-pair box". In: *Nature* 398.6730 (Apr. 1999), pp. 786–788. ISSN: 1476-4687. DOI: 10.1038/19718. URL: http://dx.doi.org/10.1038/19718 (cit. on p. 28).

[53] C. Monroe *et al.* "Demonstration of a Fundamental Quantum Logic Gate". In: *Phys. Rev. Lett.* 75 (25 Dec. 1995), pp. 4714–4717. DOI: 10.1103/PhysRe

vLett.75.4714. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.4714 (cit. on p. 29).

[54] J. I. Cirac and P. Zoller. "Quantum Computations with Cold Trapped Ions". In: *Phys. Rev. Lett.* 74 (20 May 1995), pp. 4091–4094. DOI: 10.1103/PhysRevLett.74.4091. URL: https://link.aps.org/doi/10.1103/PhysRevLett.74.4091 (cit. on p. 29).

[55] Lieven M. K. Vandersypen *et al.* "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance". In: *Nature* 414.6866 (Dec. 2001), pp. 883–887. ISSN: 1476-4687. DOI: 10.1038/414883a. URL: http://dx.doi.org/10.1038/414883a (cit. on p. 29).

[56] Ferdinand Schmidt-Kaler *et al.* "Realization of the Cirac–Zoller controlled-NOT quantum gate". In: *Nature* 422.6930 (Mar. 2003), pp. 408–411. ISSN: 1476-4687. DOI: 10.1038/nature01494. URL: http://dx.doi.org/10.1038/nature01494 (cit. on p. 29).

[57] *2013: Quantum in the Cloud | News and features | university of Bristol*. Accessed May 10, 2025. 2013. URL: https://www.bristol.ac.uk/news/2013/9720.html (cit. on p. 29).

[58] *IBM Makes Quantum Computing Available on IBM Cloud to Accelerate Innovation — uk.newsroom.ibm.com*. https://uk.newsroom.ibm.com/2016-May-04-IBM-Makes-Quantum-Computing-Available-on-IBM-Cloud-to-Accelerate-Innovation. [Accessed 25-05-2025] (cit. on p. 29).

[59] John Preskill. "Quantum Computing in the NISQ era and beyond". en. In: *Quantum* 2 (Aug. 2018). arXiv:1801.00862 [quant-ph], p. 79. ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79 (cit. on p. 29).

[60] Rajeev Acharya *et al.* "Quantum error correction below the surface code threshold". In: *Nature* (Dec. 2024). ISSN: 1476-4687. DOI: 10.1038/s41586-

024-08449-y. URL: http://dx.doi.org/10.1038/s41586-024-08449-y (cit. on pp. 29, 44).

[61] Claude E. Shannon. "A symbolic analysis of relay and switching circuits". In: *Transactions of the American Institute of Electrical Engineers* 57.12 (1938), pp. 713–723. DOI: 10.1109/T-AIEE.1938.5057767 (cit. on p. 30).

[62] Glosser.ca. *Bloch_Sphere.svg*. CC BY-SA 3.0, via Wikimedia Commons. 2012. URL: https://commons.wikimedia.org/wiki/File:Bloch_Sphere.svg (cit. on p. 33).

[63] A Yu Kitaev. "Quantum computations: algorithms and error correction". In: *Russian Mathematical Surveys* 52.6 (Dec. 1997), pp. 1191–1249. ISSN: 1468-4829. DOI: 10.1070/rm1997v052n06abeh002155. URL: http://dx.doi.org/10.1070/RM1997v052n06ABEH002155 (cit. on p. 35).

[64] C.M. Dawson and M.A. Nielsen. "The Solovay-Kitaev algorithm". In: *Quantum Information and Computation* 6.1 (Jan. 2006), pp. 81–95. ISSN: 1533-7146. DOI: 10.26421/qic6.1-6. URL: http://dx.doi.org/10.26421/QIC6.1-6 (cit. on pp. 35, 36).

[65] Adriano Barenco *et al.* "Elementary gates for quantum computation". In: *Phys. Rev. A* 52 (5 Nov. 1995), pp. 3457–3467. DOI: 10.1103/PhysRevA.52.3457. URL: https://link.aps.org/doi/10.1103/PhysRevA.52.3457 (cit. on pp. 36, 39).

[66] Jean-Luc Brylinski and Ranee Brylinski. "Universal quantum gates". In: *Mathematics of Quantum Computation*. Chapman and Hall/CRC, Feb. 2002. ISBN: 9781420035377. DOI: 10.1201/9781420035377.pt2. URL: http://dx.doi.org/10.1201/9781420035377.pt2 (cit. on pp. 36, 39).

[67] Gadi Aleksandrowicz *et al. Qiskit: An Open-source Framework for Quantum Computing*. 2021. DOI: 10.5281/zenodo.2573505 (cit. on pp. 42, 102).

[68] P.W. Shor. "Fault-tolerant quantum computation". In: *Proceedings of 37th Conference on Foundations of Computer Science*. 1996, pp. 56–65. DOI: 10.1109/SFCS.1996.548464 (cit. on p. 44).

[69] Raymond Laflamme *et al.* "Perfect Quantum Error Correcting Code". en. In: *Physical Review Letters* 77.1 (July 1996), pp. 198–201. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.77.198. URL: https://link.aps.org/doi/10.1103/PhysRevLett.77.198 (visited on 03/26/2025) (cit. on p. 44).

[70] Joschka Roffe. "Quantum error correction: an introductory guide". In: *Contemporary Physics* 60.3 (July 2019). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00107514.2019.1667078, pp. 226–245. ISSN: 0010-7514. DOI: 10.1080/00107514.2019.1667078. URL: https://doi.org/10.1080/00107514.2019.1667078 (visited on 03/26/2025) (cit. on p. 44).

[71] Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. "Resilient Quantum Computation". In: *Science* 279.5349 (1998), pp. 342–345. DOI: 10.1126/science.279.5349.342. eprint: https://www.science.org/doi/pdf/10.1126/science.279.5349.342. URL: https://www.science.org/doi/abs/10.1126/science.279.5349.342 (cit. on p. 44).

[72] Dorit Aharonov and Michael Ben-Or. "Fault-Tolerant Quantum Computation with Constant Error Rate". In: *SIAM Journal on Computing* 38.4 (2008), pp. 1207–1282. DOI: 10.1137/S0097539799359385. eprint: https://doi.org/10.1137/S0097539799359385. URL: https://doi.org/10.1137/S0097539799359385 (cit. on p. 44).

[73] 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 (cit. on p. 44).

[74] Austin G. Fowler *et al.* "Surface codes: Towards practical large-scale quantum computation". In: *Phys. Rev. A* 86 (3 Sept. 2012), p. 032324. DOI: 10.1103/PhysRevA.86.032324. URL: https://link.aps.org/doi/10.1103/PhysRevA.86.032324 (cit. on p. 44).

[75] Craig Gidney and Martin Ekerå. "How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits". In: *Quantum* 5 (Apr. 2021), p. 433. ISSN: 2521-327X. DOI: 10.22331/q-2021-04-15-433. URL: https://doi.org/10.22331/q-2021-04-15-433 (cit. on p. 44).

[76] John Preskill. "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2 (Aug. 2018), p. 79. DOI: 10.22331/q-2018-08-06-79. URL: https://doi.org/10.22331/q-2018-08-06-79 (cit. on p. 45).

[77] Amara Katabarwa *et al.* "Early Fault-Tolerant Quantum Computing". In: *PRX Quantum* 5 (2 June 2024), p. 020101. DOI: 10.1103/PRXQuantum.5.020101. URL: https://link.aps.org/doi/10.1103/PRXQuantum.5.020101 (cit. on p. 45).

[78] Kishor Bharti *et al.* "Noisy intermediate-scale quantum algorithms". In: *Rev. Mod. Phys.* 94 (1 Feb. 2022), p. 015004. DOI: 10.1103/RevModPhys.94.015004. URL: https://link.aps.org/doi/10.1103/RevModPhys.94.015004 (cit. on p. 45).

[79] 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 (cit. on p. 45).

[80] 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 (cit. on p. 45).

[81] Krishnageetha Karuppasamy *et al.* "A Comprehensive Review of Quantum Circuit Optimization: Current Trends and Future Directions". In: *Quantum Reports* 7.1 (2025). ISSN: 2624-960X. DOI: 10.3390/quantum7010002. URL: https://www.mdpi.com/2624-960X/7/1/2 (cit. on p. 45).

[82] M. Cerezo *et al.* "Variational quantum algorithms". In: *Nat. Rev. Phys.* 3.9 (Aug. 2021), pp. 625–644. DOI: 10.1038/s42254-021-00348-9. URL: https://doi.org/10.1038/s42254-021-00348-9 (cit. on p. 46).

[83] B. P. Lanyon *et al.* "Towards quantum chemistry on a quantum computer". In: *Nat. Chem.* 2.2 (Jan. 2010), pp. 106–111. DOI: 10.1038/nchem.483. URL: https://doi.org/10.1038/nchem.483 (cit. on p. 48).

[84] Cornelius Hempel *et al.* "Quantum Chemistry Calculations on a Trapped-Ion Quantum Simulator". In: *Phys. Rev. X* 8 (3 July 2018), p. 031022. DOI: 10.1103/PhysRevX.8.031022. URL: https://link.aps.org/doi/10.1103/PhysRevX.8.031022 (cit. on p. 48).

[85] Yunseong Nam *et al.* "Ground-state energy estimation of the water molecule on a trapped-ion quantum computer". In: *npj Quant. Inf.* 6.1 (Apr. 2020), p. 33. DOI: 10.1038/s41534-020-0259-3. URL: https://doi.org/10.1038/s41534-020-0259-3 (cit. on p. 48).

[86] Christopher Ferrie and Osama Moussa. "Robust and efficient in situ quantum control". In: *Phys. Rev. A* 91 (5 May 2015), p. 052306. DOI: 10.1103/PhysRevA.91.052306. URL: https://link.aps.org/doi/10.1103/PhysRevA.91.052306 (cit. on pp. 48, 86, 105).

[87] Dawei Lu *et al.* "Enhancing quantum control by bootstrapping a quantum processor of 12 qubits". In: *npj Quant. Inf.* 3.1 (Oct. 2017), p. 45. DOI: 10.1038/s41534-017-0045-z. URL: https://doi.org/10.1038/s41534-017-0045-z (cit. on p. 48).

[88] D. J. Egger and F. K. Wilhelm. "Adaptive Hybrid Optimal Quantum Control for Imprecisely Characterized Systems". In: *Phys. Rev. Lett.* 112 (24 June 2014), p. 240503. DOI: 10.1103/PhysRevLett.112.240503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.112.240503 (cit. on p. 48).

