Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea Magistrale in Matematica

Investigating barren plateaus in variational quantum algorithms via the Clifford group

Relatore:
Chiar.mo Prof.
De Palma Giacomo

Correlatore:
Chiar.mo Prof.
Pastorello Davide

Presentata da:
Giovannini Elena

Anno Accademico 2024-2025

Abstract

Quantum computing is a model of computing that is based on the laws of quantum mechanics. A promising path toward demonstrating quantum advantage in the near term is the use of Variational Quantum Algorithms (VQAs). VQAs are quantum algorithms that rely on a set of tunable parameters, the optimization of which is handled by a classical computer, creating a hybrid quantum-classical loop. The goal of a VQA is to find the set of parameters that minimizes a cost function that depends on them. The potential of these circuits is often hampered by the Barren Plateau (BP) phenomenon, which indicates that the cost function results being flat in vast regions of the parameter space far from the minimum, rendering the training of the VQA practically infeasible. The goal of this thesis is to relate the flatness of the cost function to the architectural features of a quantum circuit. This is the same problem addressed in [Napp, arXiv:2203.06174], however, we approach the problem in a different manner. To this aim we make use of the Pauli group, i.e. the group generated by the Pauli matrices along with the identity, and its normalizer, the Clifford group. We study the problem in two different configurations. In the first configuration, we assume that the architecture of the circuit is not fixed, as each gate is applied to randomly sampled sites at each step. We study this model with the properties of Markov chains and we derive an analytical upper bound on the expected magnitude of the gradient of the cost function, which decays exponentially with the number of gates in the circuit. We also numerically demonstrate that the bound does not exhibit exponential decay for shallow circuits. In the second configuration the architecture is fixed and we study the model as a random walk. For the same quantity we obtain an analytical lower bound that exponentially decays with the minimum number of gates that a trajectory of the random walk must pass trough.

Abstract

La computazione quantistica è un modello di computazone basato sulle leggi della meccanica quantistica. Un approccio promettente per dimostrare la superiorità dei computer quantistici rispetto a quelli classici, è rappresentato dagli algoritmi quantistici variazionali (VQA, Variational Quantum Algorithms). Questi sono algoritmi quantistici che si basano su un insieme di parametri regolabili, la cui ottimizzazione è affidata ad un computer classico, creando così un ciclo ibrido classico-quantistico. Lo scopo di un VQA è quello di trovare il set di parametri che minimizza una funzione di costo che dipende dai parametri stessi. Nonostante le incoraggianti premesse, l'ottimizzazione di questi circuiti è spesso ostacolata dal fenomeno del barren plateau (BP), che comporta l'appiattimento della funzione di costo in estese aree dello spazio dei parametri, lontano dal minimo. Quando presente, questa circostanza rende l'addestramento di un VQA computazionalmente proibitivo. Lo scopo di questa tesi è quello di mettere in relazione l'insorgenza di questo fenomeno con le caratteristiche del circuito quantistico utilizzato. Pur riferendosi allo stesso problema discusso in [Napp, arXiv:2203.06174], questo lavoro propone un approccio alternativo. A tal fine, si fa uso del gruppo di Pauli, ovvero il gruppo generato dalle matrici di Pauli insieme all'identità, e del suo normalizzatore, il gruppo di Clifford. Studiamo il problema, per un circuito ad n qubit, in due configurazioni distinte. Nella prima, assumiamo che l'architettura del circuito, ovvero la posizione delle porte, non sia fissata e che quindi, i siti su cui queste porte agiscono siano campionati ogni volta dalla distribuzione uniforme sulle possibili coppie di siti. Studiamo questo modello con le proprietà delle catene di Markov e ricaviamo un limite superiore sulla norma attesa del gradiente della funzione di costo, che decade esponenzialmente con il numero di qubit nel circuito. Inoltre, dimostriamo numericamente che questo limite non presenta decrescita esponenziale per circuiti poco profondi. Nella seconda configurazione invece, assumiamo che l'architettura del circuito sia fissata e studiamo il problema come una passeggiata aleatoria. In questo caso, ricaviamo un limite inferiore sulla norma attesa del gradiente, che decade esponenzialmente con il numero minimo di porte che una traiettoria della passeggiata stocastica deve attraversare.

Contents

Chapter 1

Introduction

1.1 Quantum computing

The idea of creating a computer that exploits the laws of quantum mechanics to perform computations was first introduced by Richard Feynman in 1981, during a conference on the topic "Simulating Physics with Computers". His talk was later developed into an article [1]. In his speech, Feynman addressed the problem of simulating quantum systems, highlighting that classical computers are inadequate for this task. Indeed, the space in which quantum systems are defined, the Hilbert space, grows exponentially with the number of components in the system, leading to prohibitively high computational complexity, unapproachable for classical computers. The core of his argument can be summarized in the famous final statement of the speech, which claims that since nature is not classical, any efficient simulation of it must be based on the principles of quantum mechanics.

Feynman was not the only one concerned with the computational limits of classical computers. This conclusion was reached independently [2], and slightly earlier, in 1980, by Yuri Manin. In his book [3] he discusses the exponential cost of simulating a many-particle particle system with a classical computer due to the higher complexity of quantum systems with respect to their classical counterparts. A few years before, Paul Benioff proposed a quantum mechanical model of the Turing machine[1], in other words, he showed that for any Turing machine Q, one can construct a Hamiltonian and a suitable class of initial states such that the time evolution of these states under the Hamiltonian corresponds

[1] A Turing machine is a theoretical computational model consisting of an infinite tape divided into cells, a tape head that can read and write symbols and move left or right, and a finite set of internal states. It processes input according to a set of rules and serves as a formal definition of algorithmic computation.

to the computation steps of Q [4]. These intuitions, helped give rise to the field of quantum computing.

1.1.1 Proving quantum advantage

In the following decades, the field began to develop and quantum computers, as devices that exploits the laws of quantum mechanics, started to be studied. The model that describes how these machines perform calculations is the quantum circuit. This consists of an ordered sequence of quantum gates, mathematically represented as unitary operators, acting on qubits, the fundamental units of information in a quantum computer. The notion of quantum circuit was formalized in 1985, by David Deutsch [5]. He also proposed, in the same paper, a quantum generalization of the Church–Turing hypothesis. The Church-Turing hypothesis states that

"*Every function which would be naturally regarded as computable can can be computed by the universal Turing machine* [2]."

In other words, the thesis states that the class of computable functions coincides with the class of functions that can be computed by a Turing machine. Deutsch realized that, since any computation is inherently a physical process, it was possible to deduce the CT hypothesis from the law of physics. The hypothesis, revisited, states that *every physical process can be simulated by a universal computing device*, and Deutsch himself found that quantum theory and the quantum computer are compatible with the principle.

A few years later, in 1993, Umesh Vazirani and Ethan Bernstein formulated the the Bernstein-Vazirani Algorithm [6], that proves a super-polynomial advantage of quantum computing over classical computers[3]. Later in 1994, Daniel Simon proved that a quantum computer guarantees an exponential speedup in solving the Simon's problem, an idealized version of the problem of finding the period of function [7].

Simon's algorithm later inspired Peter Shor, who developed a quantum algorithm to efficiently solve the discrete logarithm problem, and subsequently formulated the famous Shor's algorithm for factoring large numbers. Both problems are believed to be hard for classical computers and belong to the class of problems known as NP[4]. These results drew significant attention to quantum computing due to their practical implications. Indeed, Shor's algorithm [8]

[2] A universal Turing machine is a Turing machine that can simulate any other Turing machine.

[3] With super-polynomial advantage of quantum computers we mean that a quantum computer can solve a certain problem in polynomial time as a function of the input size, while a classical computer requires a super-polynomial time to solve the same instance.

[4] NP (Nondeterministic Polynomial time) is the class of decision problems for which the

provides an exponential speedup over the best-known classical algorithms for factoring, threatening the security of widely used encryption methods. Further evidence of the power of quantum computers emerged in 1995, when Lov Grover demonstrated that a quantum computer could achieve a quadratic speedup for searching through an unstructured search space. This result became known as Grover's search algorithm [9] and can be applied to a wide range of problems.

These results demonstrated that quantum computers could, in principle, be used to tackle problems that are believed to be intractable for classical computers, as the problems in the NP class. This naturally raises the question of whether all *classically hard* problems can be solved, or at least improved, using quantum algorithms. Currently, it is believed that the class of problems that are efficiently solvable by a quantum computer, BQP[5], does not contain the class NP, meaning that, apart from certain specific problems, quantum computers are not expected to solve all computationally hard problems in polynomial time. Indeed, Grover's search algorithm, which can be applied to NP-complete problems (those problems for which a solution would imply efficient solutions to all other problems in the NP class) offers at most a quadratic speedup rather than an exponential one. At the current state, what is guaranteed is that BQP contains the class of problems solvable by probabilistic Turing machines, BPP[6]. This is formalized by the inclusion BPP $\subseteq$ BQP, proven in [6]. The result implies that quantum computation generalizes probabilistic computation, offering at least the same computational power and potentially more.

Underlying the belief in the superior power of quantum computing is the quantum nature of the *qubit*, the analogous of the classical bit. Unlike bits, which can be either 0 or 1, a qubit can exist in a superposition of both states simultaneously. Even more crucially, systems composed of multiple qubits can be *entangled*, exhibiting correlations that have no classical counterpart. Superposition and entanglement allow quantum systems to process information in ways that classical computers fundamentally cannot.

However, despite the growing enthusiasm, quantum computers are not yet practically useful. The algorithms we discussed, typically require a highly idealized model of a quantum computer, where the quantum state evolves in a perfectly closed and isolated system. If this is not the case, whenever the quantum system interacts with its external environment, uncontrollable disturbances arise, manifesting as noise within the quantum circuit. This phenomenon, known as *decoherence*, implies that, in order to reliably store and process quantum infor-

problem instances, where the answer is yes, have proofs verifiable in polynomial time by a deterministic Turing machine.

[5]BQP (Bounded-Error Quantum Polynomial time) is the class of decision problems solvable by a quantum computer in polynomial time, with an error probability of at most 1/3 for all instances.

[6]BPP (Bounded-Error Probabilistic Polynomial time) is the class of decision problems solvable by a probabilistic Turing machine in polynomial time with an error probability bounded by 1/3 for all instances.

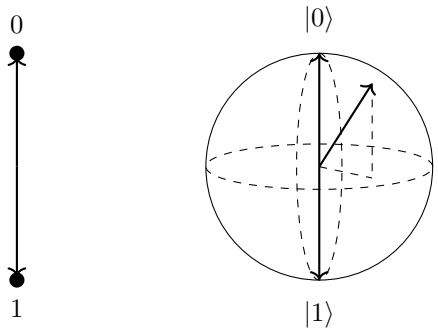


Figure 1.1: **Comparison between bits and qubits**. Bits (on the left) can assume a given value that can only be either 0 or 1. A qubit (on the right) instead, can assume values that are superposition of the states $|0\rangle$ and $|1\rangle$, the quantum equivalents to 0 and 1.

mation, the system must remain nearly perfectly isolated. At the same time, for a quantum computer to be useful, we must be able to control the system externally and read out the qubits to extract the computational result. Satisfying both conditions simultaneously makes building a functional quantum computer an extremely challenging task.

Nevertheless, it is theoretically possible to mitigate the effects of noise without compromising quantum information processing. This is achieved through the use of quantum error correction (QEC) techniques, first introduced by Shor [10]. The idea behind QEC is to protect quantum information by encoding it into highly entangled states of multiple physical qubits, such that the information can be preserved and recovered even in the presence of errors affecting individual qubits. Unfortunately, such methods come at a high cost in terms of the number of qubits required, and current quantum computers are not yet capable of supporting such computational burden.

1.1.2 Variational Quantum Algorithms and the barren plateau phenomenon

While fault-tolerant quantum computers are still years away, the pressing question is how to make effective use of today's noisy, intermediate-scale quantum (NISQ) devices [11], which are constrained by both qubit count (ranging from 50 to a few thousands) and susceptibility to errors. In order to demonstrate a real quantum advantage over classical computers, researchers have had to develop algorithms capable of operating within the constraints imposed by NISQ devices. In recent years, this effort has led to the emergence of Variational Quantum Algorithms (VQAs) [12], a class of hybrid quantum-classical algo-

rithms designed to make the most of current noisy hardware. Indeed, VQAs make use of quantum circuits whose gates depend on a set of parameters and whose output defines a parameter-dependent cost function. The optimization of this function is then delegated to classical optimizers, a feature that helps mitigate the effects of noise.

One of the main challenges limiting the applicability of variational quantum circuits is the training of the circuit itself. It has been observed that, for many circuit architectures, the optimization landscape tends to become flat in regions far from the optimal solution. As a result, the gradient of the cost function vanishes across large areas of the parameter space. This phenomenon, known as *barren plateau*, makes the optimization process extremely slow, as identifying a suitable direction for parameter updates would require exponentially high precision. In recent years, the barren plateau phenomenon has been extensively studied, and numerous papers have shown that it depends on the architectural features of the quantum circuit [13], [14], [15], [16], as well as on other factors such as excessive entanglement [17], [18] and the presence of noise in the circuits [19], [20]. More recently, Ref. [21] has argued that avoiding barren plateaus would imply exploiting a simple underlying structure in the problem, which could make the circuit classically simulable and thus nullify the quantum advantage. A more in-depth analysis of the phenomenon is presented in section 2.3.

1.2 Our results

The scope of our work is to use an approach based on the properties of Clifford and Pauli operators to study the phenomenon of barren plateau for a model of unstructured ansatz. The ansatz considered consists of parametrized unitaries acting on 1 or 2 qubits; random entangling gates acting con 2 qubits sampled independently and uniformly at random from the Haar measure over the unitary group $\mathcal{U}(4)$; and random single qubit gates randomly sampled from the Haar measure over $\mathcal{U}(2)$. In addition, we consider a local Pauli observable. As we detail in chapter 4, to study the barren plateau we can consider the second moment of the objective function, and thus we can replace the Haar measure on $\mathcal{U}(4)$ used to sample the 2-qubits entangling gates, with the uniform measure over the Clifford group $\mathcal{C}_2$. This is allowed because the Clifford group forms a 2-design[7], and thus the second moments of operators sampled from $\mathcal{C}_2$ are equivalent to those obtained with operators drawn uniformly at random from the Haar measure on $\mathcal{U}(4)$.

Our approach to the problem, consists in leveraging the properties of the action

[7] A t-design is a probability distribution over unitary operators which can duplicate properties of the probability distribution over the Haar measure for polynomials of degree t or less, more on the topic is discussed in subsection 2.3.3.

of the n-qubit Clifford group on the operators of $\mathcal{P}_n^*$, a subset of the Pauli group, within the Heisenberg picture, a formulation of quantum mechanics in which observables are time-dependent but the state of the system is independent of time. We analyze the problem in two distinct configurations. In the first configuration, we assume that the gates positions are stochastic and not fixed in advance, the sites they act upon are sampled at each step from the uniform distribution over the $\binom{n}{2}$ couples of sites in the n-qubits circuit. This assumption allows us to model the system as a relatively simple Markov chain. For this process we prove analytically an upper bound on the expected magnitude of the gradient of the cost function. The bound depends on both the number of qubits in the circuit n and on the number of entangling gates p. The result obtained, shows that, for a fixed number of qubits, the bound decays exponentially with the number of entangling gates. Moreover, the numerical results show that the decay of the expected gradient magnitude becomes slower as n increases relative to p, which is consistent with numerical simulations indicating that the relaxation time of the process grows almost linearly with the number of qubits. If instead we fix the number of gates and let n grow, we observe, thanks to the numerical results, that the gradient is no longer exponentially small in n when $n > p$, an outcome that confirms previous results that show absence of barren plateaus in sufficiently wide but shallow circuits.

The second configuration is the most important in real life applications, as it assumes that the position of the gates is fixed (i.e., we fix the architecture of the circuit) in advance. For this case, we can relate the BP to a random walk over the strings of n symbols over a 2-letter alphabet. We determine a lower bound on the expected magnitude of the gradient of the cost function that vanishes exponentially with the minimum number of gates that a trajectory of the random walk must pass trough.

1.3 Outline of the thesis

Chapter 2. In the second chapter, we introduce the fundamental principles and properties of quantum computing that are necessary for the following chapters. To this end, we present the postulates of quantum mechanics and highlight some of their key consequences. We then define the fundamental building blocks of quantum circuits: qubits and quantum gates. Subsequently, we formally introduce and discuss Variational Quantum Algorithms (VQAs). In this section we present the core components of VQAs, namely the cost function and the ansatz, and introduce some of their most relevant applications. In particular, we focus extensively on the barren plateau phenomenon, which hinders the training of parameterized quantum circuits. We review the initial results on this phenomenon and examine the underlying causes responsible for its emergence.

Chapter 4. In the fourth chapter we define $\mathcal{P}_n^*$, a restricted set of the n-qubits Pauli group that contains all the Hermitian Pauli operators apart from the identity. We then analyze the properties of the action of the n-qubit Clifford group $\mathcal{C}_n$ on $\mathcal{P}_n^*$, prove that it is transitive (Theorem 3.1) and then prove that the conjugation of an element $P \in \mathcal{P}_n^*$ by a randomly chosen Clifford operator yields a uniformly random element of $\mathcal{P}_n^*$. This result is formalized and proved in Theorem 3.2.

Chapter 5. In the fifth chapter, we address the central problem of the thesis: the study of the barren plateau phenomenon in unstructured variational circuits. After defining the ansatz setup, we recall Lemma 4.1 from Ref. [15], which allows us to simplify the analysis of the gradient by relating the natural measure of the flatness of the cost function $\mathbb{E}_V \mathbb{E}_\Theta \|f_V(\Theta)\|^2$ to the more accessible quantity $\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)$. We first study the vanishing gradient under the assumption that the position of the gates is not fixed in advance, a simplification that enables us to model the problem as a simple Markov chain. In this setting we show an analytical upper bound on the typical magnitude of the gradient that decays exponentially with the number of entangling gates (Proposition 4.4). We also numerically prove that when the number of qubits exceeds the number of non-parametrized gates, the gradient does not vanish. Subsequently, we consider the more realistic model with fixed gates positions and show (Proposition 4.5) that the lower bound previously established in Ref. [15] also holds within our framework.

Chapter 6 In the final chapter, we briefly summarize our results and discuss potential directions for future research.

Chapter 2

Quantum computing and Variational Quantum Algorithms

A quantum computer can be summarily defined as a computer that exploits the laws of quantum mechanics, the theory developed and employed to explain the behavior of physical objects at atomic and subatomic scale. As discussed in the previous chapter, the idea of such a device was first introduced in the early eighties, and has developed significantly since then.

In this chapter we aim at explaining the physical and mathematical foundations of quantum computing as well as the current state and open questions of the field. In section 2.1, we give a brief overview of the mathematical framework developed to explain these phenomenological observations, focusing mainly on the elements needed to understand the following chapters. Subsequently, in section 2.2 we introduce the basic elements of quantum computing. Finally, in section 2.3 we delve into the topic of Variational Quantum Algorithms, a highly active area of research in recent years.

2.1 Mathematical foundations of quantum mechanics

In the beginning of the 20th century, quantum mechanics was developed in order to overcome the limits of classical physics. The main phenomenological evidences presented by quantum systems that classical mechanics is unable to

explain, can be summarized in the following three properties:

i. *Randomness of measurement outcomes*: repeated measurements of the same physical quantity (observable) A, in the same physical condition (state) produce different results.

ii. *Post-measurement state*: let ψ represent the physical state of the considered quantum system and perform a measurement process on the system to measure the observable A. If we obtain the outcome a, then the state of the system after measurement changes to a new state ψ_a.

iii. *Incompatible observables*: there are observables that cannot be simultaneously measured by an experiment.

These evidences are justified by the postulates of quantum mechanics, which we now proceed to discuss. The ones we recall here, are some well known results for which we refer to [22] and [23] for a more detailed discussion and for the proof of the propositions that we state.

The central structure of the mathematical formulation of quantum mechanics is the Hilbert space $(\mathcal{H}, \langle\,|\,\rangle)$[1]. Quantum systems are generally studied in separable infinite-dimensional Hilbert spaces. However, since quantum computing does not require infinite dimensions, we limit our discussion to finite-dimensional Hilbert spaces. From now on, $\mathcal{H}$ always refers to a finite-dimensional Hilbert space. We introduce some additional notation: the set of linear operators on $\mathcal{H}$ is denoted by $\mathcal{L}(\mathcal{H})$, and the spectrum of an operator $A \in \mathcal{L}(\mathcal{H})$ is written as $\sigma(A)$. We now introduce the first postulate of quantum mechanics.

Postulate I

Any quantum system is associated to an Hilbert space $\mathcal{H}$. The state of the system is described by a positive semi-definite linear operator ρ with trace 1 acting on $\mathcal{H}$.

The operator ρ is called *density operator*, *density matrix* or *quantum state*. We denote as $\mathcal{D}(\mathcal{H})$ the set of all density operators, that is

$$\mathcal{D}(\mathcal{H}) = \{\rho \geq 0 |\ \mathrm{Tr}(\rho) = 1\}. \tag{2.1}$$

We say that a quantum state ρ is *pure* if it is an orthogonal projector of rank 1, that is

$$\rho = |\psi\rangle\langle\psi| \quad \text{for } |\psi\rangle \in \mathcal{H},\ \langle\psi|\psi\rangle = 1. \tag{2.2}$$

[1]For vectors and inner and outer products we use the Dirac notation that is typical of quantum mechanics and quantum computing.

Two vectors ψ, $\varphi \in \mathcal{H}$ with unit norm describe the same pure state if and only if

$$\exists\, \alpha \in \mathbb{R} \quad \text{such that} \quad |\psi\rangle = e^{i\alpha}|\varphi\rangle. \tag{2.3}$$

The term "pure state" refers to a quantum state that encodes the maximum knowledge of the system's physical conditions. When the information about the system's state is incomplete, the system is described by a *mixed state*:

$$\rho = \sum_i \lambda_i |\psi_i\rangle\langle\psi_i|, \quad \text{for} \quad |\psi_i\rangle \in \mathcal{H},\, \lambda_i \geq 0\ \forall i \quad \text{and} \quad \sum_i \lambda_i = 1. \tag{2.4}$$

Proposition 2.1. *The space $\mathcal{D}(\mathcal{H})$ is convex and any quantum state can be expressed as a convex combination of mutually orthogonal[2] pure states.*

A characteristic of quantum mechanics is that a quantum state can be a superposition of more physical states. We can distinguish between two types of superposition that we detail in the following definition.

Definition 2.1. *Let $\{|\psi_i\rangle\}_{i\in I} \subset \mathcal{H}$ be a finite collection of pure states. The pure state defined as the normalized linear combination of the states $|\psi_i\rangle$*

$$|\psi\rangle = \frac{\sum_{i\in I} a_i|\psi_i\rangle}{\|\sum_{i\in I} a_i|\psi_i\rangle\|} \quad \text{for } a_i \in \mathbb{C}\, \forall\, i \in I, \tag{2.5}$$

is a quantum state called coherent superposition of $\{|\psi_i\rangle\}_{i\in I}$. Let $\{\rho_i\}_{i\in I}$ be a collection of quantum states that can be pure or mixed, then the linear combination

$$\rho = \sum_{i\in I} \lambda_i \rho_i \quad \text{for } \lambda_i \geq 0 \quad \text{and} \sum_i \lambda_i = 1, \tag{2.6}$$

is a quantum state called incoherent superposition of $\{\rho_i\}_{i\in I}$.

The difference between the two lies in their relation with the classical framework. A state defined as a coherent superposition of pure states does not have a classical counterpart. Its existence is permitted by the mathematical structure of Hilbert spaces, which allows the construction of new pure states from existing ones; therefore, its nature is intrinsically quantum. On the other hand, an incoherent superposition corresponds to a classical ignorance about the state of the system: it is a statistical mixture of states and is not directly related to the quantum nature of the described system. If a system prepared in a coherent superposition interacts with the external environment, it evolves into an incoherent superposition. The process is known as *decoherence* and corresponds to an information loss of the initial state.

[2]Two pure states $\rho_1 = |\psi\rangle\langle\psi|$ and $\rho_2 = |\varphi\rangle\langle\varphi|$ are orthogonal, $\rho_1 \perp \rho_2$, if and only if $|\psi\rangle \perp |\varphi\rangle$.

Postulate II

A quantum measurement M on a quantum system is described by an outcome set X and a collection of measurement operators $\{E_x | x \in X\} \in \mathcal{L}(\mathcal{H})$ that satisfy the completeness equation

$$\sum_x E_x^\dagger E_x = \mathbb{I}, \tag{2.7}$$

The index x refers to the measurement outcome that may occur in the experiment.

