Variational Quantum Solutions for the Learning with Errors Problem and Implications on Post-Quantum Cryptography

Sava Kazakov

Master of Science

Informatics

School of Informatics

University of Edinburgh

2024

Abstract

This dissertation explores the application of NISQ-era quantum algorithms, particularly the Quantum Approximate Optimization Algorithm (QAOA) and the Variational Quantum Eigensolver (VQE), to the cryptanalysis of lattice-based cryptosystems, focusing on the Learning With Errors (LWE) problem and Kyber CRYSTALS. The research addresses the challenge of mapping LWE instances into Hamiltonians suitable for quantum optimization, proposing novel encodings tailored to the limitations of current quantum hardware. Extensive simulations using Qiskit evaluate these algorithms under realistic noise models, analyzing their effectiveness against small-scale LWE instances with varying error distributions. The findings indicate that QAOA shows potential for noise-free, small LWE instances but is highly sensitive to noise, leading to potential performance degradation. Conversely, VQE, enhanced with Conditional Value at Risk (CVaR) strategies, demonstrates greater robustness and consistency, making it more suitable for cryptanalysis under noisy conditions, although with a bias towards conservative solutions. The study contributes to understanding the practicality of quantum cryptanalysis in the NISQ era, highlighting the scalability challenges and the current limitations of these quantum approaches in breaking high-dimensional cryptosystems. While large-scale attacks on cryptosystems like Kyber remain infeasible with current technology, this research lays foundational groundwork for future studies as quantum hardware advances.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Sava Kazakov*)

Acknowledgements

I would like to express my deepest gratitude to Professor Petros Wallden for his invaluable guidance, insightful advice, and unwavering support throughout this project. His expertise and profound understanding of the subject have been instrumental in shaping the direction of my research. I am especially grateful for his patience, kindness, and encouragement, which have motivated me to persevere even in challenging times.

I would also like to extend my sincere thanks to Ioannis Kolotouros, whose assistance and readiness to answer my questions have been greatly appreciated. His willingness to offer help and share his knowledge has made a significant impact on my work, and I am truly thankful for his generosity with his time and effort.

Table of Contents

Chapter 1

Introduction

1.1 Context of the Study

Cryptography, which is fundamentally concerned with the limits of computational efficiency, saw a significant breakthrough in 1976 when Diffie and Hellman [30] introduced public key cryptography. This advancement established secure communication protocols based on computational problems like the discrete logarithm and integer factorization [30]. However, Shor's algorithm (1994) demonstrated that these problems could be efficiently solved using quantum computers, threatening traditional cryptographic systems [81]. This has catalyzed the development of post-quantum cryptography (PQC), which seeks to design cryptographic schemes resilient to quantum attacks while remaining feasible on classical architectures. The recent National Institute of Standards and Technology (NIST) PQC standardization process has identified lattice-based cryptography as a leading candidate, notably due to its reliance on hard problems like the Learning With Errors (LWE) problem, which are believed to be secure against both classical and quantum adversaries [3].

Central to post-quantum cryptographic research are cryptosystems based on lattice problems, such as the CRYSTALS-Kyber system, which is built upon variants of the LWE problem. These cryptosystems have gained attention because their underlying mathematical structures remain hard to break even with the capabilities of quantum algorithms [18]. Specifically, module lattice schemes balance computational efficiency and robust security, which positions them as promising candidates in the quantum-resilient cryptographic landscape [76, 15].

The advent of Noisy Intermediate-Scale Quantum (NISQ) devices has prompted interest in hybrid quantum-classical approaches, particularly Variational Quantum Algo-

rithms (VQAs), such as the Variational Quantum Eigensolver (VQE) and the Quantum Approximate Optimization Algorithm (QAOA). VQAs are particularly relevant because they can harness the limited qubit quality and coherence times available in current NISQ devices while still achieving meaningful results. By leveraging a combination of quantum and classical resources, VQAs can optimize cryptographic functions by encoding these functions into problem Hamiltonians that are minimized via quantum techniques [27]. Given their adaptability and reduced quantum resource requirements, VQAs offer practical approaches for tackling cryptographic problems, in the NISQ era.

Quantum computing's ability to leverage principles like superposition, entanglement, and quantum interference has led to exponential speed-ups for certain classes of problems compared to classical methods [86]. Recent advancements include more stable qubit designs, error correction techniques, and improved quantum hardware, such as superconducting qubits and trapped ion systems, all of which contribute to the growing viability of quantum computing in practical cryptanalysis [6, 87].

This research explores the application of VQE and QAOA to cryptanalysis, focusing on breaking LWE-based cryptosystems like CRYSTALS-Kyber. Specifically, the study examines how to map the LWE problem into Hamiltonians suitable for quantum optimization and subsequently derive Quadratic Unconstrained Binary Optimization (QUBO) formulations for use with QAOA. This involves simulating these mappings using Qiskit to assess the feasibility and performance of these approaches in cryptanalysis [72]. By concentrating on the LWE problem and its quantum optimization strategies, the study aims to provide insights into the effectiveness of near-term quantum algorithms for cryptanalysis and their implications for post-quantum cryptographic security. This work not only contributes to validating the robustness of PQC algorithms but also highlights the potential of hybrid quantum-classical approaches in real-world cryptanalytic applications [37].

1.2 Motivation

The motivation for this research is rooted in the importance of Hamiltonian ground state problems in quantum computing. These problems are crucial across multiple scientific disciplines, including condensed matter physics, materials science, and optimization, where finding ground states is central to solving complex models and understanding material properties. These problems are particularly challenging for classical systems, raising critical questions about the capabilities of quantum algorithms in addressing

cryptographic challenges. In the context of post-quantum cryptography, lattice-based cryptographic protocols, such as those relying on the LWE problem, form the foundation of systems like CRYSTALS-Kyber.

While these systems demonstrate robustness against classical attacks, their resistance to quantum attacks is less established due to the relatively recent emergence of structured Hamiltonians in cryptanalysis. Structured Hamiltonians refer to problem instances that exhibit symmetries or other patterns, which can potentially be exploited by quantum algorithms. Since lattice-based systems like Kyber have not been exposed to extensive cryptanalysis over decades like RSA, their conjectured hardness is based on assumptions rather than long-term empirical evidence. This conjectured difficulty is analogous to other cryptographic systems, such as the NIST finalist Rainbow, which was believed to be secure until it was recently broken by a classical approach [5].

The arrival of NISQ devices, characterized by limited qubit numbers (typically between 50 and 1000 qubits) and non-error-corrected operations with high noise levels, prompts the urgent question of whether such systems can be leveraged to break LWE-based cryptosystems. NISQ devices operate within a constrained regime where fully fault-tolerant quantum computing is not yet achievable. Current estimates suggest that fault-tolerant quantum computing may be reached within the next 10-20 years, requiring millions of qubits and robust error correction [70]. This timeline underscores the importance of assessing the performance of early quantum devices, particularly in evaluating cryptographic schemes like Kyber.

Currently, numerous governments and individuals are engaged in intercepting and storing encrypted data, such as passwords and private communications, with the anticipation of decrypting them in the future. This approach, termed *Store Now, Decrypt Later* [82, 30, 76], is based on the expectation that quantum computers, which are predicted to become operational within 10 to 20 years, will be capable of rapidly breaking widely used cryptosystems [23]. A prominent example is RSA, an asymmetric key cryptosystem whose security relies on the difficulty of factoring large numbers, a problem classified as NP [60, 74]. Although decoding a message with a key is straightforward, brute-forcing it remains infeasible with classical methods, such as the General Number Field Sieve, even on supercomputers [78]. For instance, factoring standardized prime numbers (approximately 313 digits) would take around 16 million years classically [11]. However, Shor's algorithm on a quantum computer offers an exponential speed-up [81].

In 2012, it was estimated that breaking RSA encryption required a billion physical qubits, but this figure was revised to 230 million in 2017 and further to 20 million by

2019 [36]. Despite the rapid advancements in quantum hardware, current capabilities remain insufficient, though it is anticipated that this gap will eventually close. In response to this looming threat, the National NIST initiated a competition in 2016 to identify encryption algorithms resistant to quantum attacks [76]. By 2022, NIST had selected four finalists for the PQC standard, highlighting the urgency of evaluating their resilience against quantum threats in both academic and industrial contexts and only recently they started the standardization process [71].

1.2.0.0.1 Problem statement The primary research problem addressed in this study is the development and evaluation of practical quantum algorithms for cryptanalysis, focusing on NISQ methods for attacking LWE-based cryptosystems like Regev and CRYSTALS-Kyber. Specifically, this research explores how Hamiltonians representing LWE instances can be encoded on simulated NISQ devices and investigates the benefits and drawbacks of this approach. Moreover, the study evaluates how variational methods like QAOA and VQE differ in their strategies for solving LWE problems compared to classical algorithms, considering space complexity, runtime, and the potential for optimization within the constraints of current quantum hardware.

1.2.0.0.2 Research questions This study seeks to answer the following key questions:

- Can LWE-related problems be directly simulated and encoded using Hamiltonians on simulated NISQ devices? What are the specific approaches, benefits, and challenges in doing so?
- How do the complexities and scaling behaviors of quantum algorithms for LWE, such as QAOA and VQE, compare to classical algorithms in terms of performance on simulated NISQ devices?
- What are the implications of space complexity and scalability for implementing LWE-related computations on simulated hardware, particularly in relation to Kyber?
- How can these quantum algorithms be managed and tested using quantum frameworks like Qiskit, given the constraints of NISQ-era hardware?
- What distinct advantages or insights can VQE and QAOA provide in relation to each other, considering the indirect nature of some current quantum cryptanalysis

attempts?

Addressing these questions will contribute to a more nuanced understanding of NISQ-era quantum computing's impact on post-quantum cryptography and could guide the development of more resilient cryptographic systems as quantum technology evolves.

1.3 Research objectives

This research investigates the application of the QAOA and VQE algorithms, for cryptanalysis of LWE-based cryptosystems. The focus is on evaluating how quantum algorithms challenge LWE's hardness assumptions, which are foundational to schemes like CRYSTALS-Kyber [18, 72]. The aim is to assess the practical limits of NISQ devices in solving LWE and extend these findings to more complex systems like Kyber.

1. **Security of LWE in Quantum Settings:** This research evaluates LWE vulnerability by focusing on how LWE can be mapped into Hamiltonian optimization problems. While CRYSTALS-Kyber is not the primary focus, insights gained from LWE cryptanalysis will be extrapolated to assess Kyber's resilience, given its reliance on Module Learning With Errors (MLWE) [18, 72].

2. **Optimizing Quantum Algorithm Parameters:** The research compares quantum algorithms across different parameters, including optimizers, mappings, and implementations. Both hand-coded and Qiskit implementations are tested, with focus on noise resilience and realistic gate operations. Hyperparameters such as CVaR (Conditional Value at Risk) optimization are explored [15, 27].

3. **Implications for Quantum-Resistant Cryptography:** The study explores the practical limits of quantum attacks. While specific recommendations for Kyber are beyond scope, verbal conjectures based on LWE cryptanalysis and considerations of Ring and Module LWE are discussed as a stretch goal for guiding future research [37, 81].

4. **Simulation and Benchmarking:** Extensive simulation and testing in Qiskit focus on VQAs for LWE under realistic noise models. The study benchmarks quantum algorithms against classical methods, including quantum-inspired classical algorithms, to establish a robust cryptanalysis framework for NISQ devices [27, 83].

5. **Contributing to Quantum Cryptanalysis Research:** This research bridges theoretical cryptanalysis and practical implementation. By documenting methodologies, results, and analyses, it provides a foundation for future work in quantum cryptanalysis, focusing on current hardware limitations and future advancements [86, 62].

1.4 Scope and limitations

The use of QAOA, while promising, is hindered by scalability issues. The qubit complexity grows as $O(n \log n)$, where n denotes the problem size. This growth makes QAOA challenging to apply to larger instances, particularly given that lattice-based cryptographic constructions require dimensions on the order of $O(100)$. To explore larger problem sizes, VQE was employed as it allows for the simulation of a generalized QAOA Hamiltonian without an encoded modulo function. While VQE offers more flexibility in problem scaling, our experiments revealed that its optimization landscapes are less tractable than those of QAOA, impacting overall efficiency.

The LWE instances in this study use a modulus q approximating n^2, typically set near the highest power of 2 within $n^2 - 2n^2$. The error distribution is sometimes simplified to a ternary set of $\{-1, 0, 1\}$ for computational ease. Despite this simplification, results are compared against distributions like Regev's χ distribution to assess computational differences. While the ternary approximation aids analysis, it limits the results' applicability to real-world LWE distributions.

Computational constraints significantly influenced this research, with most simulations limited to classical devices handling up to 25 qubits. Experiments involving 30 qubits were conducted but required hours, underscoring the challenges of scaling.

The choice of Qiskit as the simulation framework further restricts the study. While Qiskit offers standardized optimizers and hyperparameters, these predefined options limit algorithmic exploration. Additionally, the supported noise models and quantum circuits do not fully represent the behavior of actual quantum devices, thereby affecting the results' fidelity.

1.5 Contribution

We recognize the following contributions to the literature:

- Proposed and developed two novel Hamiltonian encodings for the LWE problem: one optimized for the VQE and another for the QAOA. These encodings are designed to efficiently leverage quantum resources while maintaining the hardness properties crucial for cryptographic analysis.

- Provided rigorous proofs of correctness for both Hamiltonians, demonstrating their validity in modeling the LWE problem. The VQE encoding incorporates a generalized cost function and supports diverse error distributions, while the QAOA encoding introduces a centered modulo operation for enhanced precision in quantum cryptanalysis.

- Introduced a novel modulo encoding scheme using auxiliary variables and penalty functions for LWE cryptanalysis in quantum algorithms, addressing issues related to error propagation and correctness when performing modular arithmetic on quantum devices. This encoding allows for more accurate representation of the LWE problem's structure.

- Conducted extensive experiments involving CVaR optimization strategies, including a newly designed ascending CVaR technique. This approach dynamically refines optimization targets, balancing exploration and exploitation, resulting in improved convergence and robustness against quantum noise.

- Performed a comparative analysis of three key approaches: VQE, QAOA, and the ascending CVaR method. This analysis identified strengths and limitations in both algorithmic performance and scalability for cryptanalysis tasks, providing detailed insights into the suitability of each method under realistic noise conditions.

- Implemented a modular and extensible `LWE` class for cryptanalysis research, enabling flexible exploration of security parameters. The class supports multiple LWE variants and error distributions, offering a versatile tool for both theoretical analysis and practical experimentation.

- Demonstrated the feasibility of attacking small-scale LWE instances using quantum algorithms, highlighting the conditions under which these approaches are most effective. The research findings indicate the potential of NISQ devices in breaking cryptographic schemes, while also identifying the current hardware limitations that must be overcome.

- Provided an in-depth analysis of the space complexity for the proposed quantum approaches, proving that the encoding schemes require $O(n \log n)$ qubits, making them scalable within the constraints of current and near-term quantum hardware.

This contributions list summarizes the significant and original aspects of this research, emphasizing both theoretical advancements and practical implementations in the context of quantum cryptanalysis.

1.6 Structure of the dissertation

The structure of this dissertation follows a logical progression, beginning with foundational concepts and culminating in detailed research findings and analysis. The Introduction establishes the context, motivation, and objectives of the study, framing the research within the growing need to assess the resilience of LWE-based cryptosystems like Kyber in the face of quantum computing advancements. Building on this, the Background explores the theoretical underpinnings, detailing lattice-based cryptography, the LWE problem, and key quantum algorithms (VQE and QAOA) that are pivotal to the research. Subsequently, the Research methodology outlines the experimental design and implementation, including the proposed Hamiltonian encodings and the use of Qiskit simulations to test the algorithms in a cryptanalytic context. The findings are then presented in the Results and analysis, where the performance of VQE and QAOA is compared across different LWE configurations, highlighting critical aspects such as scalability, noise resilience, and solution accuracy. This is followed by the Discussion, which interprets the results in relation to the broader field of post-quantum cryptography, emphasizing theoretical contributions and practical implications. Finally, the Conclusion summarizes the study's key insights and offers recommendations for future research, underscoring the importance of continued exploration as quantum hardware and cryptanalytic techniques evolve.

