Quantum many-body dynamics for combinatorial optimisation and machine learning

Constantin Dalyac

To cite this version:

Constantin Dalyac. Quantum many-body dynamics for combinatorial optimisation and machine learning. Artificial Intelligence [cs.AI]. Sorbonne Université, 2023. English. NNT : 2023SORUS275 . tel-04265956

HAL Id: tel-04265956
https://theses.hal.science/tel-04265956v1

SORBONNE UNIVERSITÉ - EDITE DE PARIS
Laboratoire d'Informatique de Sorbonne Université (LIP6)

Quantum many-body dynamics for combinatorial optimisation and machine learning

BY CONSTANTIN DALYAC

PHD THESIS IN COMPUTER SCIENCE

PhD defence held publicly in October 2023 with the following thesis committee:

- TREPS Nicolas, Professeur à Sorbonne Université, *Président du jury.*
- AYRAL Thomas, Research engineer, Eviden, *Rapporteur.*
- WALLDEN Petros, Associate Professor at University of Edinburgh, Scotland, UK. *Rapporteur.*
- DIAMANTI Eleni, Directrice de recherche au CNRS, LIP6, *Examinateur.*
- KERENEDIS Iordanis, Directeur de recherche au CNRS, IRIF, *Examinateur.*
- KASHEFI Elham, Directrice de Recherche HDR au CNRS, Sorbonne Université and Professor at University of Edinburgh, United Kingdom, *PhD supervisor.*
- BREDARIOL GRILO Alex, chercheur au CNRS, LIP6, *PhD co-supervisor.*
- HENRIET Loïc, Chief Technical Officer, Pasqal, *PhD co-supervisor.*

Keywords: quantum information, neutral atoms, combinatorial optimisation, quantum annealing, quantum machine learning

ABSTRACT

THE goal of this thesis is to explore and qualify the use of N-body quantum dynamics to solve hard industrial problems and machine learning tasks. As a collaboration between industrial and academic partners, this thesis explores the capabilities of a neutral atom device in tackling real-world problems.

First, we look at combinatorial optimisation problems and showcase how neutral atoms can naturally encode a famous combinatorial optimisation problem called the Maximum Independent Set on Unit-Disk graphs. These problems appear in industrial challenges such as Smart-Charging of electric vehicles. The goal is to understand why and how we can expect a quantum approach to solve this problem more efficiently than classical method and our proposed algorithms are tested on real hardware using a dataset from EDF, the French Electrical company. We furthermore explore the use of 3D neutral atoms to tackle problems that are out of reach of classical approximation methods. Finally, we try to improve our intuition on the types of instances for which a quantum approach can(not) yield better results than classical methods.

In the second part of this thesis, we explore the use of quantum dynamics in the field of machine learning. In addition of being a great chain of buzzwords, Quantum Machine Learning (QML) has been increasingly investigated in the past years. In this part, we propose and implement a quantum protocol for machine learning on datasets of graphs, and show promising results regarding the complexity of the associated feature space. Finally, we explore the expressivity of quantum machine learning models and showcase examples where classical methods can efficiently approximate quantum machine learning models.

DEDICATION AND ACKNOWLEDGEMENTS

SPRINKLED with moments of triumph and occasional bouts of frustration, pursuing a PhD can be quite an adventure. Although it is usually seen as a solitary exercise I had the chance to share this journey with remarkable people whom I am deeply grateful to acknowledge.

First and foremost, I would like to express my gratitude to my thesis director Elham Kashefi: thank you for your guidance over the years, your enthusiasm and your capacity to bring fantastic people around a common research topic. Thanks to you, I had the opportunity to spend entire days diving in the exciting realm of quantum computing with LIP6 and Edinburgh University scholars and I will keep fond memories of our Scottish trips.

I am also very grateful to Alex Bredariol Grilo for his kind and constant support: thank you for following my work with consistency and forcing me to write sound mathematical proofs. Your approach to problem solving is quite remarkable and your expertise in computational complexity theory helped me structure the goals of my research.

I cannot thank enough Loïc Henriet for the fantastic scientific and entrepreneurial adventure we shared. From back-of-the-envelope calculations at a bar in Barcelona to the first implementation of our quantum algorithm on a neutral atom device, I have lived a demanding but fabulous journey throughout this PhD. Thank you for sharing your passion for physics and your scientific integrity. As I learned from you, the best way to communicate ideas and proofs is by a succession of trivial implications, a step-by-step approach that I tried to apply as best as I could in this dissertation. The fact that you managed to closely supervise your three PhD students at Pasqal although your team grew from one to 66 employees in three years shows just how much you took care of us.

Lucas Leclerc and Slimane Thabet, my dear PhD co-fellows, I am so grateful for sharing all these scientific (and less scientific) moments together. It is a great honour and pleasure to work and go to conferences in exotic places with you both.

I would also like to thank all the people at Pasqal with whom I collaborated and shared scientific challenges, discussions or special moments. Louis-Paul Henry, your understanding of statistical physics is impressive and contagious[1]. Thank you for improving my research and for building such a nice team of Hamiltonians. Many thanks to Louis Vignoli (my Qbro), Chayma Bouazza, Henrique Silviero, Mauro d'Arcangelo, Matthieu Moreau, Anne-Claire Le

---

[1] I promise that one day I'll read your thesis.

Hénaff, Adrien Signoles, Lucas Béguin, Mourad Beji, Rein van den Eijnde, Wesley da Silva Coelho, Yassine Naghmouchi, Vincent Elfving and of course Georges-Olivier Reymond, Antoine Browaeys and Thierry Lahaye for co-founding this unique company.

None of this adventure would have been possible without Christophe Jurczak, who kindly introduced me to Pasqal and managed to develop an impressive quantum ecosystem in France, in Europe and across the ocean with Quantonation. Thank you also to Charles Beigbeder, Olivier Tonneau and Jean-Gabriel Boinot for sharing their office at Bourrienne: quantum computing goes well with Directoire-style.

Thank you to the whole LIP6 team and the people I met through them: Frédéric Grosshans, Eleni Diamanti, Damian Markham, Shane Mansfield and Harold Ollivier. A special thanks to Jonas Landman for building a great QML team with Slimane Thabet, Leo Monbrossou and Hela Mhiri.

Finally, as a PhD thesis finds a way to make itself quite at home in one's personal life, I'd like to thank and dedicate this thesis to my friends and family:

> *À mes parents chéris, merci d'avoir cultivé notre curiosité intellectuelle et de m'avoir poussé à donner le meilleur de moi-même,*
> *À mes frères adorés, merci pour votre émerveillement constant et votre intelligence, je suis tellement fier de vous.*
> *À ma garde rapprochée et mes amis, qui ont suivi avec intérêt (ou par politesse) mon sujet préféré: Alexandre, Max, Vincent, Anatole, Adrienne, Amaury, Stan, Philibert, Agathe, Raphaëlle, Lucas, Constance, Fanny, Aude-Marine, Théodore, Théophile, Jacob, Thomas, Orlando, Roméo, Marie-Astrid et Romain.*
> *Enfin, à Charlotte; tu es un soleil dans la vie de beaucoup de personnes mais particulièrement dans la mienne. La thèse est un projet merveilleux que je termine ici, mais je suis impatient de continuer le nôtre pour la vie.*

Les sciences ont deux extrémités qui se touchent. La première est la pure ignorance naturelle où se trouvent tous les hommes en naissant. L'autre extrémité est celle où arrivent les grandes âmes qui, ayant parcouru tout ce que les hommes peuvent savoir, trouvent qu'ils ne savent rien et se rencontrent en cette même ignorance d'où ils étaient partis.

Les pensées
Blaise Pascal

Table of Contents

INTRODUCTION

This thesis is the result of a collaboration between LIP6, a quantum information academic group and Pasqal, an experimental physics spin-off company building quantum computers made of arrays of neutral atoms. The aim of this collaboration is to conceive quantum protocols that run on a neutral atom hardware device and gaining intuition on where we can expect quantum computers to have an advantage over classical computers for practical and industrial problems. To do so requires some understanding of complexity theory, the main tool used in theoretical computer science, as well as a basic understanding of quantum mechanics and the inner functioning of neutral atom devices. Ideally, computational complexity theory provides us with an understanding of where we can anticipate quantum advantage, while a solid grasp of neutral atom technology determines the range of experiments we can conduct on the quantum processor. A crucial aim is to develop algorithms that are tailored to the device to maximise its usefulness, and study the proposed quantum algorithms with the metrics of computational complexity theory to compare them to their classical counterparts.

In Chapter 1, we detail the inner workings of a quantum processor made of neutral atoms. We explain the reason why these devices were created in the first place and describe the typical experimental setup. Then, we show how exciting atoms to specific states (called Rydberg states) can be used to simulate many-body quantum dynamics, a useful tool to explore exotic phases of matter that are out of reach for classical computers.

In Chapter 2, we translate two real-world problems related to the field of Smart-Charging to instances that can be mapped to a neutral atom quantum processor. We then estimate the quality of heuristic approaches in solving these **NP**-hard problems. Interestingly, the Maximum Independent Set on unit-disk graphs (UD-MIS) fits very naturally to the experimental platform. We therefore detail precisely how an implementation would work on the hardware and develop methods to embed the problem optimally and on large datasets.

