

POLITECNICO DI TORINO

Department of Mathematical Sciences

Master's Degree in Mathematical Engineering

Master's Degree Thesis

**Adiabatic Quantum Computing for Optimization
Problems with a Case Study on the Maximum
Independent Set Problem**



**Politecnico
di Torino**

Supervisors:

prof. Bartolomeo MONTRUCCHIO

prof. Federico DELLA CROCE

Candidate:

Ilaria Panuccio

Company Supervisor

Data Reply S.R.L

Dr. Davide Caputo

ACADEMIC YEAR 2023/2024

Acknowledgements

At the end of my journey at Politecnico di Torino, I realize that I have gained new awareness and achieved goals I never thought I could reach. There were some easy moments and others that were decidedly more difficult, but I am proud of never giving up in the face of challenges. This is the greatest advice I can give to anyone who wishes to undertake this journey in the future.

I have reached the conclusion of this Chapter thanks to the people who stood by my side and always believed in me. First and foremost, a heartfelt thank you goes to my entire family, who have never let me lack anything during these years in Turin. I thank the Politecnico for bringing Giusy, Teresa, Sara, Vittoria, and Enrico into my life, friends who have always supported me from the moment our paths crossed. Beyond the University realm, I thank my lifelong friends Carlotta, Chiara, Irene, Luisa, Tommaso and Matteo, who have always been there for me and always will be, like siblings. A special thanks to Michele, who has supported me all these years and on whom I know I can always rely, even though our lives have taken different ways.

Finally, I thank Blanca and Davide for allowing me to start my experience at Data Reply, trusting in my potential. Thanks to my colleagues in BU5 who welcomed me during the internship period and the months of writing the thesis, and with whom I still share my workdays. A special thanks to Luca and Mirko for their essential support in carrying out this thesis.

Summary

With the increasing attention and investments drawn to the Quantum Computing field, a number of different technologies and multiple algorithms have emerged in an effort to study the capabilities of this innovative computing paradigm. The thesis covers the study of the Quantum Computing formalism, the peculiarities characterizing different technologies and the potential use cases that are expected to benefit from applying Quantum Computing techniques, specifically in Combinatorial Optimization. The difference between the two main approaches of Quantum Computing, Digital Quantum Computing and Analog Hamiltonian Simulation, was studied. While the former performs computation with Quantum Gates and Circuits, the latter simulates complex quantum systems by replicating their Hamiltonian dynamics described by the Schrödinger equation. Among the methodologies of Analog Hamiltonian Simulation, Adiabatic Quantum Computing solves optimization problems by evolving a quantum system's Hamiltonian from its initial state to its final state, following the Adiabatic Theorem, wherein if the Hamiltonian changes slowly enough, a quantum system remains in its instantaneous ground state. In this work, this approach was directly applied to solve a specific combinatorial optimization problem, the *Maximum Independent Set* (MIS) problem, using two quantum annealers. The first, D-Wave, utilizes superconducting qubits, while the second, QuEra, is an emerging technology that exploits neutral atoms as computational units. This cutting-edge technology was studied in detail. In this regard, the neutral atom was defined and its possible states were analyzed, specifically identifying the Rydberg atom. The constituent components of QuEra's hardware were examined, as well as the techniques used for preparing atomic states and their dynamic evolution over time. Subsequently, the Maximum Independent Set (MIS) problem was defined, and through examples, various real-world applications were described, underlining that a graph can have multiple MISs and, for this reason, the solution to the problem may not be unique. Finally, the context of the MIS problem applied in this work was outlined, focusing on identifying the companies with the highest financial risk within a pool, located at the vertices of a small-sized graph whose edges are defined based on the covariance values between the companies. The solution to this problem is represented by the Minimum Vertex Cover of the reference graph, which is the complementary set of vertices of the Maximum Independent Set.

Sommario

Con l'attenzione crescente e gli investimenti nel campo del Quantum Computing, sono emerse diverse tecnologie e numerosi algoritmi che tentano di studiare le capacità di questo paradigma computazionale innovativo. La Tesi copre lo studio del formalismo del Quantum Computing, le peculiarità che caratterizzano diverse tecnologie e i potenziali *use cases* che si prevede beneficeranno dall'applicazione delle tecniche del Quantum Computing, in particolare nell'ottimizzazione combinatoria. È stata studiata anche la differenza tra i due principali approcci del Quantum Computing, il Digital Quantum Computing e l'Analog Hamiltonian Simulation. Mentre il primo processa dati con *Quantum Gates* e Circuiti, il secondo simula sistemi quantistici complessi replicando la loro dinamica descritta dall'equazione di Schrödinger. Tra le metodologie della Simulazione Hamiltoniana Analogica, l'*Adiabatic Quantum Computing* risolve problemi di ottimizzazione facendo evolvere l'Hamiltoniana di un sistema quantistico dallo stato iniziale a quello finale, seguendo il Teorema Adiabatico, secondo il quale se l'Hamiltoniana evolve abbastanza lentamente, un sistema quantistico rimane nel suo stato fondamentale istantaneo. In questo lavoro, tale approccio è stato applicato per risolvere un problema specifico di ottimizzazione combinatoria, il problema del *Maximum Independent Set* (MIS), utilizzando due annealer quantistici. Il primo, D-Wave, utilizza qubit superconduttori, mentre il secondo, QuEra, è una tecnologia emergente che sfrutta atomi neutri come unità computazionali. Questa tecnologia all'avanguardia è stata studiata nel dettaglio. A tal proposito, l'atomo neutro è stato definito e i suoi possibili stati sono stati analizzati, identificando l'atomo di Rydberg. Sono stati esaminati i componenti costitutivi dell'hardware di QuEra, così come le tecniche utilizzate per preparare gli stati atomici e la loro evoluzione dinamica nel tempo. Successivamente, è stato definito il problema MIS e attraverso esempi sono state descritte varie applicazioni reali, sottolineando che un grafo può avere più MIS e, per questo motivo, la soluzione del problema potrebbe non essere unica. Infine, è stato delineato il contesto del problema del MIS applicato in questo lavoro, che si concentra sull'identificazione delle aziende con il rischio finanziario più elevato all'interno di un gruppo, posizionate sui vertici di un grafo i cui archi sono definiti in base ai valori di covarianza tra le aziende. La soluzione a questo problema è rappresentata dal Minimum Vertex Cover del grafo di riferimento, che è l'insieme complementare dei vertici del Maximum Independent Set.

1	Introduction	1
2	Fundamentals of Quantum Computing	5
2.1	Quantum Information and Quantum States	5
2.2	The Qubits and the Computational Basis	6
2.3	State of a Qubit on the Bloch Sphere	7
2.4	Hilbert Spaces	8
2.5	Unitary Operators	11
2.6	Multiple Systems and Tensor Product	14
2.7	Pure and Mixed States and the Density Matrix	16
2.8	Measurement of Quantum Systems	18
2.9	Quantum Gates and Quantum Circuits	21
2.10	Entanglement	23
2.11	State Evolution of Closed Quantum Systems	25
2.12	Information Encoding	27
2.13	Conclusions	29
3	Quantum Hardware and Computation in the NISQ Era	31
3.1	The Five Quantum Hardware Criteria	31
3.2	The Principal Limitations of the Quantum Hardware	32
3.3	Hardware Outlook	34
3.3.1	Superconducting Qubits	34
3.3.2	Trapped Ion qubits	35
3.3.3	Photonic Qubits	36
3.4	The Analog Hamiltonian Simulation	37
3.4.1	Adiabatic Quantum Computing	38
3.5	Conclusions	42
4	The Neutral-Atom Quantum Technology	43
4.1	The Neutral and Rydberg atoms	43
4.2	Rydberg Platform	44
4.2.1	QuEra’s Neutral Atom Hardware	45
4.3	Neutral-atom Quantum Computing	46
4.3.1	Rydberg States and Qubits	46
4.3.2	Universal Quantum Computing with Neutral Atoms	47
4.3.3	The Rydberg Blockade	48
4.3.4	The Rydberg Hamiltonian	49
4.3.5	Measurements	50
4.3.6	The Principal Error Sources	50
4.4	Conclusions	51

5	The Maximum Independent Set problem	53
5.1	Definition of the Problem	53
5.2	Classical Approaches to Solve the MIS Problem and Related Problems	55
5.3	The Quantum Approach to MIS Problem	59
5.3.1	The Neutral Atom Approach	60
5.3.2	The Ising/QUBO Formulation	63
5.4	Conclusions	67
6	Application of the MIS Problem for Finding MVCs on a Specific UDG Case	69
6.1	The Optimization Problem	69
6.2	Check the Feasibility of the Graph Obtained from the Time Series Analysis	72
6.3	QuEra’s Solution	73
6.3.1	Step 1: map the UDG into atomic coordinates	74
6.3.2	Step 2: the adiabatic evolution parameters	75
6.3.3	Step 3: saving solutions and post processing	75
6.3.4	Step 4: result analysis	77
6.3.5	QuEra’s Results Considerations	78
6.4	D-Wave’s Solution	79
6.4.1	D-Wave’s Results Considerations	80
6.5	Two Technologies Compared	81
6.5.1	MIS Problem Solutions Benchmarks	83
6.6	Analysis of the results	84
6.7	Conclusions	85
7	Conclusions	87
A	Simulated Annealing	91
A.1	The SA algorithm’s steps	91
A.1.1	Problem definition	92
A.1.2	Exploration of the solutions space	92
A.1.3	Acceptance criterion	92
A.1.4	Temperature schedule	93

Chapter 1

Introduction

Quantum Computing represents a relatively new paradigm in which properties of quantum mechanics, are exploited to accelerate the solution of certain complex problems. Quantum mechanics is a branch of physics that emerged in the mid-20th Century. It challenged classical physics by describing both radiation and matter as having dual wave-particle properties, a concept that stands in great contrast to classical theories. While the laws of quantum physics find reason in microscopic reality, the laws of classical physics continue to prevail in the macroscopic world. The father of Quantum Computing was Richard Feynman, who stated in 1981 at MIT that classical devices are incapable of simulating quantum systems efficiently [1]. From this affirmation, the first quantum algorithm capable of solving the problem presented by Feynman was developed by Shor in 1994. This algorithm was published in 1997 in [2], explaining how to implement quantum circuits to factorize large integers in polynomial time exploiting the superposition and the quantum interference. In the following years, further algorithms were developed, capable of solving computationally difficult problems, such as Grover's search, published in 1996 in [3], which was tested in the first 2-qubit quantum computer in 1998 and which allowed quadratic speedup in database searches.

The difficulty of building quantum hardware capable of solving problems that require a discrete amount of computational time remains a major challenge. To exploit the quantum mechanics properties, the main obstacle lies in preventing the interaction between the Quantum Computing units and the external environment. If such an event were to occur, the system would be affected by decoherence, making the quantum information unreliable. For this reason, the goal of market-leading companies is to design quantum hardware that is completely *fault-tolerant* and *highly-scalable*, capable of producing reliable results even in the presence of errors during computation and without compromising performance as capacity increases. A possible solution is to utilize error correction algorithms capable of

detecting and correcting errors during the computation. These procedures primarily rely on redundancy in the number of physical resources, which, in the presence of an error in one or more computational units, prevent quantum information from being lost over time. These techniques face the maximum capacity limit of the hardware. For these reasons, the current phase of research in Quantum Computing is referred to as the “Noisy Intermediate-Scale Quantum (NISQ) era”, highlighting the limited capacity and the non-fault tolerance of the current devices, which may evolve into full-scale and noise-free quantum computers in future. Nowadays, various types of quantum hardware are built, categorized by the chosen computing units, such as superconducting qubits, neutral atoms, or ions. Note that the limited capacity in the current NISQ devices limits the types of algorithms that can be implemented, and therefore the goal is to demonstrate how Quantum Computing can solve certain problems with significantly shorter runtime than their classical counterparts. This context translates into “Quantum Supremacy,” a concept indicating the desire to prove that a quantum computer can solve a task where a classical supercomputer would not be feasible within a reasonable runtime. It’s important to emphasize that quantum computers are not aiming to surpass classical computers but rather to address certain tasks that would require a high number of classical resources.

A fundamental property of Quantum Computing is the *entanglement*, for which two qubits cannot be measured independently of each other but, once the state of one qubit has been measured, the state of the other qubit is also known in a deterministic way. This effect does not depend on the distance between the two entities and it is useful for different applications.

The classical and quantum realities are based on similar but different computational units. The classical devices are based on *bits*, while the quantum ones are based on the concept of *quantum bits*, or *qubits*. The main difference is that, while the bit can be in state 0 or 1 deterministically, the qubit is a quantum-mechanical object that is in the *superposition* of both states, a property that can be well exploited to create complex, multidimensional computational spaces with a consequent improvement in the efficiency and speed of performance. Therefore, the state of the qubit can only be described in a probabilistic way, since, with a certain probability, it can be found in the state 1 rather than in the state 0. In order to know the result, the procedure of *measurement* is of fundamental importance. In fact, at the instant the measurement is made, the qubit collapses into one of the two states with a certain probability and from that moment the knowledge of the state is deterministic. These concepts, along with the fundamentals of Quantum Computing, will be discussed in Chapter 2.

Chapter 3 will outline the five criteria that a quantum hardware should have. Further exploration will focus on the main differences between the two branches of

Quantum Computing, i.e. Universal Quantum Computing and Analog Hamiltonian Simulation, with a particular emphasis on Adiabatic Quantum Computing. The latter is a method of quantum computation based on studying the time evolution of a quantum system's Hamiltonian according to the Adiabatic Theorem. This method finds application in the field of Combinatorial Optimization, where the solution is sought in states encoding the minimum energy of the system, defined by the Hamiltonian's value. The techniques through which two technologies, QuEra and D-Wave, respectively utilize neutral atoms and superconducting qubits to develop the Adiabatic Protocol, will be analyzed.

Of particular interest in this work is the neutral atom technology QuEra, analyzed in detail in Chapter 4. Here, several foundational concepts will be defined, such as the Rydberg atom, the Rydberg Blockade phenomenon, and the Rydberg Hamiltonian, used to implement the Adiabatic Protocol. It will be examined how the latter allows for the natural encoding of constraints defining the Maximum Independent Set problem, a NP-complete problem discussed extensively in Chapter 5. This Chapter will also provide an overview of some classical approaches, as well as the Ising/QUBO formulation enabling resolution on D-Wave systems.

In Chapter 6, will be explained the context in which the MIS problem arises in this work. The objective will be to find the Minimum Vertex Cover of a graph, which is identified as the complementary set of the MIS and represents the companies with the highest financial risk within a broader group. The problem is mapped onto a small-sized graph belonging to the class of Unit Disk Graphs, which naturally encodes independence constraints in neutral atom technology. The results obtained from the quantum annealers D-Wave and QuEra will be analyzed, further validated by applying exact solver and classical heuristic, Simulated Annealing, detailed in Appendix A. It is further emphasized how the solution is not unique, as multiple feasible Maximum Independent Sets (MIS) and hence multiple feasible Minimum Vertex Covers (MVC) can exist in a graph. To achieve a unique solution, further market analysis should be conducted on companies not included in the set of common solutions, a task that goes beyond the scope of this thesis.

Chapter 2

Fundamentals of Quantum Computing

In this chapter, we will focus on the main concepts concerning the Quantum Computing, going in depth about its mathematics formulations and defining the most important notions.

2.1 Quantum Information and Quantum States

To explain quantum information, it is necessary to clearly understand the notion of information in the classical sense. These two types of information, although they have some distinct characteristics, are described mathematically in a similar way.

Suppose we have a system in which certain information is saved in the classical sense. This system can be defined with a configuration in an unambiguous way, as it can be observed in one of its possible states. For example, consider a six-sided die, once rolled, this would result in one of its six possible configurations in the set $\{1, 2, 3, 4, 5, 6\}$; if we think about the unit of computation in classical computing, the *bit*, it results in one of the two possible configurations in the set $\{0, 1\}$. So, if an object is considered as a collection of possible informations, this object is surely in one of its possible states.

Sometimes we can only describe the object's classical state in a probabilistic way. Consider the bit once again, assigning certain probabilities based, for example, on its past history, which determines whether it can be in state 0 or 1. Assuming that $\mathbb{P}(\text{bit} = 0) = \frac{1}{4}$, then $\mathbb{P}(\text{bit} = 1) = \frac{3}{4}$ and therefore its state can be denoted with the vector:

$$\vec{v} = \begin{pmatrix} \frac{1}{4} \\ \frac{3}{4} \end{pmatrix} \tag{2.1}$$

The convention adopted describes the states of a bit with the ordered set $\{0, 1\}$, therefore in (2.1) the probability of the bit being zero is placed at the top and the probability of being 1 at the bottom. Vectors of the form (2.1) for which the sum of the entries is 1 are called *probability vectors*.

Referring to a generic quantum system, the entries of the vector describing the *quantum state* are called *probability amplitudes* and the following properties hold:

- Their values belong to the set of the complex numbers \mathbb{C} ;
- The sum of their squares must be equal to 1. This property is known as *normalization constraint*.

Note that the normalization constraint is equivalent to the condition for which the Euclidean norm of the vector must be equal to 1, denoting that norm as

$$\|\psi\| = \sqrt{\sum_{k=1}^n |\alpha_k|^2} \quad (2.2)$$

where $\vec{v} = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$.

Although the difference between the two notations is subtle, Quantum Computing power comes precisely from these small mathematical changes.

Before delving further into the description of the quantum bit, called *qubit*, it is necessary to clarify the notations adopted. When referring to quantum systems it is common to use the *Dirac* notation, also known as *braket* notation, with which we indicate the column vectors with the *ket* symbol $|\cdot\rangle$ and the row vectors with the *bra* symbol $\langle\cdot|$. To be precise, the *bra* represents the *conjugate transpose* of the ket, i.e. $\langle\cdot| = |\cdot\rangle^\dagger$. With this notation, the *inner product* between the states $|\psi\rangle$ and $|\phi\rangle$ can be denoted as $\langle\psi|\cdot|\phi\rangle = \langle\psi|\phi\rangle$ and the Euclidean norm (2.2) has the following form:

$$\|\psi\| = \sqrt{\langle\psi|\psi\rangle}. \quad (2.3)$$

The *outer product* is obtained by multiplying a *bra* on the left and a *ket* on the right, i.e. $|\psi\rangle\langle\phi|$.

2.2 The Qubits and the Computational Basis

A qubit is essentially analogous to a classical bit, which is an object that can take on values in the set $\{0, 1\}$, but it can also exist in a quantum state. Precisely, *we are not certain about the qubit state until we observe (measure) the system*. As soon as this operation is carried out, the quantum system collapses into one of its possible states, allowing it to be studied in a deterministic way. In general, if a

probabilistic vector has 1 in the entry corresponding to the state a and 0 in the other entries, then that vector will be denoted as $|a\rangle$ and represents a standard basis vector. Referring to one qubit, the two possible standard basis, known as *computational basis*, are the following:

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle, \quad (2.4)$$

representing acceptable quantum state vectors, since $|1|^2 + |0|^2 = 1$. Remind that, before a *measurement*, the state of the system lives in a *superposition* of all possible states and its representation is a linear combination of standard basis vector with amplitude probabilities as scalar coefficients. Therefore, if a system described by one qubit is considered, its state is represented with the following linear combination:

$$\alpha_0|0\rangle + \alpha_1|1\rangle, \quad (2.5)$$

where $\alpha_0, \alpha_1 \in \mathbb{C}$ and $|\alpha_0|^2 + |\alpha_1|^2 = 1$.

For example, if $\alpha_0 = \frac{1}{\sqrt{2}}$ and $\alpha_1 = \frac{1}{\sqrt{2}}$, the following vector:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \quad (2.6)$$

is a quantum state vector, since its entries are complex numbers which satisfy the normalization constraint:

$$\left| \frac{1}{\sqrt{2}} \right|^2 + \left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2} + \frac{1}{2} = 1.$$

It is important to underline that the square of each probability amplitude represents precisely the probability of the quantum system of being in the state multiplied by it in the linear combination. In the case of the example (2.6), we have that the quantum state has probability $\left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2}$ of being in state 0 and probability $\left| \frac{1}{\sqrt{2}} \right|^2 = \frac{1}{2}$ of being in state 1.

2.3 State of a Qubit on the Bloch Sphere

In the previous Section, it is studied how to represent a quantum state as a linear combination, involving the computational basis and the amplitude probabilities. Consider an amplitude probability $\alpha \in \mathbb{C}$, it can be denoted as $e^{i\theta}|\alpha|$, where $|\alpha|$ is the non-negative real number associated to the magnitude of α and $e^{i\theta}$ is the

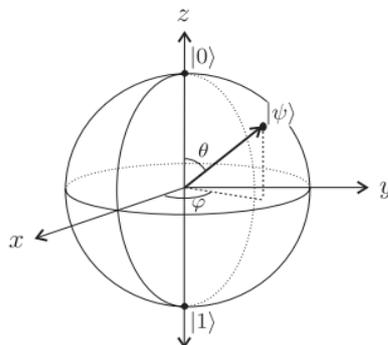


Figure 2.1: Representation of a qubit state on the Bloch sphere [4].

so called *phase factor*, with θ the *phase value*. In particular, the phase factor is defined as $e^{i\theta} = \frac{\alpha}{|\alpha|}$, its norm is equal to one and it is true that the states $|\psi\rangle$ and $e^{i\theta}|\psi\rangle$ are equivalent (they have the same *global phase*). On the other hand, two quantum states with different *relative phase*, for example, $|0\rangle + |1\rangle$ and $|0\rangle + e^{i\theta}|1\rangle$, are *physically* different from each other.

The state of a qubit can be described in an even more general form as follows:

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle \quad (2.7)$$

where $\varphi \in [0, \pi]$ and $\theta \in [0, 2\pi]$ are real parameters which allows to depict the state vector as a point on the surface of a 3-dimensional sphere, known as the *Bloch sphere*. Precisely, since a point on the surface of a sphere has Cartesian coordinates $(x, y, z) = (\sin\theta \cos\varphi, \sin\theta \sin\varphi, \cos\theta)$, a quantum state vector $|\psi\rangle$ is a *Bloch vector* $\vec{v} \in \mathbb{R}^3$ pointing from the origin to the surface, as represented in Figure 2.1.

2.4 Hilbert Spaces

In this Section, the main features of the spaces to which the unit quantum state belongs, i.e. Hilbert spaces, denoted as \mathcal{H} , are analyzed.

To define Hilbert space dimensions, we should refer to the degree of freedom of the system considered. To describe realistic quantum computational models, we will focus on systems with finite-dimensional state vectors but it is important to note that these systems can potentially be infinite-dimensional. In general, a quantum system with n qubits lies in a Hilbert space with dimension equal to the number of superposition states in which it can be, i.e. 2^n . Furthermore, since a quantum state

vector has complex entries, the reference n -dimensional Hilbert space is precisely \mathbb{C}^{2^n} .

Definition 1. Consider a Hilbert space \mathcal{H} with dimension 2^n . A set of 2^n vectors $B = \{|b_m\rangle\} \subseteq \mathcal{H}$ is an *orthonormal basis* for \mathcal{H} if:

$$\langle b_n | b_m \rangle = \delta_{nm} \quad \forall b_n, b_m \in B, \quad (2.8)$$

where δ_{nm} is the *Kronecker Delta Function*, that is equal to 1 when $n = m$ and 0 otherwise [5].

With the above definition, it holds that every state $|\psi\rangle \in \mathcal{H}$ can be written as

$$|\psi\rangle = \sum_{b_n \in B} \psi_n |b_n\rangle, \quad (2.9)$$

with $\psi_n = \langle b_n | \psi \rangle$ the *coefficients of ψ with respect to the basis $\{|b_n\rangle\}$* and $\| |\psi\rangle \| = \sum_i |\alpha_i|^2$.

The computational basis defined in Section 2.2 is an example of orthonormal basis defined with a one-qubit system. In fact:

$$\langle 0 | 1 \rangle = (1 \ 0) \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 0. \quad (2.10)$$

In addition to (2.10), there are several orthonormal basis in 2-dimensional Hilbert space. Among them, the *Hadamard basis*, denoted by $|+\rangle$ and $|-\rangle$, is of particular importance:

$$\begin{aligned} |+\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \\ |-\rangle &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \end{aligned} \quad (2.11)$$

It can be verified the normal constraint:

$$\begin{aligned} \| |+\rangle \|^2 &= \langle + | + \rangle = \frac{1}{2}(\langle 0 | + \langle 1 |)(|0\rangle + |1\rangle) = \frac{1}{2}(1 \ 1) \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 1 \Rightarrow \| |+\rangle \| = 1, \\ \| |-\rangle \|^2 &= \langle - | - \rangle = \frac{1}{2}(\langle 0 | - \langle 1 |)(|0\rangle - |1\rangle) = \frac{1}{2}(1 \ -1) \begin{pmatrix} 1 \\ -1 \end{pmatrix} = 1 \Rightarrow \| |-\rangle \| = 1, \end{aligned} \quad (2.12)$$

and the orthogonality:

$$\langle + | - \rangle = \frac{1}{2}(\langle 0 | + \langle 1 |)(|0\rangle - |1\rangle) = \frac{1}{2}(1 \ 1) \begin{pmatrix} 1 \\ -1 \end{pmatrix} = 0. \quad (2.13)$$

In order to best deal with a quantum system, it is appropriate to define the *operators*, i.e. functions that linearly map elements of a Hilbert space to other elements of that same space. These operators can be represented as matrices of size equal to the Hilbert space dimension in which they operate. In this context, consider O , $|\psi\rangle$, and $|\phi\rangle$ as a linear operator and two quantum states existing within the same Hilbert space. For the O operator, the following properties hold:

1. *Linearity*: $O(\alpha|\psi\rangle + \beta|\phi\rangle) = \alpha O|\psi\rangle + \beta O|\phi\rangle$, with $\alpha, \beta \in \mathbb{R}$.
2. *Banality*: if O is the *identity operator* I , then $I|\psi\rangle = |\psi\rangle \forall |\psi\rangle \in \mathcal{H}$
3. *Hermitian conjugate*: the conjugate Hermitian operator O^\dagger of O is such that $\langle O\psi|\phi\rangle = \langle\psi|O^\dagger\phi\rangle$, where $O^\dagger = (\overline{O})^T$.

Consider the outer product of a vector with itself, it defines a linear operator which, if applied to a quantum state vector $|\gamma\rangle$, acts in the following way:

$$|\psi\rangle\langle\psi||\gamma\rangle \rightarrow |\psi\rangle\langle\psi|\gamma\rangle = \langle\psi|\gamma\rangle|\psi\rangle.$$

Such operator is called *orthogonal projector* and projects the state $|\gamma\rangle$ in \mathcal{H} to the 1-dimensional subspace of \mathcal{H} spanned by $|\psi\rangle$.

The following definition refers to a class of operators useful to describe the *Hamiltonian* and the *observables* of a quantum system, concepts that we will explore in the next sections.

Definition 2. An operator O in a Hilbert space \mathcal{H} is called *Hermitian* if:

$$O^\dagger = O$$

i.e. it is equal to its own Hermitian conjugate.