Chapter 2

Background

2.1 Preliminaries

This section introduces key mathematical concepts and notations relevant to quantum algorithms in cryptanalysis, focusing on lattice-based schemes such as Kyber CRYSTALS.

2.1.1 Notation for LWE

The Learning with Errors (LWE) problem is foundational in lattice-based cryptography. It is defined as follows: Given a modulus q, a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$, and a noise term $\mathbf{e}$ drawn from a discrete Gaussian distribution χ, the LWE problem asks to distinguish the distribution of the noisy linear combination $\mathbf{As} + \mathbf{e}$ (where $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ is a random matrix) from a uniform distribution over $\mathbb{Z}_q^m$ [73]. Mathematically, the decisional LWE problem can be expressed as:

$$(\mathbf{A}, \mathbf{As} + \mathbf{e}) \approx \text{Uniform}(\mathbb{Z}_q^m).$$

In addition to the decisional problem, the search version of LWE seeks to recover the secret vector $\mathbf{s}$ given the samples $(\mathbf{A}, \mathbf{As} + \mathbf{e})$. The error distribution χ, typically a discrete Gaussian $\mathcal{D}_\sigma$, is critical in maintaining the problem's hardness [89, 28]. It is also important to define the exact meaning of the modulo operation, which we define to be:

$$a \mod q := a - \left\lfloor \frac{a}{q} \right\rfloor q. \tag{2.1}$$

The LWE assumption underpins the security of various cryptosystems and is believed to resist quantum attacks due to its connection with lattice problems like the Shortest Vector Problem (SVP) [59, 67, 54].

2.1.2 Rings and Ring-LWE Notation

Efficient LWE variants, such as Ring-LWE (RLWE), leverage ring structures. In RLWE, the ring $R = \mathbb{Z}[x]/(f(x))$, with $f(x) = x^n + 1$ where n is a power of two, serves as the algebraic framework. The problem shifts from vector spaces to ideal lattices, enabling more efficient operations. For a quotient ring $R_q = R/qR$, the decisional RLWE problem involves distinguishing samples $(a(x), a(x) \cdot s(x) + e(x))$, where $a(x)$ is uniformly sampled from R_q and $e(x)$ is small noise drawn from a Gaussian distribution over the ring, from random samples in R_q [67, 54]. The ring homomorphisms in RLWE reduce computational complexity from $O(n^2)$ to $O(n \log n)$, improving efficiency while maintaining the hardness assumptions.

2.1.3 Modulo-LWE and Quantum Notation

Modulo-LWE (MLWE) extends LWE by incorporating modular arithmetic over structured rings. It maintains LWE's hardness assumptions while enabling more efficient polynomial arithmetic, particularly relevant in cryptosystems like Kyber. The algebraic structure of MLWE, defined modulo both an integer q and a polynomial $f(x)$, allows operations on structured lattices that are not feasible in standard LWE. This flexibility supports more efficient cryptographic schemes, especially under hybrid classical-quantum models [67, 54, 70]. The algebraic properties of MLWE facilitate operations such as key generation and encryption, making it more versatile in practical cryptosystems while retaining quantum-resistant security.

In quantum cryptanalysis, key notations include the expectation value $\langle \psi | \hat{O} | \psi \rangle$, representing the average measurement outcome for a quantum state $|\psi\rangle$ with respect to an observable $\hat{O}$, typically a Hermitian operator. This concept is central in VQAs, where the objective is to optimize the expectation of a Hamiltonian that encodes the cryptographic problem [22, 81, 44, 56].

This section establishes the core mathematical and quantum mechanical principles relevant to cryptanalysis in the NISQ era. The discussed concepts set the stage for applying quantum algorithms like VQE and QAOA to break LWE-based cryptosystems [72, 54, 59, 14].

2.2 Cryptography

Cryptography underpins modern information security by ensuring the confidentiality, integrity, and authenticity of digital communications. At its core, cryptography depends on problems that are computationally hard to solve, providing the basis for secure data protection. For example, RSA, introduced in 1978, relies on the difficulty of factoring large composite integers [74]. However, advancements in quantum computing challenge this model.

Quantum computing poses a serious threat to existing cryptosystems. Shor's algorithm, developed in 1994, can efficiently solve the integer factorization and discrete logarithm problems that RSA and other asymmetric cryptosystems depend on [81]. With sufficiently advanced quantum systems, RSA encryption could be broken, rendering once-secure communications vulnerable [3]. This expected vulnerability has accelerated the development of quantum-resistant alternatives.

Cryptographic systems are generally divided into symmetric and asymmetric categories. Symmetric cryptography, like AES, uses a single key for both encryption and decryption and remains relatively resilient to quantum attacks, as Grover's algorithm only offers a quadratic speedup [40, 15]. In contrast, asymmetric systems like RSA and Diffie-Hellman are significantly threatened by quantum algorithms like Shor's [30, 15]. The imminent quantum threat has led to "store now, decrypt later" strategies, prompting the development of post-quantum cryptography and initiatives like NIST's PQC Standardization project [76].

Among the leading PQC candidates are lattice-based cryptographic systems, including CRYSTALS-Kyber and CRYSTALS-Dilithium, which are built on the hardness of lattice problems, notably LWE [3, 52]. The LWE problem, introduced by Regev in 2005, remains computationally challenging for both classical and quantum systems [72]. The relevance of these problems lies in their connection to the Shortest Vector Problem (SVP). SVP's worst-case hardness implies the hardness of average-case LWE instances, making it a critical link in demonstrating the security of lattice-based cryptography [2].

Lattice-based cryptography's security derives from problems like SVP and the Closest Vector Problem (CVP), which are computationally intractable in high-dimensional spaces [67, 26]. These problems exhibit exponential scaling with dimensionality, ensuring robustness against classical and quantum attacks. Moreover, the worst-case to average-case reductions in lattice problems strengthen their reliability for cryptographic applications [59].

To improve efficiency while preserving security, structured LWE variants like Ring-LWE and Module-LWE have been developed [54]. These variants are foundational in practical deployments, as seen in systems like NTRU, which leverages ring-lattice structures for compact key sizes and fast operations [45].

2.3 LWE

The Learning with Errors problem, introduced by Regev in 2005, is a fundamental concept in lattice-based cryptography, playing a pivotal role in post-quantum cryptographic schemes [72]. The problem involves recovering a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$ from noisy linear equations $\mathbf{a}_i \cdot \mathbf{s} + e_i = b_i \mod q$, where $\mathbf{a}_i$ is public and e_i is a small error sampled from a discrete Gaussian distribution [59, 72]. The presence of this error term ensures that even with access to quantum resources, the problem remains computationally challenging. Regev's key contribution was demonstrating that LWE is at least as hard as solving certain worst-case lattice problems, specifically the SVP and CVP, under specific parameterizations [72]. This reduction was the first of its kind, establishing a direct connection between worst-case lattice problems and the average-case complexity of LWE. LWE is believed to be in QMA but not BQP, indicating its resistance even to quantum algorithms [59].

2.3.1 Regev's contribution to lattice-based cryptography

By establishing that breaking LWE is at least as difficult as solving these lattice problems, Regev's reduction became the cornerstone of post-quantum cryptography. Additionally, Regev introduced an encryption scheme based on LWE, encoding individual bits as noisy linear combinations of secret key components. This scheme laid the groundwork for more advanced constructions, including fully homomorphic encryption and identity-based encryption [17, 54]. The connection between LWE and lattice problems ensures that efficient algorithms for solving LWE would imply breakthroughs in solving SVP and CVP, which remain computationally infeasible even with quantum resources.

2.3.2 Regev's public encryption scheme

Regev's LWE-based encryption scheme encodes each bit into a noisy linear equation, where the noise ensures resistance to decryption attempts [72]. This simple yet powerful

design can be extended to multi-bit encryption using methods like dual-Regev encryption [54]. Regev's system has proven versatile, enabling applications such as fully homomorphic encryption while maintaining robustness against classical and quantum adversaries. Further enhancements like Ring-LWE and Module-LWE have improved efficiency without compromising security [54]. Regev's recommended parameters include choosing q as a prime between n^2 and $2n^2$, setting $m = 1.1 \cdot n \log q$, and defining the noise rate $\alpha = \frac{1}{\sqrt{n}\log^2 n}$, balancing security with practical performance [72].

2.3.3 Security parameters and classical approaches

The security of LWE-based systems relies on the careful selection of parameters: the modulus q, dimension n, noise rate α, and error distribution [59, 67]. The error distribution, often Gaussian or binary, and the noise rate α determine the problem's hardness. Selecting these parameters correctly ensures computational intractability while avoiding inefficiencies. Classical solutions like lattice reduction (BKZ) and combinatorial approaches (BKW) still require exponential time with appropriately chosen parameters, reinforcing LWE's resistance to known attacks [58, 8]. Regev's recommended parameter choices for LWE offer a balance between security and correctness, ensuring that noise levels are manageable while still thwarting adversarial attacks [72].

Traditional methods such as Gaussian elimination are ineffective due to the noise amplification that masks the secret vector, making them impractical for LWE decryption [59, 67]. The reduction of LWE to hard lattice problems combined with the absence of efficient quantum algorithms for LWE strengthens the case for its post-quantum security [72, 19]. Despite decades of cryptanalytic efforts, no polynomial-time solutions have emerged for LWE, underscoring its central role in post-quantum cryptography research.

2.4 VQE

The Variational Quantum Eigensolver is a hybrid quantum-classical algorithm designed to approximate the ground state energy of a given Hamiltonian $\hat{H}$. Mathematically, the Hamiltonian is expressed as:

$$\hat{H} = \sum_i c_i P_i,$$

where P_i are tensor products of Pauli operators and c_i are real coefficients. VQE optimizes a quantum circuit parameterized by θ to minimize the expectation value

$\langle\psi(\theta)|\hat{H}|\psi(\theta)\rangle$. This iterative process is carried out using a classical optimizer that adjusts θ based on measurement results, allowing current NISQ devices to address complex cryptographic problems like those found in LWE-based systems [70, 22].

2.4.1 VQE Notation and optimization

In the context of quantum algorithms, the variational principle states that for any parameterized quantum state $|\psi(\theta)\rangle$, the expectation value $E(\theta) = \langle\psi(\theta)|\hat{H}|\psi(\theta)\rangle$ is an upper bound on the ground state energy E_0. The VQE algorithm seeks to iteratively adjust θ to minimize $E(\theta)$, thereby approximating E_0 [56]. The flexibility of VQE makes it suitable for NISQ devices by distributing computational tasks between quantum circuits and classical optimizers, enabling practical applications even with current noisy hardware [27].

While originally developed for quantum chemistry, VQE has been extended to combinatorial optimization, where problem instances are mapped to Hamiltonians. A notable example is the Max-Cut problem, where the objective is to partition a graph such that the number of edges between partitions is maximized. This problem is encoded into an Ising Hamiltonian that VQE can optimize [10]. In this scenario, a parameterized quantum circuit generates candidate bitstrings representing potential solutions, which are evaluated by a cost function guiding the classical optimizer in refining θ.

2.4.2 Ansatz selection for cryptographic problems

Ansatz selection is crucial in determining VQE's performance. In the context of cryptanalysis and combinatorial optimization, hardware-efficient ansatzes are typically employed due to their reduced circuit depth, aligning well with the near term limitations. However, these ansatzes may limit expressibility, impacting the solution quality for complex problems [49]. For cryptographic tasks like solving LWE instances, problem-specific ansatzes that incorporate the structural properties of lattice problems are preferred. These ansatzes balance expressibility and feasibility, leveraging known symmetries or algebraic properties inherent to LWE. Advanced methods like qubit-ADAPT-VQE further improve performance by dynamically constructing the ansatz during the optimization process, tailoring it to the specific requirements of the problem being solved [46]. In cryptanalysis, such adaptive strategies are particularly useful when dealing with high-dimensional lattice problems, allowing for more efficient exploration of solution spaces while managing the computational overhead.

2.5 QAOA

The Quantum Approximate Optimization Algorithm (QAOA) is a variational quantum algorithm designed to solve combinatorial optimization problems, particularly those representable as Quadratic Unconstrained Binary Optimization (QUBO) instances. QAOA operates by alternating between a problem Hamiltonian $\hat{H}_P$ that encodes the objective function and a mixer Hamiltonian $\hat{H}_M$ that facilitates exploration of the solution space. The QAOA ansatz is constructed as:

$$|\psi(\gamma,\beta)\rangle = e^{-i\beta\hat{H}_M} e^{-i\gamma\hat{H}_P} |\psi_0\rangle,$$

where γ and β are variational parameters optimized through classical routines to minimize the expectation value of the problem Hamiltonian. This alternating sequence can be extended for deeper circuits, commonly denoted by the parameter p, leading to:

$$|\psi(\gamma,\beta)\rangle = \prod_{j=1}^{p} e^{-i\beta_j\hat{H}_M} e^{-i\gamma_j\hat{H}_P} |+\rangle^n.$$

QAOA's utility lies in its ability to discretize the evolution of a quantum state towards a solution, providing flexibility in parameter tuning that is particularly beneficial for NISQ devices [34, 91].

2.5.1 QUBO representation in cryptanalysis

QUBO problems are essential in QAOA's application to cryptanalysis, as many combinatorial problems, including those underlying cryptographic challenges like LWE, can be expressed in this format. A general QUBO problem is defined as:

$$f(x) = \sum_i a_i x_i + \sum_{i<j} b_{ij} x_i x_j,$$

where $x_i \in \{0,1\}$ are binary variables. Mapping cryptographic problems to this form allows for their representation as diagonal Hamiltonians in a quantum framework, enabling efficient exploration and solution finding on quantum hardware [41].

2.5.2 Comparison to quantum annealing

QAOA's formulation is closely related to quantum annealing, with both methods aiming to approximate adiabatic evolution. However, QAOA differs by discretizing the process into alternating unitary operations parameterized by γ and β. This discretization provides

greater control over the evolution path, allowing for targeted optimization based on specific problem instances [85]. While quantum annealing relies on a continuous interpolation between initial and final Hamiltonians, QAOA's stepwise approach offers enhanced tunability, which can be advantageous in exploring rugged optimization landscapes characteristic of cryptographic problems like those encountered in LWE [13].

2.5.3 Strengths and weaknesses of QAOA relative to VQE

QAOA and VQE are both variational quantum algorithms, yet they serve distinct purposes. While VQE is primarily used for finding ground states in problems like quantum chemistry, QAOA is specifically tailored for combinatorial optimization. QAOA's structured approach with fewer variational parameters results in simpler optimization landscapes, making it better suited for problems like Max-Cut or lattice-based cryptanalysis tasks. However, QAOA's performance is highly dependent on circuit depth and parameter optimization, both of which are constrained by noise and decoherence in current quantum hardware [34, 91]. VQE, although more flexible in handling a broader range of Hamiltonians, suffers from more complex optimization challenges, making it less efficient for certain combinatorial tasks compared to QAOA [55].

2.5.4 Challenges and practical considerations in QAOA implementation

The efficacy of QAOA is contingent on several factors, including the initialization of parameters, the classical optimization routine employed, and the circuit depth. As the depth p increases, the quantum state better approximates the solution, but this also introduces noise-related complications on NISQ devices. Issues like barren plateaus, where gradients vanish, present significant hurdles to efficient optimization, requiring advanced strategies like layerwise training or adaptive methods for improved performance [55, 91]. Furthermore, practical implementations must account for hardware constraints, limiting the scalability of QAOA to larger problem instances [21].