In Chapter 3, we take a step back from hardware implementation and look at UD-MIS from the lens of computational complexity theory. More precisely, we unravel the complexity of MIS by using insights from computational approximation theory and statistical physics. We then build graph instances for which a phenomenon reminiscing Anderson localisation appears during quantum annealing, that can explain why it should also be hard to solve the problem quantumly. Furthermore, we develop an embedding scheme that allows us to embed more general graphs than UD graphs, an exciting perspective to tackle more difficult graphs with quantum schemes. Our embedding scheme has been realised experimentally using 3D arrays of atoms in collaboration with a Korean academic group.

In the following chapters, we look at the growing field of quantum machine learning (QML). After a gentle introduction to the field, we recall that Noisy Intermediate Scale Quantum devices (NISQ) can be used as learning models. We therefore describe in Chapter 4 how a neutral atom quantum processor can be used to create quantum feature maps for graph ML tasks. We implement a new quantum evolution kernel and show interesting results on the complexity of the resulting feature space.

In Chapter 5, we take a step back once again from hardware implementation on neutral atoms and try to understand the expressive power of using NISQ devices as learning models. In particular, we show that although some variational quantum circuits (VQC) can be hard to simulate classically, it is possible to approximate the resulting quantum learning model by using random Fourier features (RFF), a well-known tool used to approximate high-dimensional kernels in classical ML. Understanding the regimes for which RFF is not effective helps to build quantum learning models that are truly out of classical approximation schemes.

This thesis is based on the following papers:

- [DHJ^{+}21] Dalyac, C., Henriet, L., Jeandel, E., Lechner, W., Perdrix, S., Porcheron, M., Veshchezerova, M. (2021). *Qualifying quantum approaches for hard industrial optimization problems. A case study in the field of smart-charging of electric vehicles.* EPJ Quantum Technology, 8(1), 12.

- [DHK^{+}23] Dalyac C., Henry LP., Kim M., Ahn J., Henriet L. *Exploring the impact of graph locality for the resolution of MIS with neutral atom devices*, *In proceedings* (2023)

- [ADL^{+}23] Dalyac, C., Albrecht, B., Leclerc, L., Ortiz-Gutiérrez, L., Thabet, S., D'Arcangelo, M., ... Henriet, L. (2023). *Quantum feature maps for graph machine learning on a neutral atom quantum processor.* Physical Review A, 107(4), 042615.

- [LTD^{+}22] Landman, J., Thabet, S., Dalyac, C., Mhiri, H., Kashefi, E. (2022). *Classically Approximating Variational Quantum Machine Learning with Random Fourier Features.* ICLR 2023

I had the opportunity to participate to two other papers that are not detailed in this thesis:

- [HTDH21] Henry, L. P., Thabet, S., Dalyac, C., Henriet, L. (2021). *Quantum evolution kernel: Machine learning on graphs with programmable arrays of qubits.* Physical Review

A, 104(3), 032416.

- [SGD+22] Silvério, H., Grijalva, S., Dalyac, C., Leclerc, L., Karalekas, P. J., Shammah, N., ... Henriet, L. (2022). *Pulser: An open-source package for the design of pulse sequences in programmable neutral-atom arrays.* Quantum, 6, 629.

Finally, two parts of this thesis will be included in a paper in preparation:

- Dalyac C., Leclerc L., Henriet L., A.B. Grilo, E. Kashefi, *Experimental signature of quantum phase transitions in quantum annealing for the maximum independent set problem.*

Chapter 1

Rydberg atoms and quantum simulation of spin models

1.1 Understanding physics using simulation

In physics, equations of motions describe the behaviour of systems in time. While physicists have been able to find the equations that accurately describe the evolution of planets, liquids or atoms, solving them analytically (*i.e.* with pen and paper) for specific instances is usually intractable. In order to predict the dynamics of a satellite or forecast weather conditions, scientists have come up with numerical approximation methods to solve these equations. Some of these methods involve approximating the solution to the equation by breaking it down into small pieces and using iterative techniques to solve each piece.

In the XVIII$^{\text{th}}$ century, the Swiss mathematician Leonhard Euler developed a numerical method known today as the "Euler method" [Eul24]. In this method, time is cut into small intervals and a first-order method builds approximate solutions at each step. The accuracy of this method depends on the size of the interval time chosen. The Euler method and higher order ones was used during WWII to calculate the trajectories of shells, taking into account variables such as air density, temperature or wind and took between 20 to 40 hours to compute by hand [QZQ+18].

An alternative but creative technique was to solve the equations of motions using *analog machines*. An analog machine is a device that is subject to the same physical laws of the system of interest but over which an operator has a certain level of control. For example, Lord Kelvin invented in 1872 an analog machine that predicted the tides at a given location based on astronomical observations [Tho81]. The device consisted of a series of interconnected gears, levers, and rods that moved in a pattern that represented the ebb and flow of the tides. The gears could be adjusted to reflect the current conditions at the location being analysed, such as the water depth or the time and position of the Moon and Sun in the sky. This powerful device could predict the tide over a period of several hours or days. It worked so well that it was used in World War II to predict tides on the French beach for the D-Day invasion of Normandy [Ehr08]. However, it was expensive to build and the speed of calculation was limited. Analog computers were eventually superseded by digital computers thanks to the explosion in the number of transistors that allowed to apply approximation methods much faster: it took just under a minute to calculate the shell trajectories on the first big digital computer (ENIAC) [GH20]. Throughout the decades, digital computers became increasingly powerful and solved more and more complex equations: in 2019, scientists were able to simulate with high resolution the formation of thousands of galaxies in the universe in a volume of 30×10^6 parsecs [1]. This simulation ran on $24,000$ CPUs, each containing more than a billion transistors [PNS+19]. The result of the simulation required a storage memory of 846.5 terabytes (Tb).

In the realm of quantum physics, scientists are also very eager to simulate the dynamics of quantum systems. A well-known model that captures important properties of matter is the spin-$\frac{1}{2}$ model. In this model, the magnetic moments of atoms or molecules are represented as quantum mechanical spin-$\frac{1}{2}$ particles, which can have two possible states often referred to as "up" and "down". Spin models become particularly interesting when local interactions between spin display emergent macroscopic behaviour that cannot be explained by the behaviour of individual spins. Such behaviours include phase transitions or macroscopic properties such as magnetism. However, in order to witness such phenomenons, physicists sometimes have to calculate the dynamics of spin models for hundreds of spins. Unfortunately, determining the quantum dynamics of spins requires an enormous amount of computer memory. To convince ourselves, suppose that we want to predict the evolution of a single spin described by a wavefunction $|\psi(t)\rangle$. We apply a time-independent Hamiltonian H such that for any given time t, $|\psi(t)\rangle$ is described by the Schrodinger equation

$$i\hbar\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle. \tag{1.1}$$

The solution to the Schrodinger equation is given by $|\psi(t)\rangle = \exp\{-i\hbar Ht\}|\psi(0)\rangle$. To find the numerical value of $|\psi(t)\rangle$ requires to store the state $|\psi(0)\rangle$ in memory. A quantum state is described by a vector of complex numbers that grows exponentially with the size of the system. While a single spin is described by a vector of size 2, a system composed of N spin$-\frac{1}{2}$ requires a vector of size 2^N. Already with 50 spins, the state vector requires, with a float32 precision, 1230Tb of memory to be stored in a computer, which is already more memory than the gigantic astrophysical simulation of Ref. [PNS+19]! Now if you double the number of spins, the memory required blows up to 10^{30} bytes which represents a thousand times more memory than all humankind storage in 2023 [RRG18].

---

[1] A parsec corresponds to 3.26 lightyears, or 31 trillion kilometers.

This inescapable exponentiation of quantum systems convinced scientists like Richard Feynman that if we were to compute dynamics of quantum systems efficiently for large systems, then the computer itself should be quantum [Fey82]. In other words, because the quantum computer is also subject to quantum mechanics it can efficiently store quantum states. In parallel, experimental physicists managed to control physical systems of smaller and smaller sizes and in 2012 Serge Haroche was awarded the Nobel prize in physics for his pioneering work in controlling individual quantum systems [RBH01]. With his team, he developed techniques for manipulating and measuring the quantum states of single atoms and photons with high precision. The capacity to control a quantum state laid the basis for the realisation of an *analog* quantum computer: in similar fashion to Lord Kelvin's device, quantum physicists thought of using the dynamics of an ensemble of controllable atoms to mimic spin models that cannot be simulated on classical computers.

In the following section, we will describe in more detail how experimentalists have managed to simulate spin models in the past years by using arrays of neutral atoms trapped in optical tweezers. By manipulating up to hundreds of atoms, the dynamics that are observed are well beyond the reach of classical computing. We will describe in more detail how such a neutral-atom device works, and what types of models it can implement. While these experimental devices have been used for many-body physics exploration, this thesis explores the potential of quantum dynamics to solve other types of problems such as combinatorial optimisation and machine learning tasks.

1.2 Description of a neutral atom device for Hamiltonian simulation

In this section, we briefly describe the inner workings of a neutral-atom device. Detailed explanations of the physics involved can be found in [BVC$^+$13, Ldk18]. While operating on the microscopic scale, today's neutral-atom quantum computer is relatively big (though smaller than the ENIAC!). Made of 5000 different components, it measures 1.7m x 2m x 3m for a total weight close to 1500kg. It embarks five different laser sources to control the different steps of the quantum computation. In total, it uses 5kW of energy which is close to the energy consumption of an electric water heater, more than a hundred times lower than current supercomputers (1300kW) [AH23] and four times lower than quantum computers with dilution fridges like Google's Sycamore chip [AAB$^+$19].

The building block of the computation in a neutral atom device is the rubidium atom, commonly used in experimental physics. The two main challenges behind the experimental set-up are to:

(i) trap the atoms into a pre-defined layout
(ii) manipulate their inner state to control their quantum evolution.