For an Hermitian operator, the *Spectral Decomposition Theorem* holds:

$$H = \sum_i \lambda_i |\psi_i\rangle\langle\psi_i| \quad (2.14)$$

where $\lambda_i \in \mathbb{R}$ are the eigenvalues and $|\psi_i\rangle$ are orthonormal eigenvectors. In other words, the normal operators are always diagonalizable.

For example, consider the X operator, defined as:

$$\begin{aligned} X|0\rangle &= 1, \\ X|1\rangle &= 0, \end{aligned} \quad (2.15)$$

with matrix representation:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad (2.16)$$

and spectrum $\{-1, 1\}$, defining the matrix

$$\Lambda = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \quad (2.17)$$

The corresponding eigenvectors are, respectively, $(\frac{1}{\sqrt{2}} \ \frac{1}{\sqrt{2}})^T$ and $(\frac{1}{\sqrt{2}} \ -\frac{1}{\sqrt{2}})^T$, for which it can be associated the matrix:

$$P = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}. \quad (2.18)$$

The diagonalization of X is defined as:

$$X = P^T \Lambda P = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}. \quad (2.19)$$

Using the Dirac notation, we can write:

$$X = |0\rangle\langle 1| + |1\rangle\langle 0| \quad (2.20)$$

$$P = |+\rangle\langle 0| + |-\rangle\langle 1| = |0\rangle\langle 0| - |1\rangle\langle 1|$$

where the eigenvectors are the Hadamard basis defined in (2.11):

$$\begin{aligned} |+\rangle &\equiv \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \\ |-\rangle &\equiv \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle. \end{aligned}$$

2.5 Unitary Operators

In this Section, *unit operators* as operations on quantum states are addressed.

A Unitary Operator U is a complex matrix such that:

$$\begin{aligned} UU^\dagger &= \mathbb{1} \\ U^\dagger U &= \mathbb{1} \end{aligned} \quad (2.21)$$

where $\mathbb{1}$ is the Identity matrix.

It holds that an $N \times N$ matrix is unitary if and only if

$$\|U|\psi\rangle\| = \|\psi\rangle\| \quad (2.22)$$

for every N -dimensional quantum state $|\psi\rangle$. This means that, if U is multiplied by any vector, its Euclidean norm doesn't change and it results in an another

quantum state vector. For this reasons, the unitary matrices represent the set of linear mappings that map quantum state vectors to others quantum state vectors [6].

We can introduce the most significant Quantum Computing unitary operators with the following examples.

Example 1. *Pauli Operators*, identified by the following four matrices:

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \quad (2.23)$$

These operators are often denoted as $X \equiv \sigma_x$, $Y \equiv \sigma_y$ and $Z \equiv \sigma_z$.

By multiplying X on the left to the state vectors $|0\rangle$ and $|1\rangle$, we obtain:

$$\begin{aligned} X|0\rangle &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle, \\ X|1\rangle &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle. \end{aligned} \quad (2.24)$$

Note that this operation maps the state $|0\rangle$ to the state $|1\rangle$ and vice versa. For this reason, the X operator is also known as *bit flip* or *NOT* operator.

By multiplying Z on the left to the states $|0\rangle$ and $|1\rangle$, we obtain:

$$\begin{aligned} Z|0\rangle &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle, \\ Z|1\rangle &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} = -|1\rangle. \end{aligned} \quad (2.25)$$

Note that Z maps the state $|0\rangle$ to $|0\rangle$ and the state $|1\rangle$ to $-|1\rangle$ and it is also called *Phase Operator*.

Example 2. *Hadamard Operator* described by the matrix:

$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}. \quad (2.26)$$

As done in the previously example, multiplying H on the left to the states $|0\rangle$ and $|1\rangle$, we obtain:

$$\begin{aligned} H|0\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = |+\rangle, \\ H|1\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} = |-\rangle. \end{aligned} \quad (2.27)$$

This result shows that the Hadamard operator maps $|0\rangle$ to $|+\rangle$ and $|1\rangle$ to $|-\rangle$. If we apply H to the left of both $|+\rangle$ and $|-\rangle$ vectors, we obtain:

$$\begin{aligned} H|+\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle, \\ H|-\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle. \end{aligned} \quad (2.28)$$

Putting together the results obtained in (2.27) and (2.28), we have:

$$\begin{aligned} H|0\rangle &= |+\rangle, \\ H|1\rangle &= |-\rangle, \\ H|+\rangle &= |0\rangle, \\ H|-\rangle &= |1\rangle. \end{aligned} \quad (2.29)$$

Example 3. *Phase operator* defined by the following matrix:

$$P_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}, \quad (2.30)$$

with $\theta \in \mathbb{R}$. From the set of the phase operators, we have two important examples:

$$\begin{aligned} S &= P_{\pi/2} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}, \\ T &= P_{\pi/4} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix}. \end{aligned} \quad (2.31)$$

Applying the T operator to the state $|+\rangle$ and exploiting the linearity of the matrices multiplication, we have:

$$T|+\rangle = T \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \right) = \frac{1}{\sqrt{2}}T|0\rangle + \frac{1}{\sqrt{2}}T|1\rangle. \quad (2.32)$$

Developing the same calculations of previously examples, we obtain:

$$\begin{aligned} T|0\rangle &= |0\rangle, \\ T|1\rangle &= \frac{i+1}{\sqrt{2}}|1\rangle. \end{aligned} \quad (2.33)$$

Finally, substituting in (2.32), we can find:

$$T|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1+i}{2}|1\rangle \quad (2.34)$$

Note that $\mathbb{1}$ and Z operations defined in (2.23) are, respectively, the phase matrix P_0 and P_π .

The examples above were only discussed using a single unitary operator. The computation involving the composition of unitary operators is represented simply by multiple matrix multiplications, where each matrix corresponds to a unitary operator.

2.6 Multiple Systems and Tensor Product

In this Section, the handling of quantum information in the context of multiple systems is explored.

In Section 2.4, it is studied that quantum states live in a 2^n -dimensional Hilbert space, with n the number of qubits needed to describe that state. Consider the interaction of two Hilbert spaces \mathcal{H}_1 and \mathcal{H}_2 with dimensions $N = 2^n$ and $M = 2^m$. A useful tool for representing this combination is the *tensor product*, denoted by the symbol \otimes . Therefore, considering two different vector states $|\psi\rangle$ and $|\phi\rangle$ from, respectively, \mathcal{H}_1 and \mathcal{H}_2 , their tensor product $|\psi\rangle \otimes |\phi\rangle$ lives in a new Hilbert space $\mathcal{H}_1 \otimes \mathcal{H}_2$ with dimension $N \times M$. Explicitly:

$$|\psi\rangle = \begin{pmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \end{pmatrix}, \quad |\phi\rangle = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_M \end{pmatrix},$$

with their tensor product defined as:

$$|\psi\rangle \otimes |\phi\rangle = \begin{pmatrix} \psi_1 \cdot \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_M \end{pmatrix} \\ \vdots \\ \psi_N \cdot \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_M \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \psi_1\phi_1 \\ \psi_1\phi_2 \\ \vdots \\ \psi_1\phi_N \\ \psi_2\phi_1 \\ \psi_2\phi_2 \\ \vdots \\ \psi_2\phi_M \\ \vdots \\ \psi_N\phi_1 \\ \psi_N\phi_2 \\ \vdots \\ \psi_N\phi_M \end{pmatrix}. \quad (2.35)$$

The expression $|\psi\rangle \otimes |\phi\rangle$ is usually simplified with the Dirac notation $|\psi\phi\rangle$. To better understand this concept, consider all the possible tensor products of the

vectors that compose the orthonormal computational basis, i.e. $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$. Given that $|0\rangle = (1\ 0)^T$ and $|1\rangle = (0\ 1)^T$, it holds:

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \quad (2.36)$$

The tensor product between two quantum state vectors is still a quantum state vector. This can be seen through the computation of the Euclidean norm:

$$\begin{aligned} \|\psi\rangle \otimes |\phi\rangle\| &= \sqrt{\sum_{(a,b) \in \Sigma \times \Gamma} |\langle ab|\psi \otimes \phi\rangle|^2} = \\ &= \sqrt{\sum_{a \in \Sigma} |\langle a|\psi\rangle|^2 \sum_{b \in \Gamma} |\langle b|\phi\rangle|^2} = \\ &= \|\psi\rangle\|^2 \cdot \|\phi\rangle\|^2 = 1. \end{aligned} \quad (2.37)$$

The last equation holds since $|\psi\rangle$ and $|\phi\rangle$ are quantum state vectors.

The concept of tensor product can be easily generalized when examining three or more systems. Consider n different systems X_1, \dots, X_n with classical quantum state vectors ψ_1, \dots, ψ_n respectively. The *product state* of the joint system (X_1, \dots, X_n) is defined as:

$$|\psi\rangle = |\psi_1\rangle \otimes \dots \otimes |\psi_n\rangle \quad (2.38)$$

It is important to underline that the concepts of product state and that of *independence* are closely related. In fact, when a set of systems X_1, X_2, \dots, X_n is in the product state defined by $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle$, the system X_i is in the corresponding quantum state $|\psi_i\rangle$ and the states of the n -dimensional systems have nothing to do with each other.

While the tensor product reflects the concept of independence between quantum systems, there are quantum states of multiple systems that cannot be represented as the tensor product of the involved states. In this case, a characteristic phenomenon of quantum mechanics called *entanglement* is referred to, which reflects correlation between systems. In other words, *every quantum state vector which cannot be represented as a tensor product represents an entangled state*. We will cover the entanglement and its properties in the Section 2.10.

2.7 Pure and Mixed States and the Density Matrix

Until now, the state of a quantum system has always been represented as a column vector or as a linear combination of orthogonal quantum state vector, with the properties defined in Section 2.1. This representation refers to the *pure state*. Suppose now to have a qubit which has a probability $\frac{1}{3}$ of being in the pure state $|\psi_1\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and a probability $\frac{2}{3}$ of being in the pure state $|\psi_2\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. In this way, we can represent the system state putting together the two possible pure states $|\psi_1\rangle$ and $|\psi_2\rangle$ with the corresponding probability distribution, forming a *mixed state*. This definition can be generalized by considering an arbitrary number of pure states of n qubits. In such case, a mixed state over n qubits can be represented with the following set:

$$\{(|\psi_1\rangle, p_1), (|\psi_2\rangle, p_2), \dots, (|\psi_k\rangle, p_k)\}, \quad (2.39)$$

which means that the system is in the pure n -qubits state $|\psi_i\rangle$ with probability p_i , for $i = 1, \dots, k$. Note that, with this definition, a pure state is a mixed state where all the probabilities p_i are zero except one.

There is an equivalent and more useful representation of mixed state, which involves special kinds of operators on the Hilbert space, called *density operators*, represented by *density matrices*, defined as:

$$\rho = |\psi\rangle\langle\psi|. \quad (2.40)$$

More general, the density operator for an ensemble of pure states such as (2.39) is defined as:

$$\rho = \sum_{i=1}^k p_i |\psi_k\rangle\langle\psi_k|, \quad (2.41)$$

i.e. a linear combination of pure state, with the probabilities p_i of the system to be in the pure state $|\psi_i\rangle$ as scalar coefficients. For example, the pure state $|\psi\rangle = \alpha_1|0\rangle + \alpha_2|1\rangle$ with $|\alpha_1|^2 + |\alpha_2|^2 = 1$, has density matrix equal to:

$$\rho = |\psi\rangle\langle\psi| = \begin{pmatrix} |\alpha_1|^2 & \alpha_1\alpha_2^* \\ \alpha_1^*\alpha_2 & |\alpha_2|^2 \end{pmatrix}, \quad (2.42)$$

where the symbol $*$ denote the complex conjugation operation. From ρ , some useful properties of the considered quantum system can be obtained. Indeed, the diagonal terms indicate the probabilities associated with the system being in the respective states. Specifically, if $|\alpha_1|^2 > 0$ and $|\alpha_2|^2 > 0$, then the system is in a superposition state of $|0\rangle$ and $|1\rangle$. The off-diagonal ρ entries of a pure state are

positive values representing the *coherence* between the quantum states, a concept that will be further explored in Chapter 3.

Consider the density matrix of a generic mixed state:

$$\rho_{mixed} = |\alpha_1|^2|0\rangle\langle 0| + |\alpha_2|^2|1\rangle\langle 1| = \begin{pmatrix} |\alpha_1|^2 & 0 \\ 0 & |\alpha_2|^2 \end{pmatrix}, \quad (2.43)$$

it can be seen that the off-diagonal entries are zero, indicating the absence of interference terms. This is due to the fact that the quantum density matrix of a mixed state is defined as a convex combination of density matrices corresponding to pure states. Since pure states are orthogonal to each other, the off-diagonal terms of mixed states density have value equal to zero.

In Section 2.3 it is observed the equivalence between two quantum states that differ in *global phase*. Exploiting the density matrix of a quantum state vector to represent all the states formed by all the equivalent vectors, it holds that two different vectors $|\psi\rangle$ and $|\phi\rangle$ define the same density matrix if and only if they differ in global phase, i.e. $|\phi\rangle = e^{i\theta}|\psi\rangle$.

In Section 2.3, it is also studied how to represent a pure state as a vector pointing to the surface of the Bloch sphere. Mixed states can also be represented by vectors within this sphere; in fact, they correspond to points in the interior of the Bloch sphere. That is, if $\rho = \sum_{i=1}^k p_i |\psi_k\rangle\langle\psi_k|$ and if the Bloch vector for the pure state $|\psi_i\rangle$ is $\vec{v} = (\sin\theta \cos\varphi, \sin\theta \sin\varphi, \cos\varphi) = (\alpha_{x,i}, \alpha_{y,i}, \alpha_{z,i})$, then we can represent the mixed state ρ with the following Bloch vector:

$$\rho = \sum_i p_i (\alpha_{x,i}, \alpha_{y,i}, \alpha_{z,i}) = \left(\sum_i p_i \alpha_{x,i}, \sum_i p_i \alpha_{y,i}, \sum_i p_i \alpha_{z,i} \right), \quad (2.44)$$

which is a linear combination of probabilities and elements of the vector representing the pure state $|\psi_i\rangle$.

Having discussed mixed states and tensor products, a fundamental principle in quantum mechanics is now introduced: the *No-Cloning Theorem*. Precisely, cloning a pure state $|\psi\rangle \in \mathbb{C}^n$ is a procedure which results in a separable state $|\psi\rangle \otimes |\psi\rangle$ [7]. This method starts with the addition of an ancilla system which is not related to the state being cloned, i.e. $|\psi\rangle \otimes |0\rangle$ and aims to find a unitary operator \mathbf{U} such that:

$$\mathbf{U}(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle. \quad (2.45)$$

The above procedure should work for any state in the Hilbert space \mathbb{C}^n . Therefore, considering a different pure state $|\phi\rangle$ and the inner product with $|\psi\rangle$ together with $|0\rangle$, we obtain:

$$\langle 0| \langle \phi| |\psi\rangle |0\rangle = \langle 0| \langle \phi| \mathbf{U}^\dagger \mathbf{U} |\psi\rangle |0\rangle = \langle \phi| \langle \phi| \psi\rangle |\psi\rangle, \quad (2.46)$$

in which it holds that:

$$\langle \phi | \psi \rangle = \langle \phi | \psi \rangle^2, \quad (2.47)$$

which implies:

$$\langle \phi | \psi \rangle = 1 \text{ or } \langle \phi | \psi \rangle = 0.$$

Therefore, as stated by the *No-Cloning Theorem*, *it is impossible to create an identical copy of an arbitrary unknown quantum state*. This statement has important consequences in Quantum Computing, since exact copy of data is not possible, ensuring the security of quantum communication and highlighting the inherent limitations of quantum operations.

2.8 Measurement of Quantum Systems

In this Section, the quantum systems measurements will be analyzed. Remind that a quantum system is a physical system which evolves over time, therefore its state vector is actually a function of time $|\psi(t)\rangle$. Furthermore, we always refer to closed systems with the ideally condition for which they don't interact with the external environment. Assuming these considerations as true, it holds that *the evolution of the state vector of a closed quantum system is linear*. That is, if the operator \mathbf{U} maps $|\psi\rangle$ to $\mathbf{U}|\psi\rangle$, then:

$$\mathbf{U} \sum_i \alpha_i |\psi_i\rangle = \sum_i \alpha_i \mathbf{U} |\psi_i\rangle \quad (2.48)$$

with $\sum_i |\alpha_i|^2 = 1$. In Section 2.5 it is outlined that the only operators which preserve such norm are the *unitary* operators. This allows to state the *Evolution Postulate*.

Evolution Postulate: the time-evolution of the state of a closed quantum system is described by a unitary operator. That is, for any closed system evolution, there exists a unitary operator \mathbf{U} such that, if the initial system is $|\psi_1\rangle$, then, after the evolution, the state of the system will be

$$|\psi_2\rangle = \mathbf{U}|\psi_1\rangle. \quad (2.49)$$

Suppose that, after a certain time in which an evolution is taking place, we want to measure the system in order to know some of its properties. It is important to underline that when a measurement is taken, the system interact in some ways with the external environment, violating the conditions of the evolution postulate. For this reason, it is worth noting that the evolution of a system during a measurement is not unitary, but it is sufficient to consider a measurement postulate that considers the different aspects.

Consider an apparatus which provides a classical description of the measurement outcome, that is, if a state is described by the linear combination $\sum_i \alpha_i |i\rangle$, then the outcome of the measurement is the label i with probability $|\alpha_i|^2$. Through this reasoning, combining the postulate of evolution, the *Measurement Postulate* can be stated.

Measurement Postulate: for a given orthonormal basis $B = \{|b_i\rangle\}$ of a state space \mathcal{H}_A for a system A , it is possible to perform a *Von Neumann Measurement* on system \mathcal{H}_A with respect to the basis B that, given a state:

$$|\psi\rangle = \sum_i \alpha_i |b_i\rangle \quad (2.50)$$

outputs a label i with probability $|\alpha_i|^2$ and leaves the system in state $|b_i\rangle$.

The *Von Neumann measurement* belongs to the set of the so called *projective measurements*, i.e. defined by an orthogonal projector P with following properties:

1. *Hermitian:* $P^\dagger = P$
2. *Idempotent:* $P^2 = P$

This type of measurement projects the input state into an orthogonal subspace, with probability equal to the squared value of the probability amplitude of the $|\psi\rangle$ component in that subspace. This is true since each identity operator I can be written as a sum of orthogonal projectors P_i . Therefore, a projective measurement will result in label i with probability $\mathbb{P}(i) = \langle \psi | P_i | \psi \rangle$ and leaves the rest of the system in a normalized state, defined as $\frac{P_i |\psi\rangle}{\sqrt{\mathbb{P}(i)}}$.

Another significant insight regarding projective measurements involves certain Hermitian operators, known as *observables*, which operate on the space where the state of the system belongs. Consider an observable M . It, being Hermitian, admit the following *spectral decomposition*:

$$M = \sum_i m_i P_i \quad (2.51)$$

where m_i are the (real) eigenvalues of M and P_i are orthogonal projectors on the eigenspace of M . Therefore, measuring an observable is equivalent to carrying out a projective measurement with respect to the decomposition of the unitary operator $I = \sum_i P_i$ and the corresponding result will be the real eigenvalue m_i .

The Measurement Postulate also confirms the equivalence between two states that differ in global phase. In fact, by measuring the state $e^{i\theta} |\psi\rangle = \sum_i \alpha_i e^{i\theta} |\psi_i\rangle$, the result i will be obtained with probability

$$p(i) = \alpha_i^* e^{-i\theta} \alpha_i e^{i\theta} = \alpha_i^* \alpha_i = |\alpha_i|^2, \quad (2.52)$$

which is equivalent to the probability of obtaining the result i from the state $|\psi\rangle = \sum_i \alpha_i |\psi_i\rangle$. In other words, measurements of states that differ in global phase will result in the same statistics.

Let's see what happens when a measurement is made of a system composed by multiple states, such as:

$$|\psi\rangle = \sqrt{\frac{5}{11}}|00\rangle + \sqrt{\frac{1}{11}}|01\rangle + \sqrt{\frac{3}{11}}|10\rangle + \sqrt{\frac{2}{11}}|11\rangle. \quad (2.53)$$

From the Measurement Postulate, the state $|00\rangle$ results with probability $\frac{5}{11}$, state $|01\rangle$ with probability $\frac{1}{11}$, the state $|10\rangle$ with probability $\frac{3}{11}$ and the state $|11\rangle$ with probability $\frac{2}{11}$. From this, we will have that the probability of obtaining 0 in the first qubit will be the sum $\frac{5}{11} + \frac{1}{11} = \frac{6}{11}$. Consideration could also be given to what would occur if only the first qubit were measured. In this regard, rewrite the state (2.53) highlighting the first qubit:

$$|\psi\rangle = \sqrt{\frac{6}{11}}|0\rangle \left(\sqrt{\frac{5}{6}}|0\rangle + \sqrt{\frac{1}{6}}|1\rangle \right) + \sqrt{\frac{5}{11}}|1\rangle \left(\sqrt{\frac{3}{5}}|0\rangle + \sqrt{\frac{2}{5}}|1\rangle \right). \quad (2.54)$$

From this, the probability of getting 0 in the first qubit is $\frac{6}{11}$ and the state of the second qubit lives in the superposition $\left(\sqrt{\frac{5}{6}}|0\rangle + \sqrt{\frac{1}{6}}|1\rangle \right)$.

Referring to *Example 2* of Section 2.5, consider a qubit prepared in one of the two states of the Hadamard basis. In this case, once the system has been measured, it has a 50% probability of being in the state $|0\rangle$ and the other 50% probability of being in the state $|1\rangle$. In fact,

$$\begin{aligned} |+\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \\ |-\rangle &= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle, \end{aligned} \quad (2.55)$$

with the square values of the amplitude probabilities equals to $\frac{1}{2}$. Therefore, a measurement of the states $|+\rangle$ and $|-\rangle$ provides no information about which of the two states were originally prepared [6]. If instead we first applied the Hadamard operation to the states $|+\rangle$ and $|-\rangle$ and then carried out the measurement, we find the state $|0\rangle$ whenever the qubit was prepared in $|+\rangle$ and the state $|1\rangle$ whenever the preparation was in $|-\rangle$. For this reason, it is said that the states $|+\rangle$ and $|-\rangle$ can be *discriminated perfectly*.

Return now to the notion of density matrix described in Section 2.7. Suppose we want to measure the state with the density operator $\rho = |\psi\rangle\langle\psi|$ in the computational basis. In general, for a state defined as a linear combination of elements in

the orthogonal basis $|\psi\rangle = \sum_i \alpha_i |b_i\rangle$, the amplitude probabilities α_i can be computed as $\alpha_i = \langle b_i | \psi \rangle = \langle b_i | \psi \rangle$, therefore $|\alpha_i|^2 = \alpha_i^* \alpha_i = \langle \psi | b_i \rangle \langle b_i | \psi \rangle$. In this case, the probability to obtain the outcome 0 from the measurement can be computed as:

$$\langle 0 | \psi \rangle \langle 0 | \psi \rangle = \langle 0 | \rho | 0 \rangle \quad (2.56)$$

which results in a real number. Remember that, in general, every number is the trace of a given 1x1 matrix and therefore the equation (2.56) can be written as follows:

$$\langle 0 | \psi \rangle \langle 0 | \psi \rangle = \text{Tr}(\langle 0 | \psi \rangle \langle 0 | \psi \rangle) = \text{Tr}(|0\rangle \langle 0 | \psi \rangle \langle \psi |). \quad (2.57)$$

From these steps, the probability to obtain the outcome $|1\rangle$ is $\text{Tr}(|1\rangle \langle 1 | \psi \rangle \langle \psi |)$. Further generalizing this notion to the case of a mixed state described by the equation (2.41), the probability to have $|0\rangle$ as a measurement result is given by:

$$\begin{aligned} \sum_i p_i \text{Tr}(|0\rangle \langle 0 | \psi_i \rangle \langle \psi_i |) &= \text{Tr} \left(\sum_i p_i (|0\rangle \langle 0 | \psi_i \rangle \langle \psi_i |) \right) \\ &= \text{Tr} \left(|0\rangle \langle 0 | \sum_i p_i \psi_i \rangle \langle \psi_i | \right) = \text{Tr}(|0\rangle \langle 0 | \rho). \end{aligned} \quad (2.58)$$

Hence, the essential aspect for quantum system measurements lies solely in the density operator itself, rather than any decomposition thereof.

2.9 Quantum Gates and Quantum Circuits

In Section 2.5, unitary operators and their properties are defined, using the Pauli, Hadamard and Phase operators as examples. These operators act on a 2-dimensional one-qubit system, and are therefore referred to as *1-qubit gates*. In general, a *quantum logic gate* is any transformation induced by some unitary operator acting on a system composed of one or more qubits. In particular, if the system is composed of n qubits, the quantum gate will be represented by a unit matrix with dimension $2^n \times 2^n$ in a 2^n -dimensional Hilbert space.

Having defined 1-qubit gates with examples such as the Pauli and Hadamard gates, two particular *2-qubit gates* are now introduced with the following examples.

Example 4. Controlled- U Gate. For any single qubit unitary operation U , a *controlled- U gate* is a two-qubit gate, with one *control* qubit and one *target* qubit [8]. Its matrix form is as follows:

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad (2.59)$$

and acts in this way:

- If the control qubit is $|1\rangle$, \mathbf{U} is applied to the target qubit;
- If the control qubit is $|0\rangle$, the target qubit is left alone,

defining the following maps:

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle & |01\rangle &\rightarrow |01\rangle, \\ |10\rangle &\rightarrow |1\rangle\mathbf{U}|0\rangle & |11\rangle &\rightarrow |1\rangle\mathbf{U}|1\rangle. \end{aligned} \quad (2.60)$$

One of the best known controlled gates is the *CNOT*-gate in which, taking into consideration the map (2.60), the \mathbf{U} operator is replaced by the *NOT* operator (also known as *X-gate*), which defines the following maps:

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle & |01\rangle &\rightarrow |01\rangle, \\ |10\rangle &\rightarrow |11\rangle & |11\rangle &\rightarrow |10\rangle. \end{aligned} \quad (2.61)$$

Example 5. The *Swap-gate* is defined with the following matrix:

$$SWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (2.62)$$

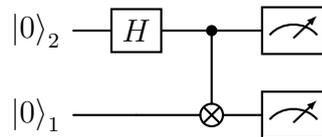
Its principal feature is that it interchanges the quantum state of two qubits and, in particular, \sqrt{SWAP} :

$$\sqrt{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1+i}{2} & \frac{1-i}{2} & 0 \\ 0 & \frac{1-i}{2} & \frac{1+i}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.63)$$

is its controlled version in the sense that the swap between the two qubits is controlled by the state of a third qubit.

In gate-model, the quantum computation is executed through a sequence of gates organized in structures known as *circuits*, into which the fundamental Quantum Computing units, the qubits, are the inputs. A quantum circuit is represented by horizontal lines, which represent the wires (as in classical circuits), each of which is traversed from left to right by a qubit. The gates are represented by rectangles, which occupy as many wires as the qubits on which it acts. At the end of the

circuit, a measurement converts the result into a binary sequence of classical bits. For the controlled- U gates, it is customary to denote with a dark dot the control qubit in the line along which that qubit flows. An example of a quantum circuit is the following:



which represents the application of the Hadamard gate to the first qubit, then a $CNOT$ operation taking the first qubit as control and the second qubit as target and the final symbol indicates the measurement of the resulting qubit evolution in the circuit. The subscripts in the input qubits states indicate the convention for which, in general, given a state $|q_1q_2\rangle$, the left qubit q_1 is the second qubit and it takes position at the top of the circuit and the right qubit q_2 is the first qubit placed at the bottom of the circuit. In our example, $|00\rangle$ is the input quantum state.