[89] Xiao Yuan *et al.* "Theory of variational quantum simulation". In: *Quantum* 3 (Oct. 2019), p. 191. DOI: 10.22331/q-2019-10-07-191. URL: https://doi.org/10.22331/q-2019-10-07-191 (cit. on p. 48).

[90] Suguru Endo *et al.* "Variational Quantum Simulation of General Processes". In: *Phys. Rev. Lett.* 125 (1 June 2020), p. 010501. DOI: 10.1103/PhysRevLett.125.010501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.125.010501 (cit. on p. 48).

[91] Xin Wang, Zhixin Song, and Youle Wang. "Variational Quantum Singular Value Decomposition". In: *Quantum* 5 (June 2021), p. 483. DOI: 10.22331/q-2021-06-29-483. URL: https://doi.org/10.22331/q-2021-06-29-483 (cit. on p. 48).

[92] Kun Wang *et al.* "Detecting and quantifying entanglement on near-term quantum devices". In: *npj Quant. Inf.* 8.1 (May 2022). URL: https://doi.org/10.1038/s41534-022-00556-w (cit. on p. 48).

[93] A. D. Muñoz-Moller *et al.* "Variational Determination of Multiqubit Geometrical Entanglement in Noisy Intermediate-Scale Quantum Computers". In: *Phys. Rev. Appl.* 18 (2 Aug. 2022), p. 024048. DOI: 10.1103/PhysRevApplied.18.024048. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.18.024048 (cit. on pp. 48, 85).

[94] Christopher Ferrie. "Self-Guided Quantum Tomography". In: *Phys. Rev. Lett.* 113 (19 Nov. 2014), p. 190404. DOI: 10.1103/PhysRevLett.113.190404. URL: https://link.aps.org/doi/10.1103/PhysRevLett.113.190404 (cit. on pp. 48, 84, 86, 105, 119).

[95] Robert J. Chapman, Christopher Ferrie, and Alberto Peruzzo. "Experimental Demonstration of Self-Guided Quantum Tomography". In: *Phys. Rev. Lett.* 117 (4 July 2016), p. 040402. DOI: 10.1103/PhysRevLett.117.040402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.117.040402 (cit. on pp. 48, 84, 119).

[96] A. Utreras-Alarcón *et al.* "Stochastic optimization on complex variables and pure-state quantum tomography". In: *Sci. Rep.* 9.1 (Nov. 2019), p. 16143. DOI: 10.1038/s41598-019-52289-0. URL: https://doi.org/10.1038/s41598-019-52289-0 (cit. on pp. 48, 85, 87, 96).

[97] Leonardo Zambrano *et al.* "Estimation of pure quantum states in high dimension at the limit of quantum accuracy through complex optimization and statistical inference". In: *Sci. Rep.* 10.1 (July 2020), p. 12781. DOI: https://doi.org/10.1038/s41598-020-69646-z. URL: https://doi.org/10.1038/s41598-020-69646-z (cit. on p. 48).

[98] Markus Rambach *et al.* "Robust and Efficient High-Dimensional Quantum State Tomography". In: *Phys. Rev. Lett.* 126 (10 Mar. 2021), p. 100402. DOI: 10.1103/PhysRevLett.126.100402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.126.100402 (cit. on pp. 48, 85).

[99] Jacob Biamonte *et al.* "Quantum machine learning". In: *Nature* 549.7671 (Sept. 2017), pp. 195–202. DOI: 10.1038/nature23474. URL: https://doi.org/10.1038/nature23474 (cit. on p. 48).

[100] Marcello Benedetti *et al.* "Adversarial quantum circuit learning for pure state approximation". In: *New J. Phys.* 21.4 (Apr. 2019), p. 043023. DOI: 10.1088/1367-2630/ab14b5. URL: https://doi.org/10.1088/1367-2630/ab14b5 (cit. on p. 48).

[101] Andrew Patterson *et al.* "Quantum state discrimination using noisy quantum neural networks". In: *Phys. Rev. Res.* 3 (1 Jan. 2021), p. 013063. DOI: 10.

1103/PhysRevResearch.3.013063. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.3.013063 (cit. on p. 48).

[102] H. Chen *et al.* "Universal discriminative quantum neural networks". In: *Quantum Mach. Intell.* 3.1 (Dec. 2020), p. 1. DOI: 10.1007/s42484-020-00025-7. URL: https://doi.org/10.1007/s42484-020-00025-7 (cit. on p. 48).

[103] D. Concha *et al.* "Training a quantum measurement device to discriminate unknown non-orthogonal quantum states". In: *Sci. Rep.* 13.1 (May 2023). URL: https://doi.org/10.1038/s41598-023-34327-0 (cit. on pp. 48, 85).

[104] Xiaosi Xu, Simon C. Benjamin, and Xiao Yuan. "Variational Circuit Compiler for Quantum Error Correction". In: *Phys. Rev. Appl.* 15 (3 Mar. 2021), p. 034068. DOI: 10.1103/PhysRevApplied.15.034068. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.15.034068 (cit. on p. 48).

[105] Edward Farhi *et al.* "The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size". In: *Quantum* 6 (July 2022), p. 759. DOI: 10.22331/q-2022-07-07-759. URL: https://doi.org/10.22331/q-2022-07-07-759 (cit. on p. 48).

[106] Leo Zhou *et al.* "Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices". In: *Phys. Rev. X* 10 (2 June 2020), p. 021067. DOI: 10.1103/PhysRevX.10.021067. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.021067 (cit. on p. 48).

[107] Matthew P. Harrigan *et al.* "Quantum approximate optimization of non-planar graph problems on a planar superconducting processor". In: *Nat. Phys.* 17.3 (Feb. 2021), pp. 332–336. DOI: 10.1038/s41567-020-01105-y. URL: https://doi.org/10.1038/s41567-020-01105-y (cit. on p. 48).

[108] Kenji Kubo *et al.* "Variational quantum simulations of stochastic differential equations". In: *Phys. Rev. A* 103 (5 May 2021), p. 052425. DOI: 10.1103/PhysRevA.103.052425. URL: https://link.aps.org/doi/10.1103/PhysRevA.103.052425 (cit. on p. 48).

[109] Michael Lubasch *et al.* "Variational quantum algorithms for nonlinear problems". In: *Phys. Rev. A* 101 (1 Jan. 2020), p. 010301. DOI: 10.1103/PhysRevA.101.010301. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.010301 (cit. on p. 48).

[110] Paula García-Molina, Javier Rodríguez-Mediavilla, and Juan José García-Ripoll. "Quantum Fourier analysis for multivariate functions and applications to a class of Schrödinger-type partial differential equations". In: *Phys. Rev. A* 105 (1 Jan. 2022), p. 012433. DOI: 10.1103/PhysRevA.105.012433. URL: https://link.aps.org/doi/10.1103/PhysRevA.105.012433 (cit. on p. 48).

[111] Dylan Herman *et al. A Survey of Quantum Computing for Finance*. 2022. DOI: 10.48550/ARXIV.2201.02773. eprint: arXiv:2201.02773. URL: https://arxiv.org/abs/2201.02773 (cit. on p. 48).

[112] Jarrod R. McClean *et al.* "Barren plateaus in quantum neural network training landscapes". In: *Nat. Comm.* 9.1 (Nov. 2018), p. 4812. DOI: 10.1038/s41467-018-07090-4. URL: https://doi.org/10.1038/s41467-018-07090-4 (cit. on p. 48).

[113] M. Cerezo *et al.* "Cost function dependent barren plateaus in shallow parametrized quantum circuits". In: *Nature Communications* 12.1 (Mar. 2021). ISSN: 2041-1723. DOI: 10.1038/s41467-021-21728-w. URL: http://dx.doi.org/10.1038/s41467-021-21728-w (cit. on p. 48).

[114] Edward Grant *et al.* "An initialization strategy for addressing barren plateaus in parametrized quantum circuits". In: *Quantum* 3 (Dec. 2019), p. 214. ISSN:

2521-327X. DOI: 10.22331/q-2019-12-09-214. URL: http://dx.doi.org/10.22331/q-2019-12-09-214 (cit. on p. 48).

[115] Zoë Holmes *et al.* "Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus". In: *PRX Quantum* 3 (1 Jan. 2022), p. 010313. DOI: 10.1103/PRXQuantum.3.010313. URL: https://link.aps.org/doi/10.1103/PRXQuantum.3.010313 (cit. on p. 48).

[116] Andrea Skolik *et al.* "Layerwise learning for quantum neural networks". In: *Quantum Machine Intelligence* 3.1 (Jan. 2021). ISSN: 2524-4914. DOI: 10.1007/s42484-020-00036-4. URL: http://dx.doi.org/10.1007/s42484-020-00036-4 (cit. on p. 48).

[117] M. Cerezo *et al.* "Does provable absence of barren plateaus imply classical simulability? Or, why we need to rethink variational quantum computing". In: arXiv:2312.09121 (Mar. 2024). arXiv:2312.09121 [quant-ph]. DOI: 10.48550/arXiv.2312.09121. URL: http://arxiv.org/abs/2312.09121 (cit. on pp. 48, 136).

[118] Ewin Tang. "A quantum-inspired classical algorithm for recommendation systems". In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2019. Phoenix, AZ, USA: Association for Computing Machinery, 2019, pp. 217–228. ISBN: 9781450367059. DOI: 10.1145/3313276.3316310. URL: https://doi.org/10.1145/3313276.3316310 (cit. on p. 49).

[119] Juan Miguel Arrazola *et al.* "Quantum-inspired algorithms in practice". In: *Quantum* 4 (Aug. 2020), p. 307. ISSN: 2521-327X. DOI: 10.22331/q-2020-08-13-307. URL: http://dx.doi.org/10.22331/q-2020-08-13-307 (cit. on p. 49).

[120] Juan José García-Ripoll. "Quantum-inspired algorithms for multivariate analysis: from interpolation to partial differential equations". In: *Quantum* 5 (Apr.

2021). Publisher: Verein zur Forderung des Open Access Publizierens in den Quantenwissenschaften, p. 431. DOI: 10.22331/q-2021-04-15-431 (cit. on pp. 49, 51, 73, 135, 136, 142, 143, 152, 158).

[121] Bernhard Jobst *et al.* "Efficient MPS representations and quantum circuits from the Fourier modes of classical image data". In: *Quantum* 8 (Dec. 2024), p. 1544. ISSN: 2521-327X. DOI: 10.22331/q-2024-12-03-1544. URL: https://doi.org/10.22331/q-2024-12-03-1544 (cit. on pp. 49, 136, 142, 152, 161).