Trapping individual atoms is already quite a *tour-de-force*. In order to so, experimentalists start with a block of rubidium metal which is heated with a diffusive oven to create an atomic vapour at a temperature of about $40°C$. Then, two successive magneto-optical traps are used to slow down and condense the vapour into a cloud of approximately 1mm^3, containing approximately 10^6 cold rubidium atoms. These atoms are trapped at the centre of the vacuum system showed in Figure 1.1, where the temperature is of the order of $10\mu K$ (a *million* times colder than outer-space). Then, a second trapping laser system isolates individual atoms within this ensemble. From a cloud of atoms, the goal is to trap individual atoms. It is possible to create a micrometer-sized optical trap by focusing a *trap laser* across a high numerical aperture lens situated inside the vacuum chamber. One atom from the cloud therefore gets locked in the trap and because of the extremely small trapping volume, only one atom can be loaded at a time: if a second atom enters the trap, the collision between the atoms will eject both of them [SRPG01]. This ensures that each trap, called an *optical tweezers*, contains at most a single atom. This stochastic optical tweezers has a loading efficiency of about 50%. Interestingly, a single trap can be "cloned" into hundreds of them by appropriately altering the spatial phase of the *trap laser* before the lens[Lab16]. The laser phase is modified before the lens using a Spatial Light Modulator (SLM) as shown in Figure 1.1. The laser consequently diffracts after the lens into many spots, creating layouts of arbitrary traps in 2D and in 3D. The number of traps that can be created is only limited by the optical power of the laser, and it is estimated that with current lasers one can create layouts with up to $10,000$ traps [GOS22]. Because each trap has a 50% chance of being loaded, the layout is loaded arbitrarily as shown in Figure 1.2. The idea is therefore to create $2n$ traps, obtain n trapped atoms located arbitrarily and re-arrange the atoms one-by-one into the desired layout. Moving the atoms from a trap to another trap is done with a mobile optical tweezer using a 2D Acoustico-Optic Deflector (2D-AOD in Figure 1.1). This step is done for each atom in under a millisecond. Once all the atoms are correctly assembled, a final fluorescence image is taken to confirm the successful preparation of the layout: the quantum dynamics part can now start.

The micro-traps are separated by a few micrometers ($5\mu m$) which means that the atoms are at a distance a thousand times bigger than their usual interaction length. In order to have the atoms interact with each other, the atoms can be excited to Rydberg states, which are states with a large principal quantum numbers n [ŠA18]. Precisely, the rubidium atom has a single electron in its valence orbital that can be prepared in a highly excited electronic state using a multi-photon excitation scheme. This highly excited state, called a Rydberg state, is interesting for two reasons. First, the electron is very far away from the nucleus on average when it is excited to a Rydberg state, typically 100nm which represents a thousand times the typical atomic radius. This "oversize" property allows the atom to interact with other atoms that are excited to Rydberg states over distances of dozens of micrometers, much bigger than the distance separating the micro-traps [BBL16a]. Furthermore, the lifetime of these Rydberg states is proportional to n^3 which means that it can reach lifetimes of hundreds of μs for $n = 60$ [BL20], much longer than a typical computation cycle. In other words, the Rydberg state is very stable and can be chosen as a reliable energy level thanks to its long lifetime and its capacity to interact over distances that are longer than the typical micro-traps separation length. Next, we will explain how these interactions between Rydberg levels naturally encode spin-$\frac{1}{2}$ models.

The interaction between a pair of Rydberg atoms at a distance R is dominated by an electronic dipole-dipole Hamiltonian when R is much bigger than the size of the electronic wavefunction. This interaction can change depending on the Rydberg levels selected, therefore yielding different spin Hamiltonians. First, let's suppose that the two atoms are prepared in the same Rydberg state. In that case, the dipole-dipole interaction leads to a van der Waals interaction scaling as C_6/R^6. The C_6 term depends on the Rydberg level n and is proportional to n^{11}. For a system of N atoms, the effects of van der Waals interactions are pairwise additive and the interaction Hamiltonian is

$$H_{\mathrm{vdW}} = \sum_{i<j} \frac{C_6}{R_{ij}^6} \hat{n}_i \hat{n}_j, \tag{1.2}$$

where i, j are the indices of the spins and $\hat{n}_i = |r\rangle\langle r|_i$ is the projector onto the Rydberg state of atom i. In the language of spin-$\frac{1}{2}$ models and taking $|g\rangle = |{\downarrow}\rangle$ and $|r\rangle = |{\uparrow}\rangle$, the projector can be understood as $\hat{n}_i = (\mathbb{I} + \hat{\sigma}_z^i/2)$. Another setting is to use two *different* Rydberg levels $|nS\rangle$ and $|nP\rangle$ that also have a dipole interaction but of different nature. In this case, the interaction between the two excited states corresponds to an excitation exchange, and if one maps $|nS\rangle = |{\downarrow}\rangle$ and $|nP\rangle = |{\uparrow}\rangle$, then the effective Hamiltonian is called the XY Hamiltonian and on N atoms reads

$$H_{XY} = \sum_{i<j} \frac{C_3}{R_{ij}^3} \left(\hat{\sigma}_i^+ \hat{\sigma}_j^- + \hat{\sigma}_j^+ \hat{\sigma}_i^- \right), \tag{1.3}$$

where $\hat{\sigma}^\pm = \hat{\sigma}^x \pm i\hat{\sigma}^y$.

The XY Hamiltonian is a very interesting example of a flip-flop interaction; it drives the exchange of the spin state between two neighbouring particles. It has been put forward as a possible mechanism of photosynthesis in plants [Col13]. In this model, the photons drop energy in the chloroplasts that has to be transferred to the centre of the cell where it is transformed into chemical energy using XY-Hamiltonian-like dynamics. To summarise, it is possible to prepare different interaction Hamiltonians by selecting the appropriate Rydberg levels. Recently, experimentalists even realised a XXZ Hamiltonian using microwave engineering [SWB$^+$22]. Furthermore, if we add a global coherent laser driving the transition $|{\downarrow}\rangle$ and $|{\uparrow}\rangle$ with a Rabi frequency Ω and a detuning δ, the total Hamiltonian reads, for the Ising model,

$$H_{\mathrm{Ising}} = \frac{\hbar\Omega}{2} \sum_{i=1}^{N} \hat{\sigma}_i^x - \hbar\delta \sum_{i=1}^{N} \hat{n}_i + \sum_{i<j} \frac{C_6}{R_{ij}^6} \hat{n}_i \hat{n}_j, \tag{1.4}$$

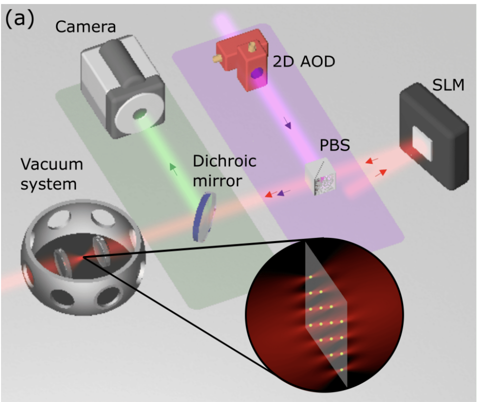


Figure 1.1: **Experimental set-up of a neutral atom device** reproduced from Ref. [HBS+20]

and for the XY model,

$$H_{\mathrm{XY}} = \frac{\hbar\Omega}{2}\sum_{i=1}^{N}\hat{\sigma}_i^x - \hbar\delta\sum_{i=1}^{N}\hat{n}_i + \sum_{i<j}\frac{C_3}{R_{ij}^3}\left(\hat{\sigma}_i^+\hat{\sigma}_j^- + \hat{\sigma}_j^+\hat{\sigma}_i^-\right). \tag{1.5}$$

Once the energy levels encoding the two-level system have been selected, the type of interaction is fixed accordingly. The idea is then to evolve the system of atoms in time and observe the dynamics. The parameters Ω, δ can be adiabatically modified in time as $\Omega(t), \delta(t)$ to prepare the groundstate of a particular Hamiltonian of interest [SZF+15]. They can also be suddenly changed to create quantum *quenched* dynamics [Mit18], an active field of research in which scientists try to understand intriguing phenomenons like many-body localisation [And58]. Experimentalists can therefore tweak the different parameters to observe many different many-body dynamics. Further degrees of freedom include changing the layout of the atoms before applying the coherent laser driving. For example, changing the layout of the atoms from a square to a triangular grid can induce frustration and create exotic dynamics in the case of antiferromagnetic interactions [Hen13]. Recent experiments have even demonstrated the capacity to move the atoms coherently in 2D during the coherent laser driving which modifies the interactions dynamically [BLS+22]: in other words, R_{ij} can evolve in time as $R_{ij}(t)$. Finally, several groups around the world have managed individual control over the atoms, in addition to the global coherent laser driving [KZI+10, GOS22, MHC+22]. This adds another degree of freedom and enables to apply local operations, opening the path to realising quantum digital gates for quantum computing [LKS+19].

While neutral atom arrays are primarily focused on simulating many body dynamics, there exists a bridge between them and more general computational problems. We will explore this in the following section.