2.5.5 Conditional Value at Risk (CVaR)

CVaR is a risk measure used in quantum optimizations to improve robustness, especially in noisy environments like those found in NISQ devices. In variational quantum

algorithms such as QAOA and VQE, CVaR targets worst-case scenarios by focusing on the tail end of the distribution, minimizing expected losses beyond a defined threshold [56]. This is critical in optimizing problems where conventional methods fail due to the randomness in quantum measurements, leading to potentially suboptimal results.

CVaR integration in VQAs enhances resilience and consistency. Conventional quantum optimization strategies often average all outcomes, leading to deviations caused by noise [27, 69]. By focusing on the most adverse outcomes, CVaR directs the optimization process toward solutions less sensitive to fluctuations, ensuring better performance under quantum noise [77]. This focus is particularly relevant for NISQ devices, where high error rates demand robust optimization strategies.

2.5.6 Ascending CVaR

Ascending CVaR refines optimization by dynamically narrowing the focus from broader to more precise percentiles of the cost distribution. Early stages consider larger percentiles (e.g., top 50%) for broad exploration, which is tightened as optimization progresses, balancing exploration with exploitation [56, 42]. This approach is effective in applications requiring precision, where gradual adjustments lead to better convergence and solution quality [75].

As CVaR percentage changes, global minima remain constant, while local minima shift, a useful property to avoid entrapment in suboptimal regions. This characteristic facilitates navigation across jagged landscapes typical of cryptographic problems like LWE [25], enabling more efficient optimization.

2.5.7 Advantages and limitations of CVaR

CVaR offers notable benefits beyond noise resilience, including mitigating barren plateaus regions with nearly zero gradients by focusing on the most challenging parts of the parameter space. This emphasis improves convergence and helps the algorithm avoid local optima [12, 80]. Empirical evidence shows that CVaR improves consistency and reliability in non-convex optimization landscapes, which is crucial in cryptographic scenarios like LWE-based cryptanalysis [39].

However, the approach introduces computational overhead due to the need for extensive sampling to evaluate tail risks. This burden is especially significant in ascending CVaR strategies, where each refinement requires multiple evaluations to accurately assess the risk distribution [51]. Limited coherence times and qubit resources on current

quantum devices exacerbate this issue, potentially hindering practical implementation.

While CVaR aims to minimize adverse outcomes, this focus can lead to overly conservative solutions, potentially overlooking regions that could yield better results if adequately explored. This conservative bias is a significant concern in high-dimensional problems, where prematurely narrowing the focus might impede the discovery of optimal solutions [42]. Careful tuning of parameters, particularly the quantile level, is essential, especially in NISQ environments where balancing computational cost and reliability is critical [77].

2.5.8 Pros and cons applying ascending CVaR

While ascending CVaR improves robustness by progressively refining the optimization target, it introduces complexities in algorithm design and practical implementation. The iterative tightening of the quantile requires precise control mechanisms, adding complexity in both optimization and hardware management [51]. The additional iterations necessary to achieve the desired precision extend the runtime, leading to trade-offs between accuracy and computational efficiency. These trade-offs are critical in cryptanalysis, where resource constraints are significant [79].

CVaR-based optimizations show promise in cryptanalysis, particularly for solving the LWE problem. LWE's high-dimensional structure and complex error distribution create an intricate landscape that benefits from the precision and robustness CVaR offers. Incorporating CVaR into QAOA and VQE could enhance exploration, potentially leading to faster convergence and higher-quality solutions [25].

2.6 Previous work and research gaps

The study by Lv et al. 2022 delves into employing VQAs to address the LWE problem using NISQ-era quantum devices. Their work introduces two approaches: the application of QAOA for improving the classical Nearest Plane algorithm and the use of a VQE to address the unique Shortest Vector Problem (uSVP), which is closely linked to LWE. Through small-scale experiments, Lv et al. demonstrate that these hybrid quantum-classical methods enhance the performance of classical cryptanalysis approaches under constrained quantum resources [53].

While Lv et al.'s contribution is significant in extending VQAs to cryptographic applications, it is limited in several respects, particularly regarding its underlying approach

to encoding the LWE problem into Hamiltonians. Their methods primarily focus on optimizing classical algorithms by introducing quantum enhancements rather than directly tackling the core LWE problem through novel quantum formulations. Specifically, the reduction of LWE to uSVP relies heavily on existing lattice reduction techniques (e.g., LLL, BKZ) and fails to explore innovative Hamiltonian encodings that directly reflect the structure of LWE. Additionally, their reliance on known algebraic reductions such as Kannan's embedding method constrains the scope of the quantum advantage achievable, as these techniques are already deeply studied in classical cryptanalysis [53].

In contrast, our research explicitly addresses these gaps by focusing on directly encoding LWE-related problems into Hamiltonians tailored for NISQ devices. Rather than adapting classical reductions, this dissertation proposes novel Hamiltonians derived from the specific algebraic structure of LWE, enabling more efficient mappings to QUBO formulations used in QAOA and VQE. This approach not only offers a fresh perspective on the cryptanalysis of lattice-based schemes but also extends the applicability of quantum algorithms beyond mere optimizations of existing techniques. Unlike Lv et al., our methodology emphasizes exploring deeper structural properties within LWE, specifically leveraging quantum principles like entanglement and interference, which are underutilized in classical reductions [53].

The research gap lies primarily in the need for a quantum-centric view of LWE that leverages Hamiltonian encodings beyond the conventional lattice reduction framework. Lv et al.'s experiments, while demonstrating incremental improvements, are limited by classical preprocessing, which diminishes the potential quantum advantage. This work seeks to fill this gap by integrating quantum-native formulations that bypass extensive classical preprocessing, focusing instead on quantum representations of the LWE challenge directly. This leads to a more direct evaluation of LWE's hardness under quantum algorithms, with implications for cryptographic security and the practical applicability of post-quantum cryptosystems like Kyber [53].

In summary, while existing literature, particularly Lv et al.'s work, advances the use of VQAs for lattice-based cryptanalysis, it remains rooted in classical methodologies. My research distinguishes itself by pioneering Hamiltonian formulations tailored specifically for LWE, setting the stage for a more profound exploration of quantum cryptanalysis, particularly in scenarios constrained by NISQ hardware.

Chapter 3

Research methodology

3.1 Conceptual framework

This research examines the application of NISQ-era quantum computing, for which purpose we employ Qiskit for our implementation. Central elements include the simulation of noise models via Qiskit `Aer`, implementation of VQE and QAOA algorithms using the Qiskit Algorithms module, and leveraging key primitives like `Sampler` and `Estimator` for performance evaluation.

3.1.1 Noise models and simulation with Qiskit Aer

Qiskit `Aer` is pivotal in this research, providing a high-performance simulation environment for replicating the noisy conditions typical in real quantum devices. `Aer` includes simulators like the `qasm_simulator`, which supports both noiseless and noisy simulations. The ability to introduce customized noise models, such as those for decoherence, gate errors, and thermal relaxation, allows for accurate modeling of specific quantum hardware profiles and allows for future extensibility. An important feature used in this research is `NoiseModel.from_backend()`, which directly imports noise characteristics from IBMQ devices, enabling realistic assessment of algorithm performance under physical noise conditions [48, 24, 63].

For tailored simulations, `Aer` provides multiple methods like `StateVector`, `DensityMatrix`, and `Stabilizer` approaches, balancing between computational efficiency and accuracy depending on circuit complexity. Although `Aer` supports GPU acceleration for large-scale testing, this research focuses primarily on `StateVector` simulations due to their superior performance for the specific algorithms employed.

Importantly, these simulations leverage the backend noise models, without GPU acceleration, to closely mimic the environments of current NISQ devices [1].

3.1.2 Qiskit algorithms module

The `qiskit_algorithms` module underpins the implementation of VQE and QAOA. A key factor in the performance of these algorithms is the choice of ansatz. The `TwoLocal` ansatz, employed in both VQE and QAOA, consists of alternating single-qubit rotations and entangling gates. Its modular design enables the customization of gate configurations, facilitating exploration of the solution space with computational efficiency. This research focuses on this ansatz due to its adaptability for lattice-based cryptographic problems, specifically in the context of LWE. Other ansatz options, like `EfficientSU2`, are considered less optimal due to their balance between expressiveness and resource requirements, making `TwoLocal` the primary choice for this study [47].

3.1.3 Noise profiles and circuit optimization

In simulating quantum environments, detailed noise models are crucial for evaluating algorithm robustness. `Aer`'s noise profiles, such as depolarizing noise for CNOT gates and readout error models, are integrated into this framework, allowing for comprehensive testing of VQE and QAOA under noisy conditions. While this research mentions these capabilities, the primary focus remains on utilizing backend-specific noise models, rather than custom-designed profiles, for more accurate replication of real-device conditions [24, 63].

3.1.4 Qiskit Primitives: Sampler and Estimator

The `Sampler` and `Estimator` primitives are critical for analyzing VQAs. The `Estimator` computes expectation values, vital for evaluating the performance of VQE and QAOA, while the `Sampler` provides probability distributions from measurement outcomes, essential for statistical analysis under noise. These primitives support integration with noise models, ensuring that the simulated conditions align closely with those expected in actual quantum hardware. By leveraging these tools, this research offers more realistic insights into algorithm performance [88, 33].

3.1.5 Algorithm implementations and customizations

This research builds upon Qiskit's implementation of VQE and QAOA, with specific adaptations for LWE cryptanalysis. QAOA is implemented using Qiskit's native framework, while VQE is extended with custom components due to the need for simulating a general cost function involving bitstring outcomes. This customization requires gradient descent methods, optimizer classes, and expectation value calculations via parameter shift rules. The `ParamShiftSamplerGradient` is employed for derivative computations, ensuring accuracy in parameter optimization [65].

In terms of optimization, the study experiments with `COBYLA` and `SPSA`, ultimately favoring `COBYLA` for its robustness and efficiency in handling the noise profiles encountered. Additionally, while Qiskit offers various gradient calculation methods, this research primarily uses default configurations, as these align with the requirements of the studied algorithms and offer a stable baseline for further exploration [38].

3.1.6 Practices and performance tracking

Ensuring reproducibility and consistent performance measurement are key aspects of this study. The research incorporates practices like setting a global seed and averaging results across multiple runs, reporting both mean and standard deviation where applicable. Performance is tracked using profiling tools to identify code segments that could benefit from optimization. These methodologies are essential for validating the results and ensuring that conclusions are based on robust, repeatable findings.

3.2 Theoretical approach and mathematical foundations

This section focuses on the mathematical proofs that underpin the cryptanalysis techniques explored in this research. The proofs provide the basis and the intuition for the implementation of the QAOA and VQE solutions.

3.2.1 Hamiltonian definition

Following the notation outline in Section 2.1, consider the following cost function, which plays a key role in modeling the problem:

$$C(\vec{x}) := \sum_i \left[\left(\sum_j A_{ij} x_j \right) - b_i \right] \tag{3.1}$$

where all operations are taken modulo q. It can easily be seen that when one plugs in the secret vector s, $C(\vec{x})$ simplifies to $\sum_i (e_i \mod q)$ (using Definition 2.1 of mod), now if $e_i < 0$ is a small negative value, then $e_i \mod q = q - e_i$ which is a relatively big quantity, considering the default security parameters (see Appendix B).

To transform this intuition into the VQE Hamiltonian, one needs to solve this modulo problem and adequately reward small error terms. This gives rise to the two following approaches:

3.2.1.0.1 Approach 1:

$$H_{\text{VQE}} := \sum_i \left[\left(\left(\sum_j A_{ij} x_j \right) - b_i \right)^2 \mod q \right] \tag{3.2}$$

3.2.1.0.2 Approach 2:

$$H_{\text{cmod}} := \sum_i \left[\left(\left(\sum_j A_{ij} x_j \right) - b_i \text{ cmod } q \right)^2 \right] \tag{3.3}$$

where crucially the *cmod* is the centred mod defined as:

$$a \text{ cmod } q = \left[\left(a + \frac{q}{2} \right) \mod q \right] - \frac{q}{2} \tag{3.4}$$

Here we discuss the intuition behind each approach. Approach 1:, solves the issue directly by using the square to negate any negative error terms, before taking the modulo operation. Approach 2: tackles the problem with negative error terms simply by permitting them in a way that preserves their closeness to the factor. Then a square of that expression is taken, so that this linear term doesn't negate the cost of any other terms. The square operation also provides mean to punish deviations polynomially with respect to their distance. In simpler terms the further you are from a factor of q, you get proportionately punished.

Moreover, both these Hamiltonians are quadratic in nature since each term in the summation is a squared difference between a linear combination of the input vector components x_j and the scalars b_i. The quadratic nature is crucial because it allows us to apply techniques such as QAOA to this problem. Quadratic Hamiltonians can be efficiently mapped to quantum circuits, where the optimization can be performed using quantum resources.

Nevertheless there is a very subtle problem with Approach 1:. When we take the square before the modulus we run the risk drastically increasing the magnitude of the

error terms e_i to a point where $e << q$ no longer holds. To demonstrate this point we examine following Figure 3.1.

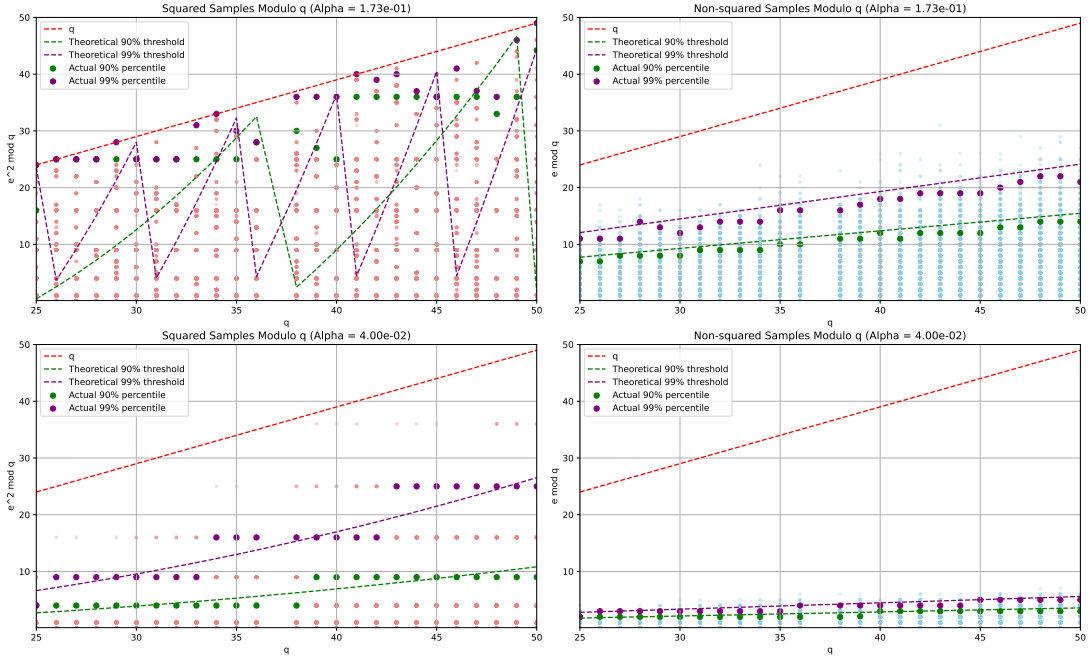

Figure 3.1: Distribution of e_i with respect to $q \in [n^2, 2n^2)$ for $n = 5$. Squared vs non-squared errors on the left vs right, respectively. $\alpha = 1/(\sqrt{n}\log^2(n))$ and $\alpha = 1/n^2$ up vs down, respectively.