[122] Leonhard Hölscher *et al.* "Quantum-inspired fluid simulation of two-dimensional turbulence with GPU acceleration". In: *Phys. Rev. Res.* 7 (1 Jan. 2025), p. 013112. DOI: 10.1103/PhysRevResearch.7.013112. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.7.013112 (cit. on p. 49).

[123] Roger Penrose *et al.* "Applications of negative dimensional tensors". In: *Combinatorial mathematics and its applications* 1 (1971), pp. 221–244 (cit. on pp. 50, 52).

[124] Steven R. White. "Density matrix formulation for quantum renormalization groups". In: *Phys. Rev. Lett.* 69 (19 Nov. 1992), pp. 2863–2866. DOI: 10.1103/PhysRevLett.69.2863. URL: https://link.aps.org/doi/10.1103/PhysRevLett.69.2863 (cit. on p. 50).

[125] Stellan Östlund and Stefan Rommer. "Thermodynamic Limit of Density Matrix Renormalization". In: *Phys. Rev. Lett.* 75 (19 Nov. 1995), pp. 3537–3540. DOI: 10.1103/PhysRevLett.75.3537. URL: https://link.aps.org/doi/10.1103/PhysRevLett.75.3537 (cit. on p. 50).

[126] F. Verstraete, D. Porras, and J. I. Cirac. "Density Matrix Renormalization Group and Periodic Boundary Conditions: A Quantum Information Perspective". In: *Phys. Rev. Lett.* 93 (22 Nov. 2004), p. 227205. DOI: 10.1103/

PhysRevLett.93.227205. URL: https://link.aps.org/doi/10.1103/PhysRevLett.93.227205 (cit. on pp. 50, 77).

[127] Guifré Vidal. "Efficient Classical Simulation of Slightly Entangled Quantum Computations". In: *Phys. Rev. Lett.* 91 (14 Oct. 2003), p. 147902. DOI: 10.1103/PhysRevLett.91.147902. URL: https://link.aps.org/doi/10.1103/PhysRevLett.91.147902 (cit. on pp. 50, 72, 77).

[128] F. Verstraete and J. I. Cirac. *Renormalization algorithms for Quantum-Many Body Systems in two and higher dimensions*. 2004. eprint: arXiv:cond-mat/0407066 (cit. on pp. 50, 72).

[129] Alberto Baiardi and Markus Reiher. "The density matrix renormalization group in chemistry and molecular physics: Recent developments and new challenges". In: *The Journal of Chemical Physics* 152.4 (Jan. 2020), p. 040903. ISSN: 0021-9606. DOI: 10.1063/1.5129672. eprint: https://pubs.aip.org/aip/jcp/article-pdf/doi/10.1063/1.5129672/19996154/040903_1_1.5129672.pdf. URL: https://doi.org/10.1063/1.5129672 (cit. on p. 51).

[130] Guifré Vidal. "Efficient Simulation of One-Dimensional Quantum Many-Body Systems". In: *Phys. Rev. Lett.* 93.4 (July 2004). Publisher: American Physical Society, p. 040502. DOI: 10.1103/PhysRevLett.93.040502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.93.040502 (cit. on pp. 51, 77).

[131] F. Verstraete, J. J. García-Ripoll, and J. I. Cirac. "Matrix Product Density Operators: Simulation of Finite-Temperature and Dissipative Systems". In: *Phys. Rev. Lett.* 93 (20 Nov. 2004), p. 207204. DOI: 10.1103/PhysRevLett.93.207204. URL: https://link.aps.org/doi/10.1103/PhysRevLett.93.207204 (cit. on pp. 51, 73).

[132] Aleksandr Berezutskii *et al.* *Tensor networks for quantum computing*. 2025. DOI: 10.48550/ARXIV.2503.08626. URL: https://arxiv.org/abs/2503.08626 (cit. on p. 51).

[133] E. Miles Stoudenmire and David J. Schwab. "Supervised Learning with Quantum-Inspired Tensor Networks". In: (2016). DOI: 10.48550/ARXIV.1605.05775. URL: https://arxiv.org/abs/1605.05775 (cit. on p. 51).

[134] Alejandro Pozas-Kerstjens *et al.* "Privacy-preserving machine learning with tensor networks". In: *Quantum* 8 (July 2024), p. 1425. ISSN: 2521-327X. DOI: 10.22331/q-2024-07-25-1425. URL: http://dx.doi.org/10.22331/q-2024-07-25-1425 (cit. on p. 51).

[135] Jose I. Latorre. *Image compression and entanglement*. 2005. DOI: 10.48550/ARXIV.QUANT-PH/0510031. URL: https://arxiv.org/abs/quant-ph/0510031 (cit. on p. 51).

[136] Sergey Dolgov, Boris Khoromskij, and Dmitry Savostyanov. "Superfast Fourier Transform Using QTT Approximation". In: *Journal of Fourier Analysis and Applications* 18.5 (Oct. 2012), pp. 915–953. ISSN: 1531-5851. DOI: 10.1007/s00041-012-9227-4 (cit. on pp. 51, 152).

[137] Jielun Chen, E.M. Stoudenmire, and Steven R. White. "Quantum Fourier Transform Has Small Entanglement". In: *PRX Quantum* 4 (4 Oct. 2023), p. 040318. DOI: 10.1103/PRXQuantum.4.040318. URL: https://link.aps.org/doi/10.1103/PRXQuantum.4.040318 (cit. on pp. 51, 152, 168).

[138] Nikita Gourianov *et al.* "A quantum-inspired approach to exploit turbulence structures". In: *Nature Computational Science* 2.1 (Jan. 2022), pp. 30–37. DOI: 10.1038/s43588-021-00181-1. URL: https://doi.org/10.10382Fs43588-021-00181-1 (cit. on pp. 51, 142).

[139] M. Fannes, B. Nachtergaele, and R. F. Werner. "Finitely correlated states on quantum spin chains". In: *Communications in Mathematical Physics* 144.3 (Mar. 1992), pp. 443–490. ISSN: 1432-0916. DOI: 10.1007/bf02099178. URL: http://dx.doi.org/10.1007/BF02099178 (cit. on p. 56).

[140] Norbert Schuch *et al.* "Computational Complexity of Projected Entangled Pair States". In: *Physical Review Letters* 98.14 (Apr. 2007). Publisher: American Physical Society, p. 140506. DOI: 10.1103/PhysRevLett.98.140506. URL: https://link.aps.org/doi/10.1103/PhysRevLett.98.140506 (visited on 03/22/2025) (cit. on p. 57).

[141] Michael Lubasch, J. Ignacio Cirac, and Mari-Carmen Bañuls. "Algorithms for finite projected entangled pair states". In: *Physical Review B* 90.6 (Aug. 2014). Publisher: American Physical Society, p. 064425. DOI: 10.1103/PhysRevB.90.064425. URL: https://link.aps.org/doi/10.1103/PhysRevB.90.064425 (visited on 03/22/2025) (cit. on p. 57).

[142] Joseph Tindall, Miles Stoudenmire, and Ryan Levy. *Compressing multivariate functions with tree tensor networks*. 2024. eprint: arXiv:2410.03572 (cit. on p. 57).

[143] Y.-Y. Shi, L.-M. Duan, and G. Vidal. "Classical simulation of quantum many-body systems with a tree tensor network". In: *Physical Review A* 74.2 (Aug. 2006). Publisher: American Physical Society, p. 022320. DOI: 10.1103/PhysRevA.74.022320. URL: https://link.aps.org/doi/10.1103/PhysRevA.74.022320 (visited on 03/22/2025) (cit. on p. 57).

[144] Robert N. C. Pfeifer, Glen Evenbly, and Guifré Vidal. "Entanglement renormalization, scale invariance, and quantum criticality". In: *Physical Review A* 79.4 (Apr. 2009). Publisher: American Physical Society, p. 040301. DOI: 10.1103/PhysRevA.79.040301. URL: https://link.aps.org/doi/10.1103/PhysRevA.79.040301 (visited on 03/22/2025) (cit. on p. 58).

[145] Carl Eckart and Gale Young. "The Approximation of One Matrix by Another of Lower Rank". In: *Psychometrika* 1.3 (1936), pp. 211–218. DOI: 10.1007/BF02288367 (cit. on p. 59).

[146] N. Halko, P. G. Martinsson, and J. A. Tropp. "Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions". In: *SIAM Review* 53.2 (2011), pp. 217–288. DOI: 10.1137/090771806. eprint: https://doi.org/10.1137/090771806. URL: https://doi.org/10.1137/090771806 (cit. on p. 60).

[147] S. V. Dolgov, B. N. Khoromskij, and I. V. Oseledets. "Fast Solution of Parabolic Problems in the Tensor Train/Quantized Tensor Train Format with Initial Application to the Fokker–Planck Equation". In: *SIAM Journal on Scientific Computing* 34.6 (2012), A3016–A3038. DOI: 10.1137/120864210. eprint: https://doi.org/10.1137/120864210. URL: https://doi.org/10.1137/120864210 (cit. on pp. 62, 142).

[148] I. V. Oseledets. "Tensor-Train Decomposition". In: *SIAM Journal on Scientific Computing* 33.5 (2011), pp. 2295–2317. DOI: 10.1137/090752286. eprint: https://doi.org/10.1137/090752286. URL: https://doi.org/10.1137/090752286 (cit. on p. 62).

[149] M B Hastings. "An area law for one-dimensional quantum systems". In: *Journal of Statistical Mechanics: Theory and Experiment* 2007.08 (Aug. 2007), P08024. DOI: 10.1088/1742-5468/2007/08/P08024. URL: https://dx.doi.org/10.1088/1742-5468/2007/08/P08024 (cit. on p. 64).

[150] F. Verstraete and J. I. Cirac. "Matrix product states represent ground states faithfully". In: *Phys. Rev. B* 73 (9 Mar. 2006), p. 094423. DOI: 10.1103/PhysRevB.73.094423. URL: https://link.aps.org/doi/10.1103/PhysRevB.73.094423 (cit. on p. 64).

[151] J. Eisert, M. Cramer, and M. B. Plenio. "Colloquium: Area laws for the entanglement entropy". In: *Rev. Mod. Phys.* 82 (1 Feb. 2010), pp. 277–306. DOI: 10.1103/RevModPhys.82.277. URL: https://link.aps.org/doi/10.1103/RevModPhys.82.277 (cit. on p. 64).

[152] Daniel M. Greenberger, Michael A. Horne, and Anton Zeilenger. "Going Beyond Bell's Theorem". In: *Bell's Theorem, Quantum Theory and Conceptions of the Universe*. Ed. by Menas Kafatos. Kluwer Academic Publishers, 1989, pp. 69–72 (cit. on p. 66).