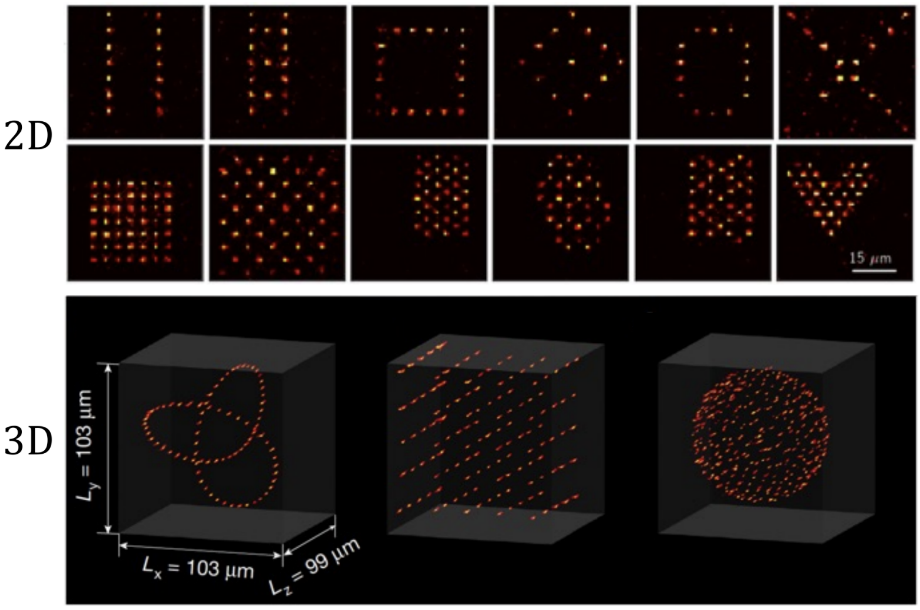


Figure 1.2: **Trapping of the atoms in 2D and 3D** reproduced from Ref. [BLL+18].

1.3 Spin models also map combinatorial problems

The essence of combinatorial problems lies in making choices from a set of discrete elements in order to optimise a given objective, while satisfying a set of constraints. Usually, the number of discrete elements to be chosen from grows exponentially with the size of the problem, making an exhaustive search for the optimal solution impractical. Let us illustrate this concept with a simple example[2]. Suppose that you are planning a dinner for a group of friends but you know that some people do not get along. You can represent this social network with a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ are the vertices that represent your friends and an edge $e = (i, j) \in \mathcal{E}$ between two vertices i and j translates that these two people do not get along very well. Willing for a calm and enjoyable night, you decide to invite the maximum number of people that get along with each other: in your graph, this corresponds to finding a maximum subset of the nodes that are not connected to each other. On small graphs this can be done by hand but when the number of friends n grows, you have to use another technique. A naive and inefficient approach is to check all subsets of the n guests and choose the biggest one for which everyone gets well together. In that case, the number of operations is 2^n, which scales exponentially with the number of guests: it would take a thousand years to solve this problem on the best supercomputer today, just with $n = 70$ guests (it would require $2^{70} \approx 10^{21}$ steps). Surprisingly, no general method has been found to solve this problem more efficiently than this naive approach. For this reason this problem, called the *Maximum Independent Set* (MIS), has

---

[2] This example is taken from Arora and Barak [AB09].

been cast to the **NP**-hard (nondeterministic polynomial time) complexity class, which implies that there is no known polynomial-time algorithm that can solve the problem exactly on all instances [GJ78]. Furthermore, one can also associate a cost function to the MIS problem. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an independent set is defined as a subset $\mathcal{S}$ of the vertices such that no two vertices of $\mathcal{S}$ share an edge in $\mathcal{G}$. Mathematically, $\mathcal{S}$ is an independent set of $\mathcal{G}$ if and only if

$$\mathcal{S} \subseteq \mathcal{V} \quad \text{s.t.} \quad \forall (x, y) \in \mathcal{S}^2, (x, y) \notin \mathcal{E}. \tag{1.6}$$

A maximum independent set $\mathcal{S}^*$ corresponds to an independent set of maximum cardinality. Any possible solution to this problem consists in separating the vertices of $\mathcal{G}$ into two distinct classes: an independent one and the others. We attribute a status z to each vertex, where $z_i = 1$ if vertex i belongs to the independent set, and $z_i = 0$ otherwise. The Maximum Independent Set corresponds to the minima of the following cost function:

$$C(z_1, \dots, z_n) = -\sum_{i=1}^{n} z_i + U \sum_{\langle i,j \rangle} z_i z_j, \tag{1.7}$$

where $U \gg \Delta(\mathcal{G})$, where $\Delta(\mathcal{G})$ is the degree of the vertex with maximum degree, $\langle i, j \rangle$ represents nodes in $\mathcal{E}$ (the edges), and $n = |\mathcal{V}|$. This cost function favours having a maximal number of atoms in the 1 state, but the fact that $U \gg 1$ strongly penalises two vertices in state 1 that share an edge. In mathematics, this type of cost function is called a Quadratic Unconstrained Binary Optimisation (QUBO) formulation [BH91]. QUBOs are ubiquitous in operational research and represent many optimisation problems such as network flows, scheduling, max-cut, max-clique or vertex cover [KGAR04]. Given binary variables z_i, QUBOs are written as the following general problem

$$\text{Minimise} \quad \sum_{i=1}^{n} q_{ii} z_i + \sum_{i,j=1}^{n} q_{ij} z_i z_j, \tag{1.8}$$

where q_{ii} represents the weight on the node i and q_{ij} represent the weight of the constraints between nodes i and j. Interestingly, this formulation is exactly the same as classical Ising models of spins. Indeed, a classical Ising model can be written as a quadratic function of a set of n spin-$\frac{1}{2}$ as

$$H(s_1, \dots, s_n) = \sum_{i=1}^{n} h_{ii} s_i + \sum_{i,j=1}^{n} J_{ij} s_i s_j. \tag{1.9}$$

This remarkable parallel between QUBOs and spin models opened a fertile area of research where methods from statistical physics such as *simulated annealing* have been used to solve difficult QUBO instances [KGJV83, MM09] and conversely, computational complexity theory have given hardness results on solving problems such as polymer protein folding [BL98]. In 2014, all of Karp's 21 **NP**-complete problems where explicitly transformed into an equivalent Ising spin formulation [Luc14]. Later, the authors of the pioneer publication [PWZ+18] noticed that the Hamiltonian of interacting Rydberg atoms naturally realises the QUBO formulation of an interesting combinatorial problem. We build upon these results to propose in the following chapter of this thesis the embedding of two practical industrial problems related to Smart-Charging of electric vehicles.

Part I

Combinatorial optimisation with Rydberg atoms

Chapter 2

Quantum approaches for industrial problems

In this chapter we develop quantum approaches to solve two industrial problems drawn from the growing sector of smart-charging of electrical vehicles (EVs). In section 2.1 the problems are explicitly translated into QUBO instances that correspond to the max-k-cut and the Maximum Independent Set (MIS) respectively. In section 2.2 we encode these high-dimensional discrete problems in the Hilbert space spanned by the neutral atom device's many-body quantum system and evaluate the performance of the Quantum Approximate Optimisation Algorithm (QAOA) [FGG14, PSTZ+18]. These numerical results suppose a perfect embedding of the problems and a noiseless device, hence we explore in section 2.4 the practical issues faced when trying to embed these industrial problems with available neutral atom devices. We provide details on the noise model of a neutral atom device and describe what type of pulses we can run naturally on it. The QAA approach is more natural regarding the hardware constraints and noisy simulations give us a precise idea of the results that we can expect on the device. At the time this chapter is written, the instances are expected to run on hardware in the following weeks. This first part of this chapter is based on the paper

- [DHJ+21] Dalyac, C., Henriet, L., Jeandel, E., Lechner, W., Perdrix, S., Porcheron, M., Veshchezerova, M. (2021). *Qualifying quantum approaches for hard industrial optimization problems. A case study in the field of smart-charging of electric vehicles.* EPJ Quantum Technology, 8(1), 12.

and the second part will be summarised in a paper in preparation.

2.1 Modelling two smart-charging tasks as NP-hard problems

Smart-charging is an umbrella term that encompasses all optimisation problems linked to the charging of electric vehicles (EVs). The recent surge in the number of EVs presents both new challenges and additional possibilities for electricity management. On the one hand, new issues such as charging task allocation, scheduling, and cost optimisation arise because of the charging times of these vehicles and their unpredictable load on the electrical system [RB20]. On the other hand, vehicle batteries could be used both as energy storage and power supply devices, a technique called *vehicle-to-grid* (V2G) which would improve significantly the flexibility of the electric system and reduce the use of fossil fuels during high-peak demand [GSS21].

Many difficult problems lie ahead of this scheme, related to the optimal management of the electric system in terms of cost while satisfying various technical constraints. These include among others the modulation of electricity demand taking into account the potentially high demand specific to EV loads: a 2021 study showed that an electric car used 32% of the total electric consumption of a household [DZAG21]. Another challenge is the availability of buffer required to guarantee the frequency stability of the grid when charging EVs [QZB10]. Interestingly, these problems translate to large sized combinatorial optimisation problems represented in Figure 2.1.

Vocabulary and model hypotheses

Both problems will be tackled under the following assumptions:

1. A load station is made up of several charging points, each of them loading at most a single electric vehicle (EV) at a given time step;
2. The charging points are *parallel identical* machines that supply the same power. The charging time of a given EV is thus independent of the charging point it is scheduled on;
3. We consider neither additional job characteristics and constraints (release/due dates, charging profile imposed by the battery state) nor global resource constraints on the load station (maximal power deliverable at a given time step);
4. Preemption is not allowed: a load task cannot be interrupted to be resumed later, on the same charging point or another one.