If the state of the system before the measurement is ρ, then the probability that the outcome x occurs is

$$\mathbb{P}(x|\rho) = \mathrm{tr}[E_x \rho E_x^\dagger]. \tag{2.8}$$

As we mentioned at the beginning of this chapter, performing a measurement on a quantum system alters its state. In particular, we say that the state *collapses*. If the outcome x occurs, then the state of the system after measurement is

$$M_x(\rho) = \frac{E_x \rho E_x^\dagger}{\mathrm{tr}[E_x \rho E_x^\dagger]}. \tag{2.9}$$

For a pure state $\rho = |\psi\rangle\langle\psi|$, equations (2.8) and (2.9) simplify as follows

$$\mathbb{P}(x|\rho) = \langle\psi|E_x^\dagger E_x|\psi\rangle, \quad M_x(\rho) = \frac{E_x|\psi\rangle \langle\psi| E_x^\dagger}{\langle\psi|E_x^\dagger E_x|\psi\rangle}. \tag{2.10}$$

Note that the completeness equation (2.7) reflects the requirement that probabilities sum to one.

$$\sum_x \langle\psi|E_x^\dagger E_x|\psi\rangle = \sum_x \mathbb{P}(x) = 1. \tag{2.11}$$

We briefly introduce the *positive operator-valued measure* (POVM) formalism, a mathematical framework well suited for studying the probabilities of measurement outcomes. Define the operators

$$M_x = E_x^\dagger E_x, \tag{2.12}$$

then, for any measurement M, the following proposition holds:

Proposition 2.2. *Let $\{M_x | x \in X\}$ be a set of linear operators, then there exist a measurement M with operators $\{E_x | x \in X\}$ such that $M_x = E_x^\dagger E_x$ for all $x \in X$ if and only if $M_x \geq 0$ and $\sum_{x \in X} M_x = \mathbb{I}$.*

The operators M_x are called the POVM elements associated with the measurement and the complete set is known as a POVM. Given a POVM the probability of outcome x is

$$\mathbb{P}(x|\rho) = \mathrm{tr}[\rho M_x], \tag{2.13}$$

or, for a pure state

$$\mathbb{P}(x|\rho) = \langle\psi|M_x|\psi\rangle. \tag{2.14}$$

Thus, the set of operators M_x is sufficient to determine the probabilities of the different measurement outcomes.

Suppose that the measurement operators described above also satisfy the condition of being Hermitian and pairwise orthogonal $E_x E_{x'} = \delta_{xx'} E_x$. Then we can call the measure a *projective measure*.

Definition 2.2. *(Projective measurement) A measure M is a projective measurement if each E_x is an orthogonal projector, that is* $\mathrm{supp}(E_x) \perp \mathrm{supp}(E_{x'})$ $\forall x \neq x'$, $x, x' \in X$, *with*

$$\mathcal{H} = \bigoplus_{x\in X}^{\perp} \mathrm{supp}(E_x) \iff E_x E_{x'} = \delta_{xx'} E_x \text{ and } \sum_{x\in X} E_x = \mathbb{I}. \tag{2.15}$$

In this case, the POVM elements are the measurement operators themselves since $M_x = E_x^\dagger E_x = E_x$. For a projective measurement we denote the measurement operators E_x as Π_x.

Let M be a projective measurement with outcome set $X \subset \mathbb{R}$ and projectors $\{\Pi_x | x \in X\}$. Then, we associate to the measurement an observable A which is a self-adjoint[3] operator on $\mathcal{H}$ with spectral decomposition

$$A = \sum_{x\in X} x\Pi_x. \tag{2.16}$$

Therefore, from this characterization, we see that the outcome set corresponds to the spectrum of the observable $X = \sigma(A)$, and that $\mathrm{supp}(\Pi_x)$ is the eigenspace associated with the eigenvalue x. It is important to note that also the converse is true.

Proposition 2.3. *Given an observable A, there always exists a projective measurement M associated with A.*

When we measure an observable, what we are doing is applying the associated projective measure. Then, for a pure state $\rho = |\psi\rangle\langle\psi|$ (but similarly for mixed states) the outcome is a random eigenvalue whose probability is

$$\mathbb{P}(x|\rho) = \langle\psi|\Pi_x|\psi\rangle, \tag{2.17}$$

[3]Let $A \in \mathcal{L}(\mathcal{H})$. The adjoint of A is the unique operator $A^\dagger \in \mathcal{L}(\mathcal{H})$ such that $\langle A^\dagger\varphi|\psi\rangle = \langle\varphi|A\psi\rangle$ for any $|\varphi\rangle, |\psi\rangle \in \mathcal{H}$. The operator A is self-adjoint if $A = A^\dagger$.

and, given that the outcome x has occurred, the post measurement state is

$$\frac{\Pi_x|\psi\rangle}{\sqrt{\mathbb{P}(x|\rho)}}. \tag{2.18}$$

For a system with quantum state $|\psi\rangle \in \mathcal{H}$ the average value of the measure of the observable A, is found very easily through some simple calculations

$$\langle A\rangle = \mathbb{E}[A] = \sum_{x\in\sigma(A)} x\mathbb{P}(x||\psi\rangle\langle\psi|) \tag{2.19}$$

$$= \sum_{x\in\sigma(A)} x\langle\psi|\Pi_x|\psi\rangle \tag{2.20}$$

$$=\langle\psi|\left(\sum_{x\in\sigma(A)} x\Pi_x\right)|\psi\rangle \tag{2.21}$$

$$=\langle\psi|A|\psi\rangle. \tag{2.22}$$

We can now address the third phenomenological evidence that we introduced at the beginning of this chapter, that is the existence of observables that cannot be measured simultaneously. Let $\mathbb{P}(A = a \text{ and } B = b)$ be the joint probability of measuring the value a and b of the observables A and B respectively. If this value is well defined we say that the two observables are compatible.

Proposition 2.4. *Two observables A and B are compatible $\iff$ $[A, B] := AB - BA = 0$.*

If this is not verified then the joint probability is not well-defined and A and B cannot be measured simultaneously.

Postulate III

The state space of a composite physical system is the tensor product of the state spaces of the component physical systems. Thus, if we have systems numbered 1 through n and the system number i is prepared in the state $|\psi_i\rangle \in \mathcal{H}_i$, then the joint state of the total system is $|\Psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \ldots \otimes |\psi_n\rangle \in \mathcal{H} = \bigotimes_{i=1}^{n} \mathcal{H}_i$.

Therefore, composite systems exhibit an internal structure that enables the distinction of two or more subsystems, which can be independently observed through local measurements. The converse also holds: individual quantum systems can be combined to form composite systems.

We can now define what entangled states are.

Definition 2.3. *The pure state $\Psi \in \mathcal{H}_A \otimes \mathcal{H}_B$ is called separable if it can be written in the product form*

$$\Psi = \psi_A \otimes \psi_B, \quad \text{for } \psi_A \in \mathcal{H}_A,\, \psi_B \in \mathcal{H}_B. \tag{2.23}$$

If this is not possible, then we say that the state is entangled.

Thus, the system is separable if the two subsystems are uncorrelated and each of them presents a well defined state. If instead the system is entangled, the two subsystems are subject to a correlation that is purely quantum and does not have a classical counterpart. The definition can be extended to mixed states. Given a density operator $\rho \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B)$, we say that it is separable if it can be written as

$$\rho = \sum_i \lambda_i \rho_i^{(A)} \otimes \rho_i^{(B)}, \quad \text{for} \quad \lambda_i \geq 0 \quad \text{and} \quad \sum_i \lambda_i = 1,\ \forall i, \tag{2.24}$$

where $\rho_i^{(S)} \in \mathcal{D}(\mathcal{H}_S),$ for $S = A, B,$. Otherwise, the state is entangled.

Postulate IV

The evolution of a closed quantum system is described by the Schrödinger equation

$$i\hbar \frac{d}{dt}\rho(t) = [H(t), \rho(t)], \tag{2.25}$$

where $H(t)$ is a time-dependent self-adjoint linear operator acting on $\mathcal{H}$ called Hamiltonian of the system and $\hbar$ the Plank constant.

If the state of the system is a pure state Eq. (2.25) becomes

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H(t) |\psi(t)\rangle . \tag{2.26}$$

Analogously, the time evolution of the system can be described by a unitary transformation, that is the state of the system at time t_1 is related to the state of the system at time t_2 by a unitary operator U depending only on the times t_1 and t_2,

$$\rho(t_2) = U\rho(t_1)U^\dagger, \tag{2.27}$$

or, equivalently, for pure states

$$|\psi(t_2)\rangle = U |\psi(t_1)\rangle . \tag{2.28}$$

If H is time-independent the unitary describing the evolution is defined as

$$U = \exp\left(-i\frac{(t_2 - t_1)}{\hbar} H\right). \tag{2.29}$$

2.2 Quantum computing

Quantum computing is a model of computation that describes information processing with devices based on the laws of quantum physics. The fundamental unit at the basis of quantum computers is the qubit, the quantum counterpart of a classical bit. The framework for describing quantum computations is provided by quantum circuits, which are analogous to classical logic circuits. The building blocks of a quantum circuits, along with qubits, are the gates. We now describe the properties of qubits as well as the fundamental type of gates that appear in quantum circuits. Subsequently, we formally define what a quantum circuit is.

2.2.1 Qubits

The quantum counterparts of bits are *qubits*. The two possible states for a qubit are the ones defined by the vectors $|0\rangle$ and $|1\rangle$, that correspond to the states 0 and 1 of classical bits. The difference, is that a qubit can also be in a state other than the two mentioned above. Indeed, as a consequence of the properties of quantum mechanics that we discussed in the previous section, a qubit's state can be any coherent superposition of $|0\rangle$ and $|1\rangle$

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{2.30}$$

where $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$ due to the normalization of the state.

The special states $|0\rangle$ and $|1\rangle$ are known as *computational basis states*, and form an orthonormal basis for $\mathbb{C}^2$, the Hilbert space associated to the qubit.

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \langle 0\,|0\rangle = \langle 1\,|1\rangle = 1, \quad \langle 1\,|0\rangle = \langle 0\,|1\rangle = 0. \tag{2.31}$$

Consequently, any state of the type in Eq. (2.30), has vector representation

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}. \tag{2.32}$$

Since we impose the normalization condition $|\alpha|^2 + |\beta|^2 = 1$, Eq. (2.30) can be written in the following form

$$|\psi\rangle = e^{i\gamma}\left(\cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle\right), \quad \text{for} \quad \theta, \varphi, \gamma \in \mathbb{R}. \tag{2.33}$$

By (2.3) the factor $e^{i\gamma}$ can be ignored and we can write

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle\,. \tag{2.34}$$

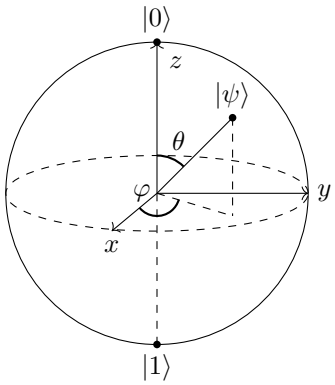


Figure 2.1: **The Bloch sphere**. The state is represented as a point on the surface of the sphere. The point is defined by the angles θ and φ that respectively represent the distance from the z and the x axis. The state $|0\rangle$ corresponds to $\theta = 0$, while the state $|1\rangle$ corresponds to $\theta = \pi$.

We see that the numbers θ and φ define a point on the three-dimensional unit sphere, called the *Bloch sphere* (see Figure 2.1). Thus, a quantum state can be visualized as a three-dimensional vector pointing to the surface of the Bloch sphere.

The framework we have described generalizes naturally to composite systems, except for the Bloch sphere, which can only be drawn for single-qubit systems. Consider a classical 2-bit system: the possible configurations are $00, 11, 10, 01$. Analogously, a 2-qubit system is described in the Hilbert space $\mathcal{H} = (\mathbb{C}^2)^{\otimes 2}$, which has four computational basis states that are $\{|00\rangle, |11\rangle, |01\rangle, |10\rangle\}$[4]. By the first postulate of quantum mechanics, any pure state in $\mathcal{D}(\mathcal{H})$ is described by the vector

$$|\psi\rangle = \alpha_{0,0}\,|00\rangle + \alpha_{1,1}\,|11\rangle + \alpha_{0,1}\,|01\rangle + \alpha_{1,0}\,|10\rangle\,, \tag{2.35}$$

with the normalization condition $\sum_{i,j} |\alpha_{i,j}|^2$ for $i, j \in \{0, 1\}$. The same approach applies for any n-qubit system with $n \geq 2$, where the quantum system is described by a state vector $|\psi\rangle \in (\mathbb{C}^2)^{\otimes n}$.

2.2.2 Gates

Definition 2.4. *(Quantum gate) A k-qubit quantum gate is a unitary operator acting on $(\mathbb{C}^2)^{\otimes k}$.*

Thus, quantum gates can act on as many qubits as we want. We focus on single

[4]The notation $|ab\rangle$ is equivalent to $|a\rangle \otimes |b\rangle$.

and 2-qubit quantum gates as these are the type of gates that we use in the following chapters.

The Pauli matrices

Before listing the most used quantum gates, it is useful to introduce the Pauli matrices.

$$X = \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.36}$$

The three operators are connected through the following relation:

$$\sigma_i \sigma_j = i\varepsilon_{ijk}\sigma_k + \delta_{ij}\mathbb{I}, \tag{2.37}$$

where ε_{ijk} is the Levi-Civita symbol and δ_{ij} the Dirac delta.

The matrices X and Z, together with the identity $\mathbb{I} \equiv \sigma_0$, define the n-qubit Pauli group $\mathcal{P}_n$ for $n \geq 1$.

Definition 2.5. *The n-qubit Pauli group $\mathcal{P}_n$ is defined as the subgroup of the unitary group $\mathcal{U}(2^n)$ consisting of all n-fold tensor products of n elements of*

$$\mathcal{P} \equiv \mathcal{P}_1 \coloneqq \langle X, Z, i\mathbb{I} \rangle. \tag{2.38}$$

In the definition of $\mathcal{P}$ the operator Y is not needed since by (2.37) $Y = iXZ$.

The three Pauli matrices, in addition, are all self-adjoint and thus, are associated with a quantum projective measurement

$$\sigma(X) = \{+1, -1\}, \quad \Pi_{+1} = |+\rangle\langle+|, \quad \Pi_{-1} = |-\rangle\langle-|. \tag{2.39}$$

$$\sigma(Y) = \{+1, -1\}, \quad \Pi_{+1} = |i\rangle\langle i|, \quad \Pi_{-1} = |-i\rangle\langle -i|. \tag{2.40}$$

$$\sigma(Z) = \{+1, -1\}, \quad \Pi_{+1} = |0\rangle\langle 0|, \quad \Pi_{-1} = |1\rangle\langle 1|. \tag{2.41}$$

Where the states $|+\rangle, |-\rangle, |i\rangle, |-i\rangle$ take the following form when expressed in the computational basis

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}, \quad |i\rangle = \frac{|0\rangle + i|1\rangle}{\sqrt{2}}, \quad |-i\rangle = \frac{|0\rangle - i|1\rangle}{\sqrt{2}}. \tag{2.42}$$

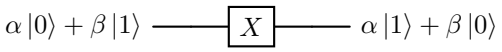
Moreover, the set $\{\mathbb{I}, X, Y, Z\}$ defines a basis for the self-adjoint 2×2 matrices. An important consequence of this is that any self-adjoint operator M, can be written as a real linear combination of Pauli matrices

$$M = \sum_{i=0}^{3} \lambda_i \sigma_i \quad \text{for} \quad (\lambda_0, \lambda_1, \lambda_2, \lambda_3) \in \mathbb{R}^4. \tag{2.43}$$

Single qubit gates

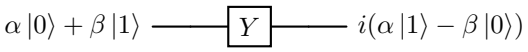
Single-qubit gates are represented by 2×2 unitary transformations that can be written as matrices. Below, we list the most important ones.

i. **Pauli X gate**. The Pauli X gate acts as the quantum equivalent of the classical NOT gate and is commonly referred to as the *bit-flip* gate because its action on a quantum state swaps the states $|0\rangle$ and $|1\rangle$.



ii. **Pauli Z gate**. The Pauli Z gate, also known as the *phase-flip* gate, leaves the state $|0\rangle$ unchanged while mapping $|1\rangle$ to $-|1\rangle$.

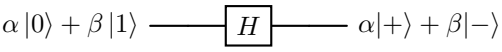
iii. **Pauli Y gate**. The Pauli Y gate acts as a combination, up to a phase, of a Pauli X and Pauli Z gate due to (2.37).



iv. **Hadamard gate**. The Hadamard gate is defined by the matrix

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{2.44}$$

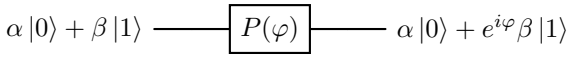
The gate acts on a quantum state transforming $|0\rangle$ into $|+\rangle$ and $|1\rangle$ into $|-\rangle$.



v. **Phase-shift gates**. Phase-shift gates are a family of quantum gates, each represented in matrix form as

$$P(\varphi) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}, \tag{2.45}$$

where φ is the phase-shift with period 2π. The action of the gate on a quantum state is



Different values of φ define different gates. Some notable examples include the T gate, where $\varphi = \frac{\pi}{4}$, and the S gate (also known as *phase gate*) where $\varphi = \frac{\pi}{2}$. Moreover, that the Pauli Z gate is also a phase shift gate, with $\varphi = \pi$.

vi. **Pauli rotation gates**. When exponentiated, the Pauli matrices give rise to three useful classes of unitary operators: the rotation operators about the x, y, and z axes of the Bloch sphere. The gates are defined by the following equations

$$R_x(\theta) \equiv e^{-i\theta X/2} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}X = \begin{pmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \tag{2.46}$$

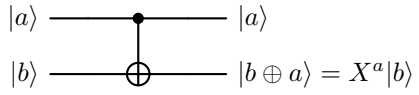
$$R_y(\theta) \equiv e^{-i\theta Y/2} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}Y = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix} \tag{2.47}$$

$$R_z(\theta) \equiv e^{-i\theta Z/2} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}Z = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix} \tag{2.48}$$

Two qubit quantum gates

Two-qubit quantum gates are represented by 4×4 matrices that act on two qubits. Below, we list some of the most important ones.

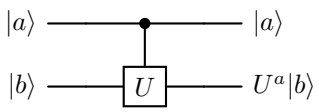
i. **Controlled NOT gate**. The CNOT gate acts on two qubits and, depending on the state of the first qubit, either applies the NOT operation to the second qubit or leaves it unchanged. For $a, b \in \{0, 1\}$ we have



What happens is that if $a = 0$ then $b \oplus a = b$, if $a = 1$ then $b \oplus a = \bar{b}$. The gate is represented in matrix form as

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{2.49}$$

ii. **Controlled U gate**. The CU gate is a generalization of the CNOT gate. Depending on the value of the first qubit, a unitary operator U is either applied to the second qubit or not. For $a, b \in \{0, 1\}$ we have

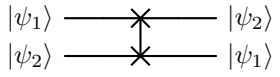


In matrix form is represented as

$$\mathrm{CU} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{pmatrix}. \tag{2.50}$$

where each u_{ij} is the coefficient in position i, j of the matrix U.

iii. **Swap gate**. The gate swaps the state of two qubits

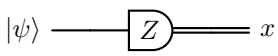


In matrix form is represented as

$$\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{2.51}$$

Measurement gates

Another important operation is the *measurement in the canonical basis*, which we represent with the following symbol



where $x \in \{0, 1\}$ is a classical output. The operation converts a single qubit state into a probabilistic classical bit. The measurement is described by the POVM $\{\Pi_{+1} = |0\rangle\langle 0|,\ \Pi_{-1} = |1\rangle\langle 1|\}$. The observable corresponding to the computational basis measurement is the Pauli Z operator, whose eigenvalues $+1$ and -1 are associated with the outcomes $|0\rangle$ and $|1\rangle$, respectively.

For a pre-measurement state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, the probability of obtaining outcome 1 is

$$\mathbb{P}(+1|\ |\psi\rangle) = \langle\psi|\,\Pi_{+1}\,|\psi\rangle = |\alpha|^2. \tag{2.52}$$

Similarly, the probability of obtaining outcome -1 is

$$\mathbb{P}(-1|\ |\psi\rangle) = \langle\psi|\,\Pi_{-1}\,|\psi\rangle = |\beta|^2. \tag{2.53}$$

After a measurement with outcome 1, the system collapses into the state

$$\frac{\Pi_{+1}|\psi\rangle}{\sqrt{\langle\psi|\Pi_{+1}|\psi\rangle}} = |0\rangle\,, \tag{2.54}$$

while, if the outcome is -1

$$\frac{\Pi_{-1}|\psi\rangle}{\sqrt{\langle\psi|\Pi_{-1}|\psi\rangle}} = |1\rangle\,. \tag{2.55}$$

The $\{|0\rangle, |1\rangle\}$ basis is not the only one allowed for measurements, although it is the most commonly used. For example another possible basis is $\{|+\rangle, |-\rangle\}$ and the observable corresponding to this measurement is Pauli X.

We now detail a few more properties of quantum measurements. In a composite system, a measurement can act on multiple qubits. A measurement is said to be *local* if it acts non-trivially only on a small subset (typically one or a few qubits) of the larger quantum system, and trivially (as the identity) on the remaining qubits[5]. An example is the measurement associated with the following n-qubit observable:

$$M = \mathbb{I}\otimes\mathbb{I}\otimes\ldots\otimes Z\otimes\ldots\otimes\mathbb{I}\otimes\mathbb{I}. \tag{2.56}$$

By the linearity of quantum mechanics, the expectation value of a composite observable of the form (2.56) on a product state such as $|0^n\rangle$ factorizes as the classical product of the expectation values on each individual qubit:

$$\langle 0^n|M|0^n\rangle = \langle 0|\mathbb{I}|0\rangle \times \ldots \times \langle 0|Z|0\rangle \times \ldots \times \langle 0|\mathbb{I}|0\rangle \tag{2.57}$$

As shown in equation (2.43), any single-qubit observable can be decomposed as a linear combination of Pauli operators. For this reason, it is useful, particularly for the upcoming chapters, to understand how the expectation value of a sum of operators is computed. Given an observable M of the form (2.43), its expectation value on the state ψ is equal to the sum of the expectation values of each term in the decomposition:

$$\langle\psi|M|\psi\rangle = \sum_{i=0}^{3}\lambda_i\langle\psi|\sigma_i|\psi\rangle\,. \tag{2.58}$$

2.2.3 Quantum circuits

The concepts introduced above now enable us to outline the structure of a quantum circuit.

[5]Eventually we may write that an observable is k-local to specify the number of qubits it acts non trivially upon.

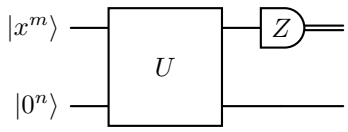


Figure 2.2: **Generic structure of an $m+n$-qubits quantum circuit**. The figure shows a possible generic structure for a quantum circuit. The gate U is any unitary quantum gate acting on the circuit. The qubit $|x^m\rangle$ is a single qubit in the computational basis and $|0^n\rangle = |0\dots0\rangle$ are the ancillary qubits. The circuit performs a measurement in the computational basis on each of the first m qubits.

Definition 2.6. *An n-qubit quantum circuit consists of*

i ***A suitable state space**. The quantum circuit operates on the state space is $(\mathbb{C}^2)^{\otimes n}$ with computational basis $|x_1\dots x_n\rangle$ for $x_i = 1, 0$.*

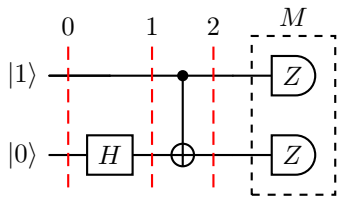
ii ***Preparation of the starting state**. The starting state is prepared in the computational basis and it is assumed that this preparation can be done in at most n steps.*

iii ***Unitary quantum gates**. Gates can be applied to any subset of the n qubits.*

iv ***Measurement**. Measurements can be performed in the computational basis or in any other basis, and may be applied to one or more qubits.*

In addition, a quantum computer may require classical resources, as certain tasks can be significantly simplified if parts of the computation are carried out classically. It may also require the use of ancillary qubits, that are extra qubits added to the quantum circuit to assist the computation and generally initialized in a known state (generally $|0\rangle$).

As an example, consider the 2-qubits circuit below where the 2-qubit separable observable $M = Z \otimes Z$ is measured.