[153] Don N. Page. "Average entropy of a subsystem". In: *Phys. Rev. Lett.* 71 (9 Aug. 1993), pp. 1291–1294. DOI: 10.1103/PhysRevLett.71.1291. URL: https://link.aps.org/doi/10.1103/PhysRevLett.71.1291 (cit. on p. 67).

[154] D. Perez-Garcia *et al.* "Matrix product state representations". In: *Quantum Info. Comput.* 7.5 (July 2007), pp. 401–430. ISSN: 1533-7146 (cit. on p. 68).

[155] Michael Zwolak and Guifré Vidal. "Mixed-State Dynamics in One-Dimensional Quantum Lattice Systems: A Time-Dependent Superoperator Renormalization Algorithm". In: *Phys. Rev. Lett.* 93 (20 Nov. 2004), p. 207205. DOI: 10.1103/PhysRevLett.93.207205. URL: https://link.aps.org/doi/10.1103/PhysRevLett.93.207205 (cit. on p. 73).

[156] J. Gidi *et al.* "Stochastic optimization algorithms for quantum applications". In: *Phys. Rev. A* 108 (3 Sept. 2023), p. 032409. DOI: 10.1103/PhysRevA.108.032409. URL: https://link.aps.org/doi/10.1103/PhysRevA.108.032409 (cit. on pp. 82, 197).

[157] David Saad, ed. *On-Line Learning in Neural Networks*. Cambridge University Press, Jan. 1999. DOI: 10.1017/cbo9780511569920. URL: https://doi.org/10.1017/cbo9780511569920 (cit. on p. 83).

[158] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization". In: *J. Mach. Learn. Res.* 12.null (July 2011), pp. 2121–2159. ISSN: 1532-4435. URL: https://dl.acm.org/doi/10.5555/1953048.2021068 (cit. on p. 83).

[159] Tijmen Tieleman, Geoffrey Hinton, *et al.* "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude". In: *COURSERA: Neural networks for machine learning* 4.2 (2012), pp. 26–31 (cit. on p. 83).

[160] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2014. DOI: 10.48550/ARXIV.1412.6980. eprint: arXiv:1412.6980. URL: https://arxiv.org/abs/1412.6980 (cit. on p. 83).

[161] Timothy Dozat. "Incorporating Nesterov Momentum into Adam". In: *Proceedings of the 4th International Conference on Learning Representations*. 2016, pp. 1–4 (cit. on p. 83).

[162] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. *On the Convergence of Adam and Beyond*. 2019. DOI: 10.48550/ARXIV.1904.09237. eprint: arXiv:1904.09237. URL: https://arxiv.org/abs/1904.09237 (cit. on p. 83).

[163] J. A. Nelder and R. Mead. "A Simplex Method for Function Minimization". In: *Comput. J.* 7.4 (Jan. 1965), pp. 308–313. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308. URL: https://doi.org/10.1093/comjnl/7.4.308 (cit. on p. 83).

[164] M. J. D. Powell. "An efficient method for finding the minimum of a function of several variables without calculating derivatives". In: *Comput. J.* 7.2 (Jan. 1964), pp. 155–162. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.2.155. eprint: https://academic.oup.com/comjnl/article-pdf/7/2/155/959784/070155.pdf. URL: https://doi.org/10.1093/comjnl/7.2.155 (cit. on p. 83).

[165] Stephen G. Nash. "Newton-Type Minimization via the Lanczos Method". In: *SIAM J. Numer. Anal.* 21.4 (1984), pp. 770–788. DOI: 10.1137/0721052. URL: https://doi.org/10.1137/0721052 (cit. on p. 83).

[166] D. Kraft. *A software package for sequential quadratic programming*. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht. Wiss. Berichtswesen d. DFVLR, 1988. URL: https://books.google.ch/books?id=4rKaGwAACAAJ (cit. on p. 83).

[167] R. Fletcher and C. M. Reeves. "Function minimization by conjugate gradients". In: *Comput. J.* 7.2 (Jan. 1964), pp. 149–154. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.2.149. eprint: https://academic.oup.com/comjnl/article-pdf/7/2/149/959725/070149.pdf. URL: https://doi.org/10.1093/comjnl/7.2.149 (cit. on p. 83).

[168] M. J. D. Powell. "A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation". In: *Advances in Optimization and Numerical Analysis*. Springer Netherlands, 1994, pp. 51–67. DOI: 10.1007/978-94-015-8330-5_4. URL: https://doi.org/10.1007/978-94-015-8330-5_4 (cit. on p. 83).

[169] Richard H. Byrd *et al.* "A Limited Memory Algorithm for Bound Constrained Optimization". In: *SIAM J. Numer. Anal.* 16.5 (1995), pp. 1190–1208. DOI: 10.1137/0916069. eprint: https://doi.org/10.1137/0916069. URL: https://doi.org/10.1137/0916069 (cit. on p. 83).

[170] Marucha Lalee, Jorge Nocedal, and Todd Plantenga. "On the Implementation of an Algorithm for Large-Scale Equality Constrained Optimization". In: *SIAM J. Numer. Anal.* 8.3 (1998), pp. 682–706. DOI: 10.1137/S1052623493262993. URL: https://doi.org/10.1137/S1052623493262993 (cit. on p. 83).

[171] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer New York, 2006. DOI: 10.1007/978-0-387-40065-5. URL: https://doi.org/10.1007/978-0-387-40065-5 (cit. on p. 83).

[172] Pauli Virtanen *et al.* "SciPy 1.0: fundamental algorithms for scientific computing in Python". In: *Nat. Methods* 17.3 (Feb. 2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2. URL: https://doi.org/10.1038/s41592-019-0686-2 (cit. on p. 83).

[173] Harold J. Kushner and Dean S. Clark. *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. Vol. 26. Springer New York, 1978. ISBN: 978-0-387-90341-5. DOI: 10.1007/978-1-4684-9352-8 (cit. on pp. 83, 88).

[174] Harold J. Kushner and G. George Yin. *Stochastic Approximation Algorithms and Applications*. Springer New York, 1997. ISBN: 978-1-4899-2698-2. DOI: 10.1007/978-1-4899-2696-8 (cit. on pp. 83, 88).

[175] Arthur E. Albert and Leland A. Gardner. *Stochastic Approximation and Nonlinear Regression*. The MIT Press, 2003. DOI: 10.7551/mitpress/6464.001.0001. URL: https://doi.org/10.7551/mitpress/6464.001.0001 (cit. on pp. 83, 88).

[176] James Spall. "Introduction to Stochastic Search and Optimization. Estimation, Simulation, and Control". In: *IEEE Trans. Neural Netw. Learn. Syst.* 18 (May 2007). DOI: 10.1002/0471722138 (cit. on pp. 83, 88, 102).

[177] S. Bhatnagar, H. L. Prasad, and L. A. Prashanth. *Stochastic Recursive Algorithms for Optimization*. Vol. 434. Springer London, 2013. ISBN: 978-1-4471-4284-3. DOI: 10.1007/978-1-4471-4285-0 (cit. on pp. 83, 88).

[178] Wim Lavrijsen *et al.* "Classical Optimizers for Noisy Intermediate-Scale Quantum Devices". In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, Oct. 2020, pp. 267–277. DOI: 10.1109/

qce49297.2020.00041. URL: https://doi.org/10.1109/qce49297.2020.00041 (cit. on p. 83).

[179] James C. Spall. "A Stochastic Approximation Technique for Generating Maximum Likelihood Parameter Estimates". In: *1987 American Control Conference*. 1987, pp. 1161–1167. DOI: 10.23919/ACC.1987.4789489 (cit. on pp. 84, 90).

[180] Abhinav Kandala *et al.* "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". In: *Nature* 549.7671 (Sept. 2017), pp. 242–246. DOI: 10.1038/nature23879. URL: https://doi.org/10.1038/nature23879 (cit. on pp. 84, 86, 109).

[181] O. V. Borzenkova *et al.* "Variational simulation of Schwinger's Hamiltonian with polarization qubits". In: *Appl. Phys. Lett.* 118.14 (Apr. 2021), p. 144002. DOI: 10.1063/5.0043322. URL: https://doi.org/10.1063/5.0043322 (cit. on p. 84).

[182] Ikko Hamamura and Takashi Imamichi. "Efficient evaluation of quantum observables using entangled measurements". In: *npj Quan. Inf.* 6.1 (June 2020), p. 56. DOI: 10.1038/s41534-020-0284-2. URL: https://doi.org/10.1038/s41534-020-0284-2 (cit. on pp. 84, 85).

[183] Francisco Escudero *et al. Hardware-efficient entangled measurements for variational quantum algorithms*. 2022. DOI: 10.48550/ARXIV.2202.06979. eprint: arXiv:2202.06979. URL: https://arxiv.org/abs/2202.06979 (cit. on pp. 84, 85).

[184] Pablo Díez-Valle, Diego Porras, and Juan José García-Ripoll. "Quantum variational optimization: The role of entanglement and problem hardness". In: *Phys. Rev. A* 104 (6 Dec. 2021), p. 062426. DOI: 10.1103/PhysRevA.104.062426. URL: https://link.aps.org/doi/10.1103/PhysRevA.104.062426 (cit. on p. 84).

[185] Marcello Benedetti *et al.* "Parameterized quantum circuits as machine learning models". In: *Quantum Sci. Technol.* 4.4 (Nov. 2019), p. 043001. DOI: 10.1088/2058-9565/ab4eb5. URL: https://doi.org/10.1088/2058-9565/ab4eb5 (cit. on p. 84).

[186] Stefano Mangini *et al.* "Quantum computing model of an artificial neuron with continuously valued input data". In: *Mach. Learn.: Sci. Technol.* 1.4 (Oct. 2020), p. 045008. DOI: 10.1088/2632-2153/abaf98. URL: https://doi.org/10.1088/2632-2153/abaf98 (cit. on p. 84).

[187] Gabriele Agliardi and Enrico Prati. "Optimal Tuning of Quantum Generative Adversarial Networks for Multivariate Distribution Loading". In: *Quantum Rep.* 4.1 (Feb. 2022), pp. 75–105. DOI: 10.3390/quantum4010006. URL: https://doi.org/10.3390/quantum4010006 (cit. on p. 84).

[188] Zhibo Hou *et al.* "Experimental realization of self-guided quantum process tomography". In: *Phys. Rev. A* 101 (2 Feb. 2020), p. 022317. DOI: 10.1103/PhysRevA.101.022317. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.022317 (cit. on p. 84).