2.1.1 (SC1) The minimisation of total weighted load completion time is a max-k-cut instance

We consider $J = \{1, \ldots, n\}$ charging jobs of n EVs with duration $T = \{t_1, \ldots, t_n\}$ to be scheduled on a set $I = \{1, \ldots, k\}$ of k charging points. An integer weight $w_j > 0$ is associated to each job j that measures its importance: we might want to prioritise for example the charge of emergency vehicles. The time at which a load j ends is called the *completion time* and is noted $\mathcal{C}_j$. The goal is to minimise the weighted total time of completion of the charges

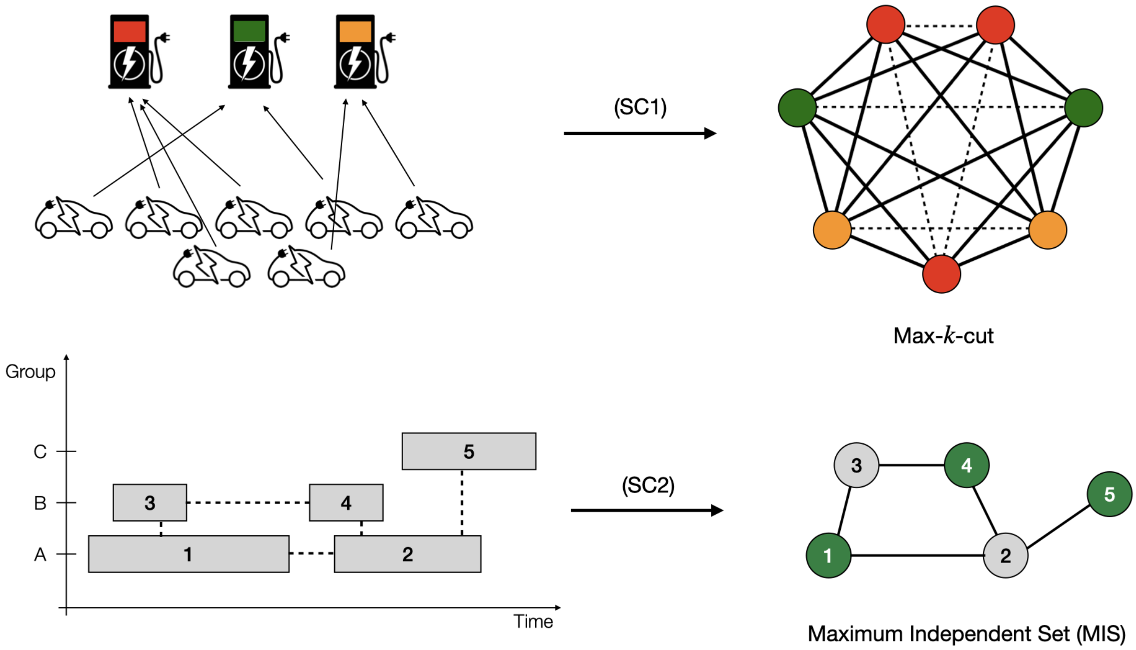


Figure 2.1: **Smart-charging problems and their modelling as NP-hard problems.** The first problem looks at the distribution of n vehicles when k charging stations are available. This problem (SC1) translates to a max-k-cut instance on a complete graph of size n. The second problem is related to the scheduling of n charging tasks under group and time constraints. In this case, the problem (SC2) translates to finding the MIS of a graph of size n.

$$\sum_{j\in J} w_j \mathcal{C}_j, \tag{2.1}$$

The problem (SC1) associated to minimising equation (2.1) is a classical scheduling problem known to be NP-hard in the general case [GLLRK79]. It can be solved in polynomial time if $k = 1$ as the optimal scheduling is given by executing them in the decreasing order defined by the ratio w_j/t_j (a rule known as the *Smith Rule* [Smi56]). Therefore the only problem in constructing good schedules for instances with k machines is to assign the jobs appropriately to each machine. If the number of machines k is fixed, (SC1) can be solved by pseudo-polynomial algorithms[1], typically based on dynamic programming and in the case where m is not fixed (SC1) is NP-hard [GLLRK79].

We now translate this problem into a max-k-cut instance where the goal is to divide the vertices of a graph into k disjoint sets to maximise the number of (weighted) edges that are cut or crossed between these sets. First, we can decompose the weighted total time of completion as

$$\sum_{j\in J} w_j \mathcal{C}_j = \sum_{j\in J} w_j (t_j + \mathcal{W}_j), \tag{2.2}$$

---

[1] An algorithm is said to be *pseudo-polynomial* if it is polynomial in the numeric values of its data, but super-polynomial in the length of their binary encoding.

where $\mathcal{W}_j$ is the waiting time before completing task j. The waiting time $\mathcal{W}_j$ decomposes as

$$\mathcal{W}_j = \sum_{i \prec j} t_i, \tag{2.3}$$

where i are all jobs scheduled before job j on the same machine. We can re-write equation (2.2) as

$$\sum_{j \in J} w_j \mathcal{C}_j = \sum_{j \in J} w_j t_j + \sum_{j \in J} \sum_{i \prec j} w_j t_i. \tag{2.4}$$

Recall that the ordering $\prec$ is given by the Smith rule, which runs job i before job j if $w_i/t_i < w_j/t_j$, or equivalently $w_i t_j < w_j t_i$. Therefore given a partition of the jobs in k groups $\{\mathcal{V}_1, \dots, \mathcal{V}_k\}$, the weighted total time of completion can be written as

$$\begin{aligned} \sum_{j \in J} w_j \mathcal{C}_j &= \sum_{j \in J} w_j t_j + \sum_{m=1}^{k} \sum_{u,v \in \mathcal{V}_m} \min\{w_u t_v, w_v t_u\}, && (2.5) \\ &= \sum_{j \in J} w_j t_j + \sum_{u,v \in \mathcal{V}} e_{uv}(1 - \delta_{u,v}), && (2.6) \end{aligned}$$

where $e_{uv} = \min\{w_u t_v, w_v t_u\}$ and $\delta_{u,v} = 1$ if u and v are in the same subset and $\delta_{u,v} = 0$ otherwise. Because the first term of equation (2.6) is constant, the minimisation of the weighted total time of completion corresponds to minimising the second term which in turn corresponds to maximising the k-cut of graph $\mathcal{G}$ [Sku98, YY03].

QUBO formulation of max-k-cut

If we let z_u be the index of the machine assigned to job u, the cost function C associated to the maximal-k-cut reads

$$C(z_1, \dots, z_n) = \sum_{u \neq v} e_{uv}(1 - \delta_{z_u, z_v}), \tag{2.7}$$

where δ is the Dirac function equal to 1 if $z_u = z_v$ and 0 otherwise. In the case where $k = 2$, the problem reduces to $z_i \in \{-1, 1\}$ and eq.(2.7) can be expressed as a QUBO instance:

$$C(z_1, \dots, z_n) = \sum_{u \neq v} e_{uv} \frac{1 - z_u z_v}{2}. \tag{2.8}$$

Without loss of generality if $k = 2^l$ where $l \in \mathbb{N}$, then each k is uniquely identified with a binary $\{\pm 1\}^l$ string $z_1 \dots z_l$ and the general max-k-cut reads

$$C(\mathbf{z}_1, \dots, \mathbf{z}_n) = \sum_{u \neq v} e_{uv} \prod_{k=1}^{l} \frac{1 - z_u^{(k)} z_v^{(k)}}{2}. \tag{2.9}$$

where $\mathbf{z}_i = (z_i^{(1)}, \dots, z_i^{(l)})$. This cost function can be encoded in a quantum operator $\hat{C}$ acting on $n \times l$ qubits. In this encoding each vertex of $\mathcal{V}$ is associated to a set of l qubits $\mathbf{z} = \{z^{(1)}, \dots, z^{(l)}\}$

that will indicate the colour of the vertex. Note that we need $nl \approx n \ln k$ qubits to encode the entire colouring of the graph. In the specific case where $k = 2^l$, the computational basis that spans the Hilbert space of our platform corresponds exactly to all the 2^{nl} possible colourings of the graph. A natural extension for all natural k would be to use qudits instead of qubits [BKKT20].

We can build a cost operator $\hat{C}$ that is diagonal in the computational basis, such that $\hat{C}|\mathbf{z}_1 \dots \mathbf{z}_n\rangle = C(\mathbf{z}_1 \dots \mathbf{z}_n)|\mathbf{z}_1 \dots \mathbf{z}_n\rangle$, by writing :

$$\hat{C} = - \sum_{1 \leq u < v \leq n} e_{uv} \left(1 - \frac{1 + \hat{\sigma}_u^{(1)} \hat{\sigma}_v^{(1)}}{2} \dots \frac{1 + \hat{\sigma}_u^{(l)} \hat{\sigma}_v^{(l)}}{2} \right), \tag{2.10}$$

where $\hat{\sigma}_u^{(i)}$ corresponds to a Pauli-Z matrix $\hat{\sigma}_z$ acting on the atom $i \in \{1, \dots, l\}$ associated to vertex u. The operator $\hat{C}$ in equation (2.10) can be decomposed as a sum of operators with $\{2, 4, \dots, 2l\}$-body interaction terms. Interaction terms involving more than 2-body operators are not directly implementable on most quantum computing platforms which only support 2-qubit gates. Instead, they can be decomposed as sums of two-body terms, which can be realized with CNOT gates. An example of the resulting circuit for an edge is shown in Figure 2.2.

