

University of Minho
School of Engineering

Márcio Eduardo Lima Mano

ADAPT-VQC
Adaptive Variational Quantum Classifier



University of Minho
School of Engineering

Márcio Eduardo Lima Mano

ADAPT-VQC
Adaptive Variational Quantum Classifier

Masters Dissertation
Master's in Physics Engineering

Dissertation supervised by
Luís Paulo Santos
André Sequeira

Copyright and Terms of Use for Third Party Work

This dissertation reports on academic work that can be used by third parties as long as the internationally accepted standards and good practices are respected concerning copyright and related rights.

This work can thereafter be used under the terms established in the license below.

Readers needing authorization conditions not provided for in the indicated licensing should contact the author through the RepositóriUM of the University of Minho.

License granted to users of this work:



CC BY

<https://creativecommons.org/licenses/by/4.0/>

Acknowledgements

I would like to extend my heartfelt appreciation to all those who played a crucial role in the completion of this work and the attainment of my master's degree.

First and foremost, I want to express my deep gratitude to my supervisors. They not only introduced me to the captivating realm of quantum machine learning but also proposed an engaging and thought-provoking theme for this dissertation. Their unwavering support, encouragement, and guidance throughout the entire process were invaluable.

I would also like to extend my heartfelt thanks to my parents, brother, and the rest of my family. Over the past five years, they have been unwavering pillars of strength, providing constant support and comfort during the most challenging moments. Their presence and understanding mean the world to me, and I am profoundly grateful for their love and encouragement.

I want to express my sincere appreciation to my girlfriend for being by my side and providing unwavering support over the past years. Your companionship and encouragement have been a constant source of strength, and I am truly thankful for your presence in my life.

I'd like to extend my gratitude to all my friends, both old and new, who have accompanied me on this journey. Your friendship has been a source of joy and support, and the good times we've shared both inside and outside of classes have enriched my life. Thank you for being a part of my experience and for the cherished moments we've created together.

This work is dedicated to each one of you, and I am immensely grateful for the privilege of sharing this journey with all of you. Your contributions, support, and presence have made this accomplishment all the more meaningful. Thank you for being an integral part of my journey.

Statement of Integrity

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, Braga, october 2023

Márcio Eduardo Lima Mano

Abstract

This dissertation explores the potential of Quantum Computing, particularly in the context of Variational Quantum Algorithms (VQA), like the Variational Quantum Classifier (VQC). It focuses on overcoming challenges such as noise in quantum circuits and optimization complexities.

The research introduces adaptive strategies for VQC, enabling dynamic adjustments to circuit depth and expressibility during training. This flexibility aims to improve classification accuracy on datasets.

The dissertation starts with a literature review of VQA algorithms, especially adaptive strategies, drawing insights from various domains, including chemistry.

Next, it details the proposed adaptive approaches for VQC and presents rigorous experiments to evaluate their performance across diverse datasets, comparing them to the standard VQC. The results show promise with improved accuracy and reduced circuit depth.

However, it's important to note that this work primarily serves as an introduction to the concept of adaptive approaches for classification, focusing on enhancing VQC within its existing context.

In summary, this dissertation provides insights into enhancing VQC performance and contributes incrementally to quantum computation in the realm of classification. Adaptive VQC addresses challenges posed by noisy quantum devices and offers opportunities for further research in quantum classification algorithms.

Keywords QML, VQC, VQA, Depth, Ansatz

Resumo

Esta dissertação explora o potencial da Computação Quântica, particularmente no contexto de Algoritmos Quânticos Variacionais (VQA), como o Classificador Quântico Variacional (VQC). Ele se concentra na superação de desafios como ruído em circuitos quânticos e complexidades de otimização.

Esta dissertação introduz estratégias adaptativas para VQC, permitindo ajustes dinâmicos na profundidade e expressibilidade do circuito durante o treino. Esta flexibilidade visa melhorar a precisão da classificação de um conjunto de dados.

A dissertação começa com uma revisão da literatura sobre algoritmos VQA, especialmente estratégias adaptativas, extraindo insights de vários domínios, incluindo da química.

A seguir, detalha as abordagens adaptativas propostas para VQC e apresenta experimentos rigorosos para avaliar seu desempenho em diversos conjuntos de dados, comparando-os com o VQC padrão. Os resultados são promissores pois mostraram uma maior precisão e profundidade de circuito reduzida.

No entanto, é importante notar que este trabalho serve principalmente como uma introdução ao conceito de abordagens adaptativas para classificação, focando no aprimoramento do VQC dentro do seu contexto existente.

Em resumo, esta dissertação fornece ideias sobre como melhorar o desempenho do VQC e contribui de forma incremental para a computação quântica no domínio da classificação. O VQC adaptativo aborda desafios colocados por dispositivos quânticos ruidosos e oferece oportunidades para pesquisas adicionais em algoritmos de classificação quântica.

Palavras-chave QML, VQC, VQA, Profundidade, Ansatz

Contents

- 1 Introduction 1**
 - 1.1 Context 1
 - 1.2 Motivation 3
 - 1.3 Contribution 4
 - 1.4 Summary 5

- 2 Variational Quantum Algorithms 6**
 - 2.1 Context 6
 - 2.2 Standard VQA's 7
 - 2.2.1 VQE 7
 - 2.2.2 QAOA 8
 - 2.2.3 VQC 8
 - 2.3 Adaptive VQA's 9
 - 2.3.1 Rotoselect Algorithm 9
 - 2.3.2 ADAPT 11
 - 2.3.3 TETRIS-ADAPT 12
 - 2.3.4 Vans 13
 - 2.3.4.1 Simplification of the Circuit 14
 - 2.3.4.2 Parameter Optimization 14
 - 2.3.4.3 Vans characteristics 15
 - 2.3.5 Summary 15

- 3 Problems, Challenges, and Proposed Solutions 19**
 - 3.1 Proposed Solutions 20
 - 3.1.1 Rotoselect 21
 - 3.1.2 ADAPT 24

4	Experimental Result: Rotoselect	28
4.1	Analysis of Performance for Two-Qubit Systems	28
4.2	Analysis of Performance for Three-Qubit Systems	35
4.3	Analysis of Performance for Four-Qubit Systems	40
4.4	Summary	46
5	Experimental Result: ADAPT	48
5.1	Analysis of Performance for Two-Qubit Systems	48
5.2	Analysis of Performance for Three-Qubit Systems	51
5.3	Analysis of Performance for Four-Qubit Systems	52
5.4	Summary	53
6	Comparative Analysis	55
6.1	Analysis of Performance for Two-Qubit Systems	55
6.2	Analysis of Performance for Three-Qubit Systems	56
6.3	Analysis of Performance for Four-Qubit Systems	58
6.4	Summary	60
7	Conclusions and future work	62
7.1	Conclusions	62
7.2	Prospect for future work	63
A	Support work	69
A.1	Rotoselect	69
	A.1.1 Four-Qubit Systems AngleEmbedding Encode	69
A.2	ADAPT	69
	A.2.1 Four-Qubit Systems AngleEmbedding Encode	69
A.3	Standard VQC	70
	A.3.1 Two-Qubit Systems	70
	A.3.2 Three-Qubit Systems	71
	A.3.3 Four-Qubit Systems	72
A.4	Comparative Analysis: Circuits	74
	A.4.1 Circuit Analysis: Two Qubits	74
	A.4.2 Circuit Analysis: Three Qubits	76

A.4.3	Circuit Analysis: Four Qubits	78
-------	---	----

List of Figures

- 1 Schematic diagram of a **VQA**, from [1] 2
- 2 An illustrative example of the Rotoselect algorithm, sourced from [2] 10
- 3 Schematic representation of the Rotoselect algorithm adapted for a classification problem, depicting the two qubits engaged in the process. 21
- 4 Examples of Exponentiation of Pauli Strings¹ 25
- 5 Single Layer of IQPEncoding Encoding for Two Qubit 29
- 6 Structure of the Initial Single Layer in the Fixed Ansatz for the Two-Qubit Rotoselect Algorithm 29
- 7 Evaluating the Cost and Approximated Cost for the Quantum Circuit Architecture With a Single Layer of Encoding and Ansatz and Two Qubits 30
- 8 Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, and Two Qubits . . 30
- 9 Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, and Two Qubits 30
- 10 Quantum Circuit Architecture: Two Encode Layers, One Ansatz Repetition, and Two Qubits 31
- 11 Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, and Two Qubits 31
- 12 Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, and Two Qubits . 32
- 13 Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, and Two Qubits 32
- 14 Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4) 33
- 15 Comparison of Initial Ansatz Structures for Different Numbers of Ansatz Layers 33
- 16 Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) while maintaining a fixed number of encode layers (one encode layer) with normalized curves . 34

17	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) while maintaining a fixed number of encode layers (two encode layers) with normalized curves	34
18	Examples of Different Qubit Entanglement Strategies	35
19	Cost and Approximated Cost Evaluation for the Quantum Circuit Architecture with One Encode Repetition, One Ansatz Repetition and Three Qubits	36
20	Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, Linear Entanglement , and Three Qubits	36
21	Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, Full Entanglement , and Three Qubits	37
22	Quantum Circuit Architecture: One Encode Layer, Two Ansatz Layers, Linear Entanglement , and Three Qubits	37
23	Quantum Circuit Architecture: One Encode Layer, Two Ansatz Layers, Full Entanglement , and Three Qubits	38
24	Quantum Circuit Architecture: One Encode Layer, Three Ansatz Layers, Linear Entanglement , and Three Qubits	38
25	Quantum Circuit Architecture: One Encode Layer, Three Ansatz Layers, Full Entanglement , and Three Qubits	39
26	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) using the "Linear Entanglement" Strategy, with normalized curves.	39
27	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) using the "Full Entanglement" Strategy, with normalized curves.	40
28	Cost and Approximated Cost Evaluation for the Quantum Circuit Architecture with One Encode Layer, One Ansatz Layer, and Four Qubits	41
29	Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, Linear Entanglement , and Four Qubits	42
30	Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, Full Entanglement , and Four Qubits	42
31	Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, Linear Entanglement , and Four Qubits	43
32	Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, Full Entanglement , and Four Qubits	43

33	Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, Linear Entanglement , and Four Qubits	44
34	Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, Full Entanglement , and Four Qubits	44
35	Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4), with normalized curves.	45
36	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves (One Encode repetition).	45
37	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves (Two Encode repetition).	46
38	Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Two Qubits	49
39	Quantum Circuit Architecture: Two Encode Layers, Ten Parameterized Operators and Two Qubits	49
40	Quantum Circuit Architecture: Three Encode Layers, Ten Parameterized Operators and Two Qubits	50
41	Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4)	51
42	Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Three Qubits	52
43	Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Four Qubits	53
44	Quantum Circuit Architecture: Two Encode Layers, Ten Parameterized Operators and Four Qubits	53
45	Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4))	54
46	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT. (These curves were sampled)	56
47	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT for a Single Ansatz Layer (These curves were sampled).	57

48	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Ansatz Layers (These curves were sampled).	57
49	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Single Encode Layer and Single Ansatz Layer (These curves were sampled).	58
50	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Single Encode Layer and Two Ansatz Layers (These curves were sampled).	59
51	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Encode Layers and One Ansatz Layer (These curves were sampled).	60
52	Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Encode Layers and Two Ansatz Layers (These curves were sampled).	60
53	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves, using angle-based encoding.	69
54	Quantum Circuit Architecture: Angle Based Encoding, Ten Parameterized Operators and Four Qubits	70
55	Comparison of Quantum Circuits with Different Numbers of Encode Layers (E1-E4) and One Ansatz Layer (Smoothed Curves).	70
56	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and a Single Encode Layer (Smoothed Curves).	71
57	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and Two Encode Layers (Smoothed Curves).	71
58	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) Using Conventional VQC.	72
59	Comparison of Quantum Circuits with Different Numbers of Encode Layers (E1-E4) and a Single Ansatz Layer (Smoothed Curves).	72
60	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and a Single Encode Layer (Smoothed Curves).	73
61	Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and Two Encode Layers (Smoothed Curves).	73
62	Antazes Generated by the Rotoselect Algorithm for a Single Ansatz Layer	74
63	Antazes used by the Standard VQC version for a Single Ansatz Layer	74
64	Antazes Generated by the ADAPT Algorithm	75

65	Antazes Generated by the Rotoselect Algorithm for a Single Ansatz Layer with Different Entanglement Strategies	76
66	Antazes Generated by the Rotoselect Algorithm for Two Ansatz Layers with Different Entanglement Strategies	76
67	Anstaz used by the Standard VQC version for a Single Encode and Anstaz Layer with Different Entanglement Strategies	76
68	Anstaz used by the Standard VQC version for a Single Encode and Two Anstaz Layers with Different Entanglement Strategies	76
69	Antazes Generated by the ADAPT Algorithm	77
70	Antazes Generated by the Rotoselect Algorithm for a Single Encode and Anstaz Layer with Different Entanglement Strategies	78
71	Antazes Generated by the Rotoselect Algorithm for a Single Encode Layer and Two Anstaz Layers with Different Entanglement Strategies	78
72	Antazes Generated by the Rotoselect Algorithm for Two Encode and a Single Anstaz Layer with Different Entanglement Strategies	78
73	Antazes Generated by the Rotoselect Algorithm for Two Encode Layers and Two Anstaz Layers with Different Entanglement Strategies	78
74	Anstaz used by the Standard VQC version for a Single Encode and Anstaz Layer with Different Entanglement Strategies	79
75	Anstaz used by the Standard VQC version for a Single Encode and Two Anstaz Layers with Different Entanglement Strategies	79
76	Antazes Generated by the ADAPT Algorithm	80

List of Tables

1 Adaptive Algorithms simple overview 18

Acronyms

ADAPT Adaptive Derivative Assembled Problem Tailored.

ML Machine Learning.

NISQ Noisy Intermediate-Scale Quantum.

PCA Principal Component Analysis.

QAOA Quantum Approximate Optimization Algorithm.

QML Quantum Machine Learning.

QUBO Quadratic Unconstrained Binary Optimization.

VQA Variational Quantum Algorithm.

VQC Variational Quantum Classifier.

VQE Variational Quantum Eigensolver.

Glossary

accuracy Accuracy is a performance metric commonly used to evaluate the effectiveness of a machine learning model. It represents the proportion of correctly predicted instances (samples) in the total number of instances in the dataset.

ansatz Parametrized quantum circuit used to represent a solution to a given problem. It serves as a trial wavefunction, typically employed in variational algorithms, where the objective is to find the optimal parameters that minimize a certain cost function.

barren plateau In the context of quantum computing and quantum algorithms, a barren plateau refers to a specific challenge that arises when training Quantum Neural Networks (QNNs), or a parameterized circuit, using classical optimization algorithms. It describes a situation where the gradients of the QNN's parameters become exponentially small as the number of qubits in the quantum circuit increases, leading to severe obstacles in optimizing the network.

cost In the context of machine learning and optimization algorithms, cost (also known as loss or objective function) is a measure that quantifies the discrepancy between the predicted output of a model and the actual target values in a dataset. The objective of a machine learning algorithm is to minimize this cost function during the training process to achieve accurate predictions on new, unseen data.

depth Quantum circuit depth refers to the number of sequential layers of quantum gates in a quantum circuit.

expectation value In the context of quantum mechanics, it signifies the typical or average value of a physical quantity (represented by an operator) that a quantum system is expected to have when subjected to measurement.

neural network Neural networks are computational models inspired by the structure and functioning of the human brain. They are a subset of machine learning algorithms designed to process com-



plex data and perform various tasks, including pattern recognition, classification, regression, and optimization.

short coherence time property of quantum systems in which quantum states, including superpositions and entanglements, are highly susceptible to disruption and can exist only briefly before experiencing decoherence. Coherence time quantifies the duration for which a quantum system retains its delicate quantum characteristics and executes coherent quantum operations without substantial interference from external influences.

Chapter 1

Introduction

1.1 Context

Quantum computation has garnered significant attention due to its potential to solve tasks that classical computation struggles to accomplish within a reasonable timeframe [3]. One such example is the efficient factorization of large integers, which can theoretically be achieved on quantum computers using Shor's algorithm [4].

The recent surge in interest in **Machine Learning (ML)** within both quantum and classical computing can be attributed to the vast amount of data that requires processing. With the exponential growth of data in recent years, researchers have been compelled to explore new and improved methods of data processing [5].

Pattern recognition in data stands as a common application of machine learning. **Quantum Machine Learning (QML)** has emerged as a promising approach, capable of identifying distinct patterns that classical **ML** may overlook, particularly when dealing with extensive datasets. Quantum computers process information differently, leveraging phenomena like quantum coherence and entanglement [5]. However, at present, there is insufficient evidence to suggest that quantum algorithms consistently outperform the best classical alternatives.

The current state of quantum computers is characterized by noise and **short coherence time**, making real-world applications of **ML** problems almost unfeasible on quantum processors. These quantum computers are referred to as **Noisy Intermediate-Scale Quantum (NISQ)** devices [6]. The presence of noise and coherence limitations introduces errors into the system, thus constraining the performance of certain algorithms [7]. To address these challenges, algorithms like the **Variational Quantum Algorithm (VQA)** family have been developed, which remain at the forefront of the quest for quantum advantage [1].

Variational or parametrized quantum circuits are quantum circuits dependent on free parameters.

These circuits typically yield a scalar **cost** for a given task based on the **expectation value** (with potential classical post-processing). The free parameters, denoted as θ , are tuned through a classical optimization algorithm that queries the quantum device. The iterative optimization process seeks to find better parameter candidates with each step, thereby addressing systematic errors automatically during optimization. This feature makes variational circuits particularly appealing for near-term quantum devices [8].

Algorithms employing parameterized circuits, such as the **VQA** family, utilize a **cost** function $C(\theta)$ to encode the problem and a set of parameters θ to encode the solution. The parameterized **ansatz**, akin to a **neural network**, is trained to minimize the **cost** function. In some cases, a training dataset ρ_k is used during optimization, while a testing dataset validates the model. Through an iterative loop between a quantum and a classical computer, the quantum device estimates the **cost** function or its gradient, and a classical optimizer navigates the **cost** function landscape to minimize it. Upon meeting a specific criterion, the process terminates, and the **VQA** output provides an estimation of the solution [1].

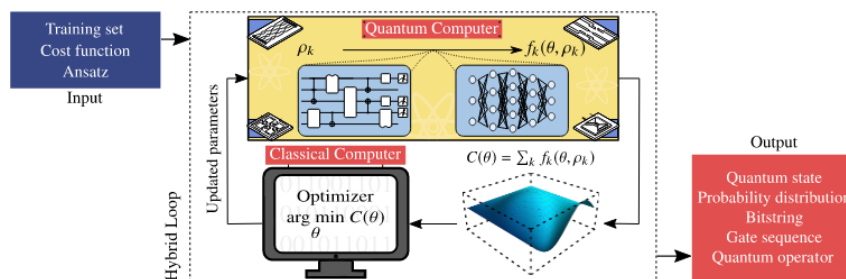


Figure 1: Schematic diagram of a **VQA**, from [1]

The hybrid computation approach, which involves utilizing both classical and quantum computation, offers a promising compromise. This technique employs quantum parameterized circuits in conjunction with classical optimizers to optimize the parameters within the circuit.

Numerous researchers have investigated these algorithms as potential candidates for achieving a quantum advantage on **NISQ** devices, finding applications in various domains such as **dynamical simulation** [9], **combinatorial optimization** [10], and, notably, **machine learning** [11].

Despite considerable efforts, **VQA**, including applications like the **Variational Quantum Classifier (VQC)**, are not without their limitations. These constraints stem from both quantum and classical considerations, and although **VQA** shows promise, it is not yet perfect. **VQC**, as the name implies, is specifically designed for the classification of datasets.

1.2 Motivation

The motivation behind the **VQC** lies in its ability to accurately predict the labels of input data points, akin to classical **neural network** where non-linearity plays a crucial role in achieving successful outcomes [1].

To identify patterns in the data, a **VQC** employs a quantum circuit comprising an encoding block $E(x)$ for converting classical data points into quantum states (when dealing with classical datasets), a parameterized circuit block $U(\theta)$ serving as the **neural network**, and a measurement block M_γ responsible for transforming quantum states into an **expectation value** that can be processed by a classical device, where γ corresponds to the type of measurement. The parameters θ are iteratively adjusted using a classical optimizer to minimize a **cost** function $\mathcal{C}(\theta)$, which encodes the classification problem. The optimization loop continues until a specific criterion is met.

Various challenges arise in **VQC** and **VQA** due to errors introduced by noise in the quantum system, which increases with the number of gates and qubits (also known as the **ansatz depth**). Optimization issues, such as falling into local minima traps and encountering **barren plateaus**, can also hinder performance [12, 7].

Improving **VQC** and **VQA** requires addressing the issue of noise in the circuit, which is influenced not only by the hardware but also by the number and type of gates. Noise can be advantageous in certain scenarios, such as when dealing with saddle points [13], but generally, its presence is undesirable. One approach to reducing noise is to adopt a shallower **ansatz**, known as a hardware-efficient ansatz, which utilizes easily implemented gates, such as rotation gates, and minimizes **depth** overhead by applying two-qubit gates to adjacent qubits that are readily connected [1, 14]. However, in some cases, increasing the **ansatz depth** becomes necessary to introduce more parameters, leading to more errors and optimization difficulties, especially when informed initial parameter guesses are not available, or problem-specific **ansatz** designs are required [1].

An increase in the number of parameters in the circuit enhances its expressibility, i.e., its ability to represent any state on the Hilbert space. While this may seem advantageous, it can impede optimization by flattening the **cost** landscape and making gradient-based optimization challenging. Overparameterization, where the number of parameters surpasses a critical threshold M_c , can help reduce the number of local minima in both quantum and classical machine learning. However, it can also complicate optimization due to increased expressibility [1, 15, 16].

To tackle the aforementioned challenges, this dissertation aims to introduce innovative approaches for crafting problem-specific **ansatz**. These approaches fall under the category of **adaptive algorithms**,

characterized by their ability to generate an **ansatz** tailored to the specific problem during the training phase. This process grants the optimizer complete control over the required **depth**, potentially eliminating the need to experiment with various **ansatz** structures for problem-solving purposes. Additionally, this approach offers a fixed form of parameter initialization, streamlining the optimization process.

1.3 Contribution

As mentioned in Section 1.2, the primary challenges leading to **VQC** failure are noise and optimization difficulties. Therefore, this dissertation aims to propose an optimization method for building problem-agnostic circuits suitable for classifying datasets. The key is to design an **ansatz** with a short **depth** yet sufficient expressibility to capture essential patterns. Generally, higher **depth** corresponds to higher expressibility, but an optimal balance must be struck to avoid issues with the classical optimizer, such as **barren plateaus** and local minima.

To achieve this objective, this dissertation intends to investigate adaptive approaches already employed in other **VQA** methodologies. These approaches provide the optimizer with control over the **ansatz** structure. This freedom allows the optimizer to adjust the **cost** function landscape to avoid **barren plateaus** and local minima traps effectively. The optimizer is tasked with optimizing not only the parameters of the operators but also the composition and/or position of the operators. This reduction in the number of parameters makes the system more robust to noise and **barren plateaus** while approximating a local minimum close to the global minimum, where the solution lies [7, 17]. Although the training process may be slower than standard **VQC** training due to optimizing the **ansatz** structure, better **accuracy** and a more generalized circuit that can be applied to different datasets are expected.

Since no existing literature covers an adaptive approach specifically for **VQCs**, this dissertation will study and adapt existing adaptive algorithms from other problem domains, particularly in the chemistry field, to classification problems. The proposed approach will be evaluated through several steps:

1. Adapting existing adaptive algorithms for classification problems;
2. Verifying the applicability of the adapted algorithms in the classification paradigm;
3. Developing an adaptive variational algorithm tailored to classification problems based on insights from existing adaptive variational algorithms;
4. Comparing the results of the proposed adaptive approach with standard **VQC** in terms of **accuracy** and **depth**.

The dissertation aims to offer fresh insights into addressing the challenges encountered in Variational Quantum Classification (**VQC**), thereby paving the way for enhanced performance in quantum classification tasks. It achieves this by demonstrating scenarios where adaptive approaches outperform the standard **VQC** algorithm. Specifically, this work demonstrates the application of two adaptive algorithms, the Rotoselect algorithm and the ADAPT algorithm, to classify three different datasets of varying dimensions. These datasets include two fictitious datasets and one real dataset, the 'Iris' dataset. Through this exploration, it becomes evident that these approaches can yield improved results with relatively short-depth ansatzes.

The Rotoselect algorithm achieves this by dynamically modifying the entanglement in the ansatz, accomplished through adjustments in the axes and rotation angles. In contrast, the ADAPT algorithm capitalizes on the sequential inclusion of operators, underscoring the advantages of this approach. Both adaptive approaches surpassed the standard **VQC** in certain scenarios, particularly when operating with a reduced number of parameters. The reduction in parameter count remains a primary objective for these adaptive algorithms. While these results may not constitute groundbreaking advancements in the field of **QML**, they shed light on the potential for creating adaptive solutions in Variational Quantum Classification, a domain within quantum computing that has thus far lacked substantial progress.

1.4 Summary

To summarize, **VQA**'s algorithms are good candidates for achieving some type of quantum advantage on noisy devices due to their ability to mitigate systematic noise and the fact that they can be used to solve a variety of problems, making it an interesting type of algorithm to study.

VQC is a specific algorithm in the **VQA** family that suffers from the same issues as the other algorithms in the **VQA** family due to their similar structure. As evidenced by several studies that have showcased enhancements over the standard version, it is strongly suggested that an adaptive variant of the **VQC** could potentially offer advantages, particularly with regard to accuracy, which stands as one of the primary metrics for evaluating the performance of a classification model. By allowing the optimizer to dynamically adjust the ansatz structure during the training process, the adaptive approach can achieve an optimal balance between circuit depth and expressibility, which in turn may lead to improved accuracy in classification tasks. As a result, the adaptive **VQC** holds promise as a potential solution to address the common challenges faced by **VQC** algorithms, thereby advancing the state-of-the-art in quantum classification and opening new avenues for quantum advantage on noisy devices.