Using gates to construct quantum circuits, it is possible to define any quantum algorithm, and this concept resides in *Universal Quantum Computing*. The circuit structure allows computations to be performed on a small set of qubits sequentially, manipulating them by leveraging the quantum properties of superposition and interference. Thanks to these properties, it is possible to achieve results that surpass the capabilities of classical computers. The main issue, as we will see in the next Chapter, lies in the introduction of *decoherence* during the computation process, which could lead to unreliable results.

2.10 Entanglement

The phenomenon of *entanglement* represents one of the main differences between classical and quantum systems; it refers to the ability of two qubits to behave as if they were a single object, exhibiting correlated properties that cannot be explained by classical physics.

It was studied how a n -qubits system state is defined as the superposition of all 2^n possible states, representing it as the linear combination of them. In particular, if the qubits are entangled, the system's state *cannot* be represented as a tensor product of the individual qubit states.

For example, consider the two quantum states $|\psi\rangle$ and $|\phi\rangle$ which live in the space

spanned by the computational basis $\{|0\rangle, |1\rangle\}$. We can write the inner product of $|\psi\rangle$ and $|\phi\rangle$ as:

$$|\psi\phi\rangle = \alpha_1 |00\rangle + \alpha_2 |01\rangle + \alpha_3 |10\rangle + \alpha_4 |11\rangle, \quad (2.64)$$

with $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \in \mathbb{C}$ such that $|\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 + |\alpha_4|^2 = 1$. In this case, it was possible to write the product of the two states as a linear combination of all 2^2 possible states and for this reason it is not an entangled state.

Considering instead the four *Bell states*:

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}, \frac{|00\rangle - |11\rangle}{\sqrt{2}}, \frac{|01\rangle + |10\rangle}{\sqrt{2}}, \frac{|01\rangle - |10\rangle}{\sqrt{2}}, \quad (2.65)$$

and try to represent the first of them as the inner product of the state $|\psi\rangle = \alpha_1 |0\rangle \alpha_2 |1\rangle$ with the state $|\phi\rangle = \beta_1 |0\rangle \beta_2 |1\rangle$, thus:

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}} = \alpha_1 \beta_1 |00\rangle + \alpha_1 \beta_2 |01\rangle + \alpha_2 \beta_1 |10\rangle + \alpha_2 \beta_2 |11\rangle. \quad (2.66)$$

Rewriting the right expression of (2.66) as:

$$(\alpha_1 |0\rangle + \alpha_2 |1\rangle) \otimes (\beta_1 |0\rangle + \beta_2 |1\rangle), \quad (2.67)$$

it can be verified that there do not exist $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathbb{C}$ which satisfy the equality with the first Bell state. In the same way, it can be shown that none of the Bell states (2.65) can be represented as a tensor product of individual qubit states, thereby forming an orthonormal basis for the space of all states defined by two entangled qubits.

The real advantage of two entangled qubits comes with the measurement of the system state. Consider again the first Bell state in (2.65) and investigate what happens to the state of the second qubit once the state of the first has been measured. To measure the rightmost qubit in the state $|0\rangle$, applying the projection matrix

$$\mathbb{1} \otimes |0\rangle \langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad (2.68)$$

to the state $|\psi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$, thus:

$$(\mathbb{1} \otimes |0\rangle \langle 0|) |\psi\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}} |00\rangle. \quad (2.69)$$

As a result, the system is in the state $|00\rangle$ with probability $|\frac{1}{\sqrt{2}}|^2 = \frac{1}{2}$. In particular, the rightmost qubit can be treated deterministically, as its state collapses into the pure state $|0\rangle$. From this information, given the entanglement, it can be verified that also the second qubit has collapsed into a pure state. To achieve this, make second projection of the leftmost qubit from $\frac{1}{\sqrt{2}}|00\rangle$ in the state $|0\rangle$:

$$\begin{aligned} (|0\rangle\langle 0| \otimes \mathbb{1}) \frac{1}{\sqrt{2}} |00\rangle &= \frac{1}{\sqrt{2}} \left(\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right) \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} |00\rangle \end{aligned} \quad (2.70)$$

obtaining the same result of the rightmost qubit projection in the state $|0\rangle$ in (2.69). This computation shows a significant property of measuring two entangled qubits, wherein upon measuring the state of one qubit, the information about the collapsed state of the other qubit are promptly acquired. Consequently, they can be regarded as a unified entity. Referring to the Bell state $|\psi\rangle$ defined previously, the system can be in the state $|00\rangle$ with probability $\frac{1}{2}$ and in the state $|11\rangle$ with the same probability. After a measurement, if one of the two qubits is in the state $|0\rangle$, then the second one is *deterministically* in the state $|0\rangle$; if one of the two qubits is in the state $|1\rangle$, then the other one is *deterministically* in the state $|1\rangle$. This property holds for all the other Bell states listed in (2.65) as well as for more complex entangled qubits systems.

2.11 State Evolution of Closed Quantum Systems

A quantum system is totally described by its state and its evolution over time. Therefore, if we denote with $|\psi(t_1)\rangle$ and $|\psi(t_2)\rangle$ the state of the considered quantum system at time t_1 and t_2 , it holds that:

$$|\psi(t_2)\rangle = U(t_1, t_2) |\psi(t_1)\rangle \quad (2.71)$$

where U is a *unitary operator* which depends only on the time t_1 and t_2 , with the properties defined in Section 2.5.

In physics, it is stated that a closed quantum mechanical system evolves over time

according to the *Schrödinger equation*

$$i\hbar \frac{\partial |\psi(t)\rangle}{\partial t} = H(t) |\psi(t)\rangle \quad (2.72)$$

where $i = \sqrt{-1}$, \hbar is the *Planck's constant* and $H(t)$ is the *Hamiltonian* operator. The solution of the Schrödinger equation (2.72) for two fixed time instants t_1 and t_2 , not considering the Planck's constant, is the following:

$$|\psi(t_2)\rangle = e^{-iH(t_2-t_1)} |\psi(t_1)\rangle \quad (2.73)$$

whose form is analogous to the equation (2.71). Indeed, the Hamiltonian H is a Hermitian operator such that $e^{-iH(t_2-t_1)}$ is a unitary operator, with which it is immediate to state that the *Evolution Postulate* stated in Section 2.8 is a direct consequence of the Schrödinger equation (2.72).

The Hamiltonian provides the energy value of the associated quantum system and the study of its eigenvectors and eigenvalues, known as *eigenstates*, provides important results exploitable in Quantum Computing. In Section 2.4 it is observed how the Spectral Decomposition Theorem holds for Hermitian operators. Therefore, if the Hamiltonian is independent of time, the time-independent Schrödinger is given by the following equation:

$$E |\psi\rangle = H |\psi\rangle \quad (2.74)$$

where E is a (real) eigenvalue of H , denoting the energy of the system. Solving (2.74) means find the eigenvectors of the Hamiltonian H , also called *eigenstates*, which represent the stationary states for the system. Therefore, if we find an eigenstate ψ_α of H , the time evolution operator acts in this way:

$$H\psi_\alpha = E_\alpha\psi_\alpha \rightarrow e^{-iHt} |\psi_\alpha(0)\rangle = e^{iE_\alpha t} |\psi_\alpha(0)\rangle \quad (2.75)$$

providing the solution to the time-dependent Schrödinger equation (2.72):

$$|\psi(t)\rangle = \sum_{\alpha} c_{\alpha} e^{-iE_{\alpha} t} |\psi_{\alpha}(0)\rangle \quad (2.76)$$

where $\mathbf{U}_E(t) = e^{-iE_{\alpha} t}$ is a unitary operator which preserves the l_2 -norm of the state, i.e. the normalization constraint of the amplitude probabilities holds at every time step and, therefore, the probabilities of the eigenstates doesn't change [7]. This property has an important consequence in Quantum Computing, since the study of the quantum system time evolution can be made through the operator $\mathbf{U}_E(t)$ which, as unitary operator, can reproduce all the fundamental gates. As a result, $\mathbf{U}(t) = e^{-iHt}$ can be reproduced as a combination of elementary gates.

The energy study of a quantum system state can be useful not only for gate model Quantum Computing. In fact, the state of a quantum system with minimum energy, called *ground state*, can represent the solution to many optimization problems. This property is particularly exploited by the *Adiabatic Quantum Computing*, analyzed in 3.4, which formulates the objective function of a generic optimization problem as the Hamiltonian of a quantum system, in order to find the solution by searching for the minimum energy value.

2.12 Information Encoding

In order to exploit the notions of Quantum Computing explained in the previous Sections, it is necessary to adopt techniques that allow real data to be encoded in qubits. In this section, a briefly look at some of these techniques will be explored, detailed explained in [9].

Basis Encoding

This encoding represents the most direct technique for quantum computation because it represents real numbers as binary numbers and then transforms them into a quantum state in the computational basis [10]. Specifically, any number $x \in \mathbb{R}$ is assumed to be approximable using k decimal places and translatable into binary form by adding an extra sign bit. Therefore, we can write x as:

$$x \approx \sum_{i=0}^n b_i 2^i + \sum_{i=1}^k b_{-i} 2^{-i}, \quad (2.77)$$

with $b_i, b_{-i} \in \{0, 1\}$. By this formulation, the real number is prepared as a quantum state of $n + k - 2$ qubits.

Consider a vector $\vec{v} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$, its basis encoding is computed concatenating the approximation of each component as in (2.77). Therefore, if $b_s^i b_n^i \dots b_0^i b_{-1}^i \dots b_{-k}^i$ is the binary approximation of the element x^i , we have a concatenation vector for \vec{v} of the form $(b_s^1 b_n^1, \dots, b_0^1 b_{-1}^1 \dots b_{-k}^1 \dots b_s^n b_n^n \dots b_0^n b_{-1}^n \dots b_{-k}^n) \in \{0, 1\}$ which can be used to prepare its quantum state in as follow:

$$|b_s^1 b_n^1 \dots b_0^1 b_{-1}^1 \dots b_{-k}^1 \dots b_s^n b_n^n \dots b_0^n b_{-1}^n \dots b_{-k}^n\rangle. \quad (2.78)$$

Once the binary approximation of a real number is computed, the gate X^{b_i} is applied to the qubit i to define the corresponding circuit which create the quantum state representing the real number x .

Given a dataset $\mathcal{D} = \{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_n\} \subseteq \mathcal{R}^n$, it can be represented by the uniform

superposition of the binary encoded states of its elements, i.e.

$$|D\rangle = \frac{1}{\sqrt{m}} \sum_{i=1}^m |\vec{v}_i\rangle. \quad (2.79)$$

For example, consider $\vec{v}_1 = (01, 10)^T$ and $\vec{v}_2 = (11, 00)^T$ with basis states representation, respectively, $\vec{v}_1 = |0110\rangle$ and $\vec{v}_2 = |1100\rangle$. Following the definition in (2.79), the superposition has the following expression:

$$\mathcal{D} = \frac{1}{\sqrt{2}} |0110\rangle + \frac{1}{\sqrt{2}} |1100\rangle = \frac{1}{\sqrt{2}} |6\rangle + \frac{1}{\sqrt{2}} |12\rangle. \quad (2.80)$$

In this case, the vector:

$$(0, 0, 0, 0, 0, 0, \frac{1}{\sqrt{2}}, 0, 0, 0, 0, 0, \frac{1}{\sqrt{2}}, 0, 0, 0)^T, \quad (2.81)$$

represent the amplitude vector in computational basis of \mathcal{D} in (2.80).

Amplitude Encoding

This method of information encoding associates classical information with quantum amplitudes. To achieve this, several methods are employed, all aiming to find the correct combination of probability amplitudes that maps classical information, such as a vector of real numbers, into an object in a Hilbert space.

In general, the amplitude encoding of a N -dimensional data input point $\mathbf{x} = (x_1, x_2, \dots, x_N) \in \mathbb{C}^{2^n}$ is the quantum state defined as follow:

$$|\mathbf{x}\rangle = \sum_{i=1}^N \frac{x_i}{\|\mathbf{x}\|} |i\rangle. \quad (2.82)$$

In this way, the quantum amplitudes are associated to each element of the vector \mathbf{x} .

However, this approach has significant limitations: the inability to implement a nonlinear map on the amplitudes in a unitary fashion and the requirement that only normalized classical vectors can be processed.

Hamiltonian Encoding

This method concerns the encoding of matrices through the association of Hamiltonians, inspired by the Schrödinger equation (2.72) which characterizes the evolution of a quantum system.

This encoding starts with a square matrix $A \in \mathbb{C}^{2^n \times 2^n}$ which represents the initial data and ends with an Hamiltonian associated matrix H_A to represent the Schrödinger equation in this way:

$$|\psi(t)\rangle = e^{iH_A t} |\psi(0)\rangle. \quad (2.83)$$

where $|\psi(0)\rangle$ is the initial state and $|\psi(t)\rangle$ is the quantum state that encodes the Hamiltonian information.

If the starting matrix A is Hermitian, we can set $H_A := A$. If the last condition is not true, we can define H_A with the augmented matrix:

$$H_A := \begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix}. \quad (2.84)$$

The implementation of the evolution (2.83) is named *Hamiltonian simulation* and it is the core of the Quantum Adiabatic Computing approach, explored in Section 3.4.1. In particular, the goal of the Hamiltonian simulation is to find a state $|\psi_{sim}\rangle$ such that:

$$\| |\psi_{sim}\rangle - |\psi(t)\rangle \| \leq \epsilon \quad (2.85)$$

where ϵ is a given precision, $\| \cdot \|$ is a suitable norm and $|\psi(t)\rangle$ is the solution of the Schrödinger equation (2.83).

2.13 Conclusions

In this Chapter, the fundamentals of Quantum Computing have been illustrated, including the definition of qubits and quantum states, with a focus on the Bloch sphere; the definition of computational bases and the relevant Hilbert spaces along with their properties. Unitary operators and their properties have been introduced, which are useful in the definition of quantum gates and the measurement phase. In this regard, the main unitary operators, such as Pauli, Hadamard, and Phase operators, have been defined. Concerning multiple systems, the tensor product has been introduced and analyzed, focusing on how to recognize an entangled state using this object. The notion of pure and mixed states has also been studied, along with the definition of the density matrix, highlighting the No-Cloning Theorem. A particular focus has been given to the procedure of measuring a quantum system, providing the postulate of evolution, which involves properties related to unitary operators, and the postulate of measurement, defining the Von Neumann measurement and a study on how the spectral decomposition of a Hermitian operator can be exploited for measurements. The concept of Universal Quantum Computing has been analyzed with the definitions of the main gates, such as CNOT, SWAP,

and \sqrt{SWAP} . The phenomenon of entanglement has been studied in detail, emphasizing how it can be utilized for quantum computation. Finally, we focused on the time evolution of a closed quantum system, analyzing the Schrödinger equation and its solution, also mentioning Adiabatic Quantum Computing, an approach to quantum computation different from Universal Quantum Computing that directly exploits the time evolution of the Hamiltonian associated with the system.

Chapter 3

Quantum Hardware and Computation in the NISQ Era

The term “NISQ,” coined in 2018 by the American theoretical physicist John Preskill, stands for “Noisy Intermediate-Scale Quantum” and describes the current state of quantum hardware. Nowadays, although the number of qubits available on quantum hardware from leading companies is growing every year, many challenges still need to be addressed, such as error tolerance and the introduction of decoherence during computation, concepts that will be discussed further.

In this Chapter, we will explore various aspects of quantum hardware, outlining different technologies. We will also examine key techniques for encoding real-world data into qubits before focusing on Analog Hamiltonian Simulation, therefore the Adiabatic Quantum Computing.

3.1 The Five Quantum Hardware Criteria

In the NISQ era, a Quantum Computing technology should embody as much as possible all the five criteria listed by the American theoretical physicist David Di Vincenzo in [11], which are the following:

1. *A scalable physical system with well characterized qubits.*

The addition of one or more qubits to a quantum system should not compromise computational efficiency. By “well characterized,” we mean that its physical parameters should be well defined, including its internal Hamiltonian (which determines the energy eigenstates); the presence and coupling to other states; interactions with other qubits; and the coupling to external fields that should be exploited for the manipulation of its state.

2. *The ability to initialize the state of the qubits to a simple fiducial state, such*

as $|00 \dots 0\rangle$.

Before starting a quantum calculation, it is usual to initialize the qubits with a certain value, in order to know their initial state. Furthermore, quantum error correction frequently requires the presence of qubits in a low-entropy state, such as $|0\rangle$. In this regard, it is important to take into consideration the speed with which a qubit can be reset. There are two main approaches for setting the state of a qubit: “cooling” the system when the ground state of its Hamiltonian is in the state of interest or via projection of the system into the desired state.

3. *Long relevant decoherence times, much longer than the gate operation time.*

When a quantum system interact with the external environment, decoherence arises within the system. The “decoherence time”, as will be observed in 3.2, is the time with which a generic state $|\psi\rangle$ is transformed into a mixed state. Therefore, the longer the decoherence time, the better the performance of the quantum hardware.

4. *A “universal” set of quantum gates.*

Typically, a quantum algorithm is specified by a sequence of unitary operators U_1, U_2, \dots, U_n each acting on a small number of qubits. All these can be translated in physical terms, identifying Hamiltonians which generates the unitary transformation $U_1 = e^{iH_1t/\hbar}$, $U_2 = e^{iH_2t/\hbar}, \dots, U_n = e^{iH_nt/\hbar}$. It is important to underline that the implementation of quantum gates can introduce decoherence into the system, due to the presence of some systematic or random errors in the Hamiltonians definitions.

5. *A qubit-specific measurement capability.*

When measuring a quantum system, the actual outcomes may differ from the ideal ones. In fact, in an ideal context, as will be seen in 3.2 we refer to density matrices with nonzero off-diagonal terms while in a real context it can be not true due to the decoherence. Therefore, if in an ideal case we have the 100% quantum efficiency, in the real case we have a certainly smaller percentage.

3.2 The Principal Limitations of the Quantum Hardware

In Quantum Computing theory, it is stated that if $|\psi_1\rangle$ and $|\psi_2\rangle$ are two states of a quantum system, then $\alpha_1|\psi_1\rangle + \alpha_2|\psi_2\rangle$ represents the superposition of the two states, with $\alpha_1, \alpha_2 \in \mathbb{C}$ and $|\alpha_1|^2 + |\alpha_2|^2 = 1$. However, this superposition concept

is very difficult and sometimes prohibitive to construct [12].

In Section 2.7, the representation of a generic quantum system state as a density matrix is explored. Remember that the diagonal entries of the density matrix represent the probabilities of finding the system in the corresponding states, while the off-diagonal entries indicate the coherence between them. If two quantum states exhibit coherence, it means that their quantum attributes, such as relative phase and magnitude, are interconnected and capable of meaningful interaction. This coherence represents a unique aspect of the wave-like behavior of matter and serves as the cornerstone for Quantum Computing and technology. However, the difficulty of quantum computers in recreating closed systems is precisely reflected in these terms, and for this reason, the concept of *decoherence* is introduced. In particular, when a quantum hardware performs computations, it cannot guarantee that the system remains isolated from the external environment. This behavior leads to increasingly intense interactions with the environment, resulting in the decrease of coherence and the progressive vanishing of the off-diagonal terms, ultimately transforming the initial quantum state into a mixed state. Therefore, if theoretically the off-diagonal terms are nonzero, in practice, with the introduction of decoherence, the density matrix takes the following form:

$$\rho_{dec} = \begin{pmatrix} |\alpha_1|^2 & 0 \\ 0 & |\alpha_2|^2 \end{pmatrix}.$$

In this way, even if a quantum state is prepared in a correct way, it could quickly become *unobservable*. The decoherence, which gave birth to the *theory of quantum decoherence*, introduced by H.D Zeh in [13], is one of the main problem for the Quantum Computing because once the coherence of the system dissipates from its density matrix, we lose the ability to perform computations. Consequently, all operations must be completed *before* to the *decoherence time*, i.e. the time limit before decoherence enters in the system.

The actual limit of the current quantum computers lies in their inability to perform long computations without incurring some error, albeit small. The question is, what is the maximum number of gates that can be applied before noise is introduced into the system. There is a very important result that demonstrates that, if the error required to execute each gate remains below a certain constant threshold, then this limit in the number of gates does not exist. This result is stated more formally in the following theorem:

Threshold Theorem: a quantum circuit on n qubits and containing $p(n)$ gates may be simulated with probability of error at most ϵ using, for some constant c ,

$$O(\log^c(p(n)/\epsilon)p(n))$$

gates on hardware whose components fail with probability at most p , provided p is below some constant threshold, $p < p_{th}$ and given reasonable assumptions about the noise in the underlying hardware.

Note that this theorem is the equivalent Von Neumann's threshold theorem for classical computing, with quantum circuits instead of classical ones.

Another major challenge for quantum hardware concerns *scalability*. This means that, as the number of qubits in the hardware increases, the level of decoherence and the number of computational errors should not increase. Nowadays, the leading Quantum Computing companies aim to build completely *fault-tolerant* hardware, capable of obtaining reliable results even in the presence of errors during the quantum computation. All this is made possible thanks to some error correction algorithms that primarily rely on using groups of multiple *physical* qubits to define single *logical* qubits, which can be used as quantum computational units. In this way, quantum information is encoded in multiple physical qubits, and if some of them are affected by errors, the quantum information is not lost. However, this technique need a large number of qubits, which clashes with the limited capacity of quantum hardware. At the time of writing, IBM's transmon-based quantum processor has the goal of reaching 100,000 qubits by 2033 [14].

3.3 Hardware Outlook

The key component of a quantum hardware is represented by the *Quantum Process Unit* (QPU), a special processor which allows the computation exploiting the quantum properties, such as entanglement and superposition.

As mentioned previously, the decoherence represents the central challenge in building a quantum device. Quantum hardware must be designed to preserve quantum information while also providing controllability of the system from external sources. Therefore, there is a trade-off between coherence in a sense of immutability of information and decoherence, i.e. the ability to control the system. In this section, we will broadly explore the main technologies currently employed in the construction of quantum hardware.

3.3.1 Superconducting Qubits

Superconducting qubits have been known as the leading technology for building scalable quantum processor architectures [15]. Nowadays, quantum computers based on superconducting qubits are developing very rapidly and with them the number of available qubits is also growing, along with their quality. This allows the achievement of high fidelity construction of two-qubit gates in 2020 (99.5%), as reported in [16].

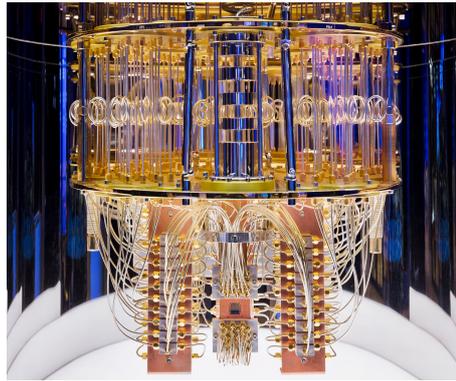


Figure 3.1: IBM's quantum computer.

Specifically, superconducting qubits are solid state electrical circuits with some advantages:

- *High designability.* It means that it is easy to define different types of qubits and to adjust the parameters needed.
- *Scalability.* Utilizing state-of-the-art chip manufacturing technologies enables the production of high-quality devices, promoting efficiency in manufacturing and scalability.
- *Easy to couple.* This is an intrinsic property of the superconducting qubits.
- *Easy to control.* The operations and measurements of superconducting qubit circuits can be performed using simple, readily available microwave devices.

However, there is an important disadvantage, that is the short coherence times. Additionally, they necessitate dilution refrigerators to maintain sufficiently low temperatures. Significant advancements in the capacity of such cryostats are essential before the construction of a device with millions of qubits can proceed. For this reason, the building of quantum superconducting hardware with high number of qubits is still a challenge. The Figure 3.1 shows the IBM's superconducting qubits quantum computer.

3.3.2 Trapped Ion qubits

This technology, in general, is based on the arrangement of ions in a lattice by the means of optical lasers. In particular, neutral-atom quantum technology belongs to this category, identifying ions as neutral atoms. By the term “neutral” we refer to atoms that lack an electric charge, thus possessing an equal number of electrons and protons. QuEra, a pioneering company in the field of neutral atom

technology, in [17] explains how, to manipulate the atom, a cooling phase by laser beams carry the particles at microkelvin temperatures. In particular, QuEra uses the neutral Rubidium atom (Rb-87), which intrinsically encodes a qubit in its electronic state. Consequently, Quantum Computing and quantum information processing is enabled by the manipulation of these atoms by laser pulses. The neutral atom hardware, called Aquila, is shown in Figure 3.2 and will be described in more detail in Chapter 4.

The main advantages of a Rydberg atom technology are the following:

- *Perfect nature of qubits*: they are identical to each other and are simultaneously capable of storing and processing quantum information.
- *Error resilience*: Rydberg atoms interact on demand. When atoms are not excited, they are robust to errors regardless the total number of qubits in the system.
- *Reconfigurable layouts*: the lasers are free to move in the space and the atoms can be arranged in (almost) any possible geometry.
- *High scalability*: the compact dimensions and the efficient control mechanisms allow the number of qubits to be significantly increased without the need of interconnections.

The disadvantages of the neutral atom technology concern the experimental complexity of working with them, as it is necessary to have lasers working with extreme precision, together with a good cooling system and vacuum cells to guarantee isolation.

3.3.3 Photonic Qubits

In this *optics* technologies the qubits are represented by photons, which are the smallest units of the light. The main advantage of the photon is their long coherence time, which allow high-fidelity calculations and for this reason, they represent a very promising avenue for the realization of quantum technologies. All quantum applications need a source of *single photons* and it is not so easy to obtain [18]. For this reasons many technologies tend to generate a sort of single-photon approximation. In recent years, significant advancements have occurred in enhancing the characteristics of authentic single-photon emitters. However, their creation remains challenging and they lack substantial spectral flexibility. For example, the French company *Quandela* has recently been focusing on exploring the use of non-classical light quantum states for Quantum Computing and quantum communication [19]. They can be produced from classical pulses through nonlinear

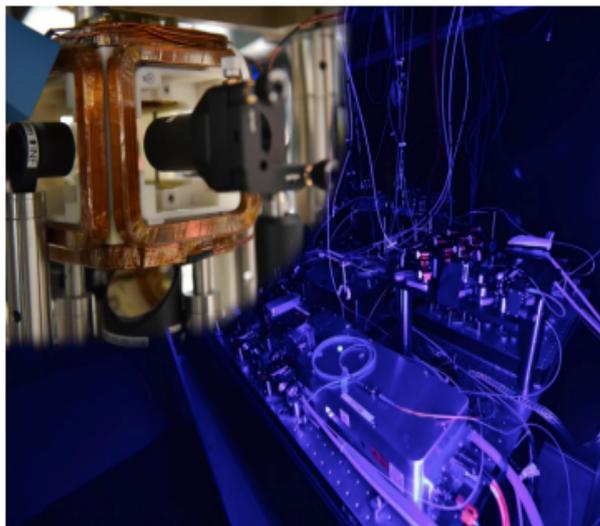


Figure 3.2: inside Aquila there is a set of optical elements, lasers and cameras (bottom right) that focus on a vacuum cell (inserted top left) filled with a dilute gas of Rubidium atoms. In an area less than three human hairs wide, laser fields are carefully controlled to manipulate the state of up to 256 qubits to perform quantum calculations. The image was taken from [17].

processes such as parametric down-conversion or emission from individual quantum emitters.