Figure 3.1 demonstrates several interesting aspects. First we observe (top left) that following Regev security parameters for α, namely $\alpha = 1/(\sqrt{n}\log^2(n))$ and going with Approach 1: for the Hamiltonian is intractable as the plot suggest that squaring the error (assuming we want to have $x = s$) is going over the range of the modulus, essentially contributing near uniform spread, where we couldn't hope to identify the solution. This is exemplified by the theoretical percentiles wrapping around as a result of the application of the modulus. Secondly, we observe that Approach 1: is only viable when the standard deviation is reduced with α (bottom left). It is important to state that this is a reasonable suggestion and follows the security recommendations as we us $\alpha = \frac{1}{\text{Poly}(n)}$. And it works well as the 99^{th} percentile mass of the distribution is in the first $\frac{q}{2}$ range of the modulus, which is a promising sign for the ability of VQA to find this as an optimal solution. Finally, we note that approach 2 where we take the centre mod before the square has potential to work well with both definitions of α as again there is a big separation and the probability mass is lumped near small values.

Looking at figure 3.2 one might observe similar tendencies. We once again can conclude that using Regev's definition of α and squaring the magnitude of the error before applying the modulus operations ends up covering the modulus range, which

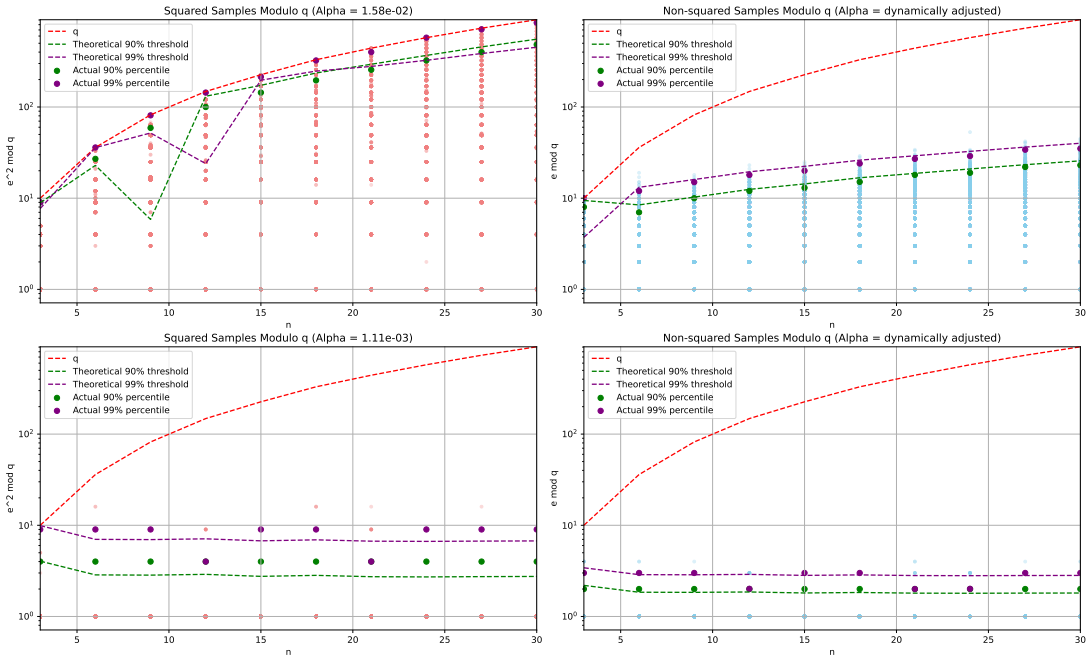

Figure 3.2: Distribution of e_i with respect to $n \in [3,30]$ in increments of 3. Squared vs non-squared errors on the left vs right, respectively. $\alpha = 1/(\sqrt{n}\log^2(n))$ and $\alpha = 1/n^2$ up vs down, respectively.

ends up masking the solution. Apart from that we can also observe something that we will see in the results and that is that increasing n tends to improve the correctness properties of Regev's encryption and it also drastically improves the chances of finding the solution with VQAs. It can be seen that as n increases, the distribution of e_i stays near constant, continually expanding the gap between the error terms and the modulus, perpetually bettering the correctness properties of Regev's 1 bit encryption and success rate of our VQA solutions.

The formulations above effectively encodes the goal of finding a vector $\vec{x}$ such that the distance (or norm) between the computed vector and the target is minimized. The cost function $C(\vec{x})$ captures this distance as the sum of squared errors. Minimizing this function helps identify a vector that is close to the secret vector $\vec{s}$, particularly in cases where the errors (represented by deviations from b_i) are small. We explore Approach 1: further in our VQE testing, where it is more viable as we mainly tackle ternary error distributions $-1,0,1$ for scalability reasons. There the squaring of the magnitude is of no consequence as the errors remain the same. We explore Approach 2: in our QAOA testing.

3.3 QAOA Hamiltonian and Modulo Encoding

To apply QAOA we extend approach 2 by translating the problem into a QUBO formulation that can be executed on a quantum devices. The QAOA Hamiltonian is structured as:

$$H_{\text{QAOA}} := \sum_i r_i^2 + Pf(\vec{x}), \tag{3.5}$$

$$\text{where} \quad f(\vec{x}) := \sum_i \left[\left(\left(\sum_j A_{ij} x_j \right) - b_i - k_i q - r_i \right)^2 \right] \tag{3.6}$$

This Hamiltonian introduces auxiliary variables and a penalty function in order to encapsulate the behaviour of the modulus operation. This is achieved by noticing that $\sum_j A_{i,j} x_j - b_i$ can be expressed as $k_i q + r_i$. Where the quotient k_i and the remainder $r_i \in [0, q)$ are auxiliary integers when dividing by q and P is a large penalty constant for the penalty function $f(\vec{x})$. This formulation ensures that the modulus operation is respected during the optimization.

Now, we claim that this Hamiltonian encodes the modulus operation correctly and further it requires an asymptotic $O(n \log n)$ number of qubits to encode an LWE instance of n dimensions.

3.3.0.0.1 Proof of encoding correctness The encoding is proven to be correct by showing that the optimal solution satisfies the modulus constraint and further that any other $\vec{x}$ is being penalized. Start by considering:

$$\sum_j A_{ij} x_j - b_i = k_i q + r_i \tag{3.7}$$

$$\Rightarrow \quad \sum_j A_{ij} x_j - b_i \mod q = k_i q + r_i \mod q \tag{3.8}$$

$$\Rightarrow \quad \sum_j A_{ij} x_j - b_i \mod q = r_i \tag{3.9}$$

$$\Rightarrow \quad (\sum_j A_{ij} x_j - b_i \mod q)^2 = r_i^2 \tag{3.10}$$

Again here assuming Regev's Definition 2.1 of modulo. The penalty function $f(\vec{x})$ (3.6) enforces the relationship by penalizing any deviation from the exact modulus result. If (3.7) holds for some k_i and r_i, then $f(\vec{x}) = 0$, resulting in no additional penalty

resulting in $H_{\text{QAOA}} = \sum_i r_i^2$, which is equal to the original Hamiltonian from Approach 2: (3.3) as demonstrated by Equation (3.10). On the other hand, if $\vec{x}$ is not the solution it either has a higher probability of contributing with a bigger remainder or it is punished by the penalty function $f(\vec{x})$ and a large positive constant P is applied. P is subject to heuristics as a sufficiently large constant ensures that the penalty term has a significant impact on the Hamiltonian dynamics, however, not too big as to obstruct the smooth learning of the objective function.

3.3.0.0.2 Proof of qubit complexity The complexity analysis chosen in this research is a straightforward asymptotic qubit count. Both the QAOA (3.5) and VQE (3.2) Hamiltonians share the initial vector encoding, which cannot be significantly reduced. To encode the vector $\vec{x}$ one needs $n \log q$ bits for the dimensions of the LWE instance times the bits required to express an integer $\mod q$. It is worth mentioning that setting q to be the largest prime just before a power of 2 in the range $[n^2, 2n^2)$ aims to optimize the number of bits required to express each integer, while providing the largest possible modulus, which we later show that improves correctness and probability of finding the solution. Worst case scenario this turns out to be $n \log_2(2n^2 - 1) \approx 2n(\log_2(n) + 1)$ and $2n \log_2(n)$ in the best case, which turns out to be $O(n \log n)$. However, cryptography is concerned with constants and for the sake of demonstrating improvements the complexities are represented with their coefficients together with their asymptotic complexities. Therefore, $O(n \log n)$ and more precisely $2n(\log_2(n) + 1)$ is sufficient for VQE.

In addition to encoding the potential solution, QAOA needs to encode the auxiliary variables. More specifically, r_i can be represented with $\log_2(2n^2 - 1) \approx 2(\log_2(n) + 1)$ and $2 \log_2(n)$ bits in the worst and best scenarios, respectively, which is $O(\log n)$. Alternatively, k_i must be big enough to cover the possible range of $\frac{\sum_j A_{ij} x_j - b_i}{q}$. It turns out that one requires $\log_2(2n^3 - 3n + 2) \approx 3(\log_2(n) + 1)$ and $\log_2(n^3 - 2n + 2) \approx 3 \log_2(n)$ in the worst and best cases, respectively. Asymptotically this is again $O(\log n)$. It should also be noted that both k_i and r_i are dependent on i and are independent of each other, hence the number of additional qubits is $O(n \log n)$ and more precisely $n(2 \log_2(n) + 2 + 3 \log_2(n) + 3) = 5n(\log_2(n) + 1)$. To arrive at the total count needed to run QAOA, the qubits encoding the potential solution are added to amount to $7n(\log_2(n) + 1)$ in the worst case or $O(n \log n)$ asymptotically. Finally, remark that many cryptographic systems target a security of 128 bits (including CRYSTALS-Kyber although using Module-LWE), meaning that an instance of that magnitude could be attempted with 7168 perfect logical qubits.

3.4 Research design

3.4.1 Implementation contribution

In this thesis, I developed a comprehensive approach leveraging NISQ algorithms for cryptanalysis, specifically targeting Kyber CRYSTALS and LWE. My implementation of classical methods, including a brute-force eigensolver and exhaustive search (Appendices C.1.1 and C.1.2), established reliable baselines to validate quantum approaches. I constructed a modular QUBO Hamiltonian (Appendix C.1.3) for adaptable problem encoding, ensuring accurate modeling. Finally, by applying QAOA to solve the QUBO (Appendix C.1.4), I demonstrated the feasibility of breaking cryptographic schemes with current quantum devices, bridging classical and quantum cryptanalysis.

This thesis introduces a robust approach to optimizing quantum circuits for cryptanalysis using gradient-based methods. The gradient descent optimization (Appendix C.2.1) was implemented from scratch, allowing precise control over the parameter updates. Additionally, I developed a parameter shift rule to manually calculate gradients (Appendix C.2.2), enhancing flexibility in adjusting which derivatives to compute. The expectation value computation (Appendix C.2.3) was carefully designed, incorporating noise considerations and penalization strategies for invalid solutions, ensuring accurate and reliable results when applied to the LWE problem.

3.4.2 LWE

The LWE problem is a foundation for post-quantum cryptography, and understanding its instance generation process is essential for developing cryptanalytic attacks. This section explains LWE instance generation with a focus on parameter selection, error distributions, and their impact on decryption correctness and attack success rates. We detail how security parameters influence decryption errors, using plots to visualize the link between errors and our cryptanalytic methods.

3.4.3 Generating LWE instances

Building on top of Section 2.1 we are generating LWE instances using a systematic process, namely the coefficient matrix $A \in \mathbb{Z}_q^{m \times n}$ is populated with uniformly random integers modulo q and the secret vector $s \in \mathbb{Z}_q^n$ is also sampled u.a.r., though it has been demonstrated as not strictly necessary for hardness guarantees. Despite this, uniform

sampling remains widely used due to its simplicity and well-understood behavior [73]. The error vector $e \in \mathbb{Z}_q^m$ is generated using a discrete Gaussian distribution centered at 0 with a standard deviation of αq, where $\alpha \sim O\left(\frac{1}{\text{Poly}(n)}\right)$ (defaults to $\frac{1}{\sqrt{n}\log^2(n)}$ as per Regev [73]). Additionally, we run tests where the errors are sampled from a ternary distribution $\{-1,0,1\}$, which, while offering reduced security, allows for specific analysis relevant to VQAs (see Approach 1:). Furthermore, we reiterate the core parameters, the most important of which is the problem dimension n. It affects the scalability and security of the lattice and dictates the modulus $q \sim O(\text{Poly}(n))$ (prime number $\in [n^2, 2n^2)$ Regev), and the number of linear equations $m \sim O(\text{Poly}(n))$ (with default value $1.1 \cdot n \log q$). Each of these parameters plays a key role in defining the hardness of the LWE problem, which in turn affects the correctness properties of both encryption/decryption and VQA-based cryptanalysis.

3.4.4 Understanding correctness and decryption error

The correctness of decryption depends on ensuring that the sum of errors does not exceed $q/4$. Without errors, the value $b - \langle a, s \rangle$ would be exactly 0 or $\lfloor q/2 \rfloor$, corresponding to an encrypted bit of 0 or 1. However, errors introduce deviations that can cause incorrect decryption if they push $b - \langle a, s \rangle$ beyond the threshold $q/4$. The sum of m error terms, each sampled from a Gaussian distribution with standard deviation αq, results in a total standard deviation of $\sqrt{m}\alpha q$. Correctness is maintained as long as this value remains below $q/4$, leading to the requirement:

$$\sqrt{m}\alpha q < \frac{q}{4} \implies \sqrt{m}\alpha < \frac{1}{4}.$$

This expression guides the parameter selection, especially for small n where $q \approx n^2$. For example, if $n = 4$, q is the smallest prime greater than 16 (which is 17), and α is approximately $\frac{1}{\sqrt{4}\log^2(4)}$, the standard deviation becomes $\sqrt{4} \times \alpha \times 17$. This remains below $q/4 = 4.25$, ensuring a low probability of decryption error.

3.4.5 Plotting and visual analysis

To analyze correctness conditions, we plot the expectation function and its conservative bound, Figure 3.3. The function $\sqrt{m}\alpha q$ represents the total error contribution across all samples, while $q/\log(n)$ provides a conservative bound on acceptable error levels for successful decryption [73]. As expected, with small LWE instances decryption errors are significant and the intersection of these functions with $q/4$ indicates the

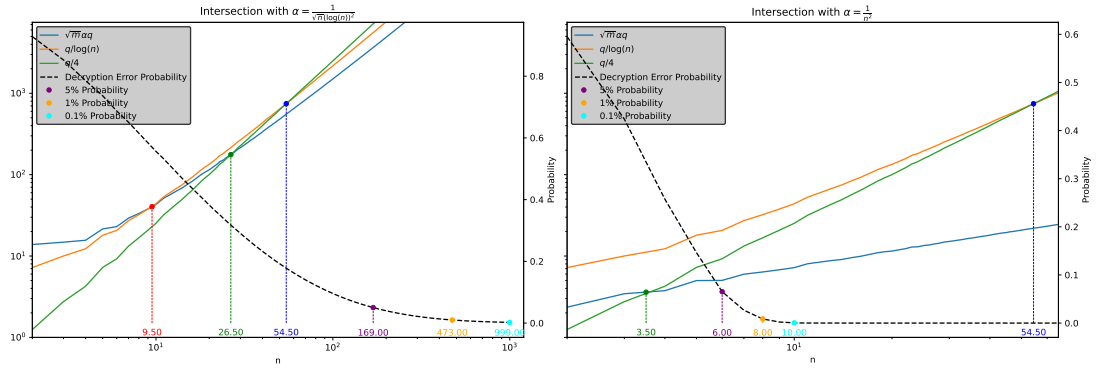

Figure 3.3: LWE correctness and decryption error probability with different α-s. The left y-axis (log scale) shows $\sqrt{m}\alpha q$, $q/\log(n)$, and $q/4$. The right y-axis shows the decryption error probability, using the CDF at 3 thresholds.

parameter cutoff points, where decryption error probability becomes acceptable. In the case of $\alpha = \frac{1}{\sqrt{n}\log^2(n)}$ this happens at 26.5 and for $\alpha = \frac{1}{n^2}$ it is 3.5, which indicates that correctness is easily achieved with $\alpha = \frac{1}{n^2}$ for a negligible penalty in security. This is further demonstrated by the CDF percentile points, which are at $169, 473, 999$ for the Regev definition of α and $6, 8, 10$ for the latter definition. Comparing the two plots essentially evaluates different choices for α, highlighting how it influences correctness. For small n, a smaller α results in smaller error rates, but as n increases, optimizing α becomes critical for maintaining correctness while balancing computational efficiency. Crucially, we observe that this decryption correctness is tightly related to the success rate of our VQE and QAOA approaches as seen in Chapter 4.