Chapter 2

Variational Quantum Algorithms

2.1 Context

Quantum Machine Learning (QML) is an emerging interdisciplinary research field that combines principles from quantum physics and **Machine Learning (ML)**. While quantum computers hold the promise of solving complex problems beyond the capabilities of classical computers, the current **Noisy Intermediate-Scale Quantum (NISQ)** devices have limited capabilities for **ML** applications. To harness the potential of quantum computing in real-world applications, it is essential to address the complexity of quantum algorithms [11].

Efforts have been made to reduce the complexity of quantum algorithms, including employing hybrid approaches that combine quantum and classical neural networks for classification tasks and using pre-processing techniques like **Principal Component Analysis (PCA)** to reduce dataset size and facilitate quantum hardware utilization, leading to enhanced quantum system performance [11]. These approaches commonly aim to shift the computational workload from quantum processors to classical processors.

The adaptive approach, a problem-specific algorithm, serves as a focal point for this dissertation. It aims to reduce the burden on quantum processors by reducing the number of gates used while concurrently increasing the computational tasks performed on classical processors through optimization of operators' composition and position. Recent research in the chemistry field has explored this adaptive approach, particularly with the application of **Variational Quantum Eigensolver (VQE)** [7, 18, 19] and **Quantum Approximate Optimization Algorithm (QAOA)** [20, 21, 17], both members of the **Variational Quantum Algorithm (VQA)** family.

2.2 Standard VQA's

This chapter delves into the fundamental aspects of **Variational Quantum Algorithm (VQA)**, exploring its key algorithms and their applications. It begins with an examination of the **Variational Quantum Eigensolver (VQE)**, a prominent **VQA** used to find the minimum eigenvalue of a given Hermitian operator. **VQE** plays a crucial role in quantum chemistry, where it is applied to estimate the energy of a molecule's ground state, a vital parameter in understanding chemical systems and reactions. Through a parameterized ansatz and classical optimization, **VQE** empowers researchers to explore complex quantum systems and obtain valuable insights into their behavior.

In the subsequent sections, we also explore the **Quantum Approximate Optimization Algorithm (QAOA)** and the **Variational Quantum Classifier (VQC)**, both important members of the **VQA** family. **QAOA** excels in solving **Quadratic Unconstrained Binary Optimization (QUBO)** problems, offering significant potential for tackling optimization challenges across various domains. On the other hand, **VQC** provides a powerful tool for classification tasks, enabling quantum-based machine learning on quantum and classical datasets alike.

By comprehensively understanding the core concepts and applications of these standard **VQA** algorithms, this chapter lays the foundation for exploring advanced techniques, including the adaptive approach that aims to address the challenges of noise and optimization difficulties on quantum processors. Through this exploration, the objective is to unlock the full potential of **VQC** and advance the state of quantum computing for solving real-world problems.

2.2.1 VQE

Typically utilized to determine the minimum eigenvalue of an operator, the **Variational Quantum Eigensolver (VQE)** is employed in fields like chemistry to estimate a molecule's ground state energy (E_{gs}), which corresponds to the state with the lowest energy.

The **VQE** process commences with an initial state, denoted as $|\psi_i\rangle$ (commonly $|0\rangle^{\otimes n}$ or the Hartree Fock reference state). Subsequently, a variational ansatz with a predefined structure, $U(\theta)$, is applied to generate the state $|\psi(\theta)\rangle = U(\theta)|\psi_i\rangle$. The associated cost function, $\mathcal{C}(\theta)$, is defined as the **expectation value** of the Hermitian operator, i.e., $\mathcal{C}(\theta) = \langle\psi(\theta)|H|\psi(\theta)\rangle$, which corresponds to the energy. To find the ground state energy, the objective is to minimize the cost function, expressed as

$$E_{gs} = \underset{\theta}{\operatorname{argmin}} \mathcal{C}(\theta) \quad [22].$$

The minimization process entails leveraging a quantum computer to calculate the **expectation value**

$\langle \psi(\theta) | H | \psi(\theta) \rangle$, while simultaneously adjusting the parameter θ with the aid of a classical computer to minimize $\mathcal{C}(\theta)$. This iterative process seeks to converge to the ground state energy, providing valuable insights into molecular systems and their energetics.

2.2.2 QAOA

Commonly employed to tackle **Quadratic Unconstrained Binary Optimization (QUBO)** problems, such as Max Cut in graph theory or identifying the minimum energy state [23], the **Quantum Approximate Optimization Algorithm (QAOA)** operates differently compared to fixed parameterized ansatz approaches.

In **QAOA**, the problem under consideration is encoded into a Hamiltonian, serving as a unitary operator with a parameter denoted as $\exp(-i\beta H_P)$ —referred to as a problem unitary—where β represents the parameter. Additionally, **QAOA** employs an additional unitary transformation, $\exp(-i\gamma U_M)$, with $U_M = X \otimes X \otimes \dots \otimes X$, where γ corresponds to the mixer unitary's parameter. This mixer unitary typically consists of rotations along the x-axis applied to all qubits with the same parameter. The **QAOA** algorithm alternates between these unitaries a variable number of times, contingent on the specific problem at hand, each time using distinct parameters $\theta = (\beta, \gamma)$ [10, 1].

The associated cost function, denoted as $\mathcal{C}(\theta)$, is defined as the **expectation value** of the problem Hamiltonian, i.e., $\mathcal{C}(\theta) = \langle \psi(\beta, \gamma) | H_P | \psi(\beta, \gamma) \rangle$. Ideally, **an infinite number of layers would yield an exact solution**, but practical constraints make this infeasible. Consequently, the number of layers is determined to achieve a sufficiently precise solution, a characteristic that distinguishes **QAOA** from most **VQAs**.



2.2.3 VQC

A **Variational Quantum Classifier (VQC)** comprises four fundamental components: **an encoding block**, **a parameterized circuit block (ansatz)**, **a measurement block**, and **classical post-processing**. Notably, **VQC** and **VQE** are closely aligned, often employing the same ansatz structure.

The primary distinction between **VQE** and **VQC** arises when dealing with classical datasets. **In the case of VQC**, **an encoding block is integrated into the architecture**. This encoding block serves the purpose of translating each data point into a quantum state compatible with quantum processing. Additionally, classification tasks frequently involve post-processing steps, differentiating the cost function employed in classification problems from those in other **VQAs**. **Consequently, the cost function for VQC is defined as**



$\mathcal{C}(\theta) = f(\langle \psi(x, \theta) | \hat{O} | \psi(x, \theta) \rangle)$, where f represents a post-processing function, x denotes the data

points, and θ signifies the optimization parameters [1].

Recent research has explored various approaches for designing effective **VQC** structures. However, the choice of the most suitable approach depends heavily on the specific context and requirements of the problem at hand [1].

2.3 Adaptive VQA's

The **VQA** detailed in Section 2.2 have demonstrated impressive problem-solving capabilities across various domains. However, they encounter challenges in achieving the requisite expressiveness for handling complex problems. These algorithms often demand substantial circuit **depth**, which can lead to optimization challenges such as getting stuck in local minima or encountering **barren plateaus**.

In response to these challenges, researchers have explored alternatives to the fixed **ansatz** structure employed in standard **VQA** algorithms. This exploration has given rise to adaptive variational algorithms, with **ADAPT-VQE** [7, 18, 19] being a notable example, primarily applied within the realm of quantum chemistry.

Adaptive variational algorithms possess a unique capability to dynamically adjust the **ansatz** structure based on the specific objective function and available data. This adaptive approach empowers the optimizer not only to optimize the parameters θ but also to fine-tune the rotation gates influenced by these parameters. This level of adaptability provides the optimizer with enhanced control over the optimization process within predefined rules.

Although considerable progress has been made in applying adaptive strategies to **VQE** and **QAOA** algorithms, there remains a relatively unexplored domain: the application of an adaptive approach to the **Variational Quantum Classifier (VQC)** algorithm. This dissertation sets out to investigate the feasibility and effectiveness of such an adaptive approach within the context of the **VQC** algorithm.

2.3.1 Rotoselect Algorithm

The Rotoselect algorithm is an adaptive variational algorithm originally introduced in the field of quantum chemistry. It was initially applied to estimate the ground state of the Lithium Hydride molecule [24]. This innovative algorithm demonstrates the power of adaptivity in variational quantum approaches.

In the Rotoselect algorithm, the optimizer gains the capability to optimize not only the parameters θ but also the axis for the rotations influenced by these parameters. The approach keeps the unparameterized two-qubit gates fixed and only permits variations in the single-qubit gates (rotation gates) along with their

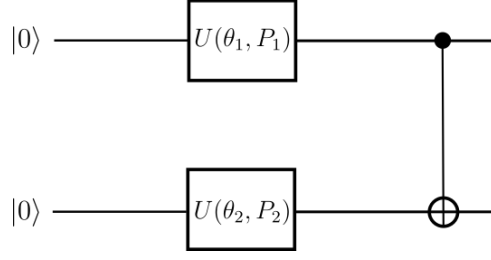


Figure 2: An illustrative example of the Rotoselect algorithm, sourced from [2]

corresponding parameters, all based on an initial **ansatz**. The optimizer then explores different gates from a set of generators denoted as $G \in \sigma_z, \sigma_x, \sigma_y$, where σ_i represents a Pauli matrix. It seeks the optimal combination of these gates that yields the minimum energy for the optimized parameters.

The approach of Rotoselect entails updating the parameters $\theta = \theta_1, \dots, \theta_M$ and gate choices $G = G_1, \dots, G_M$ one by one, employing a closed-form expression to calculate the optimal value of the d^{th} parameter θ_d^* . This optimization is performed while maintaining the other parameters and gate choices fixed. For a given Hamiltonian of interest H , with an objective function represented as the expectation value $\langle H \rangle$, the optimal value of each individual parameter is determined using a closed-form expression, as presented in Equation 2.2

$$\theta_d^* = \arg \max_{\theta_d} \langle H \rangle_{\theta_d} \quad (2.1)$$

$$= -\frac{\pi}{2} - \arctan \left(\frac{2 \langle H \rangle_{\theta_d=0} - \langle H \rangle_{\theta_d=\frac{\pi}{2}} - \langle H \rangle_{\theta_d=-\frac{\pi}{2}}}{\langle H \rangle_{\theta_d=\frac{\pi}{2}} - \langle H \rangle_{\theta_d=-\frac{\pi}{2}}} \right) \quad (2.2)$$

Upon updating all the angles for $d = 1, \dots, M$ a cycle, the algorithm can proceed to initiate a new cycle, unless a predetermined stopping criterion is met [24].

The equation 2.2 is derived from the fact that the **expectation value** of an operator (in this case of the Hamiltonian operator, H) can be expressed as a sine function of each individual parameter:

$$\langle H \rangle_{\theta_d} = A \sin(\theta_d + B) + C \quad (2.3)$$

Where A , B , and C are constants that depend on the values of Equation 2.3 for $\theta_d = 0, \pi, \pi/2$, and $-\pi/2$ presented originally in the paper by [24].

$$A = \frac{1}{2} \sqrt{(\langle H \rangle_0 - \langle H \rangle_\pi)^2 + (\langle H \rangle_{\frac{\pi}{2}} - \langle H \rangle_{-\frac{\pi}{2}})^2} \quad (2.4)$$

$$B = \arctan \left(\frac{\langle H \rangle_0 - \langle H \rangle_\pi}{\langle H \rangle_{\frac{\pi}{2}} - \langle H \rangle_{-\frac{\pi}{2}}} \right) \quad (2.5)$$

$$C = \frac{1}{2} (\langle H \rangle_0 + \langle H \rangle_\pi) \quad (2.6)$$

While the Rotoselect algorithm has shown significant improvements over the standard **VQE**, further enhancements are still possible. For instance, eliminating redundant gates and providing the optimizer control over the placement of two-qubit gates could yield better results. However, these improvements require prior knowledge of the problem, making it challenging to devise a problem-agnostic algorithm for classification tasks, where such prior knowledge is often unavailable without a thorough study of the problem at hand.

The subsequent sections will delve into the exploration and development of an adaptive approach for the **VQC** algorithm. The goal is to unlock its full potential and address the challenges posed by noise on quantum processors and optimization difficulties on classical processors. Through an investigation of the Rotoselect algorithm and its potential adaptations to **VQC**, this research aims to pave the way for more effective and versatile quantum-based **ML** solutions.

2.3.2 ADAPT

The **Adaptive Derivative Assembled Problem Tailored (ADAPT)** algorithm is a versatile approach that can be employed with either **QAOA** or **VQE**, known as ADAPT-QAOA [17, 20] and ADAPT-VQE [7, 18], respectively.

In contrast to the fixed structure of the Rotoselect algorithm, the **ADAPT** algorithm introduces a dynamic approach to gradually expand the **ansatz**. This expansion involves the incremental inclusion of operators from a predefined pool, allowing for precise control over the **ansatz's depth** during the optimization process. One common method for creating this operator pool is by generating Pauli strings of various sizes. These strings are then subjected to exponential operations, resulting in unitary operators of the form $\exp(-i\theta_d S_d)$. Here, θ_d represents the parameter associated with a particular Pauli string, denoted as S_d . Each of these Pauli strings is a tensor product of n operators selected from the generator set, which includes X , Y , Z , and I , for n qubits. It's worth noting that the size of the operator pool grows exponentially with the number of qubits. However, the search for the operator with the most significant gradient involves a brute-force approach, leading to a substantial increase in execution time and the con-

sumption of extensive classical and quantum resources. Consequently, this brute-force methodology is impractical for large systems.



The optimizer calculates the gradients of all unitary operators within the set with respect to θ_d . It then adds the operator with the largest gradient in magnitude to the ansatz. This addition of a new operator and parameter increases the dimension of the cost function in a direction that results in the most significant reduction in cost. This process is repeated iteratively until certain criteria are met, effectively constructing the adaptive **ansatz**.

An advantage of **ADAPT** over Rotoselect is the freedom to define the pool of operators, provided they are Hermitian operators. This flexibility empowers the optimizer to choose from a range of operators, including those with varying entanglement entropy. By incorporating Pauli strings of different sizes in the pool, the optimizer can select between operators with low entanglement (shorter strings) and those with higher entanglement (longer strings) [17].

ADAPT eliminates the need for prior knowledge of the problem's necessary **depth**, as the optimizer determines the appropriate **depth** during training. However, a maximum **depth** can be predefined. The algorithm aims to mitigate **barren plateaus** by reducing the number of parameters and attempts to bypass local traps by reshaping the cost function landscape. This process guides the optimization toward approximating a global minimum [7].

While **ADAPT** strives to maintain a minimum **depth**, there is no guarantee that all gates are essential for the cost function. Redundant gates can still exist, which may impact the algorithm's performance. Moreover, **ADAPT**'s iterative nature may result in longer computational times, particularly when dealing with large operator pools [7].

2.3.3 TETRIS-ADAPT

From the perspective of the authors, [19], the TETRIS-ADAPT algorithm represents an advancement over the **ADAPT** algorithm by further reducing the **ansatz depth** without requiring additional quantum resources.

The strategy employed in the TETRIS-ADAPT algorithm is akin to the **ADAPT** algorithm described in Sec.2.3.2 but with a crucial difference in how operators are added. Instead of introducing one operator at a time, TETRIS-ADAPT incorporates multiple operators and their parameters simultaneously in each iteration.

The algorithm starts by adding the operator with the highest gradient and then continues to include the operator with the next highest gradient, ensuring that the qubits affected by the new operator differ

from those influenced by the previously added operator. This process is repeated iteratively, introducing operators with smaller or equal gradients, until all available qubits are utilized or the operator pool is depleted.

As justified in [19], TETRIS-ADAPT achieves circuits shallower than the ADAPT algorithm while maintaining a similar number of CNOTs. By offering the optimizer greater freedom to explore the Hilbert space within each iteration, this algorithm enhances the expressibility of ADAPT-VQE without increasing the **ansatz depth**.

TETRIS-ADAPT yields results comparable to the ADAPT algorithm but with significantly reduced **depth**. Its improved resilience to noise and errors holds promise for real-world applications on quantum hardware, making it a potential candidate for practical implementation.

2.3.4 Vans

The Vans (Variable Ansatz) algorithm was designed to address trainability and noise-related challenges by maintaining a shallow **ansatz** [12]. This algorithm can dynamically add and remove gates based on a set of predefined rules.

Similar to previous algorithms, the Vans algorithm requires the definition of a pool of operators. However, it offers more flexibility in the choice of operators. Any operator can be added to the pool, as long as the parameterized operators can be compiled to the identity operator. In other words, a parameterized operator $V(\lambda) \in \mathcal{D}$ exists only if there is a set of parameters λ such that $V(\lambda) = I$. The authors [12] suggest allowing some deviation to increase the number of regions that can be explored, which would otherwise require numerous iterations.

The process of adding gates in the Vans algorithm starts by randomly selecting one operator from the set of predefined operators, with each operator having an equal chance of being chosen. Next, the algorithm determines which qubits the chosen operator will be applied to. This qubit selection process is stochastic, but it is weighted for two or more qubit operators. Qubits with low or no entropy are given a higher probability of being selected, which effectively increases the overall entropy of the system without introducing excessive error.

Once the operator and the qubits are chosen, the operator is added to the **ansatz** and initialized to the identity, or a state close to it if needed [12].

2.3.4.1 Simplification of the Circuit

When a new operator is added, the Vans algorithm performs a circuit simplification step to remove redundant and unnecessary gates, ensuring a concise and efficient **ansatz**. The simplification process involves the following rules:

1. **Removal of Initial CNOT gates:** CNOT gates acting at the beginning of the circuit are removed;
2. **Removal of Initial Rotations:** Rotations about the z-axis that act at the beginning of the circuit are removed;
3. **Elimination of Consecutive CNOTs:** Consecutive CNOT gates that share the same control and target qubits are eliminated. This rule helps reduce redundant CNOT operations in the circuit;
4. **Combination of Consecutive Rotations:** Two or more consecutive rotations around the same axis and acting on the same qubit are combined to form a single rotation. This rule aims to consolidate multiple rotations into more concise representations;
5. **Consolidation of Sequential Single-Qubit Rotations:** If three or more single-qubit rotations are sequentially acting on the same qubit, they are combined into a general single-qubit rotation of the type $R_z(\theta_1)R_x(\theta_2)R_z(\theta_3)$ or $R_x(\theta_1)R_z(\theta_2)R_x(\theta_3)$, which has the same action as the preceding rotations;
6. **Removal of Non-Influential Gates:** Gates that do not significantly affect the cost function are removed. This is determined by comparing the cost values obtained from the **ansatz** with and without the gate. If there is no significant impact or if the **cost** function increases with the addition of the gate, it is removed from the circuit.

However, in a classification problem, where the input state to the **ansatz** is not the all-zero state $|0\rangle^{\otimes n}$, rules (1) and (2) can be omitted since the initial gates may have a significant effect on the circuit's behavior.

2.3.4.2 Parameter Optimization

After the simplification stage, the Vans algorithm proceeds to optimize the parameters of the **ansatz**. The optimization process is followed by another round of circuit simplification. The cycle of optimization and simplification continues until the circuit reaches a point where no further simplification is possible.

During the parameter optimization step, the algorithm aims to find the values of the parameters that minimize the **cost** function. The parameters are updated iteratively using optimization techniques such as gradient descent or other classical optimization algorithms.

After each optimization iteration, the algorithm evaluates the new **cost** value obtained from the updated circuit. The system accepts the alterations with a probability based on the change in the **cost** value: the bigger the **cost** difference (as long as the new **cost** is lower than the previous one), the higher the acceptance probability. This probabilistic acceptance mechanism allows the algorithm to explore the parameter space effectively and avoid getting stuck in local minima.

By iteratively optimizing the parameters and simplifying the circuit, the Vans algorithm achieves a balance between expressibility and **depth**, leading to more efficient and effective quantum circuits for classification tasks [12].

2.3.4.3 Vans characteristics

The Vans algorithm offers the user the highest degree of flexibility in building the operator pool among all the algorithms mentioned in this document. It not only adds operators but also removes redundant gates, resulting in a reduction of the circuit's **depth** and improved noise resistance.

As a stochastic algorithm, there is no guarantee of finding the best solution for the problem, and there is a possibility of failure to find an appropriate solution, although the probability of such occurrences is low.

While the authors [19] do not specify the exact number of resources used, given the tasks performed by this algorithm, it can be expected to be computationally demanding and relatively slow, requiring a substantial amount of resources. Nevertheless, the trade-off for improved flexibility and noise resistance may justify the computational cost in certain applications.

2.3.5 Summary

The adaptive approaches discussed here are innovative variations of **VQAs** that aim to address the challenges posed by noise on quantum processors and optimization difficulties on classical processors. These adaptive algorithms leverage the flexibility of quantum circuits to dynamically adjust their structure and parameters, making them more efficient and versatile in tackling complex problems.

1. Rotoselect Algorithm:

The Rotoselect algorithm brings adaptivity into play by facilitating the direct optimization of individ-

ual parameters, circumventing the traditional reliance on gradient descent methods. This approach involves the exploration of different rotation gates, with parameters and gate choices being updated sequentially. By optimizing parameters one at a time, this method streamlines the circuit optimization process, making parameter optimization faster and more straightforward. Its successful application in quantum chemistry underlines its effectiveness, particularly in the context of ground-state estimation.

2. **ADAPT Algorithm:**

The **Adaptive Derivative Assembled Problem Tailored (ADAPT)** algorithm is a remarkable and adaptive variational approach compatible with **QAOA** and **VQE**. It stands out for its dynamic growth of the **ansatz**, providing the optimizer with fine-grained control over the circuit's **depth**. The algorithm's architecture relies on a set of Pauli strings, which are exponentiated to create corresponding unitary operators. By calculating gradients with respect to the parameters, the algorithm selects the operator with the most significant gradient (in absolute value) and incorporates it into the **ansatz**, initializing parameter optimization. This iterative process continues until specific termination criteria are met, such as reaching a maximum **depth** or achieving a desired convergence threshold.

The adaptivity of the **Adaptive Derivative Assembled Problem Tailored (ADAPT)** algorithm facilitates efficient exploration of the Hilbert space and aims to circumvent **barren plateaus** and local traps that often impede traditional **Variational Quantum Algorithm (VQA)** methods. Consequently, the **ADAPT** algorithm demonstrates enhanced trainability and noise resilience, surpassing conventional approaches. Its inherent adaptivity positions it as a compelling choice for versatile and widely applicable **Quantum Machine Learning (QML)** tasks. The **ADAPT** algorithm holds significant promise in empowering quantum algorithms to address a multitude of real-world challenges spanning diverse fields, including quantum chemistry simulations, optimization, and **Machine Learning (ML)** tasks.

3. **TETRIS-ADAPT Algorithm:**

The TETRIS-ADAPT algorithm draws inspiration from **ADAPT** and introduces an even more efficient approach to constructing shallower circuits. Unlike **ADAPT**, TETRIS-ADAPT adds multiple gates and parameters simultaneously, effectively increasing the number of parameters without significantly increasing the circuit **depth**. Remarkably, this improved approach maintains the same level of performance and yields identical solutions as the **ADAPT** algorithm.

The key advantage of TETRIS-ADAPT lies in its ability to substantially reduce circuit **depth** while preserving solution quality. This remarkable characteristic holds significant promise for real-world quantum applications, where noise and limited quantum resources pose challenges. By efficiently exploring the parameter space and optimizing circuit structures, TETRIS-ADAPT empowers quantum algorithms to achieve enhanced expressibility without compromising on the quality of results.

With TETRIS-ADAPT's innovative methodology, quantum computing applications, ranging from quantum chemistry simulations to optimization tasks and **ML**, stand to benefit from improved efficiency and effectiveness. As quantum technologies continue to evolve, adaptive algorithms like TETRIS-ADAPT provide a compelling avenue for unlocking the full potential of quantum computing in tackling complex real-world problems.

4. **Vans Algorithm:** The Vans algorithm stands out as a versatile and adaptive approach, offering users a wide range of options to construct the operator pool. This algorithm possesses the unique ability to add and remove gates, optimizing both parameters and circuit structure. By eliminating redundant gates, it effectively reduces the circuit **depth** and enhances noise resilience. Although its stochastic nature may not always guarantee the optimal solution, its adaptability makes it well-suited for a diverse array of problems. The Vans algorithm's flexibility and efficiency make it a promising candidate for various **QML** tasks, providing a valuable tool for tackling real-world challenges in quantum computing applications.

In conclusion, these adaptive approaches demonstrate the potential of adaptivity in **VQAs** to achieve quantum advantage on noisy devices and outperform standard **VQA** algorithms. Their dynamic adjustments of circuit structures and parameter optimization enable improved performance in various **QML** applications, opening avenues for further research and advancements in quantum computing.

Table 1: Adaptive Algorithms simple overview

Algorithms	Vans	RotoSelect	ADAPT	TETRIS-ADAPT
Pool	$\forall V(\lambda) \in D \exists \lambda^*: V(\lambda^*) = \mathbb{1}$	$\sigma_x, \sigma_y, \sigma_z$	Set of Pauli Strings	Set of Pauli Strings
Operator Choice	Stochastically	Minimal value of the cost function	Largest gradient in magnitude	Largest gradient in magnitude
Depth	Variable	Fixed	Variable	Variable
Entanglement flexibility	Flexible	Fixed	Flexible	Flexible

Chapter 3

Problems, Challenges, and Proposed Solutions

As discussed in Chapter 1, **Variational Quantum Algorithm (VQA)** face significant challenges, particularly concerning noise and difficulties in classical optimization. Researchers have focused on addressing these issues, with specific emphasis on algorithms like **Variational Quantum Eigensolver (VQE)** and **Quantum Approximate Optimization Algorithm (QAOA)** [7, 18, 25, 24, 26, 20, 21, 17]. However, the application of adaptive algorithms to classification problems remains relatively limited, with only the Vans algorithm [12], discussed in Section 2.3.4, giving them attention.

While the Vans algorithm provides adaptive capabilities for classification, its "greedy" nature demands significant computational resources for practical-sized problems.