Finally, it is important to note that the photons can be produced in normal environment condition, facilitating the hardware construction.

3.4 The Analog Hamiltonian Simulation

Regardless of the technology used by the hardware, there are two paradigms through which devices can approach Quantum Computing, i.e. *analog* and *digital*. The substantial difference between the two methods lies in the way the calculation is performed. While digital computing devices encode the problem in a sequence of gates, each of which acts on one or two qubits, an analog Hamiltonian simulator exploits a well-controlled system that allows defining time-dependent parameters with which the Hamiltonian describes the dynamics of the quantum system studied. In other words, in digital computation, the problem is discretized into smaller parts, while in analog simulation, the system's dynamics are controlled through parameters that are continuous variables over time, allowing a specific physical law to be simulated. It is important to underline that the gate model is a *universal* quantum computation model, as it allows any arbitrary operation to be

approximated to any desired precision using a finite set of quantum gates. On the other hand, analog simulation, while being a very powerful Quantum Computing model for solving specific problems, does not necessarily possess the aforementioned property of “universality,” limiting its range of applications.

The gate model was discussed in Section 2.9, while the analog simulation will be explored in the next Section.

Of particular interest in this work is an analog quantum approach that closely relates to Hamiltonian information encoding, called *Adiabatic Quantum Computing*. The hardware that utilizes this approach is referred to as *quantum annealers*, and specifically, this work will apply two such technologies. The first is D-Wave, which uses superconducting qubits, and the second is QuEra, which utilizes neutral atoms.

3.4.1 Adiabatic Quantum Computing

The Adiabatic Quantum Computing is a quantum approach based on the *Adiabatic Theorem*, which states that *if a system is initially prepared in the ground state $|g(0)\rangle$ of a time-dependent Hamiltonian $H(t)$, the time evolution described by the Schrödinger equation (2.72) will approximately keep the system’s state at time t in the corresponding ground state $|g(t)\rangle$ of $H(t)$, provided that the evolution is performed sufficiently slowly [20].*

The Hamiltonian of the system can be decomposed into two parts:

$$H(t) = H_{\text{initial}} + H_{\text{target}}. \quad (3.1)$$

Defining $s(t) : [0, t_f] \rightarrow [0, 1]$ as the *schedule function*, it holds that:

$$H(s) = (1 - s)H_{\text{initial}} + sH_{\text{target}}, \quad (3.2)$$

with the time t_f such that the final state represented by $H(s)$ is ϵ close in l_2 -norm to the ground state of H_{target} . This means that, at the initial instant, the system is described by the Hamiltonian H_{initial} , whose ground state is easily prepared. Following a sufficiently slow evolution, at the final instant t_f , the Hamiltonian of the system is H_{target} , which represents the solution to the problem of interest. Consequently, H_{target} must be defined in such a way as to encode the problem to be solved.

Since the optimal solution of a combinatorial optimization problem is a combination that minimizes the objective function, quantum annealing exploits quantum physics to find a combination of elements such as to obtain states with the lowest possible energy. For this reason, from a physical point of view, we try to solve an optimization problem which can be translated into an energy minimization problem.

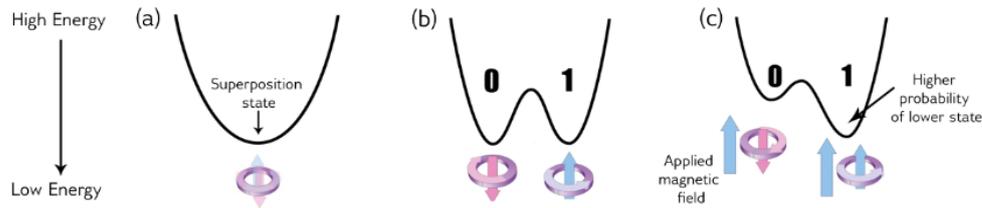


Figure 3.3: Quantum annealing energy steps of a qubit. The process begins with a qubit in a superposition state, therefore with a single energy valley (a), subsequently a barrier is raised as the quantum annealing process runs and we have the so-called *double-well potential* (b) where there is a 50% probability that the qubit collapses into one of the two states. We can control this probability by applying an external magnetic field (c). The image was taken from the D-wave documentation available in [21].

Adiabatic Quantum Computing in D-Wave

In the QPU of D-Wave, the computing units are *superconducting* qubits whose states are implemented as circulating currents, with corresponding magnetic fields. The direction of the central magnetic field follows the right-hand rule, so if the current circulates counterclockwise, the magnetic field will point upwards, while if the current circulates clockwise, the magnetic field will point downwards. A qubit whose magnetic field points downward is in the state 0, otherwise it is in the state 1. However, being a quantum object, it lives in the superposition of two states and at the end of the quantum annealing process, it will collapse into one of the two states, following the scheme of the Figure 3.3.

It is important to note that we can control the probability with which a qubit collapses into one state rather than the other through an external magnetic field. The power of this computation emerges when the phenomenon of entanglement is achieved by linking multiple qubits using a mechanism called *coupler*. The biases and couplings define an energy landscape and the goal is to find the landscape with the lowest energy level. Consequently, at the end of quantum annealing, the qubits are in a classical state which minimizes the energy of the problem.

The simulated annealing algorithm, treated in more detail in Appendix A, is the classical counterpart of the quantum annealing which exploits a particular thermodynamic phenomenon in metallurgy to map solutions of an optimization problem to atomic configurations [7]. Quantum annealing doesn't care about the thermodynamic phenomenon and exploits a mechanism called *tunneling* which doesn't have any classical counterpart. The term "tunneling" stems from the idea of a tunnel within the energy barrier that facilitates the passage of a quantum object, as reported in Figure 3.4, enabling the exploration of configurations that would not

be accessible using classical heuristics. In this way, the algorithm avoids getting trapped in local minima, increasing the probability of finding the global optimum. It is important to note that the higher the energy barrier, the lower the probability of quantum particles tunneling through it.

The strong limitation of the quantum annealing is the concept that it is not a universal Quantum Computing technique and mathematically the superconducting-qubits systems only support the *Ising formulation* of the Hamiltonian, whose variables can assume values in the set $\{-1, 1\}$. We will study this formulation in more detail in Section 5.3.2.

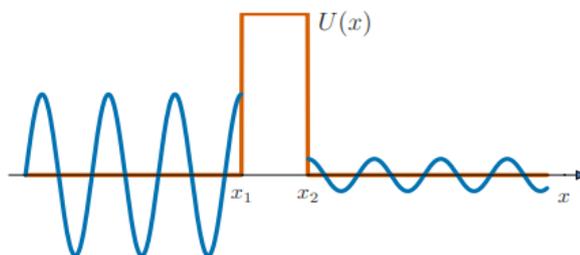


Figure 3.4: Quantum particle tunneling. We can see a spatial range $x_1 < x < x_2$ with energy $U(x)$ greater than the blue energy waveform of the particle. After the tunneling, the particle has the same energy but less amplitude. The image is reproduced from [7].

Adiabatic Quantum Computing in QuEra

The adiabatic Quantum Computing is exploited by the QuEra's quantum hardware (*Aquila*) QPU. In this technology, a qubit is represented by a neutral Rubidium atom (Rb-87), which is arranged in a lattice thanks to ultra-precise laser tweezers. The lattice in which the atoms can be arranged is arbitrary and they are distant from each other by an amount of the order of a nanometer (μm). Through energetic manipulations, a neutral atom can assume multiple states and QuEra specifically harnessing three of them: the *ground state* $|g\rangle$, in which the atom minimizes its energy; the *Rydberg state* $|r\rangle$, reached through the excitation of the atom which then reaches a high level of energy and the *hyperfine state* $|h\rangle$. To implement quantum dynamics, i.e. the passage of a qubit from one state to another, Aquila uses lasers with different wavelengths, as shown in the Figure 3.5.

Since the transition from the ground state $|g\rangle$ to the Rydberg state $|r\rangle$ is difficult to generate with a single laser, it is performed in two photonic transitions with λ wavelengths of 420 nm and 1013 nm . The manipulation of transitions between various states that allows quantum calculations is determined by time-dependent

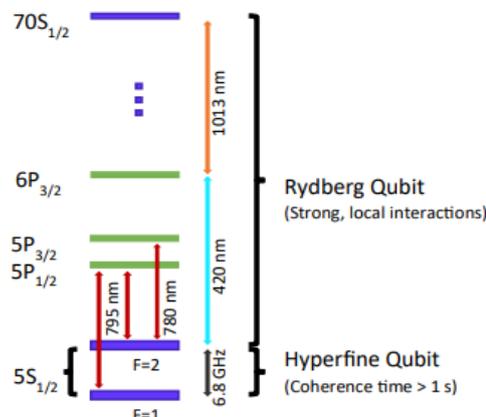


Figure 3.5: The arrows represent the various optical fields used for the transitions on the various states. The lines represent the various states of the qubits. Specifically, purple lines are the qubit states and green lines are intermediate states exploited for some manipulations.

parameters that precisely control the optical fields. These parameters are the following:

- $\Omega(t)$: the *Rabi drive amplitude*. This parameter sets the frequency with which each atom transitions from the ground state to the Rydberg state, assuming no other interactions. This parameter is a function of the laser amplitude.
- $\Delta(t)$: the *detuning* parameter represents the laser resonance offset and determines how close the laser is to the atomic transition.
- $\Phi(t)$: the *phase* of the laser determines the direction in which each qubit localizes within the lattice.

The analog calculation mode finds itself precisely in the possibility of controlling these parameters which continuously depend on time. In fact, instead of indicating a gate sequence as in digital calculation, the user can define the time series of the Rabi drive amplitude $\Omega(t)$, of the detuning $\Delta(t)$ and of the laser phase $\Psi(t)$. Along with these parameters, the user can further set the positions of the atoms in the lattice and control all of them during the time evolution. The neutral atom technology is analyzed in more detail in Chapter 4.

We conclude this section by remembering that the gate-model is considered a *universal* model for Quantum Computing, while analog computing is not. This

is due to the fact that any quantum algorithm is able to be implemented with a certain sequence of quantum gates, while the techniques adopted for analog computing are not able to simulate any function or algorithm. This does not mean that the gate-model is better than analog calculation but only that the latter is better suited to specific tasks, while the former is a universal approach.

3.5 Conclusions

In this Chapter, the term NISQ was initially defined along with Di Vincenzo's five criteria pertaining to quantum hardware construction. In this regard, the main error resources were presented, which still pose challenges in achieving reliable quantum hardware. Subsequently, various types of qubits utilized by quantum technologies were briefly outlined, including superconducting qubits, ions, and photons. A particular focus was placed on distinguishing between Universal Quantum Computing and Analog Hamiltonian Simulation, with an emphasis on Adiabatic Quantum Computing and the statement of the Adiabatic Evolution Theorem outlining its properties. Additionally, two quantum annealers, D-Wave and QuEra, were further explored, whose QPUs utilize Adiabatic Quantum Computing for solving combinatorial optimization problems.

Chapter 4

The Neutral-Atom Quantum Technology

In Chapter 3, the main features of the QuEra’s neutral atom technology, called *Aquila*, were briefly explained. In this Chapter, it will be studied how to exploit neutral atoms for Quantum Computing, with a focus on the methodologies adopted by QuEra. The field of research is rapidly advancing in applying this approach to both quantum circuits (digital computing) and the development of many-body Hamiltonians over time (analog simulation). Although this Chapter also discusses how quantum gates can be defined using neutral atom technology, at the time of writing, *Aquila* doesn’t support the possibility to implement quantum gates and therefore we will refer to its QPU only from the analog simulation side.

4.1 The Neutral and Rydberg atoms

First of all, it is important to give a definition of neutral and Rydberg atoms. A neutral atom is an atom with zero electric charge, i.e. the number of electrons equals the number of protons. The Rydberg atom, named for the Swedish scientist Johannes Rydberg, is in a highly excited state with one or more electrons that have been moved to an orbital away from the nucleus. Therefore, a neutral atom can be manipulated into a Rydberg state by a process called *excitation*. This procedure can be induced with various techniques, including the use of lasers which generate specific wavelength, as seen in Section 3.4.

A technology based on Rydberg atoms exploits the highly energetic states that can be created by exciting the neutral atoms. Usually, alkaline atoms (such as Lithium, Potassium or Rubidium) are considered, which in nature have zero electrical charge. Therefore, the Rydberg atom and the neutral atom, although not the same thing, are concepts that get confused when referring to quantum technology

that exploits their energy states.

4.2 Rydberg Platform

The first studies concerning trapped neutral atoms as Quantum Computing units date back to the years 2000 [22] and 2001 [23] and are still a subject of great interest. The idea behind these studies is to encode quantum information in the internal states of single atoms or in the collective excitations of sets of atoms, which interactions are mediated by electronically highly excited Rydberg states [24]. The platforms that exploit this technology aims to respect both the criteria for quantum computers by Di Vincenzo [11] listed in Chapter 3, and the criteria for quantum simulators listed by Cirac and Zoller in [25], making both the definition of gate-models and analog simulation possible. These criteria are summarized below:

1. *Quantum system*: the atoms are manipulated thanks to a cooling phase and then trapped in optical microtraps. The system is composed of atoms arranged in a lattice, each of which possesses discrete quantum states, exploited to encode qubits and interactions between them, thus guaranteeing a certain number of degrees of freedom. The neutral particles can be isolated almost completely from their environment, promoting system scalability.
2. *Inizialization*: the atoms that define the qubits are all identical to each other and have well-defined energy levels, so they can easily be initialized to a known state through techniques involving dissipative optical pumping.
3. *Coherence*: the coherence time varies from a few milliseconds to a hundred milliseconds and depends on the type of atom and the type of states involved.
4. *Interactions*: atoms can be easily manipulated using laser tweezers, which can also induce interactions between them. The interactions that are created are very particular and can define useful states both for defining multi-qubit gates and in the field of analog simulation.
5. *Measurement*: the most widely used technique to measure a quantum system composed of Rydberg atoms is via sensitive single-atom fluorescence imaging from the fundamental states. Using special techniques, it was demonstrated that, although a measurement is destructive for a quantum system, it is possible to have a high-fidelity interpretation.
6. *Verification*: for a system of a few qubits, verifying the correctness of the result obtained can be carried out using a technique called tomography. For larger systems, it is necessary to compare the result with that obtained from

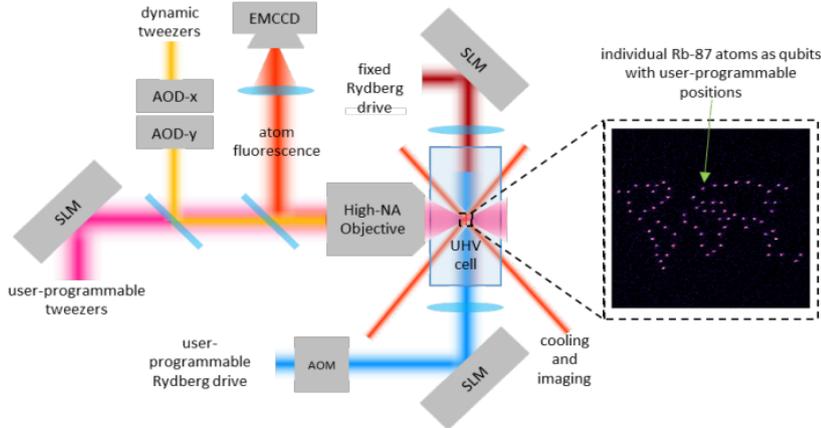


Figure 4.1: Diagram of how Aquila works. The **pink** laser is controlled by a spatial light modulator (SLM) and positions up to 256 atoms in a specified geometry; the **yellow** laser dynamically positions the atoms and contributes to their ordering; the set of **red** lasers brings the atoms to temperatures of μK ; the **dark red** and **dark blue** lasers contribute to the passage of the atoms from the fundamental states to the Rydberg ones and finally the **orange** laser and the camera provide an image of the atoms arrangement thanks to the fluorescence. The image is reproduced from [24].

a classic high-performance computer. Rydberg atom technology has high tunability and spatial reconfiguration of the system, thus allowing comparison with exact classical numerical methods.

4.2.1 QuEra’s Neutral Atom Hardware

QuEra’s hardware platform, Aquila, is a room-temperature quantum device that uses Rubidium-87 atoms, which are brought to microkelvin temperatures by laser beams inside a vacuum cell. The vacuum cell is necessary to isolate the atoms from the external environment and the low temperatures help the manipulation of the atoms. The central unit of Aquila is a 2 cm scale glass vacuum cell where atoms are arranged thanks to lasers into a 2D lattice in an area of width less than $200\mu\text{m}$ (the width of about three human hairs). The six different waveforms of laser light concentrated in the vacuum cell are represented in Figure 4.1, illustrating the functional block of Aquila.

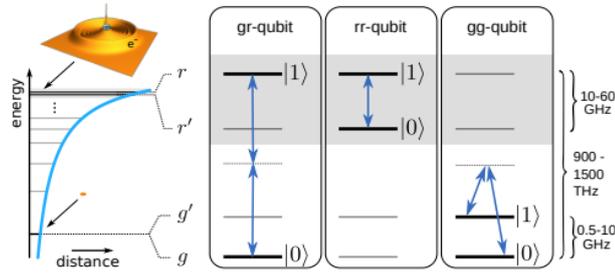


Figure 4.2: On the left, the energy function encoding the various possible states of the atom. We note that the three types of qubits are also distinguished by the level of energy required with which they can be encoded. These states can be modified through the use of optical fields. The image was taken from [24].

4.3 Neutral-atom Quantum Computing

This Section will delve into the main aspects of Quantum Computing based on the Rydberg atoms, focusing on the methodologies adopted by QuEra.

4.3.1 Rydberg States and Qubits

The Rydberg atoms have a well-defined energy level structure, which is exploited to coherently manipulate their internal states. We have seen that the neutral atom can mainly represent two possible states, namely the Ground state $|g\rangle$ and the Rydberg state $|r\rangle$. While the first is naturally encoded by localizing all the electrons in the orbital closest to the nucleus, the second requires an external excitation phase that allows one or more electrons to occupy orbitals at a relatively greater distance from the nucleus. For this reason, the ground state is a *stable* state while the Rydberg state is strongly *unstable* and the atom tends to minimize its energy.

To bring an atom into the Rydberg state, only one of its electrons needs to move to the outermost orbital. This electron is called *valence electron*. The possibility of coherently and easily manipulating these states also allows the interaction between valence and classical electrons in the ground state, storing quantum information and defining three classes of Rydberg qubits, with specific properties. These classes are distinguished by the number of ground states that compose a qubit and are illustrated in Figure 4.2.

- *Ground-Rydberg qubits (gr)*. These represent the simplest class of Rydberg qubits, composed of a valence electron in the strongly-interacting Rydberg state $|r\rangle \equiv |1\rangle$ and electrons in the weakly-interacting Ground state $|g\rangle \equiv |0\rangle$.

This type of qubit is short-lived but at the same time allows entanglement, therefore multi-qubit interaction. The form of qubit coupling gr is:

$$-\frac{C_6}{R^6} |11\rangle \langle 11| \quad (4.1)$$

where C_6 is an interaction constant which depends on the type of atom chosen and R represents the distance between the atoms in the lattice. These interactions are always active and this makes difficult to work with the qubits individually, this is why the gr -qubits represent good candidates for high-fidelity entangling operations and quantum simulation.

The energy needed to encode this type of qubit depends on the type of atom chosen. For the Rubidium-87 atom it is necessary to use a laser that reaches ultraviolet light wavelengths. This process is very exhausting for a single laser, so two lasers of different wavelengths are combined together, as can be seen in the Figure 3.5.

- *Rydberg-Rydberg qubits (rr)*. These qubits are encoded using two different Rydberg states, $|r\rangle \equiv |1\rangle$ and $|r'\rangle \equiv |0\rangle$. This type of qubit allows a high degree of flexibility for engineering applications including interactions and long-range bipolar exchange.
- *Ground-Ground qubits (gg)*. This qubit is defined thanks to two hyperfine sublevels of the electronic ground state $|g\rangle \equiv |0\rangle$ and $|g'\rangle \equiv |1\rangle$, this is why it is known as *hyperfine qubit*.

4.3.2 Universal Quantum Computing with Neutral Atoms

Coding via two stable ground states allows to define a qubit with long memory, offering the best performance in terms of coherence time and switchable interactions. For this reason, the hyperfine qubit turns out to be the best candidate for the gate-model with neutral atoms. In the QuEra's quantum computer, the hyperfine qubit is encoded by two hyperfine ground states which are separated in energy by a transition frequency of ≈ 6.8 GHz. The coherence time is relatively long (~ 1 s) thanks to the weak interactions of these qubits with other qubits and with the external environment. Past studies have shown that 2-qubit hyperfine gates achieve fidelity levels of around 97.4% [26], reaching a level of 99.5% in more recent studies [27]. The latest result published in [28] concerns the use of hyperfine qubits for encoding logical qubits in the context of quantum error correction. By logical qubit we mean a structure made up of a certain number of hyperfine physical qubits. The redundancy in the number of physical qubits does not allow error propagation in the event that one or more physical qubit fails. The results

obtained in this study open the doors to a new way to approach the Universal Quantum Computing that involves the use of large-scale logical processors that allow efficient error correction.

Remind that Aquila currently operates with Ground-Rydberg qubits only, with which analog simulation is carried out. Nowadays efforts are underway to incorporate hyperfine qubits onto the platform, thereby enabling Universal Quantum Computation with this technology.

4.3.3 The Rydberg Blockade

Understanding the phenomenon of the Rydberg Blockade is crucial for leveraging the properties of the neutral atoms to solve specific optimization problems, which will be discussed in the next Chapter. This phenomenon is exploited to create entanglement since, given an atom in the Rydberg state, we can deterministically determine in which states the neighboring atoms are configured.

In the previous Sections it is observed that a neutral atom with the valence electron positioned in the outermost orbit is in a strongly excited state, the Rydberg state. The phenomenon of the Rydberg Blockade can be explained by taking into consideration a set of atoms and their mutual distances. If, in this set, there are zero or only one atom in the Rydberg state, then there is no energy shift and the system remains unchanged. If, however, two (or more) atoms are simultaneously in the Rydberg state, there is an energy shift given by a particular intermolecular force called *Van der Waals interaction* which depends on the sixth power of the distance between the atoms

$$V_{ij} = \frac{C_6}{|\vec{x}_i - \vec{x}_j|^6} \quad (4.2)$$

where $C_6 = 5.42 \times 10^{-24} \text{ rad} \frac{m^6}{s}$ denotes the interaction constant between two Rydberg states of two Rubidium-87 atoms; \vec{x}_i and \vec{x}_j are the euclidean coordinates of the atoms i and j . Since this interaction is inversely proportional to the sixth power of the distance, it will be enough to bring the atoms involved closer or further away by a few micrometers to greatly increase or decrease the energy of the system. For example, a distance of only $4\mu\text{m}$ induces a very high energy value ($V_{ij} = 1320 \text{ rad}/\mu\text{s}$) which makes the excitation of both atoms unstable, while with a distance of $16\mu\text{m}$ the energy value becomes negligible ($V_{ij} = 0.32 \text{ rad}/\mu\text{s}$) and both atoms can be in the Rydberg state. The Rydberg Blockade phenomenon, shown in Figure 4.3, is based on this concept: within a certain radius, two atoms cannot be in the Rydberg state at the same time, as if the excitation of one atom “blocked” the excitation of the neighboring atom.

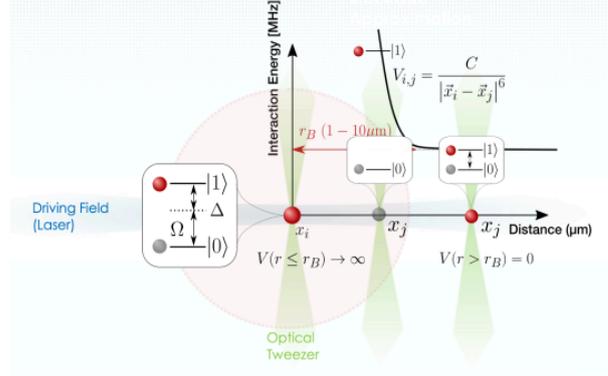


Figure 4.3: Rydberg Blockade mechanism. The atom placed at the origin of the axes is excited thanks to the Δ and Ω parameters of the laser field that pilots the energy passages. We can see that the red circle includes the gray atom positioned at a distance smaller than the Rydberg radius and for this reason remains in the ground state, while for the rightmost atom the transition from the $|g\rangle$ state to the state $|r\rangle$ is possible. The figure was taken from [17].

4.3.4 The Rydberg Hamiltonian

In Section 3.4.1 it was explained how the states of the neutral atoms can be manipulated thanks to the definition of the parameters $\Delta(t)$ and $\Omega(t)$ which define the laser fields. Once the parameters are set, the system follows a specific quantum analog dynamic, considering both the interaction between individual atoms with the laser and electronic fields that define the Rydberg excited state, and the interaction between neighboring atoms, leading to the phenomenon of the Rydberg Blockade. The Hamiltonian involving the parameters mentioned, called *Rydberg Hamiltonian*, is defined as following:

$$H_{ryd}(t) = \frac{\Omega(t)}{2} \sum_i e^{i\Phi(t)} |g_i\rangle \langle r_i| + e^{-i\Phi(t)} |r_i\rangle \langle g_i| - \Delta(t) \sum_i \hat{n}_i + \sum_{i<j} V_{ij} \hat{n}_i \hat{n}_j, \quad (4.3)$$

where $|r_i\rangle \equiv |1\rangle$ is the excited Rydberg state; $|g_i\rangle \equiv |0\rangle$ is the Ground state; $\hat{n}_i = |r_i\rangle \langle r_i|$ is the number of the Rydberg excitations and V_{ij} is the Van der Waals interaction (4.2). The meaning of the parameters $\Delta(t)$, $\Omega(t)$ and $\Phi(t)$ is explained in Section 3.4. As regards \vec{x}_i , these indicate the positions of each atom, determining the strength of the Rydberg-Rydberg interactions. Note that the last summation involves the number of Rydberg excitations with the multiplicative coefficient equal to the Van Der Waals interaction (4.2). When referring to the Rydberg Hamiltonian (4.3), the term *multi-body* is used, underlining that the expression explicitly includes the interactions among all the atoms present in the studied system.

The evolution of the state is described by the following unitary operator:

$$|\psi\rangle = \mathcal{T} e^{-i \int_0^T H_{ryd}(t) dt} |0\rangle \quad (4.4)$$

where that Planck's constant \hbar is equal to 1; the time evolution of the Hamiltonian starts at $t = 0$ and ends at $t = T$ and \mathcal{T} is the *time-ordered evolution operator* which ensures that the operators are applied in the correct chronological order throughout the evolution.