[189] J.C. Spall. "Adaptive stochastic approximation by the simultaneous perturbation method". In: *IEEE Trans. Autom. Contr.* 45 (Nov. 2000), pp. 1839–1853. DOI: 10.1109/TAC.2000.880982 (cit. on pp. 84, 91, 92, 102).

[190] Chen Wang. *An overview of SPSA: recent development and applications*. 2020. DOI: 10.48550/ARXIV.2012.06952. eprint: arXiv:2012.06952. URL: https://arxiv.org/abs/2012.06952 (cit. on p. 84).

[191] Xun Zhu and James C. Spall. "A modified second-order SPSA optimization algorithm for finite samples". In: *Int. J. Adapt. Control Signal Process.* 16.5 (2002), pp. 397–409. DOI: 10.1002/acs.715. URL: https://doi.org/10.1002/acs.715 (cit. on pp. 84, 103).

[192] James Stokes *et al.* "Quantum Natural Gradient". In: *Quantum* 4 (May 2020), p. 269. DOI: 10.22331/q-2020-05-25-269. URL: https://doi.org/10.22331/q-2020-05-25-269 (cit. on pp. 84, 94).

[193] S. Amari. "Natural Gradient Works Efficiently in Learning". In: *Neural Comput.* 10.2 (Feb. 1998), pp. 251–276. DOI: 10.1162/089976698300017746. URL: https://doi.org/10.1162/089976698300017746 (cit. on p. 85).

[194] Tengyuan Liang *et al.* "Fisher-Rao Metric, Geometry, and Complexity of Neural Networks". In: *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. Ed. by Kamalika Chaudhuri and Masashi Sugiyama. Vol. 89. Proceedings of Machine Learning Research. PMLR, 16–18 Apr 2019, pp. 888–896. URL: https://proceedings.mlr.press/v89/liang19a.html (cit. on p. 85).

[195] Julien Gacon *et al.* "Simultaneous Perturbation Stochastic Approximation of the Quantum Fisher Information". In: *Quantum* 5 (Oct. 2021), p. 567. DOI: 10.22331/q-2021-10-20-567. URL: https://doi.org/10.22331/q-2021-10-20-567 (cit. on pp. 85, 94, 102).

[196] Harry Buhrman *et al.* "Quantum Fingerprinting". In: *Phys. Rev. Lett.* 87 (16 Sept. 2001), p. 167902. DOI: 10.1103/PhysRevLett.87.167902. URL: https://link.aps.org/doi/10.1103/PhysRevLett.87.167902 (cit. on p. 85).

[197] Qisheng Wang *et al.* "Quantum Algorithm for Fidelity Estimation". In: *IEEE Trans. Inf. Theory* 69.1 (Jan. 2023), pp. 273–282. DOI: 10.1109/tit.2022.3203985. URL: https://doi.org/10.1109/tit.2022.3203985 (cit. on p. 85).

[198] Pranav Gokhale *et al. Minimizing State Preparations in Variational Quantum Eigensolver by Partitioning into Commuting Families*. 2019. DOI: 10.48550/ARXIV.1907.13623. eprint: arXiv:1907.13623. URL: https://arxiv.org/abs/1907.13623 (cit. on p. 85).

[199] Sergey Bravyi *et al.* *Tapering off qubits to simulate fermionic Hamiltonians*. 2017. DOI: 10.48550/ARXIV.1701.08213. eprint: 1701.08213. URL: https://arxiv.org/abs/1701.08213 (cit. on p. 85).

[200] Naoki Yamamoto. *On the natural gradient for variational quantum eigensolver*. 2019. DOI: 10.48550/ARXIV.1909.05074. eprint: arXiv:1909.05074. URL: https://arxiv.org/abs/1909.05074 (cit. on p. 85).

[201] David Wierichs, Christian Gogolin, and Michael Kastoryano. "Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer". In: *Phys. Rev. Res.* 2 (4 Nov. 2020), p. 043246. DOI: 10.1103/PhysRevResearch.2.043246. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.043246 (cit. on p. 85).

[202] Barnaby van Straaten and Bálint Koczor. "Measurement Cost of Metric-Aware Variational Quantum Algorithms". In: *PRX Quantum* 2 (3 Aug. 2021), p. 030324. DOI: 10.1103/PRXQuantum.2.030324. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.030324 (cit. on p. 85).

[203] W. Wirtinger. "Zur formalen Theorie der Funktionen von mehr komplexen Veränderlichen". In: *Math. Ann.* 97 (1927), pp. 357–376. URL: http://eudml.org/doc/182642 (cit. on pp. 85, 87, 95).

[204] Ken Kreutz-Delgado. *The Complex Gradient Operator and the CR-Calculus*. 2009. eprint: arXiv:0906.4835v1. URL: https://arxiv.org/abs/0906.4835v1 (cit. on pp. 85, 96, 98).

[205] Yuan Yao *et al.* "Complex natural gradient optimization for optical quantum circuit design". In: *Phys. Rev. A* 105 (5 May 2022), p. 052402. DOI: 10.1103/PhysRevA.105.052402. URL: https://link.aps.org/doi/10.1103/PhysRevA.105.052402 (cit. on pp. 85, 99).

[206] Songchuan Zhang, Youshen Xia, and Jun Wang. "A Complex-Valued Projection Neural Network for Constrained Optimization of Real Functions in Complex Variables". In: *IEEE Trans. Neural Netw. Learn. Syst.* 26.12 (2015), pp. 3227–3238. DOI: 10.1109/TNNLS.2015.2441697 (cit. on pp. 85, 87).

[207] Akira Hirose. *Complex-Valued Neural Networks*. Vol. 400. Springer Berlin Heidelberg, 2012. ISBN: 978-3-642-27631-6. DOI: 10.1007/978-3-642-27632-3 (cit. on pp. 85, 87).

[208] O. M. Smirnov and C. Tasse. "Radio interferometric gain calibration as a complex optimization problem". In: *Mon. Not. R. Astron. Soc.* 449.3 (Apr. 2015), pp. 2668–2684. ISSN: 0035-8711. DOI: 10.1093/mnras/stv418. URL: https://doi.org/10.1093/mnras/stv418 (cit. on pp. 85, 87).

[209] Juan Miguel Arrazola *et al.* "Machine learning method for state preparation and gate synthesis on photonic quantum computers". In: *Quantum Sci. Technol.* 4.2 (Jan. 2019), p. 024004. DOI: 10.1088/2058-9565/aaf59e. URL: https://doi.org/10.1088/2058-9565/aaf59e (cit. on p. 85).

[210] Nathan Killoran *et al.* "Continuous-variable quantum neural networks". In: *Phys. Rev. Res.* 1 (3 Oct. 2019), p. 033063. DOI: 10.1103/PhysRevResearch.1.033063. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.1.033063 (cit. on p. 85).

[211] J. Cortés-Vega *et al.* "Detecting entanglement of unknown states by violating the Clauser–Horne–Shimony–Holt inequality". In: *Quantum Inf. Process.* 22.5 (May 2023). URL: https://doi.org/10.1007/s11128-023-03953-y (cit. on p. 85).

[212] Navin Khaneja *et al.* "Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms". In: *J. Magn. Reson.* 172.2 (Feb. 2005), pp. 296–305. DOI: 10.1016/j.jmr.2004.11.004. URL: https://doi.org/10.1016/j.jmr.2004.11.004 (cit. on pp. 86, 105, 113).

[213] Laurent Sorber, Marc Van Barel, and Lieven De Lathauwer. "Unconstrained Optimization of Real Functions in Complex Variables". In: *SIAM J. Opt.* 22.3 (2012), pp. 879–898. DOI: 10.1137/110832124. URL: https://doi.org/10.1137/110832124 (cit. on p. 87).

[214] Herbert Robbins and Sutton Monro. "A Stochastic Approximation Method". In: *Ann. Math. Stat.* 22.3 (1951), pp. 400–407. DOI: 10.1214/aoms/1177729586. URL: https://doi.org/10.1214/aoms/1177729586 (cit. on p. 88).

[215] J. Kiefer and J. Wolfowitz. "Stochastic Estimation of the Maximum of a Regression Function". In: *Ann. Math. Stat.* 23.3 (1952), pp. 462–466. DOI: 10.1214/aoms/1177729392. URL: https://doi.org/10.1214/aoms/1177729392 (cit. on pp. 88, 90).

[216] S. Amari and S.C. Douglas. "Why natural gradient?" In: *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*. IEEE, 2002. DOI: 10.1109/icassp.1998.675489. URL: https://doi.org/10.1109/icassp.1998.675489 (cit. on p. 93).

[217] Guangrong Yan and H. Fan. "A Newton-like algorithm for complex variables with applications in blind equalization". In: *IEEE Trans. Signal Process.* 48.2 (2000), pp. 553–556. DOI: 10.1109/78.823982. URL: https://doi.org/10.1109/78.823982 (cit. on p. 98).

[218] M. Muñoz *et al. Complex Field Formulation of the Quantum Estimation Theory.* 2022. DOI: 10.48550/ARXIV.2203.03064. eprint: arXiv:2203.03064. URL: https://arxiv.org/abs/2203.03064 (cit. on p. 100).

[219] Walter Ritz. "Über eine neue Methode zur Lösung gewisser Variationsprobleme der mathematischen Physik". In: *J. Reine Angew. Math.* 135 (Jan. 1909),

pp. 1–61. DOI: 10.1515/crll.1909.135.1. URL: https://doi.org/10.1515/crll.1909.135.1 (cit. on p. 107).

[220] J. W. Rayleigh. "In Finding the Correction for the Open End of an Organ-Pipe". In: *Phil. Trans.* 161 (1870), p. 75 (cit. on p. 107).

[221] Alberto Peruzzo *et al.* "A variational eigenvalue solver on a photonic quantum processor". In: *Nat. Comm.* 5.1 (July 2014), p. 4213. DOI: 10.1038/ncomms5213. URL: https://doi.org/10.1038/ncomms5213 (cit. on p. 107).

[222] Moshe Shapiro and Paul Brumer. *Quantum Control of Molecular Processes*. Wiley-VCH Verlag GmbH & Co. KGaA, Dec. 2011. DOI: 10.1002/9783527639700. URL: https://doi.org/10.1002/9783527639700 (cit. on p. 112).

[223] Steven Chu. "Cold atoms and quantum control". In: *Nature* 416.6877 (Mar. 2002), pp. 206–210. DOI: 10.1038/416206a. URL: https://doi.org/10.1038/416206a (cit. on p. 112).