The challenges faced by the **VQC** algorithm present an opportunity to explore the potential of adaptive algorithms such as Rotoselect and **ADAPT**. These algorithms hold promise in enhancing **VQA**'s robustness, efficiency, and noise resistance. This dissertation aims to investigate the adaptation of Rotoselect and **ADAPT** algorithms to the **VQC** framework, paving the way for more powerful and practical quantum-based machine learning solutions for real-world challenges.

The dissertation primarily addresses challenges posed by classical optimizers and quantum limitations in **VQC**. The objective is to achieve improved performance and robustness by reducing the **ansatz depth** through classical computation, known for its reliability compared to quantum computation.

A key focus is on developing problem-tailored **ansatz** structures to handle challenges such as **barren plateaus**, overfitting, and underfitting during problem trainability. A fixed parameter initialization approach will be devised for universal application to diverse problems while efficiently managing algorithm complexity.

Additionally, the research emphasizes evaluating the efficiency of adaptive algorithms with respect to circuit depth and solution accuracy. This involves exploring additional benchmarking and evaluation metrics to emphasize circuit **depth** reduction and solution accuracy enhancement in adaptive algorithms for **VQC**.

It is important to note that it was considered a set of fictitious and real datasets composed of classical data. Consequently, encoding blocks are used to convert classical data into quantum states. Choosing an appropriate encoding is crucial for achieving desirable results. Two encoding strategies were utilized, although the in-depth study of encoding techniques lies outside the scope of this dissertation.

Moreover, it is imperative to conduct a comprehensive resource analysis for adaptive algorithms. While the Rotoselect algorithm offers valuable insights into resource utilization [24], a more thorough examination is essential to evaluate the practicality and scalability of these adaptive approaches. Which is not done in this dissertation due to time constraints.

By combining experimental analysis and rigorous evaluation, this dissertation aims to introduce adaptive approaches in the **VQC** domain and bring quantum machine learning closer to practical implementation in various domains. The goal is to contribute to the development of quantum-based machine learning solutions that effectively address real-world challenges and applications.

3.1 Proposed Solutions

This dissertation is dedicated to the adaptation of two algorithms, namely, the Rotoselect algorithm (Section 2.3.1) and the ADAPT algorithm (Section 2.3.2). The selection of these specific algorithms is driven by distinct factors:

The Rotoselect algorithm is appealing due to its simplicity and intuitive design. However, it has a significant limitation: it relies directly on the expected value of a Hamiltonian for its optimization. In the context of solving the classification problems addressed in this dissertation, this reliance can potentially introduce a bottleneck to the core problem, mainly due to the need for post-processing associated with the expected value.



Conversely, the ADAPT algorithm stands out due to its versatility. It is applicable in both Variational Quantum Eigensolver (VQE) and Quantum Approximate Optimization Algorithm (QAOA) scenarios. This versatility positions it as a promising candidate for adaptation to classification problems.

These algorithms have been chosen for adaptation based on their unique attributes, and this dissertation explores their potential to address classification challenges effectively.

Among the studied algorithms, the Vans algorithm, despite its complexity and versatility, is not considered appropriate for this master's thesis. Its computational demands and extensive optimization steps make it challenging to implement within the constraints of this study. On the other hand, while the TETRIS-ADAPT algorithm showcases improvement over the ADAPT algorithm, it demonstrates enhanced perfor-

mance with a significant number of qubits. In the present dissertation, the emphasis is on maintaining a reasonable number of qubits to ensure manageable optimization times and computational resources.

The selected algorithms, Rotoselect and ADAPT, will be thoroughly investigated in the context of classification problems. The objective is to explore their adaptation potential, overcome challenges related to trainability, and enhance their performance and applicability in quantum machine learning applications. By analyzing and adapting these algorithms, this research seeks to contribute to the development of more robust and efficient quantum-based machine learning solutions.

3.1.1 Rotoselect

As elucidated in Section 2.3.1, the Rotoselect algorithm’s optimizer systematically modifies the rotation angle and axis individually, while maintaining the other gates and angles fixed within the quantum circuit.

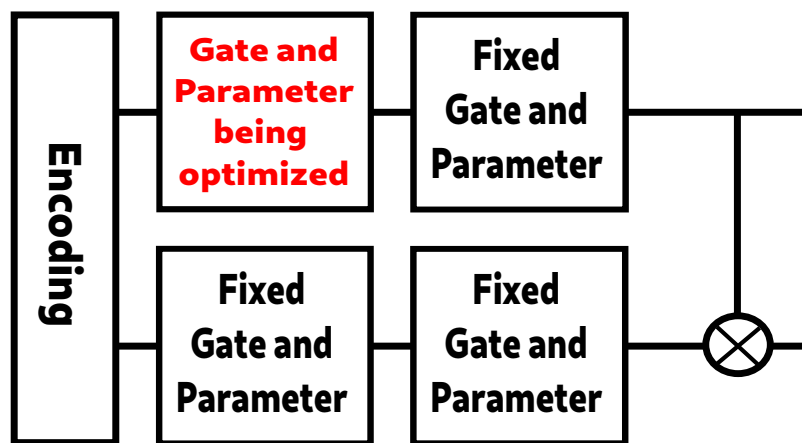


Figure 3: Schematic representation of the Rotoselect algorithm adapted for a classification problem, depicting the two qubits engaged in the process.

Figure 3 portrays the adapted schematic of the Rotoselect algorithm tailored for a classification problem involving two qubits.

In this illustration, the gate denoted by the red label symbolizes the gate undergoing optimization, while the remaining gates remain unaltered. The circuit is instantiated three times, with each instance sharing a common structure except for the gate bearing red labels, which vary in angular orientation. Specifically, one instance encompasses a rotation around the xx axis, another revolves around the yy axis, and the final instance gyrates around the zz axis.

While the original Rotoselect algorithm relies on Equation 2.2 to ascertain the optimal value for the

gate in each of the three instances, it is essential to acknowledge that this equation no longer holds validity in the context of a classification scenario.

Adapting the Rotoselect algorithm for classification tasks introduces fresh challenges and mandates alternative strategies for determining the optimal gate values.

Moreover, it is noteworthy that the expected value of an operator can be represented as a sinusoidal function (Eq. 2.3) [24], assuming that all parameters except one remain constant. This facilitates the deduction of a **cost** function from the acquired outcome. Viewing the quantum circuit as a fusion of an encode block, an ansatz block, and a measurement block, each individual data point within the dataset can be considered as a fixed-parameter vector. Consequently, the output of each circuit can be elegantly described as a sinusoidal function that correlates with a corresponding classification label. For a labeled dataset D composed of M data points, where each data point x_m is associated with a label $l_m \in \{-1, 1\}$, it is possible to utilize the following cost function in Equation 3.1 for the binary classification task.


$$\mathcal{C}(\theta) = \sum_m^M (A_m \sin(\theta + B_m) + C_m - l_m)^2 \quad (3.1)$$

Where M represents the total number of data points, and the constants A_m , B_m , and C_m are determined as described in [24]:

$$A_m = \frac{1}{2} \sqrt{(\langle M \rangle_0^m - \langle M \rangle_\pi^m)^2 + \left(\langle M \rangle_{\frac{\pi}{2}}^m - \langle M \rangle_{-\frac{\pi}{2}}^m \right)^2} \quad (3.2)$$

$$B_m = \arctan \left(\frac{\langle M \rangle_0^m - \langle M \rangle_\pi^m}{\langle M \rangle_{\frac{\pi}{2}}^m - \langle M \rangle_{-\frac{\pi}{2}}^m} \right) \quad (3.3)$$

$$C_m = \frac{1}{2} (\langle M \rangle_0^m + \langle M \rangle_\pi^m) \quad (3.4)$$

Within these equations, $\langle M \rangle_\theta^m$ signifies the execution of the quantum circuit and retrieval of its **expectation value** for the data point m , employing the independent parameter θ . This implies that, for each optimization step, the quantum circuit is employed a total of $4 \times 3 \times M$ times. 

The constants A_m , B_m , and C_m are pivotal within the context of the Rotoselect algorithm for a classification problem, as they enable the computation of the **cost** function and the determination of an optimal gate value to enhance classification accuracy. These constants encapsulate essential information to guide the optimization process and tailor the **ansatz** structure to the precise classification task at hand.

In the classification paradigm, determining the optimal value for θ is no longer as straightforward as in the original implementation of the Rotoselect algorithm. Nevertheless, Equation 3.1 can guide a classical



optimizer in identifying the optimal value of θ that minimizes the **cost** function. Although this approach deviates from the gradient-free nature of the original algorithm, it remains effective in training the model for binary classification problems. Specifically, by assigning labels of 1 or -1 to each data point and utilizing the **expectation value** of the Pauli string $Z \otimes Z \otimes \dots \otimes Z$, which ranges from 1 to -1 , the classification model can be successfully trained.

Importantly, this adaptation introduces novel challenges and avenues for the development of efficient and resilient quantum-powered machine learning methodologies. It is imperative to underscore that this approach exclusively applies to binary classification tasks. The considerations of multi-label classification necessitate alternative approaches, although such deliberations lie beyond the scope of this dissertation, which remains exclusively focused on binary classification problems.

The algorithm employed for the adapted Rotoselect procedure is outlined below:

Algorithm 1: Rotoselect Algorithm

Input: Initial rotations axes G , Cost function $\mathcal{C}(\theta)$, Dataset \mathcal{D}^M

Output: Optimal rotation angles θ^* , Set of rotation axis G^*

Initialize rotation angles $\theta_d = 0$ for $d = 1, \dots, D$;

repeat

for $d = 1, \dots, D$ **do**

 Fix all gates and parameters except θ_d and G_d ;

 Initialize $\mathcal{C}_{\text{array}}$ as an empty array;

 Initialize G_{array} as an empty array;

 Initialize θ_{array} as an empty array;

foreach $P \in \{\sigma_x, \sigma_y, \sigma_z\}$ **do**

$G_d \leftarrow P$;

foreach $D_m \in \mathcal{D}^M$ **do**

 Compute $\langle M \rangle_0^m$, $\langle M \rangle_\pi^m$, $\langle M \rangle_{\frac{\pi}{2}}^m$ and $\langle M \rangle_{-\frac{\pi}{2}}^m$ to find the values for A_m , B_m
 and C_m ;

end

 Optimize θ_d by minimizing Eq. 3.1 using a classical optimizer;

 Calculate cost function value $\mathcal{C}(\theta)$ (using the values A_m , B_m and C_m);

 Append $\mathcal{C}(\theta)$ to $\mathcal{C}_{\text{array}}$;

 Append G_d to G_{array} ;

 Append θ_d to θ_{array} ;

end

 Find index i of the smallest cost value in $\mathcal{C}_{\text{array}}$;

 Update θ^* with $\theta_{\text{array}}[i]$;

 Update G^* with $G_{\text{array}}[i]$;

end

until Reaching Iteration Limit;

return θ^* , G^* ;

3.1.2 ADAPT

As introduced earlier (Section 2.3.2), the **ADAPT** algorithm is a versatile approach for constructing quantum **ansatz** circuits from scratch or enhancing the **depth** of existing ones. Similar to the rotoselect

algorithm, **ADAPT** possesses the unique ability to adapt the structure of a quantum circuit's **ansatz**, incorporating gate types and their corresponding parameters in response to the specific dataset under analysis. This adaptability empowers **ADAPT** to dynamically tailor the architecture of the quantum circuit, optimizing its performance while adhering to objectives such as maintaining a shallow **depth** and a limited parameter count.

One of the primary challenges faced by the **ADAPT** algorithm is the identification of an optimal set of operators for generating the **ansatz**. Although numerous options exist, this work focuses solely on utilizing the exponentiation of Pauli string operators ($e^{i\theta_d S_d}$). Here, S_d represents the Pauli string operator, and θ_d denotes the associated parameter. The operator pool comprises various conceivable combinations of Pauli operators (X , Y , Z , and I). Consequently, the pool's size is $4^n - 1$, where n denotes the number of qubits; the subtraction of 1 accounts for cases where the Pauli string exclusively consists of identity operators ($I \otimes I \otimes \dots \otimes I$), lacking meaningful parameterized gating.

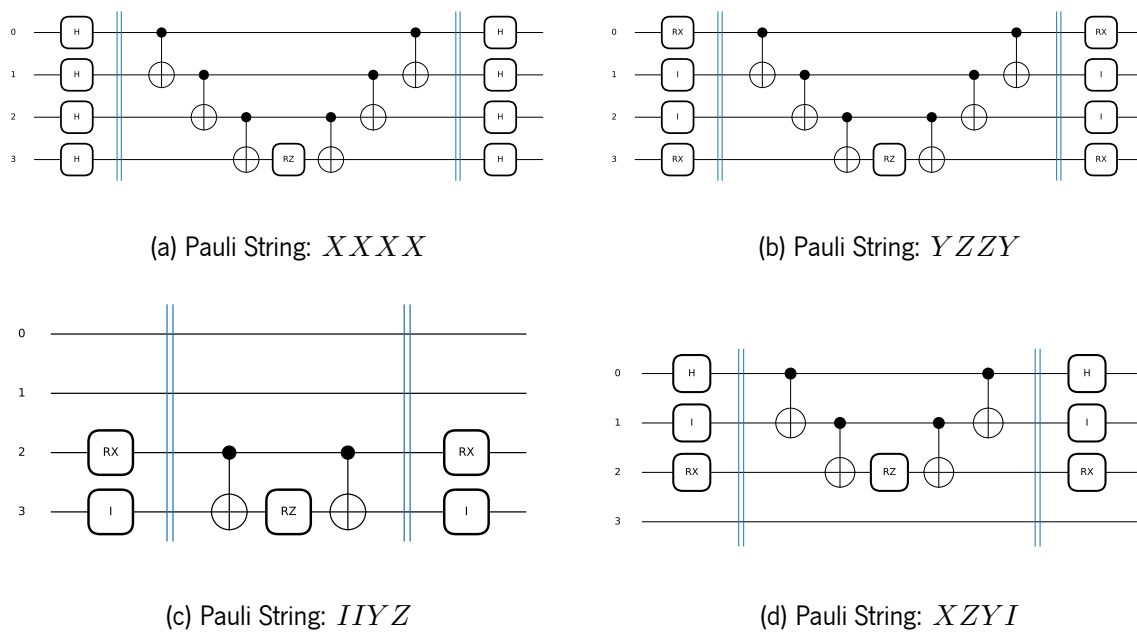


Figure 4: Examples of Exponentiation of Pauli Strings¹

The **cost** function employed in this algorithm is similar to the one used in the rotoselect algorithm:

$$C(\theta) = \sum_m^M (\langle Z \otimes Z \otimes \dots \otimes Z \rangle_m - l_m)^2 \quad (3.5)$$

ADAPT operates through a systematic process that begins with the addition of one operator from the operator pool to the **ansatz** circuit. The operator's parameter added is initialized to zero, as this choice

¹ The R_X gates rotate by $-\pi/2$ radians, and the gate I is the identity gate.

ensures that the operator compiles to the identity operation. Subsequently, the gradient of the newly added operator is computed. Among the operators evaluated for addition, the one displaying the highest gradient in terms of absolute value is then selected for inclusion as a permanent element in the **ansatz**, for the cases shown in Section 5 all the operators in the pool are evaluated. This strategic approach is carefully designed to ensure that the chosen operator wields the most significant impact in reducing the **cost** function. Once integrated, the new operator's associated parameter undergoes refinement via a standard optimization algorithm. This iterative process continues, incorporating further operators into the **ansatz** until a predefined stopping criterion is met.

Several criteria can serve as stopping conditions for the **ADAPT** algorithm. One method involves evaluating whether the gradients of operators within the pool exhibit comparable absolute values, if not the algorithm is terminated. This assessment indicates that the discriminative influence of these operators is diminishing. Alternatively, termination could be triggered when the highest gradient value drops below a predetermined threshold. This situation implies that the operator pool's contribution to the optimization process has become less impactful. Another viable criterion involves employing a fixed parameter count, ensuring that the constructed **ansatz** conforms to specific resource constraints.


The **ADAPT** algorithm's effectiveness lies in its adaptive and data-driven approach to constructing quantum **ansatz** circuits, tailoring them to the specific problem at hand. By iteratively selecting and optimizing operators from a predefined pool, **ADAPT** strikes a balance between circuit **depth**, parameter count, and optimization performance. This makes it a valuable tool for quantum machine learning tasks, where optimizing circuit resources while achieving high performance is crucial.

In summary, the **ADAPT** algorithm provides a robust framework for quantum **ansatz** construction, dynamically responding to the demands of classification problems. By strategically leveraging the exponentiation of Pauli string operators and optimizing their parameters iteratively, **ADAPT** showcases the potential of quantum circuit customization for enhanced problem-solving capabilities.

Algorithm 2: ADAPT-VQC Algorithm

Input: Initial Ansatz A_0 (can be None), Cost function $\mathcal{C}(\theta)$, Dataset \mathcal{D}^M

Output: Optimal rotation angles θ^* , Set of Pauli strings A^*

Generate the Pool of Pauli Strings P_s ; 

if A_0 is None **then**

 Initialize θ as an empty list;

 Initialize A as an empty list;

end

else

 Initialize θ with arbitrary values and optimize it;

$A \leftarrow A_0$

end

repeat

foreach $S_d \in P_s$ **do**

 Initialize Gradient Dictionary $G = \{ \}$;

 Append S_d to A ;

$\theta_d \leftarrow 0$;

 Append θ_d to θ ;

 Calculate the Gradient in absolute value $|G_d|$ for choosing θ_d ;

$G[S_d] \leftarrow |G_d|$;

 Remove S_d from A ;

 Remove θ_d from θ ;

end

 Choose S_d with maximum gradient from G ;

 Append S_d to A ;

$\theta_d \leftarrow 0$;

 Append θ_d to θ ;

 Optimize θ for the chosen S_d ;

until Reaching Limit Number of Parameters;

$A^* \leftarrow A$;

$\theta^* \leftarrow \theta$;

return θ^* , A^* ;


Chapter 4

Experimental Result: Rotoselect

This section is dedicated to showcasing significant outcomes derived from the implementation of the RotoSelect algorithm across a variety of datasets with distinct dimensions. The assessment involves the utilization of two different angle-based encoding methods with the algorithm, as it will be possible to see the performance of this algorithm is highly dependent on the encoding used. Modifications will be applied to the circuit's structure, like variations in the number of encode and ansatz layers and different entanglement strategies for three and four qubits. The ensuing analysis aims to provide a comprehensive understanding of the algorithm's behavior, featuring graphical depictions of accuracy, entanglement, and other pertinent metrics over the training process. In order to make reading of this section easier additional graphs can be found in Appendix A.

The examination begins with datasets spanning varying dimensions, progressing from the smallest to the largest.

4.1 Analysis of Performance for Two-Qubit Systems

The initial dataset under scrutiny is the "Ad Hoc" dataset, sourced from the "qiskit" library [27]. This elementary dataset comprises only two features and is generated using the "IQPEmbedding" encoding. The data is linearly separable through the application of the same encoding technique. The dataset includes a testing dataset with 150 samples and a training dataset with 500 samples. 

To initiate the investigation, the number of encoding layers will be increased, and their influence on performance will be assessed. Figure 5 illustrates a single repetition of the "IQPEmbedding" encoding block, the encoding strategy used to try and solve this classification problem, with additional layers involving a sequential duplication of the depicted structure.

The initial ansatz structure comprises a single layer, which includes two orthogonal axis rotations (Y and Z) and CNOT entanglement. Additionally, two more orthogonal axis rotations are added at the end

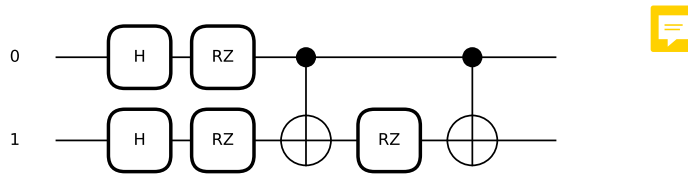


Figure 5: Single Layer of IQPEmbedding Encoding for Two Qubit

for parameter fine-tuning.

The selection of this initial ansatz is entirely arbitrary. It was chosen primarily because it's a commonly employed ansatz capable of addressing a broad spectrum of classification problems.

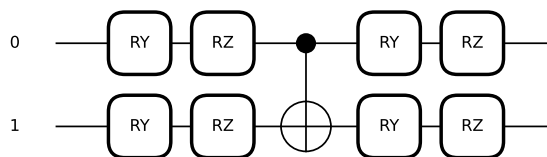


Figure 6: Structure of the Initial Single Layer in the Fixed Ansatz for the Two-Qubit Rotoselect Algorithm

During optimization, 10 cycles sequentially optimize individual rotation gates and their angles. Each step represents the optimization of one rotation angle, with each cycle corresponding to the optimization of every gate and its respective parameter on the ansatz. Figure 7 demonstrates a close approximation between the actual cost and the estimated cost, as outlined in equation 3.1, indicating a highly accurate approximation.

The optimization results for a single encode and ansatz layer are illustrated below. Figure 8a highlights the highest accuracy achieved on the testing dataset, reaching 88%.

Figure 8b demonstrates that the algorithm, despite having the freedom to choose the rotation axes and angles of the gates, is capable of altering the circuit's entanglement¹ even without the ability to modify entanglement gates like a CNOT gate. Consequently, the resulting ansatz structure is as depicted in Figure 9.

Unfortunately, the previous circuit structure did not yield a high-confidence solution. There are several

¹ This metric, known as the Meyer-Wallach measure, was introduced by David A. Meyer and Norbert R. Wallach in their paper [28]. It serves as a tool for quantifying entanglement in pure multi-qubit states, drawing inspiration from the concept of concurrence.

Concurrence is a well-established measure of entanglement tailored for two-qubit states. The Meyer-Wallach measure takes this concept a step further, applying it to systems with more than two qubits. It's defined by deconstructing the multi-qubit state into two-qubit subsystems and quantifying the entanglement that exists between these subsystems.

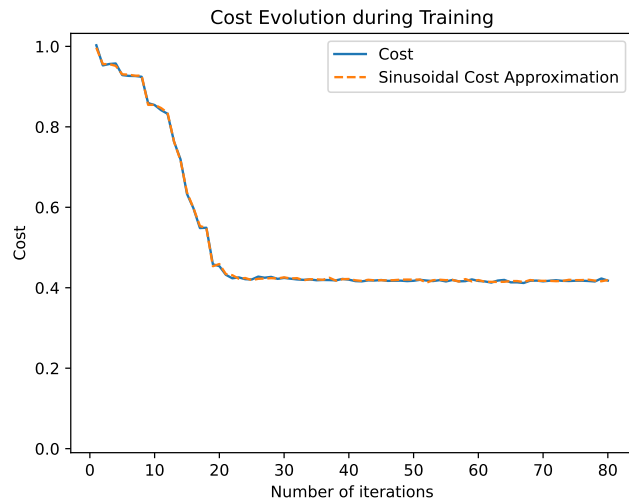
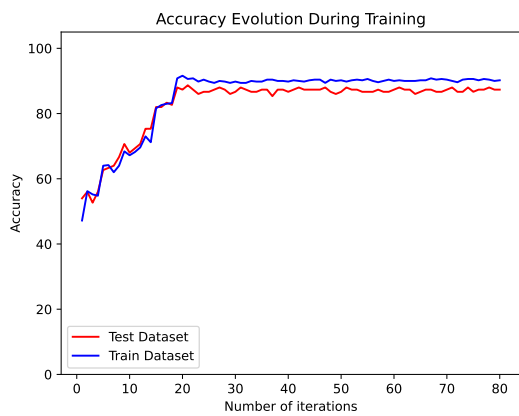
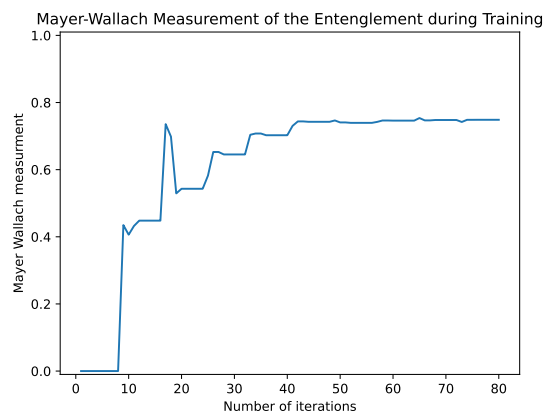


Figure 7: Evaluating the Cost and Approximated Cost for the Quantum Circuit Architecture With a Single Layer of Encoding and Ansatz and Two Qubits



(a) Accuracy



(b) Entanglement

Figure 8: Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, and Two Qubits

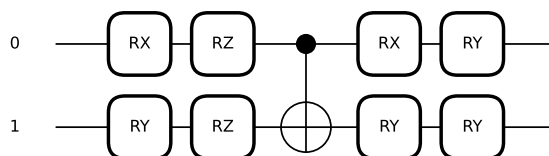


Figure 9: Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, and Two Qubits

adjustable aspects within the structure, such as the number of parameters, the addition of more ansatz layers, or the incorporation of additional encoding layers. In the subsequent investigation, the emphasis will be on varying the number of encoding layers since increasing the number of ansatz layers would significantly extend the computation time.

By increasing the number of encode layers while keeping the initial ansatz structure identical to that shown in Figure 6 and initializing the parameters at zero, valuable insights can be gleaned from the results, illustrated in Figure 10.