4.3.5 Measurements

To carry out quantum computation and therefore manipulate atoms, they must be trapped in cavities forming a lattice or, where technology allows it, an arbitrary geometry.

QuEra's technology allows the arrangement of atoms into arbitrary geometries and bases measurements on the removal of atoms from traps. More specifically, the lasers that trap the atoms in place during quantum evolution are deactivated to avoid any external influence. Once the evolution of the dynamics has come to an end, these optical lasers are reactivated and the wave function collapses into a particular logical state. Lasers trap atoms in the ground state, while atoms in the Rydberg state are anti-trapped and pushed out of the lattice cavity. Finally, thanks to fluorescence, it is possible to visualize the state of the system through an image: *if a cavity is empty, it indicates that the atom is in the Rydberg state, while if the cavity is occupied, the atom is in the ground state.* These steps are schematized in Figure 4.4.

The basis chosen to interpret the measurement is the Z basis, in which the Z axis of the Bloch sphere is taken as reference. Therefore, if the cavity is empty, the measurement will return 0, otherwise it will return 1. It is important to underline that this type of measurement is destructive, this means that *at each measurement cycle the atoms must be repositioned to their respective places and carry out the entire cycle again.* This step of reloading the atoms is relatively slow compared to the time required for Quantum Computing ($\sim 10\mu s$) but the team of QuEra is currently working on the possibility of partial charging, in order to speed up this process.

4.3.6 The Principal Error Sources

Although the technology offers very precise results given the high quality of the lasers and the nature of the atom, there are sources of decoherence that may drive to a reduction in state fidelity. These sources are the following:

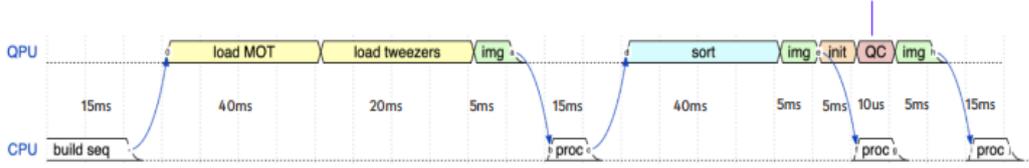


Figure 4.4: The internal steps of Aquila processor. The magneto-optical trap (MOT) is initially loaded and then static traps are filled with atoms from the MOT. A first image of the random occupation of each trap is acquired and processed to allow the reordering of the atoms through the optical tweezers according to the input given by the user. A second image is processed to verify the previous step. The quantum computation is then performed in a very short time. The laser traps are reactivated afterward to confine the atoms in the ground state and remove those in the Rydberg state. Ultimately, the final image is created, interpreted as a bits string in the Z basis. This cycle starts again, placing the atoms back in the vacuum cell and proceeding with another cycle. The image was taken from [17].

- *State decoherence and scattering.* This is the main source of error and is due to the intrinsic decoherence of the atom as its energy decays. Aquila uses two sequences of lasers to go from the ground state to the Rydberg state. Between these two lasers, an intermediate state is created whose energy decays incoherently.
- *Mesaurement.* Based on the atom's exit from the cavities, measurements can be inaccurate if the procedure has not occurred perfectly.
- *Laser Noise.* Although Aquila uses very stable lasers, the presence of noise cannot be ruled out. This causes coherent shot-to-shot variance and time-dependent noise in the parameters Δ and Ω .
- *Atom motion.* Although the atoms are cooled to microkelvin temperatures, they maintain a minimal level of thermal motion which causes a coherent shot-to-shot variance in the detuning Δ .
- *Inhomogeneity.* Due to the imperfect holography of the Rydberg lasers, the Rabi frequency and detuning may vary slightly along the two-dimensional array.

4.4 Conclusions

This Chapter has been dedicated to the analysis of the neutral atom quantum technologies in general, with a focus on QuEra and its hardware, Aquila. First of

all, clarity was provided on the definitions of neutral and Rydberg atoms, followed by an analysis on platforms utilizing these atoms as computational units, with a detailed description of QuEra's hardware. Subsequently, the three types of qubits that can be encoded using the energy levels of an atom were defined, focusing on the potential of the hyperfine state in performing Universal Quantum Computing. It was also emphasized that Universal Quantum Computing, at the time of writing, is not implementable on the Aquila platform. The phenomenon of Rydberg Blockade, crucial for implementing the adiabatic protocol, has been analyzed in detail, defining the Van der Waals interaction and studying how it drastically varies with the distance between two atoms. Using the Rydberg Hamiltonian, the function governing the dynamics of the system composed of atoms arranged in a geometry arbitrarily chosen by the user was defined. Special focus was given to the cycle through which Aquila performs analog simulation, emphasizing the final state measurement phase and outlining the main error sources, primarily related to decoherence and laser noise.

Chapter 5

The Maximum Independent Set problem

In this Chapter, the Maximum Independent Set (MIS) problem on graphs will be addressed, analyzing some of the classical algorithms present in literature and how this type of problem is naturally encoded in neutral atom technology on a particular class of graphs, called Unit Disk Graph (UDG). Finally, it will be studied the binary formulations which allows to solve the MIS problem also in D-Wave's superconducting qubit technology.

It is important to emphasize that the literature on this topic is very extensive, along with the formulation of both exact and approximate classical algorithms for finding the MIS on large dimension graphs. This work focuses on how to exploit the neutral atoms technology to exactly solve the MIS problem and how this formulation can be adapted for optimization problems in different contexts. Therefore, the goal is not to solve the MIS on large graphs, but rather to study the quantum approach adopted by the chosen technology.

5.1 Definition of the Problem

Given a set of variables

$$\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$$

the goal of combinatorial optimization problems is to find the combination of variables which minimizes or maximizes the objective function $\mathcal{F}(x_1, x_2, \dots, x_n)$.

This type of problem can be applied to many real-world contexts, such as portfolio optimization, telecommunications and computer networks, production and logistics management, and much more. Many combinatorial optimization problems can be mapped on graphs, assigning each variable to a node, with edges added

whenever a certain relationship or constraint needs to be represented.

The optimization problems can be classified in classes of computational complexity, which is defined based on the dimension of the problem *instance*, i.e. the problem parameters with specific assignment. Define f as the *time complexity function* associated with an algorithm and $g : \mathbb{N} \rightarrow \mathbb{N}$. The *worst case* in time complexity is defined as:

$$f(n) \in O(g(n)) \text{ if } \exists \hat{n}, c > 0 \text{ s.t. } f(n) \leq c \cdot g(n), \forall n \geq \hat{n}.$$

With this notation, an algorithm is classified as *polynomial* (P) if its complexity class is $O(g(n))$, with $g(n)$ a polynomial function of n . If $g(n)$ is non-polynomial, e.g. $g(n) = 2^n$, the algorithm's complexity is *non-polynomial* and its application becomes prohibitive with the increasing of the instance dimension n . It is important to underline that the inefficiency of a non-polynomial algorithm with respect to a polynomial one is showed when n reaches considerable dimensions. If n is small, the difference between the two classes is negligible.

The *Non-Deterministic Polynomial* (NP) time-complexity class is defined as the set of all decision problems solvable in the worst case by a *non-deterministic* polynomial algorithm. In reference to a decision problem, where the answer is either “yes” or “no,” one can associate with the answer “yes” a set of data (*certificate*) that allows its identification in polynomial time by an algorithm. In this regard, a non-deterministic algorithm recognizes the “yes” instance in two phases:

1. *Hypothesis*: assume to have a “yes”-certificate of the problem;
2. *Checking phase*: verification that the hypothesis is correct.

If the checking phase requires polynomial time, the non-deterministic algorithm is polynomial.

To proceed with further division of the NP-class, it is necessary to clarify the concept of *reducibility*: a problem X' reduces to problem X ($X' \propto X$) if for any instance of X' it is possible to construct in polynomial time an instance X such that from the solution of X we can derive always in polynomial time the solution of X' . The class of problem X such that $X \in \text{NP}$ and $\text{SAT} \propto X$ is called *NP-complete*, with SAT the *satisfiability problem*, which involves to determine if there is a way to assign truth values (“yes” or “no”) to variables in a Boolean formula such that the entire formula evaluates to “yes”. The main feature of the NP-complete problems is that they are equivalently hard and each NP-complete problem can be reduced in any other NP-complete problem. Therefore, if it is possible to demonstrate the existence of a polynomial algorithm capable to solve a NP-complete problem, then $\text{P} = \text{NP}$, which is considered highly improbable.

Finally, a problem X is *NP-hard* if there is no possible to determine if $X \in \text{NP}$,

but it holds that:

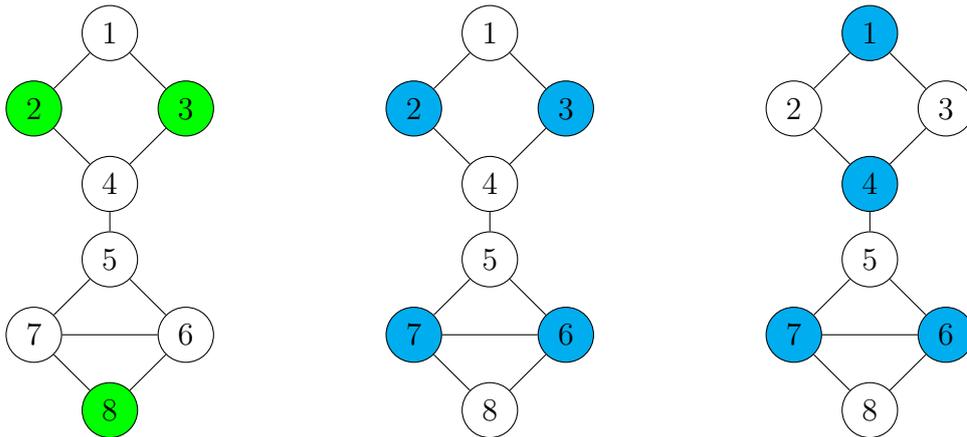
$$\exists X' \text{ NP-complete s.t. } X' \propto X,$$

i.e. X is at least difficult as a NP-complete problem.

5.2 Classical Approaches to Solve the MIS Problem and Related Problems

Given a graph $G=(V,E)$, where V is the vertex set and E is the edge set, an *independent set* of G is a subset $I \subset V$ for which there is no edge between two vertices [29]. An independent set of G is *maximal* if adding vertices to the set implies the presence of an edge between two vertices. An independent set of G is *maximum* if it, compared with all possible independent sets of G , has maximum cardinality.

Examples of these definitions are in the following graphs:



where the green set in the left is a *Maximal Independent Set*, while the blue sets are *Maximum Independent Sets*. Note that the MIS in a graph is not unique. In fact, the sets of vertices $\{2, 3, 7, 6\}$ and $\{1, 4, 7, 6\}$ are two valid MISs of the same graph.

The great interest from the research community in this problem derives from its multiple real-world applications, as positioning of radio antennas [30], social network analysis [31], VLSI Circuit Design [32], chemistry with graph theory [33], schedules design [34] and much more.

The MIS problem belongs to the class of NP-complete problems since:

- Given a set of vertices V , it is possible to verify the satisfiability of the

independence constraints and the maximum dimension of the independence set of vertices in polynomial time;

- It was demonstrated that many NP-complete problem may be reduced to the MIS problem in polynomial time [35].

The first condition is true since this verification is directly proportional to the graph's dimension. Indeed, given an independent set $S \subset V$, for each pair of vertices (i, j) in S , it is verified that the edge $ij \notin E$, involving $O(|S|^2)$ checks.

When the graph is small, the MIS problem can be solved with a simple Integer Linear Programming (ILP) algorithm and implemented in a few seconds using classical solvers, as **Gurobi**[36] or **CBC**[37]. However, if the graph has a considerable number of nodes, this formulation becomes infeasible; for this reason, the MIS problem is the subject of many studies involving both exact algorithms, such as branch-and-bound and exhaustive search, as well as approximate algorithms and heuristics [38].

Concerning exact approaches, according to Fomin and Kratsch in [39], the time complexity of branching algorithms for MIS converges to $O(1.2^n)$. In 1986, J. Roberson in [40] published results from an exact algorithm with a time complexity of $O(1.2109^n)$. In 2017, this result was further improved by Xiao and Nagamochi who achieved a time complexity of $O(1.1996^n)$ [41] with an approach called "*Measure and Conquer*". This method consists of four steps:

1. *Graph measurement*: identification and analysis of graph's properties in order to simplify the MIS problem;
2. *Graph reduction*: with the first step's results, the MIS problem can be reduced and simplified in a smaller subproblem;
3. *Conquering the subproblem*: techniques of MIS solution are applied in the subproblem;
4. *Solution combination*: the solution found in the subproblems are combined together to form a unique solution of the original problem.

Algorithms aiming to approximate the exact optimal solution were also sought but the MIS problem is very difficult to approximate [29]. The study [40] has demonstrated that there is no MIS approximation algorithm that guarantees an approximate solution equal to the exact solution except for a multiplicative constant (known as *approximation factor*). In addition, J.Hastad in [42] proved that there is no polynomial approximation algorithm with approximation factor $n^{1-\epsilon}$, where $\epsilon \in (0, 1)$, unless the NP class is proven to be in reality the Zero-error Probabilistic Polynomial time (ZPP) class, which includes the entire category of

problems that can be solved in polynomial time using a probabilistic algorithm whose probability of error is zero. From this, we can understand that the equivalence between the NP and ZPP class is highly improbable. Furthermore, the goodness of the approximate solutions depends on two characteristic factors of the graph: the number of nodes n and the maximum degree Δ , i.e. the maximum number of edges connected to a single node. The best *approximation ratio*, i.e. the fraction between the approximate solution and the exact optimal one was published in [43] and is equal to $O(n(\log \log n)^2/(\log n)^3)$.

Many studies, such as [44] and [45], instead focus on the efficient computation of near-optimum independent sets for large-scale graphs. In particular, in [46], an iterative approach is applied involving exact/inexact reduction rules until the graph is empty. Recent studies, such as [29], instead focus on computing MIS in *dynamic* graphs, where vertices and edges are dynamically added/removed, mimicking the real behavior of social networks.

The literature regarding the MIS problem is very extensive, including also problems directly connected with the MIS, such as the *Minimum Vertex Cover* and the *Maximum Clique*.

The Minimum Vertex Cover Problem

Given an undirected graph $G = (V, E)$, with V the set of vertices and E the set of edges, the objective of the Minimum Vertex Cover (MVC) problem is to find the set of vertices with minimum size V' such that:

$$\forall i, j \in E : i \in V' \vee j \in V'. \quad (5.1)$$

In other word, the MVC is the set of vertices with minimum cardinality such that every vertices in the graph is incident with at least one vertex in that set, i.e. the complementary set of vertices of the MIS [47]. This problem can be adapted in many real world problems, for example, to solve immunization problems on networks [30] or to identify the group of companies with the greatest impact within a set of financially related firms, application which will be discussed in detail in the next Chapter.

For small-sized graph, this problem can be solved by applying exact algorithms as brute force, Integer Linear Programming and branch-and-bound algorithms. In this regard, the study [48] aims to two novel lower bounds to help prune the search space. The first one is based on the value of the degrees of vertices and the second one concerns the MaxSAT reasoning, i.e. maximizing the SAT problem. The experiment's results show that this approach is better than previously branch-and-bound approaches and some heuristics. However, in [49] it is demonstrated that the MVC problem is NP-complete, since it belongs to the NP complexity

class and there exists problems in NP that can be reduced to the MVC problem in polynomial time. Therefore, finding the solution of this problem with large-scale graph utilizing exact algorithm is infeasible. The study [47], based on past works concerning *Genetic Algorithms* (GA) [50] [51], [52], [53], [54], proposed a new hybrid approach, called *NHGA*. In general, a GA is an heuristics which mimics the behavior of the process of natural selection to *approximate* the solutions of the optimization problem. The heuristic proposed in the cited study involves a new method to generate the initial population, which takes into consideration the degree of each vertex of the graph through the adjacency matrix. The more recent study [55] concerns an approach able to find optimal or near-optimal solutions in polynomial time. It is based on the *Malatya centrality value algorithm*, consisting on two steps: computing the Malatya centrality values of the nodes as the summation of the ratio of the node's degree to the adjacent nodes' degrees for each node and selecting the node for the MVC problem solution based on the node with the maximum Malatya centrality value.

The Maximum Clique Problem

This problem appears in various forms within optimization and graph theory and has numerous real-world applications. For example, in [30] this set represents the solution of a portfolio optimization problem; in [56] is described an application on the gas industry and also biomedical applications are studied in [57][58].

Given a graph $G = (V, E)$ with vertex set V and edge set $E \subseteq \binom{V}{2}$, the subset of G induced by a vertex subset $S \subseteq V$ is denoted by $G[S] = (S, E \cap \binom{S}{2})$. A graph G is complete if all possible edges are defined, i.e. $E = \binom{V}{2}$. A subset of vertices S is a maximum *clique* if its induced subgraph $G[S]$ is complete and its cardinality $\omega(G)$ is the maximum among all cliques. Note that S is a clique in G if and only if S is an *independent set* on \bar{G} , i.e. the *complementary* of G [59].

It was demonstrated that the maximum clique problem is NP-complete [49]. Therefore, for large-scale graph, the solution of the problem is computationally hard. Many studies concern the *Mixed Integer Program* (MIP) formulation, based on the undesirable properties of the integer programming formulation:

$$\omega(G) = \max \sum_{i \in V} x_i \quad (5.2)$$

$$\text{conflicts : } x_i + x_j \leq 1, \quad \forall i \in \bar{E} \quad (5.3)$$

$$x_i \in \{0, 1\} \quad \forall i \in V, \quad (5.4)$$

where \bar{E} is the complementary set of E . From the independence constraints in (5.2), solving the MIS problem on the complementary graph \bar{G} is equivalent to find the maximum clique of the original graph G .

The formulation 5.2 is impractical for large-sized graph due to the weak LP relaxation, the quadratic number of constraints and non-zeros coefficients in the constraint matrix when applied to sparse graphs and poor guarantees on the number of branch-and-bound nodes required to solve it. Past studies on MIP formulations aim to strengthen the LP relaxation of this formulation with the addition of reliable inequalities [60][61], adopted by some MIP solvers as CBC [37] and Gurobi [36]. Other approaches [62][63] concern the aggregation of the conflicts in (5.2) with the big M constraints:

$$\sum_{j \in V \setminus N\{i\}} x_j \leq M_i(1 - x_i) \quad \forall i \in V \quad (5.5)$$

with $M_i = n - |N(i)| - 1$, $N(i)$ the subset of vertices neighboring the vertex i . The formulation with this constraint involves $n^2 - 2m$ non-zero elements, with $n = |V|$ and $m = |E|$. The recent study [64] provides new MIPs formulation, based on the cited approaches. The smallest MIP proposed involves only $O(n + m)$ non-zeros, solved in $O(2^d n)$ branch-and-bounds nodes, where d is the *graph degeneracy*, a measure of the graph's sparsity, while the strongest MIP proposed visits $O(1.62^d n)$ nodes. Denoting with $g = (d+1) - \omega$, $O(2^g n)$ nodes were visited with a best-bound node selection strategy. When g is small, only $O(n)$ nodes were visited.

In [59] is analyzed a technique which allows the parallel cliques search, dividing the problem into smaller ones. Legal coloring of graph nodes serves as a widely adopted method to establish the upper estimates of clique sizes.

Other studies, as [65] [66], focusing on solving the Maximum Clique Transversal problem, which aims to find the smallest set $T \subseteq V$ such that every maximum clique of G contains at least one node from T .

It's important to emphasize that the optimization problems analyzed have been and continue to be a significant subject of study. The literature on this topic is very extensive, Section 5.2 is just a brief overview of the studies concerning these topics.

5.3 The Quantum Approach to MIS Problem

In the next Sections, it will be studied how two quantum technologies, QuEra and D-Wave, solve the MIS problem by exploiting, respectively, the manipulation of the neutral atom and the superconducting qubits.

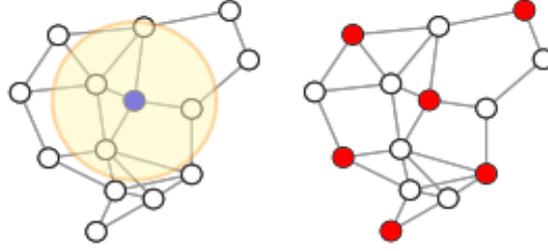


Figure 5.1: An example of Unit Disk Graph (UDG) on the left and the correspondent MIS on the right, taken from [67].

5.3.1 The Neutral Atom Approach

In Chapter 4 the details of the neutral atom technology was studied, with a focus on the Rydberg Blockade phenomenon. In this Section, it will be shown that the Rydberg Blockade phenomenon naturally encodes the independence constraint between variables in a specific class of graphs, called *Unit Disk Graphs* (UDG). These are defined by a set of vertices positioned in 2-dimensional space and by a set of arcs that connects each pair of vertices following a precise rule, showed in Figure 5.1: once a circle of unit radius centered in each node has been drawn, two vertices share the same arc if and only if they share the same circumference. The unitary radius used to define UDGs is identified with the *Rydberg radius*, whose value will be further discussed later.

Consider a graph in which each node is identified by a neutral atom. Exploiting the Rydberg Blockade phenomenon, the independence constraint on the resulting graph will be:

- *Violated*, if the edge connecting two atoms has a length greater than the Rydberg radius. In this case, both atoms can configure in the excited state.
- *Not violated*, if the edge connecting two atoms has a length smaller than the Rydberg radius. In this case the excitation of both atoms is energetically unfavorable and therefore only one of them encodes the Rydberg state.

From this rule, shown in Figure 5.1, it is easy to see how the Rydberg Blockade phenomenon, induced by the excitation of atoms from ground states $|0\rangle$ to excited states $|1\rangle$, naturally defines the MIS problem on UDG whose circumferences are defined by the Rydberg radius R_b .

The definition of the MIS problem on UDG is hardware-efficient since, once the atoms are positioned in the lattice, no further auxiliary atoms (qubits) are required

to provide the problem solution. This represents a great advantage, especially when the hardware capacity is limited as in Aquila which has, at the time of writing, a maximum of 256 atoms. Furthermore, already at first, the Rydberg Blockade mechanism limits the evolution mainly to the subspace spanned by the states that obey the independence constraints of the problem graph. The quantum algorithms that enable the solution of the MIS problem exploit the global excitation of the atoms through homogeneous laser pulses with a Rabi frequency $\Omega(t)$, laser Phase $\Phi(t)$ and Detuning parameter $\Delta(t)$ dependent on time. We have seen that these parameters define the Rydberg Hamiltonian (4.3), which can be decomposed in two parts as following:

$$H_{ryd} = H_q + H_{cost} \quad (5.6)$$

$$H_q = \frac{\Omega(t)}{2} \sum_i e^{i\Phi(t)} |g_i\rangle \langle r_i| + e^{-i\Phi(t)} |r_i\rangle \langle g_i|$$

$$H_{cost} = -\Delta(t) \sum_i \hat{n}_i + \sum_{i<j} V_{ij} \hat{n}_i \hat{n}_j$$

where H_q stands for *quantum driver* and H_{cost} is the *cost function*. By appropriately manipulating $\Omega(t)$ and $\Delta(t)$ for well-defined time intervals, the transition from H_q to H_{cost} and vice versa allows the MIS problem to be naturally formulated on a UDG with Rydberg radius that varies based on the value of the parameters. In particular, the solution is encoded in the ground states of the atoms, i.e. with the lowest energy value. In this way, all the possible independent sets of the UDG are encoded in the low energy subspace and we can exploit this property by evolving the system from one ground state to another, visiting all the possible independent sets and treating them as entangled states. These steps are implemented in the *Adiabatic Protocol*.

The Adiabatic Protocol

The goal of this procedure is to prepare low-energy states such that their measurement encodes possible solutions to the combinatorial optimization problem studied. In this regard, Adiabatic Algorithms represent a well-suited class to carry out this task, as they allow the slowly changing state of the system following a time-dependent Hamiltonian which is, in our case, the Rydberg Hamiltonian (4.3). In the following steps, it is explained how the values of the parameters $\Omega(t)$, $\Delta(t)$ and $\Phi(t)$ can drive the Rydberg Blockade to define the independence constraints in Unit Disk Graphs. Note that the dependence on the time is omitted, focusing only on the parameters value.

- $\Omega = 0$, $\Delta > 0$, $\Phi = 0$. Referring to the expression (5.6), only H_{cost} is considered. Suppose that two atoms are R away from each other. If $\Delta > V_{ij}$,

it is energetically favorable the double Rydberg excitation state $|rr\rangle$ (note the negative sign before the parameter Δ in H_{cost}). If instead $\Delta < V_{ij}$, the excitation of one atom blocks the excitation of the other atom and the double state in which only one atom is in the Rydberg state is favorable, thus defining the superposition

$$\frac{1}{2} |gr\rangle \pm \frac{1}{2} |rg\rangle$$

encoded with the *static Blockade radius* equal to $R_b = (C_6/\Delta)^{1/6}$.

- $\Omega > 0, \Delta = 0, \Phi = 0$. Referring to the expression (5.6), we are considering only H_q . In this case, if the atoms are at a relatively large distance and $\Omega \gg V_{ij}$, they will oscillate independently from the state $|g\rangle$ to the state $|r\rangle$ with a time-dependent sinusoidal trend. If instead $\Omega \ll V_{ij}$, it is energetically prohibitive for both atoms to be in the Rydberg state at the same time and therefore the double excitation state $|rr\rangle$ is blocked. The critical distance at which the energy scale for transitioning atoms from the state $|g\rangle$ to the state $|r\rangle$ equals the interaction energy ($\Omega = V_{ij}$) is called *dynamic Rydberg radius*, and it is given by $R_b = (C_6/\Omega)^{1/6}$.
- $\Omega > 0, \Delta > 0, \Phi = 0$. In this case, the Hamiltonian is actually composed of both H_q and H_{cost} . The combination of the two effects previously described define an energy scale encoding the *Rydberg radius* $R_b = (C_6/\sqrt{\Delta^2 + \Omega^2})^{1/6}$.

Adiabatic state preparation enables the implementation of the Rydberg Blockade with a time-dependent Rydberg radius, varying according to the parameters $\Omega(t)$ and $\Delta(t)$. Specifically, the Adiabatic Algorithm starts with the definition of the initial parameters that define the starting Hamiltonian. These parameters follow a well-defined trend over time, until the target Hamiltonian is reached. If the evolution follows a slow enough dynamic, then the temporal evolution brings the ground state of the initial Hamiltonian into the ground state of the target Hamiltonian H_{targ} . The procedure can be represented by waveforms, as shown in Figure 5.2 and it is summarized in the following steps:

- At time $t_0 = 0$, the ground state $|0\rangle$ is encoded in H_{cost} , with $\Omega(t_0) = 0$ and $\Delta(t_0) < 0$.
- At time $t_1 > 0$, $\Omega(t_1)$ and $\Delta(t_1)$ slowly increase in value, encoding the Rydberg Hamiltonian (4.3).
- At time $t_2 > 0$, $\Omega(t_2) > 0$ and $\Delta(t_2) = 0$ and the state is described by H_q .