[224] Ramon van Handel, John K. Stockton, and Hideo Mabuchi. "Modelling and feedback control design for quantum state preparation". In: *J. Opt. B: Quantum Semiclass. Opt.* 7.10 (Sept. 2005), S179–S197. DOI: 10.1088/1464-4266/7/10/001. URL: https://doi.org/10.1088/1464-4266/7/10/001 (cit. on p. 112).

[225] Seth Lloyd. "Coherent quantum feedback". In: *Phys. Rev. A* 62 (2 July 2000), p. 022108. DOI: 10.1103/PhysRevA.62.022108. URL: https://link.aps.org/doi/10.1103/PhysRevA.62.022108 (cit. on p. 112).

[226] J. Werschnik and E. K. U. Gross. "Quantum optimal control theory". In: *J. Phys. B* 40.18 (Sept. 2007), R175–R211. DOI: 10.1088/0953-4075/40/18/r01. URL: https://doi.org/10.1088/0953-4075/40/18/r01 (cit. on p. 113).

[227] Daniel Burgarth *et al.* "Generalized Adiabatic Theorem and Strong-Coupling Limits". In: *Quantum* 3 (June 2019), p. 152. DOI: 10.22331/q-2019-06-12-152. URL: https://doi.org/10.22331/q-2019-06-12-152 (cit. on p. 113).

[228] Gregory Quiroz. "Robust quantum control for adiabatic quantum computation". In: *Phys. Rev. A* 99 (6 June 2019), p. 062306. DOI: 10.1103/PhysRevA.99.062306. URL: https://link.aps.org/doi/10.1103/PhysRevA.99.062306 (cit. on p. 113).

[229] Brian B. Zhou *et al.* "Accelerated quantum control using superadiabatic dynamics in a solid-state lambda system". In: *Nature Phys.* 13.4 (Nov. 2016), pp. 330–334. DOI: 10.1038/nphys3967. URL: https://doi.org/10.1038/nphys3967 (cit. on p. 113).

[230] Zhaoyan Wu and Hui Yang. "Validity of the quantum adiabatic theorem". In: *Phys. Rev. A* 72 (1 July 2005), p. 012114. DOI: 10.1103/PhysRevA.72.012114. URL: https://link.aps.org/doi/10.1103/PhysRevA.72.012114 (cit. on p. 113).

[231] Kaisa Laiho, Malte Avenhaus, and Christine Silberhorn. "Characteristics of displaced single photons attained via higher order factorial moments". In: *New J. Phys.* 14.10 (Oct. 2012), p. 105011. DOI: 10.1088/1367-2630/14/10/105011. URL: https://doi.org/10.1088/1367-2630/14/10/105011 (cit. on p. 119).

[232] A. M. Brańczyk *et al.* "Self-calibrating quantum state tomography". In: *New J. Phys.* 14.8 (Aug. 2012), p. 085003. DOI: 10.1088/1367-2630/14/8/085003. URL: https://doi.org/10.1088/1367-2630/14/8/085003 (cit. on p. 119).

[233] C. R. Müller *et al.* "Quantum polarization tomography of bright squeezed light". In: *New J. Phys.* 14.8 (Aug. 2012), p. 085002. DOI: 10.1088/1367-2630/14/8/085002. URL: https://doi.org/10.1088/1367-2630/14/8/085002 (cit. on p. 119).

[234] A. Chiuri *et al.* "Tomographic characterization of correlations in a photonic tripartite state". In: *New J. Phys.* 14.8 (Aug. 2012), p. 085006. DOI: 10.1088/1367-2630/14/8/085006. URL: https://doi.org/10.1088/1367-2630/14/8/085006 (cit. on p. 119).

[235] S Wallentowitz, B. Seifert, and S. Godoy. "Local sampling of the quantum phase-space distribution of a continuous-wave optical beam". In: *New J. Phys.* 14.10 (Oct. 2012), p. 105019. DOI: 10.1088/1367-2630/14/10/105019. URL: https://doi.org/10.1088/1367-2630/14/10/105019 (cit. on p. 119).

[236] C. Sayrin *et al.* "Optimal time-resolved photon number distribution reconstruction of a cavity field by maximum likelihood". In: *New J. Phys.* 14.11 (Nov. 2012), p. 115007. DOI: 10.1088/1367-2630/14/11/115007. URL: https://doi.org/10.1088/1367-2630/14/11/115007 (cit. on p. 119).

[237] S. L. Christensen *et al.* "Toward quantum state tomography of a single polariton state of an atomic ensemble". In: *New J. Phys.* 15.1 (Jan. 2013), p. 015002. DOI: 10.1088/1367-2630/15/1/015002. URL: https://doi.org/10.1088/1367-2630/15/1/015002 (cit. on p. 119).

[238] Roberto Rey-de-Castro *et al.* "Time-resolved quantum process tomography using Hamiltonian-encoding and observable-decoding". In: *New J. Phys.* 15.2 (Feb. 2013), p. 025032. DOI: 10.1088/1367-2630/15/2/025032. URL: https://doi.org/10.1088/1367-2630/15/2/025032 (cit. on p. 119).

[239] M. W. Mitchell *et al.* "Certified quantum non-demolition measurement of material systems". In: *New J. Phys.* 14.8 (Aug. 2012), p. 085021. DOI: 10.1088/1367-2630/14/8/085021. URL: https://doi.org/10.1088/1367-2630/14/8/085021 (cit. on p. 119).

[240] Mădălin Guţă, Theodore Kypraios, and Ian Dryden. "Rank-based model selection for multiple ions quantum tomography". In: *New J. Phys.* 14.10 (Oct.

2012), p. 105002. DOI: 10.1088/1367-2630/14/10/105002. URL: https://doi.org/10.1088/1367-2630/14/10/105002 (cit. on p. 119).

[241] H. Häffner *et al.* "Scalable multiparticle entanglement of trapped ions". In: *Nature* 438.7068 (Dec. 2005), pp. 643–646. DOI: 10.1038/nature04279. URL: https://doi.org/10.1038/nature04279 (cit. on p. 119).

[242] Lijian Zhang *et al.* "Recursive quantum detector tomography". In: *New J. Phys.* 14.11 (Nov. 2012), p. 115005. DOI: 10.1088/1367-2630/14/11/115005. URL: https://doi.org/10.1088/1367-2630/14/11/115005 (cit. on p. 119).

[243] Giorgio Brida *et al.* "Quantum characterization of superconducting photon counters". In: *New J. Phys.* 14.8 (Aug. 2012), p. 085001. DOI: 10.1088/1367-2630/14/8/085001. URL: https://doi.org/10.1088/1367-2630/14/8/085001 (cit. on p. 119).

[244] Aamir Anis and A. I. Lvovsky. "Maximum-likelihood coherent-state quantum process tomography". In: *New J. Phys.* 14.10 (Oct. 2012), p. 105021. DOI: 10.1088/1367-2630/14/10/105021. URL: https://doi.org/10.1088/1367-2630/14/10/105021 (cit. on p. 119).

[245] Shun Watanabe, Ryutaroh Matsumoto, and Tomohiko Uyematsu. "Tomography increases key rates of quantum-key-distribution protocols". In: *Phys. Rev. A* 78 (4 Oct. 2008), p. 042316. DOI: 10.1103/PhysRevA.78.042316. URL: https://link.aps.org/doi/10.1103/PhysRevA.78.042316 (cit. on p. 119).

[246] Richard D. Gill and Serge Massar. "State estimation for large ensembles". In: *Phys. Rev. A* 61 (4 Mar. 2000), p. 042312. DOI: 10.1103/PhysRevA.61.042312. URL: https://link.aps.org/doi/10.1103/PhysRevA.61.042312 (cit. on p. 119).

[247] Jorge Gidi *et al.* *Pseudospectral method for solving PDEs using Matrix Product States*. 2024. DOI: 10.48550/ARXIV.2409.02916. URL: https://arxiv.org/abs/2409.02916 (cit. on pp. 134, 198).

[248] Eric Bonvin *et al.* "State Expansion of a Levitated Nanoparticle in a Dark Harmonic Potential". In: *Physical Review Letters* 132.25 (June 2024), p. 253602. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.132.253602 (cit. on pp. 135, 137).

[249] M. Roda-Llordes *et al.* "Numerical simulation of large-scale nonlinear open quantum mechanics". In: *Phys. Rev. Res.* 6 (1 Mar. 2024), p. 013262. DOI: 10.1103/PhysRevResearch.6.013262. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.6.013262 (cit. on pp. 135, 137, 138).

[250] Sarah K. Leyton and Tobias J. Osborne. *A quantum algorithm to solve nonlinear differential equations*. 2008. DOI: 10.48550/ARXIV.0812.4423. URL: https://arxiv.org/abs/0812.4423 (cit. on p. 135).

[251] Paula García-Molina, Javier Rodríguez-Mediavilla, and Juan José García-Ripoll. "Quantum Fourier analysis for multivariate functions and applications to a class of Schrödinger-type partial differential equations". In: *Phys. Rev. A* 105 (1 Jan. 2022), p. 012433. DOI: 10.1103/PhysRevA.105.012433. URL: https://link.aps.org/doi/10.1103/PhysRevA.105.012433 (cit. on pp. 135, 143).

[252] Javier Gonzalez-Conde *et al.* "Efficient quantum amplitude encoding of polynomial functions". In: *Quantum* 8 (Mar. 2024), p. 1297. ISSN: 2521-327X. DOI: 10.22331/q-2024-03-21-1297. URL: https://doi.org/10.22331/q-2024-03-21-1297 (cit. on pp. 135, 143).

[253] L. Hales and S. Hallgren. "An improved quantum Fourier transform algorithm and applications". In: *Proceedings 41st Annual Symposium on Foundations of*

Computer Science. 2000, pp. 515–525. DOI: 10.1109/SFCS.2000.892139 (cit. on pp. 135, 168).

[254] Sitan Chen *et al.* "The complexity of NISQ". In: *Nature Communications* 14.1 (Sept. 2023), p. 6001. ISSN: 2041-1723. DOI: 10.1038/s41467-023-41217-6 (cit. on p. 135).

[255] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum Algorithm for Linear Systems of Equations". en. In: *Physical Review Letters* 103.15 (Oct. 2009), p. 150502. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.103.150502 (cit. on p. 135).

[256] Dominic W. Berry *et al.* "Quantum Algorithm for Linear Differential Equations with Exponentially Improved Dependence on Precision". en. In: *Communications in Mathematical Physics* 356.3 (Dec. 2017), pp. 1057–1081. ISSN: 1432-0916. DOI: 10.1007/s00220-017-3002-y (cit. on p. 135).

[257] Michael Lubasch *et al.* "Variational quantum algorithms for nonlinear problems". In: *Physical Review A* 101.1 (Jan. 2020). ISSN: 2469-9934. DOI: 10.1103/physreva.101.010301. URL: http://dx.doi.org/10.1103/PhysRevA.101.010301 (cit. on p. 136).