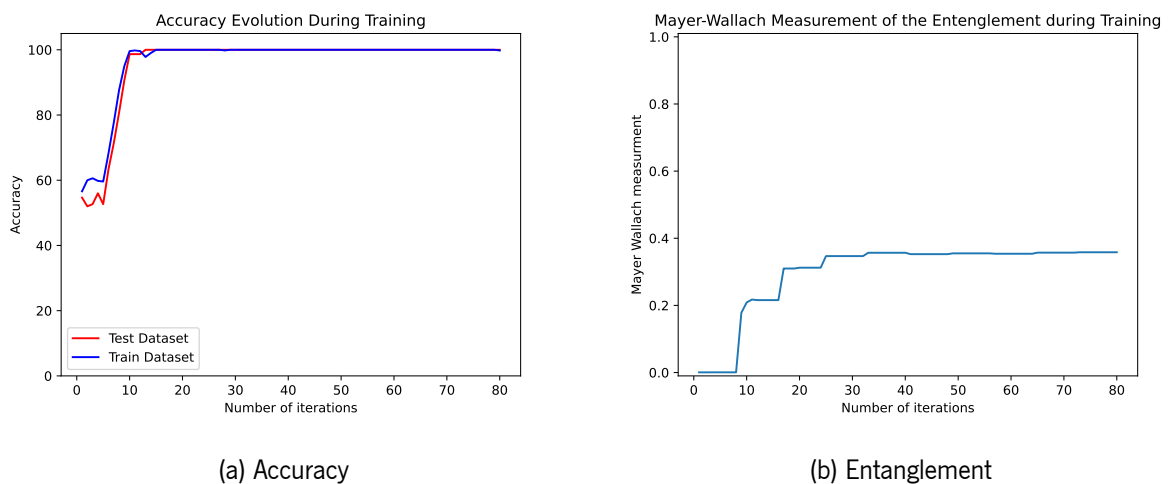


Figure 10: Quantum Circuit Architecture: Two Encode Layers, One Ansatz Repetition, and Two Qubits

As it can be seen, there is a significant improvement in accuracy. The testing dataset accuracy now peaks at 100%, which is a remarkable result for a classification problem.

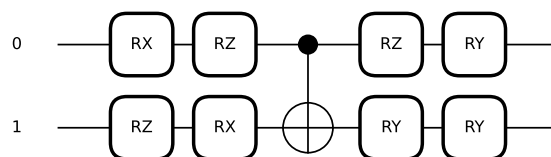


Figure 11: Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, and Two Qubits

This final structure closely resembles the one shown in Figure 9, where 62.5% of the gates remain the same. This highlights the adaptability of the algorithm, which only by changing the number of encode layers can modify the ansatz structure to find better solutions. Furthermore, it demonstrates that the preferred gates that the algorithm tends to change are the first 4 gates (before the CNOT gate), with 50%

of them being altered, while the last 4 gates (after the CNOT gate) experience only one change.

With an increase in the number of encode layers to three, the obtained results are illustrated in Figure 12.

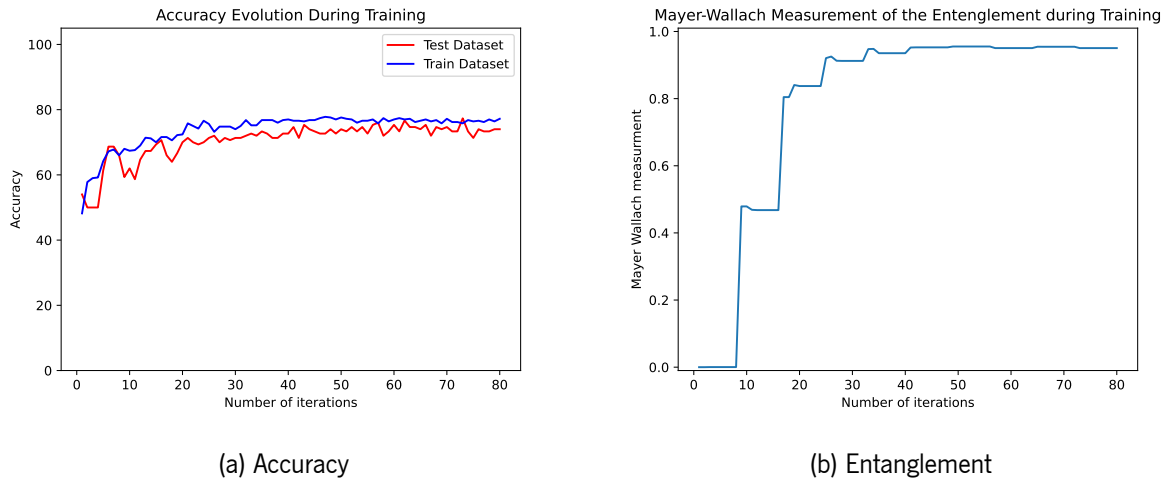


Figure 12: Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, and Two Qubits

Here can easily be seen that there is a drastic decrease in performance, where now the testing dataset accuracy only peaks at 77.3(3)%

It is also possible to see in figure 12b that the optimizer has a difficult problem, while compared with figure 8b, in controlling the necessary entanglement.

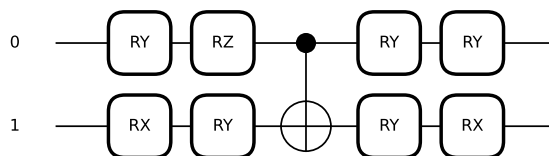


Figure 13: Final Quantum Circuit Ansatz Derived from the RotoSelect Algorithm for the Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, and Two Qubits

Is possible to see that in figure 13 the ansatz structure differs the most from the other two, explaining why the algorithm results in worse results.

As evident from the presented figures, it becomes apparent that the algorithm's optimal performance is achieved when employing two encode layers and a single ansatz layer. That can be even better noticed when the accuracy associated with the testing dataset for each of the quantum circuit architectures deployed above is plotted in the same graph, Figure 14.

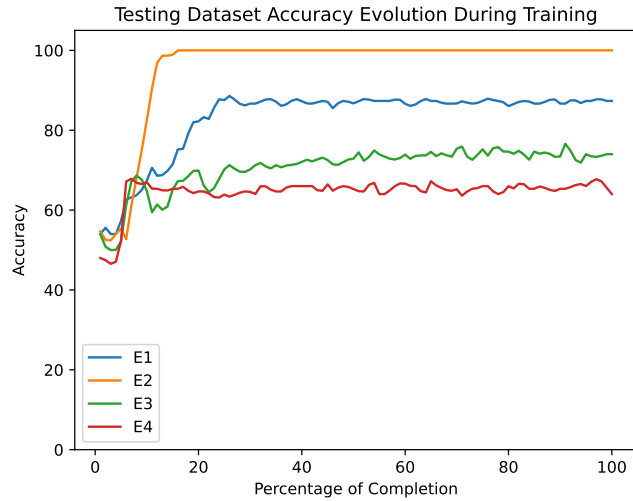


Figure 14: Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4)

The significant drop in algorithm performance for structures with more than two layers of encoding blocks emphasizes the pivotal role of an effective encoding strategy. This observation underscores the RotoSelect algorithm’s sensitivity, akin to traditional Variational Quantum Classifiers, to the quality of the chosen encoding approach.

To further investigate the role of the number of ansatz layers, a fixed number of encoding layers will be maintained while systematically increasing the number of ansatz layers. Instead of directly duplicating the number of gates and thereby significantly increasing the parameter count, each additional ansatz layer introduces only a pair of orthogonal rotations per qubit, while retaining the last two sets of rotation gates at the end. This approach aligns with the primary objective of the adaptive RotoSelect algorithm, which prioritizes maintaining a low circuit depth. This concept is visually demonstrated in Figure 15.

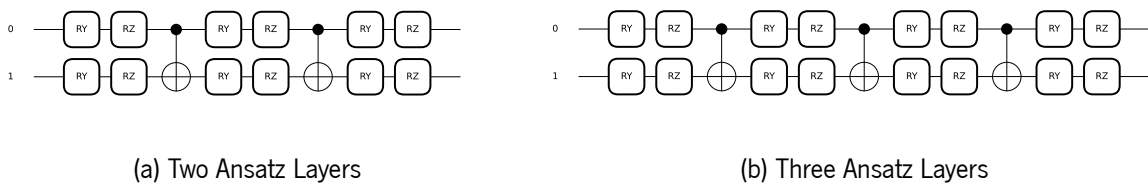


Figure 15: Comparison of Initial Ansatz Structures for Different Numbers of Ansatz Layers

The analysis of the number of ansatz layers will focus on various layer counts. Additional graphs and details are available in Appendix A to enhance the readability of the results section.

A comprehensive comparison of all results is depicted in Figure 16. To ensure smoother visualiza-

tions and enable the comparison of convergence patterns, 100 data points were sampled from the original curves. In this representation, each point corresponds to a percentage of the algorithm’s execution progress. Notably, every 10 points collectively represent one cycle for all the curves.

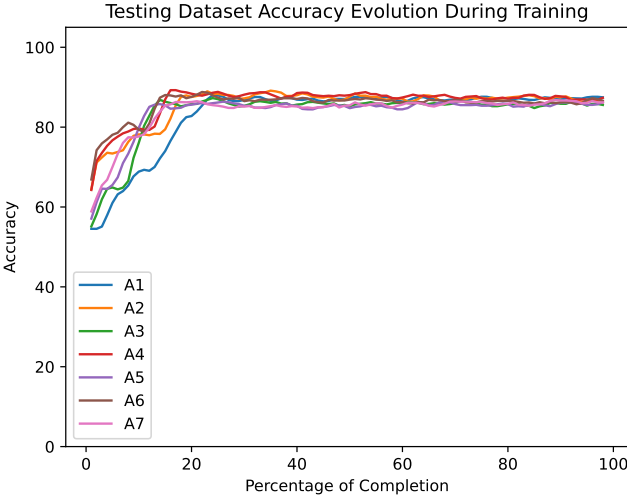


Figure 16: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) while maintaining a fixed number of encode layers (one encode layer) with normalized curves

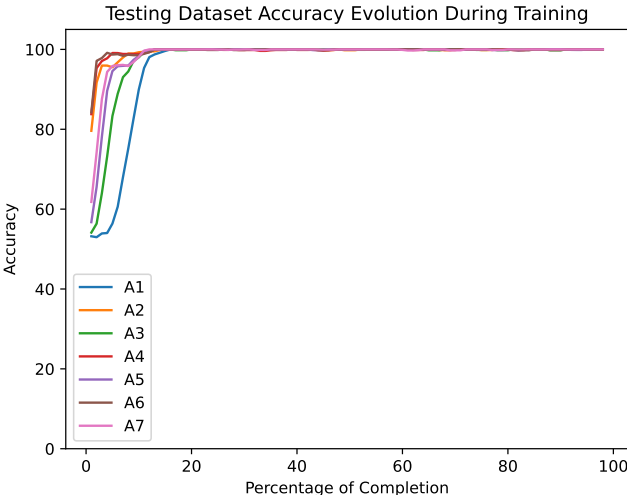


Figure 17: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) while maintaining a fixed number of encode layers (two encode layers) with normalized curves

Increasing the number of ansatz layers has a minimal impact on algorithm accuracy. However, it’s important to note that ansatz structures with more gates and parameters require fewer cycles to achieve comparable accuracy compared to those with fewer gates. Keep in mind that each cycle involving more

parameters and gates also entails more iterations per cycle. Interestingly, having an even number of ansatz repetitions results in the most efficient convergence time.

4.2 Analysis of Performance for Three-Qubit Systems



Introducing a binary fictitious dataset comprising three features, each falling within the range of $0, 2\pi$. Unlike the previous dataset, a unique encoding strategy is being employed here due to certain challenges encountered. The earlier encoding technique proved ineffective in solving this dataset. Both the adaptive algorithms and the conventional Variational Quantum Classifier (VQC) faced difficulties in finding solutions. Consequently, it became necessary to explore an alternative encoding methodology to address this issue. In this instance, a straightforward angle embedding strategy, as described in [29], was employed. This strategy entails representing one feature per qubit using angle encoding. It's important to note that the results obtained using the encoding technique illustrated in Figure 5 will not be presented in this work. These results fail to provide insights into the functioning of the adaptive approaches for this specific dataset. As previously mentioned, the encoding technique plays a critical role in the classification task. However, investigating this role falls beyond this work's objectives, so all the results for this particular dataset will be presented with a single encode layer.

This dataset consists of 70 data points for the training dataset and 30 data points for the testing dataset. For each experiment with different numbers of ansatz layers and parameters, 30 cycles will be performed. As a result, each gate and parameter will be optimized 30 times.

Using the initial ansatz structure depicted in Figure 15 for more than two qubits provides flexibility in implementing various qubit entanglement arrangements. This study will explore two distinct entanglement strategies to gain a deeper understanding of the algorithm's behavior.

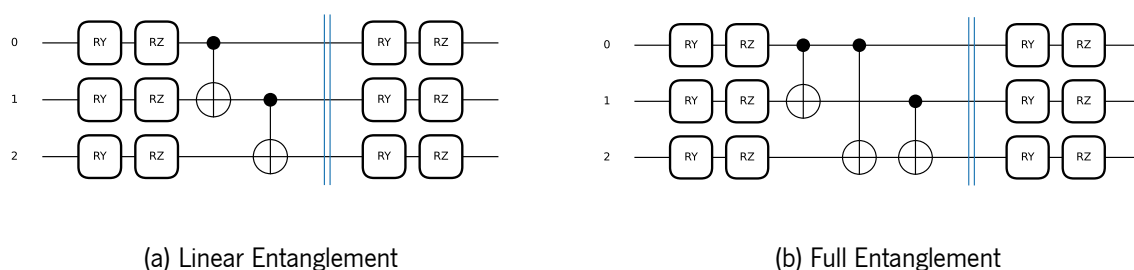


Figure 18: Examples of Different Qubit Entanglement Strategies

These arrangements are known as linear entanglement, illustrated in Figure 18a, and full entanglement, demonstrated in Figure 18b. Notably, the roselect algorithm preserves the positioning and quantity

of *CNOT* gates. Consequently, an expectation arises that these varying entanglement strategies will yield divergent results and exhibit distinct trends in performance.

In Figure 19, the cost function and its approximation defined in Equation 3.1, for a single ansatz layer and encoding using both entanglement strategies are presented. The figures show that the approximation maintains reasonable accuracy, although it doesn't reach the precision exhibited by the two-qubit algorithm (Figure 7).

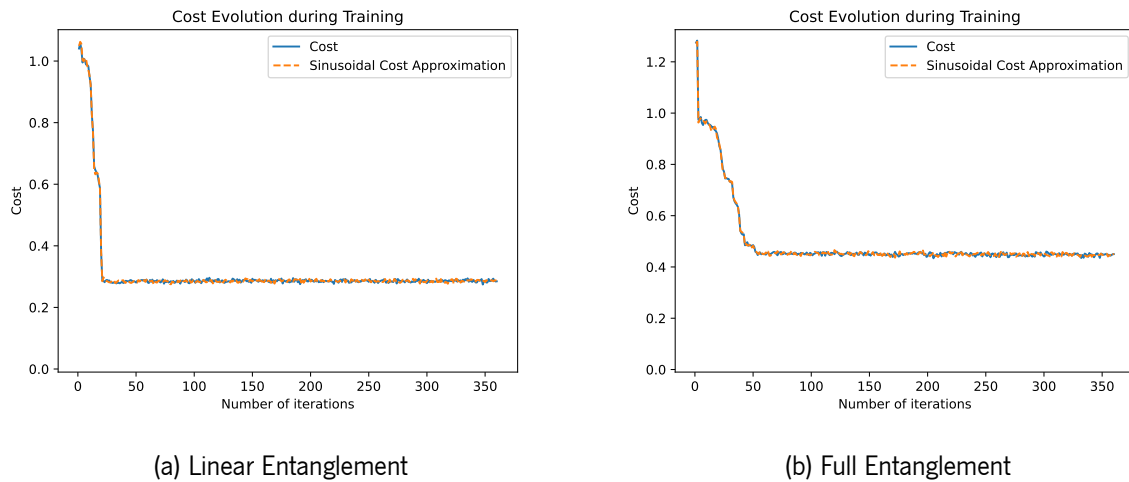


Figure 19: Cost and Approximated Cost Evaluation for the Quantum Circuit Architecture with One Encode Repetition, One Ansatz Repetition and Three Qubits

The classification outcomes for this dataset are displayed below. Results for one to seven ansatz layers will be exhibited, commencing with one layer in Figure 20 and 21.

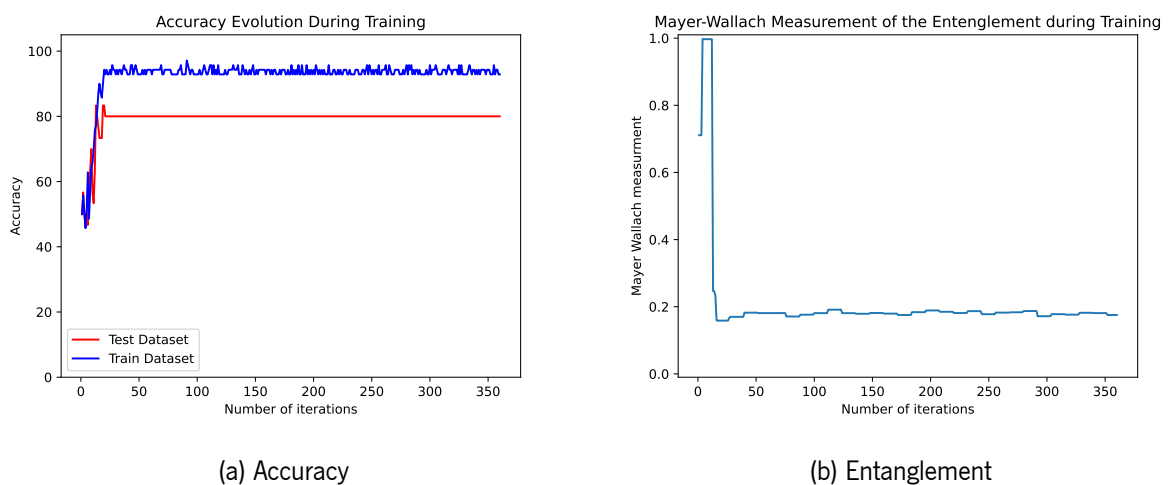
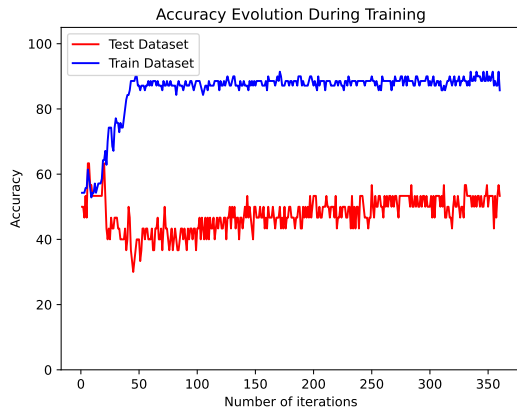
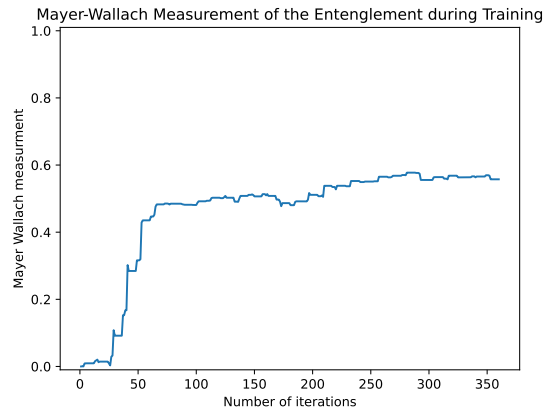


Figure 20: Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, **Linear Entanglement**, and Three Qubits



(a) Accuracy

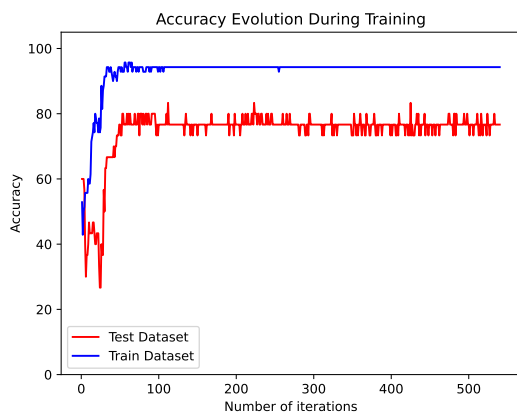


(b) Entanglement

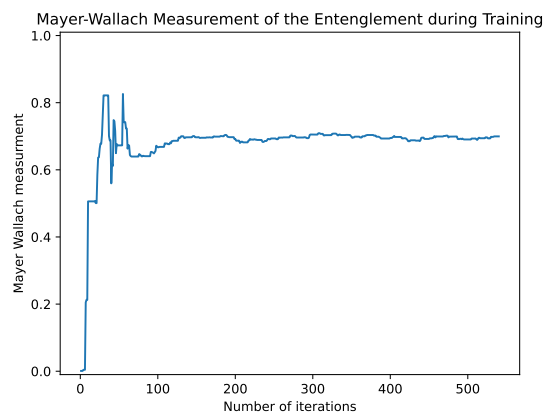
Figure 21: Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, **Full Entanglement**, and Three Qubits

Comparing Figures 20 and 21, it's evident that the algorithm performs better with linear entanglement. Notably, the total entanglement entropy increases for linear entanglement but then quickly drops to around 0.2.

By increasing the number of parameters and consequently the depth of the ansatz using the approach depicted in Figure 15, the following results are observed in Figures 22 and 23



(a) Accuracy

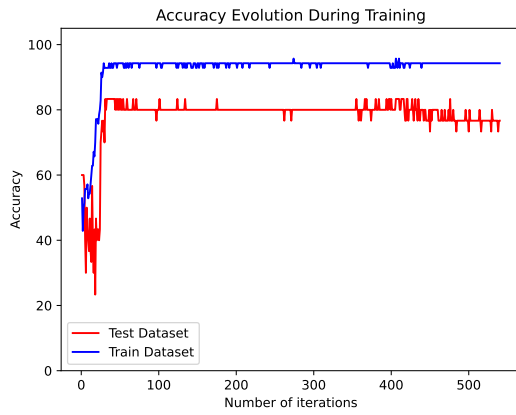


(b) Entanglement

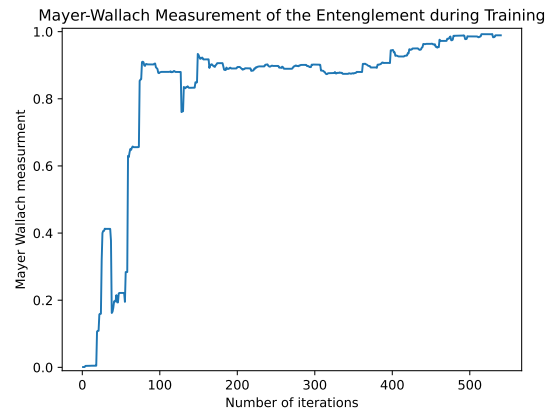
Figure 22: Quantum Circuit Architecture: One Encode Layer, Two Ansatz Layers, **Linear Entanglement**, and Three Qubits

Though not immediately evident, the algorithm's performance was slightly inferior to that depicted in Figure 20b. Notably, the accuracy curves for the testing dataset (red curve) exhibit more fluctuations compared to the graph in Figure 20a, where the accuracy value for the testing dataset initially peaked at

83.(3)% but quickly stabilized at 80%. In the case of two ansatz layers (Figure 22a), although the value still peaked at 83.(3)%, its average value was lower than 80%, with a median of 76.(6)%.



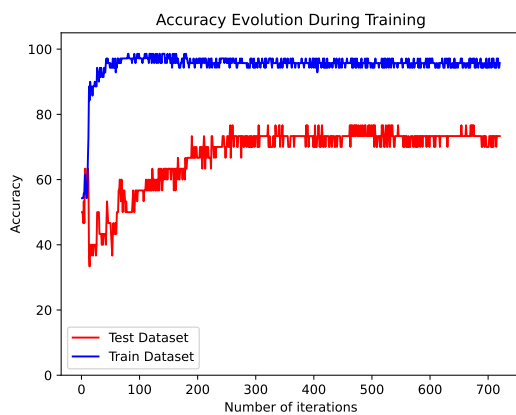
(a) Accuracy



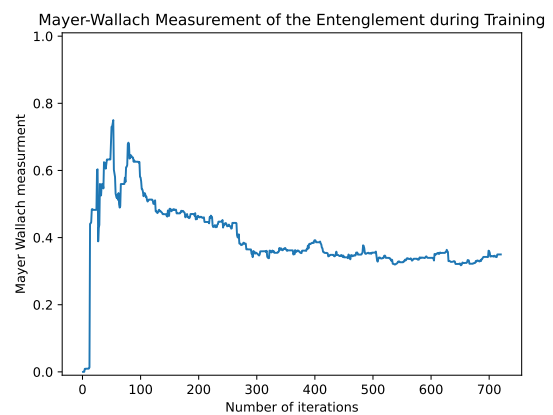
(b) Entanglement

Figure 23: Quantum Circuit Architecture: One Encode Layer, Two Ansatz Layers, **Full Entanglement**, and Three Qubits

On the contrary, a notable enhancement in performance becomes apparent with the "Full Entanglement" strategy when contrasted against the scenario involving a single ansatz layer (Figure 21a). Nevertheless, in this context, the algorithm exhibits indicators that could be interpreted as an early stage of "overfitting" the results. This observation aligns with the slight upswing in the total entanglement of the ansatz, as depicted in Figure 23b. Although this phenomenon has not surfaced in earlier results, its comprehensive investigation is deferred in this study due to time constraints.

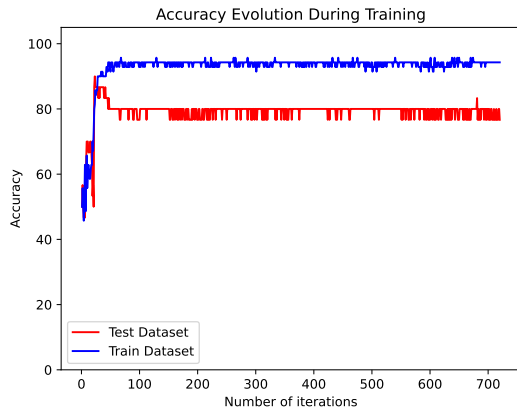


(a) Accuracy

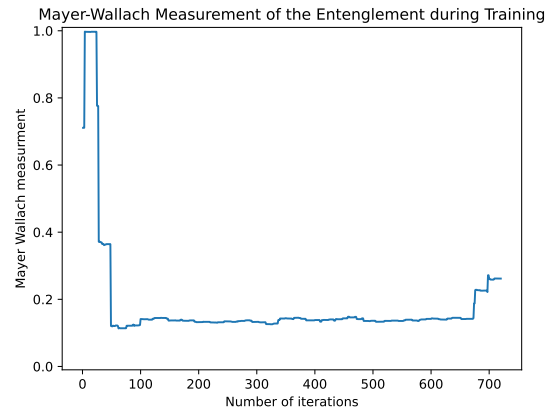


(b) Entanglement

Figure 24: Quantum Circuit Architecture: One Encode Layer, Three Ansatz Layers, **Linear Entanglement**, and Three Qubits



(a) Accuracy



(b) Entanglement

Figure 25: Quantum Circuit Architecture: One Encode Layer, Three Ansatz Layers, **Full Entanglement**, and Three Qubits

When considering three ansatz layers, a distinct decline in performance becomes evident for the "linear entanglement" strategy. However, a minor increase in performance is observed for the "full entanglement" strategy

A comprehensive comparison of all the results is presented in a single graph in figures 26 and 27, where testing accuracy curves have been uniformly sampled with 100 points each. This approach ensures smoother visualizations and enables the direct comparison of the convergence patterns across different circuit structures.