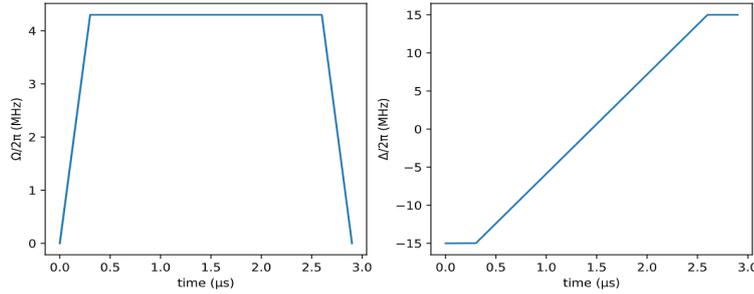


Figure 5.2: The waveforms representing the steps of an Adiabatic Protocol. From the left to the right, the evolution of the parameters Ω and Δ through the time. The Φ laser phase waveform has been omitted, as it is zero for all time instants. The image was taken from the QuEra documentation, available in [68].

- The process continues with $\Delta > 0$ and $\Omega > 0$, until reaching the final instant $t_{fin} > 0$, where $\Delta(t_{fin}) > 0$, $\Omega(t_{fin}) = 0$ and $H_{targ} = H_{cost}$.

The steps through which QuEra’s hardware solves the MIS on UDG are summarized in the Figure 5.3.

5.3.2 The Ising/QUBO Formulation

In this section, the method used by the quantum annealer D-Wave for solving the Maximum Independent Set problem on graphs will be described. Following the Quantum Annealing principle, explained in Chapter 3, the solution of an optimization problem is encoded in the ground state of a Hamiltonian with the lowest energy level. In this way, the minimization of the objective function is equivalent to the minimization of the energy and the optimal solution typically corresponds to the *global* minimum. Due to the definition of the states of superconducting qubits and hardware constraints, D-Wave accepts only *Ising* and *QUBO* formulations which are equivalent under a change of variables.

The Ising Formulation

In this formulation, commonly used in statistical mechanics, the variables can assume values from the discrete set $\{-1, 1\}$. Specifically, the value -1 is associated to a spin that points downwards, while +1 is associated to the opposite spin. The couplings, i.e. the relationships between spins, exploited for Quantum Computing, are correlations and anti-correlations. The objective function of an Ising model

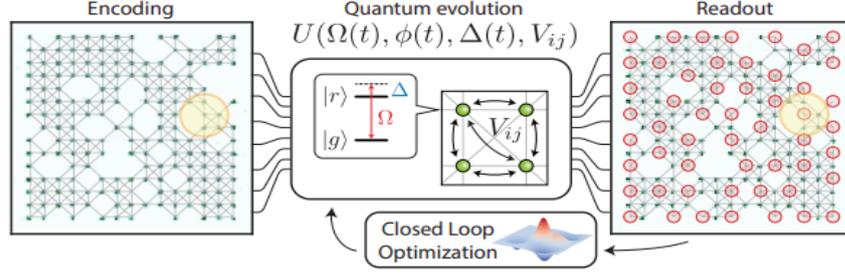


Figure 5.3: Firstly, the atoms are placed at the vertices of the target graph, which corresponds to the Unit Disk Graph describing the optimization problem considered, with radius equal to the Rydberg radius. Subsequently, the system evolves following the dynamics described by the Hamiltonian which depends on the definition over time of the parameters Ω , Δ and Φ . Finally, once the evolution is completed, the state of the system is measured using fluorescence, as described in Chapter 4. The image was taken from [67].

has the following form:

$$H_{Ising}(s) = \sum_{i=1}^N h_i s_i + \sum_{i=1}^N \sum_{j>i}^N J_{ij} s_i s_j, \quad (5.7)$$

where the variables s are the spins, the coefficients h represent the biases and the quadratic coefficients relating the spins are reported in the matrix J_{ij} . The task of the problem is to minimize the Hamiltonian (5.7).

The QUBO Formulation

The QUBO formulation is typically exploited in computer science, as the variables take values in the discrete set $\{0, 1\}$, which corresponds to the set $\{\text{FALSE}, \text{TRUE}\}$. The objective function is formulated as follows:

$$H_{QUBO}(x) = \sum_{i=1}^N q_{ii} x_i + \sum_{i<j}^N q_{ij} x_i x_j, \quad (5.8)$$

where x_i are binary variables; q_{ii} represents the coefficients of the linear terms and q_{ij} are the coefficients of the quadratic terms. Note that (5.8) can be rewritten more briefly as follows:

$$\mathbf{x}^T Q \mathbf{x}, \quad (5.9)$$

where Q is an upper triangular real matrix of size $N \times N$ and the goal is to find the solution which encodes the minimum of (5.8).

The equivalence between Ising (5.7) and QUBO (5.8) formulations can be highlighted by writing the binary variable x as a function of the variable s in the following way:

$$x = \frac{s + 1}{2}, \quad (5.10)$$

hence:

$$\begin{aligned} s = -1 &\Rightarrow x = 0 \\ s = +1 &\Rightarrow x = 1. \end{aligned}$$

Therefore, it holds that:

$$\min_{s_i \in \{-1,1\}} H_{ising}(s) \iff \min_{x_i \in \{0,1\}} \mathbf{x}^T Q \mathbf{x}. \quad (5.11)$$

To solve an optimization problem, D-Wave maps the objective function into a specific graph as follows:

- The variables s_i for the Ising formulation and x_i for the QUBO formulation are mapped to the graph's nodes and represent the physical qubits in the QPU;
- The quadratic coefficients representing the interactions between the variables, i.e. J_{ij} for the Ising formulation and Q_{ij} for the QUBO formulation, are mapped to the graph's edges and represent the couplers in the QPU.

The process by which the problem variables are mapped into qubits in the D-Wave's QPUs is called *minor embedding*.

Minor Embedding and D-Wave's QPUs topology

With minor embedding, the quadratic objective function is mapped into certain graphs, which represent the topology of the D-Wave's QPUs. Specifically, there are three different types of topologies, called *Chimera*, *Pegasus* and *Zephyr*. The three architecture are made up of structures of qubits connected to each other, called *cells*. The connections within the cell are called *internal couplers* and those between cells are called *external couplers*. In Chimera graph, shown on the left in Figure 5.4, these two types of couplers are the only ones present. In advanced QPUs, as the one that will be used in this work, the Pegasus topology is utilized. Qubits are aligned either vertically or horizontally, as in the Chimera topology, but similarly oriented qubits are also shifted. The couplers are classified as internal, external and odd. A unit cell of Pegasus is composed of 24 qubits, each connected with other 12 qubits through internal couplers and 15 different qubits with external couplers; the odd couplers connect similarly aligned pairs of qubits. An example of Pegasus with 4x4 unit cells (P_4) is shown on the right in Figure 5.4.

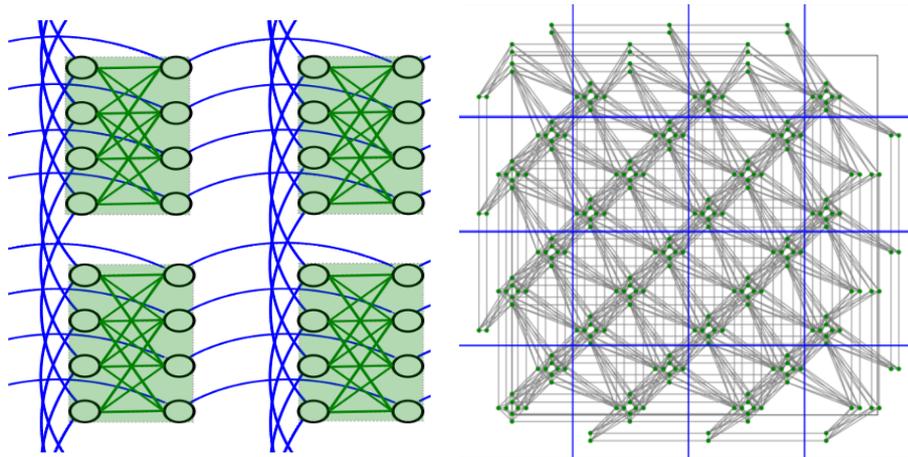


Figure 5.4: On the left, an example of Chimera graph of size 2×2 (2 cells for each column and 2 cells for each row, highlighted in green). Each cell is further composed of 9 qubits, which are interconnected with each other and with qubits from other cells, identifying the external couplers. On the right, Pegasus unit cells in a P_4 graphs; the qubits are the green dots and the couplers are the grey lines. The images was taken from the D-Wave documentation, available in [69].

With the minor embedding, the logical qubits, i.e. those that represent the objective function of the problem, are mapped into physical qubits, i.e. the hardware qubits. It's important to highlight that a logical qubit can be mapped to multiple physical qubits, depending on the complexity of interactions described by the objective function. Despite the complexity of the mentioned topologies, the qubits in the hardware are not fully connected. Therefore, it may be necessary to introduce additional qubits to facilitate proper interactions between variables.

To solve the MIS problem using the D-Wave's QPU, it is necessary to define a *sampler* that takes the problem in binary quadratic form as input, performs minor embedding, and returns the configuration of variables that minimizes the objective function. Therefore, it is essential to define an appropriate binary quadratic problem such that one of the possible solutions corresponds to the Maximum Independent Set (MIS). The specific quadratic binary formulation used in D-Wave is explained in [70] and will be explored in the next Section.

Ising Formulation of MIS Problem

In this Section, the Ising formulation adopted for the MIS problem will be illustrated. Remember that the Ising and the QUBO formulations are equivalent up to a change of variables (5.10).

In this formulation, given a graph G , an edge ij belongs to the set of edge E if

and only if the subsets of vertices V_i and V_j are such that $V_i \cap V_j \neq \emptyset$, where V_i maps to a vertex $i \in V$. A Hamiltonian H is therefore defined as the sum of two Hamiltonians, H_A and H_B , defined as follows:

$$H_{MIS} = H_A + H_B$$

$$H_A = A \sum_{ij \in E} s_i s_j \quad H_B = -B \sum_i s_i \quad (5.12)$$

Note that H_A is minimized only when the set of edges is such that the independence constraint is not violated and H_B takes into account the number of sets that are included. The real parameters A and B are useful for weighting the constraints that represent the Hamiltonian. In this case, by setting $A > B$ it will never be favorable to involve two edges that share the same vertex, thus preserving the independence constraint of the problem.

5.4 Conclusions

In this Chapter, the Maximum Independent Set problem is studied on a small-sized UDG. First of all, the definition of a solution to a combinatorial optimization problem is provided, and the complexity classes of computational problems are described. Subsequently, maximal and maximum independent sets of a graph are defined, along with relevant examples. Several real-world applications of the MIS problem related to other vertex sets connected to it have been presented. A particular focus has been given to classical exact and approximate algorithms, emphasizing their extensive presence in literature and associated time complexities. Given the extensive literature on the subject, some studies concerning algorithms for the MIS, MVC, and Maximum Clique problem have been cited and briefly described. Subsequently, the methodologies implemented by QuEra and D-Wave for solving the MIS problem were presented. Regarding QuEra, a particular class of graphs called Unit Disk Graphs was studied, and it was analyzed how the Rydberg Blockade phenomenon naturally encodes the independence constraint on these graphs. In this context, the adiabatic protocol capable of correctly implementing the problem definition using changes in laser Detuning and Rabi frequency was outlined in more detail. Finally, the Ising formulation enabling the solution of the MIS problem on D-Wave was studied, with emphasis on the topology of its QPUs and the minor embedding procedure, which allows defining the problem on the hardware's QPUs.

Chapter 6

Application of the MIS Problem for Finding MVCs on a Specific UDG Case

In this Chapter, the techniques explained previously will be applied in order to solve an example of a specific combinatorial optimization problem that finds context in financial fields. The solutions were found through the application of neutral-atom and superconducting-qubits approaches, emphasizing the steps implemented for QuEra's technology.

At the time of writing, it was not possible to implement the algorithm directly on QuEra's QPU, therefore it was exploited the local Analog Hamiltonian Simulator, available on *Amazon Braket*¹ [71], which operates with only about ten atoms, limiting the dimension of the problem, constituting just an example of a MIS problem on a small-sized graph. For the QUBO formulation of the MIS problem, it was possible to exploit one of the D-Wave's QPUs tanks to the free cloud service *D-Wave Leap* [72]. Furthermore, when mapping from the mathematical formulation to the graphical representation, careful attention must be paid to the resulting graph, which must not violate the UDG constraints explained in Section 5.3.1.

6.1 The Optimization Problem

In Chapter 5 it was observed that the Maximum Independent Set of a graph is closely related to other sets of vertices, whose properties can be leveraged to encode solutions to combinatorial optimization problems applicable in real-world

¹Amazon Braket is a fully managed Quantum Computing service that allows the user to interact with some technologies on the market.

scenarios. Among these, of particular interest in our application is a subset of vertices called *Minimum Vertex Cover* (MVC), denoted as m_{vc} .

Given a graph G with a set of vertices V and edges E , the Minimum Vertex Cover is defined as the smallest possible subset of vertices $C \subseteq V$ such that every edge $ij \in E$ has at least one vertex in C . Identifying the MIS as M_{is} , it holds that:

$$m_{vc} = V \setminus M_{is} \quad (6.1)$$

i.e. the Minimum Vertex Cover is the complementary set of vertices of the Maximum Independent Set.

The optimization problem studied in this work is explained as following: *given a pool of companies with correlated assets values trends, identify the minimum set of companies that have the most significant impact.* By “impact”, we mean the set of companies whose removal from the initial group would minimize the overall financial risk.

Consider the possibility of defining a graph $G = (V, E)$ in the following way:

- The set of nodes V of the graph corresponds to the set of companies;
- The set of edges E connects the companies according to some rules, which depend on the problem definition.

Suppose to have a group of companies listed on the Italian stock exchange and to have access to the historical series of annual trends of their assets. Thanks to the time series, it is possible to compute the variance and covariance matrix Σ , which can be a valid candidate as a starting point in defining the relationships between the companies, i.e. the edges of the graph. In our case, having represented each company as a node of the graph, we connect with an edge those companies that have a covariance value both greater than zero and greater than a certain threshold, denoted by β . In this way, we are representing a set of companies with:

- The same trend from the point of view of the financial risk. In fact, by defining positive covariance, we are connecting all those companies with a similar trend and therefore, in the event of a collapse of the first company’s assets value, the probability of the collapse of the second company’s assets value is greater than zero.
- Connections deriving from non-negligible relationships. If we did not define any threshold, the resulting graph would be almost *fully connected* and therefore useless from an application point of view (the MIS should be composed of only one node). Furthermore, many of the relationships, even if defined with a covariance value greater than zero, would be negligible compared to others. In other words, the definition of the threshold $\beta > 0$ allows us to consider only the relatively important correlations.

By doing so, the resulting graph identifies a pool of companies with a similar trend of financial risk and our objective will be to select the minimum set of companies with the greatest impact in a specific context. This set is represented precisely by the MVC, whose solution can be represented as the complementary set of vertices of the MIS.

The problem just described has the following Integer Linear Programming (IP) formulation:

$$\min_{i \in V} \sum_{i \in V} x_i \quad (6.2)$$

$$\text{s.t. } x_i + x_j \geq 1 \quad (6.3)$$

$$\forall (i, j) \in E : \Sigma[i, j] \geq \beta \quad (6.4)$$

$$x_i \in \{0, 1\} \forall i \in V, \beta > 0 \quad (6.5)$$

where the objective function (6.2) aims to minimize the vertices included in the optimal subset; the constraint (6.3) ensures that, for each edge, at least one vertex is included in the subset (guaranteeing the optimality of the solution) and the constraint (6.4) ensures that the edges exist if and only if the companies involved have covariances greater than the value of the threshold β .

Since the neutral-atom technology can only solve the MIS problem, the UDG describing the problem is first defined. Subsequently, the MIS problem on it is solved, which can be formulated with the following ILP:

$$\max_{i \in V} \sum_{i \in V} x_i \quad (6.6)$$

$$\text{s.t. } x_i + x_j \leq 1 \quad (6.7)$$

$$x_i \in \{0, 1\} \forall i \in V. \quad (6.8)$$

Finally, the sought MVCs are defined as the complement vertex sets of the MISs in the reference UDG.

In this work, it was considered a dataset of companies and their respective asset values on the Italian stock exchange for the entire year 2023, available on *Google Finance* [73]. After having defined the appropriate time series, the threshold has been set to the 25% of the maximum covariance value of the resulting matrix Σ , obtaining the 15-nodes graph shown in Figure 6.1. It is useful to report the map from the node number to the company label:

0: AutME; 1: Pirelli; 2: Avio; 3: ENI; 4: FINECO; 5: AirFrance;
 6: STMI; 7: PosteIT; 8: Stellantis; 9: MEDIOLANUM; 10: LEONARDO;
 11: SNAM; 12: AQUAFIL; 13: AristonHolding; 14: UniCredit.

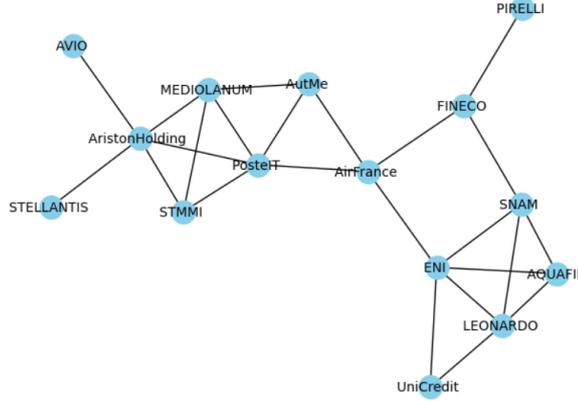


Figure 6.1: The graph resulting from the initial dataset of companies and from a threshold $\beta \approx 1.385$.

6.2 Check the Feasibility of the Graph Obtained from the Time Series Analysis

As specified in Section 5.3.1, neutral atom quantum technology naturally defines the MIS problem on Unit Disk Graphs. Therefore, it is first necessary to verify that the graph encoding the optimization problem (6.6), generated by following the steps listed in Section 6.1, does not violate the definition of Unit Disk Graph. Consider a set of vertices V and a list of edges E , defined according to the constraint (6.4). The spatial representation of the resulting graph G , therefore the Euclidean coordinates of the vertices in V , depends on the kind of *layout* chosen. For the purpose of this work, it was selected the *Kamada Kawai layout*², applying an optimization algorithm for positioning the graph vertices in the Euclidean plane. More precisely, this algorithm minimizes a cost function that takes into account the nodes distances and the differences between the real and the predicted lengths. This cost function can be formulated as follows:

$$C = \sum_{i=1}^N \sum_{j=i+1}^N k_{ij} \left(\frac{d_{ij} - l_{ij}}{l_{ij}} \right)^2, \quad (6.9)$$

where N is the number of nodes in the graph; k_{ij} is a constant that depends on the distance between the nodes i and j ; d_{ij} is the Euclidean distance between i and j at the current position; l_{ij} is the ideal length between i and j . Thanks to this layout, it was possible to obtain a symmetric and well-balanced graph representation.

²Type of layout defined in *NetworkX* package in *Python* library.

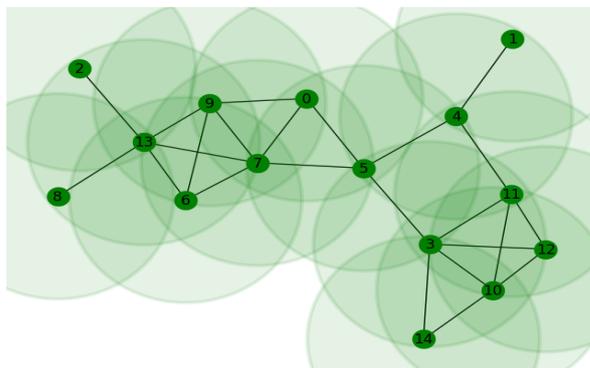


Figure 6.2: The definition of the UDG starting from the vertices of the original graph represented in Figure 6.1. Defining the circumferences for each nodes with radius R_{UDG} , all the vertices with distance less than that value are connected.

Once the set of vertices V is arranged in the coordinates optimized by the chosen layout, it is possible to connect all those vertices that have Euclidean distance less than a certain radius, called R_{UDG} . To define the value of this radius, it was useful to reason about the edges lengths of the original graph G . In fact, the limit value for which we could define a graph as UDG is identified in the maximum length of its edges. In this way, by imposing the constraint that the union of any two vertices is less than R_{UDG} , all the edges of the graph G will also be defined in G_{UDG} , as represented in Figure 6.2. Finally, it was made a comparison between the original graph and the one obtained with R_{UDG} : only if the list of edges of G and G_{UDG} coincides, G can be defined as UDG and we can take it as the starting graph to obtain the MIS problem solution by exploiting the neutral atom technology.

6.3 QuEra's Solution

In this Section, the steps of the algorithm implemented to exploit the QuEra's local analog simulator will be analyzed, summarized in the following list:

1. *Map the UDG into atomic coordinates;*
2. *Definition of the Adiabatic Algorithm;*
3. *Saving solutions and post-processing;*
4. *Solution analysis.*

The following Sections will be dedicated to an in-depth analysis of the steps mentioned above.

6.3.1 Step 1: map the UDG into atomic coordinates

Once verified that the graph obtained is UDG, it is necessary to move to dimensions that allow the analog simulator to work. This means that, given the Euclidean coordinates of the set of vertices V of the graph $G \equiv G_{UDG}$, we need to define a new graph G_{atom} and a new set of vertices V_{atom} that respects the dimensions and the spatial constraints of the hardware. In fact, in Section 4.2.1 it is explained that the atoms must be arranged in an area with a width of less than $200\mu\text{m}$ in order to undergo the appropriate laser manipulations. Therefore, it is required to scale the graph down to micrometer-level coordinates while preserving the original shape, thus performing a *geometric reduction* by multiplying each vertex of the set V by a scale constant a , defined as follows:

$$a = \frac{R_b}{R_{UDG}}. \quad (6.10)$$

From the set V in the new atomic coordinates, the edges of the graph G_{atom} will be generated following the rules defining a UDG with radius equal to the Rydberg radius R_b . Applying this step, we can see how the structure of G is reflected in G_{atom} :

$$a * D(i, j) \leq R_b \Rightarrow \frac{R_b}{R_{UDG}} * D(i, j) \leq R_b \Rightarrow D(i, j) \leq R_{UDG}, \quad (6.11)$$

where D is the symmetric matrix of Euclidean distances between the nodes of the graph G .

Now, the question arises as to what value should be assigned to the parameter R_b in the ratio (6.10) to ensure the correct encoding of the independence constraint. In Section 5.3.1, we have observed that the parameters of the adiabatic evolution define, over time, different type of Rydberg radius, which values define the Blockade phenomenon. It is worth underlining that there is no formula that provides an exact value of R_b for the ratio a . In this work, it was possible to define a value that gave optimal results tanks to many tests carried out with different UDGs and considering the approximation illustrated in QuEra's documentation, available in [74], that is:

$$\frac{C_6}{R_b^6} \sim \sqrt{\Delta^2 + \Omega^2}, \quad (6.12)$$

with $C_6 = 5.42 \times 10^{-24} \text{ rad} * m^6/s$, Δ and Ω equal to their maximum value in the Adiabatic Protocol.

6.3.2 Step 2: the adiabatic evolution parameters

Once the atoms have been arranged in the vacuum cell, the next step is to define the parameters $\Delta(t)$ and $\Omega(t)$ for the Adiabatic Protocol. The Adiabatic Algorithm was implemented following the AWS³ tutorial available in [75] and taking as reference the study [67] and the paper [17] regarding the values of the parameters. Remember that the Adiabatic Protocol consists of various changes in the values of $\Delta(t)$ and $\Omega(t)$ which can remain constant, increase or decrease for certain time intervals. Therefore, it was necessary to set sub-intervals of time $\delta t_i = t_i - t_{i-1}$ between the initial instant t_{in} and the final instant t_f in which the values of Δ and Ω change, as illustrated by the adiabatic procedure. Following several tests involving different parameter values, the algorithm was implemented with the following components, illustrated as waveforms in Figure 6.3.

- *Time intervals.* Between the initial instant $t_{in} = 0$ and the final instant $t_{fin} = 4 \mu s$, the instants $t_1 = 0.6 \mu s$ and $t_2 = 3.4 \mu s$ define three sub-intervals: $\delta t_1 = 0.6 \mu s$, $\delta t_2 = 2.8 \mu s$ and $\delta t_f = 0.6 \mu s$. Note that the length of this time interval is $4 \mu s$ which is exactly the estimated time for which the annealing schedule of Aquila is coherent.
- *Rabi frequency.* The Adiabatic Algorithm requires the starting value of Ω equal to zero ($\Omega(t_{in}) = 0$), then grows ($\Omega(t_1) = \Omega$), remains constant for a given time interval ($\Omega(t_2) = \Omega(t_1)$) and return to zero at the final instant ($\Omega(t_f) = 0$). The chosen value for the implementation was $\Omega = 15 \text{ rad}/\mu s$.
- *Laser detuning.* This parameter takes on a negative value in the initial instant ($\Delta(t_{in}) = -\Delta$), remains constant for the first time interval ($\Delta(t_1) = \Delta(t_{in})$), and then increases ($\Delta(t_2) = \Delta$) until it remains constant for the last time interval ($\Delta(t_2) = \Delta(t_f)$). In this work, it was considered $|\Delta| = 30 \text{ rad}/\mu s$.
- *Lase phase.* The Φ laser phase was set to zero at each time instant, as required by the Adiabatic Protocol.

6.3.3 Step 3: saving solutions and post processing

Once the values of the parameters have been defined, the analogue quantum simulator of QuEra is called, which simulates the dynamics described by the Rydberg Hamiltonian (4.3) with the parameters represented by the waveforms in Figure

³Amazon Web Services (AWS) is a platform owned by Amazon that offers more than 200 cloud computing, content processing and delivery services worldwide.

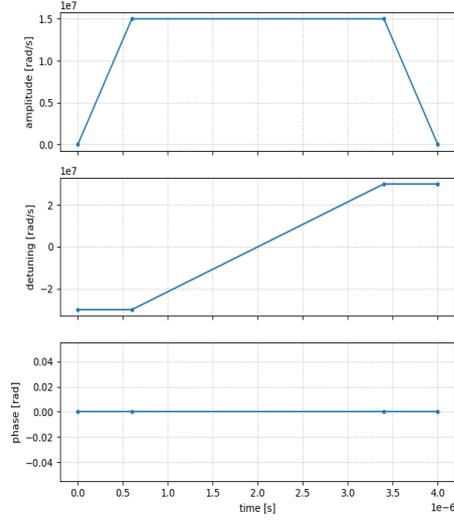


Figure 6.3: Ω , Δ , and Φ waveforms for the time interval considered.