Let us see what happens to the state of the system after each gate

$$|\psi_0\rangle = |10\rangle\,, \tag{2.59}$$

$$|\psi_1\rangle = H\,|\psi_0\rangle = |1\rangle \otimes H\,|0\rangle = |1\rangle \otimes \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{|10\rangle + |11\rangle}{\sqrt{2}}, \tag{2.60}$$

$$|\psi_2\rangle = \mathrm{CNOT}\,|\psi_1\rangle = |1\rangle \otimes \frac{(|1\rangle + |0\rangle)}{\sqrt{2}} = \frac{1}{\sqrt{2}}(|11\rangle + |10\rangle). \tag{2.61}$$

The expected value of the measurement of M is the product of the following expectations:

$$\langle Z\rangle_1 = \langle 1|\,Z\,|1\rangle = \langle 1|\,(|0\rangle\,\langle 0| - |1\rangle\,\langle 1|)\,|1\rangle = -1, \tag{2.62}$$

$$\langle Z\rangle_2 = \frac{1}{2}(\langle 0| + \langle 1|)Z(|0\rangle + |1\rangle) = \frac{1}{2}(\langle 0| + \langle 1|)(|0\rangle\,\langle 0| - |1\rangle\,\langle 1|)(|0\rangle + |1\rangle) = 0, \tag{2.63}$$

where $\langle Z\rangle_i$ is the expectation of the observable Z on the i-th subsystem. Then

$$\langle M\rangle = \langle\psi_2|\,M\,|\psi_2\rangle = 0 \times (-1) = 0. \tag{2.64}$$

2.2.4 The Haar measure

While not directly related to the physical structure of the circuit, it is worth briefly discussing the Haar measure, as it formalizes the fundamental concept of drawing unitary matrices uniformly at random. This is essential for our purposes, since in the following chapters we require this type of sampling.

Definition 2.7. *The Haar measure on the unitary group $\mathcal{U}(d)$ is the unique probability measure μ_H that is both left and right invariant over the group $\mathcal{U}(d)$, i.e., for all integrable functions f and for all $V \in \mathcal{U}(d)$, we have:*

$$\int_{\mathcal{U}(d)} f(U)\,d\mu_H(U) = \int_{\mathcal{U}(d)} f(UV)\,d\mu_H(U) = \int_{\mathcal{U}(d)} f(VU)\,d\mu_H(U). \tag{2.65}$$

In addition, for every measurable subset S of $\mathcal{U}(d)$, the Haar measure satisfies the following properties

$$\int_S 1\,d\mu_H(U) \geq 0, \tag{2.66}$$

and

$$\int_{\mathrm{U}(d)} 1\,d\mu_H(U) = 1. \tag{2.67}$$

Therefore, it represents a probability measure and we can denote the integral of any function $f(U)$ over the Haar measure as the expected value of $f(U)$ with respect to the probability measure μ_H,

$$\mathbb{E}_U[f(U)] := \int_{\mathrm{U}(d)} f(U)\,d\mu_H(U). \tag{2.68}$$

An in depth study of the properties and applications of the Haar measure is beyond our scope and we refer to [24] for a more comprehensive study of the topic.

2.3 Variational Quantum Algorithms

Quantum computers promise to bring significant benefit for a various number of applications. As we discussed in the previous sections, this new technology leads to an exponential speed up in many challenging problems for classical computers.

Despite these encouraging premises, implementing quantum algorithms in practice is far from straightforward. The main obstacle lies in the very nature of quantum mechanics: whenever a quantum system interacts with the external environment, an uncontrollable disturbance in the system is produced, which manifests as noise in the quantum circuit. To prevent errors from spreading, the system must be perfectly isolated from the external environment. At the same time, however, we need to control the qubits states through external agents. Satisfying both requirements simultaneously, makes building a functional quantum computer a challenging goal. Nevertheless, it is theoretically possible to overcome the effect of noise without compromising the quantum information process: this is done thanks to the use of quantum error correction techniques (QEC). Unfortunately, the expenses of such methods in terms of the number of qubits is, at the present day, still far from current experimental capabilities.

With the advent of fault-tolerant quantum computers still many years away, the key question is how to make the best use of the current generation of quantum devices, known as NISQ (Noisy Intermediate-Scale Quantum) computers [11]. As the name suggests, these devices must deal with noise and are limited to a qubit count ranging from around 50 to a few thousands. A promising approach to achieving quantum advantage in the NISQ era is the use of Variational Quantum Algorithms (VQAs). Such algorithms employ parametrized quantum circuits to be run on a quantum computer and then outsource the parameter optimization to classical optimizers. This hybrid approach satisfies the constraints of NISQ devices, in particular it allows to keep the quantum circuit depth shallow, a feature that helps mitigating the noise [12].

In this section, we firstly examine the key components of Variational Quantum Algorithms, and study some significant applications. Subsequently, we focus on the training processes of these algorithms and the challenges that arise, with particular emphasis on the central topic of this thesis, the barren plateaus phenomenon.

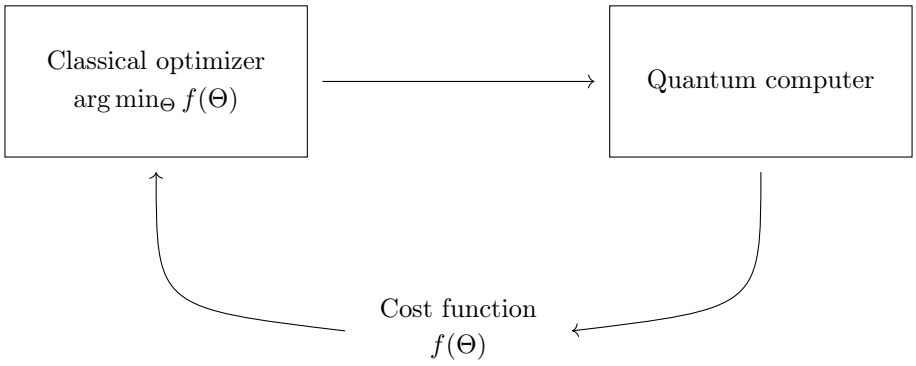


Figure 2.3: **Diagram of the hybrid quantum-classical loop of a VQA:** a quantum computer estimates the cost function defined in Eq. (2.69). Subsequently, the parameters are optimized using classical optimizers that employ methods such as gradient descent to identify the optimal update direction.

2.3.1 The foundations of VQA: the ansatz and the cost function

Variational Quantum Algorithms are parametrized algorithms, thus they depend on a vector of trainable parameters $\Theta = \{\theta_1, \ldots, \theta_m\}$. These parameters may be continuous or discrete and are usually assumed independent from each other. Another building block of a VQA is the ansatz U, which represents the specifications for the arrangement and type of quantum gates in the circuit, and how these depend on the set of parameters [25]. Finally, once the ansatz is fixed, one defines the cost function $f(\Theta)$ that can be expressed as

$$f(\Theta) = \langle\psi_0|U^\dagger(\Theta)MU(\Theta)|\psi_0\rangle, \tag{2.69}$$

or in density matrix notation,

$$f(\Theta) = \mathrm{tr}\{M\rho(\Theta)\}, \tag{2.70}$$

where $\rho(\Theta) = U(\Theta)^\dagger|\psi_0\rangle\langle\psi_0|U(\Theta)$. For a circuit with n qubits, $U(\Theta) \in \mathcal{U}(2^n)$ is the unitary defined by the ansatz and the parameter vector, that acts on the starting state ψ_0, and M an observable. Sometimes f is also called loss function [12], objective function [15] or, in the Quantum Machine Learning context, model function[6] [26].

Definition 2.8. *The goal of a VQA is finding a solution to the optimization problem*

$$\Theta^* = \arg\min_\Theta f(\Theta). \tag{2.71}$$

[6] In QML applications, what is referred to as the model function coincides with our definition of the function f. However, it is important to note that this is not the function that is generally minimized; nonetheless, the minimization problem depends on f.

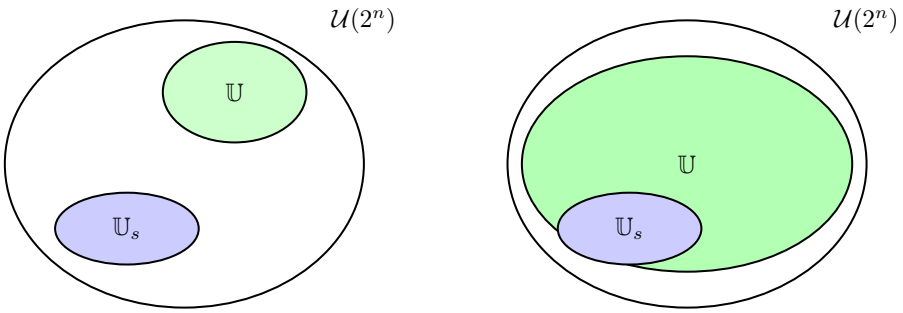


Figure 2.4: **Expressibility of an ansatz:** on the left we see an inexpressive ansatz, it explores a small portion of the unitary group, and if not tailored for the specific problem it may not reach the desired solution. On the right an expressive ansatz is represented, it explores the unitary space uniformly and it is likely to reach a solution for any given problem.

It is worth to spend a few more words on the ansatz, since the way it is built determines the success of the routine.

Assumption 2.1. *Without loss of generality, a parametrized ansatz $U(\Theta)$ can be expressed as*

$$U(\Theta) = \prod_j e^{-i\theta_j A_j} T_j, \tag{2.72}$$

where T_j are fixed unitaries, and A_j an Hermitian operator such that $(A_j)^2 = \mathbb{I}$.

The choice of the appropriate ansatz is crucial for the cost function to be trainable. Indeed, once the ansatz is fixed, each possible parameter vector Θ defines a different unitary $U(\Theta)$ and, therefore, a different quantum circuit. The set of all possible parameters $\{\Theta^{(1)}, \Theta^{(2)}, \ldots, \Theta^{(k)}\}$ defines the corresponding set of unitaries $\mathbb{U} = \{U^{(1)}, U^{(2)}, \ldots, U^{(k)}\}$ where $U^{(i)} = U(\Theta^{(i)})$. The generated set $\mathbb{U}$ is a subset of the unitary group $\mathcal{U}(2^n)$, and the way it explores $\mathcal{U}(2^n)$ has deep consequences on the trainability and the success of the algorithm.

Borrowing the notation from [14] we name $\mathbb{U}_s$ the set of unitaries that minimize the cost function. This set may contain one or more elements depending on the number of minima of $f(\Theta)$. A VQA finds a solution to the problem only if $\mathbb{U}_s \cap \mathbb{U} \neq \emptyset$ and an ansatz that satisfies this condition is said complete. Complete ansätze are relatively easy to build if prior knowledge about the problem is available. However, this is not always possible, and it may happen that we have very limited information on where $\mathbb{U}_s$ lies. Under these circumstances, our ignorance on the expected solution makes building specific ansätze an impossible task. Consequently, to increase the likelihood of identifying a valid solution, the ansatz must be designed to explore the unitary space as extensively and uniformly as possible: an ansatz that satisfies this requirement is said to be

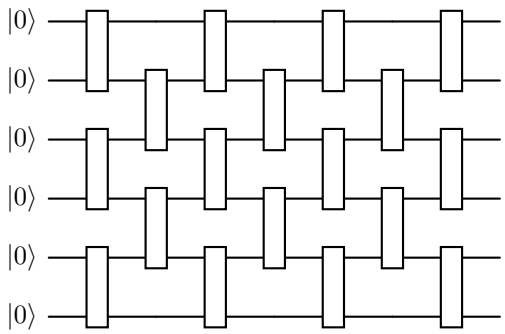


Figure 2.5: **Alternating Layer Ansatz (ALT)**. The ALT is a problem-agnostic ansatz whose entangling gates are restricted to entangle only local qubits in each layer. It has been shown that for circuits with local observables it is possible to avoid trainability issues if an ALT with depth $O(\log(n))$ is used [16], while still maintaining the typical expressibility of problem agnostic ansätze [27].

expressive. At first glance, it may seem convenient to design a highly expressive ansatz that can be applied to a wide range of problems. However, to explore the whole unitary group, an ansatz needs an exponential number of parameters. Thus, this strategy presents significant challenges in terms of trainability as demonstrated in Ref. [14].

Hardware efficient ansatz

Following the previous section's discussion, ansätze can be grouped into two main categories. The first includes those that are generally inexpressive but complete, these are known as problem inspired ansätze and they are constructed using specific knowledge about the problem. The second category is the one of those ansätze that are employed when no relevant information about the problem is available; these are referred to as problem agnostic ansätze. To fulfill the scope of being adaptable to a wide range of problems, the latter must be sufficiently expressive to ensure completeness.

The most famous problem-agnostic ansatz is the Hardware Efficient Ansatz (HEA) [25]. The HEA is well-known for leveraging gates that are native to the specific quantum hardware. Indeed, the ansatz employs unitaries T_j and $e^{-i\theta_j A_j}$ drawn from a set of gates determined by the native connectivity of the hardware being used. If non-native gates are used, the implementation of an ansatz on real quantum hardware may require more gates than anticipated, since the device might not directly support the gates assumed in the mathematical formulation. By contrast, using a Hardware Efficient Ansatz ensures that the model employs exactly the gates required for physical implementation, making

it more representative of real-world performance. In addition to this, its high expressibility makes this ansatz adaptable to a wide range of problems. However, as we discussed previously, this characteristic, while being its main advantage, is also its greatest weakness as it affects the trainability of the cost function.

2.3.2 Applications

VQAs provide a framework that can be used to address a wide array of tasks. In this section we present the main applications of these architectures. It is important to notice that the ones we present here do not exhaust all possible uses of VQA. A more comprehensive list can be found in Ref. [12].

Finding the ground state: the Variational Quantum Eigensolver

The Variational quantum eigensolver (VQE) is arguably the most famous application of VQAs. Indeed, variational quantum algorithms were initially proposed as a tool for finding the ground state of quantum systems. The goal of such algorithms is to find the eigenstate $|\psi\rangle$ that minimizes the expectation of an Hamiltonian H

$$\langle\psi|H|\psi\rangle. \tag{2.73}$$

The architecture of VQEs follows the structure we have defined in the previous section: a parametrized ansatz $U(\Theta)$ acts on a starting state $|\psi\rangle$. The cost function is the expectation of the Hamiltonian H in the state $|\psi(\Theta)\rangle = U(\Theta)|\psi\rangle$. This simple structure makes it possible to study the problem from a different perspective: instead of directly finding the eigenstate $|\psi^*\rangle$ that minimizes the expectation of H, we look for the parameter vector Θ^* that solves the minimization problem

$$\Theta^* = \arg\min_{\Theta}\langle\psi(\Theta)|H|\psi(\Theta)\rangle. \tag{2.74}$$

An important challenge in VQEs is the implementation of the Hamiltonian as an observable, which is not always straightforward. Fortunately, any observable can be expressed as a linear combination of Pauli operators σ_j

$$H = \sum_j c_j\sigma_j. \tag{2.75}$$

Even more conveniently, in many practical applications, the terms in this summation correspond to local observables (i.e., 1- or 2-qubit operators), which allows for an efficient estimation of H.

Optimization: Quantum Approximate Optimization Algorithm

Finding the ground state of a Hamiltonian is an inherently quantum task. Nevertheless, the applications of VQAs extend beyond the purely quantum domain. Indeed, another popular usage of such algorithms is to solve classical optimization problems. The most famous optimization application of VQAs is the quantum approximate optimization algorithm (QAOA); it has been proposed by Farhi et al. [28] and it approximately solves combinatorial optimizations problems such as Constraint-Satisfaction[7] and Max-Cut [8].

We briefly present the SAT problem[9] discussed in the original paper by Farhi. The formulation of the problem relies on two fundamental elements. The first is the variables vector $z = (z_1, \ldots, z_n)$, $z_i \in \{0, 1\}$. The second is the set of constraints $C_\alpha(z)$, for $\alpha = 1, \ldots, k$. Each constraint $C_\alpha(z)$ depends on a subset of the elements of the vector z and it is satisfied for certain assignments of those variables and unsatisfied for others assignments [28]. In particular whenever $C_\alpha(z)$ is satisfied, $C_\alpha(z) = 1$, while contrarily, if unsatisfied $C_\alpha(z) = 0$. Therefore, the objective function of the problem is associated to the following expression:

$$C(z) = \sum_{\alpha=1}^{k} C_\alpha(z). \tag{2.76}$$

Equation (2.76) is represented by the expectation value of a quantum operator C:

$$\langle\psi|C|\psi\rangle. \tag{2.77}$$

In matrix representation, C contains on its diagonal the number of statements satisfied by a bit string z, while outside the diagonal, all elements are zero. The optimization problem can be solved by first finding the state $|\psi\rangle$ that maximizes (2.77) and then sampling computational basis states from it, which represent the solution bit strings to the problem. The starting state is generally a uniform superposition $|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_z |z\rangle$. The ansatz uses the operator C itself along with a second operator, that we call B, that is a sum of Pauli X acting on all qubits,

$$B = \sum_{i=1}^{n} \sigma_x^i. \tag{2.78}$$

[7] A Constraint Satisfaction Problem (CSP) is a mathematical problem in which one or more variables must be assigned values that satisfy a set of constraints. The goal is to find an assignment that satisfies all the constraints.

[8] A Max-Cut problem is solved by finding the maximum cut of a graph, that is a partition of the graph's vertices into two complementary sets, such that the number of edges between the two is as large as possible.

[9] A particular instance of CSPs is the Boolean Satisfiability Problem (SAT), where variables are boolean and constraints are logical formulas.

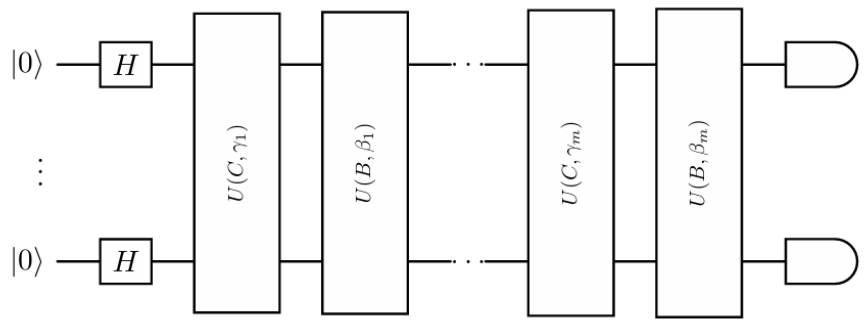


Figure 2.6: **General structure of the QAOA**: the starting state is prepared by the Hadamard gates acting on each qubit. This generates the superposition $|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_z |z\rangle$. Subsequently, the ansatz consists of an alternating sequence of unitaries generated by the operators B and C.

The circuit architecture consists of an alternating sequence of the unitaries generated by the operators B and C:

$$U(B, \beta_j) = e^{-i\beta_j B}, \tag{2.79}$$

$$U(C, \gamma_j) = e^{-i\gamma_j C}. \tag{2.80}$$

For the parameters β_j and γ_j, $j = 1 \dots m$. Then,

$$|\psi(\Theta)\rangle = U(B, \beta_m)U(C, \gamma_m) \dots U(B, \beta_1)U(C, \gamma_1)|\psi\rangle. \tag{2.81}$$

Let Θ^* be the $2m$-dimensional vector that satisfies the optimization problem

$$\Theta^* = \arg\max_{\Theta} \langle\psi(\Theta)|C|\psi(\Theta)\rangle. \tag{2.82}$$

Then, if the maximum is obtained, sampling computational basis states from $|\psi(\Theta^*)\rangle$ provides good candidates for z.

Ref. [28] shows that, with the ansatz we just described, the maximum of (2.76) is obtained for $m \to \infty$

$$\lim_{m\to\infty} \max_{\Theta} \langle\psi(\Theta)|C|\psi(\Theta)\rangle = \max_z C(z). \tag{2.83}$$

Finding the optimal values of β_i and γ_i for $i = 1 \dots m$ is a nonconvex problem with many local optima, therefore finding a way to efficiently train the algorithm still remains an active field.

Quantum Machine Learning

Machine learning (ML) is a sub-branch of artificial intelligence that focuses on identifying and learning patterns from a given dataset. This knowledge is then used in order to generalize these patterns to unseen data, with the aim of making accurate predictions. Unlike many other areas of scientific research, machine learning is known for its typically empirical approach. Generally, when an algorithm proves to be particularly effective, it is mostly the result of trial and error than the direct application of a mathematical theory.

The introduction of quantum computing in machine learning follows from a simple reasoning. Quantum mechanics is known for its ability to find counter-intuitive patterns in data, therefore, we may hope that a quantum computer is able to recognize patterns that are difficult to recognize classically [29], [30]. Clearly, quantum machine learning (QML) still faces all the problems arising from near term quantum devices. It is therefore impossible to outsource the entire machine learning pipeline to a quantum computer. The most used solution to circumvent the problem is to implement the machine learning model as a quantum algorithm, and then train the model through a classical computer. Obviously, the most suitable class of quantum algorithms for this purpose is the class of VQAs [26].

A machine learning model is a function $f : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ is the set of input datas and $\mathcal{Y}$ the set of output datas.

Definition 2.9. *(Deterministic quantum model) Let $\mathcal{X}$ be the set of input data. Let $U(x, \Theta)$ be a quantum circuit that depends on an input $x \in \mathcal{X}$ and on a parameter vector $\Theta \in \mathbb{R}^m$. Let M be a quantum observable and $|\psi_0\rangle$ a starting state. The function*

$$f(x, \Theta) = \langle \psi_0 | U(x, \Theta)^\dagger M U(x, \Theta) | \psi_0 \rangle \tag{2.84}$$

defines a deterministic variational quantum model.

The structure of the ansatz in a variational circuit for quantum machine learning follows the general architecture of a VQA ansatz. What differs in this context is the need to incorporate input data into the circuit. A typical choice for $U(x, \Theta)$ consists of a data-embedding block, $D(x)$, and a parameterized block, $P(\Theta)$ (see Fig. 2.7). Clearly, both blocks can be further decomposed.

$$D(x) = S_{l+1} \prod_{k=1}^{l} e^{-i x_k A_k} S_k, \tag{2.85}$$

$$P(\Theta) = T_{m+1} \prod_{j=1}^{m} e^{-i \theta_j A_j} T_j, \tag{2.86}$$

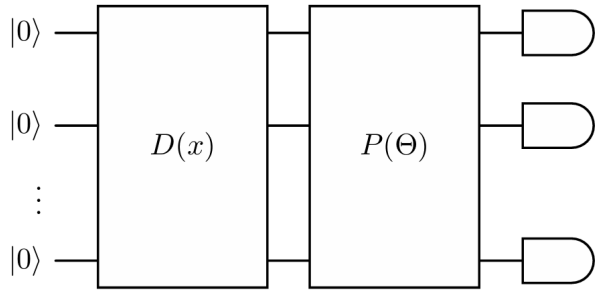


Figure 2.7: **General architecture of a VQA for quantum machine learning**. The ansatz $U(x, \Theta)$ consists of a block responsible for data encoding $D(x)$, and a block composed of parameterized gates $P(\Theta)$. While for convenience we represented the two as separate, it is important to notice that the gates for data encoding and the parametrized ones can also be mixed.

where, for all $k = 1, \dots, l$ and $j = 1, \dots, m$, S_k and T_j are fixed unitaries and A_k, A_j are given Hermitian operator.

The probabilistic nature of quantum mechanics allows for the implementation of probabilistic quantum models, which can be either supervised or unsupervised. While the latter are particularly relevant for practical applications, we briefly discuss both paradigms to highlight and understand their differences.

Definition 2.10. *(Supervised probabilistic quantum model) Let $\mathcal{X}$ and $\mathcal{Y}$ be respectively the input and output domains. Let $U(x, \Theta)$ be a quantum circuit defined as above and $|\psi_0\rangle$ a starting state. Let M be an observable such that each outcome of a measurement of M is associated with a possible output $y \in \mathcal{Y}$, that is $M = \sum_{y \in \mathcal{Y}} y|y\rangle\langle y|$. A supervised probabilistic quantum model for a conditional distribution is defined by*

$$p_\Theta(y|x) = |\langle y|\psi(x, \Theta)\rangle|^2, \tag{2.87}$$

where $|\psi(x, \Theta)\rangle = U(x, \Theta)|\psi_0\rangle$.

If the labels associated with the input data are not available, we need to implement an unsupervised model. In this case, data are not encoded with a circuit. Here, the measurement outcomes are associated with data points $x \in \mathcal{X}$ instead of the labels $y \in \mathcal{Y}$.

Definition 2.11. *(Unsupervised probabilistic quantum model) Let $\mathcal{X}$ be the input domain, and $P(\Theta)$ be a parametrized unitary that defines the vector $|\psi(\Theta)\rangle = P(\Theta)|\psi_0\rangle$ for a given starting state $|\psi_0\rangle$. Let M be a measurement with outcomes that correspond to the inputs x, that is $M = \sum_{x \in \mathcal{X}} x|x\rangle\langle x|$. An unsupervised probabilistic quantum model is defined by the distribution*

$$p_\Theta(x) = |\langle x|\psi(\Theta)\rangle|^2. \tag{2.88}$$

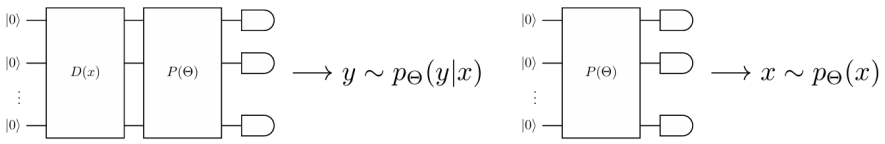


Figure 2.8: **Supervised probabilistic quantum models**: on the left a variational quantum algorithm is used as a supervised QML model of a conditional distribution. The circuit samples the values $y \in \mathcal{Y}$ from the probability distribution $p_\Theta(y|x)$. In the right image is represented a variational quantum algorithm used as an unsupervised supervised QML model of a distribution. The circuit samples value $x \in \mathcal{X}$ from the probability distribution $p_\Theta(x)$.