3.4.6 LWE implementation

The Python `LWE` class used in this study is scalable, modular, and type-safe, designed to accommodate different parameter settings. The class follows Regev's recommendations as default but allows for manual overrides of all key parameters, including n, q, α, m, and the error distribution.

Key features include: Customizable n, q, and α values; Support for different error distributions: Gaussian, ternary, uniform; Flexible encryption and decryption methods that can adjust the subset size of equations; Built-in functions for analyzing the Hamiltonian value, crucial for VQAs.

This modular design facilitates systematic experimentation with LWE parameters, making it well-suited for testing various cryptanalytic approaches.

Chapter 4

Results and analysis

This section provides an overview of the results, describing how the results are organized and presented.

4.0.1 QAOA results and analysis

The LWE instances explored in this study (A to F) were carefully selected for their experimental feasibility and qubit requirements, all under the 31-qubit limit supported by the Aer simulator. Each instance represents specific configurations of the LWE problem, varying in dimension, modulus q, and error distribution. A, B, C and D are 2D lattices with a modulus q close to 2^k, while E and F are 3D lattices with $q = 7$, which is **not** in the security bounds. A and B use $m = n$, while C and D use $m = 1.1 \cdot n \log q$. And finally A, C and E follow a ternary distribution for their errors and B, D and F use $\alpha = \frac{1}{\text{Poly(n)}}$ with a discrete Gaussian for the errors.

Figure 4.1 (Left) presents the results of QAOA's solution correctness, highlighting the impact of noise across different LWE instances. The performance of QAOA is notably inconsistent, especially for small instances such as A and B. In these cases, where $m = n$, overfitting occurs, leading to solutions that, although minimize the Hamiltonian, fail to correspond to the correct secret vector $\mathbf{s}$. This is primarily due to the limited linear equations in such configurations, allowing multiple solutions that erroneously achieve minimal eigenvalues by canceling out the Hamiltonian without addressing the error in the system. Consequently, the probability of finding the correct solution remains low, with higher variance, particularly under noisy conditions.

The noise significantly exacerbates this issue, reducing the solution accuracy across all instances. This trend is consistent with the limitations of NISQ devices, where

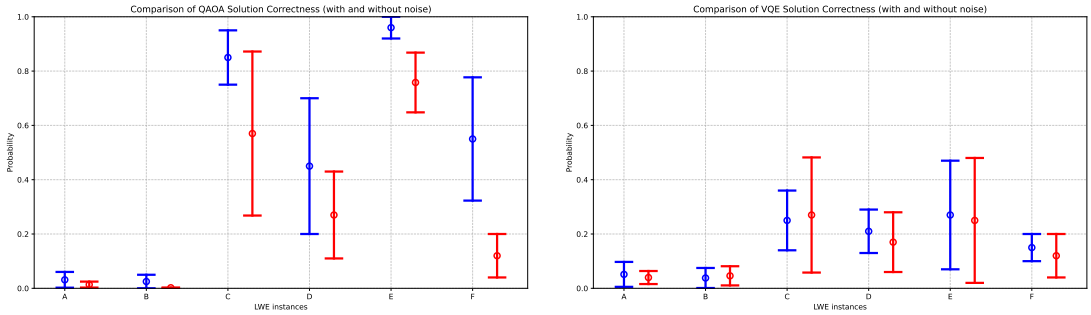

Figure 4.1: Left: Comparison of QAOA. Right: Comparison of VQE solution correctness (with and without noise). The red bars represent noisy conditions, while the blue bars correspond to the noise-free scenario. Results are presented with their mean and standard deviation across different sample sizes: 20 samples for instances A and B, 4 for C and D, and 10 for E and F.

noise disrupts the optimization process, hindering convergence to the correct solution. Specifically, instances C and D, which follow Regev's recommendations, demonstrate a higher likelihood of identifying the correct solution in the absence of noise. Instance C, using a ternary error distribution, performs well by reliably finding the global minimum, underscoring the role of error distribution in solution accuracy.

A clear distinction is observed between instances utilizing a ternary error distribution (A, C, E) and those with a discrete Gaussian distribution (B, D, F). The ternary error distribution, with its limited range, contributes to a more consistent performance in QAOA, as it reduces variance and allows for better optimization. In contrast, the complexity introduced by discrete Gaussian errors introduces more variability, complicating convergence and solution identification.

Interestingly, Instance E, with $n = 3$, exhibits the best performance, aligning with prior finding from our Section 3.2.1 and Section 3.4.2 that smaller LWE instances tend to have decryption errors, irrespective χ. These results indicate that while QAOA can effectively solve small LWE instances under various conditions, its scalability and robustness are limited by noise and problem complexity. Nonetheless, QAOA show great potential in that it manages to find the correct solution s with high probability.

4.0.2 VQE results and analysis

Figure 4.2 demonstrates VQE's performance under two different CVaR strategies. The left plot shows results using a constant 0.5% CVaR, characterized by an initial period of

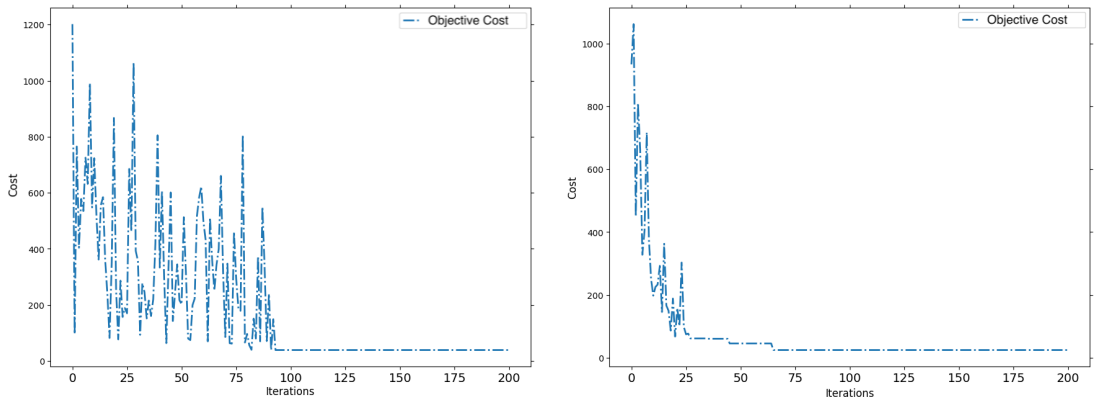

Figure 4.2: Left: Static CVaR on VQE Performance with a constant CVaR percentage of 0.5%. Right: Ascending CVaR strategy, starting from 90% and incrementally narrowing down to 99% by 1% every 10 iterations. For $n = 6$

instability due to sparse sampling, followed by rapid convergence once the algorithm reaches a critical threshold. The right plot illustrates the ascending CVaR strategy, which incrementally narrows the search space from a broad percentile to a more focused region. This method is particularly effective in balancing exploration and exploitation, allowing for robust performance even in the presence of quantum noise.

The conservative nature of CVaR is apparent in the results. While VQE does not always converge to the exact solution with complete certainty, it consistently achieves better-than-random outcomes, especially under noisy conditions. Instances C and D show around a 20% improvement in solution probability over baseline, demonstrating CVaR's effectiveness in cryptographic problem settings.

Figure 4.1 (Right) provides a comparison of VQE correctness across the same LWE instances tested with QAOA. The experimental setup involved multiple runs for each instance: 20 for A and B, 4 for C and D, and 10 for E and F. VQE's solution probabilities were measured by sampling the circuit 10,000 times per run, offering a more conservative yet stable performance profile compared to QAOA. Although VQE's absolute correctness is lower than QAOA's, it proves more resilient under noise, maintaining consistent, albeit cautious, solution quality.

4.0.3 Comparative analysis: QAOA vs. VQE

Comparing QAOA and VQE reveals distinct trade-offs in accuracy, scalability, and resilience to noise. QAOA tends to be more accurate for small instances but suffers under noisy conditions and larger problem sizes due to its combinatorial complexity.

In contrast, VQE, particularly when using CVaR strategies, offers a more scalable solution, with a conservative bias that prioritizes robustness over absolute correctness. The ascending CVaR method, starting at 90% and incrementally narrowing to 99%, likely explains the smooth convergence observed, allowing broader initial exploration before focusing on high-quality solutions.

From a resource efficiency perspective, VQE's qubit requirements scale more favorably as demonstrated by the 4.2 with $n = 6$. With a space complexity of $O(n \log n)$, VQE requires fewer qubits than QAOA, which demands up to $7n(\log_2(n) + 1)$ qubits in the worst case. This efficiency makes VQE more suitable for larger cryptographic instances, where scalability is a critical concern.

Overall, while QAOA shows promise for small to medium LWE instances, VQE, with its noise resilience and adaptive optimization strategies, presents a more reliable option for tackling larger and more complex cryptographic challenges in post-quantum scenarios.

Chapter 5

Discussion

The discussion consolidates the research findings, situating them within the context of post-quantum cryptographic research. This study explored the application of QAOA and VQE to cryptanalysis of Learning With Errors based cryptosystems like CRYSTALS-Kyber. The focus was on evaluating the practicality of using NISQ devices for cryptanalytic tasks, analyzing the impacts of algorithmic approaches and noise models on solution accuracy, scalability, and hardware feasibility.

5.1 Interpretation of results

The experimental results show that both QAOA and VQE can address small LWE instances effectively, but scaling these algorithms to cryptographically relevant dimensions remains challenging. QAOA showed promising performance for small LWE problems, particularly in noise-free conditions, but it is notably sensitive to noise, leading to high variance and reduced reliability. On the other hand, VQE, particularly when enhanced with CVaR optimization, demonstrated greater resilience to noise, maintaining more consistent performance across different LWE configurations.

One key insight is the role of error distribution in determining algorithm robustness. LWE instances with ternary error distributions exhibit more stable performance compared to Gaussian errors due to the reduced variability, allowing for more effective convergence in quantum optimization processes. Additionally, VQE's more efficient qubit usage and scalability make it better suited for larger cryptographic instances, even though it tends toward conservative solutions due to CVaR's cautious optimization focus.

5.2 Comparison with existing literature

This research builds upon and extends previous studies exploring the application of VQAs to cryptanalysis, particularly in LWE-based cryptosystems. Studies like Lv et al. have emphasized hybrid quantum-classical approaches that enhance classical cryptanalysis methods, primarily relying on lattice reduction techniques. In contrast, this study directly encodes LWE instances into Hamiltonians specifically tailored for quantum optimization, allowing for a more direct assessment of the cryptosystem's resilience. The findings reinforce that while quantum advantages are evident under specific conditions, fully exploiting quantum-native methods is crucial for broader cryptanalytic applications.

The comparison of QAOA and VQE reveals distinct differences in handling problem structure and noise. QAOA, with its discrete optimization approach, is more effective for combinatorial tasks but suffers under current NISQ hardware constraints, whereas VQE, leveraging continuous parameter spaces, provides a more adaptable framework for complex cryptographic problems. These results contribute to the growing body of research suggesting that hybrid quantum-classical algorithms, when properly optimized, hold significant promise for future cryptanalysis.

5.3 Theoretical implications

The encoding strategies developed introduce novel Hamiltonian formulations, advancing beyond classical reductions by leveraging quantum principles such as superposition and entanglement. Specifically, incorporating centered modulo operations and auxiliary variables in the QAOA Hamiltonian enhanced the accuracy in modulus-based cryptographic tasks, a crucial element for lattice-based cryptosystems.

Moreover, the scalability analysis shows that while current hardware limitations impose constraints, the $O(n \log n)$ qubit complexity provides a feasible pathway for addressing small to medium-sized cryptographic instances. This study highlights the need to refine error distributions and noise models in cryptanalysis, offering theoretical insights into optimizing quantum algorithms for structured cryptographic problems. However, fully breaking cryptosystems like CRYSTALS-Kyber remains beyond the capabilities of current NISQ devices, emphasizing that this research primarily provides groundwork rather than definitive cryptanalytic breakthroughs.

5.4 Practical implications

While breaking LWE-based cryptosystems like CRYSTALS-Kyber with current quantum technology remains infeasible, this research shows significant progress in evaluating cryptographic resilience. The application of CVaR strategies in VQE offers improved robustness against noise, a critical factor in practical cryptanalysis given the inherent limitations of NISQ devices. VQE's qubit-efficient design presents a scalable framework for larger cryptanalytic attacks as quantum hardware matures, particularly when combined with advanced noise-aware optimization strategies.

Additionally, the study shows that future cryptanalytic work should prioritize optimizing parameter settings and exploring hybrid methods that combine quantum subroutines with classical algorithms. The research reinforces the necessity of continuously assessing the security of proposed standards like CRYSTALS-Kyber as quantum algorithms and hardware evolve.

5.5 Limitations

This study is constrained by several factors, notably the use of Qiskit and classical simulations, which limit the scope of results as they do not fully replicate real quantum hardware behavior. The noise models employed, while detailed, cannot perfectly simulate decoherence and gate errors in actual devices. Additionally, the small-scale LWE instances considered are far from the dimensions required for real-world cryptographic applications like CRYSTALS-Kyber, which generally necessitate hundreds of dimensions. The simplified ternary error distribution used in some experiments, although analytically convenient, diverges from the Gaussian profiles typical of LWE cryptosystems.

Further, the research's reliance on specific algorithmic configurations—such as the choice of ansatz, optimizers, and noise models—means results could vary with different setups. VQE and QAOA performance is particularly sensitive to hyperparameter settings. While this study provides robust benchmarks, further exploration of alternative configurations is necessary for a comprehensive understanding of these algorithms' potential.

5.6 Future research

Future work should focus on scaling quantum cryptanalysis of LWE and Module-LWE to realistic cryptographic dimensions, exploring advanced noise mitigation techniques and error correction strategies. Investigating alternative ansatz designs and hybrid quantum-classical algorithms could enhance the feasibility of breaking cryptographic systems. Expanding the study to include fully fault-tolerant quantum devices, once available, will be essential in determining the full potential of quantum cryptanalysis against post-quantum cryptosystems.

Given the positive results from CVaR-based optimizations, future research could explore more sophisticated CVaR strategies (like Best-CVaR) or adaptive techniques that dynamically adjust optimization parameters. Additionally, analyzing the resilience of other PQC finalists, such as those built on structured lattices, will be critical for understanding the broader implications of quantum cryptanalysis as technology progresses.

In conclusion, while this study did not fully achieve the goal of weakening Kyber CRYSTALS nor Regev's primitive or fully analyzing the security of Module-LWE, it establishes a strong foundation for future research in quantum cryptanalysis. The findings emphasize the importance of continuous evaluation of post-quantum cryptographic systems as quantum hardware and algorithms mature, setting the stage for more comprehensive cryptanalytic approaches in the future.