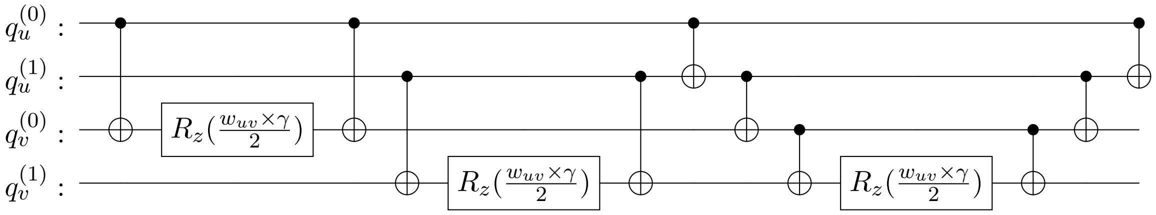


Figure 2.2: A quantum circuit implementing a term $e^{-i\gamma C_{u,v}}$ corresponding to the edge $\langle u, v \rangle$ in binary encoding for $k = 4$. Here, $R_z(.)$ is a single-qubit rotation along the z-axis.

Because for the instances of max-k-cut the graphs are complete (*i.e.* all nodes are connected), the cost operator in Eq. (2.10) involves coupling terms between all qubits. However, all quantum chips have a finite connectivity in practice. The physical realization of the desired terms between remote qubits thus requires the introduction of a large number of SWAP gates, that are detrimental to the performance of the procedure. In section 2.4 we introduce a hardware-efficient implementation of max-k-cut on Rydberg atom arrays which minimizes this overhead.

2.1.2 (SC2) The optimal scheduling of load time intervals within groups is an Maximum Independent Set instance

We now consider the following problem: given a set of load tasks represented as *intervals* on a timeline, such that each of them belongs to a specific *group*, for example distinct vehicle fleets of a company, select a subset of these loads which

(i) maximises the number of non-overlapping tasks
(ii) ensures that at most one load in each group is completed.

The goal is here to both minimise the completion time of the selected loads and to guarantee that no group will be over-represented in the schedule.

This problem belongs to the class of *Interval Scheduling* problems [KLPS07]. More precisely, it is a *Group Interval Scheduling* problem. It can be restricted without loss of generality to the case where all the groups contain the same number of tasks k and is **NP**-hard for $k \geq 3$ [Spi99].

Let $I = \{(s_1, e_1) \dots (s_n, e_n)\}$, be the set of intervals representing load job starting and ending dates, and $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be the graph whose vertices in $\mathcal{V}$ correspond to intervals in I. All intervals are also assigned to a group G. An edge (i, j) in $\mathcal{E}$ exists if interval i and j overlap or if i and j belong to the same group G. A set $\mathcal{S}$ of vertices is called an independent set when no two elements of $\mathcal{S}$ are adjacent in the graph. An independent set of $\mathcal{G}$ represents a feasible solution to the problem, and its Maximum Independent Set (MIS) is the optimal one[2].

Following our protocol for quantum algorithm qualification on neutral atom devices, we will limit ourselves to specific instances of (SC2) that can be formulated as MIS on two-dimensional *Unit-Disk* (UD) graphs. These geometrical graph are graphs in which two vertices are coupled by an edge if the distance between them is below a threshold value. This choice is motivated by the fact that a neutral atom quantum processor is particularly well suited to natively implement the MIS on Unit-Disk graphs [PSTZ+18]. Of course, this protocol requires to transform (SC2) graphs to Unit-Disk graphs, a procedure presented in the appendix A.1.

2.1.2.1 Rydberg blockade and independent sets

When looking for the MIS of a graph G, we separate the vertices into two distinct classes: an independent one and the others. We can attribute a status z to each vertex, where $z_i = 1$ if vertex i is attributed to the independent set, and $z_i = 0$ otherwise. The Maximum Independent Set corresponds to the minima of the following QUBO cost function:

$$C(z_1, \dots, z_n) = -\sum_{i=1}^{n} z_i + U \sum_{(i,j)\in\mathcal{E}} z_i z_j, \tag{2.11}$$

where $U \gg \Delta(G)$ and $\Delta(G)$ is the degree of the vertex with maximum degree. In this cost function, we want to promote a maximal number of atoms to the 1 state, but the fact that $U \gg 1$ strongly penalises two adjacent vertices in state 1. The minimum of $C(z_1, \dots, z_n)$ therefore corresponds to the maximum independent set of the graph.

---

[2] The above formulation supposes that a *starting date* is assigned to each load task, in order to represent it as an interval on the time line. In a real smart-charging management system, such dates could be fixed by the users of the vehicles, imposed by technical constraints or decided by the smart-charging manager.

Interestingly, the operator $\hat{C}$ associated with the cost function of equation (2.11) can be natively realised on a neutral atom platform [PSTZ$^+$18], with some constraints on the graph edges. We map a ground state and a Rydberg state of each atom to a spin 1/2, where $|1\rangle = |r\rangle$ is a Rydberg state and $|0\rangle = |g\rangle$ is a ground state. An atom in a Rydberg state has an excited electron with a very high principal quantum number and therefore exhibits a huge electric dipole moment. As such, when two atoms are excited to Rydberg states, they exhibit a strong van der Waals interaction. Placing n atoms at positions $\mathbf{r}_j$ in a 2D plane, and coupling the ground state $|0\rangle$ to the Rydberg state $|1\rangle$ with a laser system enables the realisation of the Hamiltonian

$$H = \sum_{i=1}^{n} \frac{\hbar\Omega}{2} \hat{\sigma}_i^x - \sum_{i=1}^{n} \frac{\hbar\delta}{2} \hat{\sigma}_i^z + \sum_{j<i} \frac{C_6}{|\mathbf{r}_i - \mathbf{r}_j|^6} \hat{n}_i \hat{n}_j. \tag{2.12}$$

Here, Ω and δ are respectively the Rabi frequency and detuning of the laser system and $\hbar$ is the reduced Planck constant. The first two terms of equation (3.56) govern the transition between states $|0\rangle$ and $|1\rangle$ induced by the laser, while the third term represents the repulsive Van der Waals interaction between atoms in the $|1\rangle$ state. More precisely, $\hat{n}_i = (\hat{\sigma}_i^z + 1)/2$ counts the number of Rydberg excitations at position i. The interaction strength between two atoms decays as $|\mathbf{r}_i - \mathbf{r}_j|^{-6}$.

The shift in energy originating from the presence of two nearby excited atoms induces the so-called *Rydberg blockade* phenomenon, illustrated in Fig. 2.3(a). More precisely, if two atoms are separated by a distance smaller than the Rydberg blockade radius $r_b = (C_6/\hbar\Omega)^{1/6}$, the repulsive interaction will prevent them from being excited at the same time. On the other hand, the sharp decay of the interaction allows us to neglect this interaction term for atoms distant of $\mathbf{r} \gg r_b$[3]. As such, for $\Omega = 0$, the Hamiltonian in Eq. (3.56) is diagonal in the computational basis and enables to realize $H|z_1, \ldots, z_n\rangle = (\hbar\delta/2)C(z_1, \ldots, z_n)|z_1, \ldots, z_n\rangle$, with the cost function specified in Eq.(2.11), and for which there is a link between atoms i and j if they are closer than r_b apart.

To conclude, the MIS problem on Unit-disk graphs can be naturally encoded in the Ising Hamiltonian of a neutral-atom device. This means that we can use an analog control over the atoms which will likely offer better performances than digital approaches as it is a natural control over the device.

2.2 Tackling (SC1) and (SC2) with the Quantum Approximate Optimisation Algorithm (QAOA)

In the previous section, we have demonstrated how two Smart-Charging problems (SC1) and (SC2) can be mapped into QUBO instances which in turn can be encoded in the ground-state

---

[3] We will see later on that in practice some atoms are at a distance $r \simeq r_b$ and their interaction must be considered in the simulations.

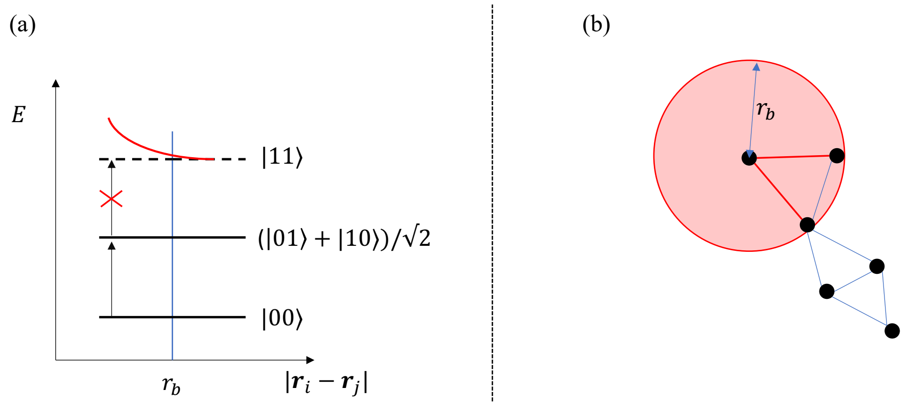


Figure 2.3: (a) Illustration of the Rydberg blockade effect. When two atoms are far apart, $|\mathbf{r}_i - \mathbf{r}_j| > r_b$, they don't interact. On the other hand, if they are separated by less than the Rydberg blockade radius, $|\mathbf{r}_i - \mathbf{r}_j| < r_b$, a strong interaction prevents the two atoms to be in the state $|1\rangle$ at the same time. (b) Rydberg blockade and independent sets of a graph. Rydberg atoms correspond to the vertices of a UD-graph. There are edges between adjacent vertices if the distance between them is smaller than the Rydberg blockade radius r_b, as illustrated for the top-left vertex. Due to the Rydberg blockade effect, the dynamics of the system is restricted to independent sets.

of a many-body quantum Hamiltonian $\hat{C}$. The challenge remains now in the preparation of the ground-state of $\hat{C}$ which is not a trivial task: in the following section we will describe the Quantum Optimisation Approximation Algorithm (QAOA), a well-known quantum approach for preparing the ground-state of many-body quantum systems.