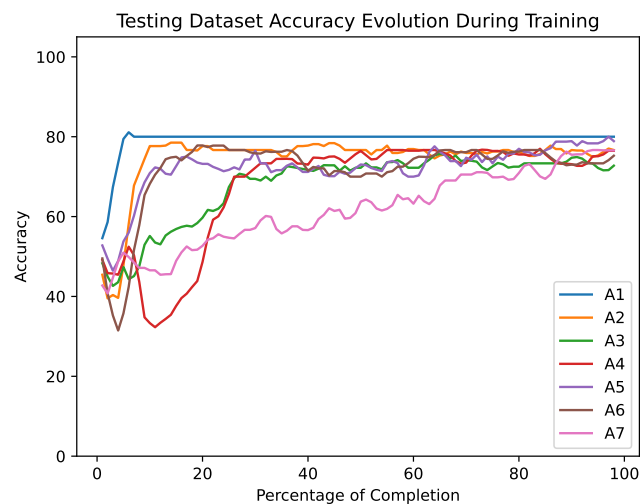


Figure 26: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) using the "Linear Entanglement" Strategy, with normalized curves.

The analysis of figure 26 demonstrates that the algorithm attains its optimal performance with a single ansatz layer. This observation is intriguing, as it indicates that the solution with the fewest parameters produces the best results for the problem. Such an outcome aligns with the desired behavior for adaptive algorithms.

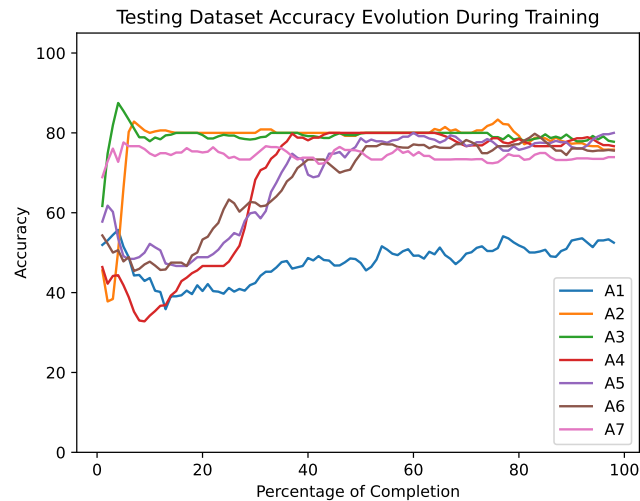


Figure 27: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) using the "Full Entanglement" Strategy, with normalized curves.

In figure 27, the solution with the fewest parameters exhibited the poorest results, while solutions with two and three ansatz layers yielded the best outcomes.

This disparity between figures 26 and 27 underscores the profound influence of the ansatz structure. Evidently, the strategy with a fewer number of *CNOT* gates performed better with fewer parameters, whereas the strategy with more gates exhibited inferior performance, particularly with only one repetition. This observation emphasizes that while the algorithm can tailor solutions to specific problems, the provided ansatz structure significantly impacts the results it generates.

4.3 Analysis of Performance for Four-Qubit Systems

A new binary dataset with four features is employed using the well-known "Iris" dataset. Merging its three classes into two in order to create a binary dataset. The training phase includes 70 data points, while the testing phase comprises 30.

The choice of encoding between the "IQPEmbedding," as illustrated in figure 5, and the "AngleEmbedding" encoding employed in Section 4.2 for a typical classification task may seem straightforward. The

"AngleEmbedding" approach yields superior results, which can be seen in Appendix A. However, for the sake of a straightforward comparative analysis of the results, the "IQPEmbedding" encoding will be used.

To gain deeper insights into the algorithm's performance on this specific problem, the "IQPEmbedding" encoding technique will be employed. Similar to Section 4.1, the influence of encoding will be demonstrated, and akin to Section 4.2, two entanglement techniques will be studied.

Before presenting the results, it's worth noting in Figure 28 that the approximation demonstrates remarkable performance. It's evident that, as shown in Figure 28a, the cost function appears to have converged at an earlier stage, but it continues to decrease further. This aspect poses a challenge for the algorithm, as determining whether the cost has genuinely converged or could still decrease is difficult without trial and error. Consequently, defining the number of cycles remains a complex task

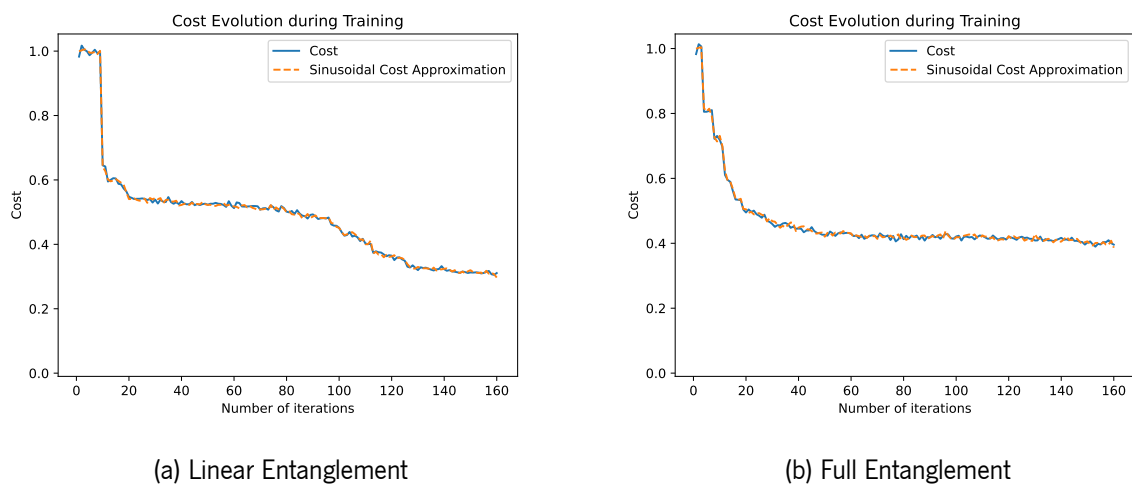
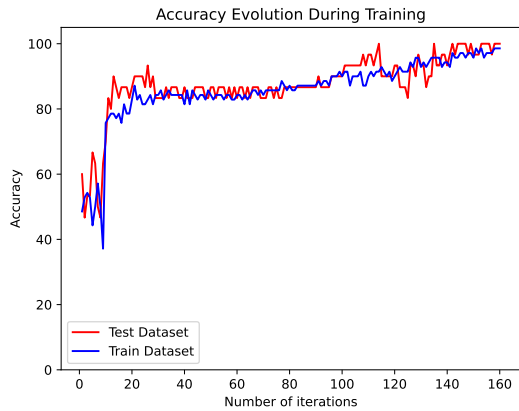


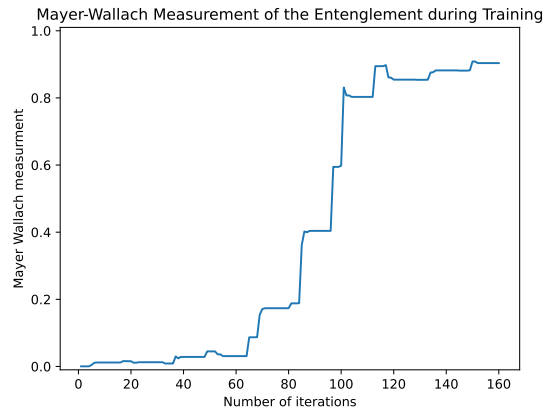
Figure 28: Cost and Approximated Cost Evaluation for the Quantum Circuit Architecture with One Encode Layer, One Ansatz Layer, and Four Qubits

Comparing the figures 29 and 30 the linear entanglement strategy demonstrates superior performance compared to the full entanglement strategy. Notably, in the linear case, the accuracy of the testing dataset surpasses that of the training dataset. The significance of a well-chosen encoding technique is evident, as only the linear entanglement strategy peaks at 100% accuracy. However, achieving this level of accuracy required a considerably higher number of cycles compared to the results shown in Figures 53a and 53b where the entanglement strategy did not have a significant impact.

Similar to the results presented in section 4.1, the algorithm performs optimally with two encode layers, particularly in structures with the fewest *CNOT* gates. It appears that this ansatz configuration provides the algorithm with enhanced control over entanglement. After an initial increase in entanglement, the algorithm reaches approximately 100% accuracy, and the circuit's entanglement stabilizes, with minor

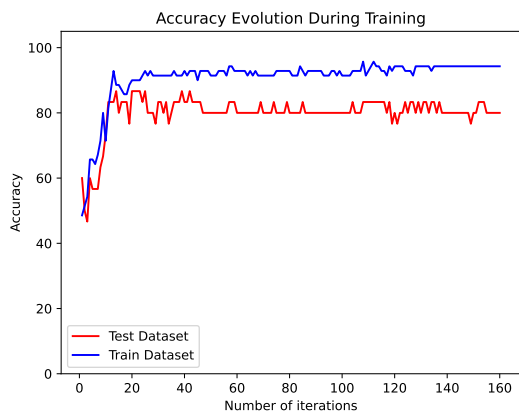


(a) Accuracy

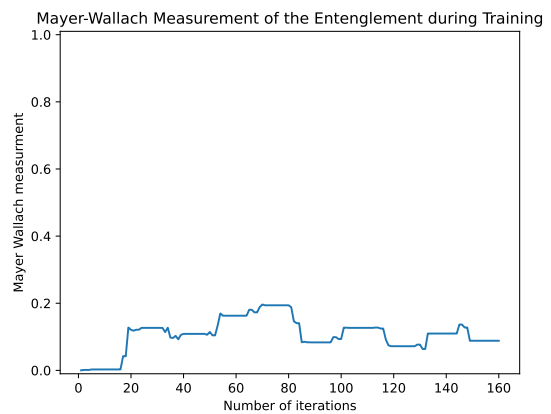


(b) Entanglement

Figure 29: Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, **Linear Entanglement**, and Four Qubits



(a) Accuracy



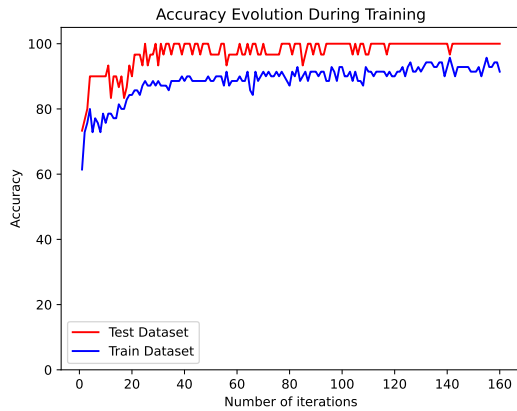
(b) Entanglement

Figure 30: Quantum Circuit Architecture: One Encode Layer, One Ansatz Layer, **Full Entanglement**, and Four Qubits

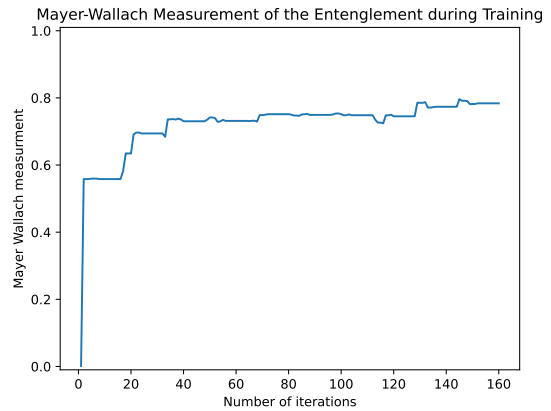
fluctuations

Figure 32, which involves two encode layers, reveals that the algorithm encounters difficulties in achieving satisfactory results. This is in stark contrast to Figure 31, where the testing dataset's accuracy consistently surpasses the training accuracy throughout the optimization process. This disparity underscores the complex interplay between the optimization landscape, entanglement structure, and the specific characteristics of the problem.

When the number of encode layers was increased to three, as illustrated in figures 33 and 34, a noticeable degradation in the algorithm's performance for the linear entanglement strategy becomes evident.

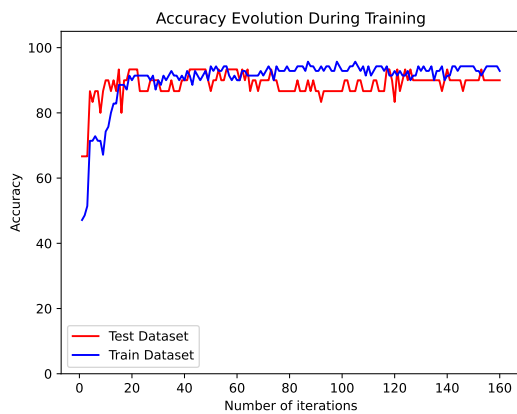


(a) Accuracy

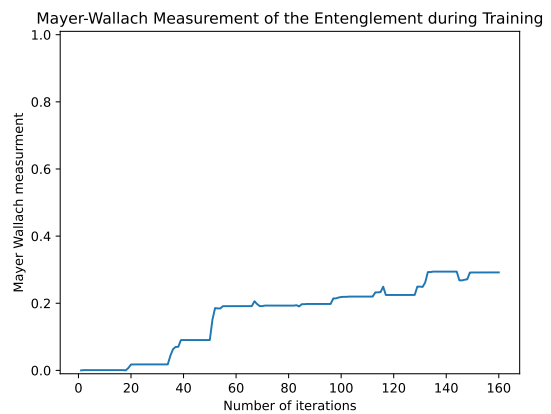


(b) Entanglement

Figure 31: Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, **Linear Entanglement**, and Four Qubits



(a) Accuracy



(b) Entanglement

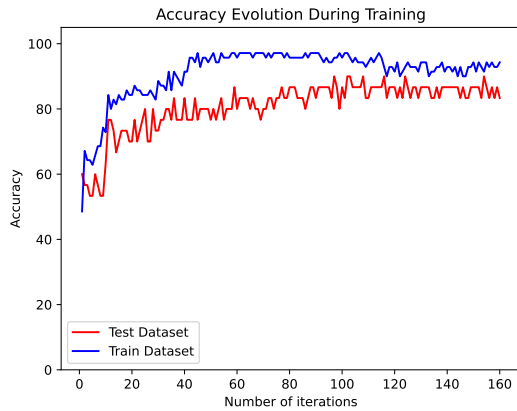
Figure 32: Quantum Circuit Architecture: Two Encode Layers, One Ansatz Layer, **Full Entanglement**, and Four Qubits

The overall performance experiences a decline. This intricate relationship between entanglement entropy, optimization, and performance underscores the complexity of quantum optimization.

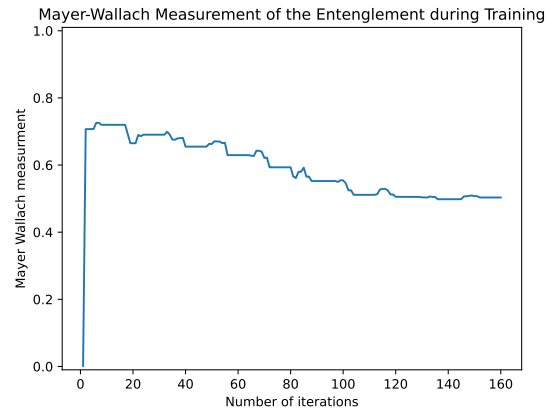
Interestingly, within the full entanglement strategy, augmenting the number of encode layers did not hinder but, in fact, marginally improved the algorithm's performance.

To gain a clearer understanding of the influence of the number of encode layers on the algorithm's performance, the subsequent graphs depict the accuracy of the testing dataset for four distinct numbers of encode layers (ranging from one to four) for each of the employed entanglement strategies.

This allows for a detailed comparison of how the algorithm's performance responds to variations in

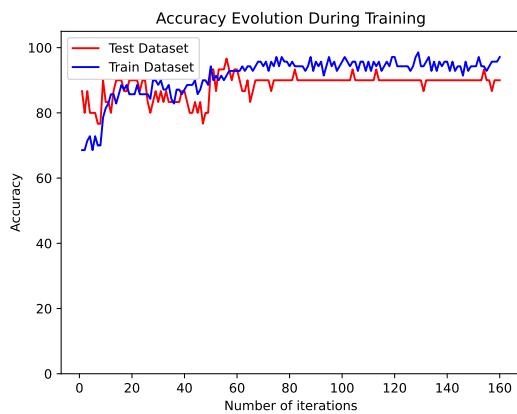


(a) Accuracy

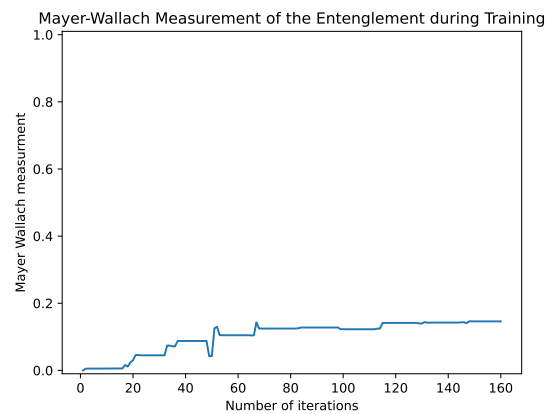


(b) Entanglement

Figure 33: Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, **Linear Entanglement**, and Four Qubits



(a) Accuracy



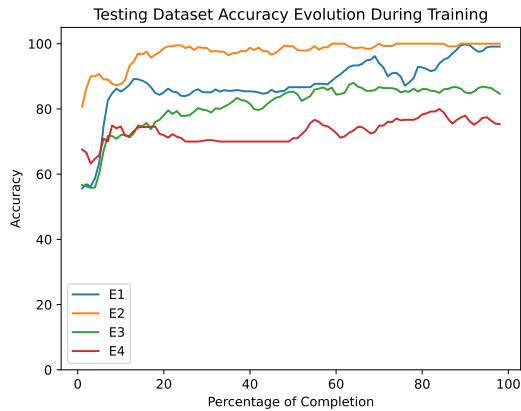
(b) Entanglement

Figure 34: Quantum Circuit Architecture: Three Encode Layers, One Ansatz Layer, **Full Entanglement**, and Four Qubits

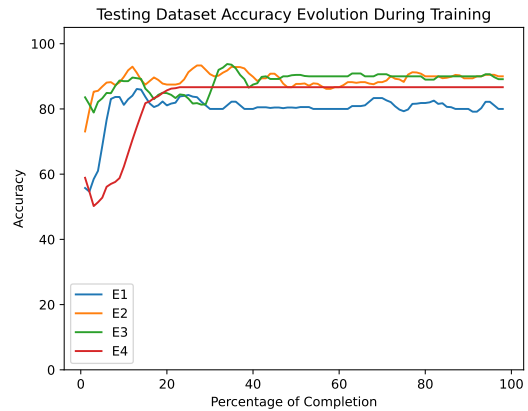
the number of encode layers across different entanglement strategies. By evaluating these trends, a more comprehensive perspective on the impact of the number of encode layers can be obtained.

Both Figures 35a and 35b provide insightful evidence of the substantial influence encoding has on the algorithm's performance, reaffirming the earlier discussions regarding the paramount role of effective encoding techniques. Importantly, these results underscore the pivotal nature of proper encoding, as it can significantly shape the algorithm's performance trajectory.

Of particular interest is the differing response of the linear entanglement and full entanglement strategies to encoding techniques. The linear entanglement strategy appears to be more susceptible to the



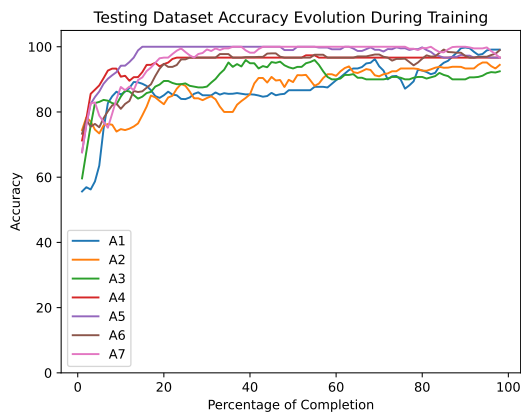
(a) Linear Entanglement



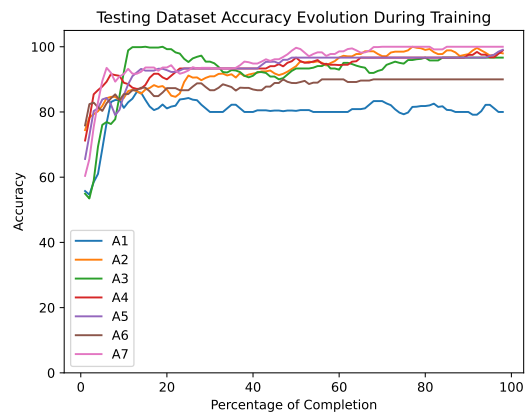
(b) Full Entanglement

Figure 35: Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4), with normalized curves.

impact of encoding, showcasing its best performance with two encode layers while experiencing a decline with four encode layers. In contrast, the full entanglement strategy doesn't appear to be as affected as the linear entanglement strategy. It consistently achieves almost the same performance across different numbers of encode layers, with the exception of one encode layer.



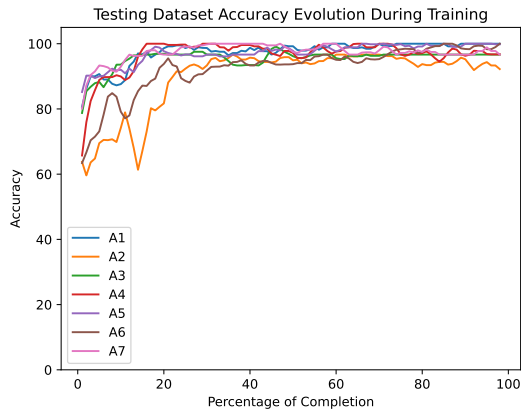
(a) Linear Entanglement



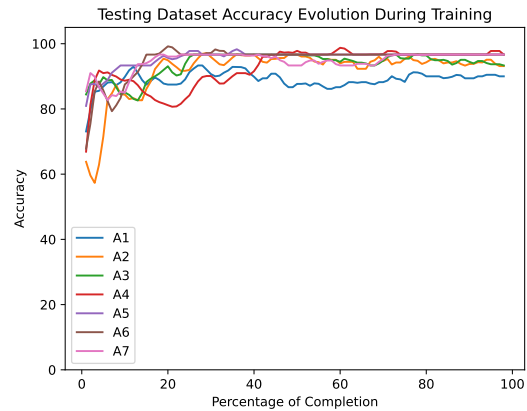
(b) Full Entanglement

Figure 36: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves (One Encode repetition).

An analysis of ansatz depth's influence reveals that, as shown in Figures 36a, 36b, 37a, and 37b, its impact is less pronounced compared to the effect of encoding. However, a crucial observation emerges: the Linear Entanglement strategy consistently yields similar results across various depths, emphasizing its resilience to changes in ansatz depth. On the contrary, the Full Entanglement strategy exhibits sensitivity



(a) Linear Entanglement



(b) Full Entanglement

Figure 37: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves (Two Encode repetition).

to ansatz depth, echoing the findings in Section 4.1. Notably, employing two encode repetitions (Figures 37a and 37b) consistently provides improved results, and the discrepancy in outcomes stemming from varying ansatz depth is notably narrower compared to using a single encode layer (Figures 36a and 36b).

4.4 Summary

The algorithm's performance is influenced by several factors, including ansatz structure, ansatz depth, entanglement strategy, encode structure, and encode depth. Remarkably, the algorithm consistently achieves high accuracy, often reaching 100% precision.

Among the findings, the linear entanglement strategy consistently performs well across different numbers of ansatz layers, proving its robustness. The impact of encoding is substantial, with effective encoding techniques significantly enhancing the algorithm's performance. Interestingly, the full entanglement strategy shows greater sensitivity to ansatz depth, while the Linear Entanglement strategy remains relatively stable.

Furthermore, the relationship between entanglement entropy and performance isn't straightforward. One conclusion can be drawn: after the algorithm reaches a confident solution, the Mayer-Wallach Measurement of entanglement seems to stabilize. This suggests that the algorithm, even though it has no ability to change the entanglement gates directly by having the freedom to adjust the axes of rotations and their angles, can effectively control the entanglement within the circuit.

In conclusion, the performance of the RotoSelect algorithm is influenced by complex interactions

among various parameters. Effective encoding techniques and entanglement strategies are crucial, but the optimal approach can vary depending on the specific problem and context. This underscores the need for an adaptive algorithm that also selects the two-qubit gates, as learning the structure of entanglement is crucial, which the RotoSelect algorithm does not account for.

Chapter 5

Experimental Result: ADAPT

This section explores the results of the ADAPT algorithm across datasets of varying dimensions. One of the key differences between the ADAPT algorithm and the previously discussed RotoSelect algorithm is the ADAPT algorithm's ability to incorporate unparameterized two-qubit gates like CNOTs. This feature provides greater control over the entanglement of the quantum circuit. Additionally, the ADAPT algorithm does not have a fixed circuit structure. Instead, it builds the ansatz from scratch during the optimization process. This means that new operators are added dynamically, taking into account the partial gradients of these operators, as discussed in Section 2.3.2.