The differences between the two paradigms can be visualized in Fig. 2.8.

A comprehensive review of QML is beyond the scope of this work, a more in-depth analysis of the subject can be found in Ref. [31]. Instead, we briefly explore some meaningful applications of these architectures.

Classifiers Classification is the most straightforward application of QML. Consider a training dataset whose elements are the inputs/labels couples $(x^{(i)}, y^{(i)})$. Our goal is to train the algorithm to predict the label of each input and apply this knowledge to new datasets as well. The cost function represents the difference between the true label and the expectation value of an easily measurable observable:

$$f(x, \Theta) = \sum_i [y^{(i)} - \langle \psi_0 | U(x, \Theta)^\dagger M U(x, \Theta) | \psi_0 \rangle]^2. \tag{2.89}$$

Generative models: Variational Generator A variational generator model is an unsupervised probabilistic quantum model. These types of models are often known with the name of Born machines, in analogy with the classical Boltzmann machines [32]. The goal of a variational generator is learning a probability distribution that generates a given data set. Let $\{x^{(i)}\}_{i=1}^d$ be a dataset of size d sampled from a probability distribution $p(x)$ that is unknown. The probability distribution is learned as the parametrized probability distribution

$$p_\Theta(x) = |\langle x | \psi(\Theta) \rangle|^2, \quad |\psi(\Theta)\rangle = U(\Theta)|\psi_0\rangle. \tag{2.90}$$

The goal is to minimize the divergence between $p_\Theta(x)$ and $p(x)$. Since the latter is not known, this is achieved by minimizing the negative log-likelihood, which corresponds to minimizing the Kullback-Leibler divergence between the true

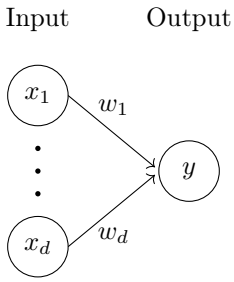


Figure 2.9: **Basic structure of a perceptron**: the perceptron is the fundamental unit of neural networks. It is a non-linear model defined by an activation function φ, which takes as input the vector $\mathbf{x}$ and the weights, and computes the output y.

distribution and the model. Therefore, the cost function is defined as follows

$$f(\Theta) = -\frac{1}{d}\sum_{i}^{d} d\log(p_\Theta(x^{(i)})). \tag{2.91}$$

Quantum Neural Networks Neural networks are a class of machine learning models whose architecture is inspired by the structure of biological brains. Indeed, their basic building blocks are interconnected nodes, which are indeed called neurons, and the connections among these nodes determine the performance of the network. Let $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y}$ be, respectively, the input and output space. The fundamental unit of a neural network is the perceptron (see Fig. 2.9), a non linear model defined by the function

$$\begin{aligned} f : \mathcal{X} &\to \mathcal{Y}, \\ f(\mathbf{x}, \mathbf{w}) &= \varphi(\mathbf{w} \cdot \mathbf{x}), \end{aligned}$$

where $\mathbf{x} \in \mathcal{X}$ and $\mathbf{w} \in \mathbb{R}^d$ are respectively the input datas and the weights vector. In particular, $\varphi : \mathbb{R} \to \mathbb{R}$ is a non-linear function called activation function. Many perceptrons can be combined into the more powerful structure of a neural network.

It is widely believed that replicating the architecture of classical neural networks on quantum hardware could potentially lead to significant advancements. In particular, quantum hardware offer opportunities such as submitting superpositions of inputs and creating entanglement among neurons [23]. However, implementing a quantum analogue of the perceptron presents substantial challenges. For instance, Schuld et al. proposed a model of quantum perceptron that relies on the Quantum Fourier Transform (QFT) [33]. However the QFT is currently unfeasible on NISQ devices. Therefore, different frameworks must be explored.

At present, the standard approach for constructing quantum neural networks is through variational quantum algorithms. However, a fundamental challenge arises from the fact that quantum mechanics is not a natural choice for neural networks. Indeed, non-linearity is an essential characteristic of classical neural network architectures, whereas quantum systems are governed by linear and unitary dynamics. This discrepancy has led to one of the central research directions in the field: the development of a coherent framework capable of reconciling the non-linear behavior typical of classical neural networks with the linear and unitary dynamics characteristic of quantum circuits [34]. In response to this challenge, a growing body of research is currently focused on exploring new strategies to introduce effective nonlinearities into quantum algorithms, with the goal of overcoming this gap and improving the performance of quantum neural networks[35].

2.3.3 Training a VQA

As discussed in the previous section, training a Variational Quantum Algorithm means finding the parameter Θ that minimizes the cost function. Although some gradient-free optimization methods have been proposed, gradient-based methods are more commonly used since, for general cost functions, they are more efficient if the gradient can be accessed directly [36]. In particular, the most widely used strategy is the gradient descend method: at each iteration the parameters are updated in the direction that minimizes gradient. Calculating the gradient means calculating the partial derivative of the cost function with respect to each of the elements of Θ. A first straightforward approach to approximate such partial derivatives is the finite-difference method

$$\frac{\partial f(\Theta)}{\partial \theta_l} \approx \frac{f(\Theta) - f(\Theta + \Delta)}{\|\Delta\|}, \tag{2.92}$$

where $(\Delta)_j = \delta_{i,l}\epsilon$. However, such direct approach has some unavoidable drawbacks, especially when the minimum has to be approached closely, when the optimization landscape has many saddle point, and when the output of the circuit has a high variance [26].

In such situations, having access to the analytical gradient would greatly improve the likelihood of the routine's success. In general, finding an analytical expression for the gradient is not guaranteed. Indeed, if one computes the partial derivative of $f(\Theta)$ the result is not an expectation value anymore [26]. Luckily, there is a way to overcome this issue. In many situations, and especially in the most relevant ones, it is possible to show that the gradient can be computed analytically by measuring the outcome of the circuit multiple times with shifted parameters. This method is defined as *parameter-shift rule*.

Definition 2.12. *(Parameter-shift rule) Let* $f(\Theta) = \langle\psi_0|U^\dagger(\Theta)MU(\Theta)|\psi_0\rangle$, *and let* Θ *be a vector of classical parameters. The parameter-shift rule is the*

identity

$$\frac{\partial f(\Theta)}{\partial \theta_l} = \sum_i a_i f(\Theta + \Delta^{(i)}) \tag{2.93}$$

where $(\Delta^{(i)})_j = \delta_{l,j} s_i$, *and* $\{a_i\}$ *and* $\{s_i\}$ *are real numbers.*

One could argue that equation (2.93) reminds of the finite-difference method, but this is not the case. A first difference is the fact that the shifts $\{s_i\}$ are not infinitesimal. This, by itself, is already a significant improvement to (2.92) where, obtaining a meaningful result requires $\epsilon \ll 1$. Indeed, the usefulness of such shifts is nullified by the noise in present-day quantum hardware. Moreover, a quantum computer can only estimate the expectation value, therefore the biggest difference between (2.93) and (2.92) is that the former computes the estimate of the analytic gradient, while the latter outputs the estimate of the approximate gradient.

An easy way to see the difference between the two methods is considering the function $f = \sin x$, whose first order derivative is $f'(x) = \cos x$. Making use of the trigonometric identity $2 \sin s \cos x = \sin(x + s) - \sin(x - s)$ we obtain the parameter-shift rule that computes the exact derivative

$$\cos x = \frac{\sin(x + s) - \sin(x - s)}{2 \sin s}. \tag{2.94}$$

This can be compared to the finite-difference method which gives the result

$$\cos x = \frac{\sin x - \sin(x + \epsilon)}{\epsilon} + O(\epsilon^2). \tag{2.95}$$

Even though in the following chapters we focus exclusively on the gradient, is worth mentioning that a parameter-shift rule can also be applied to obtain higher-order derivatives [37].

Barren Plateaus

Origins and definition All the information we have gathered up to this moment suggests that VQAs may actually be able to demonstrate quantum advantage in the near future. Despite these encouraging evidence, the search for optimal solutions remains hindered by an additional challenge. Under given circumstances, it may happen that the cost function results flat in wide intervals in the parameters space, far from the minimum Θ^*. This flatness is due to the fact that the gradient of the cost function is exponentially suppressed with the number of qubits and layers, in regions that do not contain the solution, causing gradient-based optimization methods to fail. Such phenomenon is called barren plateau (BP), and it may nullify the advantages that VQAs are supposed to

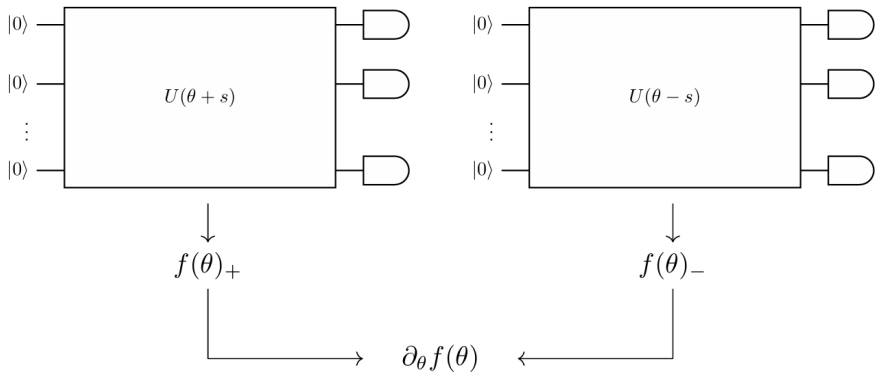


Figure 2.10: **Parameter-shift rule**. Graphical illustration of the computation of a partial derivative of the cost function using the two-term parameter-shift rule. A one-parameter ansatz is considered. The quantum circuit is evaluated twice, once for each shifted parameter value, and the corresponding cost function values are measured. The partial derivative is then obtained as a linear combination of the two expectation values.

bring. Indeed, a vanishing gradient means that optimization is exponentially slow since detecting the descent direction of the gradient would require high precision measurement. Yet, as discussed in the previous section, high-precision is not advisable for NISQ devices. What is even more interesting is that, contrary to what one would think, gradient-free optimization methods do not solve the barren plateau problem [38].

Before diving into the meaning of such phenomenon we give a more rigorous definition of the barren plateau: a cost function is said to exhibit a barren plateau when its gradient concentrates exponentially in the number of qubits around a zero mean value. It is possible to distinguish two types of concentration of the partial derivatives [39]. The first is defined as follows.

Definition 2.13. *(Probabilistic barren plateau) Let $f(\Theta)$ be the cost function defined as in (2.69), and let the parameters Θ be sampled from the uniform distribution over the parameter space. Then $f(\Theta)$ exhibits a probabilistic barren plateau if*

$$\mathrm{Var}_\Theta[\partial_{\theta_l} f(\Theta)] \in O\left(\frac{1}{b^n}\right), \tag{2.96}$$

for some $b > 1$ and for some $\theta_l \in \Theta$.

Notice that we do not need Eq. (2.96) to hold for all $\theta_l \in \Theta$. A straightforward

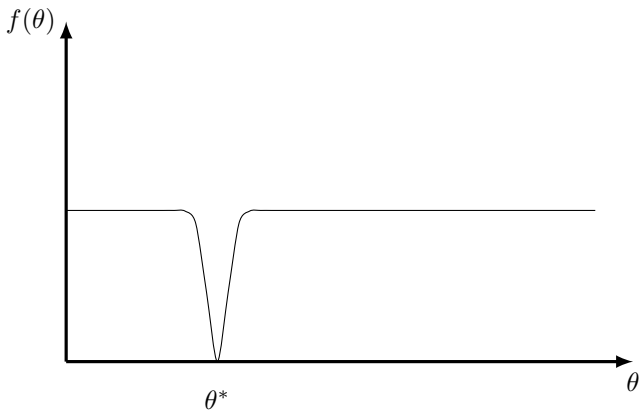


Figure 2.11: **Barren plateau phenomenon**. The figure shows the landscape of a cost function exhibiting probabilistic barren plateau. The landscape of the cost function is mostly flat, and the minimum is located in a narrow region with non vanishing gradient called narrow gorge.

consequence of Def. 2.13, is the following inequality

$$\mathbb{P}(|\partial_{\theta_l} f(\Theta) - \mathbb{E}_\Theta[\partial_{\theta_l} f(\Theta)]| \geq \delta) \in O\left(\frac{1}{b^n}\right), \quad \forall\, \delta > 0. \tag{2.97}$$

Thus, the probability that the partial derivatives of the cost functions deviates from their means by more than δ is exponentially small.

Def. 2.13 is an average statement about the landscape of the cost function. Hence, while being mostly flat, it may still contains limited regions with steep gradients near certain minima. These areas are often referred to as fertile valleys [40], or narrow gorges [41]. Figure 2.11 illustrates a one-dimensional example of a typical cost function landscape influenced by probabilistic BP. A second type of barren plateau can be identified which, unlike the first, does not allow for the presence of narrow gorges and is known as deterministic BP.

Definition 2.14. *(Deterministic barren plateau) Let $f(\Theta)$ be the cost function defined as in (2.69), and let the parameters Θ be sampled from the uniform distribution over the parameter space. Then $f(\Theta)$ exhibits a deterministic barren plateau if*

$$|\partial_{\theta_l} f(\Theta) - \mathbb{E}_\Theta[\partial_{\theta_l} f(\Theta)] \in O\left(\frac{1}{b^n}\right), \tag{2.98}$$

for some $b > 1$ and for all $\theta_l \in \Theta$.

Considering that the most common type of situation is the one described in Def. 2.13, we focus on probabilistic barren plateaus.

While the presence of a barren plateau can be readily identified for a given problem, predicting a priori whether a specific algorithm is susceptible to this phenomenon remains a significant challenge. Thus, substantial research efforts have been focused on understanding the conditions under which barren plateaus emerge and the underlying reasons of such phenomenon. To understand better the reasons behind the BP we need, in the first place, to understand what a unitary t-design, or simply t-design, is.

Definition 2.15. *A unitary t-design is a distribution $\{p_k, U_k\}$ over a set of unitaries $U \in \mathcal{U}(d)$ with probabilities p_k such that the average over polynomials $P_t(U_k)$ up to t-th degree in the elements of the unitary and its conjugate transpose are equal to averages over the Haar measure $\mu(U)$ of the unitary group,*

$$\sum_k p_k P_t(U_k) = \int_{\mathcal{U}(d)} P_t(U)\, d\mu(U). \tag{2.99}$$

In other words, a t-design is a probability distribution over $\mathcal{U}(d)$, such that averaging a polynomial P_t on the distribution is equal to averaging the same polynomial over the uniform distribution over the whole unitary group. The usefulness of t-designs comes from the fact that, while sampling from the Haar measure over the unitary group is computationally expensive, t-designs allow us to evaluate complex expressions sampling from a subset of the whole group, and therefore reducing the computational cost.

Barren plateau for deep variational ansätze Considering that we are interested in the variance of the gradient of the cost function, we focus our attentions on 2-designs. Random circuits are often proposed as initial guesses for quantum algorithms, and it has been shown that such circuits generate approximate unitary t-designs at a depth of $O(nt^{5+o(1)})$ [42]. We say that a given circuit $U(\Theta)$ is deep, if the number of layers is big enough to guarantee that the distribution of unitaries corresponding to random parameter choices forms an approximate 2-design [39]. The first article that highlighted the emergence of the BP phenomenon [13], demonstrated that the latter arise for deep random circuits. More formally Ref. [13] shows that the following Theorem holds.

Theorem 2.1. *Consider a quantum circuit $U(\theta)$ with n qubits defined as*

$$U(\theta) = U_- e^{-i\theta H} U_+, \tag{2.100}$$

where H is Hermitian. The circuits U_- and U_+ are independent and either one of them or both are a 2-design. Then

$$\mathrm{Var}[\partial_\theta f(\theta)] = O(2^{-n}). \tag{2.101}$$

Where we used the formulation of Ref. [26]. Therefore, for expressive ansätze such as the ones that forms 2-designs, the barren plateau cannot be avoided.

A second result obtained in Ref. [14], links the BP to the expressibility of $U(\Theta)$.

Theorem 2.2. *Let $\mathbb{U}$ be a subset of unitaries. Let*

$$\mathcal{A}_{\mathbb{U}}^{(t)}(\cdot) \coloneqq \int_{\mathcal{U}(2^n)} V^{\otimes t}(\cdot)(V^\dagger)^{\otimes t} d\mu(V) - \int_{\mathbb{U}} U^{\otimes t}(\cdot)(U^\dagger)^{\otimes t} dU, \tag{2.102}$$

represent the distance of $\mathbb{U}$ from being a t-design. If $\mathcal{A}_{\mathbb{U}}^{t} = 0$ then $\mathbb{U}$ is a t-design. Let the ansatz be a bipartite parametrized ansatz,

$$U(\Theta) = U_-(\Theta)U_+(\Theta) \tag{2.103}$$

where $U_-(\Theta)$ and $U_+(\Theta)$ are defined as in (2.72), and let $\mathbb{U}_-$ and $\mathbb{U}_+$ be the ensembles associated with the two. The quantities that capture the expressibility of the circuit with respect to the starting state and the observable are

$$\mathcal{E}_+^{\psi_0} \coloneqq \|\mathcal{A}_{\mathbb{U}_+}^{(2)}(\rho_0^{\otimes 2})\|_2, \tag{2.104}$$

and

$$\mathcal{E}_-^{M} \coloneqq \|\mathcal{A}_{\mathbb{U}_-}^{(2)}(M^{\otimes 2})\|_2, \tag{2.105}$$

for $\rho_0 = |\psi_0\rangle\langle\psi_0|$. Then, for a cost function of the type (2.69):

$$\mathrm{Var}[\partial_{\theta_l} f(\Theta)] \leq O(2^{-n}) + g(\mathcal{E}_+^{\psi_0}, \mathcal{E}_-^{M}), \tag{2.106}$$

where we define

$$g(x,y) = 4xy + \frac{2^{n+2}(x\|M\|_2^2 + y\|\rho_0\|_2^2)}{2^{2n}-1}. \tag{2.107}$$

The norm $\|\cdot\|_2$ denotes the Frobenius norm. The result gives us a measure of the barren plateau depending on the distance $\mathbb{U}_+$ and $\mathbb{U}_-$ are from being a 2-design. A similar result is found when we consider the distance from being a 2-design of only one of $\mathbb{U}_\pm$.

We can immediately understand the connection between barren plateaus and circuit expressibility: the more an ansatz explores fully and uniformly the unitary space the more the barren plateau is likely to arise. This connection is clearly against what one would hope for, since the ideal situation would be to have an expressive ansatz that is able to guarantee a solution for a wide range of problems. However, if we think about it for a moment, the correlation is not so improbable. In fact, Eq. (2.69) can be expressed in a Hilbert-Schmidt product formulation

$$f(\Theta) = \mathrm{Tr}[MU(\Theta)\psi_0 U(\Theta)^\dagger] = \langle M, U(\Theta)\psi_0 U(\Theta)^\dagger\rangle_{HS}, \tag{2.108}$$

Thus, roughly speaking, minimizing the cost function means trying to anti-align two vectors in an extremely large Hilbert space which is not an easy task. Indeed, one can imagine that the starting state is sequentially rotated by some random chosen rotations across all dimensions: any single step does not have great effects on the overall final position [26].

Barren plateau for unstructured variational ansatz The results presented above are limited to ansätze that are either exact or approximate 2-designs. While the former is highly unrealistic, even the latter can be restrictive, as an ansatz does not necessarily need to form an approximate 2-design to exhibit barren plateau. Together with these findings, it has been shown that local observables can avoid the appearance of barren plateaus [16]. To address and blend these results, we present a more general theorem from Napp [15], which establishes a connection between the flatness of the loss landscape and the architectural parameters of the circuit, such as the number of layers, the regular connectivity[10], the number of qubits, and the locality of the observable.

To present the results obtained by Napp, we first recall that any observable can be decomposed as a sum of Pauli operators. The generalization of Eq. (2.43) to n-qubit systems is the following:

$$M = \sum_{\mathbf{x}} c_{\mathbf{x}} M_{\mathbf{x}}, \tag{2.109}$$

where each $M_{\mathbf{x}}$ is the tensor product of n single-qubit operators drawn from $\mathcal{P}$

$$M_{\mathbf{x}} = \sigma_{x_1} \otimes \ldots \otimes \sigma_{x_n}, \tag{2.110}$$

for $\mathbf{x} = (x_1, \ldots, x_n)$ and $x_i \in \{0, 1, 2, 3\}$. We assume without loss of generality that $c_0 = 0$ and we define $|\mathbf{x}|$ ad the number of non zero elements of $\mathbf{x}$. Thus $|\mathbf{x}|$ represents the locality of the operator.

Assumption 2.2. *Consider a parametrized HEA $U(\Theta)$ with the following types of gates:*

i. Random entangling gates. They act non trivially on two, possibly non adjacent, sites and are chosen independently and uniformly at random from the Haar measure over $\mathcal{U}(4)$.

ii. Parametrized gates, of the form $W_l e^{i\theta_j A_j} W_r$ where each A_j is Hermitian and acts non trivially on at most two sites, θ_j is the j-th element of the parameter vector Θ and W_j^l and W_j^r are arbitrary fixed gates acting on the same sites as A_j.

Moreover, we assume the following constraints on the architecture:

i. Each qubit is acted upon by an entangling two-qubit gate at least once.

ii. Each entangling two-qubit gate acting on qubits i and j may be preceded and succeeded by an arbitrary number of parameterized gates acting on one or both of these sites, but parameterized gates which cannot be placed in this way are not allowed.

[10] The regular connectivity defines the maximum number of layers of parallel gates that must be applied before some gate acts between an arbitrary proper subset of qubits and its complement.

Finally, let $\tilde{U}$ be the circuit $U(\Theta)$ with the parametrized gates removed. We define l and r as the depth and the regular connectivity of $\tilde{U}$.

To study the barren plateau phenomenon in this unstructured setting, Napp establishes a connection between the flatness of the loss landscape and a certain family of random walks, which were also analyzed in [43].

Definition 2.16. *For a variational circuit $U(\Theta)$ defined as in assumption 2.2 we define the associated random walk $\mathcal{M}_{\tilde{U}}$ as follows:*

i. Each site is initialized independently with label S with probability $\frac{1}{3}$ and with label I otherwise.

ii. Each entangling gate in $\tilde{U}$ "acts" on a pair of sites as follows:

(a) If the two sites are in the configuration (I, I) or (S, S), then the gate leaves the configuration unchanged.

(b) If the two sites are in the configuration (I, S) or (S, I) then the gate flips the sites to the configuration (S, S) with probability $\frac{1}{5}$ and to (I, I) otherwise.

For any realization V of the entangling gates, the quantity studied by Napp to estimate the presence of barren plateaus is $\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2$, which serves as an alternative to the more natural and commonly used measure of gradient flatness, $\mathbb{E}_V \mathbb{E}_\Theta \|\nabla f_V(\Theta)\|^2$. As shown in [15], these two quantities are directly related, making the former a more tractable way to analyze the latter. As proved in [15], the expectation can be expressed as

$$\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2 = \sum_{\mathbf{x}} |c_{\mathbf{x}}|^2 g_{\mathbf{x}}, \tag{2.111}$$

for $g_{\mathbf{x}} = \langle 0^n | U(\Theta)^\dagger M_{\mathbf{x}} U(\Theta) | 0^n \rangle^2$. This finding is detailed and proved in [15]. Napp's results are valid for qudits[11] of dimension greater than or equal to 2. However, since in chapter 4 we focus on the qubit case, we also present the following theorem specialized to qubits.

Theorem 2.3. *Consider an n qubit circuit with an ansatz $U(\Theta)$ defined as in assumption 2.2, then*

$$g_{\mathbf{x}} \geq \max \left\{ \left(\frac{1}{3}\right)^{\mathbf{x}} \left(\frac{1}{5}\right)^{l|\mathbf{x}|}, \left(\frac{1}{3}\right)^n \right\}, \tag{2.112}$$

and

$$g_{\mathbf{x}} \leq \left(\frac{4}{3}\right)^n \left(\frac{4}{5}\right)^{\lfloor l/r \rfloor} 2^{-|\mathbf{x}|} + (2^n - 1)^{-1}. \tag{2.113}$$

[11] Qudits are a generalization of qubits. They are defined in a Hilbert space of dimension $d \geq 2$.

The theorem sets a lower and an upper bound for the flatness of the landscape. In particular the lower bound shows that the landscape is no flatter than an exponential in the product of the depth and locality. The second term on the right side of (2.112) refers to the case when the starting configuration of the random walk is S^n. The upper bound instead, implies the following corollary.