[258] Abhijat Sarma *et al.* "Quantum variational solving of nonlinear and multidimensional partial differential equations". In: *Physical Review A* 109.6 (June 2024). ISSN: 2469-9934. DOI: 10.1103/physreva.109.062616. URL: http://dx.doi.org/10.1103/PhysRevA.109.062616 (cit. on p. 136).

[259] Michael Lubasch, Pierre Moinier, and Dieter Jaksch. "Multigrid renormalization". In: *Journal of Computational Physics* 372 (2018), pp. 587–602. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.06.065. URL: https://www.sciencedirect.com/science/article/pii/S0021999118304431 (cit. on pp. 136, 142).

[260] Paula García-Molina, Luca Tagliacozzo, and Juan José García-Ripoll. *Global optimization of MPS in quantum-inspired numerical analysis*. 2023. arXiv: 2303.09430 [quant-ph] (cit. on pp. 136, 143).

[261] Sebastian Paeckel *et al.* "Time-evolution methods for matrix-product states". In: *Annals of Physics* 411 (2019), p. 167998. ISSN: 0003-4916. DOI: https://doi.org/10.1016/j.aop.2019.167998. URL: https://www.sciencedirect.com/science/article/pii/S0003491619302532 (cit. on pp. 136, 164).

[262] Erika Ye and Nuno F. G. Loureiro. "Quantum-inspired method for solving the Vlasov-Poisson equations". In: *Phys. Rev. E* 106 (3 Sept. 2022), p. 035208. DOI: 10.1103/PhysRevE.106.035208. URL: https://link.aps.org/doi/10.1103/PhysRevE.106.035208 (cit. on pp. 136, 143).

[263] Boris N. Khoromskij. "O(Dlog N)-Quantics Approximation of N-d Tensors in High-Dimensional Numerical Modeling". In: *Constructive Approximation* 34.2 (Oct. 2011), pp. 257–280. ISSN: 0176-4276, 1432-0940. DOI: 10.1007/s00365-011-9131-1. (Visited on 11/24/2023) (cit. on p. 136).

[264] Samuel M. Greene and Victor S. Batista. "Tensor-Train Split-Operator Fourier Transform (TT-SOFT) Method: Multidimensional Nonadiabatic Quantum Dynamics". In: *Journal of Chemical Theory and Computation* 13.9 (2017). PMID: 28763215, pp. 4034–4042. DOI: 10.1021/acs.jctc.7b00608. eprint: https://doi.org/10.1021/acs.jctc.7b00608. URL: https://doi.org/10.1021/acs.jctc.7b00608 (cit. on pp. 136, 142).

[265] Ningyi Lyu, Micheline B. Soley, and Victor S. Batista. "Tensor-Train Split-Operator KSL (TT-SOKSL) Method for Quantum Dynamics Simulations". In: *Journal of Chemical Theory and Computation* 18.6 (2022). PMID: 35649210, pp. 3327–3346. DOI: 10.1021/acs.jctc.2c00209. eprint: https://doi.org/10.1021/acs.jctc.2c00209. URL: https://doi.org/10.1021/acs.jctc.2c00209 (cit. on pp. 136, 142).

[266] Micheline B. Soley *et al.* "Functional Tensor-Train Chebyshev Method for Multidimensional Quantum Dynamics Simulations". In: *Journal of Chemical Theory and Computation* 18.1 (2022). PMID: 34898201, pp. 25–36. DOI: 10.1021/acs.jctc.1c00941. eprint: https://doi.org/10.1021/acs.jctc.1c00941. URL: https://doi.org/10.1021/acs.jctc.1c00941 (cit. on pp. 136, 142).

[267] David K. Hoffman *et al.* "Analytic banded approximation for the discretized free propagator". In: *The Journal of Physical Chemistry* 95.21 (Oct. 1991), pp. 8299–8305. ISSN: 0022-3654, 1541-5740. DOI: 10.1021/j100174a052 (cit. on pp. 136, 153, 157, 159).

[268] David K. Hoffman and Donald J. Kouri. "Distributed approximating function theory: a general, fully quantal approach to wave propagation". In: *The Journal of Physical Chemistry* 96.3 (Feb. 1992), pp. 1179–1184. ISSN: 0022-3654. DOI: 10.1021/j100182a030 (cit. on pp. 136, 157).

[269] David K. Hoffman, Mark Arnold, and Donald J. Kouri. "Properties of the optimum distributed approximating function class propagator for discretized and continuous wave packet propagations". In: *The Journal of Physical Chemistry* 96.16 (Aug. 1992), pp. 6539–6545. ISSN: 0022-3654. DOI: 10.1021/j100195a007 (cit. on pp. 136, 158).

[270] David Hoffman and Donald Kouri. "Distributed approximating functionals: A new approach to approximating functions and their derivatives". In: *Third International Conference on Mathematical and Numerical Aspects of Wave Propagation SIAM* (Jan. 1995) (cit. on pp. 136, 153, 159).

[271] D. K. Hoffman *et al.* "Variational derivation and extensions of distributed approximating functionals". In: *Journal of Mathematical Chemistry* 20.1 (Mar. 1996), pp. 117–140. ISSN: 0259-9791, 1572-8897. DOI: 10.1007/BF01165159.

URL: http://link.springer.com/10.1007/BF01165159 (visited on 06/19/2024) (cit. on p. 136).

[272] Bernhard G. Bodmann *et al.* "Hermite Distributed Approximating Functionals as Almost-Ideal Low-Pass Filters". In: *Sampling Theory in Signal and Image Processing* 7.1 (Jan. 2008), pp. 15–38. ISSN: 1530-6429. DOI: 10.1007/BF03549483 (cit. on pp. 136, 160).

[273] J. A. C. Weideman and B. M. Herbst. "Split-Step Methods for the Solution of the Nonlinear Schrödinger Equation". In: *SIAM Journal on Numerical Analysis* 23.3 (1986), pp. 485–507. ISSN: 0036-1429 (cit. on p. 137).

[274] C. Gonzalez-Ballestero *et al.* "Levitodynamics: Levitation and control of microscopic objects in vacuum". In: *Science* 374.6564 (2021), eabg3027. DOI: 10.1126/science.abg3027 (cit. on pp. 137, 140, 179).

[275] Juan J. G. Ripoll and Víctor M. Pérez-García. "Barrier resonances in Bose-Einstein condensation". In: *Phys. Rev. A* 59 (3 Mar. 1999), pp. 2220–2231. DOI: 10.1103/PhysRevA.59.2220. URL: https://link.aps.org/doi/10.1103/PhysRevA.59.2220 (cit. on pp. 137, 138).

[276] Aditi Mitra. "Quantum Quench Dynamics". In: *Annual Review of Condensed Matter Physics* 9.1 (Mar. 2018), pp. 245–259. ISSN: 1947-5462. DOI: 10.1146/annurev-conmatphys-031016-025451. URL: http://dx.doi.org/10.1146/annurev-conmatphys-031016-025451 (cit. on p. 138).

[277] Sumit R. Das. *Quantum Quench and Universal Scaling*. Dec. 2020. DOI: 10.1093/acrefore/9780190871994.013.55. URL: https://oxfordre.com/physics/view/10.1093/acrefore/9780190871994.001.0001/acrefore-9780190871994-e-55 (cit. on p. 138).

[278] Pasquale Calabrese and John Cardy. "Time Dependence of Correlation Functions Following a Quantum Quench". In: *Phys. Rev. Lett.* 96 (13 Apr. 2006),

p. 136801. DOI: 10.1103/PhysRevLett.96.136801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.96.136801 (cit. on p. 138).

[279] A. Iucci and M. A. Cazalilla. "Quantum quench dynamics of the Luttinger model". In: *Phys. Rev. A* 80 (6 Dec. 2009), p. 063619. DOI: 10.1103/PhysRevA.80.063619. URL: https://link.aps.org/doi/10.1103/PhysRevA.80.063619 (cit. on p. 138).

[280] Pasquale Calabrese, Fabian H. L. Essler, and Maurizio Fagotti. "Quantum Quench in the Transverse-Field Ising Chain". In: *Phys. Rev. Lett.* 106 (22 June 2011), p. 227203. DOI: 10.1103/PhysRevLett.106.227203. URL: https://link.aps.org/doi/10.1103/PhysRevLett.106.227203 (cit. on p. 138).

[281] Fabian H. L. Essler, Stefano Evangelisti, and Maurizio Fagotti. "Dynamical Correlations After a Quantum Quench". In: *Phys. Rev. Lett.* 109 (24 Dec. 2012), p. 247206. DOI: 10.1103/PhysRevLett.109.247206. URL: https://link.aps.org/doi/10.1103/PhysRevLett.109.247206 (cit. on p. 138).

[282] Ashley Setter, Jamie Vovrosh, and Hendrik Ulbricht. "Characterization of non-linearities through mechanical squeezing in levitated optomechanics". In: *Applied Physics Letters* 115.15 (Oct. 2019). _eprint: https://pubs.aip.org/aip/apl/article-pdf/doi/10.1063/1.5116121/14528387/153106_1_online.pdf, p. 153106. ISSN: 0003-6951. DOI: 10.1063/1.5116121. URL: https://doi.org/10.1063/1.5116121 (cit. on pp. 140, 179).

[283] Lukas Neumeier *et al.* "Fast quantum interference of a nanoparticle via optical potential control". In: *Proceedings of the National Academy of Sciences* 121.4 (2024), e2306953121. DOI: 10.1073/pnas.2306953121. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2306953121. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2306953121 (cit. on pp. 140, 179).

[284] M. Roda-Llordes *et al.* "Macroscopic Quantum Superpositions via Dynamics in a Wide Double-Well Potential". In: *Phys. Rev. Lett.* 132 (2 Jan. 2024), p. 023601. DOI: 10.1103/PhysRevLett.132.023601. URL: https://link.aps.org/doi/10.1103/PhysRevLett.132.023601 (cit. on pp. 140, 179).

[285] Alec Dektor, Abram Rodgers, and Daniele Venturi. "Rank-Adaptive Tensor Methods for High-Dimensional Nonlinear PDEs". In: *Journal of Scientific Computing* 88.2 (2021), p. 36. ISSN: 1573-7691. DOI: 10.1007/s10915-021-01539-3. URL: https://doi.org/10.1007/s10915-021-01539-3 (cit. on p. 142).