Two angle-based encoding methods will be employed to emphasize their influence on algorithm performance. Structural modifications involve changes in the number of encode layers while maintaining consistency with 10 parameterized operators for all experiments. Each parameter optimization is performed using the SPSA¹ optimizer from the PennyLane library, with 300 iterations per optimization step. This sums up to a total of 10×300 iterations for the experiments, the pool used for all the results is the same one described in Section 2.3.2.

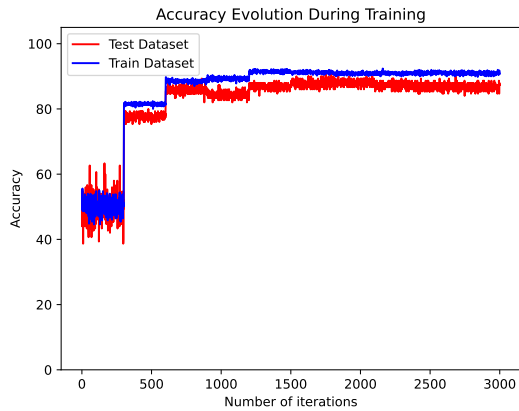
The objective is to provide a comprehensive understanding of the algorithm's behavior, using graphical representations for metrics like accuracy and cost throughout training.

This section follows a structure similar to Section 4 and employs the same datasets.

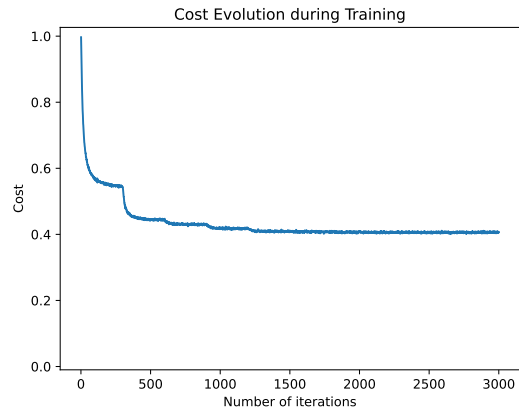
5.1 Analysis of Performance for Two-Qubit Systems

Utilizing the same dataset featured in Section 4.1 and employing the "IQPEmbedding" encoding depicted in Figure 5, the objective is to try and solve the problems with the inclusion of 10 parameterized operators. The results will be presented below, with an examination of variations in the number of encode layers.

¹ The SPSA algorithm is designed to efficiently estimate gradients through perturbations during optimization. These perturbations involve introducing random parameter changes within the objective function. This unique approach equips SPSA with robustness, enabling it to approximate gradients accurately, even when faced with challenges such as noise, quantum hardware limitations, or other sources of uncertainty [30].



(a) Accuracy

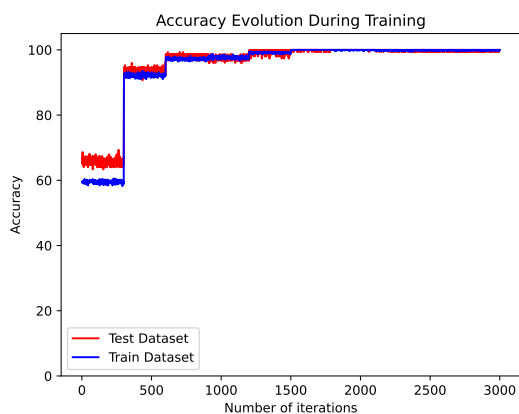


(b) Cost

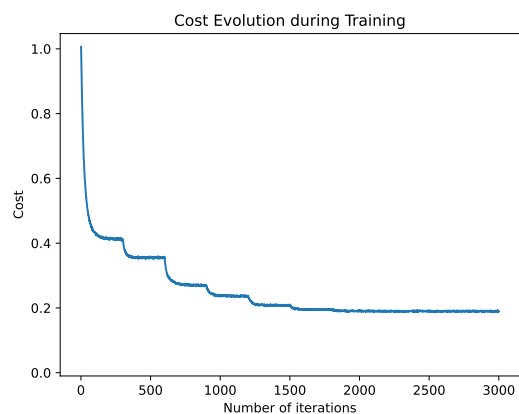
Figure 38: Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Two Qubits

Several noteworthy observations can be made instantly when observing figure 38. Firstly, especially at the outset of the optimization process, there is a discernible pattern: when a new operator is introduced to the circuit, it manifests as a sudden spike in the accuracy curve (as seen in Figure 38a), and conversely, as a sharp dip in the cost curve (as seen in Figure 38b). These behaviors are desirable in this type of algorithm because they indicate that adding operators has a positive impact on algorithm performance, with accuracy increasing and cost decreasing as a result.

Another notable observation is the presence of frequent fluctuations in the accuracy curve (as seen in Figure 38a). These fluctuations occur during the parameter optimization process, and while expected to some extent, they are generally not desired.



(a) Accuracy



(b) Cost

Figure 39: Quantum Circuit Architecture: Two Encode Layers, Ten Parameterized Operators and Two Qubits

Increasing the number of encode layers to two has notable effects. The dips in the cost function become more frequent with the addition of operators, while the fluctuations in accuracy become less conspicuous. Interestingly, the accuracy reaches a peak at 100%, and the cost function converges to almost half the value achieved with a single encode layer. This behavior underscores the influence of the number of encode layers on the optimization process.

The results in both figures 38 and 39 align with those obtained in Section 4.1, particularly in terms of accuracy.

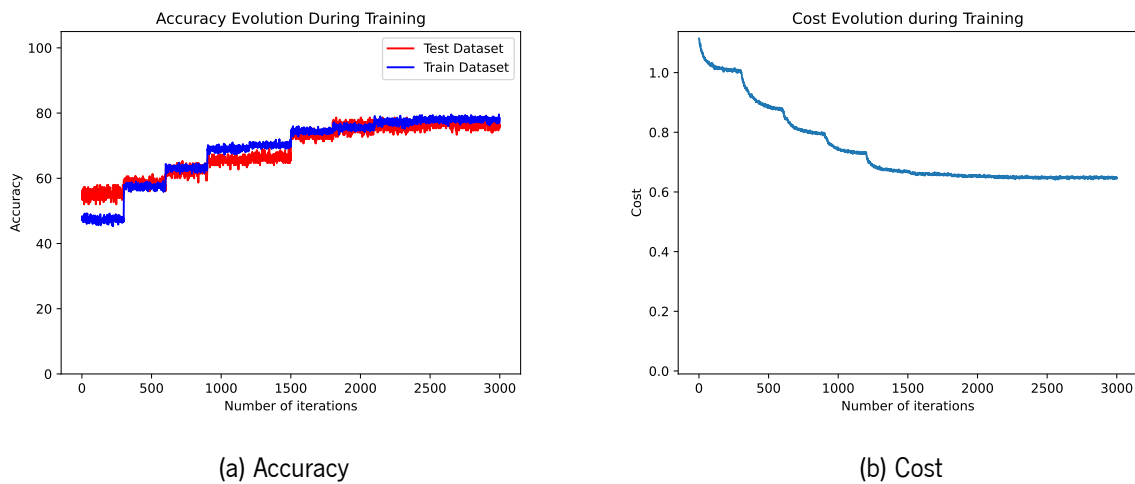


Figure 40: Quantum Circuit Architecture: Three Encode Layers, Ten Parameterized Operators and Two Qubits

As observed in section 4.1, the implementation with two encode layers yields the most favorable results. With three encode layers, figure 40, the jumps in the accuracy curve and the dips in the cost function become less pronounced when compared to the results in figure 39. However, it's important to note that the fluctuations in accuracy are more conspicuous in this scenario compared to the fluctuations in figure 38.

When comparing figure 41 with figure 14, it becomes apparent that adaptive algorithms, although fundamentally distinct from each other in terms of their underlying methodologies and approaches, surprisingly demonstrate striking similarities in terms of accuracy. This observation raises intriguing questions about the fundamental principles that may be influencing the convergence of these seemingly disparate algorithms.

It is noteworthy that despite differences in their core mechanisms, both adaptive algorithms appear to converge toward comparable levels of accuracy. This convergence behavior may suggest the existence of certain common features or principles inherent in the problems they address or in the optimization

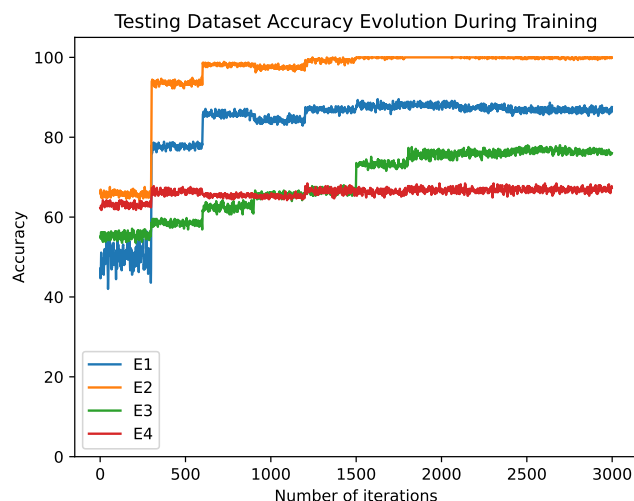


Figure 41: Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4)

landscapes they navigate.

Furthermore, an exploration and comprehension of these accuracy similarities could yield valuable insights into the broader realm of optimization algorithms, particularly those employing adaptive strategies. It might open the door to the development of hybrid algorithms that harness the strengths of both approaches, potentially leading to even more impressive results. This cross-fertilization of ideas and techniques is a hallmark of scientific progress that can lead to exciting breakthroughs across various domains.

5.2 Analysis of Performance for Three-Qubit Systems

Utilizing the identical dataset and "AngleEmbending" encode technique employed in Section 4.2.

In figure 42a, a distinctive behavior in the accuracy curve becomes evident. Initially, it peaks at an impressive 90%, showcasing the algorithm's rapid capacity to achieve high accuracy. However, when introducing a new operator, an intriguing pattern emerges. Instead of experiencing a further boost in accuracy, as one might expect, the accuracy takes a sharp decline, settling at around 80%. This observation implies that merely adding more operators to the quantum circuit doesn't necessarily guarantee improved accuracy on the testing dataset. It's worth noting that while the cost function decreases, indicating optimization progress, this decrease primarily reflects an increase in the accuracy of the training dataset.

Remarkably, this phenomenon shares similarities with Figure 25a, where the accuracy of the testing dataset also exhibits an early surge to 90% during optimization but then gradually descends to around 80%. This intriguing parallel in behavior prompts further investigation into the underlying dynamics of both the

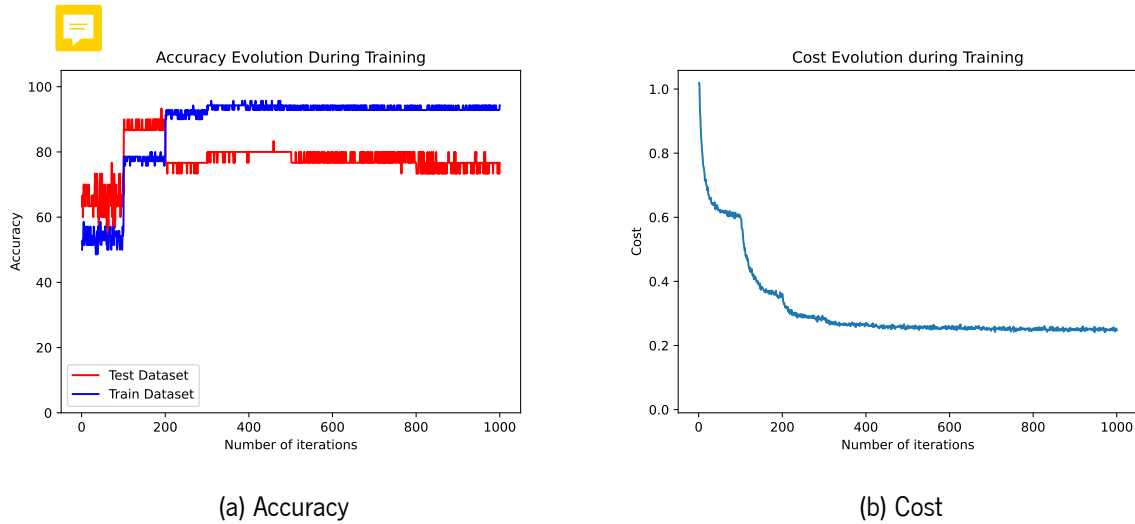


Figure 42: Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Three Qubits

ADAPT algorithm and the RotoSelect algorithm, as such insights could potentially lead to advancements in optimization algorithms.

5.3 Analysis of Performance for Four-Qubit Systems

In the case of the "Iris" dataset, both "AngleEmbedding" encode and the "IQPEmbedding" technique were applied, following the methodology used in Section 4.3, the results for the "AngleEmbedding" technique can be seen in Appendix A.

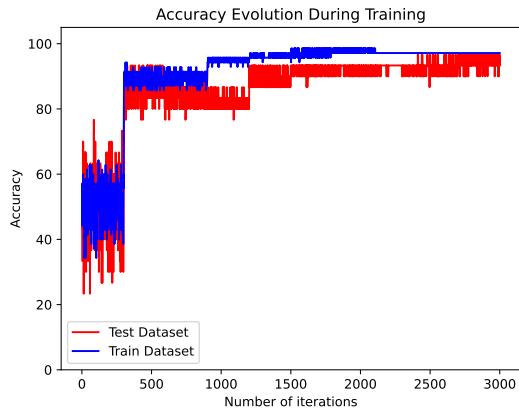
Using the "IQPEmbedding" encoding, which previously yielded suboptimal results in Section 4.3, it was expected that this encoding approach would also lead to less favorable outcomes for this specific dataset when compared to the "AngleEmbedding" encode showcased in figure 54.

As it can be seen in figure 43, the testing dataset peaked at 96.6% accuracy. This accuracy, although quite high, falls slightly short of the perfect 100% accuracy achieved with the angle-based encoding, and it's noteworthy that this level of accuracy was attained with the addition of more operators than in the angle-based encoding scenario.

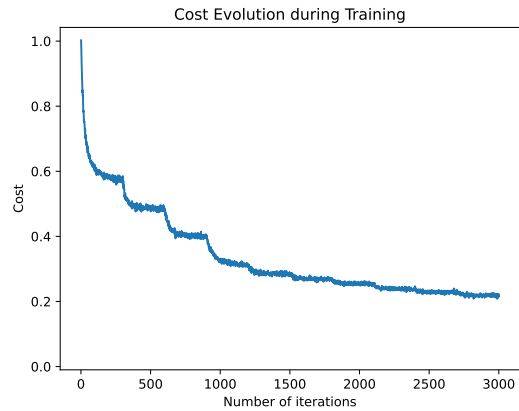
Increasing the number of encode layers to two the results are showcased in figure 44.

In contrast to previous sections, increasing the number of encode layers to two did not yield an improvement in performance. This comparison is more apparent when examining figure 45.

In Figure 45, it is evident that the strategy with a single encode layer yielded the best results, even though they are very similar to the results obtained from the two encode layers structure.

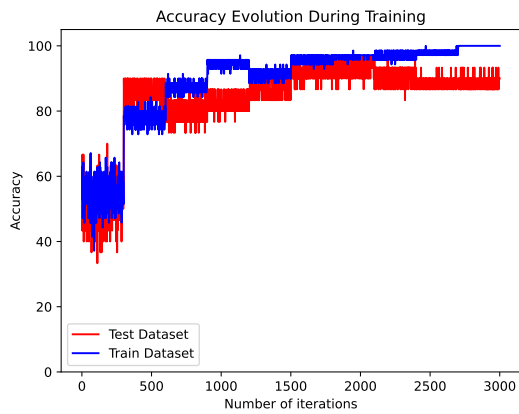


(a) Accuracy

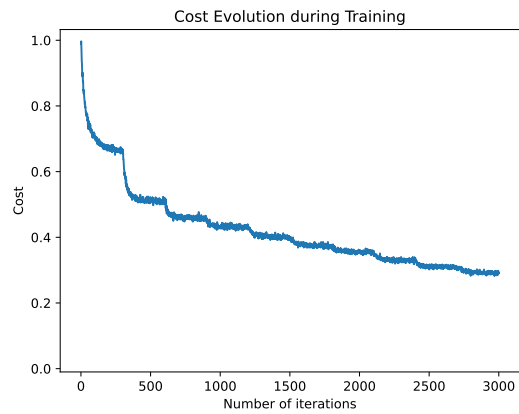


(b) Cost

Figure 43: Quantum Circuit Architecture: One Encode Layer, Ten Parameterized Operators and Four Qubits



(a) Accuracy



(b) Cost

Figure 44: Quantum Circuit Architecture: Two Encode Layers, Ten Parameterized Operators and Four Qubits

5.4 Summary

The ADAPT algorithm is a versatile quantum machine learning approach. It exhibits a high degree of flexibility in adapting to various datasets and encoding strategies. ADAPT efficiently optimizes parameterized quantum circuits, enabling them to tackle complex classification tasks. It demonstrates the potential to achieve high accuracy, often reaching 100% accuracy on both training and testing datasets, albeit with variations depending on factors such as encode strategy, number of encode layers and dataset characteristics. The algorithm's sensitivity to these factors highlights the importance of effective encoding techniques and the potential for optimizing quantum circuits to solve classification problems. The operator pool used

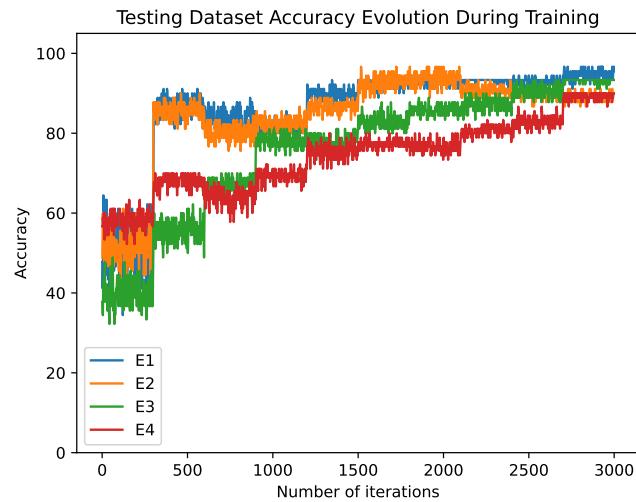


Figure 45: Comparative Analysis of Quantum Circuits with Different Numbers of Encode Layers (E1, E2, E3, E4))

in these problems appears to be a reasonable choice. However, as the problem size grows, the complexity of finding the operator with the highest gradient in magnitude significantly increases thanks to its brute force search method. This pool was chosen for its generality and its applicability to various problems beyond classification. However, this generality comes at the cost of increased complexity. In contrast, the ADAPT-VQE and ADAPT-QAOA algorithms, which have been extensively studied, offer problem-specific operator pools. For instance, specific pools have been designed for solving VQE problems [7] and for the QAOA [20]. Exploring the design of an efficient operator pool tailored to the ADAPT-VQC case could be a promising avenue for future research.



Chapter 6

Comparative Analysis

This section undertakes a comparative assessment of three quantum machine learning algorithms: RotoSelect, ADAPT, and the conventional (Standard) Variational Quantum Classifier (VQC). The central aim is to evaluate their performance across various datasets and problem dimensions, providing insights into their relative strengths and weaknesses.

Several performance metrics, including accuracy, convergence speed, robustness, and sensitivity to hyperparameters, can be employed to scrutinize these algorithms. However, the primary focus of this analysis is accuracy. By comparing their capabilities, this study aims to offer guidance for selecting the most suitable algorithm for specific problem domains, while also identifying potential avenues for future research in quantum machine learning.

For a comprehensive comparison, this section presents results obtained using the conventional Variational Quantum Classifier (VQC) with varying hyperparameters. These outcomes will be juxtaposed with the results achieved using the Rotoselect and ADAPT algorithms. It's essential to note that the same datasets employed in sections 4 and 5 are utilized here to maintain consistency. The ansatz structure employed is identical to that used for the RotoSelect algorithm, leveraging the Efficient SU-2 design as depicted in figure 15. The results for the conventional version of the VQC algorithm can be found in Appendix A.

This performance comparison primarily focuses on a small number of parameters, aligning with the core objective of these adaptive algorithms: minimizing circuit depth and parameter count.

6.1 Analysis of Performance for Two-Qubit Systems

This comparative analysis employs the "Ad Hoc" dataset with the "IQPEmbedding" encoding strategy. The results are obtained using the SPSA optimizer from the "PennyLane" library, comprising 1500 iterations. Although the dataset is relatively straightforward, its substantial sample size introduces complexity, rendering it a suitable choice for assessing the efficiency of quantum algorithms. The utilization of a ro-

bust optimizer, coupled with a generous number of iterations, ensures a comprehensive exploration of the optimization landscape, facilitating a thorough performance evaluation.

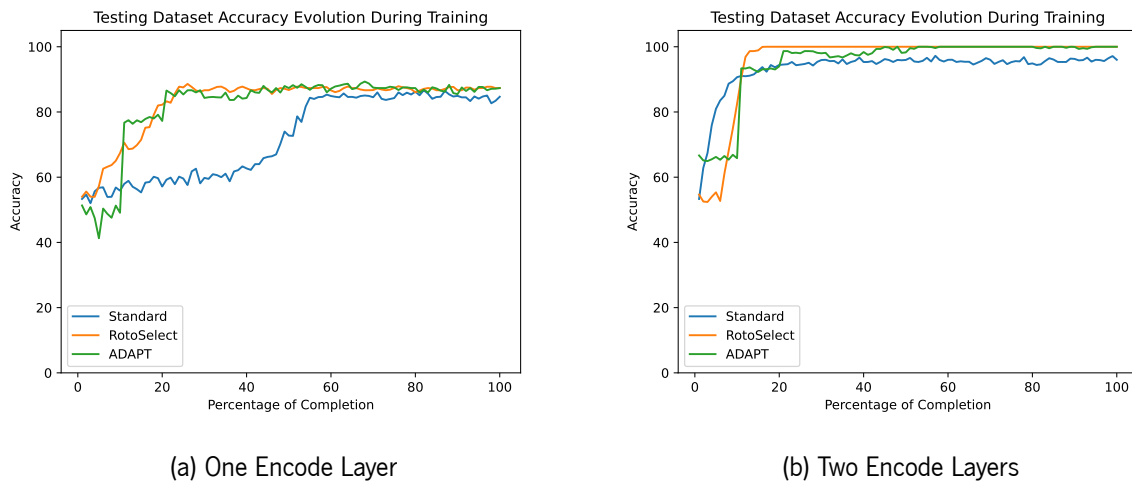


Figure 46: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT. (These curves were sampled)

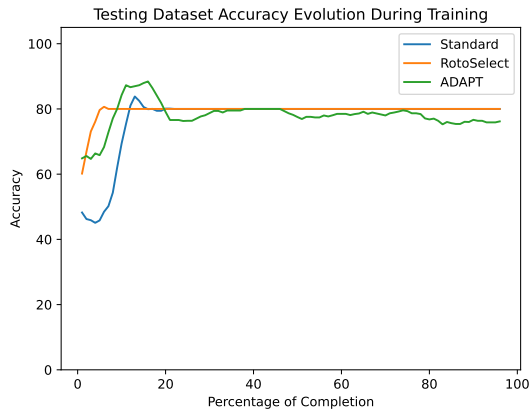
Despite the subtle differences in results, particularly noticeable with two encode layers as depicted in Figure 46b, the standard approach remains the only one incapable of achieving perfect 100% accuracy for two encode layers. This outcome is noteworthy because it suggests that both the ADAPT and RotoSelect algorithms can achieve better performance with fewer parameters, leading to reduced circuit depth. It's worth noting that while the standard and RotoSelect approaches use only eight parameters, the ADAPT approach reaches 100% accuracy at around 80% of the algorithm's completion with ten parameters. This implies that the ADAPT approach can also attain 100% accuracy with just eight parameters.

6.2 Analysis of Performance for Three-Qubit Systems

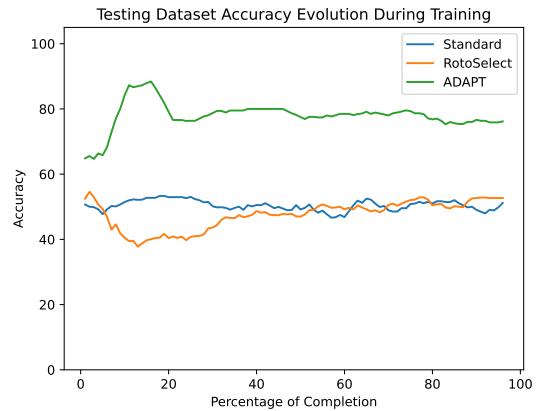
This comparative analysis employs the same dataset as Sections 4.2 and 5.2, along with the "AngleEmbedding" encode strategy utilized in those sections. The results are generated using the SPSA optimizer algorithm, consistent with Section 6.1, spanning a total of 2500 iterations. As a three-qubit circuit is utilized, similar to Section 4.2, distinct entanglement strategies, both linear and full, will be examined, the graphs associated with the standard version can be seen in the Appendix A.

When comparing the results obtained using the structure with the fewest parameters, specifically a single ansatz layer, as shown in figure 47, two valuable insights emerge.