Corollary 2.1. *For* $\mathbf{x}$ *such that* $|\mathbf{x}| \geq n/2$

$$g_{\mathbf{x}} \leq 2^{-\Theta(1)\cdot n}. \tag{2.114}$$

While, for $\mathbf{x} \neq 0$ *and for* $l > l^* = \Theta(1) \cdot rn$*, Eq.* (2.112) *takes the following form*

$$g_{\mathbf{x}} \leq 2^{-n} + 2^{-\Theta(1)\cdot(l-l^*)/r-|\mathbf{x}|}. \tag{2.115}$$

Equation (2.114) states that barren plateau is always obtained for global observables, simplifying the results of [16]. Equation (2.114) instead, shows that the magnitude of the gradient decays exponentially fast to 2^{-n} in the depth once the latter exceeds $O(r \cdot n)$.

Is it possible to escape barren plateaus? Together with deep circuits and global observables, other situations that are prone to induce BP in the optimization problem are noisy circuits [19], [20] and excess of entanglement [17], [18]. Ref. [39] argues that all these causes can ultimately be attributed to what Ref. [21] named a "*curse of dimensionality*", which refers to the fact mentioned above: both the evolved state and the observable lay in an extremely large Hilbert space.

Several strategies have been proposed to mitigate the causes of BP. A first approach would be to use problem-inspired ansätze, but as we argued, it is not always possible. Luckily, this is not the only strategy proposed, others rely on embedding symmetries into the circuit's architecture [44] or dynamics with small Lie algebras [45]. Theorem 2.3 shows that the barren plateau can be avoided if a local observable is used instead of a global one and the number of layer is kept shallow. Thus, the barren plateau can be avoided if the observable is chosen with care, a fact that restores the hope placed in VQAs. Another meaningful case is the one of warm-starts [46]. These are smart initialization strategies whose scope is initializing the state nearby the minimum in a region that corresponds to a "patch" (namely, small region) with guaranteed gradients [47]. These regions, correspond to the ones that we previously named narrow gorges but, as the study argues, the name is misleading since the width of these regions vanishes at worst polynomially in the number of trainable parameters and not exponentially as it was initially believed. The hope behind these strategies lies in the fact that smart initializations have been a well-known tool for avoiding gradient issues in classical machine learning.

However, a question arises, and it is wether the absence of BP implies classical simulability. Ref. [21] argues that the answer, in many cases, is yes. Cerezo

et al. studied a wide range of situations in which the barren plateau has been avoided by means of tricks that make use of some simple underlying structure of the problem. The results shows that in many cases the same structure used to avoid barren plateaus can be used to efficiently calculate the cost function classically. Indeed, the strategies employed to avoid BP analyzed by Ref. [21], all result in a restriction of the space explored by the ansatz. In particular the latter gets reduced from an exponentially large space, the whole unitary group, to a polinomially large subspace. This implies that the observable and the evolved state are as well polynomially large objects in the given subset, therefore classically simulable. More recently, these claims have been supported by a paper by Angrisani et al., which presents a classical algorithm for estimating the expectation values of arbitrary observables on most quantum circuits for most values of the parameters, in any circuit architecture, where each circuit layer is sampled according to a probability distribution that is invariant under single-qubit rotations. This result shows that, for a large class of quantum circuits, it is possible to classically compute the associated expectation values to within a small additive error [48]. The properties of circuits in this class are satisfied by a wide range of deep and shallow unstructured parameterized quantum circuits of different topologies currently used by variational quantum algorithms, including some that claim to avoid barren plateaus while escaping classical simulability.

This does not imply that the work carried out to investigate the causes and triggering factors of the barren plateau phenomenon should be disregarded. Indeed, Ref. [21] and [48] do not cover all possible variational algorithms; moreover, although rare, there are still cases where barren plateaus can be avoided without falling into regimes of classical simulability, as shown in the same [21]. Additionally, it may happen that even when polynomial-time classical simulation is possible, the associated computational cost may remain prohibitive, thus allowing for potential polynomial quantum advantages.

In summary, the study of the barren plateau phenomenon remains relevant. Not only do algorithms exist that can both avoid barren plateaus and remain hard to simulate classically, but understanding the conditions under which BP can be avoided may also help identify cases where quantum computation offers no real advantage, allowing us to focus our efforts on regimes where a real quantum speedup is possible.

In the following chapters, we focus on the architectural characteristics of an ansatz that lead to a flat landscape. Our goal is the same of Ref. [15], which is establishing a connection between the flatness of the loss and the circuit architecture. While aiming at the same target, we approach the problem in a different way, making use of the convenient properties of the Clifford group, which we analyze in chapter 3.

Chapter 3

The Clifford group and its action on the Pauli group

The n-qubit Clifford group $\mathcal{C}_n$ is defined as the normalizer of the Pauli group $\mathcal{P}_n$ in $\mathcal{U}(2^n)$:

$$C_n = \{V \in \mathcal{U}(2^n) | V^\dagger \mathcal{P}_n V \subseteq \mathcal{P}_n\}/U(1). \tag{3.1}$$

By definition, the Pauli group $\mathcal{P}_n$ is a normal subgroup of $\mathcal{C}_n$. The importance of the Clifford group lies in several factors. It has a wide range of applications such as quantum error-correcting codes [49], and quantum data hiding [50]. Furthermore, operations within the Clifford group can be efficiently simulated on a classical computer, a fact that makes tractable working with relatively large Clifford group circuits [51]. Beside all that, the property that makes the Clifford group fundamental for our work is the fact that it is a unitary 2-design. In addition, it is also a 3-design [52] but fails to be a 4-design [53]. In this chapter we discuss the action of the Clifford group over the Pauli group. These results serve as a foundation for the next chapter.

3.1 The action of the Clifford group

The Clifford group acts on the n-qubit Pauli group $\mathcal{P}_n$ via conjugation. Since the conjugation operation preserves the spectrum of the operator it acts upon, the elements of the Clifford group map each non-identity hermitian operator into another non-identity hermitian operator, while $\pm\mathbb{I}$ can only be conjugated into itself.

Definition 3.1. *$\mathcal{P}_n^*$ is the set of all the hermitian elements of the Pauli group that are not proportional to the identity:*

$$\mathcal{P}_n^* = \{P \in \mathcal{P}_n | P = P^\dagger\} \setminus \{\pm\mathbb{I}\}. \tag{3.2}$$

For our purpose it is necessary to show that the operator resulting from the conjugation of $P \in \mathcal{P}_n^*$ through $C \in \mathcal{C}_n$ is sampled from the uniform distribution on $\mathcal{P}_n^*$. To do so, we need the following Theorem.

Theorem 3.1. *There exists exactly one orbit of $\mathcal{P}_n^*$ under the action of $\mathcal{C}_n$.*

Proof. In the one-qubit case it is easy to show the existence of a single orbit. Indeed, the following relations hold:

i. $H^\dagger X H = Z$,

ii. $H^\dagger(-X)H = -Z$,

iii. $H^\dagger Y H = -Y$,

iv. $S^\dagger X S = Y$,

v. $S^\dagger Y S = -X$,

vi. $S^\dagger(-Y)S = -X$.

Therefore, any operator of $\mathcal{P}_1^*$ is conjugate to any other element of $\mathcal{P}_1^*$ by means of S and H, the phase and Hadamard gate, which are the generators of $\mathcal{C}_1$.

The same can be shown when $n = 2$. For this purpose it is necessary to introduce the CNOT gate which, along with S and H, is one of the generators of $\mathcal{C}_n$ with $n \geq 2$. Moreover, to prove the Theorem when $n \geq 2$, it is not important whether a single-qubit operator within an element of $\mathcal{P}_n^*$ is a Pauli X, Y, or Z; what matters is only whether it is equal to the identity or not. This is due to the fact that the transitivity of the action for $n = 1$, guarantees that every element of $\mathcal{P}_1^*$ is conjugate to every other element of the set by means, respectively, of a Clifford operator of the type $\mathbb{I} \otimes J$, $J \otimes \mathbb{I}$ or a product of both, where $J = \{H, S\}$. To simplify the proof, it is convenient to consider any operator of $\mathcal{P}_1^*$ as equivalent and denote it generically by W. The equivalence relation induces a partition in $\mathcal{P}_n^*$, where each class contains all the n-fold tensor products acting non trivially on the same qubits but using different single-qubit operators. $\mathcal{P}_2^*$ is partitioned into the following classes:

i. $\mathbb{I} \otimes W$;

ii. $W \otimes W$;

iii. $W \otimes \mathbb{I}$;

To prove the existence of a single orbit it is necessary to show that it is possible to conjugate at least one element from a given class to one in the previous class. This can be done using the CNOT operator: it allows at least one tensor product belonging to the first class to conjugate to a tensor product in the second class and one in the second to conjugate to one in the third:

$$\mathrm{CNOT}^\dagger(\mathbb{I} \otimes Z)\mathrm{CNOT} = Z \otimes Z,$$

$$\mathrm{CNOT}^\dagger(X \otimes X)\mathrm{CNOT} = X \otimes \mathbb{I}.$$

It follows that any operator in $\mathcal{P}_2^*$ is conjugate to any other operator in $\mathcal{P}_2^*$ under the action of some Clifford operator, proving the existence of a single conjugacy class.

In order to generalize this approach to n-qubits, it is sufficient to show that, given an element of $\mathcal{P}_n^*$, there exists a path connecting it to any other element of $\mathcal{P}_n^*$ by conjugation via Clifford operators. Since X, Y, and Z are equivalent and represented as W, it is possible, in order to avoid the use of heavy notation, to represent each n-fold tensor product of different equivalence classes of $\mathcal{P}_n^*$ as a string of n bits: '1' represents W and '0' the identity operator. The positions of the 1s and the 0s in the bit string indicate the sites acted upon by the operators Ws and the $\mathbb{I}$s, respectively. Obviously, the all '0' string is not allowed since it would represent the identity operator which is not an element of $\mathcal{P}_n^*$. The existence of a single orbit for $n = 1$ implies that all the bit strings of a given class are conjugate. On the other hand, the case $n = 2$ implies that the bit strings from two different classes are conjugate if they differ from each other by no more than a pair of (not necessary neighboring) bits within the string, such that, in both strings, at least one bit of this pair is '1'.

The existence of a single orbit can be demonstrated showing that any given string p is conjugate to the string with '1' in the first position and '0' everywhere else trough a finite sequence of n-bit strings p_l. To this scope let p be any of the possible n-bit strings and suppose that p has k nonzero bits.

i. Let the i-th non-zero bit be the first bit in p_l such that that is equal to 1 and preceded by a 0. If there is no such bit, p_l already has the desired structure.

ii. Consider the i-th non-zero bit and the zero bit at position i.

iii. Define the string p_{l+1} by swapping the positions of these two bits in p_l.

iv. Repeat points 1 to 3 for every $i = 1...k$.

The second part generates the string with '1' in the first position followed by $n - 1$ zeroes.

i. Starting from $i = k$, consider the i-th and $(i-1)$-th non zero bits in position i and $i-1$.

ii. Define the new string by changing the value of the i-th bit from '1' to '0'.

iii. Repeat points 1 and 2 for every $i = k...2$.

This means that the element of $\mathcal{P}_n^*$ which acts non trivially only on the first qubit is conjugate to any other element of the set, showing the existence of a single orbit of $\mathcal{P}_n^*$ under the action of $\mathcal{C}_n$.

□

We are now ready to present the main result.

Theorem 3.2. *Let φ represent the action of $\mathcal{C}_n$ over $\mathcal{P}_n^*$:*

$$\begin{aligned} \varphi : \mathcal{C} \times \mathcal{P}_n^* &\to \mathcal{P}_n^*, \\ (C, P) &\longmapsto C^\dagger P C, \end{aligned}$$

and let C be sampled from the uniform distribution on $\mathcal{C}_n$. Then, for any $P \in \mathcal{P}_n^*$, $\varphi(C, P)$ is uniformly distributed over $\mathcal{P}_n^*$.

Proof. A direct consequence of Theorem 3.1 is that the cardinality of this one orbit is $|\mathcal{P}_n^*|$. From the orbit-stabilizer Theorem, we have that for every $P \in \mathcal{P}_n^*$ the following holds:

$$|\text{orb}(P)||\text{stab}(P)| = |\mathcal{C}_n|, \tag{3.3}$$

$$|\mathcal{P}_n^*| = \frac{|\mathcal{C}_n|}{|\text{stab}(P)|},$$

$$|\mathcal{P}_n^*| = [\mathcal{C}_n : stab(P)], \tag{3.4}$$

where $[\mathcal{C}_n : \text{stab}(P)]$ is the number of right cosets of $\text{stab}(P)$ in $\mathcal{C}_n$. A right coset of $\text{stab}(P)$ si defined as:

$$\text{stab}(P)C = \{SC \,|\, S \in \text{stab}(P)\}. \tag{3.5}$$

Every element in $\text{stab}(P)C$ acts on a given $P \in \mathcal{P}_n^*$ as follows:

$$(SC)^\dagger P (SC) = C^\dagger S^\dagger P S C = C^\dagger P C = P', \tag{3.6}$$

where the second equality comes from the properties of the stabilizer. An element $C' \in \mathcal{C}_n$ belongs to the same right coset as C, namely $\text{stab}(P)C$, if $C' = SC$ for some $S \in \text{stab}(P)$. Hence, the following equality holds true:

$$(C')^\dagger P C' = (SC)^\dagger P (SC) = P'. \tag{3.7}$$

Therefore, each right coset contains exactly all the elements of $\mathcal{C}_n$ that act on P in the same way, that is, the elements of of the n-qubit Clifford group that conjugate P into a fixed $P' \in \mathcal{P}_n^*$. The result in equation (3.4) means that for every $P \in \mathcal{P}_n^*$ the number of different elements of $\mathcal{P}_n^*$ in which it can be conjugated is exactly $|\mathcal{P}_n^*|$.

The cardinality of each right coset of $\text{stab}(P)$ is $|\text{stab}(P)|$. This means that, for any P, P', the number of elements of $\mathcal{C}_n$ that map P in P' is a fixed number:

$$|\{C \in \mathcal{C}_n | C^\dagger P C = P'\}| = |\text{stab}(P)| = \frac{|\mathcal{C}_n|}{|\mathcal{P}_n^*|}. \tag{3.8}$$

This leads to the following relation which concludes the proof:

$$\begin{aligned} \mathbb{P}(P' = C^\dagger P C) &= \frac{|\{C \in \mathcal{C}_n | C^\dagger P C = P'\}|}{|\mathcal{C}_n|} \\ &= \frac{|\mathcal{C}_n|/|\mathcal{P}_n^*|}{|\mathcal{C}_n|} \\ &= \frac{1}{|\mathcal{P}_n^*|}. \end{aligned} \tag{3.9}$$

□

A direct consequence of Theorem 3.2 is the following corollary.

Corollary 3.1. *Let C be sampled from the uniform distribution over $\mathcal{C}_n$ and P be an element of $\mathcal{P}_n^*$. Let $P' = C^\dagger P C$. Then each non-identity factor in the tensor product defining P' is independently distributed over $\{X, Y, Z\}$ with uniform probability $1/3$.*

3.1.1 The action of the Clifford group over $\mathcal{P}_2^*$

We briefly discuss the case where $n = 2$, since it is the one that matters the most for our results. The set $\mathcal{P}_2^*$ contains 30 elements; therefore, any element of $\mathcal{P}_2^*$ acted upon by an operator C sampled uniformly from $\mathcal{C}_2$, is mapped uniformly over those 30 elements.

Let W represents any non-identity Pauli matrix. The equivalence relation among X, Y, and Z induces a partition in $\mathcal{P}_2^*$, where each class contains all the 2-fold tensor products acting non trivially on the same qubits but using different single-qubit Pauli operators. Then we can see that $\mathcal{P}_2^*$ is partitioned into the following classes:

i. $\pm\mathbb{I} \otimes W$;

ii. $\pm W \otimes W'$;

iii. $\pm W \otimes \mathbb{I}$.

Proposition 3.1. *For any $P \in \mathcal{P}_2^*$, we have the following*

$$\mathbb{P}(C^\dagger PC \in [\pm\mathbb{I} \otimes W]) = \frac{1}{5}; \tag{3.10}$$

$$\mathbb{P}(C^\dagger PC \in [\pm W \otimes W']) = \frac{3}{5}; \tag{3.11}$$

$$\mathbb{P}(C^\dagger PC \in [\pm W \otimes \mathbb{I}]) = \frac{1}{5}. \tag{3.12}$$

In each of the above cases, W is uniformly distributed over the set of Pauli operators $\{X, Y, Z\}$.

3.2 Application to quantum circuits

We now show how to exploit the results of the previous sections in a quantum circuit. Assume that V is a quantum Clifford group circuit that acts on n qubits. The starting state is assumed to be $|0\rangle^{\otimes n}$. Let $V = V_p \dots V_1$, where each V_t is salmpled uniformly at random from $\mathcal{C}_2$ acts on two qubits. Note that the gates are applied in the order V_p first, V_1 last, that is, we index gates from the output backwards. Let M, the observable, be an element of $\mathcal{P}_n^*$. For any operator S we briefly introduce the following notation:

- $\mathrm{supp}(S)$ = set of the sites S acts non trivially upon;
- $|\mathrm{supp}(S)|$= is the cardinality of the set $\mathrm{supp}(S)$;
- $S_{i\dots j}$= the restriction of S to the sites from i to j if S is an n-folds tensor product of single-qubit operators.

The expected value of the measurement of M at the end of circuit V is calculated as follows:

$$\langle 0^n|V^\dagger MV|0^n\rangle = \langle 0^n|V_p^\dagger \dots V_1^\dagger MV_1 \dots V_p|0^n\rangle. \tag{3.13}$$

Let us now proceed by evolving the observable rather than the state. If we look at the operator product $V_1^\dagger MV_1$ we can clearly see that this is a conjugation of a $\mathcal{P}_n^*$ operator carried out by a Clifford operator and, depending on the sites V_1 acts upon, we have two possible results:

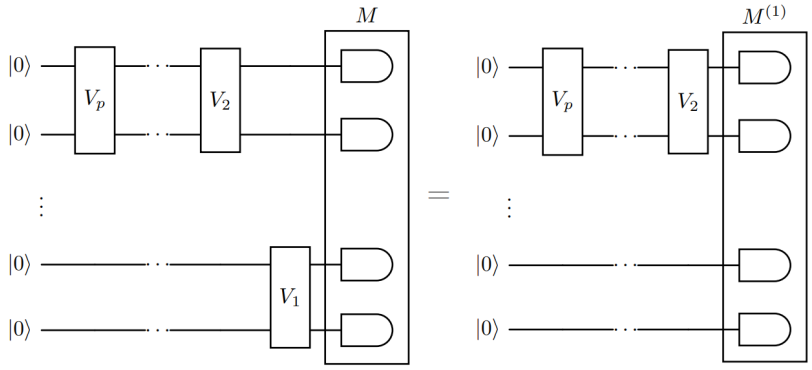


Figure 3.1: **Application to circuits of Theorem 3.2**. We "absorb" the effect of the gate V_1 into the observable defining a new operator $M^{(1)}$. We represent the observable with a measurement gate acting on each qubit separately. This can be done because M is the tensor product of n single-qubit operators, therefore, the outcome of its measurement is equal to the product of the outcome of the measurement of its single-qubit operators, as explained in section 2.2.

$$V_1^\dagger M V_1 = M^{(1)} = \begin{cases} M & \text{if } \operatorname{supp}(M) \cap \operatorname{supp}(V_1) = \emptyset, \\ M' & \text{if } \operatorname{supp}(M) \cap \operatorname{supp}(V_1) \neq \emptyset. \end{cases} \tag{3.14}$$

If we assume that the sites V_1 acts upon are the sites $i, i+1$ for $i = 1 \dots n$, the operator M' is defined by the following tensor product:

$$M' = M_{1\dots i-1} \otimes M'_{i,i+1} \otimes M_{i+2\dots n} \tag{3.15}$$

By Theorem 3.2 the operator $M'_{i,i+1}$ is drawn uniformly from $\mathcal{P}_2^*$. Then we can write (3.13) as follows:

$$\langle 0^n | V_p^\dagger \dots V_2^\dagger M^{(1)} V_2 \dots V_p | 0^n \rangle. \tag{3.16}$$

The same process is done for $M^{(1)}$:

$$V_2^\dagger M^{(1)} V_2 = M^{(2)} = \begin{cases} M^{(1)} & \text{if } \operatorname{supp}(M) \cap \operatorname{supp}(V_2) = \emptyset, \\ M^{(1')} & \text{if } \operatorname{supp}(M) \cap \operatorname{supp}(V_2) \neq \emptyset. \end{cases} \tag{3.17}$$

Assuming again that V_2 acts upon the sites $i, i+1$ for $i = 1 \dots n$, $M^{(1')}$ is given by the following tensor product

$$M^{(1')} = M^{(1)}_{1\dots i-1} \otimes M^{(1')}_{i,i+1} \otimes M^{(1)}_{i\dots 2,n} \tag{3.18}$$

with $M^{(1')}_{i,i+1}$ drawn uniformly from $\mathcal{P}_2^*$.

The same process is iterated for each V_t, and, after performing each conjugation we obtain the following expression for (3.13):

$$\langle 0^n | V^\dagger M V | 0^n \rangle = \langle 0^n | M^{(p)} | 0^n \rangle, \tag{3.19}$$

where $M^{(p)} \in \mathcal{P}_n^*$ and depends on the outcome of the previous p conjugations.

The procedure can be generalized for V_t acting on non-adjacent qubits. In the following chapter we show how to exploit this property to derive an estimate of the variance.

Chapter 4

Quantifying the barren plateau phenomenon

As seen in section 2.3, the barren plateau phenomenon is a significant obstacle to the development of practical quantum speedups in the near term. This is especially true when considering highly unstructured, random-looking ansätze. Our goal, in this chapter, is to develop a simple and straightforward method to evaluate the magnitude of the gradient of the cost function for such an ansatz. Our approach relies on the result obtained in chapter 3, in particular in section 3.2. As motivated in the corresponding chapter, Theorem 3.2 enables us to simplify the expression of the expected value of the measurement of an observable $M \in \mathcal{P}_n^*$. Looking back to section 3.2 we see that we obtain the following equality:

$$\langle 0^n | V^\dagger M V | 0^n \rangle = \langle 0^n | M^{(p)} | 0^n \rangle, \tag{4.1}$$

for any circuit $V = V_1 ... V_p$ whose gates V_t are drawn uniformly at random from the Clifford group.

As we argued in the previous chapter, $M^{(p)} \in \mathcal{P}_n^*$, and therefore is an n-fold tensor product of single qubit Pauli operators where at least one of the n operators differs from the identity, see Fig. 4.1 for an example. If any of these n single-qubit Pauli operators is a Pauli-X or a Pauli-Y, then by the properties of the aforementioned operators and by the properties of tensor products discussed in chapter 2, the expectation value of $M^{(p)}$ for a system starting in the state $|0\rangle^{\otimes n}$ is zero.

In this chapter we see how to use this nice property to estimate the magnitude of the gradient of the cost function. Firstly we introduce a highly-unstructured

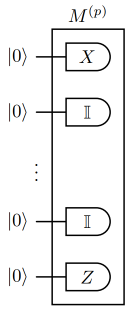


Figure 4.1: **Observable** $M^{(p)}$. The measurement gate representing the observable $M^{(p)}$ that results from (4.1), acts non trivially on one or more sites, since the operator $M^{(p)}$ is an element of $\mathcal{P}_n^*$.

HEAs, similar to the one described in [15], and we obtain the same expression to estimate the barren plateau. Subsequently we focus on two different models: in section 4.3 we assume a random placement of the gates and we quantify the barren plateau with the properties of Markov chains; in section 4.4, instead, we fix the architecture and derive a lower bound on the expected value of the magnitude of $\|\nabla f(\Theta)\|$ using the results of section 4.2.

4.1 Setup and notation

We consider a parametrized circuit $U(\Theta)$ with parameter vector $\Theta = \{\theta_1, \dots, \theta_m\}$, acting on n qubits whose starting state is $|0\rangle^{\otimes n}$.

Assumption 4.1. *We assume that the gates in the parametrized circuit $U(\Theta)$ are of the following two types:*

i. Random entangling two-qubit gates: they act non trivially on two sites, which can possibly be non adjacent, and are sampled independently and uniformly at random from the Haar measure over $U(4)$.

ii. Deterministic parametrized gates of the form $W_j^l e^{-iA_j\theta_j} W_j^r$, where A is Hermitian and acts non trivially on at most two qubits, $\theta \in \mathbb{R}$ is the parameter and W_j^l and W_j^r are arbitrary fixed gates acting on the same sites as A_j.

Moreover, we assume that each A_j has spectrum in $\{-1, 1\}$.