[286] Matanya B. Horowitz, Anil Damle, and Joel W. Burdick. "Linear Hamilton Jacobi Bellman Equations in high dimensions". In: *53rd IEEE Conference on Decision and Control*. IEEE, Dec. 2014. DOI: 10.1109/cdc.2014.7040310. URL: https://doi.org/10.11092Fcdc.2014.7040310 (cit. on p. 142).

[287] Elis Stefansson and Yoke Peng Leong. "Sequential alternating least squares for solving high dimensional linear Hamilton-Jacobi-Bellman equation". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2016, pp. 3757–3764. DOI: 10.1109/IROS.2016.7759553 (cit. on p. 142).

[288] Alex Gorodetsky, Sertac Karaman, and Youssef Marzouk. "High-dimensional stochastic optimal control using continuous tensor decompositions". In: *The International Journal of Robotics Research* 37.2-3 (2018), pp. 340–377. DOI: 10.1177/0278364917753994. eprint: https://doi.org/10.1177/0278364917753994. URL: https://doi.org/10.1177/0278364917753994 (cit. on p. 142).

[289] Sergey Dolgov, Dante Kalise, and Karl K. Kunisch. "Tensor Decomposition Methods for High-dimensional Hamilton–Jacobi–Bellman Equations". In: *SIAM Journal on Scientific Computing* 43.3 (2021), A1625–A1650. DOI: 10.

1137/19M1305136. eprint: https://doi.org/10.1137/19M1305136. URL: https://doi.org/10.1137/19M1305136 (cit. on p. 142).

[290] Mathias Oster, Leon Sallandt, and Reinhold Schneider. *Approximating the Stationary Bellman Equation by Hierarchical Tensor Products*. 2019. DOI: 10.48550/ARXIV.1911.00279. URL: https://arxiv.org/abs/1911.00279 (cit. on p. 142).

[291] Rui Hong *et al.* "Functional tensor network solving many-body Schrödinger equation". In: *Phys. Rev. B* 105 (16 Apr. 2022), p. 165116. DOI: 10.1103/PhysRevB.105.165116. URL: https://link.aps.org/doi/10.1103/PhysRevB.105.165116 (cit. on p. 142).

[292] Lorenz Richter, Leon Sallandt, and Nikolas Nüsken. *Solving high-dimensional parabolic PDEs using the tensor train format*. 2021. DOI: 10.48550/ARXIV.2102.11830. URL: https://arxiv.org/abs/2102.11830 (cit. on p. 142).

[293] Sergey Dolgov *et al.* "Polynomial Chaos Expansion of Random Coefficients and the Solution of Stochastic Partial Differential Equations in the Tensor Train Format". In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1 (Jan. 2015), pp. 1109–1135. DOI: 10.1137/140972536. URL: https://doi.org/10.11372F140972536 (cit. on p. 142).

[294] Martin Eigel, Max Pfeffer, and Reinhold Schneider. "Adaptive stochastic Galerkin FEM with hierarchical tensor representations". In: *Numerische Mathematik* 136.3 (Nov. 2016), pp. 765–803. DOI: 10.1007/s00211-016-0850-x. URL: https://doi.org/10.10072Fs00211-016-0850-x (cit. on p. 142).

[295] N Gourianov. "Exploiting the structure of turbulence with tensor networks". PhD thesis. University of Oxford, 2022 (cit. on p. 142).

[296] Ivan Oseledets and Eugene Tyrtyshnikov. "TT-cross approximation for multidimensional arrays". In: *Linear Algebra and its Applications* 432.1 (2010), pp. 70–88. ISSN: 0024-3795. DOI: https://doi.org/10.1016/j.laa.2009.07.024. URL: https://www.sciencedirect.com/science/article/pii/S0024379509003747 (cit. on pp. 149, 151).

[297] Sergey Dolgov and Dmitry Savostyanov. "Parallel cross interpolation for high-precision calculation of high-dimensional integrals". In: *Computer Physics Communications* 246 (2020), p. 106869. ISSN: 0010-4655. DOI: https://doi.org/10.1016/j.cpc.2019.106869. URL: https://www.sciencedirect.com/science/article/pii/S0010465519302565 (cit. on pp. 149, 151).

[298] Juan José Rodríguez-Aldavero *et al.* *Chebyshev approximation and composition of functions in matrix product states for quantum-inspired numerical analysis*. 2024. arXiv: 2407.09609 [quant-ph]. URL: https://arxiv.org/abs/2407.09609 (cit. on pp. 149, 151, 168).

[299] Pavel Holoborodko. *Smooth Noise Robust Differentiators*. http://www.holoborodko.com/pavel/numerical-methods/numerical-derivative/smooth-low-noise-differentiators/. 2008 (cit. on pp. 151, 170).

[300] Nicole DeGregorio and Srinivasan S. Iyengar. "Adaptive Dimensional Decoupling for Compression of Quantum Nuclear Wave Functions and Efficient Potential Energy Surface Representations through Tensor Network Decomposition". In: *Journal of Chemical Theory and Computation* 15.5 (May 2019), pp. 2780–2796. ISSN: 1549-9618. DOI: 10.1021/acs.jctc.8b01113 (cit. on p. 152).

[301] Colston Chandler and Archie G. Gibson. "Uniform Approximation of Functions with Discrete Approximation Functionals". In: *Journal of Approximation Theory* 100.2 (Oct. 1999), pp. 233–250. ISSN: 00219045. DOI: 10.1006/jath.1999.3325 (cit. on p. 153).

[302] E. T. Goodwin. "The evaluation of integrals of the form". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 45.2 (Apr. 1949), pp. 241–245. ISSN: 0305-0041, 1469-8064. DOI: 10.1017/S0305004100024786 (cit. on p. 154).

[303] Norman Dyson. "Peak distortion, data sampling errors and the integrator in the measurement of very narrow chromatographic peaks". In: *Journal of Chromatography A* 842.1 (May 1999), pp. 321–340. ISSN: 0021-9673. DOI: 10.1016/S0021-9673(99)00299-X (cit. on p. 154).

[304] J. A. C. Weideman. "Numerical Integration of Periodic Functions: A Few Examples". In: *The American Mathematical Monthly* 109.1 (Jan. 2002), pp. 21–36. ISSN: 0002-9890, 1930-0972. DOI: 10.1080/00029890.2002.11919836 (cit. on p. 154).

[305] Edson Pindza and Eben Maré. "Solving the Generalized Regularized Long Wave Equation Using a Distributed Approximating Functional Method". In: *International Journal of Computational Mathematics* 2014 (Aug. 2014), pp. 1–12. ISSN: 2356-797X, 2314-856X. DOI: 10.1155/2014/178024 (cit. on p. 158).

[306] Thomas Mazzoni. "A functional approach to pricing complex barrier options". In: *The European Journal of Finance* 20.5 (May 2014), pp. 399–418. ISSN: 1351-847X, 1466-4364. DOI: 10.1080/1351847X.2012.713174 (cit. on p. 159).

[307] S.J.L. van Eijndhoven and J.L.H. Meyers. "New orthogonality relations for the Hermite polynomials and related Hilbert spaces". In: *Journal of Mathematical Analysis and Applications* 146.1 (Feb. 1990), pp. 89–98. ISSN: 0022-247X. DOI: 10.1016/0022-247x(90)90334-c. URL: http://dx.doi.org/10.1016/0022-247X(90)90334-C (cit. on p. 163).

[308] Jutho Haegeman *et al.* "Time-Dependent Variational Principle for Quantum Lattices". In: *Physical Review Letters* 107.7 (Aug. 2011). Publisher: American

Physical Society (APS). DOI: 10.1103/physrevlett.107.070601. URL: https://doi.org/10.1103%2Fphysrevlett.107.070601 (cit. on p. 164).

[309] Laurens Vanderstraeten, Jutho Haegeman, and Frank Verstraete. "Tangent-space methods for uniform matrix product states". In: *SciPost Physics Lecture Notes* (Jan. 2019). Publisher: Stichting SciPost. DOI: 10.21468/scipostphyslectnotes.7. URL: https://doi.org/10.21468scipostphyslectnotes.7 (cit. on p. 164).

[310] Juan José García-Ripoll. "Time evolution of Matrix Product States". In: *New Journal of Physics* 8.12 (Dec. 2006), p. 305. DOI: 10.1088/1367-2630/8/12/305. URL: https://dx.doi.org/10.1088/1367-2630/8/12/305 (cit. on p. 164).

[311] Haruo Yoshida. "Symplectic Integrators for Hamiltonian Systems: Basic Theory". In: *Symposium - International Astronomical Union* 152 (1992), pp. 407–411. DOI: 10.1017/S0074180900091440 (cit. on pp. 168, 176).

[312] Masuo Suzuki. "Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations". In: *Physics Letters A* 146.6 (1990), pp. 319–323. ISSN: 0375-9601. DOI: https://doi.org/10.1016/0375-9601(90)90962-N. URL: https://www.sciencedirect.com/science/article/pii/037596019090962N (cit. on p. 168).

[313] Masuo Suzuki. "General theory of fractal path integrals with applications to many-body theories and statistical physics". In: *Journal of Mathematical Physics* 32.2 (Feb. 1991). _eprint: https://pubs.aip.org/aip/jmp/article-pdf/32/2/400/19166143/400_1_online.pdf, pp. 400–407. ISSN: 0022-2488. DOI: 10.1063/1.529425. URL: https://doi.org/10.1063/1.529425 (cit. on p. 168).

[314] R I McLachlan and P Atela. "The accuracy of symplectic integrators". In: *Nonlinearity* 5.2 (Mar. 1992), pp. 541–562. ISSN: 1361-6544. DOI: 10.1088/

0951-7715/5/2/011. URL: http://dx.doi.org/10.1088/0951-7715/5/2/011 (cit. on p. 168).

[315] B. Cano and J. M. Sanz-Serna. "Error Growth in the Numerical Integration of Periodic Orbits, with Application to Hamiltonian and Reversible Systems". In: *SIAM Journal on Numerical Analysis* 34.4 (Aug. 1997), pp. 1391–1417. ISSN: 0036-1429, 1095-7170. DOI: 10.1137/S0036142995281152 (cit. on p. 176).

[316] Andrew Stuart and A R Humphries. *Cambridge monographs on applied and computational mathematics: Dynamical systems and numerical analysis series number 2*. Cambridge, England: Cambridge University Press, Nov. 1998 (cit. on p. 176).

[317] Hugo A. Carril *et al.* "Formation of multiple BGK-like structures in the time-asymptotic state of collisionless Vlasov-Poisson plasmas". In: *Phys. Rev. E* 107 (6 June 2023), p. 065203. DOI: 10.1103/PhysRevE.107.065203. URL: https://link.aps.org/doi/10.1103/PhysRevE.107.065203 (cit. on p. 197).