The two key insights arise from this analysis:



(a) Linear

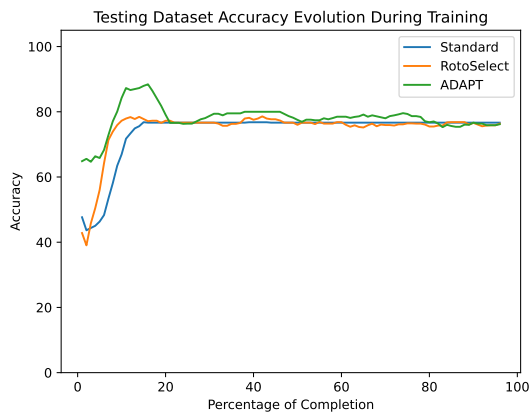


(b) Full

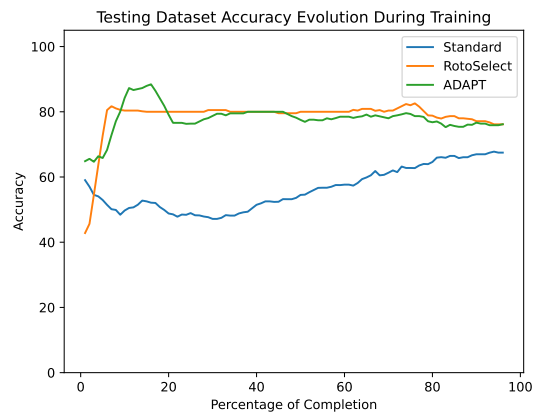
Figure 47: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT for a Single Ansatz Layer (These curves were sampled).

First, in the case of the linear entanglement strategy, both the RotoSelect algorithm and the Standard approach outperform the ADAPT algorithm, despite the latter achieving a higher peak value in accuracy.

Second, when employing the full entanglement strategy, the ADAPT algorithm surpasses the other two approaches. This highlights an advantage of the ADAPT algorithm, where the entanglement is determined by the optimizer rather than being predefined as a hyperparameter.



(a) Linear



(b) Full

Figure 48: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Ansatz Layers (These curves were sampled).

In figure 48, where structures with two ansatz layers are employed, a notable distinction arises among the algorithms, especially concerning the use of 18 parameters. The RotoSelect and the conventional VQC algorithms yield different results for the full entanglement strategy.

Firstly, it becomes evident that the ADAPT algorithm, despite using only ten parameters, can achieve nearly equivalent performance to the other two algorithms when they employ the linear entanglement strategy. However, in the case of full entanglement, both the ADAPT and RotoSelect algorithms outperform the standard VQC version. This outcome is significant as it demonstrates that, with the same structure as the standard VQC, RotoSelect can achieve better results and converge to a higher value more rapidly.

6.3 Analysis of Performance for Four-Qubit Systems

In this analysis, the "Iris" dataset utilized is identical to that used in sections 4.3 and 5.3. The encoding strategy used for this analysis is the "IQPEmbedding", as illustrated in Figure 5, this encoding strategy gives more modest results not picking at 100%. Furthermore, the same optimizer employed previously, SPSA, is used with an iteration count of 2500, this results are displayed in the Appendix A.

The primary variation introduced is the alteration of the entanglement strategy, leading to the results depicted in figure 49.

Comparing the results of the three algorithms in figure 49, it's evident that regardless of the entanglement strategy, the ADAPT algorithm consistently outperforms the standard version of the VQC algorithm. In the case of the linear entanglement strategy, as shown in figure 49a, the RotoSelect algorithm is even capable of achieving a perfect 100% accuracy.

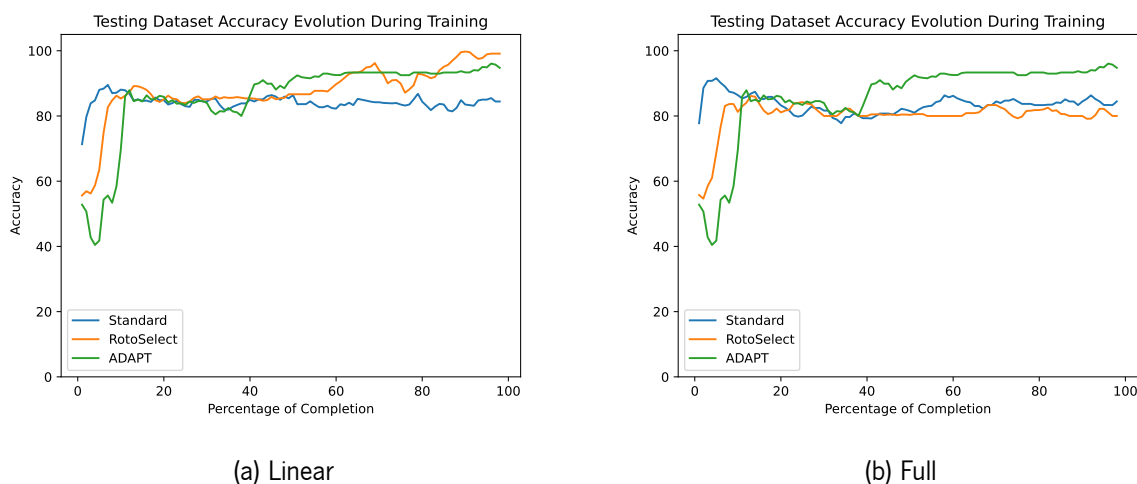


Figure 49: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Single Encode Layer and Single Ansatz Layer (These curves were sampled).

When the number of ansatz layers increases to two, as depicted in figure 50, a noticeable enhancement in performance becomes evident. The adaptive algorithms outperform the Standard version of the VQC

algorithm, particularly in the case of the linear entanglement strategy.

This observation highlights the effectiveness of the adaptive approaches in improving classifier performance. The increased flexibility and adaptability of these algorithms enable them to excel, even in scenarios where the Standard VQC algorithm falls short. These results underscore the potential advantages of incorporating adaptive strategies into quantum machine learning workflows, particularly when dealing with limitations with the circuit depth.

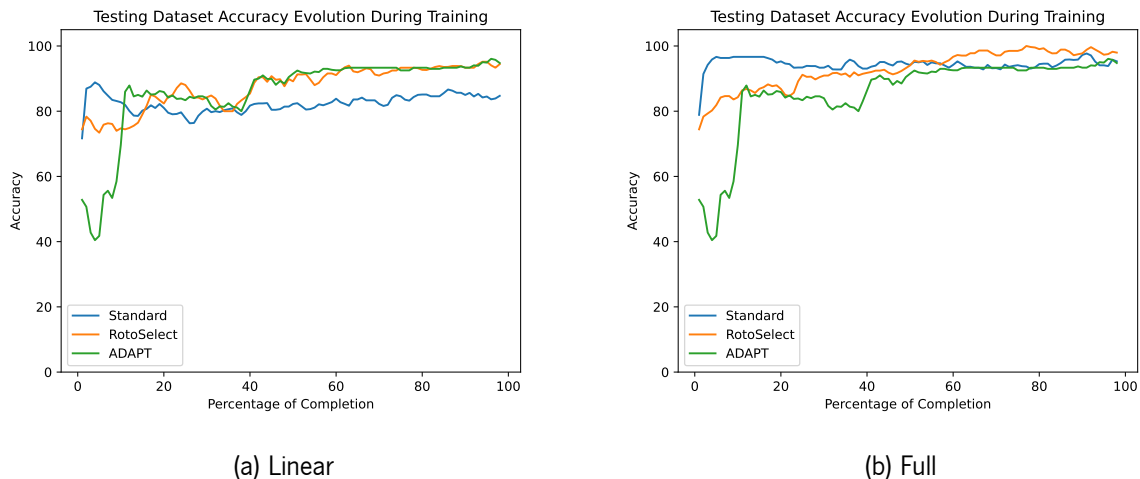
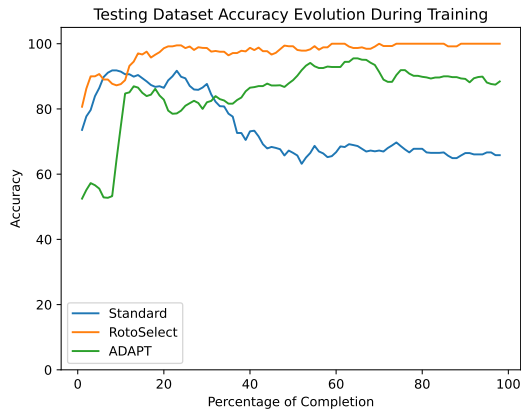


Figure 50: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Single Encode Layer and Two Ansatz Layers (These curves were sampled).

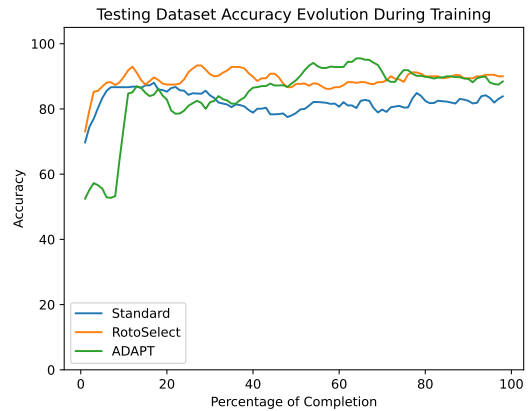
When the number of encode layers increases to two and the number of ansatz layers decreases to one, the results illustrated in figure 51 unveil a distinct pattern. Notably, for the linear entanglement strategy, the adaptive approaches significantly outperform the Standard version of the VQC algorithm.

However, when the number of ansatz layers increases to two, as depicted in figure 52, an interesting shift in performance occurs. In this scenario, both entanglement strategies exhibit the Standard version of the VQC outperforming the adaptive algorithms.

This observation suggests that while adaptive algorithms may demonstrate improved performance in certain contexts, they do not universally surpass the Standard VQC algorithm. Rather, their performance lies in proximity, highlighting their potential as a viable approach for ansatz design. However, further refinement is necessary to optimize their performance for quantum classification tasks. These findings underscore the complexity of quantum machine learning and the need for tailored approaches based on specific problem requirements.

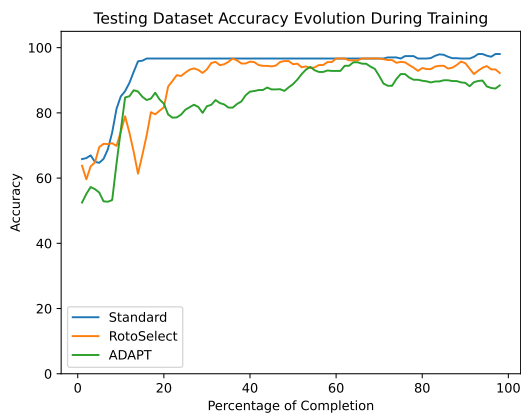


(a) Linear

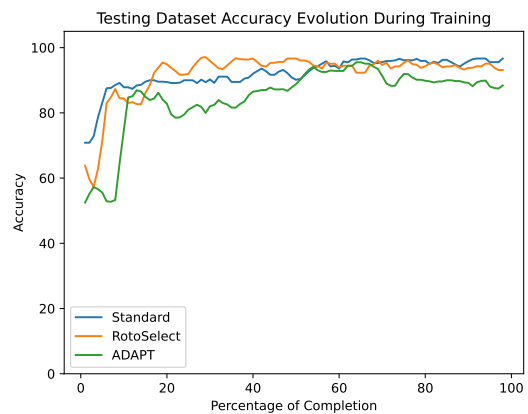


(b) Full

Figure 51: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Encode Layers and One Ansatz Layer (These curves were sampled).



(a) Linear



(b) Full

Figure 52: Comparison of Three Algorithms: Standard, Rotoselect, and ADAPT with Two Encode Layers and Two Ansatz Layers (These curves were sampled).

6.4 Summary

In this section, we provide a summarized comparative analysis of three quantum machine learning algorithms: RotoSelect, ADAPT, and the conventional (Standard) Variational Quantum Classifier (VQC). The main objective is to assess strengths and weaknesses across different datasets and problem dimensions.

The key findings are as follows:

Two-Qubit Systems

- RotoSelect and ADAPT outperformed the Standard VQC, particularly when employing a single en-

code layer;

- Even with fewer parameters, the adaptive algorithms achieved equivalent or superior accuracy compared to the Standard VQC, demonstrating their potential to reduce circuit depth.

Three-Qubit Systems

- The choice of entanglement strategy significantly affected algorithm performance, showing the main advantage of the ADAPT algorithm;
- In scenarios with a single ansatz layer, the Standard VQC outperformed ADAPT for linear entanglement, while ADAPT excelled with full entanglement;
- With two ansatz layers, both adaptive algorithms surpassed the Standard VQC, especially for full entanglement.

Four-Qubit Systems

- The adaptive algorithms consistently outperformed or achieved comparable performance to the Standard VQC, regardless of the entanglement strategy;
- The adaptive algorithms excelled in scenarios with fewer ansatz layers and multiple encode layers, emphasizing their effectiveness in specific contexts.

Overall, the adaptive algorithms demonstrated their potential to improve classifier performance, especially in reducing circuit depth. However, their performance varied depending on problem characteristics, such as the choice of encoding strategy, entanglement strategy, and the number of ansatz and encode layers. These findings highlight the complexity of quantum machine learning and the need for tailored approaches based on specific problem requirements.

Chapter 7

Conclusions and future work

7.1 Conclusions

In this comprehensive exploration of quantum machine learning algorithms, the investigation into the performance and behavior of two adaptive algorithms, ADAPT and RotoSelect, in comparison to the conventional Variational Quantum Classifier (VQC) has yielded several key conclusions and insights:

1. **Promise of Adaptive Algorithms:** Both ADAPT and RotoSelect algorithms have demonstrated potential in adapting and optimizing quantum circuits for classification tasks. They exhibit competitive performance across diverse experiments, highlighting their viability in quantum machine learning;
2. **Hyperparameter Sensitivity:** The performance of adaptive algorithms is notably sensitive to hyperparameter settings, including the number of encode repetitions and ansatz repetitions. Fine-tuning these parameters proves crucial for achieving optimal results;
3. **Encoding Strategy Impact:** The selection of an encoding strategy significantly influences algorithm performance. While angle-based encoding generally proves effective, the performance of the "IQPEmbedding" encoding varies depending on the dataset and algorithm. Understanding dataset characteristics is essential for choosing the appropriate encoding strategy;
4. **Relevance of Entanglement Strategies:** The choice of entanglement strategy within an ansatz circuit has a substantial impact on the performance of adaptive algorithms. Different strategies, such as linear and full entanglement, exhibit distinct effects on algorithm convergence and accuracy;
5. **Reduced Circuit Complexity:** Adaptive algorithms excel in delivering accurate results with reduced circuit depth and fewer parameters compared to the conventional VQC. This aligns with their primary objective of compressing quantum circuits;

6. **Competitive Performance:** Adaptive algorithms frequently outperform the conventional VQC, particularly in scenarios featuring linear entanglement. However, under specific conditions, the Standard VQC algorithm may demonstrate superior performance;
7. **Complexity of Quantum Machine Learning:** The findings underscore the intricate nature of quantum machine learning. There is no one-size-fits-all algorithm; instead, the approach must be tailored to the specific problem;
8. **Potential for Hybrid Approaches:** Combining the strengths of both adaptive algorithms may pave the way for hybrid algorithms that leverage both approaches, potentially enhancing overall performance;
9. **Future Research Directions:** Further research is warranted to refine and optimize adaptive algorithms for quantum classification tasks. Exploring additional datasets and problem domains will provide a more comprehensive understanding of their capabilities and limitations.

In summary, this study contributes some insights to the field of adaptive approaches for quantum machine learning. It underscores the potential of adaptive algorithms while emphasizing the significance of meticulous hyperparameter tuning and the selection of an appropriate encoding strategy. As quantum computing technology advances, these algorithms may play a pivotal role in harnessing quantum computers' power for practical machine learning applications.

7.2 Prospect for future work

As the field of quantum machine learning continues to evolve, there remains a significant scope for enhancing the performance and versatility of adaptive quantum algorithms. In this section, we outline potential avenues for future research and development, focusing on two promising algorithms: RotoSelect and ADAPT. These algorithms have shown promise in simplifying quantum circuit designs and improving classification accuracy, but further advancements are needed to fully unlock their potential.

RotoSelect:

1. **Circuit Structure Variations:** An intriguing direction for future research involves exploring innovative methods to introduce greater flexibility into the circuit structure. By developing techniques that enable dynamic modifications to the circuit topology during the optimization process, researchers could adapt the quantum circuit to the complexity of specific problems, potentially reducing the need for manual hyperparameter tuning;

2. **Dynamic Entanglement Strategies:** Exploring the adaptability of entanglement strategies represents a pivotal area for future research. Developing strategies that dynamically adjust the level of entanglement based on specific problem characteristics could lead to the creation of more efficient quantum circuits suitable for a broader spectrum of problems. Notably, algorithms like ADAPT and VAns already possess the capacity to introduce entanglement, and in the case of the VAns algorithm, even remove it. This adaptability paves the way for versatile and problem-tailored quantum solutions [1, 7].
3. **Gates' Limitations:** One of the most significant drawbacks of the Rotoselect algorithm is its reliance on specific entanglement strategies. To enhance this algorithm and enable it to tailor ansatz structures more effectively for individual problems, granting it greater control over circuit entanglement is essential;
4. **Advanced Cost Function Optimization:** To further enhance the algorithm's efficiency, researchers could delve into advanced approaches for optimizing the cost function directly. By developing quantum-native optimization techniques tailored to the specific cost functions encountered in quantum machine learning, it may be possible to achieve more accurate and reliable results.

ADAPT:

1. **Operator Generation Methods:** Innovations in operator generation are crucial for ADAPT's continued development. Researchers can investigate techniques that dynamically construct operators based on the current quantum state, training data, or optimization progress. These approaches have the potential to create more efficient and expressive operator sets;
2. **Efficient Operator Pools:** An intriguing avenue for future research involves expanding the scope of operators employed by ADAPT. By introducing diverse pools of operators, such as higher-order or custom-defined operators, researchers can bolster the algorithm's capacity for expression and classification. This enhancement enables the algorithm to tackle an even broader spectrum of problems. Additionally, exploring optimized methods for selecting these operators is imperative, especially when dealing with large-scale systems, as exhaustive searches are computationally impractical.
3. **Improved Parameter Optimization:** The optimization of parameters is a critical aspect of ADAPT. Future work should focus on identifying and implementing more robust and efficient opti-

mization algorithms. These enhancements aim to reduce the parameter optimization fluctuations observed in previous results, ensuring more stable convergence and improved accuracy;

4. **Ansatz Customization:** To tailor the ansatz more effectively, research efforts can be directed toward methods that automatically remove unnecessary gates or adjust the circuit structure during the optimization process. These adaptive strategies seek to strike a balance between performance and resource efficiency.

Numerous opportunities for enhancement remain, particularly in harnessing the adaptive capabilities to construct the encoding block. This is significant as the encoding block's role in shaping the classification algorithms' performance has become increasingly evident.

In summary, the future work outlined here underscores the commitment to advancing the capabilities of RotoSelect and ADAPT. These algorithms hold some promise in the quantum machine learning landscape, and ongoing research and development efforts are poised to unlock their full potential. As quantum computing technology matures, these enhancements can play a pivotal role in harnessing its power to solve complex real-world problems.

Bibliography

- [1] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational Quantum Algorithms. *Nat Rev Phys*, 3(9):625–644, September 2021. arXiv:2012.09265 [quant-ph, stat].
- [2] Angus Lowe. Quantum circuit structure learning. https://pennylane.ai/qml/demos/tutorial_rotoselect, 09 2019. Date Accessed: 2023-05-11.
- [3] Alex Wilkins. Quantum computers proved to have ‘quantum advantage’ on some tasks, 2022.
- [4] Shor IBM. Shor’s algorithm, 2022.
- [5] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, September 2017. Number: 7671 Publisher: Nature Publishing Group.
- [6] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018. arXiv:1801.00862 [cond-mat, physics:quant-ph].
- [7] Harper R. Grimsley, George S. Barron, Edwin Barnes, Sophia E. Economou, and Nicholas J. Mayhall. ADAPT-VQE is insensitive to rough parameter landscapes and barren plateaus, April 2022. arXiv:2204.07179 [physics, physics:quant-ph].
- [8] PennyLane. Variational circuits – PennyLane, 2022.
- [9] Ying Li and Simon C. Benjamin. Efficient Variational Quantum Simulator Incorporating Active Error Minimization. *Phys. Rev. X*, 7(2):021050, June 2017. Publisher: American Physical Society.
- [10] QAOA Qiskit. Solving combinatorial optimization problems using QAOA.

- [11] Danyal Maheshwari, Daniel Sierra-Sosa, and Begonya Garcia-Zapirain. Variational Quantum Classifier for Binary Classification: Real vs Synthetic Dataset. *IEEE Access*, 10:3705–3715, 2022. Conference Name: IEEE Access.
- [12] M. Bilkis, M. Cerezo, Guillaume Verdon, Patrick J. Coles, and Lukasz Cincio. A semi-agnostic ansatz with variable structure for quantum machine learning, January 2022. arXiv:2103.06712 [quant-ph, stat].
- [13] Ying Fang, Zhaofei Yu, and Feng Chen. Noise Helps Optimization Escape From Saddle Points in the Synaptic Plasticity. *Front Neurosci*, 14:343, April 2020.
- [14] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, September 2017. Number: 7671 Publisher: Nature Publishing Group.
- [15] Yuxuan Du, Zhuozhuo Tu, Xiao Yuan, and Dacheng Tao. Efficient measure for the expressivity of variational quantum algorithms. *Phys. Rev. Lett.*, 128(8):080506, February 2022. arXiv:2104.09961 [quant-ph].
- [16] Martin Larocca, Nathan Ju, Diego García-Martín, Patrick J. Coles, and M. Cerezo. Theory of over-parametrization in quantum neural networks, September 2021. arXiv:2109.11676 [quant-ph, stat].
- [17] Yanzhu Chen, Linghua Zhu, Chenxu Liu, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. How Much Entanglement Do Quantum Optimization Algorithms Require?, May 2022. arXiv:2205.12283 [quant-ph].
- [18] Ho Lun Tang, V.O. Shkolnikov, George S. Barron, Harper R. Grimsley, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansatzes on a Quantum Processor. *PRX Quantum*, 2(2):020310, April 2021. Publisher: American Physical Society.
- [19] Panagiotis G. Anastasiou, Yanzhu Chen, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. TETRIS-ADAPT-VQE: An adaptive algorithm that yields shallower, denser circuit ansatzes, September 2022. arXiv:2209.10562 [quant-ph].

- [20] Linghua Zhu, Ho Lun Tang, George S. Barron, F. A. Calderon-Vargas, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer, July 2022. arXiv:2005.10258 [quant-ph].
- [21] Jiahao Yao, Haoya Li, Marin Bukov, Lin Lin, and Lexing Ying. Monte Carlo Tree Search based Hybrid Optimization of Variational Quantum Circuits. In *Proceedings of Mathematical and Scientific Machine Learning*, pages 49–64. PMLR, September 2022. ISSN: 2640-3498.
- [22] VQE Qiskit. Simulating Molecules using VQE.
- [23] Mark Lewis and Fred Glover. Quadratic Unconstrained Binary Optimization Problem Preprocessing: Theory and Empirical Analysis, May 2017. arXiv:1705.09844 [cs].
- [24] Mateusz Ostaszewski, Edward Grant, and Marcello Benedetti. Structure optimization for parameterized quantum circuits, January 2021. arXiv:1905.09692 [quant-ph].
- [25] Hugh G. A. Burton, Daniel Marti-Dafcik, David P. Tew, and David J. Wales. Exact electronic states with shallow quantum circuits through global optimisation, June 2022. arXiv:2207.00085 [physics, physics:quant-ph].
- [26] Quantum circuit structure learning — PennyLane, 2022.
- [27] Qiskit Development Team. Qiskit machine learning ad hoc data, 2018. Date Accessed: 2023-01-24.
- [28] David A. Meyer and Nolan R. Wallach. Global entanglement in multiparticle systems. *Journal of Mathematical Physics*, 43(9):4273–4278, aug 2002.
- [29] M. Schuld and F. Petruccione. *Machine Learning with Quantum Computers*. Quantum Science and Technology. Springer International Publishing, 2021.
- [30] J.C. Spall. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):817–823, 1998.

Appendix A

Support work

A.1 Rotoselect

A.1.1 Four-Qubit Systems AngleEmbedding Encode

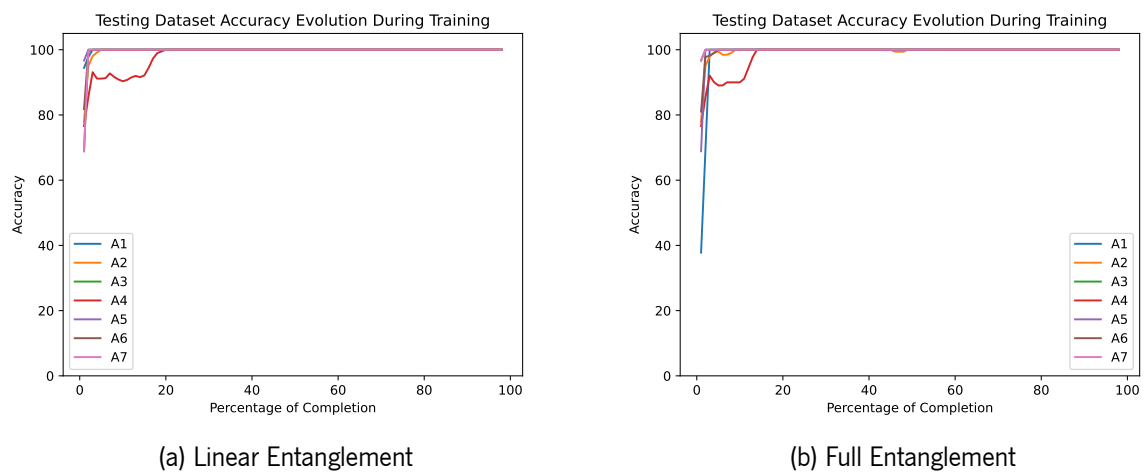


Figure 53: comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7), with normalized curves, using angle-based encoding.

Both Figure 53a and 53b clearly demonstrate the algorithm's ability to rapidly achieve 100% accuracy shortly after initialization, regardless of the number of parameters. This outcome underscores the algorithm's proficiency in achieving high performance with shallower ansatz structures, aligning with its objective of minimizing ansatz depth. Notably, the entanglement strategy appears to have minimal impact on the algorithm's performance.

A.2 ADAPT

A.2.1 Four-Qubit Systems AngleEmbedding Encode

As depicted in figure 54a, the ADAPT algorithm effectively resolves the problem using a minimal parameter count of only three, achieving a perfect accuracy rate of 100% for both the training and testing datasets.