The assumption on the spectrum of the A_js guarantees that each $e^{-iA_j\theta_j}$ is

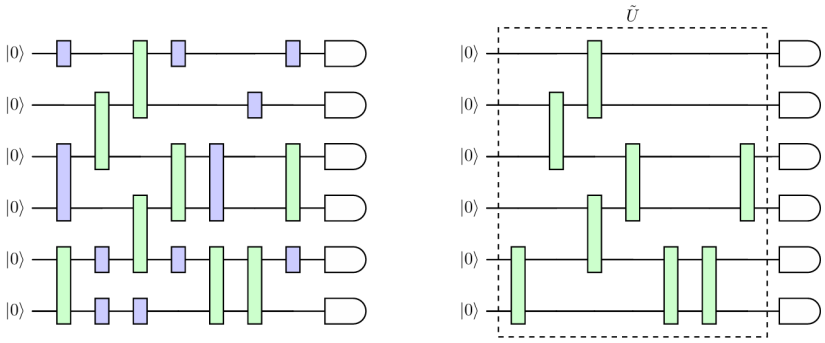


Figure 4.2: **Representation of the ansatz $U(\Theta)$ and $\tilde{U}$.** In green are represented the unparameterized entangling gates. In blue the single and two qubits parametrized gates. We note that, as required by assumption 4.2, each qubit of the circuit is acted upon by at least one 2-qubits entangling gate. On the right is represented $\tilde{U}$, as defined in Def. 4.1.

periodic in θ_j with period 2π, a condition required to prove Lemma 4.1[1]. We additionally demand the model to satisfy some structural constraints.

Assumption 4.2. *We require the model to satisfy the following constraints:*

i. Each qubit is acted upon by an entangling two-qubit gate at least once.

ii. Each entangling two-qubit gate acting on qubits i and j may be preceded and succeeded by an arbitrary number of parameterized gates acting on one or both of these sites, but parameterized gates which cannot be placed in this way are not allowed.

We now introduce the notation that we use throughout the following pages, closely following the one adopted by Napp in [15].

Definition 4.1. *For any variational circuit $U(\Theta)$, define $\tilde{U}$ as the same circuit with all the parametrized gates are removed. V represents a particular realization of the entangling gates in $\tilde{U}$, p the total number of entangling gates, and l the number of layers in $\tilde{U}$.*

If we consider the orthogonal basis of $\mathbb{C}^{2\times 2}$, namely $\mathcal{P} = \{\mathbb{I}, X, Y, Z\}$, any n-qubit operator M can be decomposed as in Eq. (2.109). We consider the specific

[1] The condition is also non restrictive since any single quibt Hermitian gate A with spectrum not in $\{-1, 1\}$ can be rescaled to have its spectrum in $\{-1, 1\}$. On the other hand, in real-life applications, the two-qubit Hermitian gates commonly considered are tensor products of Pauli gates, and thus satisfy the requirement.

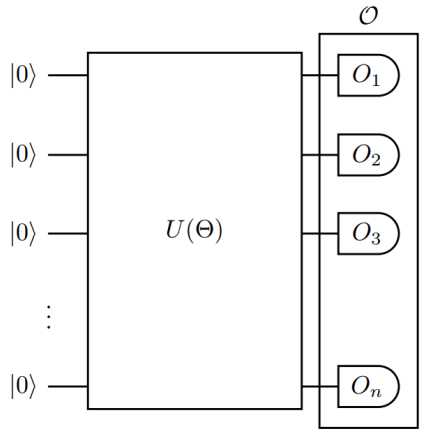


Figure 4.3: **Local Pauli observable**. In the image, each measurement gate represents a local n-qubit Pauli observable defined as in Def. 4.2, that acts non trivially only on the sites on which it is represented.

observable whose decomposition such that each term of the summation acts non trivially on one site only.

Definition 4.2. *$\mathcal{O}$ is a local observable given by the sum of single qubit Pauli observables:*

$$\begin{aligned}\mathcal{O} = \sum_{k=1}^{n} c_k O_k = & c_1(W_1 \otimes \mathbb{I} \otimes ... \otimes \mathbb{I}) + \\ & + c_2(\mathbb{I} \otimes W_2 \otimes ... \otimes \mathbb{I}) + \\ & ... \\ & + c_n(\mathbb{I} \otimes ... \otimes \mathbb{I} \otimes W_n),\end{aligned} \tag{4.2}$$

where $W_k \in \{X, Y, Z\}$, $\forall\, k = 1...n$ and $c_k = 2^{-n} tr(O_k^\dagger \mathcal{O})$.

4.2 A measure for barren plateaus

We now turn our attention to finding a way to estimate the flatness of the landscape of the model function. Let $f_V(\Theta)$ denote the cost function induced by the realization V of the entangling gate:

$$f_V(\Theta) = \langle 0^n | U(\Theta)^\dagger \mathcal{O} U(\Theta) | 0^n \rangle. \tag{4.3}$$

A natural measure for the flatness of the landscape of the cost function, with respect to both the random choice of the entangling gates and and uniformly

over the parameter space, is the following:

$$\mathbb{E}_V \mathbb{E}_\Theta \|\nabla f_V(\Theta)\|^2. \tag{4.4}$$

The next lemma, stated and proved in [15], shows that $\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2$ is a good measure as well, by directly relating it to (4.4). To prove this result we assume that the final gates applied to each qubit j consist of a sequence of parametrized Pauli rotations of the form $e^{i\alpha_j X/2} e^{i\beta_j Z/2}$. This request does not impose a constraint on the circuit's architecture as, by the definition of the Haar measure, the sequence does not change the probability distribution of the entangling gates, which allow us to incorporate the sequence of rotations in the last entangling gate acting on the same site.

Lemma 4.1. *Let $f_V(\Theta)$ be as previously defined, then*

$$\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2 \leq \mathbb{E}_V \mathbb{E}_\Theta \|\nabla f_V(\Theta)\|^2 \leq 4 \left(\sum_i \|A_j\|^2 \right) \|\mathcal{O}\| \sqrt{\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2}. \tag{4.5}$$

The result allows us to focus on $\mathbb{E}_V \mathbb{E}_\Theta f(\Theta)^2$ which is easier to compute with our approach.

We now replace the Haar measure on $\mathcal{U}(4)$ with the uniform measure over the Clifford group $\mathcal{C}_2$. Since the Clifford group forms a 2-design, the second moments of operators sampled from $\mathcal{C}_2$ are equivalent to those obtained with operators drawn uniformly at random from the Haar measure on $\mathcal{U}(4)$. We notice that is not necessary to evaluate the expectation over Θ, as the entangling gates form unitary 2-designs, and the 2-design property is preserved under application of a fixed unitary. Therefore the parameterized gates can be ignored, simplifying the expression. Let $V = V_p, \ldots, V_1$ denote the p entangling gates, as in section 3.2 we invert the usual convention and thus V_t is the t-th gate from the end of the circuit. Hence, we have the following result:

$$\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2 = \mathbb{E}_V \mathbb{E}_\Theta \langle 0^n | U^\dagger(\Theta) \mathcal{O} U(\Theta) | 0^n \rangle^2 \tag{4.6}$$

$$= \mathbb{E}_V \langle 0^n | V^\dagger \mathcal{O} V | 0^n \rangle^2. \tag{4.7}$$

If we take into account the definition 4.2 for the observable $\mathcal{O}$, (4.7) becomes

$$\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2 = \mathbb{E}_V \sum_{k,k'} c_k \overline{c_{k'}} \langle 0^n | V^\dagger O_k V | 0^n \rangle \langle 0^n | V^\dagger O_{k'}^\dagger V | 0^n \rangle \tag{4.8}$$

$$= \mathbb{E}_V \sum_{k=1}^{n} c_k^2 \langle 0^n | V_p^\dagger \dots V_1^\dagger O_k V_1 \dots V_p | 0^n \rangle^2 \tag{4.9}$$

$$= \sum_{k=1}^{n} \mathbb{E}_V \left[c_k^2 \langle 0^n | V_p^\dagger \dots V_1^\dagger O_k V_1 \dots V_p | 0^n \rangle^2 \right]. \tag{4.10}$$

The cross terms in (4.8) vanish because we can insert a layer of single-qubit gates, sampled from the uniform measure over $\mathcal{C}_1$, acting on each qubit j after all the two-qubit entangling unitaries. This does not change the global unitary $\tilde{U}$, as the effect of this final layer can be absorbed into the preceding two-qubit entangling gates. Moreover, (4.9) follows from the fact that each O_k is self-adjoint and that c_k is always real being the trace of the product of two self-adjoint operators.

Fix $k \in \{1, 2, \dots, n\}$, then the operator

$$V_p^\dagger \dots V_1^\dagger O_k V_1 \dots V_p, \tag{4.11}$$

is a composition of p conjugations of O_k, operated by p 2-qubit Clifford operators. Following the results of section 3.2 we have that:

$$V_p^\dagger \dots V_1^\dagger O_k V_1 \dots V_p = O_k^{(p)}, \tag{4.12}$$

where $O_k^{(p)} \in \mathcal{P}_n^*$ and thus it is a tensor product of n self-adjoint operators of the single-qubit Pauli group, with the constraint that it cannot be $\pm \mathbb{I}^{\otimes n}$. Therefore (4.10) takes the following form

$$\mathbb{E}_V \mathbb{E}_\Theta f_V(\Theta)^2 = \sum_{k=1}^{n} \mathbb{E}_V \left[c_k^2 \langle 0^n | O_k^{(p)} | 0^n \rangle^2 \right]. \tag{4.13}$$

We know that if even just one element of $O_k^{(p)}$ is either X or Y, then

$$\langle 0^n | O_k^{(p)} | 0^n \rangle = 0. \tag{4.14}$$

On the other hand, if all the elements of $O_k^{(p)}$ are either $\mathbb{I}$ or Z, then

$$\langle 0^n | O_k^{(p)} | 0^n \rangle = \pm 1 \tag{4.15}$$

By Corollary 3.1, the probability that each non identity element of $O_k^{(p)}$ is either X, Y or Z is $\frac{1}{3}$. Thus, we evaluate the variance of the objective function as the

expected value of the probability that $O_k^{(p)}$ is an n-fold tensor product of only Z and $\mathbb{I}$ single-qubit operators:

$$\sum_{k=1}^{n} \mathbb{E}_V \left[c_k^2 \langle 0^n | O_k^{(p)} | 0^n \rangle^2\right] = \sum_{k=1}^{n} c_k^2 \, \mathbb{E}_V \left[\left(\frac{1}{3}\right)^{|\mathrm{supp}(O_k^{(p)})|}\right]. \tag{4.16}$$

4.3 Randomly placed gates model

To obtain a first result on the flatness of the landscape of $f_V(\Theta)$ we assume that V is not fixed at the start.

Assumption 4.3. *The two qubits acted upon by each entangling gate V_t are sampled each time from the uniform distribution over all the $\binom{n}{2}$ couples of qubits, for every $t = 1...p$. Moreover, each layer t contains a single entangling gate V_t, for $t = 1 \dots l$. We denote a particular realization of the entangling gates in this model as $V^{(R)}$.*

Notice that assumption 4.3 implies that $l = p$. This particular choice makes calculating (4.16) much easier since the number of sites on which $O_k^{(p)}$ acts non trivially, namely $|\mathrm{supp}(O_k^{(p)})|$, can be studied with a particularly simple Markov process.

Since the following results hold for every single O_k, we introduce a sequence of random variables in order to describe compactly $|\mathrm{supp}(O_k^{(t)})|$ for all $k = 1, \dots, n$ and $t = 1, \dots, p$.

Definition 4.3. *Let $X_0, X_1, X_2 \dots$ be the sequence of random variables such that*

$$X_t = |\mathrm{supp}(O_k^{(t)})|, \quad \text{for } t = 1...p, \tag{4.17}$$

and

$$X_0 = |\mathrm{supp}(O_k)|, \tag{4.18}$$

for all $k = 1 \dots n$.

We can state the following proposition.

Proposition 4.1. *The sequence of random variables $X_0, X_1, X_2 \dots$, defined on the state space $\mathcal{X} = \{1, 2, 3, ..., n\}$, is a Markov chain whose transition matrix P is defined by the following coefficients:*

$$P_{ij} = \begin{cases} \frac{2}{5}\frac{i(i-1)}{n(n-1)} & \text{if } j = i-1, \\ \frac{3}{5}\frac{2i(n-i)}{n(n-1)} & \text{if } j = i+1, \\ 1 - \frac{2}{5}\frac{i(i-1)}{n(n-1)} - \frac{3}{5}\frac{2i(n-i)}{n(n-1)} & \text{if } j = i. \end{cases} \tag{4.19}$$

Proof. The state space $\mathcal{X}$ represents the values the random variables can assume through the process, indeed it must be at least 1 by definition 4.2 and it cannot be greater than n. Each transition probability from a state i to a state j, where $i, j = 1 \dots n$, is represented by the corresponding coefficient of the transition matrix $P_{i,j}$. As in subsection 3.1.1 we denote by W any non identity element of $\mathcal{P}$.

Consider the action of the t-th entangling gate V_t for $t = 1 \dots p$, and let $\text{supp}(V_t) = \{l, k\}$ for $l, k = 1, \dots, n,\ l \neq k$.

If $j = i - 1$, the transition probability from $X_{t-1} = i$ to $X_t = j$ is given by the product of two values:

$$P_{i,i-1} = \mathbb{P}\left(V_t^\dagger O_{l,k}^{(t-1)} V_t \in [\pm\mathbb{I} \otimes W] \cup [\pm W \otimes \mathbb{I}]\right) \times \mathbb{P}(\text{supp}(V_t) \subseteq \text{supp}(O^{(t-1)})). \tag{4.20}$$

By Proposition 3.1 and by combinatorial results we obtain

$$P_{i,i-1} = \frac{2}{5}\frac{i(i-1)}{n(n-1)}. \tag{4.21}$$

Similarly, if $j = i + 1$ the transition probability from $X_{t-1} = i$ to $X_t = j$ is given by

$$\begin{aligned} P_{i,i-1} =& \mathbb{P}\left(V_t^\dagger O_{l,k}^{(t-1)} V_t \in [\pm W \otimes W']\right) \\ & \times \mathbb{P}\left(\left(\text{supp}(V_t) \not\subset \text{supp}(O^{(t-1)})\right) \cap \left(\text{supp}(V_t) \cap \text{supp}(O^{(t-1)}) \neq \emptyset\right)\right) \end{aligned} \tag{4.22}$$

By Proposition 3.1 and by combinatorial results we obtain

$$P_{i,i+1} = \frac{3}{5}\frac{2i(n-i)}{n(n-1)}. \tag{4.23}$$

If $j = i$, then the transition probability from $X_{t-1} = i$ to $X_t = j$ is

$$P_{i,i} = \frac{(n-i)(n-i-1)}{n(n-1)} + \frac{3}{5}\frac{i(i-1)}{n(n-1)} + \frac{2}{5}\frac{2i(n-i)}{n(n-1)}. \tag{4.24}$$

The first term is the probability that $\text{supp}(V_t) \cap \text{supp}(O^{(t-1)}) = \emptyset$. The second is the product of the two probabilities

$$\mathbb{P}\left(V_t^\dagger O_{l,k}^{(t-1)} V_t \in [\pm\mathbb{I} \otimes W] \cup [\pm W \otimes \mathbb{I}]\right), \tag{4.25}$$

and

$$\mathbb{P}\left(\left(\text{supp}(V_t) \not\subset \text{supp}(O^{(t-1)})\right) \cap \left(\text{supp}(V_t) \cap \text{supp}(O^{(t-1)}) \neq \emptyset\right)\right). \tag{4.26}$$

The last term is the result of the product

$$\mathbb{P}\left(V_t^\dagger O_{l,k}^{(t-1)} V_t \in [\pm W \otimes W']\right) \times \mathbb{P}(\text{supp}(V_t) \subseteq \text{supp}(O^{(t-1)})). \tag{4.27}$$

Case three of (4.19) can be shown to be equivalent to (4.24) but it provides a more compact way of expressing the result.

All the entries outside the three principal diagonals are zero. This is due to the fact that each Clifford gate acts on two qubits only and therefore, the number of non-identity operators cannot increase/decrease by more than one unit at a time. □

The definition of transition matrix can be found in Appendix A. In the following pages we analyze the results obtained through the study of this Markov process.

4.3.1 The stationary distribution

The process can be easily proved to be irreducible and aperiodic (both properties are defined in Appendix A). Indeed, the tridiagonal structure of the matrix clearly indicates that all states are reachable from any given state and that the period of each state is 1. Therefore, there exists a unique stationary distribution to which the system converges (see Appendix A). We call such probability distribution π and we find that the following expression holds:

$$\pi_i = \frac{1}{4^n - 1} \frac{3^i n!}{(n-i)! i!}, \quad i = 1...n. \tag{4.28}$$

The stationary distribution shows how the system behaves in the long time, which, in our case, means after a large number of gates.

Observation 4.1. *The expected value of X_t when the system is in the stationary state is*

$$\mathbb{E}[X_t] = \left(\frac{3}{4}n\right) \frac{4^n}{4^n - 1}. \tag{4.29}$$

Proof.

$$\begin{aligned} \mathbb{E}[X_t] = \sum_{i=1}^{n} i\pi_i &= \frac{1}{4^n - 1} \sum_{i=1}^{n} \frac{3^i n!}{(n-i)!(i-1)!} \\ &= \frac{3n}{4^n - 1} \sum_{\mu=0}^{n-1} \frac{3^\mu (n-1)!}{(n-1-\mu)!\mu!} \\ &= \frac{3n}{4^n - 1} 4^{n-1}, \end{aligned} \tag{4.30}$$

where we used the change of variable $\mu = i - 1$. □

Observation 4.1 means that if the circuit reaches the stationary state, the expected fraction of non identity elements in O_k for $k = 1 \dots n$ is slightly more that $\frac{3}{4}$. This is due to the fact that each single qubit gate in O_k is sampled uniformly at random from the Pauli group with the condition that, for all $t = 1 \dots p$, O_k cannot be the identity element:

$$O_k^{(t)} \neq \bigotimes^{n} \mathbb{I}. \tag{4.31}$$

In the following sections, the stationary distribution π is necessary to study the expectation (4.16) for this model. Therefore, it is useful to determine how long it takes for our system to reach the stationary state, in other words, the relaxation time of the Markov chain (see Appendix A). We now prove a result that is fundamental for studying the value t_{rel}.

Proposition 4.2. *The probability distribution* (4.28) *satisfies the detailed balance equation:*

$$\pi_i P_{i,j} = \pi_j P_{j,i}. \tag{4.32}$$

Proof. The detailed balance equation is satisfied if we can prove that P is self-adjoint with respect to the scalar product defined by the matrix

$$D = \mathrm{diag}(\pi), \tag{4.33}$$

that is

$$\langle Pv, w \rangle_\pi = \langle v, Pw \rangle_\pi, \tag{4.34}$$

for any n-dimensional vectors v, w. Indeed, Equation (4.34) takes the following form

$$P^T D = DP. \tag{4.35}$$

$$P_{j,i}\pi_j = \pi_i P_{i,j} \tag{4.36}$$

The equality is immediately verified. Indeed, $P_{i,j}$ and $P_{j,i}$ are different from zero if and only if

$$j = \begin{cases} i, \\ i-1, \\ i+1. \end{cases} \tag{4.37}$$

For $j = i$ the proof is trivial. Let $j = i + 1$:

$$\pi_{i+1} = \frac{1}{4^n - 1} \frac{3^{i+1} n!}{(n-i-1)!(i+1)!}. \tag{4.38}$$

The corresponding elements of the transition matrix are

$$P_{i,i+1} = \frac{3}{5}\frac{2i(n-i)}{n(n-1)}; \tag{4.39}$$

$$P_{i+1,i} = \frac{2}{5}\frac{i(i+1)}{n(n-1)}. \tag{4.40}$$

Therefore, the detailed balance equation is satisfied if

$$\frac{1}{4^n-1}\frac{3^{i+1}n!}{(n-i-1)!(i+1)!}\frac{2}{5}\frac{i(i+1)}{n(n-1)} = \frac{1}{4^n-1}\frac{3^i n!}{(n-i)!i!}\frac{3}{5}\frac{2i(n-i)}{n(n-1)}. \tag{4.41}$$

The identity can be verified through straightforward simplifications. The same approach is used for $j = i-1$. □

4.3.2 Relaxation time of the Markov chain

For the properties of Markov chains we detail in Appendix A, Proposition 4.2 ensures that all the eigenvalues of the transition matrix are real. Therefore, it is possible to arrange the eigenvalues of P in decreasing order:

$$1 = \lambda_n^{(1)} \geq \lambda_n^{(2)} \geq ...\lambda_n^{(n)}, \tag{4.42}$$

where $\lambda_n^{(1)}$ is the eigenvalue corresponding to the stationary distribution. We study how the relaxation time t_{rel}, defined in Appendix A, varies with the number of qubits.

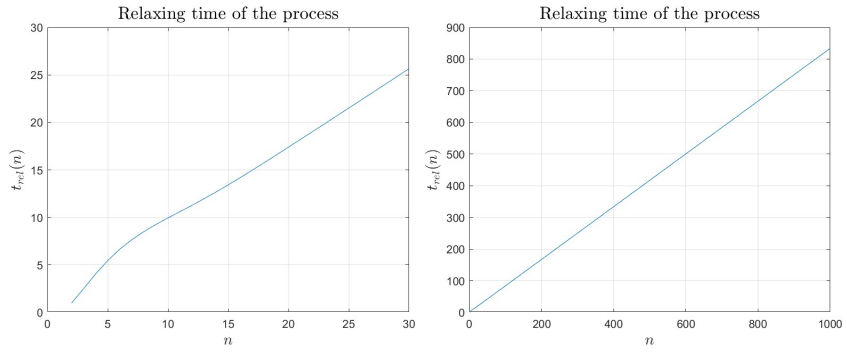


Figure 4.4: **Plot of the relaxation time as a function of n:** the two plots show how the relaxation time varies with the number of qubits in the circuit. The image on the left shows the non linear behavior of the relaxing time for small values of n. On the right we can see that t_{rel} has an asymptotic linear dependence on n. The algorithm that has been used to obtain the results is shown in Appendix B

The numerical results show that t_{rel} grows, almost linearly, with n. This means that, as the number of qubits in the circuit increases, more entangling gates are required for X_t to reach the stationary distribution. Figure 4.4 shows the result obtained.

4.3.3 Expected magnitude of the gradient

The properties of Markov chains provide us with a useful expression to compute the expected value of a function depending on the random variable. Let f be the function defined over $\mathcal{X}$ such that $f(i) = \left(\frac{1}{3}\right)^i$, each term of (4.16) is obtained as follows:

$$\mathbb{E}_\mu \left[\left(\frac{1}{3}\right)^{X_p} \right] = \mu P^p \mathbf{f}, \tag{4.43}$$

where $\mathbf{f}$ is the n dimensional vector of elements $\mathbf{f}(i) = f(i)$ and μ the probability distribution of the starting state. We firstly consider the case where the system is already in the stationary state.

Proposition 4.3. *Let $\mu = \pi$, then*

$$\mathbb{E}_\pi \left[\left(\frac{1}{3}\right)^{X_p} \right] = \frac{1}{2^n + 1}. \tag{4.44}$$

Proof.

$$\begin{aligned} \mathbb{E}_\pi \left[\left(\frac{1}{3}\right)^{X_p} \right] &= \sum_{i=1}^{n} \pi_i \frac{1}{3^i} \\ &= \frac{1}{4^n - 1} \sum_{i=1}^{n} \binom{n}{i} \\ &= \frac{1}{4^n - 1} (2^n - 1) \\ &= \frac{1}{2^n + 1} \end{aligned} \tag{4.45}$$

□

Therefore, if the starting distribution matches π, the expected magnitude of the gradient of $f_{V^{(R)}}(\Theta)$ decays exponentially with the number of qubits. However, the observable described in definition 4.2 is such that the initial distribution is represented by the row vector that is 1 in the first component and zero everywhere else.

Proposition 4.4. *Consider the n-dimensional probability vector* $\mathbf{e}_1 = (1, 0, \ldots, 0)$*, then:*

$$\mathbb{E}_{\mathbf{e}_1}\left[\left(\frac{1}{3}\right)^{X_p}\right] \leq \frac{1}{2^n+1} + \frac{1}{3}\frac{\left(\lambda_n^{(2)}\right)^p}{\lambda_n^{(n)}}. \tag{4.46}$$

Proof. We decompose the vector $\mathbf{e}_1$ into the sum of two components: one proportional to the stationary distribution π, and the other orthogonal to π with respect to the inner product associated with π:

$$\mathbf{e}_1 = \gamma\pi + v = \gamma\pi + (\mathbf{e}_1 - \gamma\pi). \tag{4.47}$$

The constant γ is chosen in such a way that $\langle \pi, v\rangle_\pi = 0$:

$$\gamma = \frac{\langle \pi, \mathbf{e}_1\rangle_\pi}{\langle \pi, \pi\rangle_\pi}. \tag{4.48}$$

Notice that γ is always less or equal than 1. Indeed, if we compute the scalar products we obtain

$$\frac{\langle \pi, \mathbf{e}_1\rangle_\pi}{\langle \pi, \pi\rangle_\pi} = \frac{\pi_1^2}{\sum_{i=1}^n \pi_i^3}. \tag{4.49}$$

However, $\sum_{i=1}^n \pi_i^3 = \pi_n^3 + \sum_{i=1}^{n-1} \pi_i^3$ and we can prove that $\pi_n^3 > \pi_1^2$, in fact by (4.28) we have that

$$\frac{(3n)^2}{(4^n-1)^2} < \frac{(3^n)^3}{(4^n-1)^3}. \tag{4.50}$$

Therefore, $\sum_{i=1}^n \pi_i^3 > \pi_1^2$ and $\gamma < 1$.