Chapter 6

Conclusion

The conclusion of this dissertation synthesizes the findings from the research on practical quantum algorithms for cryptanalysis, focusing specifically on the application of NISQ methods to attack LWE-based cryptosystems like CRYSTALS-Kyber.

6.1 Summary of the study

This research investigated the use of Variational Quantum Algorithms (VQAs), particularly the Quantum Approximate Optimization Algorithm (QAOA) and the Variational Quantum Eigensolver (VQE), to break lattice-based cryptographic systems. The core of the analysis centered on encoding the LWE problem into Hamiltonians suitable for quantum optimization, evaluating how well these algorithms perform on NISQ-era quantum devices. The study also explored the implications of using different error distributions, such as ternary and Gaussian distributions, within the cryptanalysis framework, analyzing the scalability and resilience of these quantum algorithms under realistic noise conditions.

The work introduced two novel Hamiltonian encodings tailored for LWE-based cryptosystems, enabling the problem to be expressed in a form compatible with both QAOA and VQE. Extensive experiments were conducted using Qiskit to simulate these quantum approaches under varying configurations, focusing on small-scale LWE instances. The results were benchmarked against classical algorithms and noise models to assess their practicality in a cryptographic context.

The primary findings revealed that while both QAOA and VQE show promise in cryptanalytic applications, they face significant challenges when scaled to cryptographically relevant dimensions. QAOA exhibited better accuracy in small, noise-free

environments but struggled under noisy conditions. On the other hand, VQE, enhanced with Conditional Value at Risk (CVaR) optimization, demonstrated more consistent performance across noise models, albeit with a conservative bias that prioritized robustness over solution accuracy. These results suggest that while quantum algorithms hold potential for cryptanalysis, substantial advancements in quantum hardware and algorithmic design are needed to fully realize this potential.

6.2 Final remarks

The study makes important contributions to the field of quantum cryptanalysis, particularly by providing a detailed examination of how quantum optimization methods can be applied to LWE problems, which form the backbone of many post-quantum cryptographic systems. The research underscores the challenges posed by NISQ hardware, particularly in managing noise and maintaining solution correctness as problem dimensions scale. The novel Hamiltonian encodings introduced in this dissertation offer new directions for encoding cryptographic problems into quantum frameworks, paving the way for further explorations in quantum cryptanalysis.

While this work did not achieve the goal of breaking high-dimensional cryptosystems like CRYSTALS-Kyber, it sets a critical foundation for future research in the domain. The findings highlight the importance of continuous evaluation of post-quantum cryptographic candidates as quantum technology progresses. In particular, the results suggest that hybrid quantum-classical approaches, when combined with advanced noise-aware optimization techniques, can serve as effective strategies for tackling cryptographic challenges in the near term.

Looking ahead, the next steps involve scaling these algorithms to larger instances, refining noise mitigation techniques, and exploring alternative hybrid approaches that combine classical pre-processing with quantum optimization. As quantum hardware improves, the methodologies and insights developed in this research will be instrumental in shaping the future landscape of cryptographic security in a post-quantum world.

In summary, this dissertation contributes to the growing body of research at the intersection of quantum computing and cryptanalysis, offering practical insights and novel approaches that advance our understanding of how quantum algorithms can be applied to cryptographic systems. It emphasizes the need for ongoing research as we approach an era where quantum computers may pose a genuine threat to currently deployed cryptographic standards.

Bibliography

[1] *AerSimulator - Qiskit Aer 0.15.0*. URL: https://qiskit.github.io/qiskit-aer/stubs/qiskit_aer.AerSimulator.html#aersimulator.

[2] M. Ajtai. "Generating hard instances of lattice problems". In: *Proceedings of the Annual ACM Symposium on Theory of Computing* Part F129452 (July 1996), pp. 99–108. ISSN: 07378017. DOI: 10.1145/237814.237838.

[3] Gorjan Alagic et al. "Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process". In: (). DOI: 10.6028/NIST.IR.8413. URL: https://doi.org/10.6028/NIST.IR.8413.

[4] Martin R. Albrecht, Amit Deo, and Kenneth G. Paterson. "Cold Boot Attacks on Ring and Module LWE Keys Under the NTT". In: *Cryptology ePrint Archive* (2018). URL: https://eprint.iacr.org/2018/672.

[5] Martin R. Albrecht et al. "Variational quantum solutions to the Shortest Vector Problem". In: *Quantum* 7 (Mar. 2023), p. 933. ISSN: 2521327X. DOI: 10.22331/q-2023-03-02-933. URL: https://quantum-journal.org/papers/q-2023-03-02-933/.

[6] Yuri Alexeev et al. "Quantum Computer Systems for Scientific Discovery". In: *PRX Quantum* 2.1 (Jan. 2021), p. 017001. ISSN: 26913399. DOI: 10.1103/PRXQUANTUM.2.017001/FIGURES/5/MEDIUM. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.2.017001.

[7] Alsop. *Quantum technology market revenue worldwide 2040, by segment*. en. https://www.statista.com/statistics/1317754/global-quantum-technology-market-revenue-forecast/. 2023.

[8] Sanjeev Arora and Rong Ge. "New Algorithms for Learning in Presence of Errors". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 6755 LNCS.PART 1 (2011), pp. 403–415. ISSN: 1611-3349. DOI: 10.1007/978-3-642-22006-

7{_}34. URL: https://link.springer.com/chapter/10.1007/978-3-642-22006-7_34.

[9] Roberto Avanzi et al. "CRYSTALS-Kyber Algorithm Specifications And Supporting Documentation (version 3.02)". In: (2021).

[10] R. Barends et al. "Superconducting quantum circuits at the surface code threshold for fault tolerance". In: *Nature 2014 508:7497* 508.7497 (Apr. 2014), pp. 500–503. ISSN: 1476-4687. DOI: 10.1038/nature13171. URL: https://www.nature.com/articles/nature13171.

[11] Elaine Barker. "NIST Special Publication 800-57 Part 1 Revision 5 Recommendation for Key Management: Part 1-General". In: *nist.gov* (2020). DOI: 10.6028/NIST.SP.800-57pt1r5. URL: https://doi.org/10.6028/NIST.SP.800-57pt1r5.

[12] Panagiotis Kl Barkoutsos et al. "Improving Variational Quantum Optimization using CVaR". In: *Quantum* 4 (Apr. 2020), p. 256. ISSN: 2521327X. DOI: 10.22331/q-2020-04-20-256. URL: https://quantum-journal.org/papers/q-2020-04-20-256/.

[13] Joao Basso et al. "The Quantum Approximate Optimization Algorithm at High Depth for MaxCut on Large-Girth Regular Graphs and the Sherrington-Kirkpatrick Model". In: *Leibniz International Proceedings in Informatics, LIPIcs* 232 (Oct. 2021). DOI: 10.4230/LIPIcs.TQC.2022.7. URL: http://arxiv.org/abs/2110.14206%20http://dx.doi.org/10.4230/LIPIcs.TQC.2022.7.

[14] Charles H. Bennett and David P. Divincenzo. "Quantum information and computation". In: *Nature 2000 404:6775* 404.6775 (Mar. 2000), pp. 247–255. ISSN: 1476-4687. DOI: 10.1038/35005001. URL: https://www.nature.com/articles/35005001.

[15] Daniel J. Bernstein and Tanja Lange. "Post-quantum cryptography". In: *Nature 2017 549:7671* 549.7671 (Sept. 2017), pp. 188–194. ISSN: 1476-4687. DOI: 10.1038/nature23461. URL: https://www.nature.com/articles/nature23461.

[16] Kishor Bharti et al. "Noisy intermediate-scale quantum algorithms". In: *Reviews of Modern Physics* 94.1 (Feb. 2022), p. 015004. ISSN: 15390756. DOI: 10.1103/RevModPhys.94.015004. URL: https://journals.aps.org/rmp/abstract/10.1103/RevModPhys.94.015004.

[17] D Boneh, V Shoup - Draft 0.5, and undefined 2020. "A graduate course in applied cryptography". In: *dlib.hust.edu.vnD Boneh, V ShoupDraft 0.5, 2020•dlib.hust.edu.vn* (). URL: https://dlib.hust.edu.vn/bitstream/HUST/18098/3/OER000000253.pdf.

[18] Joppe Bos et al. "CRYSTALS - Kyber: A CCA-Secure Module-Lattice-Based KEM". In: *Proceedings - 3rd IEEE European Symposium on Security and Privacy, EURO S and P 2018* (July 2018), pp. 353–367. DOI: 10.1109/EUROSP.2018.00032.

[19] Zvika Brakerski et al. "Classical hardness of learning with errors". In: *Proceedings of the Annual ACM Symposium on Theory of Computing* (2013), pp. 575–584. ISSN: 07378017. DOI: 10.1145/2488608.2488680. URL: https://dl.acm.org/doi/10.1145/2488608.2488680.

[20] Zvika Brakerski et al. "Learning with Errors and Extrapolated Dihedral Cosets". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10770 10769 LNCS (2018), pp. 702–727. ISSN: 16113349. DOI: 10.1007/978-3-319-76581-5{_}24/FIGURES/6. URL: https://link.springer.com/chapter/10.1007/978-3-319-76581-5_24.

[21] Fernando G. S. L. Brandao et al. "For Fixed Control Parameters the Quantum Approximate Optimization Algorithm's Objective Function Value Concentrates for Typical Instances". In: (Dec. 2018). URL: https://arxiv.org/abs/1812.04170v1.

[22] Sergey Bravyi et al. "The future of quantum computing with superconducting qubits". In: *Journal of Applied Physics* 132.16 (Oct. 2022), p. 160902. ISSN: 10897550. DOI: 10.1063/5.0082975/2837574. URL: /aip/jap/article/132/16/160902/2837574/The-future-of-quantum-computing-with.

[23] *Breaking RSA Encryption - an Update on the State-of-the-Art - Quintessence-Labs*. URL: https://www.quintessencelabs.com/blog/breaking-rsa-encryption-update-state-art#.

[24] *Building noise models — IBM Quantum Documentation*. URL: https://docs.quantum.ibm.com/guides/build-noise-models.

[25] Pengnian Cai et al. “Enhancing Quantum Approximate Optimization with CNN-CVaR Integration”. In: (June 2024). DOI: 10.21203/RS.3.RS-4460928/V1. URL: https://www.researchsquare.com%20https://www.researchsquare.com/article/rs-4460928/v1.

[26] Davide Castelvecchi. “Quantum hacking looms — but ultra-secure encryption is ready to deploy”. In: *Nature* (Aug. 2024). ISSN: 0028-0836. DOI: 10.1038/D41586-024-02623-Y. URL: https://www.nature.com/articles/d41586-024-02623-y.

[27] M. Cerezo et al. “Variational quantum algorithms”. In: *Nature Reviews Physics 2021 3:9* 3.9 (Aug. 2021), pp. 625–644. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00348-9. URL: https://www.nature.com/articles/s42254-021-00348-9.

[28] Henry Corrigan-Gibbs and Yael Kalai. “Public-key encryption from LWE & Implementing lattice-based cryptosystems”. In: *MIT* (2024).

[29] Giacomo De Palma et al. “Limitations of Variational Quantum Algorithms: A Quantum Optimal Transport Approach”. In: *PRX Quantum* 4.1 (Jan. 2023), p. 010309. ISSN: 26913399. DOI: 10.1103/PRXQUANTUM.4.010309/FIGURES/2/MEDIUM. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.4.010309.

[30] Whitfield Diffie and Martin E. Hellman. “New directions in cryptography”. In: *Secure Communications and Asymmetric Cryptosystems* (Jan. 2019), pp. 143–180. DOI: 10.1145/3549993.3550007/ASSET/CDA5146A-FEAB-46CB-9392-8CD9912315FC/ASSETS/3549993.3550007.FP.PNG. URL: https://dl.acm.org/doi/10.1145/3549993.3550007.

[31] Jintai Ding, Xiang Xie, and Xiaodong Lin. “A Simple Provably Secure Key Exchange Scheme Based on the Learning with Errors Problem”. In: *Cryptology ePrint Archive* (2012). URL: https://eprint.iacr.org/2012/688.

[32] Suguru Endo et al. “Hybrid quantum-classical algorithms and quantum error mitigation”. In: *Journal of the Physical Society of Japan* 90.3 (Mar. 2021). ISSN: 13474073. DOI: 10.7566/JPSJ.90.032001.

[33] *Estimator — IBM Quantum Documentation*. URL: https://docs.quantum.ibm.com/api/qiskit/qiskit.primitives.Estimator.

[34] Edward Farhi et al. “Quantum Algorithms for Fixed Qubit Architectures”. In: (Mar. 2017). URL: https://arxiv.org/abs/1703.06199v1.

[35] Jay M. Gambetta, Jerry M. Chow, and Matthias Steffen. “Building logical qubits in a superconducting quantum computing system”. In: *npj Quantum Information 2017 3:1* 3.1 (Jan. 2017), pp. 1–7. ISSN: 2056-6387. DOI: 10.1038/s41534-016-0004-0. URL: https://www.nature.com/articles/s41534-016-0004-0.

[36] Craig Gidney and Martin Ekerå. “How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits”. In: *Quantum* 5 (2021), pp. 1–31. ISSN: 2521327X. DOI: 10.22331/Q-2021-04-15-433.

[37] Miguel Ángel González de la Torre, Luis Hernández Encinas, and Araceli Queiruga-Dios. “Analysis of the FO Transformation in the Lattice-Based Post-Quantum Algorithms”. In: *Mathematics 2022, Vol. 10, Page 2967* 10.16 (Aug. 2022), p. 2967. ISSN: 2227-7390. DOI: 10.3390/MATH10162967. URL: https://www.mdpi.com/2227-7390/10/16/2967/htm%20https://www.mdpi.com/2227-7390/10/16/2967.

[38] *Gradient Framework - Qiskit Algorithms 0.3.0*. URL: https://qiskit-community.github.io/qiskit-algorithms/tutorials/12_gradients_framework.html.

[39] Camille Grange, Michael Poss, and Eric Bourreau. “An introduction to variational quantum algorithms for combinatorial optimization problems”. In: *4OR* 21.3 (Sept. 2023), pp. 363–403. ISSN: 16142411. DOI: 10.1007/S10288-023-00549-1/FIGURES/11. URL: https://link.springer.com/article/10.1007/s10288-023-00549-1.

[40] Lov K. Grover. “A fast quantum mechanical algorithm for database search”. In: *Proceedings of the Annual ACM Symposium on Theory of Computing* Part F129452 (July 1996), pp. 212–219. ISSN: 07378017. DOI: 10.1145/237814.237866.

[41] Stuart Hadfield et al. “From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz”. In: *Algorithms 2019, Vol. 12, Page 34* 12.2 (Feb. 2019), p. 34. ISSN: 1999-4893. DOI: 10.3390/A12020034. URL: https://www.mdpi.com/1999-4893/12/2/34/htm%20https://www.mdpi.com/1999-4893/12/2/34.

[42] Stuart Hadfield et al. “Quantum approximate optimization with hard and soft constraints”. In: *ITiCSE-WGR 2017 - Proceedings of the 2017 ITiCSE Conference on Working Group Reports* 2017-November (Jan. 2018), pp. 15–21. DOI: 10.1145/3149526.3149530. URL: https://dl.acm.org/doi/10.1145/3149526.3149530.

[43] Ethan H. Hansen et al. “Pulse-Level Variational Quantum Algorithms for Molecular Energy Calculations using Quanlse”. In: *QCCC 2023 - Proceedings of the 2023 International Workshop on Quantum Classical Cooperative Computing* 23 (Aug. 2023), pp. 9–12. DOI: 10.1145/3588983.3596686. URL: https://dl.acm.org/doi/10.1145/3588983.3596686.