QAOA computes approximate solutions to combinatorial optimisation problems, with a theoretical guarantee of convergence when the depth of the quantum circuit increases [FGG14]. QAOA is a variational algorithm for combinatorial problems in which a quantum processor works hand-in-hand with a classical counterpart, as illustrated in Fig. 2.4 (see Ref. [CAB+20] for a review on variational algorithm). The quantum processor is used to prepare a wave function $|\boldsymbol{z}_{\gamma,\beta}\rangle$. In the most general case, $|\boldsymbol{z}\rangle = |z_1 z_2 ... z_n\rangle$ represents a n-qudit state vector, with $z_i \in \{0, 1, .., d\}$, and the subscript in $|\boldsymbol{z}_{\gamma,\beta}\rangle$ indicates that the state belongs to a family of states that is parameterized by the angles $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$. More specifically, $|\boldsymbol{z}_{\gamma,\beta}\rangle$ is generated by the successive application of unitaries generated by the non-commutative operators $\hat{M}$ and $\hat{C}$ defined below, with angles given by $\boldsymbol{\beta} = (\beta_1, \beta_2, ..., \beta_p)$ and $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, ..., \gamma_p)$, respectively. Given an initial state $|\boldsymbol{z}_0\rangle$, the wavefunction prepared by the quantum processor takes the following form,

$$|\boldsymbol{z}_{\gamma,\beta}\rangle = e^{-i\beta_p \hat{M}} e^{-i\gamma_p \hat{C}} \dots e^{-i\beta_1 \hat{M}} e^{-i\gamma_1 \hat{C}} |\boldsymbol{z}_0\rangle. \tag{2.13}$$

A common choice for the cost operator $\hat{C}$ (also sometimes referred to as energy operator) is the diagonal operator in the computational basis, $\hat{C}|\boldsymbol{z}\rangle = C(\boldsymbol{z})|\boldsymbol{z}\rangle$, where $C(.)$ is the cost function to be optimised for, while the mixing operator $\hat{M}$ induces transitions between states in the computational basis [FGG14, HWO+17]. In the following, we will always reformulate our problems under the form of minimisation problems, by changing the sign of the cost function of the original problems. The ground state of the energy operator $\hat{C}$ corresponds to the optimal solution to the optimisation problem. The dimension p of the vectors $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ is called the depth of the algorithm.

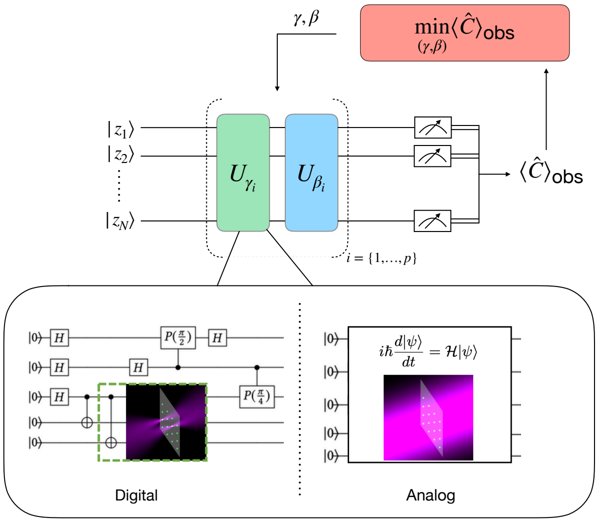


Figure 2.4: Principle of the QAOA algorithm. A quantum processor, which can be operated in either digital or analog mode, is used to prepare an ansatz wavefunction from which we construct the mean value $\langle \hat{C} \rangle_{\text{obs}}$ using numerous measurements. A classical optimiser then updates the variational parameters. Some problems are naturally tailored to the analog mode of the platform, while others require a digital mapping. In digital mode, the unitaries are built from quantum circuits, made of elementary quantum gates acting each on one or a few qubits. In the analog mode, the unitaries are built from sequences of Hamiltonians that can be controlled in a continuous manner.

The quantum state is then measured to construct an statistical estimator $\langle \hat{C} \rangle_{\text{obs}}$ of the cost function to be minimised. A classical optimisation procedure uses this estimator to update the variational parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ for the next iteration. This loop repeats until convergence to a final state, from which an estimate of the solution to the problem is extracted. The approximation ratio $\langle \hat{C} \rangle / C_{\text{opt}}$, where $C_{\text{opt}} = \min_{\mathbf{z}} C(\mathbf{z})$, measures the quality of the approximation yielded by QAOA. Once optimal parameters for QAOA are found the optimal assignment of variables is returned by repeatedly preparing and measuring the quantum circuit while saving the best result. As was pointed out in Ref. [FGG14], a solution with value at least $\langle \hat{C} \rangle - 1$ is obtained with probability $1 - \frac{1}{m}$ after $O(m \log m)$ launches where $m = C_{\text{opt}} \leq |E|$. Thus the sampling step adds only a polynomial factor and is in general not accounted in the run-time analysis of QAOA. In the following part we will explore the classical loop optimisation procedure.

2.2.1 Optimising the classical loop

Although the translation of a combinatorial problem focuses on the quantum loop of QAOA, finding the optimal parameters $(\boldsymbol{\gamma}, \boldsymbol{\beta})$ is a tricky task. We first describe how the energy landscape associated to the variational parameters depends in the (SC1) problem on the data itself.

In (SC1), the weights w_{uv} of the edges impact the phase that is applied to each basis state of the computational basis in the QAOA cycle. To visualise the impact of weights, we plot the energy (or cost-function) landscape at depth $p = 1$ as a function of the parameters (γ, β). Figure 2.5 contains numerical simulation of the energy landscape for the max-cut problem evaluated on a graph of size 10, with a resolution of 30 points along each axis. For the same problem, we re-weight the adjacency matrix of the graph by different factors. On the first figure, a high density of peaked valleys and hills indicates that optimisation is difficult and would require an important amount of function evaluations to find a decent solution. The amount of peaks and valleys is due to the fact that the cost Hamiltonian adds a phase term to each basis vector z_i. Indeed, applying $\hat{C}$ with an angle γ modifies $|z_i\rangle$ to $e^{i\gamma C_i}|z_i\rangle$, where C_i corresponds to the cost of the colouring z_i. Modifying the graph weights consequently modifies the phase applied to each basis state $|z_i\rangle$. A smaller C_i, as seen in the third figure, smooths the cost-function landscape enabling adequate local optimisation. Artificially reducing C_i too much however might over-smooth the landscape, reducing the possible phases that QAOA can apply, hence missing the global minima. We want C_i to be big enough to allow basis states z_i to acquire a phase $e^{i\phi}$ in a comfortable range. At the same time, we do not believe a phase $\phi > 2\pi$ is necessary. The re-weighting $C_i = R^* C_i$, where $R^* = \frac{2\pi}{\max(C_i)}$ satisfies the two previous conditions. Numerically, we find that re-weighting the adjacency matrices of all instances by the factor R indeed concentrates all optimal parameters in a restricted zone of parameter space.

Practically of course, calculating R^* implies knowledge of $\max C_i$, which corresponds to the exact solution to the initial problem. We therefore propose the upper bound $R = w_{\max} \frac{N^2}{4}$, where $w_{\max} = \max_{u,v} w_{uv}$. It is calculated from the best-cut on a complete graph where all weights would be equal to $w_{\max}$.

In the scenario of (SC1), we see that the choice for the optimiser depends on the normalisation of the weights. If one manages a good normalisation, then local methods of optimisation can be used with high guarantees of success.

2.2.2 Finding correct parameters: the Egg optimisation

A major hurdle in the optimisation process of quantum variational schemes is the presence of barren plateaus [MBS+18]. The phenomenon of barren plateaus refers to situations where the optimisation landscape becomes extremely flat, making it difficult to find an optimal solution using standard gradient-based optimisation methods. This flatness occurs when the gradient of the objective function with respect to the circuit parameters becomes exponentially small and can be induced by random parametrisation of the ansatz [MBS+18], optimisation with respect

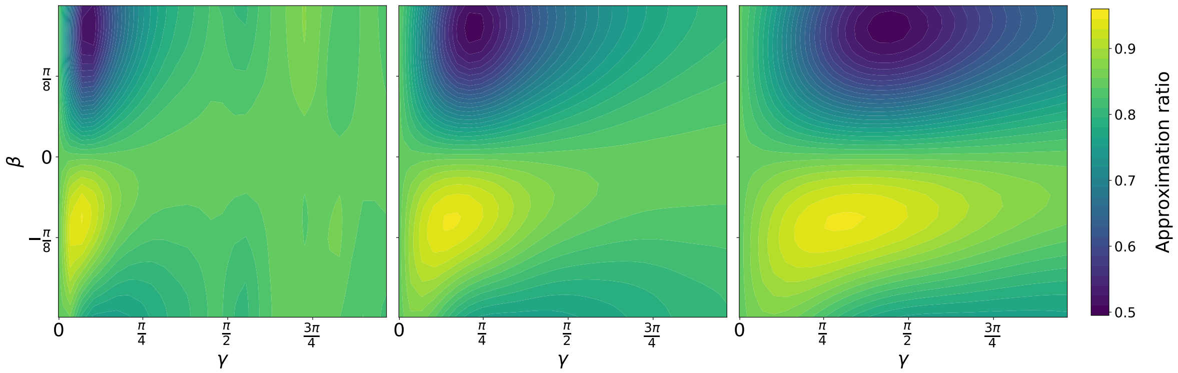


Figure 2.5: **The zoom effect**. From left to right, we normalised an instance from (SC1) respectively by $R/5$, $R/2$ and R to observe the effect on the energy landscape of $QAOA_1$.The closer to R, the better the zoom on the global minima. The re-weighting of a graph affects the energy landscape: it can therefore be used to either zoom on the point of interest to apply local optimisation, or on the contrary it might be used to zoom out of barren plateaus to explore more interesting phase spaces.