Let us now return to the main objective of this proof.

$$\mathbf{e}_1 P^p \mathbf{f} = \gamma\pi P^p \mathbf{f} + v P^p \mathbf{f} \tag{4.51}$$

$$= \gamma \frac{1}{2^n+1} + v P^p \mathbf{f}. \tag{4.52}$$

The last equality is a consequence of the result of Prop. 4.3.

Let us focus now on the second term of (4.52), in the following steps we use the scalar product defined by the stationary distribution.

$$\begin{aligned} v P^p \mathbf{f} &= (\mathbf{e}_1 - \gamma\pi) P^p \mathbf{f} && (4.53)\\ &= (\mathbf{e}_1 - \gamma\pi) P^p (D^{-1}D)\mathbf{f} \\ &= \langle (\mathbf{e}_1 - \gamma\pi) P^p D^{-1}, \mathbf{f}\rangle_\pi \\ &\leq \|(\mathbf{e}_1 - \gamma\pi) P^p D^{-1}\|_\pi \|\mathbf{f}\|_\pi \\ &\leq \|(\mathbf{e}_1 - \gamma\pi) P^p\|_\pi \|D^{-1}\|_{\pi\to\pi} \|\mathbf{f}\|_\pi && (4.54) \end{aligned}$$

Let $v_1, v_2, \ldots, v_n$ be the orthonormal basis of P with respect to the scalar product $\langle \cdot, \cdot \rangle_\pi$, in particular $v_1 = \frac{\pi}{\|\pi\|_\pi}$. Then we can write v as follows:

$$v = \sum_{i=2}^{n} \alpha_i v_i, \tag{4.55}$$

Then, the first term of (4.54), takes the following form:

$$\begin{aligned} \|(\mathbf{e}_1 - \gamma\pi)P^p\|_\pi &= \| \left(\sum_{i=2}^{n} \alpha_i v_i \right) P^p \|_\pi \\ &= \| \sum_{i=2}^{n} \alpha_i v_i \left(\lambda_n^{(i)} \right)^p \|_\pi \\ &= \sqrt{\langle \sum_{i=2}^{n} \alpha_i v_i \left(\lambda_n^{(i)} \right)^p, \sum_{j=2}^{n} \alpha_j v_j \left(\lambda_n^{(j)} \right)^p \rangle_\pi} \\ &= \sqrt{\sum_{i=2}^{n} \alpha_i^2 \left(\lambda_n^{(i)} \right)^{2p}} \\ &\leq \sqrt{\left(\lambda_n^{(2)} \right)^{2p} \sum_{i=2}^{n} \alpha_i^2} \\ &= \left(\lambda_n^{(2)} \right)^p \|v\|_\pi \\ &\leq \left(\lambda_n^{(2)} \right)^p \|\mathbf{e}_1\|_\pi \\ &\leq \left(\lambda_n^{(2)} \right)^p, \end{aligned} \tag{4.56}$$

where $\lambda_n^{(2)}$ is the second largest eigenvalue of P. The last inequality is a consequence of $\|\mathbf{e}_1\|_\pi \leq 1$ for all $n \geq 1$ by (4.28). Let's take the second term of (4.54)

$$\begin{aligned} \|D^{-1}\|_{\pi \to \pi} &= \sup_{x \neq 0} \frac{\|D^{-1}x\|_\pi}{\|x\|_\pi} \\ &= \sqrt{\sup_{x \neq 0} \frac{(D^{-1}x)^T D (D^{-1}x)}{x^T D x}} \\ &= \sqrt{\sup_{x \neq 0} R(D^{-1}, D)}, \end{aligned} \tag{4.57}$$

where $R(D^{-1}, D)$ is the generalized Rayleigh quotient. Therefore, by the prop-

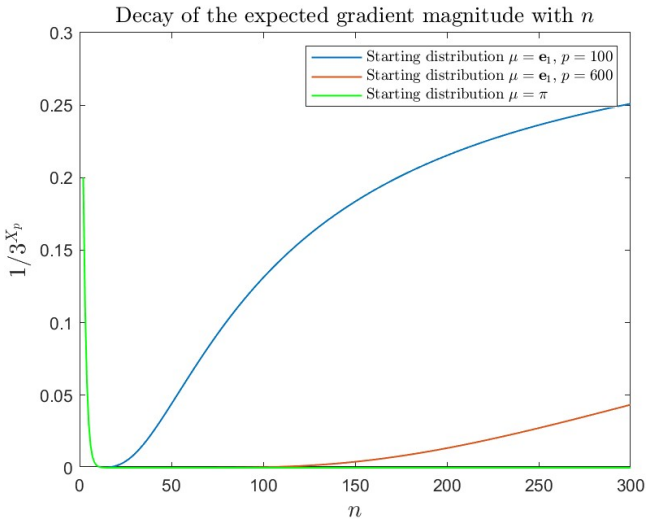


Figure 4.5: **Vanishing gradient for a fixed number of gates**. The image shows the numerical results of the decay rate of $\mathbb{E}_{\mathbf{e}_1}[1/3^{X_p}]$ with respect to the number of qubits, when the starting distribution is the vector $\mathbf{e}_1$ and π respectively. We see that as the number of qubits starts to grow with respect to the number of gates, the variance is not vanishing anymore. The algorithm used to obtain the result is shown in Appendix B.

erties of the Rayleigh quotient we have that

$$\sup_{x\neq 0} R(D^{-1}, D) = \max\{\lambda \mid \lambda \in \sigma(D^{-1}D^{-1})\} = \frac{1}{\left(\lambda_n^{(n)}\right)^2}, \tag{4.58}$$

where $\lambda_n^{(n)}$ is the least eigenvalue of the matrix P. Thus we have

$$\|D^{-1}\|_{\pi\to\pi} = \frac{1}{\lambda_n^{(n)}}. \tag{4.59}$$

To conclude, we consider the last term in (4.54):

$$\begin{aligned} \|\mathbf{f}\|_\pi = \sqrt{\mathbf{f}^T D\mathbf{f}} &= \sqrt{\sum_{i=1}^{n} \frac{1}{3^{2i}}\pi_i} \\ &\leq \sqrt{\frac{1}{3^2}\sum_{i=1}^{n}\pi_i} \\ &= \frac{1}{3} \end{aligned} \tag{4.60}$$

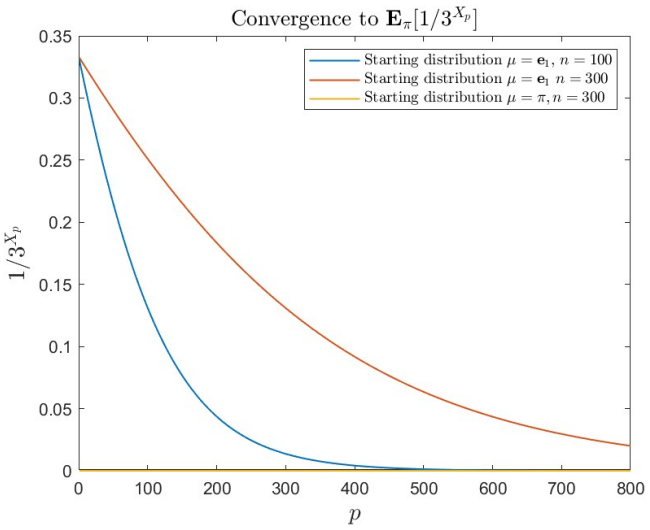


Figure 4.6: **Vanishing gradient for a fixed number of qubits**. The image shows how $\mathbb{E}_{\mathbf{e}_1}[1/3^{X_p}]$ converges to $\mathbb{E}_\pi[1/3^{X_p}]$ as p grows. Accordingly to the results of subsection 4.3.2, we notice that for greater values of n, the convergence is slower. The algorithm used to obtain the result is shown in Appendix B.

It follows that the expected value of $\left(\frac{1}{3}\right)^{X_p}$ satisfies the following inequality:

$$\mathbf{e}_1 P^p \mathbf{f} \leq \frac{1}{2^n + 1} + \frac{1}{3} \frac{\left(\lambda_n^{(2)}\right)^p}{\lambda_n^{(n)}}, \tag{4.61}$$

which proves (4.46). □

The result of Prop. 4.4 tells us that for any realization $V^{(R)}$ of the entangling gates the decay of the magnitude of the gradient depends on two values: the number of qubits n, and the number of gates p. A first observation tells us that, if we fix the number of qubits n, then the second term of Eq. (4.61) vanishes with $p \to \infty$, as one would expect, a consequence of the fact that $\lambda_n^{(2)} < 1$. Hence, for large values of p the result matches the one of Prop. 4.3. If we study the gradient magnitude numerically, accordingly to what obtained in subsection 4.3.2 we notice that as n grows, the time required to reach the stationary distribution grows as well. Thus the decay is slower thanks to the effect of the second term but, eventually, the stationary state is reached at a given point. If instead we fix the number of gates, the same result is shown from a different and more interesting perspective. What we notice, once again trough a numerical analysis, is that as n grows with respect to p, the expectation is not vanishing anymore. Indeed, the second term of Eq. (4.61) vanishes for $p \gg n$

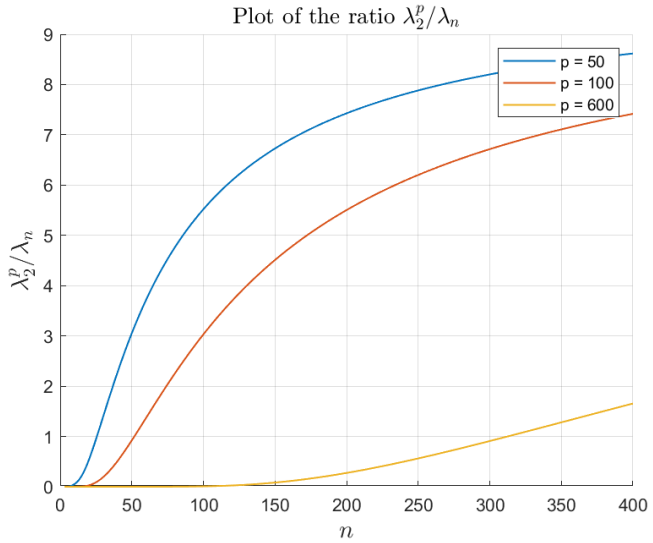


Figure 4.7: **Value of** $\left(\lambda_n^{(2)}\right)^p/\lambda_n^{(n)}$. The plot shows how the ratio $\left(\lambda_n^{(2)}\right)^p/\lambda_n^{(n)}$ grows with n for fixed values of p. We observe that the function increases rapidly until $n \sim p$, after which the curve exhibits sublogarithmic growth. This shows that the second term in (4.46) is what keeps the gradient from vanishing when $n \gg p$.

$$\frac{\left(\lambda_n^{(2)}\right)^p}{\lambda_n^{(n)}} \xrightarrow[p \gg n]{} 0. \tag{4.62}$$

On the other hand, if $p \ll n$ the trend reverses. As figure 4.5 shows, for $p \ll n$, the ratio between $\left(\lambda_n^{(2)}\right)^p$ and $\lambda_n^{(n)}$ starts growing, fast in a first moment, and then sublogarithmically.

4.4 Fixed gates model

We now turn our attention to a more suitable model for simulating real life applications. We require that the architecture of the circuit (i.e. the position of the gates) is determined and fixed at the beginning of the process, and we denote a particular realization of the entangling gates in this model as $V^{(F)}$. We can slightly modify the assumption on the structure of the layers we required in section 4.3, this makes the result more general.

Assumption 4.4. *Each layer may be composed by one or eventually more entangling gates acting on different qubit pairs.*

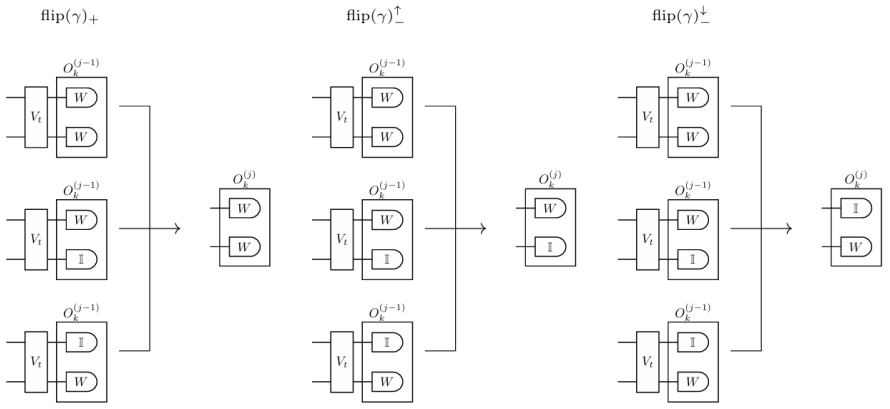


Figure 4.8: **Different types of possible transitions**. The figure illustrates which transitions are counted by each quantity: $\text{flip}(\gamma)_+$, $\text{flip}(\gamma)^{\uparrow}_{-}$, and $\text{flip}(\gamma)^{\downarrow}_{-}$, for a two-qubit observable. The first term counts all conjugations performed by the entangling gates that result in a non-identity–non-identity operator pair. The second counts those that produce an observable of the form $O_k^{(j)} = W \otimes \mathbb{I}$, and the third counts those that produce $O_k^{(j)} = \mathbb{I} \otimes W$. The same classification holds for general n-qubit observables, since the entangling gates always act on pairs of qubits.

Even though $|\text{supp}(O_k^{(p)})|$ is still a Markov process, a straightforward study of the problem through a Markov chain, is not possible anymore because of the dependency of the chain on the position of the gates. Therefore a different approach is required. Equation (4.16) can be written as follows

$$\mathbb{E}_{V^{(F)}}\left[\left(\frac{1}{3}\right)^{|\text{supp}(O_k^{(p)})|}\right] = \sum_{j=1}^{n}\left(\frac{1}{3}\right)^{j}\mathbb{P}(|\text{supp}(O_k^{(p)})| = j), \quad \forall\, k = 1\ldots n. \tag{4.63}$$

The process is a random walk over the strings of n symbols from the alphabet $\{\mathbb{I}, W\}$. Borrowing the notation from [15], we express the probability $\mathbb{P}(|\text{supp}(O_k^{(p)})| = j)$ as a sum over trajectories $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_p)$. Each γ_t is the assignment of value $\mathbb{I}$ or W to each single-qubit operator of the n-fold tensor product $O_k^{(t)}$; γ_0 is the starting configuration. Similarly to what we defined for operators, we indicate as $|\text{supp}(\gamma_t)|$ the number of non-identity operators of $O_k^{(t)}$. Clearly, for a trajectory to be considered valid, γ must satisfy the rules detailed in subsection 3.1.1. Given these, we define the probability of a trajectory as follows

$$\mathbb{P}(\gamma) = \left(\frac{3}{5}\right)^{\text{flip}(\gamma)_+}\left(\frac{1}{5}\right)^{\text{flip}(\gamma)^{\downarrow}_{-}}\left(\frac{1}{5}\right)^{\text{flip}(\gamma)^{\uparrow}_{-}}. \tag{4.64}$$

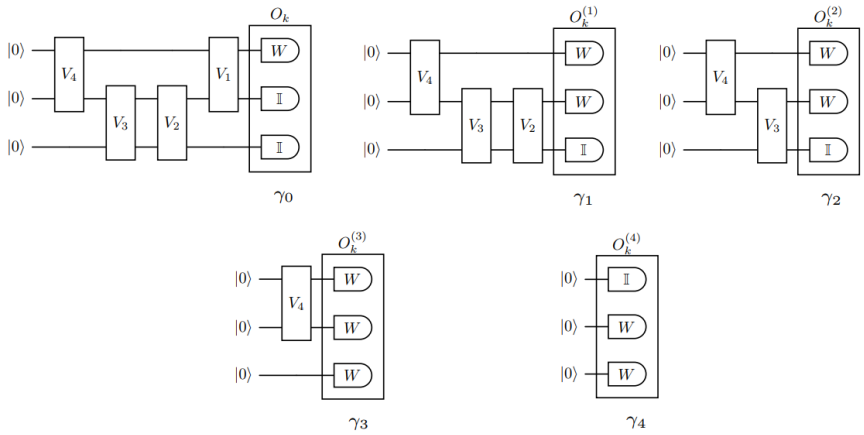


Figure 4.9: **Representation of a trajectory** γ. As an example we consider a 3-qubit circuit. The probability of the trajectory is defined by the probability of the results of the conjugations operated by the entangling gates. For this particular example we have: $\mathbb{P}(\gamma) = \frac{3}{5} \times \frac{1}{5} \times \frac{3}{5} \times \frac{1}{5} = \frac{9}{625}$.

The value $\mathrm{flip}(\gamma)_+$ is the number of times an entangling gate, conjugates the operators it acts upon into a non-identity–non-identity pair. On the other hand, the values $\mathrm{flip}(\gamma)_-^{\uparrow}$ and $\mathrm{flip}(\gamma)_-^{\downarrow}$ represent the opposite case, that is when the result of the action of an entangling gate is an identity–non-identity pair of operators. The former counts the gates whose output is $W \otimes \mathbb{I}$ while the latter counts the cases when the output is the opposite, namely $\mathbb{I} \otimes W$ (see Fig. 4.8). Clearly $\mathrm{flip}(\gamma)_+ + \mathrm{flip}(\gamma)_-^{\uparrow} + \mathrm{flip}(\gamma)_-^{\downarrow} \leq p$. Thus, the probability in Eq. (4.63) takes the following form

$$\mathbb{P}(|\mathrm{supp}(O_k^{(p)})| = j) = \sum_{\gamma} \mathbb{P}(\gamma) \mathbf{1}_{|\mathrm{supp}(\gamma_p)|=j}. \tag{4.65}$$

4.4.1 Lower bound

We find that for an unstructured ansatz as the one described in Assumption 4.4 the following lower bound on (4.16) holds.

Proposition 4.5. *(Lower bound) Let L be the smallest number of 2-qubits entangling gates that any possible trajectory γ must pass trough. Then we find*

$$\mathbb{E}_{V^{(F)}} \left[\left(\frac{1}{3} \right)^{|\mathrm{supp}(O_k^{(p)})|} \right] \geq \frac{1}{3} \left(\frac{1}{5} \right)^{L}. \tag{4.66}$$

Proof.

$$\begin{aligned}
\mathbb{E}_{V^{(F)}}\left[\left(\frac{1}{3}\right)^{|\text{supp}(O_k^{(p)})|}\right] &= \sum_{j=1}^{n}\left(\frac{1}{3}\right)^{j}\mathbb{P}(|\text{supp}(O_k^{(p)})| = j) \qquad (4.67)\\
&\geq \frac{1}{3}\mathbb{P}(|\text{supp}(O_k^{(p)})| = 1)\\
&= \frac{1}{3}\sum_{\gamma}\mathbb{P}(\gamma)\mathbf{1}_{|\text{supp}(\gamma_p)|=1}\\
&\geq \frac{1}{3}\left(\frac{1}{5}\right)^{L}.
\end{aligned}$$

Where $\left(\frac{1}{5}\right)^{L}$ is the probability of the trajectory where $|\text{supp}(\gamma_t)| = 1 \quad \forall t = 1 \dots p$. □

Chapter 5

Conclusions

The results of this thesis can be divided into two parts. The first is a purely mathematical study of the effect of the Clifford group on the Pauli group (see chapter 3). The second part concerns the application of the previously developed theory to the study of barren plateaus in variational quantum circuits (see chapter 4).

We first established that the action of a random element from the n-qubit Clifford group on an element of the restricted set $\mathcal{P}_n^*$ (see Definition 3.1), results in a uniformly random element of the same set. This result is formally stated in Theorem 3.2. The proof relies on demonstrating that the action of the Clifford group $\mathcal{C}_n$ on $\mathcal{P}_n^*$ is transitive, as shown in Theorem 3.1. This transitivity implies that any element of $\mathcal{P}_n^*$ can be mapped to any other element via some Clifford transformation, ensuring uniformity under the random action. While this property was already widely believed to hold [54], a rigorous and complete proof had not previously been documented. Our result therefore fills this gap.

In chapter 4, we applied the results obtained for the Clifford group to investigate the barren plateau phenomenon in unstructured parametrized quantum circuits that satisfy the mild conditions specified in Assumptions 4.1 and 4.2. What we found is that, for an n-qubit ansatz defined accordingly these assumptions, the study of the barren plateau phenomenon reduces to analyzing a random walk over strings of n symbols drawn from a two letters alphabet.

For the same circuit, we considered two possible models. In the first model, we assumed that the pair of qubits acted upon by each entangling gate is sampled independently at each layer from the uniform distribution over all $\binom{n}{2}$ possible qubit pairs. This framework is particularly convenient, as it allows us to study the random walk as a remarkably simple Markov chain (see Proposition 4.1). For this Markov chain, we showed, in Proposition 4.4, an upper bound on the

magnitude of the gradient that depends on both the number of qubits in the circuit n, and the number of entangling gates p. The proposition, implies that for a fixed number of qubits, the expected magnitude of the gradient of the loss decays exponentially with the number of entangling gates. This result is in agreement with the fact that as the number of gates increases, the system converges to the stationary distribution which, as shown in Proposition 4.3, is associated with an exponentially vanishing gradient in the number of qubits. Moreover, the numerical results show that the decay of the expected magnitude of the gradient, becomes slower when n increases relative to p (see Figure 4.6). This aligns with the numerical results obtained for the relaxation time of the process, shown in Figure 4.5, that illustrate that t_{rel} grows almost linearly with n.

The same result can be interpreted from a complementary perspective by fixing the number of entangling gates and allowing the number of qubits to grow. In this regime we observe, thanks to the numerical results, that the gradient no longer vanishes when $n > p$ (Figure 4.5), providing further evidence of the absence of barren plateaus in sufficiently wide but shallow circuits. This is due to the ratio, that appears in Equation (4.46), of the second and the last eigenvalue of the stochastic matrix P associated with the Markov process: $\left(\lambda_n^{(2)}\right)^p / \lambda_n^{(n)}$. As shown in Figure 4.7, the factor grows with n for a fixed number of gates, a result that could not be justified analytically as it was not possible to find an expression for the eigenvalue of the stochastic matrix P. Thus, Proposition 4.4 confirms, through a more direct approach, what has already been observed in previous works, namely that for shallow circuits with local observables, the gradient does not vanish exponentially.

Subsequently, we considered a more practically relevant scenario in which the architecture of the circuit is fixed. In this setting, we derive a lower bound on the typical magnitude of the gradient, this is formalized in Proposition 4.5. Our analysis confirms the lower bound previously obtained by [15] for the case of 1-local observables. The difference we identify lies in the definition of the random walk: while their approach relies on a random initialization of qubit labels, our random walk framework does not require this assumption.

One limitation of our work is the absence of an upper bound for the expected magnitude of the gradient in the fixed architecture model. Establishing such bound seems challenging, the difficulty arises from the vast number of possible trajectories of the random walk and the absence of easily tractable fixed points within its dynamics. This combinatorial complexity makes it challenging to apply standard techniques for bounding the gradient's magnitude from above. Moreover, although we focused on 1-local observables, our analysis can be extended to the broader class of k-local Pauli observables. Future work may build upon these results to establish such bound and further characterize the behavior of gradients in unstructured parametrized ansätze for more general Pauli observables.

Ringraziamenti

Vorrei ringraziare sinceramente il mio relatore, Prof. Giacomo de Palma, e correlatore, Prof. Davide Pastorello, per la pazienza e l'enorme disponibilità dimostratami durante questi mesi, per i consigli e per i preziosi insegnamenti che hanno alimentato in me la passione per la materia.

Vorrei poi ringraziare tutta la mia famiglia e in particolare i miei genitori per gli sforzi fatti per sostenermi durante tutti questi anni e per avermi trasmesso la passione per lo studio. Ringrazio anche mio fratello, Pietro, per tutte le risate e anche per il supporto tecnico e sportivo in questi mesi.

Grazie Maicol, per il supporto e per l'amore incondizionato datomi dal primo giorno che ci siamo conosciuti, per i viaggi e per le esperienze che hanno reso meravigliosi questi anni; e poi perché se non mi avessi parlato anni fa dei computer quantistici, chissà se avrei mai intrapreso questa strada. Questa laura è anche merito tuo.

Ringrazio le mie amiche Eleonora B., Eleonora C., Erica, Fabiola, siete per me un'ispirazione e posso solo essere grata per avervi nella mia vita e poter sempre contare su di voi. Ringrazio anche Sara, per aver reso quei cinque mesi a Bergen indimenticabili.