[44] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. “Quantum Algorithm for Linear Systems of Equations”. In: *Physical Review Letters* 103.15 (Oct. 2009), p. 150502. ISSN: 00319007. DOI: 10.1103/PhysRevLett.103.150502. URL: https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.103.150502.

[45] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. “NTRU: A ring-based public key cryptosystem”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 1423 (1998), pp. 267–288. ISSN: 1611-3349. DOI: 10.1007/BFB0054868. URL: https://link.springer.com/chapter/10.1007/BFb0054868.

[46] Zoë Holmes et al. “Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus”. In: *PRX Quantum* 3.1 (Mar. 2022), p. 010313. ISSN: 26913399. DOI: 10.1103/PRXQUANTUM.3.010313/FIGURES/8/MEDIUM. URL: https://journals.aps.org/prxquantum/abstract/10.1103/PRXQuantum.3.010313.

[47] *Improving Variational Quantum Optimization using CVaR - Qiskit Optimization 0.6.1*. URL: https://qiskit-community.github.io/qiskit-optimization/tutorials/08_cvar_optimization.html.

[48] Ali Javadi-Abhari et al. “Quantum computing with Qiskit”. In: (May 2024). URL: https://arxiv.org/abs/2405.08810v3.

[49] Abhinav Kandala et al. “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets”. In: *Nature 2017 549:7671* 549.7671

(Sept. 2017), pp. 242–246. ISSN: 1476-4687. DOI: 10.1038/nature23879. URL: https://www.nature.com/articles/nature23879.

[50] Paul Kirchner and Pierre Alain Fouque. "Revisiting lattice attacks on overstretched NTRU parameters". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10210 LNCS (2017), pp. 3–26. ISSN: 16113349. DOI: 10.1007/978-3-319-56620-7{_}1/FIGURES/2. URL: https://link.springer.com/chapter/10.1007/978-3-319-56620-7_1.

[51] Ioannis Kolotouros and Petros Wallden. "Evolving objective function for improved variational quantum optimization". In: *Physical Review Research* 4.2 (June 2022), p. 023225. ISSN: 26431564. DOI: 10.1103/PHYSREVRESEARCH.4.023225/FIGURES/13/MEDIUM. URL: https://journals.aps.org/prresearch/abstract/10.1103/PhysRevResearch.4.023225.

[52] *Kyber*. URL: https://www.pq-crystals.org/kyber/index.shtml.

[53] Lihui Lv et al. "Using Variational Quantum Algorithm to Solve the LWE Problem". In: *Entropy 2022, Vol. 24, Page 1428* 24.10 (Oct. 2022), p. 1428. ISSN: 1099-4300. DOI: 10.3390/E24101428. URL: https://www.mdpi.com/1099-4300/24/10/1428/htm%20https://www.mdpi.com/1099-4300/24/10/1428.

[54] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. "On Ideal Lattices and Learning with Errors over Rings". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 6110 LNCS (2010), pp. 1–23. ISSN: 1611-3349. DOI: 10.1007/978-3-642-13190-5{_}1. URL: https://link.springer.com/chapter/10.1007/978-3-642-13190-5_1.

[55] Jarrod R. McClean et al. "Barren plateaus in quantum neural network training landscapes". In: *Nature Communications 2018 9:1* 9.1 (Nov. 2018), pp. 1–6. ISSN: 2041-1723. DOI: 10.1038/s41467-018-07090-4. URL: https://www.nature.com/articles/s41467-018-07090-4.

[56] Jarrod R. McClean et al. "The theory of variational hybrid quantum-classical algorithms". In: *New Journal of Physics* 18.2 (Feb. 2016), p. 023023. ISSN: 1367-2630. DOI: 10.1088/1367-2630/18/2/023023. URL: https://iopscience.

iop.org/article/10.1088/1367-2630/18/2/023023%20https://iopscience.iop.org/article/10.1088/1367-2630/18/2/023023/meta.

[57] McKinsey & Company. *Record investments in quantum technology — McKinsey*. 2023. URL: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/quantum-technology-sees-record-investments-progress-on-talent-gap.

[58] Daniele Micciancio and Chris Peikert. "Hardness of SIS and LWE with Small Parameters". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8042 LNCS.PART 1 (2013), pp. 21–39. ISSN: 1611-3349. DOI: 10.1007/978-3-642-40041-4{_}2. URL: https://link.springer.com/chapter/10.1007/978-3-642-40041-4_2.

[59] Daniele Micciancio and Oded Regev. "Lattice-based Cryptography". In: *Post-Quantum Cryptography* (Jan. 2009), pp. 147–191. DOI: 10.1007/978-3-540-88702-7{_}5. URL: https://link.springer.com/chapter/10.1007/978-3-540-88702-7_5.

[60] Evgeny Milanov. "The RSA Algorithm". In: (2009).

[61] Nikolaj Moll et al. "Quantum optimization using variational algorithms on near-term quantum devices". In: *Quantum Science and Technology* 3.3 (June 2018), p. 030503. ISSN: 2058-9565. DOI: 10.1088/2058-9565/AAB822. URL: https://iopscience.iop.org/article/10.1088/2058-9565/aab822%20https://iopscience.iop.org/article/10.1088/2058-9565/aab822/meta.

[62] Michele Mosca. "Quantum algorithms". In: *Computational Complexity: Theory, Techniques, and Applications* 9781461418009 (Nov. 2012), pp. 2303–2333. DOI: 10.1007/978-1-4614-1800-9{_}144/COVER. URL: https://link.springer.com/referenceworkentry/10.1007/978-1-4614-1800-9_144.

[63] *Noise Models (qiskit_aer.noise) - Qiskit Aer 0.15.0*. URL: https://qiskit.github.io/qiskit-aer/apidocs/aer_noise.html.

[64] Iason Papadopoulos and Jiabo Wang. "Polar Codes for Module-LWE Public Key Encryption: The Case of Kyber". In: *Cryptography 2023, Vol. 7, Page 2* 7.1 (Jan. 2023), p. 2. ISSN: 2410-387X. DOI: 10.3390/CRYPTOGRAPHY7010002. URL: https://www.mdpi.com/2410-387X/7/1/2/htm%20https://www.mdpi.com/2410-387X/7/1/2.

[65] *ParamShiftSamplerGradient — IBM Quantum Documentation*. URL: https://docs.quantum.ibm.com/api/qiskit/0.46/qiskit.algorithms.gradients.ParamShiftSamplerGradient.

[66] Chris Peikert. "Public-key cryptosystems from the worst-case shortest vector problem". In: *Proceedings of the Annual ACM Symposium on Theory of Computing* (2009), pp. 333–342. ISSN: 07378017. DOI: 10.1145/1536414.1536461. URL: https://dl.acm.org/doi/10.1145/1536414.1536461.

[67] Chris Peikert and Boston -Delft. "A Decade of Lattice Cryptography". In: *Foundations and Trends® in Theoretical Computer Science* 10.4 (Mar. 2016), pp. 283–424. ISSN: 1551-305X. DOI: 10.1561/0400000074. URL: http://dx.doi.org/10.1561/0400000074.

[68] Daniel F Perez-Ramirez. "Variational Quantum Algorithms for Combinatorial Optimization". In: 1 (July 2024). URL: https://arxiv.org/abs/2407.06421v1.

[69] Alberto Peruzzo et al. "A variational eigenvalue solver on a photonic quantum processor". In: *Nature Communications 2014 5:1* 5.1 (July 2014), pp. 1–7. ISSN: 2041-1723. DOI: 10.1038/ncomms5213. URL: https://www.nature.com/articles/ncomms5213.

[70] John Preskill. "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2 (Aug. 2018), p. 79. ISSN: 2521327X. DOI: 10.22331/q-2018-08-06-79. URL: https://quantum-journal.org/papers/q-2018-08-06-79/.

[71] Gina M Raimondo and Laurie E Locascio. "Module-Lattice-Based Key-Encapsulation Mechanism Standard". In: (Aug. 2024). DOI: 10.6028/NIST.FIPS.203. URL: https://csrc.nist.gov/pubs/fips/203/final.

[72] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *Journal of the ACM (JACM)* 56.6 (Sept. 2009). ISSN: 00045411. DOI: 10.1145/1568318.1568324. URL: https://dl.acm.org/doi/10.1145/1568318.1568324.

[73] Oded Regev. "The learning with errors problem". In: *Proceedings of the Annual IEEE Conference on Computational Complexity* (2010), pp. 191–204. ISSN: 10930159. DOI: 10.1109/CCC.2010.26.

[74] R. L. Rivest, A. Shamir, and L. Adleman. "A method for obtaining digital signatures and public-key cryptosystems". In: *Communications of the ACM* 21.2 (Feb. 1978), pp. 120–126. ISSN: 15577317. DOI: 10.1145/359340.359342. URL: https://dl.acm.org/doi/10.1145/359340.359342.

[75] Jonathan Romero, Jonathan P. Olson, and Alan Aspuru-Guzik. "Quantum autoencoders for efficient compression of quantum data". In: *Quantum Science and Technology* 2.4 (Aug. 2017), p. 045001. ISSN: 2058-9565. DOI: 10.1088/2058-9565/AA8072. URL: https://iopscience.iop.org/article/10.1088/2058-9565/aa8072%20https://iopscience.iop.org/article/10.1088/2058-9565/aa8072/meta.

[76] Bruce Schneier. "NIST's Post-Quantum Cryptography Standards Competition". In: *IEEE Security and Privacy* 20.5 (2022), pp. 107–108. ISSN: 15584046. DOI: 10.1109/MSEC.2022.3184235.

[77] Maria Schuld and Francesco Petruccione. "Quantum Science and Technology Supervised Learning with Quantum Computers". In: (). URL: http://www.springer.com/series/10039.

[78] A Kak - Lecture Notes on "Computer Security, Network, and undefined 2015. "Lecture 12: Public-Key Cryptography and the RSA Algorithm". In: *engineering.purdue.eduA KakLecture Notes on "Computer and Network Security, Purdue, 2015•engineering.purdue.edu* (2024). URL: https://engineering.purdue.edu/kak/courses-i-teach/compsec/NewLectures/Lecture12.pdf.

[79] Monit Sharma, Hoong Chuin Lau, and Rudy Raymond. "Quantum-Enhanced Simulation-Based Optimization for Newsvendor Problems". In: (Mar. 2024). URL: https://arxiv.org/abs/2403.17389v3.

[80] Ruslan Shaydulin and Yuri Alexeev. "Evaluating Quantum Approximate Optimization Algorithm: A Case Study". In: *2019 10th International Green and Sustainable Computing Conference, IGSC 2019* (Oct. 2019). DOI: 10.1109/IGSC48788.2019.8957201.

[81] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *https://doi.org/10.1137/S0036144598347011* 41.2 (Aug. 2006), pp. 303–332. ISSN: 00361445. DOI: 10.1137/S0036144598347011. URL: https://epubs.siam.org/doi/10.1137/S0036144598347011.

[82] *Solving the Quantum Decryption 'Harvest Now, Decrypt Later' Problem - SecurityWeek*. URL: https://www.securityweek.com/solving-quantum-decryption-harvest-now-decrypt-later-problem/.

[83] Damian Silvio Steiger. "Software and Algorithms for Quantum Computing". In: (2018). DOI: 10.3929/ethz-b-000322770. URL: https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/322770/thesis.pdf.

[84] Samuel Stein et al. "EQC: Ensembled Qantum Computing for Variational Qantum Algorithms". In: *Proceedings - International Symposium on Computer Architecture* 22 (June 2022), pp. 59–71. ISSN: 10636897. DOI: 10.1145/3470496.3527434. URL: https://dl.acm.org/doi/10.1145/3470496.3527434.

[85] Martin Suchara et al. "A Quantum Approximate Optimization Algorithm". In: *Proceedings of the 3rd International Workshop on Post-Moore's Era Supercomputing* (Nov. 2014), p. 3. URL: https://arxiv.org/abs/1411.4028v1.

[86] Chukwudubem Umeano and Jonathan Halliwell. "IMPERIAL COLLEGE LONDON DEPARTMENT OF PHYSICS Quantum Algorithms: A Review". In: (2021).

[87] Jacob Viertel and B Eng. "QUANTUM COMPUTING FOR DESIGNING BEHAVIORAL MODEL AND QUANTUM MA-CHINE LEARNING ON A HUMANOID ROBOT". In: ().

[88] *VQE with Qiskit Aer Primitives*. URL: https://github.com/Qiskit/qiskit-tutorials/blob/master/tutorials/algorithms/03_vqe_simulation_with_noise.ipynb.

[89] Yael Kalai. *Encryption Schemes: MIT Lecture Series - 6.5610*. 2024.

[90] Kan Yao et al. "Towards crystals-Kyber: A M-LWE cryptoprocessor with area-time trade-off". In: *Proceedings - IEEE International Symposium on Circuits and Systems* 2021-May (2021). ISSN: 02714310. DOI: 10.1109/ISCAS51556.2021.9401253.

[91] Leo Zhou et al. "Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices". In: *Physical Review X* 10.2 (June 2020), p. 021067. ISSN: 21603308. DOI: 10.1103/PHYSREVX.10.021067/FIGURES/15/MEDIUM. URL: https://journals.aps.org/prx/abstract/10.1103/PhysRevX.10.021067.

Appendix A

Additional background

A.1 Economic significance

The economic implications of quantum computing are profound, as demonstrated by recent projections. According to [57], the quantum technology market could generate $93 billion in revenue by 2040, with quantum computing being the primary driver [7]. This forecast underscores the technology's significance and its potential economic impact. Additionally, the quantum security market is projected to grow from $500 million in 2022 to $9.8 billion by 2030, driven largely by post-quantum cryptography [7]. This growth trajectory indicates a rapidly expanding industry that is becoming crucial in the global economic framework. The commercial potential of quantum computing is further evidenced by the $5.4 billion raised by startups in this sector as of 2022, surpassing other areas like quantum sensing and communications [7]. This substantial investment reflects the sector's viability and anticipated transformative impact. The presence of 223 quantum computing startups as of 2022 [7] further attests to the field's dynamism and the increasing recognition of its potential to address complex problems beyond the capabilities of classical computing.

The 2023 Quantum Technology Monitor report further illustrates the global impact of quantum computing, highlighting significant investments and advancements [57]. In 2022, quantum technology startups attracted $2.35 billion, demonstrating robust growth and investor confidence. This capital influx is driving technological breakthroughs, such as IBM's development of a 1121-qubit processor and plans for a 4,000-qubit processor by 2025 [57]. These achievements mark significant milestones in quantum computing and reflect the field's rapid evolution.

Near-term quantum processing units (QPUs) hold the potential to tackle problems

currently unsolvable by classical computers. Quantum computing promises exponential advancements in areas like cryptography, optimization, and complex simulations [35], paving the way for new research and development opportunities. However, the technology is still in its early stages, within the NISQ era, presenting distinct challenges and limitations.

A.2 VQA

Variational Quantum Algorithms are designed for the NISQ era, combining quantum circuits with classical optimization to work within hardware limits like noise, decoherence, and shallow circuit depths. These algorithms adjust parameterized quantum circuits to minimize problem-specific cost functions, enabling effective exploration of solution spaces despite current hardware constraints [27, 69, 56]. VQAs remain relevant even as quantum hardware evolves, retaining their utility in a variety of computational tasks [16].

VQAs have been successfully applied to domains like quantum chemistry and optimization. For instance, VQE approximates molecular ground states, while QAOA tackles combinatorial problems like Max-Cut, sometimes outperforming classical methods [85, 68]. This adaptability underscores their importance in both current and future quantum computing contexts [70].

A.2.1 Current quantum hardware and its limitations

VQAs are limited by NISQ hardware, which typically features 50-100 qubits, short coherence times, and high error rates. These constraints restrict circuit depth, necessitating algorithms that can produce meaningful results within these limits [16, 70]. VQAs' shallow circuits make them suitable for these conditions, and their utility will persist even with the advent of fault-tolerant quantum computing.