6.3. The number of shots or measurements conducted is equal to 1000. In Section 4.3.5, the method used by QuEra to measure the state of the system was explored, underlining that this process is destructive and therefore, with each new measurement, the atoms must be re-inserted into the lattice. Remember that a solution is represented by a binary string in which the value 0 represents the atom in the Rydberg state, while the value 1 represents the atom in the neutral ground state. In the first case, the atom encodes a vertex in the Maximum Independent Set, while in the second case it identifies a vertex in the Minimum Vertex Cover set. In Section 4.3.6 it was observed that errors can occur in measurements, which reflects into the results of MIS in two possible ways:

1. The string represents a *maximal* rather than *maximum* independent set. In this case, we have a set of vertices that does not violate the independence constraint but represents a set of vertices with cardinality lower than the optimal one. This conditions, as reported in [17], can occur with $\sim 1\%$ probability.
2. The string represents a set of vertices that violates the independence constraint. In this case, the set can identify two vertices that share an edge, thus violating the independence constraint. Sets with cardinality greater than the optimal one certainly violate this constraint, but this error can also occur in sets with optimal or lower cardinality. This conditions, as reported in [17], can occur with a $\sim 1\%$ probability.

Therefore, it is necessary to study a method that verify the solutions obtained,

excluding the incorrect ones. This postprocessing phase was implemented by analyzing the positions of the zeros for each string. It holds that 0 is equivalent to an atom in the Rydberg state which corresponds to a vertex in the MIS and it also holds that two zeros cannot be represented by two neighboring nodes in the graph. Therefore, by checking the neighboring nodes for every 0 placed in each string, we have that:

- If only one node among a 0's neighbors is coded as 0, then the solution is wrong;
- If all nodes that are neighbors of a 0 are coded as 1, then, for that node in the independent set, the independence constraint is not violated.

Note that the time complexity of this post-process is proportional to the graph dimension. Therefore, this step can be computed in polynomial time.

A pertinent question arises as to how the cardinality of the MIS can be determined solely based on the simulation results. The approach involves calculating the cardinality of the MIS encoded for each string. If the dynamic is correctly implemented, the most frequently occurring cardinality among the solutions is likely to be the optimal one.

6.3.4 Step 4: result analysis

The last step concerns the analysis of the results obtained after the post-processing. First of all, through a *bar plot*, the different cardinalities found in the set of the solutions can be represented, with the respective probabilities of encountering them at the end of the simulation. As can be seen from the blue bar plot in Figure 6.4, the most frequent cardinality among 1000 simulations is 7 vertices (93%), but with a small percentage are also found sets of 6 and 8 vertices. From this result, it can be deduced that the MIS is made up of 7 nodes.

At this point, post-processing was carried out, which confirmed the cardinality of the MIS. In fact, if 7 is the optimal cardinality, then the entire set of 8 vertices is infeasible and it is therefore eliminated. The green bar plot in Figure 6.4 shown that the set of feasible solutions is made up only of independent sets of cardinality 6 and 7, confirming the thesis.

Regarding the percentage of infeasible solutions, out of a total of 930 solutions of cardinality 7, two are infeasible, defining the probability of 0.2% of finding wrong solutions in the optimal set.

Once the infeasible solutions have been removed from the set of optimal cardinality, the different types of MIS solutions can be identified. For this graph, 7 different feasible MISs were found, as shown in Figure 6.5.

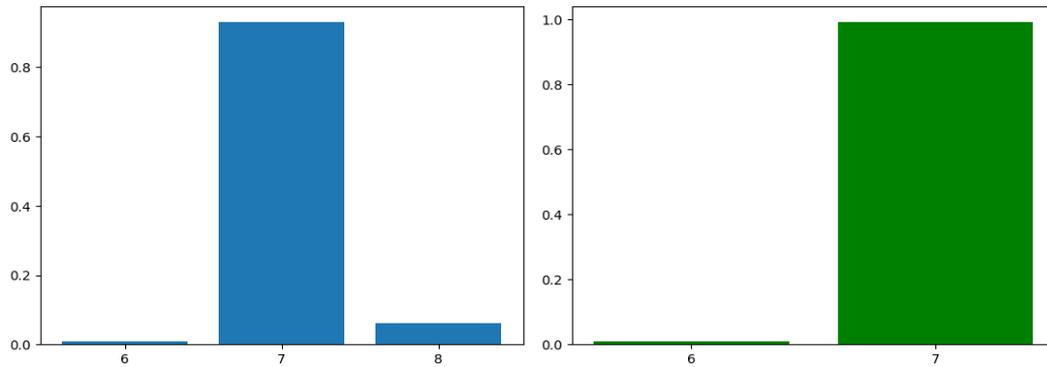


Figure 6.4: Bar plots representing the different cardinalities found in the simulations with respective probabilities. The blue bar plot represents the set of solutions obtained by the simulator; the green bar plot represents the set of solutions after the post-processing phase.

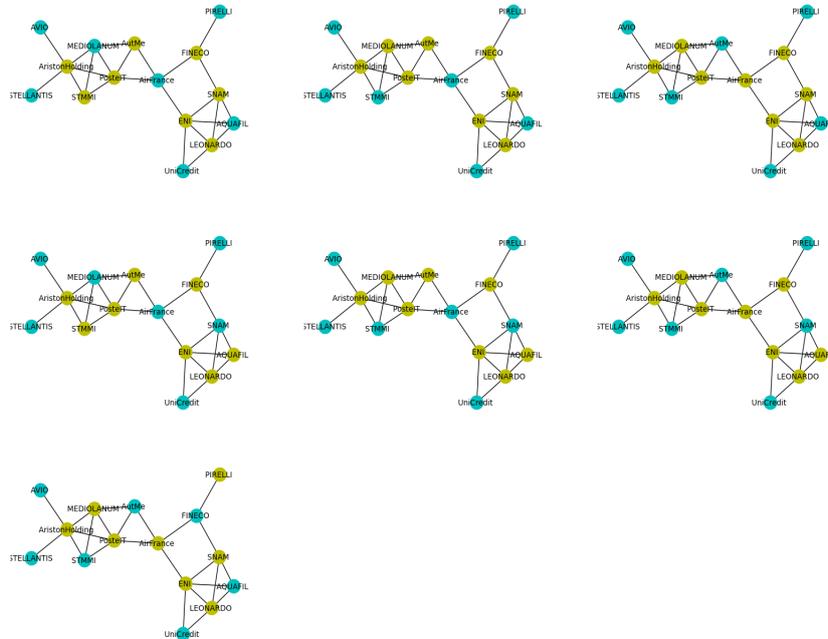


Figure 6.5: The different types of MISs found in the set of feasible solutions. The blue nodes represent the vertices in the MIS, while the yellow vertices are in the MVC set.

6.3.5 QuEra’s Results Considerations

To find the QuEra’s solution, it was exploited a local Analog Hamiltonian Simulator which *simulates* the behaviour of the neutral atom hardware to find the MIS

problem solutions. Typically, this simulator is used to test the algorithm before the implementation on the QuEra’s QPU, as it simulates the system dynamics exactly as the hardware would, as shown in [76], where the results obtained from QuEra match the predictions made by the local simulator. Additionally, Aquila’s coherence time of annealing schedule has been estimated to be $4 \mu\text{s}$, exactly the duration of the implemented Adiabatic Protocol. This suggests that the results provided by the QPU should not significantly differ from those obtained by the simulator. Despite this, the results obtained by the hardware may have some empty sites due to hardware’s atom preparation imperfections. With a probability of approximately 0.7%, the QPU fails to occupy a user-specified filled position, thereby defining the the set V_{atom} .

Concerning the time of execution, although the hardware was not used directly, the documentation states that Aquila can complete up to 10 shots per second [17], performing a single quantum computation in $\sim 10 \mu\text{s}$, while the time spent by the local simulator to perform 1000 shots and to obtain the solution to the considered MIS problem is approximately 7 minutes.

The MIS formulation on UDG is hardware-efficient for QuEra, it means that *no additional qubits are required to solve the problem*. Therefore, the number of involved qubits it’s exactly 15, the number of nodes of the graph, as shown with the pink bar in Figure 6.7.

6.4 D-Wave’s Solution

This technology exploits some quantum properties, with the aim of exploring all the solutions *simultaneously*. At the end of the annealing process, with a given probability, only one solution among a set of good solutions is chosen as the output. For this reason, it is useful to carry out the sampling phase several times, because each iteration can result in a different output. In particular, through the `EmbeddingComposite` class, it is possible to map the binary problem into one of the QPUs topology by performing the *minor embedding*, explained in Section 5.3.2. Once the logical qubits are mapped onto the physical ones, the D-Wave sampler⁴ is called, which performs from low-energy states in models defined by the Ising or QUBO formulations, illustrated in Section 5.3.2.

For the D-Wave’s results analysis, the same procedures used for the neutral atom technology were followed. Therefore, the cardinality of the MIS is determined by identifying the highest frequency among the computed values. As can be seen from the blue bar plot in Figure 6.6, even for the superconducting quantum annealer, the MIS has cardinality 7 (98.6%) and a small percentage of sets of 6 and 8 ver-

⁴The sampler used is `DWaveSampler()` from the `dwave.system` package of the D-Wave library.

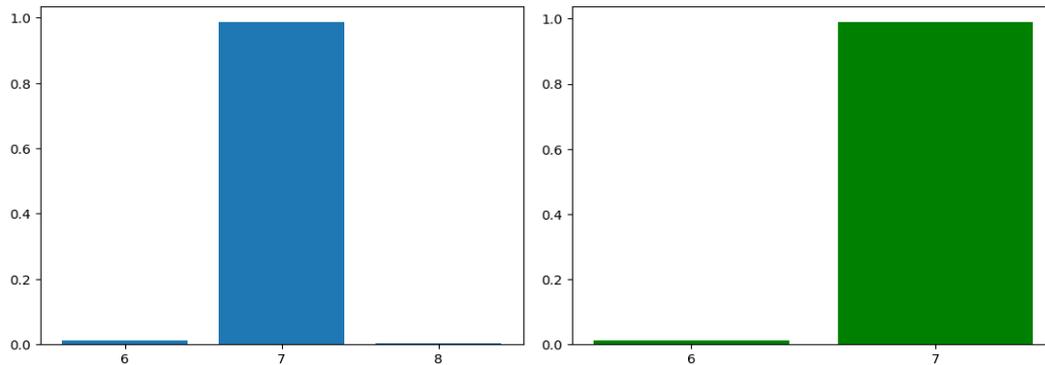


Figure 6.6: Bar plots representing the cardinalities and the respective probabilities of the solutions sets provided by 1000 runs of the D-Wave sampler. The blue bar plot represents the set of solutions obtained by the sampler; the green bar plot represents the set of solutions after the post-processing phase.

tices were found. On this set of solutions, the post-process procedure described in Section 6.3.3 is carried out and the set of totally feasible solutions is shown with the green bar plot in Figure 6.6. As illustrated, also in this case, all sets of 8 vertices are excluded.

In this results, apart from three different solutions that finds 8 vertices, there are no others wrong vertex sets. Furthermore, analyzing the different results relating to the optimal cardinality, we find the same 7 solutions as QuEra, represented in Figure 6.5.

6.4.1 D-Wave’s Results Considerations

The MIS problem with the superconducting-qubits quantum annealer was implemented in *Advantage System 5.4*, which is one of the QPUs available in *D-Wave Leap* service. This advanced QPU has a maximum capacity of 5614 physical qubits, with a topology described by the *Pegasus* graph P_{16} , illustrated in Section 5.3.2. As explained in Section 5.3.2, to solve an optimization problem, the hardware maps the *logical qubits* defining the Ising/QUBO formulation onto *physical qubits* in the QPU topology. This procedure, called minor embedding, can involve an higher number of physical qubits with respect to the logical ones and different iterations can exploit different numbers of physical qubits. To solve this specific problem, the number of qubits exploited varying in a range from 15 to 19 qubits with a higher frequency of 16 and 17 qubits, as shown with magenta bars in Figure 6.7.

Concerning the time of execution, as reported in the documentation of D-Wave

[77], the time to execute a single Quantum Machine Instruction (QMI) on a QPU is formed by two parts. The first one concerns the initialization of the program onto the QPU and the second one regards multiple sampling for the actual execution on the hardware. Considering also some overheads, the QPU access time can be estimated with:

$$T = T_p + \Delta + T_s$$

where T_p is the programming time, Δ is an initialization time spent in low-level operations and T_s is the sampling time, which can in turn be defined as:

$$T_s \approx T_a + T_r + T_d$$

where T_a is the single-sample annealing time; T_r is the single-sample readout time, i.e. the time required to read the state of a single quantum sample; T_d is the single-sample delay time, i.e. the duration between consecutive measurements of a single quantum state on a quantum processor. The solution of this problem is found with an estimated average QPU time equal to 0.017 seconds per sample, with an estimated average sampling time equals to 115.12 μs per sample, composed by $T_a \approx 20 \mu\text{s}$, $T_r \approx 74.5\mu\text{s}$ and $T_d \approx 20.6 \mu\text{s}$. To compute the energies of the returned samples, a post-processing phase is performed in an efficient way, which allows to spent only 1 μs per iteration. It is important to underline that the time to obtain the solution also depends on the *access time*, which is determined by how many QMIs are present in the system at a given time.

6.5 Two Technologies Compared

The approaches used by D-Wave and QuEra to solve the MIS problem are very different, although they refer to the same principle by which the solutions are encoded in the lowest energy states, identified by the evolution of a dynamics described by a time-dependent Hamiltonian. Although QuEra's results are obtained from a simulator and not directly from the QPU, a comparison between the solutions from the two approaches remains worthwhile.

An interesting comparison concerns the distribution of frequencies with which the two technologies find the different feasible MISs. These frequencies are reported as percentages in table 6.1, with the representation of the two different distributions in Figure 6.8. From this, it can be seen that QuEra's solutions strongly dependent on the arrangement of the atoms in the graph and their mutual distances, leading to some solutions being reported more frequently than others. D-Wave employs a different approach, which is completely independent of the problem graph's topology, resulting in a uniform distribution of the solutions found.

Another term to consider concerns the overhead of the number of qubits needed to solve the MIS problem in the two technologies. The formulation of the MIS

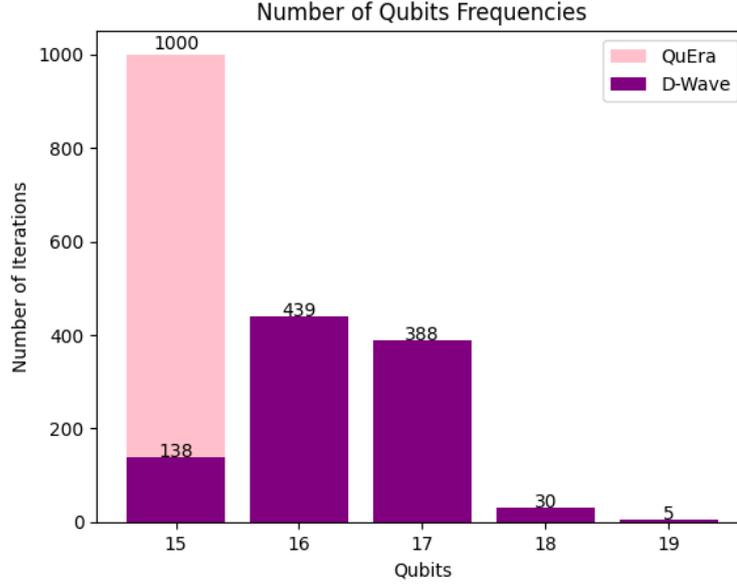


Figure 6.7: Bar plot representing the frequency of qubit usage by QuEra and D-Wave’s QPU for solving the MIS problem over 1000 iterations, with the x-axis showing the number of qubits used and the y-axis showing the number of times each qubit count was used.

MIS	QuEra %	D-Wave %
{1, 2, 5, 8, 9, 12, 14}	45.58 %	13.58 %
{1, 2, 5, 6, 8, 12, 14}	26.5 %	14.47 %
{0, 1, 2, 6, 8, 12, 14}	15.09 %	9.71 %
{1, 2, 5, 8, 9, 11, 14}	6.03 %	19.62 %
{1, 2, 5, 6, 8, 11, 14}	3.45 %	16.75 %
{0, 1, 2, 6, 8, 11, 14}	3.02 %	11.20 %
{0, 2, 4, 6, 8, 12, 14}	0.32 %	14.67 %

Table 6.1: Percentages of different MISs configurations in the solution set

on UDG is hardware-efficient in QuEra, this means that the qubits used for the computation is equal to the number of nodes in the graph, without any overhead. As regards D-Wave technology, the graphs that make up the different topologies of its QPUs are not fully connected. This translates into an overload in the number of physical qubits in order to encode the problem being considered in the most correct way. A comparison of the number of qubits used by the two technologies through the iterations is shown in the bar plot 6.7.

It is also important to note that QuEra efficiently solves the MIS problem on

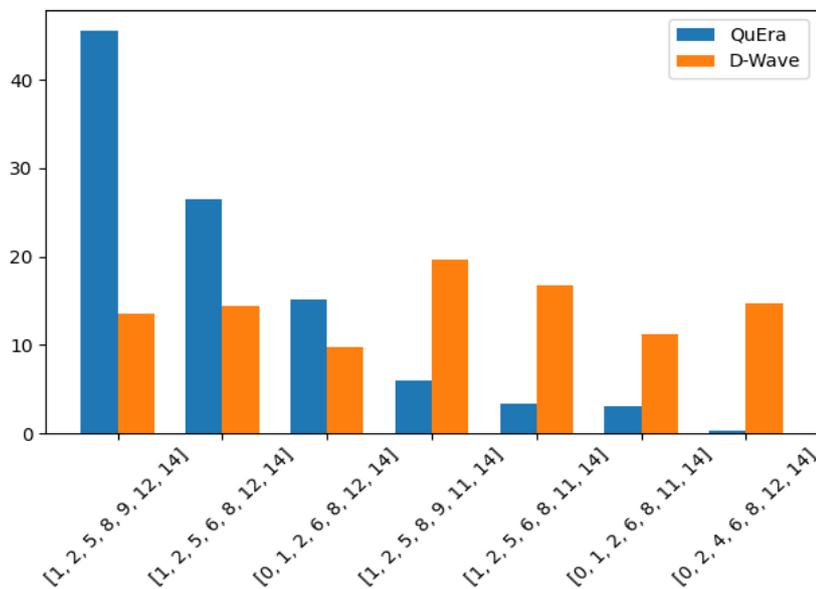


Figure 6.8: Bar plot representing the frequencies listed in the table 6.1.

UDGs, which constitute a very particular class of graphs, while D-Wave’s technology is totally independent from the problem graph’s topology. To overcome this limitation, the study [78] explain a procedure called *gadgetization*, which allows mapping a general graph into a UDG with a node overhead of at most $4N^2$, given N the number of vertices of the graph. For the purpose of this work and for the maximum capacity of the local simulator, it was decided to take into consideration a graph that can be defined a priori as UDG.

6.5.1 MIS Problem Solutions Benchmarks

In the previous Section, the results obtained from two quantum approaches was analyzed but the quality of the solutions found remains uncertain. For this reason, the analyzed problem was also solved by implementing two classical approaches: the direct formulation of the ILP (6.6) using the software `Xpress`⁵, and the *Simulated Annealing* heuristic, detailed in **Appendix A**.

Since the problem size is very small, it has been possible to implement the ILP formulation (6.6) on the software `Xpress` to find the correct MISs of the reference graph 6.1. This implementation took 0.3 seconds, providing the following seven

⁵Xpress is a mathematical optimization solver developed by FICO (Fair Isaac Corporation). It is used to solve complex mathematical programming problems, including integer linear programming (ILP).

7-nodes MISs:

$$\begin{aligned}
 & \{1, 2, 5, 8, 9, 12, 14\} \\
 & \{1, 2, 5, 6, 8, 12, 14\} \\
 & \{0, 2, 4, 6, 8, 12, 14\} \\
 & \{0, 1, 2, 6, 8, 12, 14\} \\
 & \{1, 2, 5, 6, 8, 11, 14\} \\
 & \{0, 1, 2, 6, 8, 11, 14\} \\
 & \{1, 2, 5, 8, 9, 11, 14\},
 \end{aligned} \tag{6.13}$$

thus confirming the results obtained by the two quantum approaches (6.1). Simulated Annealing, the classical counterpart of quantum annealing, is frequently used in the quantum context to obtain benchmarks for evaluating the quality of the results obtained by quantum annealers, as shown in [67]. The MISs found by this algorithm exactly match those obtained from the direct formulation (6.13) and the two quantum annealers (6.1), providing further validation of the solutions quality.

6.6 Analysis of the results

In the previous Sections, it was explored how the MIS problem can be addressed. Remember that, although the goal was to find the MIS, the financial problem solution is represented by its complement, the MVC. Since there are 7 different types of feasible MISs, there are also 7 feasible MVC sets, represented by the following sets of companies:

- ENI, PosteIT, LEONARDO, ArisonHolding, AutMe, STMMI, FINECO, SNAM;
- ENI, PosteIT, LEONARDO, ArisonHolding, AutMe, MEDIOLANUM, FINECO, SNAM;
- ENI, PosteIT, LEONARDO, ArisonHolding, MEDIOLANUM, AirFrance, FINECO, SNAM;
- ENI, PosteIT, LEONARDO, AristonHolding, AutMe, STMMI, FINECO, AQUAFIL;
- ENI, PosteIT, LEONARDO, ArisonHolding, MEDIOLANUM, AutMe, FINECO, AQUAFIL;
- ENI, PosteIT, LEONARDO, AristonHolding, MEDIOLANUM, AirFrance, FINECO, AQUAFIL;
- ENI, PosteIT, LEONARDO, AristonHolding, MEDIOLANUM, AirFrance, PIRELLI, SNAM;

Consequently, the question arises as to which solution among those identified is the optimal one.

In terms of optimization, all 7 solutions are equivalent to each other, so it cannot be decided a priori what MVCs are better than others. However, in financial terms, the solutions would not be equivalent. To find an answer to the question, it can be chosen to evaluate some market indices for each company in the MVCs. Through these parameters, it is possible to identify which solution is better than others, thus obtaining a single solution to the optimization problem. As can be seen from Figure 6.5, the companies “ENI”, “PosteIT”, “LEONARDO”, “AristonHolding” appear in all solutions and it would therefore be useless to consider them in a subsequent market analysis, as they would not influence the decision of one set being better than another. The market analysis goes beyond the scope of this work, however, some useful indices are provided below as examples.

- *Leverage ratio.* This class of metrics includes different types of ratios that consider a company’s level of debt relative to equity.
- *Liquidity ratio.* These metrics evaluate the company’s ability to pay its short-term debts.
- *Profitability ratio.* In this case, the company is evaluated based on its ability to generate profits.

Finally, it is worth to note that there is a way to consider MIS solutions as not equivalent to each other, therefore preferring one solution rather than another, without proceeding with the market analysis. In this alternative, called *Maximum Weighted Independent Set* (MWIS) problem, the MIS is formulated by assigning a weight to each vertex of the graph, following a chosen criterion. The optimal solution, in this case, will be represented by the independent set of maximum cardinality that also maximizes the sum of the nodes weights. However, in the absence of the necessary documentation at the time of writing, it was not possible to implement the MIS problem as a MWIS problem.

6.7 Conclusions

In this Chapter, the algorithm implemented to solve a small-sized example of the MIS problem using both QuEra and D-Wave technology was analyzed. Initially, the application context of the chosen problem in the real world was defined, detailing the constraints that allowed for obtaining the reference graph. After providing a mathematical formulation of the problem, the implementation that allowed for obtaining a graph as balanced as possible was studied, also focusing on the chosen layout type. Before proceeding with the algorithm, it was necessary to verify

that the graph could be identified as a UDG for some value of the radius. Subsequently, all the steps of the algorithm implemented on QuEra's analog Hamiltonian simulator were analyzed. Then, the process that mapped the graph to atomic coordinates was examined; the implemented adiabatic protocol was described, with the definition of the relevant parameters; the post-processing phase and the analysis of the obtained solutions were explained. Further considerations were made regarding the technical details of the hardware, although it was not possible to use it directly to solve this problem. Subsequently, the focus shifted to solving the MIS problem with D-Wave, emphasizing the QPU onto which the optimization problem was mapped. Following the same post-processing procedure, the results obtained for the superconducting qubit hardware were analyzed, and then considerations regarding the technical details of the QPU were made. The solutions obtained from the two technologies were compared, focusing particularly on the distribution of the MISs found in the solutions space, the number of qubits used to solve the problem, and the constraint regarding the definition of an initial UDG. This problem was further solved using a direct method and a classical heuristic, which confirmed the results obtained. Finally, given that there are multiple MISs equivalent in terms of optimization, to find the best financial solution, it would be useful to continue with a market analysis of the companies that are not part of the set common to all MVC solutions.

Chapter 7

Conclusions

The first part of this work focuses on the basics of Quantum Computing, including the definition of qubits and mathematical components, the algebraic formulation of quantum states, the definition of computational bases and their properties. Qubit states can also be represented on the Bloch sphere as vectors extending from the center to its surface, with orientation determined by the quantum state being represented. Significant emphasis has been placed on Hilbert spaces, where quantum states reside, defining fundamental elements for quantum computation such as Hermitian and unitary operators, along with their properties. Fundamental unitary operators for universal quantum computation have been defined through examples, including Pauli, Hadamard, and Phase operators. The tensor product has been defined as a useful operation for distinguishing pure and mixed states, involving the concept of density matrices. Measurement procedures have been further analyzed, crucial for interpreting results at the conclusion of a quantum computation. Although not directly applied in this work, quantum gates and circuits have been defined together with examples of operators defining principal gates like CNOT and SWAP. One of the distinctive phenomena of Quantum Computing, entanglement, has been examined with the definition of its main properties and an analysis of its applications. Lastly, an in-depth exploration of the time evolution of a closed quantum system has been provided, alongside the presentation of the Schrödinger equation and its analytical solution.

Subsequently, particular focus was given to the definition of NISQ, delving into the approaches most utilized by various Quantum Computing technologies available on the market, while exploring current challenges such as decoherence and the availability of a limited number of qubits. At the same time, it was highlighted the potential of Quantum Computing thanks to the Threshold Theorem. Further introduced were techniques of information encoding allowing the encoding of real data into qubits. The difference between the two main approaches

of Quantum Computing, Digital Quantum Computing and Analog Hamiltonian Simulation, was studied. While the former performs computation with Quantum Gates and Circuits, the latter simulates complex quantum systems by replicating their Hamiltonian dynamics described by the Schrödinger equation. Among the methodologies of Analog Hamiltonian Simulation, Adiabatic Quantum Computing solves optimization problems by evolving a quantum system's Hamiltonian from its initial state to its final state, following the Adiabatic Theorem, wherein if the Hamiltonian changes slowly enough, a quantum system remains in its instantaneous ground state. In this work, this approach was directly applied to solve a specific combinatorial optimization problem using two quantum annealers. The first, D-Wave, utilizes superconducting qubits, while the second, QuEra, is an emerging technology that exploits neutral atoms as computational units.

Afterwards, the cutting-edge technology primarily analyzed for this work, QuEra's quantum computer named Aquila, was studied in detail. In this regard, the neutral atom was defined and its possible states were analyzed, specifically identifying the Rydberg atom. The constituent components of Aquila were examined, as well as the techniques used for preparing atomic states and their dynamic evolution over time. Subsequently, the technique applied for measuring the final state was analyzed. A special focus was given to the Rydberg Blockade phenomenon, crucial for the chosen problem. In understanding the evolution of the system dynamics, the Rydberg Hamiltonian was defined along with its parameters, such as Rabi frequency, Detuning, and laser Phase. Finally, an explanation was provided regarding the potential error sources to which this technology may be susceptible.