Voglio infine ringraziare tutti i miei amici da quelli delle superiori a quelli dell'univeristà, per i ricordi bellissimi di questi anni, per le serate e i pomeriggi passati insieme.

Appendix A

Markov chains

In this chapter, we provide a brief introduction to the fundamental concepts of Markov chains necessary for our results. All propositions and theorems presented here are proven in Ref. [55]. A Markov chain (MC) is a process which moves along the elements of a set $\mathcal{X}$ in the following way: when the system is in the state $i \in \mathcal{X}$, the next position is chosen according to a fixed probability distribution $P_{i,\cdot}$ that only depends on the current state of the system. More formally we have the following definition:

Definition A.1. *(Markov chain) Let $\mathcal{X}$ be a discrete set, a Markov chain is a sequence of random variables $(X_t)_{t\geq 0} = X_0, X_1, \ldots$ taking values in $\mathcal{X}$ with the property that*

$$\mathbb{P}(X_{t+1} = j | X_0 = x_0, \ldots, X_{t-1} = x_{t-1}, X_t = i) = \mathbb{P}(X_{t+1} = j | X_t = i), \quad \text{(A.1)}$$

for all $x_0, \ldots, x_{t-1}, i, j \in \mathcal{X}$, and $t \geq 0$. The space $\mathcal{X}$ is the state space of the Markov chain.

We always assume that $\mathcal{X}$ is finite, as this is the relevant case for our purposes. Equation (A.1) is called the Markov property and it states that the probability of transitioning from a state i at time t to a state j at time $t+1$, does not depend on the sequence of states that precedes i. The probability distribution of the process at time t is arranged in a distribution vector μ_t. Moreover, since the probabilities only depends on i and j they can be arranged in a stochastic matrix P that takes the name of transition matrix.

Definition A.2. *(Stochastic matrix) A stochastic matrix is a square matrix P, whose i-th row is the distribution $P_{i,\cdot}$. Thus, it satisfies*

i. $P_{i,j} \geq 0$ *for all* i, j.

ii. For each row i $\sum_j P_{i,j} = 1$.

Multiplying by P on the right updates the distribution by a step:

$$\mu_t = \mu_{t-1} P \quad \forall\, t \geq 1. \tag{A.2}$$

Thus, if we are given an initial distribution μ_0, we can determine any distribution at time t by the following equation

$$\mu_t = \mu_0 P^t \quad \forall\, t \geq 1. \tag{A.3}$$

In particular, for the element-wise formulation of (A.3), the probability of transitioning from state i to state j at time $t \geq 1$ is given by $(P^t)_{i,j}$. We use the notation $\mathbb{E}_\mu$ to indicate expectations given that $\mu_0 = \mu$.

Definition A.3. *(Irreducibility) A chain* $(X_t)_{t\geq 0}$ *with transition matrix* P *and state space* $\mathcal{X}$ *is called irreducible if, for any two states* $i, j \in \mathcal{X}$ *there exist an integer* t *such that* $(P^t)_{i,j} > 0$.

In other words, this means that it is always possible to reach any state from any other state.

We denote as $\mathcal{T}(i) = \{t \geq 1 | (P^t)_{i,j} > 0\}$ the set of times when it is possible for the chain to return to the starting position i.

Definition A.4. *(Period of a Markov chain) Consider a Markov chain with state space* $\mathcal{X}$. *The period of a state* $i \in \mathcal{X}$ *is the greatest common divisor of* $\mathcal{T}(i)$. *If all states have period* 1 *then the chain is called aperiodic.*

The following lemma implies that for an irreducible chain, if it is possible to show that a state i is such that $\text{gdc}\{\mathcal{T}(i)\} = 1$, then the chain is aperiodic.

Lemma A.1. *If a Markov chain with state space* $\mathcal{X}$ *is irreducible, then* $\text{gdc}\{\mathcal{T}(i)\} = \text{gdc}\{\mathcal{T}(j)\}$ *for all* $i, j \in \mathcal{X}$.

Long-term behavior of a Markov chain

We are interested in understanding how the distribution behaves in the long term. A fundamental element for the study of the long-term behavior of a MC is the stationary distribution, which is a distribution π that satisfies

$$\pi = \pi P. \tag{A.4}$$

Clearly, if the starting state is π then $\mu_t = \pi$ for all $t \geq 0$. To show the existence of such distribution, we firstly have to introduce a few more definitions and properties of MCs.

Definition A.5. *(Hitting time) For $i \in \mathcal{X}$ we define the hitting time for i as*

$$\tau_i = \min\{t \geq 1 | X_t = i\}. \tag{A.5}$$

When $X_0 = i$, τ_i is also called the first return time.

The following lemma allows us to show that for irreducible chains, it is possible to define the stationary distribution.

Lemma A.2. *For any state i and j of an irreducible Markov chain, $\mathbb{E}_i(\tau_j) < \infty$.*

Corollary A.1. *Let P be the transition matrix of an irreducible Markov chain. Then there exists a unique probability distribution π satisfying $\pi = \pi P$. Moreover for all states z,*

$$\pi(z) = \frac{1}{\mathbb{E}_z \tau_z}. \tag{A.6}$$

Reversibility and time reversal

Consider a probability distribution π on $\mathcal{X}$ that satisfies

$$\pi_i P_{i,j} = \pi_j P_{j,i} \quad \text{for all } i, j \in \mathcal{X}. \tag{A.7}$$

The equations (A.7) are called the detailed balance equations.

Proposition A.1. *Let P be the transition matrix of a Markov chain with state space $\mathcal{X}$. Any distribution π satisfying the detailed balance equations (A.7) is stationary for P.*

In other words, if a chain $(X_t)_{t \geq 0}$ satisfies (A.7) and has stationary initial distribution, then the distribution of $(X_0, X_1, \ldots, X_n)$ is the same as the distribution of $(X_n, X_{n-1}, \ldots, X_0)$. For this reason, a chain with this property is called reversible. Proving that a chain is reversible is equal to proving that the matrix P is self-adjoint with respect to the inner product defined by the matrix $D = \text{diag}(\pi)$, that is

$$\langle Pv, w \rangle_\pi = \langle v, Pw \rangle_\pi, \tag{A.8}$$

for

$$\langle v, w \rangle_\pi = v^T D w. \tag{A.9}$$

We state the following lemma that proves an important property for the eigenvalues of a stochastic matrix of a reversible Markov chain.

Lemma A.3. *Let P be reversible with respect to π. Then the inner product space $(\mathbb{R}^{\mathcal{X}}, \langle \cdot, \cdot \rangle_\pi)$ has an orthonormal basis of real-valued eigenfunctions $\{f_j\}_{j=1}^{|\mathcal{X}|}$ corresponding to real eigenvalues $\{\lambda_j\}$.*

Convergence to the stationary distribution and relaxation time

The following theorem states that irreducible, aperiodic MCs converge to their respective stationary distribution.

Theorem A.1. *(Convergence theorem) Suppose that $(X_t)_{t\geq 0}$ is irreducible and aperiodic with stationary distribution π. Then there exist a constant $\alpha \in (0,1)$ and $C > 0$ such that*

$$\max_{i\in\mathcal{X}} \|(P^t)_{i,\cdot} - \pi\|_{\mathrm{TV}} \leq C\alpha^t \tag{A.10}$$

Here we used the definition of the total variation distance, which measures the distance between two probabilities μ and ν as the maximum difference between the probabilities assigned to a single event by the two distributions:

$$\|\mu - \nu\|_{\mathrm{TV}} = \max_{A\subseteq\mathcal{X}} |\mu(A) - \nu(A)|. \tag{A.11}$$

We may wonder how long it takes for the system to reach the stationary distribution. The answer to this question is given by the relaxation time t_{rel}. Define

$$\lambda_* = \max\{|\lambda| : \lambda \text{ is an eigenvalue of } P,\ \lambda \neq 1\}. \tag{A.12}$$

We call the difference $\gamma_* = 1 - \lambda_*$ the spectral gap. For a Markov chain $(X_t)_{t\geq 0}$ satisfying the detailed balance equation, by Lemma A.3 all the eigenvalues are real. Thus, we can label the eigenvalues in a decreasing order

$$1 = \lambda_1 > \lambda_2 \geq \ldots \geq \lambda_{|\mathcal{X}|}, \tag{A.13}$$

where λ_1 is the eigenvalue corresponding to the stationary distribution. For a reversible chain the spectral gap is $\gamma_* = 1 - \lambda_2$.

Definition A.6. *The relaxation time of a reversible Markov chain is defined as*

$$t_{rel} = \frac{1}{\gamma_*} = \frac{1}{1-\lambda_2}. \tag{A.14}$$

The spectral gap is a measure of how long it takes for the chain to converge to the stationary distribution. If t_{rel} is small then the chain converges quickly. Conversely, as t_{rel} grows the convergence becomes slower.

Appendix B

Codes

In this appendix we present the codes we used to obtain the numerical results presented in the thesis.

Relaxation time

Algorithm 1 was used to study the relaxation time of the Markov chain defined by Proposition 4.1. The core of the algorithm consists of a for loop that initialize the stochastic matrix **P** as the parameter **n**, representing the number of qubits in the circuit at each loop, varies from 2 to **n_max**. After the initialization of the matrix, at each loop the algorithm extracts the second largest eigenvalue and calculates the corresponding relaxation time. The result obtained is displayed in Figure 4.4.

Algorithm 1 Relaxation time of the Markov chain

```
n_max ← 1000
lambda_2 ← zero matrix of dimension 1 × n_max
t_rel ← zero matrix of dimension 1 × n_max
for n= 2 to n_max do
    P← zero matrix of dimension n×n
    Construct the transition matrix P
    lambda_2(n) ← second largest eigenvalue of P
    t_rel(n) ← 1/(1−lambda_2(n))
end for
```

Analysis of the expected magnitude of the gradient

Algorithm 2 Fixed number of qubits

```
n_val ← [100, 300]
p_max ← 800
for n in n_val do
    e ← [1, 0, ..., 0]                          ▷ Row vector of length n
    exp ← zero 1 × (p_max) matrix
    f← n×1 matrix, with f(i)= 1/3^i
    P← zero n×n matrix
    Construct the transition matrix P
    for p= 1 to p_max do
        exp(p) ←e·P^p·f
    end for
end for
```

Algorithm 2 is used to study how the decay of the expected magnitude of the gradient, which occurs as $p \to \infty$ (a result verified analytically), slows down as n increases. This result is depicted in Fig. 4.6. The algorithm considers two different values of n represented in the vector **n_max**. For the two possible values, the algorithm defines the stochastic matrix **P** (defined in Proposition 4.1), initialize the vector **f** that represents the vector defined in Equation 4.43 and computes, in a second loop, the expectation of the gradient when the initial configuration is represented by the probability vector **e**, as defined in Proposition 4.4.

Algorithm 3 Fixed number of gates

```
p_vals ← [100, 600]
n_max ← 300
exp ← zero 3 × n_max matrix
i ← 0
for p in p_vals do
    i←i+1
    for n= 2 to n_max do
        e ← [1, 0, ..., 0]                      ▷ Row vector of length n
        f← n×1 matrix, with f(i)= 1/3^i
        P← zero n×n matrix
        Construct the transition matrix P
        exp(i,n) ←e·P^p·f
    end for
end for
```

Algorithm 3 was used to study how the expected magnitude of the gradient behaves when p is fixed and n varies. As shown in Fig. 4.5, that depicts the results obtained with algorithm 3, the exponential decay of the latter does not

occur for shallow circuits. The algorithm computes the expected value of the gradient for two possible values of p, represented in the vector **p_vals**, and for n ranging from 2 to 300. The results are represented in the matrix **exp**.

Algorithm 4 was used to study the ratio $\frac{\left(\lambda_n^{(2)}\right)^p}{\lambda_n^{(n)}}$ for fixed valued of p. The results obtained, shown in Fig. 4.7, help us understand what has been obtained for shallow circuits. The algorithm computes the ratio for three possible values of p, represented in the vector **p_vals**, and for n ranging from 2 to 600. After the construction of the matrix **P**, we extract the second largest eigenvalue, **lambda_2**, and the smallest one, **lambda_n**. The values obtained for $\frac{\left(\lambda_n^{(2)}\right)^p}{\lambda_n^{(n)}}$ are stored in the matrix **ratio**.

Algorithm 4

```
p_vals ← [50, 100, 600]
n_max ← 400
ratio ← zero 3 × (n_max) matrix
i← 0
for p in p_vals do
    i←i+1
    for n= 2 to n_max do
        P← zero n×n matrix
        Construct the transition matrix P
        lambda_2 ← second largest eigenvalue of P
        lambda_n ← smallest eigenvalue of P
        ratio(i, n) ← (lambda_2)^p / lambda_n
    end for
end for
```

Bibliography

[1] Richard P Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6/7):467–488, 1982.

[2] Peter W. Shor. Introduction to quantum algorithms, 2001. URL https://arxiv.org/abs/quant-ph/0005003.

[3] Yuri Manin. Computable and uncomputable. *Sovetskoye Radio, Moscow*, 128:15, 1980.

[4] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22:563–591, 05 1980. doi: 10.1007/BF01011339.

[5] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985.

[6] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, pages 11–20, 1993.

[7] Daniel R Simon. On the power of quantum computation. *SIAM journal on computing*, 26(5):1474–1483, 1997.

[8] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, October 1997. ISSN 1095-7111. doi: 10.1137/s0097539795293172. URL http://dx.doi.org/10.1137/S0097539795293172.

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

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

[11] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, August 2018. ISSN 2521-327X. doi: 10.22331/q-2018-08-06-79. URL http://dx.doi.org/10.22331/q-2018-08-06-79.

[12] 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. *Nature Reviews Physics*, 3(9):625–644, August 2021. ISSN 2522-5820. doi: 10.1038/s42254-021-00348-9. URL http://dx.doi.org/10.1038/s42254-021-00348-9.

[13] Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9(1), November 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-07090-4. URL http://dx.doi.org/10.1038/s41467-018-07090-4.

[14] Zoë Holmes, Kunal Sharma, M. Cerezo, and Patrick J. Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum*, 3(1), January 2022. ISSN 2691-3399. doi: 10.1103/prxquantum.3.010313. URL http://dx.doi.org/10.1103/PRXQuantum.3.010313.

[15] John Napp. Quantifying the barren plateau phenomenon for a model of unstructured variational ansätze, 2022. URL https://arxiv.org/abs/2203.06174.

[16] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1), March 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-21728-w. URL http://dx.doi.org/10.1038/s41467-021-21728-w.

[17] Taylor L. Patti, Khadijeh Najafi, Xun Gao, and Susanne F. Yelin. Entanglement devised barren plateau mitigation. *Physical Review Research*, 3(3), July 2021. ISSN 2643-1564. doi: 10.1103/physrevresearch.3.033090. URL http://dx.doi.org/10.1103/PhysRevResearch.3.033090.

[18] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. Entanglement induced barren plateaus, 2021. URL https://arxiv.org/abs/2010.15968.

[19] Samson Wang, Enrico Fontana, M. Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J. Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications*, 12(1), November 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-27045-6. URL http://dx.doi.org/10.1038/s41467-021-27045-6.

[20] Phattharaporn Singkanipa and Daniel A. Lidar. Beyond unital noise in variational quantum algorithms: noise-induced barren plateaus and

limit sets. *Quantum*, 9:1617, January 2025. ISSN 2521-327X. doi: 10.22331/q-2025-01-30-1617. URL http://dx.doi.org/10.22331/q-2025-01-30-1617.

[21] M. Cerezo, Martin Larocca, Diego García-Martín, N. L. Diaz, Paolo Braccia, Enrico Fontana, Manuel S. Rudolph, Pablo Bermejo, Aroosa Ijaz, Supanut Thanasilp, Eric R. Anschuetz, and Zoë Holmes. Does provable absence of barren plateaus imply classical simulability? or, why we need to rethink variational quantum computing, 2024. URL https://arxiv.org/abs/2312.09121.

[22] M.A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information*. Cambridge Series on Information and the Natural Sciences. Cambridge University Press, 2000. ISBN 9780521635035. URL https://books.google.it/books?id=65FqEKQOfP8C.

[23] D. Pastorello. *Concise Guide to Quantum Machine Learning*. Machine Learning: Foundations, Methodologies, and Applications. Springer Nature Singapore, 2022. ISBN 9789811968976. URL https://books.google.sm/books?id=_dKkEAAAQBAJ.

[24] Antonio Anna Mele. Introduction to haar measure tools in quantum information: A beginneramp;apos;s tutorial. *Quantum*, 8:1340, May 2024. ISSN 2521-327X. doi: 10.22331/q-2024-05-08-1340. URL http://dx.doi.org/10.22331/q-2024-05-08-1340.

[25] Lorenzo Leone, Salvatore F.E. Oliviero, Lukasz Cincio, and M. Cerezo. On the practical usefulness of the hardware efficient ansatz. *Quantum*, 8: 1395, July 2024. ISSN 2521-327X. doi: 10.22331/q-2024-07-03-1395. URL http://dx.doi.org/10.22331/q-2024-07-03-1395.

[26] M. Schuld and F. Petruccione. *Machine Learning with Quantum Computers*. Quantum Science and Technology. Springer International Publishing, 2021. ISBN 9783030830984. URL https://books.google.sm/books?id=-N5IEAAAQBAJ.

[27] Kouhei Nakaji and Naoki Yamamoto. Expressibility of the alternating layered ansatz for quantum computation. *Quantum*, 5:434, April 2021. ISSN 2521-327X. doi: 10.22331/q-2021-04-19-434. URL http://dx.doi.org/10.22331/q-2021-04-19-434.

[28] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm, 2014. URL https://arxiv.org/abs/1411.4028.

[29] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, September 2017. ISSN 1476-4687. doi: 10.1038/nature23474. URL http://dx.doi.org/10.1038/nature23474.

[30] Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology*, 2(4):045001, August 2017. ISSN 2058-9565. doi: 10.1088/2058-9565/aa8072. URL http://dx.doi.org/10.1088/2058-9565/aa8072.

[31] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, November 2019. ISSN 2058-9565. doi: 10.1088/2058-9565/ab4eb5. URL http://dx.doi.org/10.1088/2058-9565/ab4eb5.

[32] David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985. ISSN 0364-0213. doi: https://doi.org/10.1016/S0364-0213(85)80012-4. URL https://www.sciencedirect.com/science/article/pii/S0364021385800124.

[33] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. Simulating a perceptron on a quantum computer. *Physics Letters A*, 379(7):660–663, March 2015. ISSN 0375-9601. doi: 10.1016/j.physleta.2014.11.061. URL http://dx.doi.org/10.1016/j.physleta.2014.11.061.

[34] Renxin Zhao and Shi Wang. A review of quantum neural networks: Methods, models, dilemma, 2021. URL https://arxiv.org/abs/2109.01840.

[35] Zoë Holmes, Nolan J. Coble, Andrew T. Sornborger, and Yi ğit Subaş ı. Nonlinear transformations in quantum computation. *Phys. Rev. Res.*, 5:013105, Feb 2023. doi: 10.1103/PhysRevResearch.5.013105. URL https://link.aps.org/doi/10.1103/PhysRevResearch.5.013105.

[36] Ken M. Nakanishi, Keisuke Fujii, and Synge Todo. Sequential minimal optimization for quantum-classical hybrid algorithms. *Physical Review Research*, 2(4), October 2020. ISSN 2643-1564. doi: 10.1103/physrevresearch.2.043158. URL http://dx.doi.org/10.1103/PhysRevResearch.2.043158.

[37] Andrea Mari, Thomas R. Bromley, and Nathan Killoran. Estimating the gradient and higher-order derivatives on quantum hardware. *Physical Review A*, 103(1), January 2021. ISSN 2469-9934. doi: 10.1103/physreva.103.012405. URL http://dx.doi.org/10.1103/PhysRevA.103.012405.

[38] Andrew Arrasmith, M. Cerezo, Piotr Czarnik, Lukasz Cincio, and Patrick J. Coles. Effect of barren plateaus on gradient-free optimization. *Quantum*, 5:558, October 2021. ISSN 2521-327X. doi: 10.22331/q-2021-10-05-558. URL http://dx.doi.org/10.22331/q-2021-10-05-558.

[39] Martin Larocca, Supanut Thanasilp, Samson Wang, Kunal Sharma, Jacob Biamonte, Patrick J. Coles, Lukasz Cincio, Jarrod R. McClean, Zoë Holmes, and M. Cerezo. A review of barren plateaus in variational quantum computing, 2024. URL https://arxiv.org/abs/2405.00781.

[40] Ricard Puig, Marc Drudis, Supanut Thanasilp, and Zoë Holmes. Variational quantum simulation: A case study for understanding warm starts. *PRX Quantum*, 6(1), January 2025. ISSN 2691-3399. doi: 10.1103/prxquantum.6.010317. URL http://dx.doi.org/10.1103/PRXQuantum.6.010317.

[41] Andrew Arrasmith, Zoë Holmes, M Cerezo, and Patrick J Coles. Equivalence of quantum barren plateaus to cost concentration and narrow gorges. *Quantum Science and Technology*, 7(4):045015, August 2022. ISSN 2058-9565. doi: 10.1088/2058-9565/ac7d06. URL http://dx.doi.org/10.1088/2058-9565/ac7d06.

[42] Jonas Haferkamp. Random quantum circuits are approximate unitary t-designs in depth $O\left(nt^{5+o(1)}\right)$. *Quantum*, 6:795, September 2022. ISSN 2521-327X. doi: 10.22331/q-2022-09-08-795. URL http://dx.doi.org/10.22331/q-2022-09-08-795.

[43] Alexander M. Dalzell, Nicholas Hunter-Jones, and Fernando G. S. L. Brandão. Random quantum circuits anticoncentrate in log depth. *PRX Quantum*, 3(1), March 2022. ISSN 2691-3399. doi: 10.1103/prxquantum.3.010333. URL http://dx.doi.org/10.1103/PRXQuantum.3.010333.

[44] Martín Larocca, Frédéric Sauvage, Faris M. Sbahi, Guillaume Verdon, Patrick J. Coles, and M. Cerezo. Group-invariant quantum machine learning. *PRX Quantum*, 3(3), September 2022. ISSN 2691-3399. doi: 10.1103/prxquantum.3.030341. URL http://dx.doi.org/10.1103/PRXQuantum.3.030341.

[45] Martin Larocca, Piotr Czarnik, Kunal Sharma, Gopikrishnan Muraleedharan, Patrick J. Coles, and M. Cerezo. Diagnosing barren plateaus with tools from quantum optimal control. *Quantum*, 6:824, September 2022. ISSN 2521-327X. doi: 10.22331/q-2022-09-29-824. URL http://dx.doi.org/10.22331/q-2022-09-29-824.

[46] Ricard Puig, Marc Drudis, Supanut Thanasilp, and Zoë Holmes. Variational quantum simulation: A case study for understanding warm starts. *PRX Quantum*, 6(1), January 2025. ISSN 2691-3399. doi: 10.1103/prxquantum.6.010317. URL http://dx.doi.org/10.1103/PRXQuantum.6.010317.

[47] Hela Mhiri, Ricard Puig, Sacha Lerch, Manuel S. Rudolph, Thiparat Chotibut, Supanut Thanasilp, and Zoë Holmes. A unifying account of warm start guarantees for patches of quantum landscapes, 2025. URL https://arxiv.org/abs/2502.07889.

[48] Armando Angrisani, Alexander Schmidhuber, Manuel S. Rudolph, M. Cerezo, Zoë Holmes, and Hsin-Yuan Huang. Classically estimating observables of noiseless quantum circuits, 2024. URL https://arxiv.org/abs/2409.01706.

[49] Daniel Gottesman. An introduction to quantum error correction and fault-tolerant quantum computation, 2009. URL https://arxiv.org/abs/0904.2557.

[50] D.P. DiVincenzo, D.W. Leung, and B.M. Terhal. Quantum data hiding. *IEEE Transactions on Information Theory*, 48(3):580–598, March 2002. ISSN 0018-9448. doi: 10.1109/18.985948. URL http://dx.doi.org/10.1109/18.985948.

[51] Daniel Gottesman. Surviving as a quantum computer in a classical world. *Textbook manuscript preprint*, 2016.

[52] Zak Webb. The clifford group forms a unitary 3-design, 2016. URL https://arxiv.org/abs/1510.02769.

[53] Huangjun Zhu, Richard Kueng, Markus Grassl, and David Gross. The clifford group fails gracefully to be a unitary 4-design, 2016. URL https://arxiv.org/abs/1609.08172.

[54] Jonas Helsen, Joel J. Wallman, and Stephanie Wehner. Representations of the multi-qubit clifford group. *Journal of Mathematical Physics*, 59 (7), July 2018. ISSN 1089-7658. doi: 10.1063/1.4997688. URL http://dx.doi.org/10.1063/1.4997688.

[55] D.A. Levin and Y. Peres. *Markov Chains and Mixing Times*. MBK. American Mathematical Society, 2017. ISBN 9781470429621. URL https://books.google.sm/books?id=f208DwAAQBAJ.