Noise, including gate errors and decoherence, is a significant challenge in NISQ devices, with error rates between 10^{-3} and 10^{-2} per gate [32, 29]. VQAs mitigate these issues through shallow circuits and error correction techniques like zero-noise extrapolation, enhancing computation reliability on current hardware [61, 29].

A.2.2 VQAs as a solution to NISQ-Era challenges

VQAs' hybrid design leverages classical optimization to adjust quantum circuit parameters, allowing them to find approximate solutions despite hardware noise and constraints [56, 16]. Shallow circuits and noise-aware techniques like pulse-level control make VQAs particularly effective for current quantum devices [84, 43].

A.3 CRYSTALS-Kyber

Kyber CRYSTALS, a lattice-based key encapsulation mechanism (KEM), has been standardized under the NIST Post-Quantum Cryptography initiative due to its resilience against both classical and quantum adversaries. The cryptographic foundation of Kyber is based on the hardness of the LWE problem and its extensions, particularly Ring-LWE and Module-LWE. Among these, Module-LWE was selected for Kyber because it offers a balance between security and efficiency by mitigating certain algebraic weaknesses inherent in Ring-LWE. Specifically, the ring structure in Ring-LWE introduces exploitable algebraic properties, such as the ideal lattice structure and symmetries in cyclotomic fields, which specialized attacks can target. These properties simplify the problem under certain parameterizations, leading to reduced security margins compared to Module-LWE [18, 9].

In contrast to Ring-LWE, Module-LWE generalizes the problem to modules over polynomial rings, reducing the exploitable structure while preserving efficiency. The module structure reduces the inherent symmetry that makes Ring-LWE more susceptible to attacks. Additionally, Module-LWE allows for more granular parameterization, providing better control over security and performance trade-offs by adjusting the underlying ring dimensions and error distribution [66, 31, 4].

Kyber's modular design supports versatile deployment, from resource-constrained IoT devices to high-performance systems. However, the use of Module-LWE introduces complexities in parameter tuning, particularly in balancing error distribution, modulus size, and computational efficiency. For instance, the standard Kyber parameters (e.g., modulus $q = 3329$ and polynomial degree $n = 256$) are selected to achieve a security level equivalent to 128 bits, balancing security and performance across varied applications [90, 18]. Despite these complexities, the flexibility of Module-LWE makes it adaptable for different security levels and performance requirements [64].

The choice to adopt Module-LWE in Kyber is partly driven by its resilience against

known attacks on LWE and Ring-LWE. Advanced lattice reduction techniques like the Blockwise Korkine-Zolotarev (BKZ) algorithm, along with algebraic attacks that exploit the ring structure in Ring-LWE, present significant risks for cryptosystems relying on these variants. For instance, attacks utilizing BKZ with block sizes optimized for q can drastically reduce security margins. The structured algebra of Ring-LWE introduces vulnerabilities not present in the less-structured Module-LWE setting, reinforcing the decision to prefer Module-LWE for Kyber [50, 20].

Kyber employs a hybrid encryption scheme combining a KEM with symmetric encryption. In Kyber, key generation involves producing a random matrix $\mathbf{A}$, along with small secret vectors $\mathbf{s}$ and $\mathbf{e}$. The public key is $(\mathbf{A}, \mathbf{A} \cdot \mathbf{s} + \mathbf{e})$, while the secret key is $\mathbf{s}$. Encryption relies on generating a random vector $\mathbf{r}$ and error vectors, with the ciphertext encoding the message through Module-LWE's hardness properties. Decryption recovers the original message using the secret key $\mathbf{s}$ while maintaining low error margins, ensuring reliable recovery without information leakage [18, 9].

To achieve IND-CCA2 security, a requirement for real-world cryptosystems, Kyber incorporates the Fujisaki-Okamoto transformation. This transformation, crucial for converting an IND-CPA scheme into a CCA2-secure KEM, involves a combination of random oracles and hashing mechanisms to reinforce security against adaptive chosen-ciphertext attacks [9]. This structure, alongside the optimized modular arithmetic and polynomial operations, ensures Kyber's suitability for a wide range of platforms [18].

Appendix B

Derivation of the simplified $C(\vec{x})$

We aim to derive the simplified expression for $C(\vec{x})$ when $\vec{x} = \vec{s}$ (where $\vec{s}$ is the secret vector). Recall that $C(\vec{x})$ is defined as:

$$C(\vec{x}) := \sum_i \left[\left(\sum_j A_{ij} x_j \right) - b_i \right] \tag{B.1}$$

where all operations are taken modulo q.

Recall that b_i is defined as:

$$b_i := \left(\sum_j A_{ij} s_j + e_i \right) \mod q$$

Substituting this into the expression for $C(\vec{s})$, we have:

$$C(\vec{s}) = \sum_i \left(\left[\left(\sum_j A_{ij} s_j \right) \mod q - \left(\sum_j A_{ij} s_j - e_i \right) \mod q \right) \mod q \right]$$

where the outer most modulo is due to the subtraction. Using the modular arithmetic distribution law:

$$(a-b) \mod q = (a \mod q - b \mod q) \mod q,$$

we can simplify the expression to:

$$C(\vec{s}) = \sum_i \left[\left(\sum_j A_{ij} s_j \right) - \left(\sum_j A_{ij} s_j \right) + e_i \mod q \right]$$

Notice that the terms $\sum_j A_{ij} s_j \mod q$ cancel out, leaving:

$$C(\vec{s}) = \sum_i e_i \mod q$$

By our definition of modulo (see Definition 2.1):

$$a \mod q := a - \left\lfloor \frac{a}{q} \right\rfloor q,$$

if e_i is a small negative value, then $e_i \mod q$ simplifies to $q - e_i$. Therefore, for those indices i where $e_i < 0$, the term $e_i \mod q$ becomes $q - e_i$, which is large compared to $\frac{q}{4}$.

This is problematic because for those indices where e_i is negative, the resulting value $q - e_i$ significantly increases the contribution of that term to $C(\vec{s})$. Consequently, the overall sum can become large, potentially causing issues in the security of the scheme.

Thus, while $C(\vec{s})$ generally simplifies to a sum of e_i terms modulo q, for certain indices where $e_i < 0$, those contributions become large, leading to a significant deviation from the expected behavior.

Appendix C

Code snippets

C.1 QAOA

C.1.1 Classical Comparison using Brute Force

```python
# Brute force solution using NumPy eigensolver.
def brute_force_solution(hamiltonian):
    numpy_eigensolver = NumPyMinimumEigensolver()
    result = numpy_eigensolver.compute_minimum_eigenvalue(hamiltonian)
    return result
```

C.1.2 Classical Exhaustive Search for LWE Problem

```python
# Iterating over all possible values of x where each element of x is in the range [0, q-1].
    for i in range(q**n):
        # Generate the current vector x in base q.
        x = np.array([(i // q**j) % q for j in range(n)])

        # Compute the Hamiltonian value for the current vector x.
        value = lwe.hamiltonian_value(x)

        # Reset best solutions if a new minimal value is found.
        if value < minimal_value:
            minimal_value = value
            best_solutions = [x]
        # Add to best solutions if the same minimal value is found.
        elif value == minimal_value:
            best_solutions.append(x)
```

```
16
17     return minimal_value, best_solutions
```

C.1.3 QUBO Hamiltonian Construction

```
# Define the quadratic program.
qubo = QuadraticProgram()

# Add binary variables to represent each integer variable.
for i in range(n):
    for j in range(num_bits):
        qubo.binary_var(name=f"x{i}_{j}")

# Add auxiliary variables k_i and r_i.
for i in range(m):
    for j in range(num_bits):
        qubo.binary_var(name=f"k{i}_{j}")

for i in range(m):
    for j in range(num_bits):
        qubo.binary_var(name=f"r{i}_{j}")

# Define the coefficients for the quadratic and linear terms.
Q = A.T @ A
c = -2 * b.T @ A
const = (b.T @ b).item()

# Convert Q and c to dictionaries to match the new binary
  ↪ variables.
quadratic_dict = {}
linear_dict = {}

# 1. r_i^2 term.
for i in range(m):
    for k in range(num_bits):
        for l in range(num_bits):
            coeff = (2**k) * (2**l)
            if coeff != 0:
                quadratic_dict[(f"r{i}_{k}", f"r{i}_{l}")] = (
                    quadratic_dict.get((f"r{i}_{k}", f"r{i}_{l}"), 0) +
                    ↪ coeff
```

```
                )

# 2. P (sum_j A_ij x_j)^2 term.
for i in range(m):
    for j in range(n):
        for j_prime in range(n):
            for k in range(num_bits):
                for l in range(num_bits):
                    coeff = P * A[i, j] * A[i, j_prime] * (2**k) * (2**l)
                    if coeff != 0:
                        quadratic_dict[(f"x{j}_{k}", f"x{j_prime}_{l}")] = (
                            quadratic_dict.get((f"x{j}_{k}",
                            ↪ f"x{j_prime}_{l}"), 0)
                            + coeff
                        )

# 3. P b_i^2 term.
const += P * np.sum(b**2)

# 4. P (q k_i)^2 term.
for i in range(m):
    for k in range(num_bits):
        for l in range(num_bits):
            coeff = P * (q**2) * (2**k) * (2**l)
            if coeff != 0:
                quadratic_dict[(f"k{i}_{k}", f"k{i}_{l}")] = (
                    quadratic_dict.get((f"k{i}_{k}", f"k{i}_{l}"), 0) +
                    ↪ coeff
                )

# 5. P r_i^2 term.
for i in range(m):
    for k in range(num_bits):
        for l in range(num_bits):
            coeff = P * (2**k) * (2**l)
            if coeff != 0:
                quadratic_dict[(f"r{i}_{k}", f"r{i}_{l}")] = (
                    quadratic_dict.get((f"r{i}_{k}", f"r{i}_{l}"), 0) +
                    ↪ coeff
                )

# 6. -2 P sum_j A_ij x_j b_i term.
```

```
for i in range(m):
    for j in range(n):
        for k in range(num_bits):
            coeff = -2 * P * A[i, j] * b[i] * (2**k)
            if coeff != 0:
                linear_dict[f"x{j}_{k}"] = linear_dict.get(f"x{j}_{k}", 0) +
                ↪ coeff

# 7. -2 P sum_j A_ij x_j q k_i term.
for i in range(m):
    for j in range(n):
        for k in range(num_bits):
            for l in range(num_bits):
                coeff = -2 * P * A[i, j] * q * (2**k) * (2**l)
                if coeff != 0:
                    quadratic_dict[(f"x{j}_{k}", f"k{i}_{l}")] = (
                        quadratic_dict.get((f"x{j}_{k}", f"k{i}_{l}"), 0) +
                        ↪ coeff
                    )

# 8. -2 P sum_j A_ij x_j r_i term.
for i in range(m):
    for j in range(n):
        for k in range(num_bits):
            for l in range(num_bits):
                coeff = -2 * P * A[i, j] * (2**k) * (2**l)
                if coeff != 0:
                    quadratic_dict[(f"x{j}_{k}", f"r{i}_{l}")] = (
                        quadratic_dict.get((f"x{j}_{k}", f"r{i}_{l}"), 0) +
                        ↪ coeff
                    )

# 9. 2 P b_i q k_i term.
for i in range(m):
    for k in range(num_bits):
        coeff = 2 * P * b[i] * q * (2**k)
        if coeff != 0:
            linear_dict[f"k{i}_{k}"] = linear_dict.get(f"k{i}_{k}", 0) +
            ↪ coeff

# 10. 2 P b_i r_i term.
for i in range(m):
```

```
    for k in range(num_bits):
        coeff = 2 * P * b[i] * (2**k)
        if coeff != 0:
            linear_dict[f"r{i}_{k}"] = linear_dict.get(f"r{i}_{k}", 0) +
            ↪ coeff

# 11. 2 P q k_i r_i term.
for i in range(m):
    for k in range(num_bits):
        for l in range(num_bits):
            coeff = 2 * P * q * (2**k) * (2**l)
            if coeff != 0:
                quadratic_dict[(f"k{i}_{k}", f"r{i}_{l}")] = (
                    quadratic_dict.get((f"k{i}_{k}", f"r{i}_{l}"), 0) +
                    ↪ coeff
                )

# Set the objective function.
qubo.minimize(constant=const, linear=linear_dict, quadratic=quadratic_dict)
```

C.1.4 Solving the QUBO with QAOA

```
# Convert the LWE instance to an Ising Hamiltonian.
qubo = qubo_hamiltonian(lwe, P=P)
op, offset = qubo.to_ising()
sampler = AerSampler(backend_options={"method": "statevector"})
qaoa = QAOA(sampler=sampler, optimizer=COBYLA())
result = qaoa.compute_minimum_eigenvalue(op)
```

C.2 VQE

C.2.1 Gradient Descent Optimization

```
def gradient_descent(self, alpha=None, which_parameters="all", shots=None):

    if which_parameters == "all":
        which_parameters = range(self.num_of_params)

    thetas = self.angles.copy()

```

```
    print("Gradient descent optimization.")

    initial_expectations = self.expectation(thetas, shots)[0]
    print(f"Initial expectation value: {initial_expectations}.")

    expectation_vals = [initial_expectations]

    for _ in range(self.maxiter):
        gradient = self.gradient(thetas, which_parameters, shots=shots)
        thetas = [
            thetas[i] - alpha * gradient[i] for i in range(len(self.angles))
        ]
        expectation, solution_probs = self.expectation(thetas, shots)
        expectation_vals.append(expectation)

        print(f"Gradient descent step - {_}: Expectation: {expectation}",
        ↪ "Most probable solutions:", solution_probs)

    return expectation_vals, self.expectation(thetas, shots)[1]
```

C.2.2 Parameter Shift Rule Gradient Calculation

```
def gradient(self, angles, which_derivatives, shots=None):
    grads = []

    # for i in range(len(angles)):
    for i in which_derivatives:
        shift_params_plus = angles.copy()
        shift_params_minus = angles.copy()
        shift_params_plus[i] += self.eta
        shift_params_minus[i] -= self.eta
        grad = (
            self.expectation(shift_params_plus, shots)[0]
            - self.expectation(shift_params_minus, shots)[0]
        ) / (2 * self.eta)
        grads.append(grad)

    return np.array(grads)
```

C.2.3 Expectation Value Calculation using AerSampler

```python
def expectation(self, angles, shots=None):
    qc = self.ansatz_circuit.assign_parameters(angles)
    result = sampler.run([qc]).result()
    counts = result.quasi_dists[0].binary_probabilities()
    expectation = 0
    solution_probs = []

    cvar_percent = 1
    # Sort counts by probability in descending order.
    sorted_counts = sorted(counts.items(), key=lambda item: item[1],
    ↪ reverse=True)

    # Calculate the cumulative probability and keep the top x\%
    cumulative_prob = 0
    selected_counts = []
    for bitstring, prob in sorted_counts:
        cumulative_prob += prob
        selected_counts.append((bitstring, prob))
        if cumulative_prob >= cvar_percent:
            break

    # Calculate the expectation value based on the selected top
    ↪ x\% outcomes
    expectation = 0
    solution_probs = []

    # This is the theoretical maximum cost for the problem.
    for bitstring, prob in selected_counts:
        x, valid = self.lwe.interpret_bitstring(bitstring)
        solution_probs.append((x, prob, valid))
        cost = self.lwe.hamiltonian_value(x)

        if not valid:
            # Calculate the penalty factor based on how much x
            ↪ exceeds q-1
            penalty_factor = sum(max(0, xi - (self.lwe.q - 1)) for xi in x) *
            ↪ 10 + 10
            expectation += cost * prob / cumulative_prob + penalty_factor
        else:
            expectation += cost * prob / cumulative_prob
```