Subsequently, the Maximum Independent Set (MIS) problem was defined, and through examples, various real-world applications were described, underlining that a graph can have multiple MIS and, for this reason, the solution to the MIS problem may not be unique. Some classical approaches were mentioned, covering both approximate and exact algorithms, emphasizing their theoretical advancements. Following this, the quantum approach used by QuEra to solve the MIS problem was described, highlighting how this technology naturally encodes the problem through the Rydberg Blockade phenomenon on a specific class of graphs known as Unit Disk Graphs. The Rydberg Blockade was described in terms of parameters defining the Rydberg Hamiltonian following the Adiabatic Protocol. Subsequently, formulations for solving the MIS problem on D-Wave's superconducting qubit technology were provided, including Ising and QUBO formulations. A special focus was given to the minor embedding, a procedure that facilitates the transition from the Ising/QUBO formulation to D-Wave's QPUs.

Finally, the context of the MIS problem applied in this work was outlined, focusing on identifying the group of companies with the highest financial risk within a large pool of related companies. The solution to this problem is represented by the

Minimum Vertex Cover of the reference graph, which connects to the Maximum Independent Set as it is exactly its complementary set of vertices. It is important to note that, at the time of writing, it was not possible to directly solve the problem using QuEra's QPU. Instead, a local Analog Hamiltonian simulator was used, which operates with only about ten atoms and thus limits the problem size. The parameters of the problem defining the reference Unit Disk Graph were then defined, followed by focusing on all the steps that allowed the implementation of the algorithm: verifying that the initial graph is a Unit Disk Graph; mapping the graph onto atomic coordinates to enable solution by the simulator; defining the Adiabatic Algorithm, and setting the parameters such as detuning, Rabi frequency, and laser phase. The solution was saved, followed by post-processing and analysis of the results. This problem was then solved using D-Wave's QPU, enabling a comparison between results obtained from two technologies based on the same resolution principle but using different approaches. The solutions obtained from both technologies show all possible feasible MISs of the reference graph, which were further compared with MISs obtained from the application of a classical heuristic, the simulated annealing. After evaluating the quality of the results, companies common to all solutions were identified, highlighting the need to conduct market analysis on non-included companies to determine the most advantageous financial solution.

Appendix A

Simulated Annealing

The Simulated annealing (SA) is a metaheuristic optimization technique presented for the first time in 1983 by *Kirkpatrick et.al* in [79] which is inspired by the *annealing process* exploited in metallurgy, with which a metal at a high temperature is slowly cooled. The peculiarity of this process is reflected in the microscopic world of particles where atoms in metals move rapidly at high temperatures and subsequently adopt a more ordered state as they cool. Following this scheme, simulated annealing tries to recreate this procedure by starting from a highly energetic state and then slowly moving to a minimum energy state, which codifies the solution of the optimization problem considered. The strength of this heuristic is the ability to not fall to local minima and therefore to provide the solution corresponding to the global optimum.

At each step, based on the current state of the system x , the SA decides *probabilistically* whether or not it is better to move to a state x^* being in the neighborhood of x . Since the objective is to obtain a solution with the lowest possible energy, it is appropriate to consider a probability distribution as a function of the system temperature. Unlike other procedures, this metaheuristic does not always accept better solutions and this prevents the algorithm from getting stuck in a local optimum.

A.1 The SA algorithm's steps

The procedure adopted by simulated annealing can be divided into the following different steps:

1. *Problem definition*
2. *Exploration of the solution space*

3. *Acceptance criterion*

4. *Temperature schedule*

which are analyzed in the next Sections.

A.1.1 Problem definition

First of all, it is necessary to define the Hamiltonian of the optimization problem $H_{problem}$, which represents the system's energy. After that, the algorithm proceeds with the definition of the initial temperature T_{max} and the initial solution, typically random. A high value of T_{max} allows the algorithm to explore a wide range of solutions, but at the same time, too high a value can make the convergence of the method hard. In our case the parameter β that controls the temperature is calibrated based on the bias values associated with the nodes of the optimization problem under consideration. Remember that, in an Ising formulation, biases are represented by the term h applied to each spin $s \in \{-1, 1\}$. The objective is to define a range of values for β that optimally explores the solution space.

A.1.2 Exploration of the solutions space

In this step, the algorithm explores the solution space by examining the neighborhood of the reference solution. Thus, considering a solution x , small perturbations yield a new solution x^* within the neighborhood of x .

A.1.3 Acceptance criterion

Considering a solution x and a solution into its neighborhood x^* , their energies are computed using the Hamiltonian of the problem:

$$E(x) = H_{problem}(x) \tag{A.1}$$

from which it can result that $E(x) > E(x^*)$ or $E(x) < E(x^*)$. At this point, the temperature T comes into play, defining the probability distribution $\mathbb{P}(x, x^*, T)$ by which one solution is chosen over another. The probability distribution considered is known as the *Boltzmann Probability Distribution* function:

$$\mathbb{P}(x, x^*, T) = e^{-\Delta E/kT} \tag{A.2}$$

where k is the *Boltzmann constant* and T is the temperature of the system in Kelvin. This function describes the probability of a system in thermal equilibrium

at temperature T to have energy E .

Defining

$$\Delta E = E(x^*) - E(x) \quad (\text{A.3})$$

as the difference between the new solution's and the reference solution's energy, the following *acceptance criterion* holds:

$$\begin{cases} \mathbb{P}(x, x^*, T) = 1 & \text{if } \Delta E < 0 \\ \mathbb{P}(x, x^*, T) = e^{-\Delta E/kT} & \text{otherwise} \end{cases} \quad (\text{A.4})$$

i.e. if the new solution has lower energy, it is accepted with probability 1; otherwise, it is accepted according to the Boltzmann probability distribution.

Note how, with this rule, solutions with higher energy are not discarded outright, allowing the algorithm to avoid getting stuck in local minima. This approach is known as *Metropolis-Hastings criterion*.

A.1.4 Temperature schedule

As mentioned earlier, an initial temperature T_{\max} is initialized so that the solution space can be explored as widely as possible. At each step, this temperature decreases gradually to a minimum value T_{\min} , following a scheduling that allows the algorithm to achieve convergence.

Initially, the algorithm accepts solutions that may be even worse than the current ones, but as the temperature decreases it becomes more selective, moving towards better solutions with higher probability. This procedure should guarantee the achievement of the global optimum.

The SA algorithm is carried out repeatedly following all the steps starting from a value $\beta_0 = 1/kT_{\max}$ until reaching a target $\beta_{\text{targ}} = 1/kT_{\min}$. The pseudocode that summarizes the previously listed steps is reported below.

Algorithm 1 Simulated Annealing pseudocode

Input: β_0, β_{targ}
Output: Solution with minimal energy x

- 1: $\beta \leftarrow \beta_0 = 1/kT_{max}$
- 2: $x \leftarrow$ *initial solution*
- 3: $E(x) \leftarrow$ *initial solution energy*
- 4: **while** $\beta < \beta_{targ}$ **do**
- 5: $x^* \leftarrow$ *new candidate solution*
- 6: $E(x^*) \leftarrow$ *new candidate energy*
- 7: $\Delta \leftarrow E = E^* - E$
- 8: **if** **Accept** ($\Delta E, \beta$) **then**
- 9: $x \leftarrow x^*$
- 10: $E(x) \leftarrow E(x^*)$
- 11: **end if**
- 12: $\beta \leftarrow$ *update β value*
- 13: **end while**
- 14: **return** x

Bibliography

- [1] Marcus C Braun. “A brief history of Quantum Computing”. In: *Medium* (2018). URL: <https://medium.com/@markus.c.braun/a-brief-history-of-quantum-computing-a5babea5d0bd>.
- [2] Peter W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. In: *SIAM J.Sci.Statist.Comput* 26 (1997). DOI: <https://doi.org/10.1137/S0097539795293172>.
- [3] Lov K. Grover. “A fast quantum mechanical algorithm for database search”. In: *Proceedings, 28th Annual ACM Symposium on the Theory of Computing (STOC)* (1996), pp. 2012–219. DOI: <https://doi.org/10.1145/237814.237866>.
- [4] Phillip Kaye, Raymond Laflamme, and Michele Mosca. “An Introduction to Quantum Computing”. In: *Oxford university press* (2006). DOI: <https://doi.org/10.1093/oso/9780198570004.001.0001>.
- [5] Giacomo Nannicini. “An Introduction to Quantum Computing, Without the Physics”. In: (2017). DOI: <https://doi.org/10.48550/arXiv.0708.0261>.
- [6] *On “Basics of quantum information”*. <https://learning.quantum.ibm.com/course/basics-of-quantum-information/single-systems/>.
- [7] Peter Wittek. “Quantum Machine Learning: What Quantum Computing Means to Data Mining”. In: *Academic Press* (2014).
- [8] Yazhen Wang. “Quantum Computation and Quantum Information”. In: *Statistical science* 27 (2012). DOI: <https://doi.org/10.1214/11-STS378>.
- [9] Maria Schuld and Francesco Petruccione. “Supervised Learning with Quantum Computers”. In: *Springer* (2018). DOI: <https://doi.org/10.1007/978-3-319-96424-9>.
- [10] Frank Leymann and Johanna Barzen. “The bitter truth about gate-based quantum algorithms in the NISQ era”. In: *Quantum Science and Technology* 5 (2020). DOI: <https://doi.org/10.1088/2058-9565/abae7d>.

-
- [11] David P. Di Vincenzo. “The Physical Implementation of Quantum Computation”. In: *Fortschritte der Physik: Progress of Physics* (2000).
- [12] Maximilian Schlosshauer. “Quantum Decoherence”. In: *Phys. Rep.* 831, 1–57 (2019). DOI: <https://doi.org/10.1016/j.physrep.2019.10.001>.
- [13] H. D. Zeh. “On the interpretation of measurement in quantum theory”. In: *Phys.* 1 69–76 1 (1970).
- [14] On “IBM releases 1,000+ qubit processor, roadmap to error correction”. <https://arstechnica.com/science/2023/12/ibm-adds-error-correction-to-updated-quantum-computing-roadmap/>.
- [15] H. L. Huang, Dachao Wu, Daojin Fan, and Xiaobo Zhu. “Superconducting Quantum Computing, A Review”. In: *Science China Information Sciences* 63, 180501 (2020). DOI: <https://doi.org/10.1007/s11432-020-2881-9>.
- [16] Yuan Xuand Ji Chu, Jiahao Yuan, Jiawei Qiu, Yuxuan Zhou, Libo Zhang, Xinsheng Tan, Yang Yu, Song Liu, Jian Li, Fei Yan, and Dapeng Yu. “High-Fidelity, High-Scalability Two-Qubit Gate Scheme for Superconducting Qubits”. In: *Academic Physical Society* (2020). DOI: <https://doi.org/10.1103/PhysRevLett.125.240503>.
- [17] Jonathan Wurtz, Fangli Liu, Phillip Weinberg, John Long, and Sheng-Tao Wang. “Aquila: QuEra’s 256-qubit neutral-atom quantum computer”. In: 1 (2023). DOI: <https://doi.org/10.48550/arXiv.2306.11727>.
- [18] Joshua W. Silverstone, Member, OSA and Damien Bonneau, Jeremy L. O’Brien, and Mark G. Thompson. “Silicon Quantum Photonics”. In: *Journal of selected topics in quantum electronics* (2017). DOI: <https://doi.org/10.1109/JSTQE.2016.2573218>.
- [19] Stephen C. Wein. “Simulating photon counting from dynamic quantum emitters by exploiting zero-photon measurements”. In: (2023). DOI: <https://arxiv.org/pdf/2307.16591>.
- [20] Tameem Albash and Daniel A. Lidar. “Adiabatic quantum computing”. In: *Rev. Mod. Phys.* 90 (2018). DOI: <https://doi.org/10.1103/RevModPhys.90.015002>.
- [21] On “What is Quantum Annealing?”. https://docs.dwavesys.com/docs/latest/c_gs_2.html/.
- [22] D. Jaksch, J.I. Cirac, P. Zoller, S.L. Rolston, R. Cote, and M.D. Lukin. “Fast quantum gates for neutral atoms”. In: (2000). DOI: <https://doi.org/10.1103/PhysRevLett.85.2208>.

-
- [23] M. D. Lukin, M. Fleischhauer, R. Cote, L. M. Duan, D. Jaksch, J. I. Cirac, and P. Zoller. “Dipole Blockade and Quantum Information Processing in Mesoscopic Atomic Ensembles”. In: (2001). DOI: <https://doi.org/10.1103/PhysRevLett.87.037901>.
- [24] M. Morgado and S. Whitlock. “Quantum simulation and computing with Rydberg-interacting qubits”. In: *AVS Quantum Sci.* 3, 023501 (2021). DOI: <https://doi.org/10.1116/5.0036562>.
- [25] J. Ignacio Cirac and Peter Zoller. “Goals and opportunities in quantum simulation”. In: *Nature Physics volume 8, pages 264–266* (2012). DOI: <https://doi.org/10.1038/nphys2275>.
- [26] Harry Levine, Alexander Keesling, Giulia Semeghini, Ahmed Omran, Tout T. Wang, Sepehr Ebadi, Hannes Bernien, Markus Greiner, Vladan Vuletić, Hannes Pichler, and Mikhail D. Lukin. “Parallel Implementation of High-Fidelity Multiqubit Gates with Neutral Atoms”. In: *Phys. Rev. Lett.* 123, 170503 (2019). DOI: <https://doi.org/10.1103/PhysRevLett.123.170503>.
- [27] Simon J. Evered, Dolev Bluvstein, Marcin Kalinowski, Sepehr Ebadi, Tom Manovitz, Hengyun Zhou, Sophie H. Li, Alexandra A. Geim, Tout T. Wang, Nishad Maskara, Harry Levine, Giulia Semeghini, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. “High-fidelity parallel entangling gates on a neutral atom quantum computer”. In: *Nature* 622, 268–272 (2023). DOI: <https://doi.org/10.1038/s41586-023-06481-y>.
- [28] Dolev Bluvstein, Simon J. Evered, Alexandra A. Geim, Sophie H. Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, J. Pablo Bonilla Ataides, Nishad Maskara, Iris Cong, Xun Gao, Pedro Sales Rodriguez and Thomas Karolyshyn, Giulia Semeghini, Michael J. Gullans, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. “Logical quantum processor based on reconfigurable atom arrays”. In: *Nature volume 626, pages 58–65* (2024). DOI: <https://doi.org/10.1038/s41586-023-06927-3>.
- [29] Xiangyu Gao, Jianzhong Li, and Dongjing Miao. “Dynamic Approximate Maximum Independent Set on Massive Graphs”. In: *IEEE 38th International Conference on Data Engineering (ICDE)* (2022). DOI: <https://doi.org/10.48550/arXiv.2009.11435>.
- [30] Jonathan Wurtz, Pedro L. S. Lopes, Christoph Gorgulla, Nathan Gemelke, and Alexander Keesling and Shengtao Wang. “Industry applications of neutral-atom quantum computing solving independent set problems”. In: (2022). DOI: <https://doi.org/10.48550/arXiv.2205.08500>.

-
- [31] M. Goldberg, D. Hollinger, and M. Magdon-Ismael. “Experimental Evaluation of the Greedy and Random Algorithms for Finding Independent Sets in Random Graphs”. In: (2005). Ed. by Nikolettseas and Sotiris E., pp. 513–523.
- [32] Bo Chen, Chris N. Potts, and Gerhard J. Woeginger. “A Review of Machine Scheduling: Complexity, Algorithms and Approximability”. In: *Handbook of Combinatorial Optimization* (1998), pp. 1493–1641. DOI: https://doi.org/10.1007/978-1-4613-0303-9_25.
- [33] Joaquín Cervera, Arjan J. van der Schaft, and Alfonso Baños. “On composition of Dirac structures and its implications for control by interconnection”. In: *Lecture Notes in Control and Information Sciences* 281 (2022), pp. 55–63. DOI: https://doi.org/10.1007/3-540-45802-6_5.
- [34] Dániel Marx. “Parameterized Complexity and Approximation Algorithms”. In: *The Computer Journal* 51 (2007), pp. 60–78. DOI: <https://doi.org/10.1093/comjnl/bxm048>.
- [35] R. M. Karp. “Reducibility Among Combinatorial Problems”. In: *Complexity of Computer Computations* (1972), pp. 84–103. DOI: https://doi.org/10.1007/978-1-4684-2001-2_9.
- [36] Tobias Achterberg, Robert E. Bixby, Zonghao Gu, Edward Rothberg, and Dieter Weninger. “Preprocessing and cutting planes with conflict graphsPre-solve Reductions in Mixed Integer Programming”. In: *INFORMS Journal on Computing* 32 (2019). DOI: <https://doi.org/10.1287/ijoc.2018.0857>.
- [37] Samuel Souza Brito and Haroldo Gambini Santos. “Preprocessing and cutting planes with conflict graphs”. In: *Haroldo Gambini Santos* 128 (2021). DOI: <https://doi.org/10.1016/j.cor.2020.105176>.
- [38] Uriel Feige and Joe Kilian. “Heuristics for Semirandom Graph Problems”. In: *Journal of Computer and System Sciences* 63 (2001), pp. 639–671. DOI: <https://doi.org/10.1006/jcss.2001.1773>.
- [39] Fedor V. Fomin and Dieter Kratsch. “Exact Exponential Algorithms”. In: *Springer Berlin Heidelberg* (2010). DOI: <https://doi.org/10.1007/978-3-642-16533-7>.
- [40] J.M. Robonson. “Algorithms for Maximum independent Sets”. In: *Journal of algorithms* 7, 425-440 (1986). DOI: [https://doi.org/10.1016/0196-6774\(86\)90032-5](https://doi.org/10.1016/0196-6774(86)90032-5).
- [41] Mingyu Xiao and Hiroshi Nagamochi. “Exact Algorithms for Maximum Independent Set”. In: *Information and Computation* 255(1):126-146 (2017). DOI: <https://doi.org/10.1016/j.ic.2017.06.001>.

-
- [42] J. Hastad. “Clique is hard to approximate within $n^{1-\epsilon}$ ”. In: *Proceedings of 37th Conference on Foundations of Computer Science* (1996). DOI: <https://doi.org/10.1109/SFCS.1996.548522>.
- [43] Uriel Feige. “Approximating Maximum Clique by Removing Subgraphs”. In: *SIAM Journal on Discrete Mathematics* 18 (2004). DOI: <https://doi.org/10.1137/S089548010240415X>.
- [44] Sebastian Lamm, Peter Sanders, Christian Schulz, Darren Strash, and Renato F. “Finding Near-Optimal Independent Sets at Scale”. In: *2016 Proceedings of the Eighteenth Workshop on Algorithm Engineering and Experiments* (2016), pp. 138–150. DOI: <https://doi.org/10.1137/1.9781611974317.12>.
- [45] Yu Liu, Jiaheng Lu, Hua Yang, Xiaokui Xiao, and Zhewei Wei. “Towards Maximum Independent Sets on Massive Graphs”. In: *Proceedings of the VLDB Endowment* 8 (2015), pp. 2122–2133. DOI: <https://doi.org/10.14778/2831360.2831366>.
- [46] Lijun Chang, Wei Li, and Wenjie Zhang. “Computing A Near-Maximum Independent Set in Linear Time by Reducing-Peeling”. In: *SIGMOD '17: Proceedings of the 2017 ACM International Conference on Management of Data* 8 (2017), pp. 1181–1196. DOI: <https://doi.org/10.1145/3035918.303593>.
- [47] Sinem Çınaroğlu and Sema Bodur. “A new hybrid approach based on genetic algorithm for minimum vertex cover”. In: *Innovations in Intelligent Systems and Applications (INISTA)* (2018), pp. 85–103. DOI: <https://doi.org/10.1109/INISTA.2018.8466307>.
- [48] Luzhi Wang, Shuli Hu, Mingyang Li, and Junping Zhou. “An Exact Algorithm for Minimum Vertex Cover Problem”. In: *Mathematics* 7 (2019), p. 603. DOI: <https://doi.org/10.3390/math7070603>.
- [49] Richard M. Karp. “Reducibility among Combinatorial Problems”. In: *Springer* (1972), pp. 85–103. DOI: https://doi.org/10.1007/978-1-4684-2001-2_9.
- [50] S. Khuri and T. Bäck. “Evolution strategies: An alternative evolutionary algorithm”. In: *Lecture Notes in Computer Science* 1063 (1995). DOI: https://doi.org/10.1007/3-540-61108-8_27.
- [51] Martin Pelikan, Rajiv Kalapala, and Alexander K. Hartmann. “Hybrid evolutionary algorithms on minimum vertex cover for random graphs”. In: *Proceedings of the 9th annual conference on Genetic and evolutionary computation* (2007), pp. 547–554. DOI: <https://doi.org/10.1145/1276958.1277073>.

-
- [52] Milanović Marija. “A Metaheuristic Approach to Solving the Generalized Vertex Cover Problem”. In: *Mathematica Balkanica New Series* (2010), pp. 267–273. ISSN: 0205-3217.
- [53] Drona Pratap Chandu. “A Parallel Genetic Algorithm for Generalized Vertex Cover Problem”. In: *International Journal of Computer Science and Information Technologies (IJCSIT)* (2014), pp. 7686–7689. DOI: <https://doi.org/10.48550/arXiv.1411.7612>.
- [54] Drona Pratap Chandu. “A hybrid heuristic for the minimum weight vertex cover problem”. In: *Asia-Pacific Journal of Operational Research* 23 (2006), pp. 271–285. DOI: <https://doi.org/10.1142/S0217595906000905>.
- [55] Selman Yakut, Furkan Öztemiz, and Ali Karci. “A new robust approach to solve minimum vertex cover problem: Malatya vertex-cover algorithm”. In: *Innovations in Intelligent Systems and Applications (INISTA)* 79 (2023), pp. 19746–19769. DOI: <https://doi.org/10.1007/s11227-023-05397-8>.
- [56] S. Vorobev, A. Kolosnitsyn, and I Minarchenko. “Determination of the Most Interconnected Sections of Main Gas Pipelines Using the Maximum Clique Method”. In: *Energies* 15 (2022). DOI: <https://doi.org/10.3390/en15020501>.
- [57] Victor Spirin and Leonid A Mirny. “Protein complexes and functional modules in molecular networks”. In: *Energies* 100 (2003), pp. 12123–12128. DOI: <https://doi.org/10.1073/pnas.2032324100>.
- [58] Sidong Liu, Yang Song, Fan Zhang, Dagan Feng, Michael Fulham, and Weidong Ca. “Clique Identification and Propagation for Multimodal Brain Tumor Image Segmentation”. In: *Brain Informatics and Health* 100 (2016), pp. 285–294. DOI: https://doi.org/10.1007/978-3-319-47103-7_28.
- [59] Sándor Szabó and Bogdan Zavalnij. “Decomposing clique search problems into smaller instances based on node and edge colorings”. In: *Discrete Applied Mathematics* 242 (2018), pp. 118–129. DOI: <https://doi.org/10.1016/j.dam.2018.01.006>.
- [60] Preprocessing and Probing Techniques for Mixed Integer Programming Problems. “Preprocessing and Probing Techniques for Mixed Integer Programming Problems”. In: *ORSA Journal on Computing* (1994). DOI: <https://doi.org/10.1287/ijoc.6.4.445>.
- [61] Manfred W. Padberg. “On the facial structure of set packing polyhedra”. In: *Mathematical Programming* 5 (1973), pp. 199–215. DOI: <https://doi.org/10.1007/BF01580121>.

-
- [62] Federico Della Croce and Roberto Tadei. “A multi-KP modeling for the maximum-clique problem”. In: *European Journal of Operational Research* 73 (1994), pp. 555–561. DOI: [https://doi.org/10.1016/0377-2217\(94\)90252-6](https://doi.org/10.1016/0377-2217(94)90252-6).
- [63] I. Douglas Moon and Sohail S. Chaudhry. “An Analysis of Network Location Problems with Distance Constraints”. In: *Management Science* 30 (1984). DOI: <https://doi.org/10.1287/mnsc.30.3.290>.
- [64] Mohammad Javad Naderi, Austin Buchanan, and Jose L. Walteros. “Worst-case analysis of clique MIPs”. In: *Mathematical Programming* 195 (2021), pp. 517–551. DOI: <https://doi.org/10.1007/s10107-021-01706-2>.
- [65] Chuan-Min Lee. “Algorithmic Aspects of Some Variations of Clique Transversal and Clique Independent Sets on Graphs”. In: *Algorithms* 14 (2021). DOI: <https://doi.org/10.3390/a14010022>.
- [66] Chuan-Min Lee. “Remarks on Parameterized Complexity of Variations of the Maximum-Clique Transversal Problem on Graphs”. In: *Symmetry* 14 (2022). DOI: <https://doi.org/10.3390/sym14040676>.
- [67] S. Ebadi et al. “Quantum Optimization of Maximum Independent Set using Rydberg Atom Arrays”. In: *Science* 376 (2022). DOI: <https://doi.org/10.1126/science.abo6587X>.
- [68] On “*Adiabatic Evolution*”. <https://queracomputing.github.io/Bloqade.jl/dev/tutorials/2.adiabatic/main/>.
- [69] On “*QPU topology*”. <https://docs.ocean.dwavesys.com/en/stable/concepts/topology.html/>.
- [70] Andrew Lucas. “Ising formulation of many NP problems”. In: *Frontiers in physics, article 5 2* (2014). DOI: <https://doi.org/10.3389/fphy.2014.00005>.
- [71] On “*Hello AHS: Run your first Analog Hamiltonian Simulation*”. <https://docs.aws.amazon.com/braket/latest/developerguide/braket-get-started-hello-ahs.html/>.
- [72] On “*D-Wave Leap*”. <https://cloud.dwavesys.com/leap/login/?next=/leap/>.
- [73] On “*Google Finance*”. <https://www.google.com/finance/>.
- [74] On “*The maximum independent set*”. <https://queracomputing.github.io/Bloqade.jl/dev/tutorials/5.MIS/main/>.
- [75] On “*AWS optimization demo*”. <https://github.com/QuEraComputing/QuEra-braket-examples/blob/main/OptimizationTutorial/>.

- [76] On "Hello AHS: Run your first Analog Hamiltonian Simulation". <https://docs.aws.amazon.com/braket/latest/developerguide/braket-get-started-hello-ahs.html/>.
- [77] On "QPU Access Time". https://docs.dwavesys.com/docs/latest/c_qpu_timing.html/.
- [78] Minh-Thi Nguyen, Jin-Guo Liu, Jonathan Wurtz, Mikhail D. Lukin, Sheng-Tao Wang, and Hannes Pichler. "Quantum optimization with arbitrary connectivity using Rydberg atom arrays". In: *PRX Quantum* 4, no. 1: 010316 (2023). DOI: <https://doi.org/10.1103/PRXQuantum.4.010316>.
- [79] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi Jr. "Optimization by Simulated Annealing". In: *Science* 220 (1983). DOI: <https://doi.org/10.1126/science.220.4598.671>.