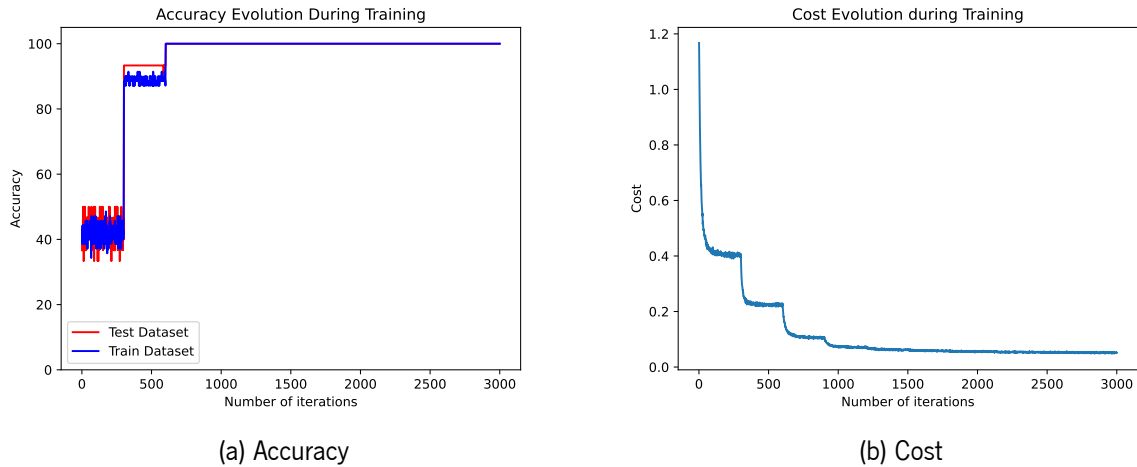


Figure 54: Quantum Circuit Architecture: Angle Based Encoding, Ten Parameterized Operators and Four Qubits

A.3 Standard VQC

A.3.1 Two-Qubit Systems

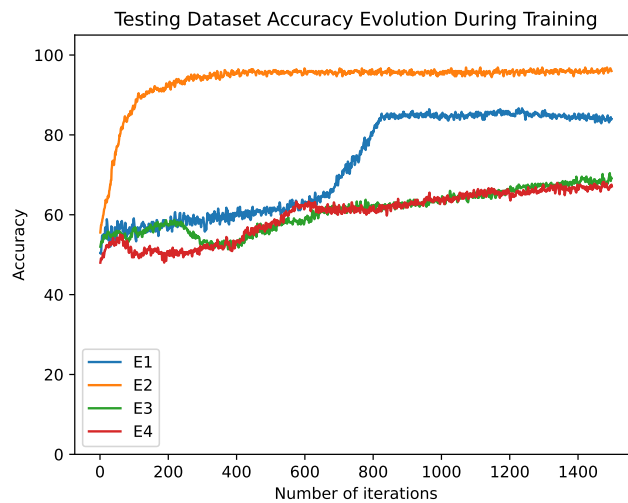


Figure 55: Comparison of Quantum Circuits with Different Numbers of Encode Layers (E1-E4) and One Ansatz Layer (Smoothed Curves).

In figure 55, the algorithm’s optimal performance is apparent when employing two encode layers. This observation aligns with the findings for both the Rotoselect algorithm in figure 14 and the ADAPT algorithm in Figure 41. It underscores the significant impact of a well-designed encode strategy on performance, irrespective of the algorithm employed.

The impact of varying the number of parameters is almost imperceptible across most approaches. However, it becomes more pronounced when considering the approach with fewer parameters, specifi-

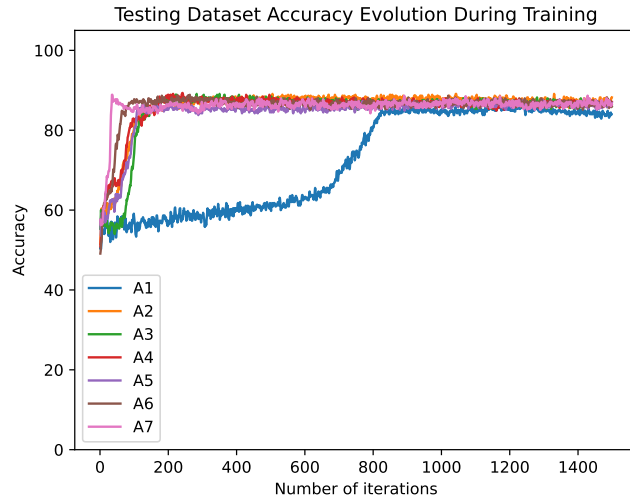


Figure 56: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and a Single Encode Layer (Smoothed Curves).



Figure 57: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and Two Encode Layers (Smoothed Curves).

cally when using a single ansatz layer, which consistently produces comparatively inferior results. This observation stands in contrast to the Rotoselect algorithm results shown in figures 16 and 17, where the differences in results for various parameter counts were less noticeable. In this case, it appears that using a lower parameter count, specifically eight parameters, leads to suboptimal outcomes. The extent of this difference becomes more evident when examining figure 46.

A.3.2 Three-Qubit Systems

The curves in figure 58 have been smoothed for clarity.

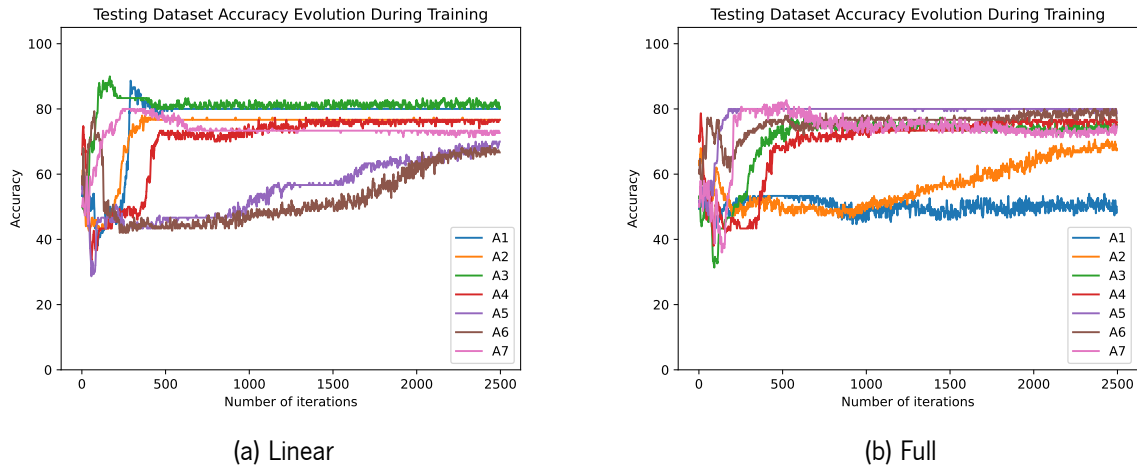


Figure 58: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) Using Conventional VQC.

In figure 58, the significant impact of the entanglement strategy on performance is evident, mirroring observations made in the RotoSelect algorithm.

A.3.3 Four-Qubit Systems



Figure 59: Comparison of Quantum Circuits with Different Numbers of Encode Layers (E1-E4) and a Single Ansatz Layer (Smoothed Curves).

As evident in the results, the number of encoding layers exerts a significant influence on performance, particularly because, in certain cases, the algorithm’s performance experiences substantial declines during the optimization process.

As evident in figure 60a, circuits with fewer gates and parameters yield comparatively poorer results. This scenario highlights the type of problem that adaptive approaches aim to address, emphasizing the creation of effective classifiers with minimal gate and parameter usage. In figure 60b, it becomes clear

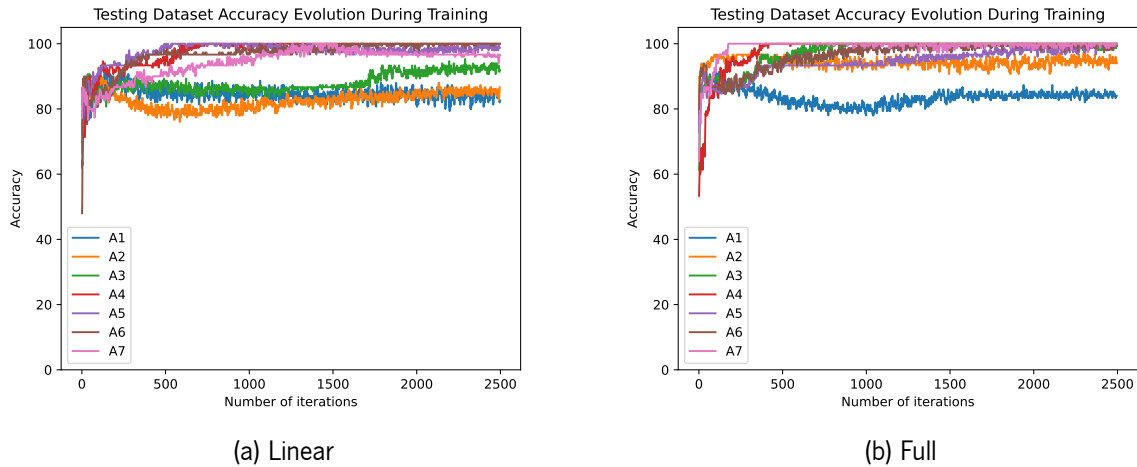


Figure 60: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and a Single Encode Layer (Smoothed Curves).

that circuits employing only one ansatz layer produce suboptimal results when compared to those utilizing two or more ansatz layers.

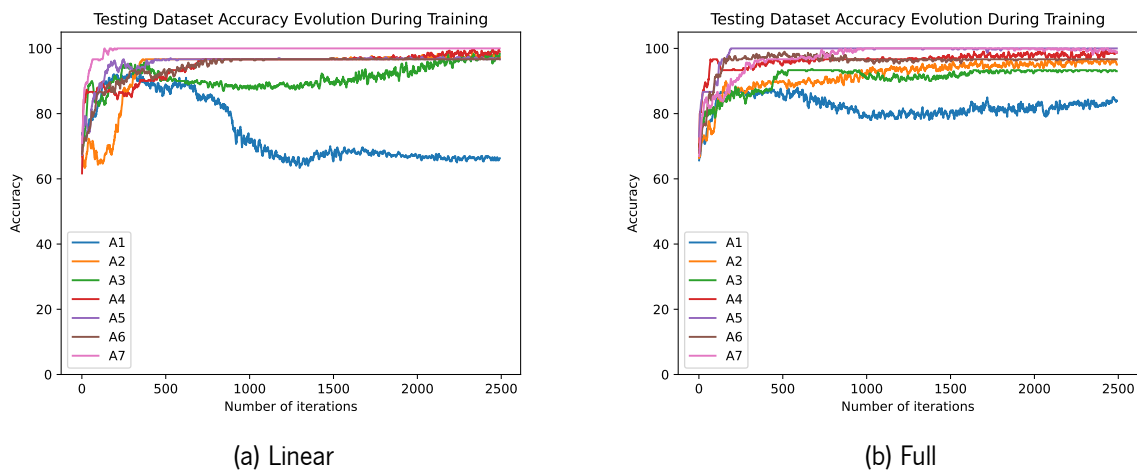


Figure 61: Comparison of Quantum Circuits with Different Numbers of Ansatz Layers (A1-A7) and Two Encode Layers (Smoothed Curves).

Increasing the number of encoding layers to two, as shown in figure 61, led to an improvement in some results. However, for the linear entanglement strategy in figure 61a, one ansatz layer resulted in a significant decline in classifier performance.

A.4 Comparative Analysis: Circuits

This section is dedicated to showcasing the final circuits generated by the three algorithms examined in this dissertation.

A.4.1 Circuit Analysis: Two Qubits

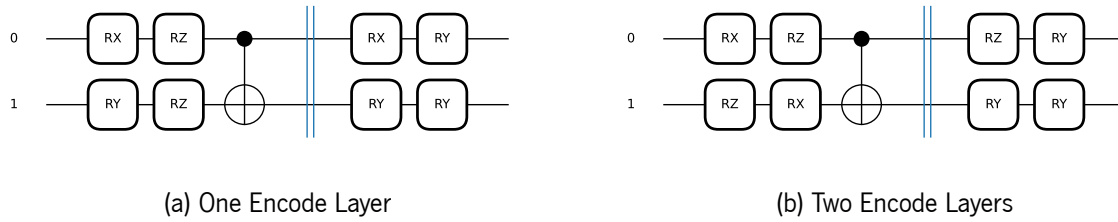


Figure 62: Ansatzes Generated by the Rotoselect Algorithm for a Single Ansatz Layer

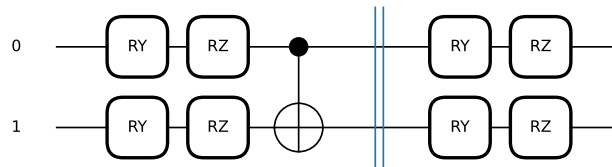
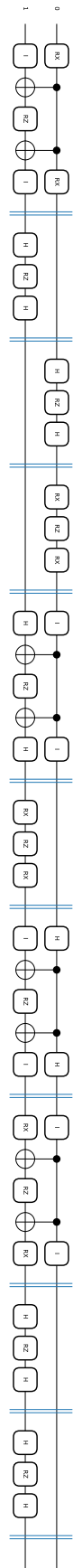
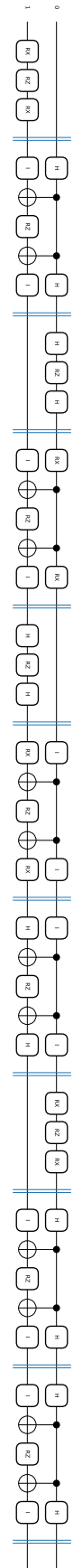


Figure 63: Ansatz used by the Standard VQC version for a Single Ansatz Layer



(a) One Encode Layer



(b) Two Encode Layers

Figure 64: Antazes Generated by the ADAPT Algorithm

A.4.2 Circuit Analysis: Three Qubits

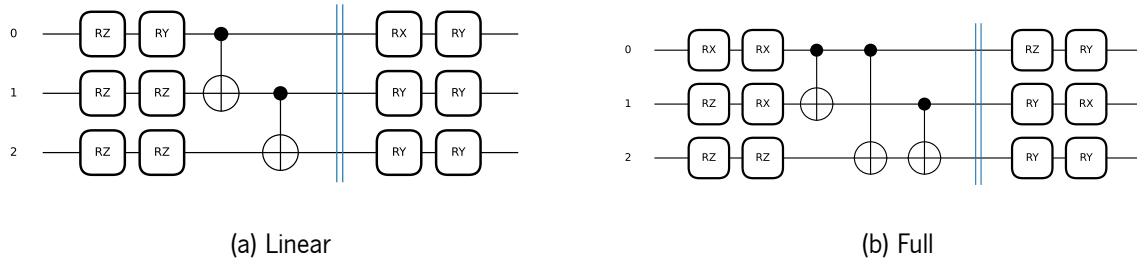


Figure 65: Antazes Generated by the Rotoselect Algorithm for a Single Ansatz Layer with Different Entanglement Strategies

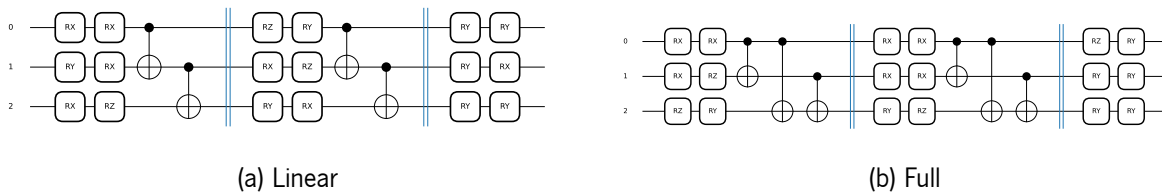


Figure 66: Antazes Generated by the Rotoselect Algorithm for Two Ansatz Layers with Different Entanglement Strategies

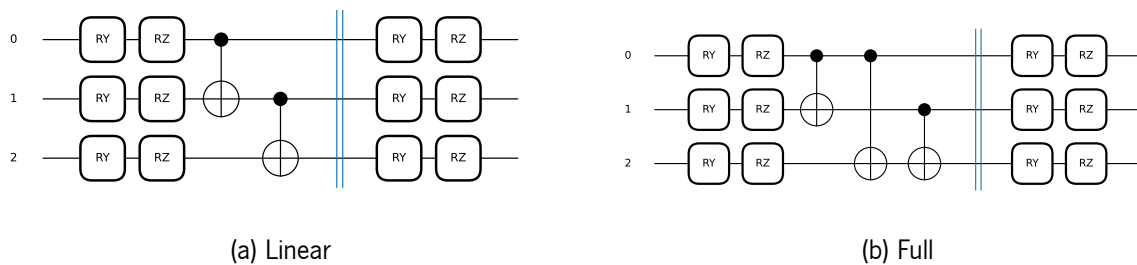


Figure 67: Anstaz used by the Standard VQC version for a Single Encode and Anstaz Layer with Different Entanglement Strategies

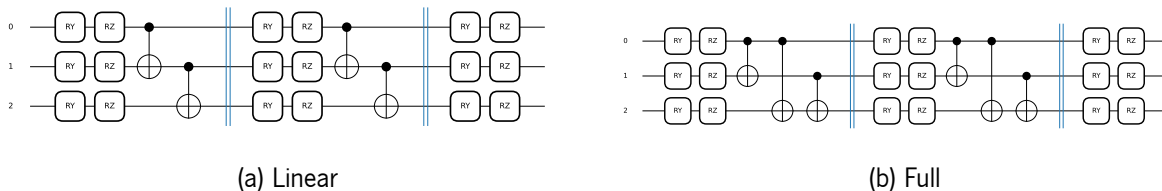


Figure 68: Anstaz used by the Standard VQC version for a Single Encode and Two Anstaz Layers with Different Entanglement Strategies

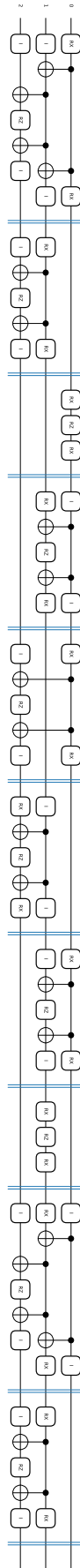


Figure 69: Antazes Generated by the ADAPT Algorithm

A.4.3 Circuit Analysis: Four Qubits

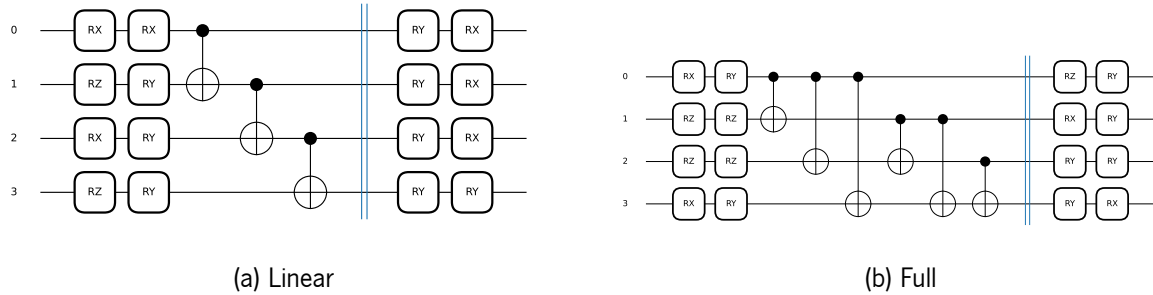


Figure 70: Antazes Generated by the Rotoselect Algorithm for a Single Encode and Anstaz Layer with Different Entanglement Strategies

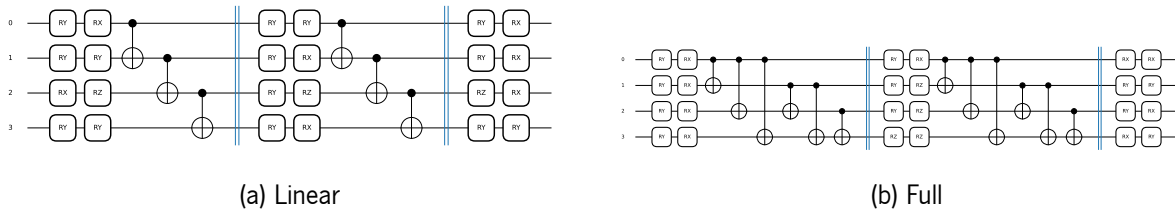


Figure 71: Antazes Generated by the Rotoselect Algorithm for a Single Encode Layer and Two Anstaz Layers with Different Entanglement Strategies

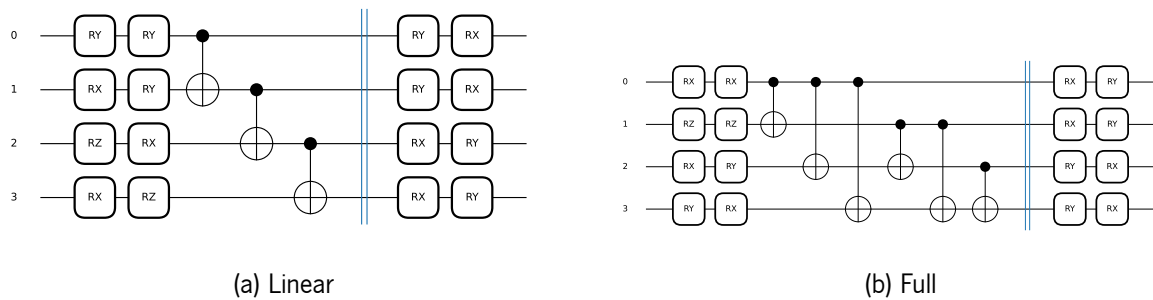


Figure 72: Antazes Generated by the Rotoselect Algorithm for Two Encode and a Single Anstaz Layer with Different Entanglement Strategies

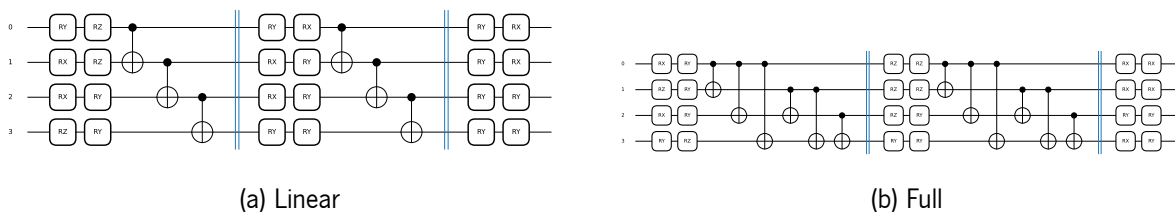


Figure 73: Antazes Generated by the Rotoselect Algorithm for Two Encode Layers and Two Anstaz Layers with Different Entanglement Strategies

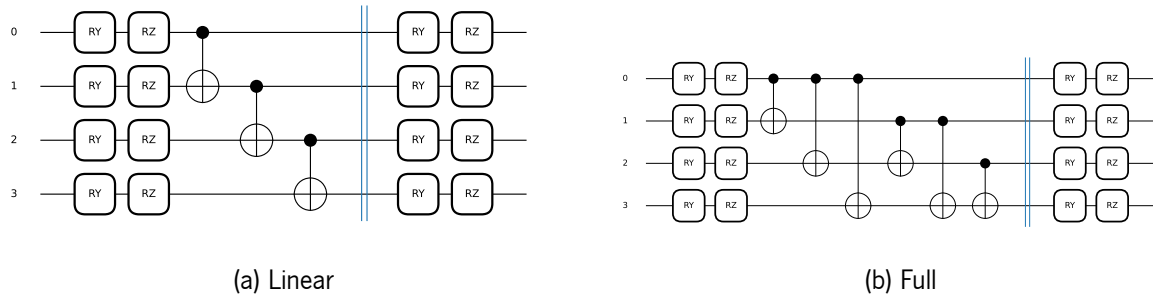


Figure 74: Anstaz used by the Standard VQC version for a Single Encode and Anstaz Layer with Different Entanglement Strategies

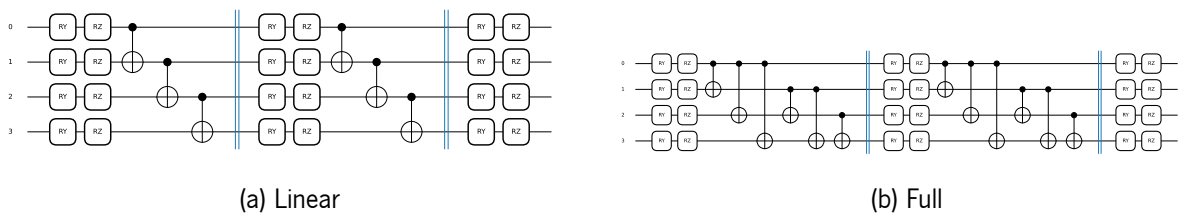
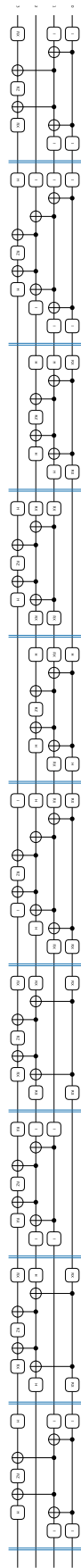
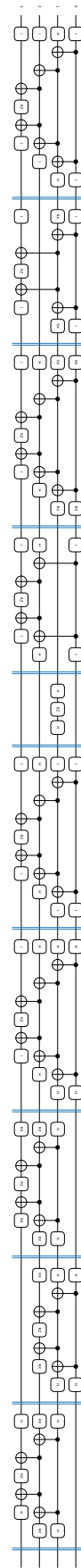


Figure 75: Anstaz used by the Standard VQC version for a Single Encode and Two Anstaz Layers with Different Entanglement Strategies



(a) One Encode Layer



(b) Two Encode Layers

Figure 76: Antazes Generated by the ADAPT Algorithm