to a global cost function [CAB⁺21] and can also be entanglement-induced [MKW21]. When the gradient is small, it means that small changes in the parameters of the quantum circuit have little effect on the objective function. Consequently, optimisation algorithms struggle to identify the direction of steepest descent, as the gradient provides weak or no guidance. The presence of barren plateaus poses a significant challenge for optimising variational quantum circuits. It can lead to slow convergence, increased sensitivity to noise and errors, and a higher likelihood of getting stuck in sub-optimal solutions. These issues can significantly limit the usefulness of variational quantum algorithms for practical applications.

In order to find the best variational parameters for p layers of QAOA we develop a method based on the idea of making an educated guess from previous layers to the new one [ZWC⁺20], hence limiting the size of the phase space for optimisation. The Educated Global Guess (Egg) optimisation process uses differential evolution (DE) [SP97] rather than a local optimisation in an attempt to find the global optimum in a wrinkled energy landscape. DE works by starting with an ensemble of points in the phase-space, called the *agent population.* Then, theses agents are moved around by recombining their coordinates, and the function is evaluated for these new agents. If the new position brings an improvement, it is kept, otherwise it is discarded. This process is repeated until convergence to a minimum, although there is no guarantee that the global minima will be found. While it cannot be sure that our method will always work perfectly, the constant growth of the approximation ratio as p increases in our results is a reassuring indicator, as illustrated on the top panel of Figure 2.10. We strongly reduced the complexity of the problem by making a global educated guess for the optimal parameters at layer p using the optimal parameters found at layer $p-1$. We use a global optimiser rather than a local one in an attempt to find the global optima in a wrinkled energy landscape. This very much improved computation time by reducing the amount of phase space addressed, and demonstrated precise results with much fewer function calls than local methods (see Figure 2.6). The algorithm is described in pseudo-code below and works as follows: find optimal parameters (γ_1^*, β_1^*) for $p = 1$. For the next layer, optimise the function $C : (\gamma_2, \beta_2) \mapsto C(\gamma_1^*, \gamma_2, \beta_1^*, \beta_2)$. As such, two variables are already fixed and the space to explore is once again only two-dimensional. Once the optimisation ends on the two new coordinates, a local optimisation is done on all the coordinates. This quick step enables to re-calibrate the previous parameters: it is therefore possible to achieve a trotterization process for high values of p.

Algorithm 1 Educated global guess (Egg) optimisation

```
Require: $|z_0\rangle, \hat{C}, \hat{M}, p$
Ensure: variational parameters $(\gamma_1, \dots, \gamma_p, \beta_1, \dots, \beta_p)$
$\gamma_1, \beta_1 \leftarrow \mathrm{DE}(|z_0\rangle, \hat{C}, \hat{M})$
for $k$ in range$(2, p)$:
    $|z_{k-1}\rangle = e^{i\beta_{k-1}\hat{M}} e^{i\gamma_{k-1}\hat{C}} \dots e^{i\beta_1\hat{M}} e^{i\gamma_1\hat{C}} |z_0\rangle$
    $\gamma_k, \beta_k = \mathrm{DE}(|z_{k-1}\rangle, \hat{C}, \hat{M})$
    local optimisation: BFGS$(\gamma_1, \dots, \gamma_k, \beta_1, \dots, \beta_k, |z_0\rangle, \hat{C}, \hat{M})$
return $(\gamma_1, \dots, \gamma_p, \beta_1, \dots, \beta_p)$
```

A choice of the optimisation routine has a high impact on the performance and cost of an experimental implementation of QAOA [GS17]. By experimentally comparing different optimisation methods we observed that for local methods, a gradient-free *Nelder-Mead* [NM65] is the best choice for our purposes: its performance is comparable to the one of the quasi-Newton *BFGS* [GMW19] (and both methods outperform the *COBYLA* routine [Pow94] which is often used in works on applied QAOA [EMW20]) while it requires less evaluations of the objective function (see Figure 2.6).

We compare these local methods with *Differential Evolution* (DE) [SP97]. We limited the number of function calls while using DE in order to have the best trade-off between global exploration and low number of function evaluation. As seen in Figure 2.6, it is a very effective method, for which the approximation ratio grows close to Nelder-Mead using however much less function evaluations with the layers. It should be kept in mind nonetheless that global optimisation methods require tuning hyper-parameters, a process that must be adjusted by hand. The optimal hyper-parameters might change from one problem to another, a reason why they are not so popular. But building optimisation processes such as DE that require little function evaluations is key in the NISQ era as it ensures quicker performances on unstable devices.

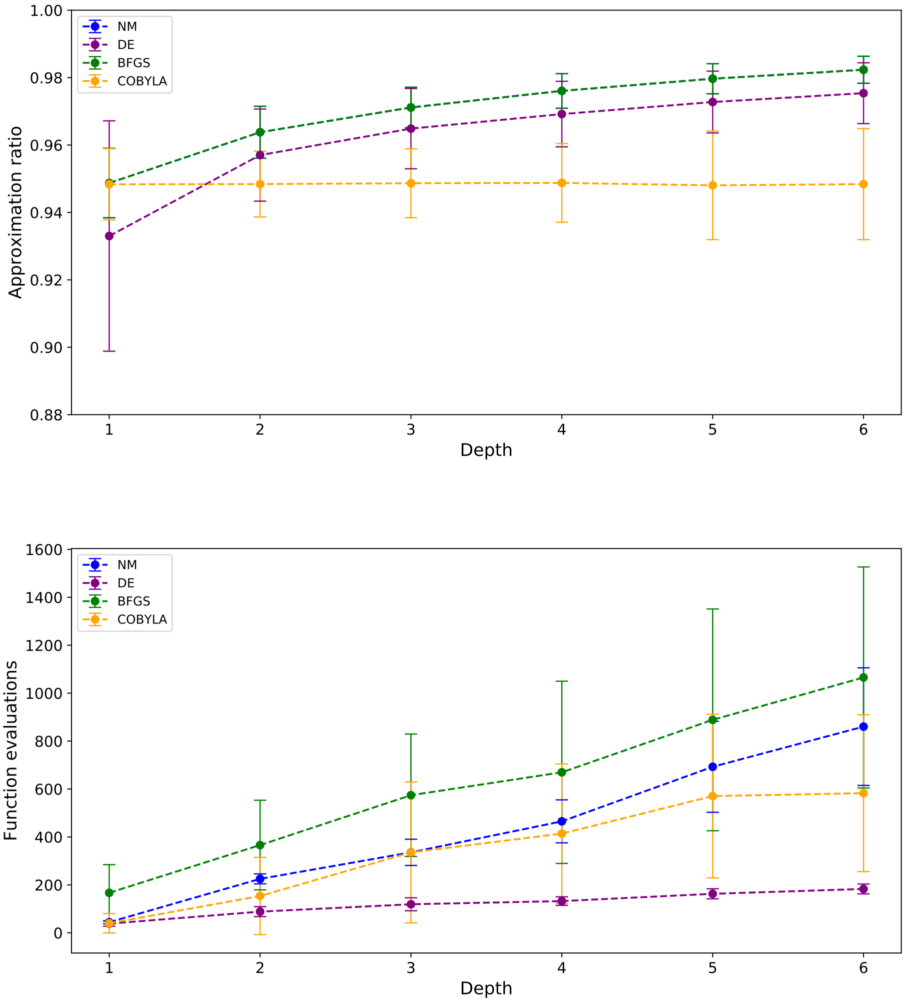


Figure 2.6: Cost and performance of different numerical optimisation methods on Max-Cut (SC1), for graphs with size $N = 10$. Differential Evolution (DE) corresponds to a global optimisation algorithm. While yielding results very close to Nelder-Mead, it does so by requiring very little number of function evaluations. Building optimisation processes such as DE that require little function evaluations is key in the NISQ era as it ensures quicker performances on unstable devices.

2.3 Results

In this section, we describe the results obtained when running QAOA on the two smart-charging problems. We first present an analytical result for (SC1) that allows to compute the mean cost in a QAOA with depth $p = 1$, coined QAOA$_1$, without any call to a quantum computer (or emulator), which is in turn useful to analyse the performance of QAOA$_1$ on large instances.

Analytical result for Max-Cut: the mean cost in the QAOA$_1$ state

An analytical expression for max-cut on unweighted graphs for $p = 1$ was derived in [WHJR18] and can be extended to the case of complete weighted graphs. For an edge $(u, v) \in \mathcal{E}$ the mean value of $C_{u,v}$ for chosen γ, β is:

$$\langle \mathbf{z}_{\gamma,\beta}|C_{u,v}|\mathbf{z}_{\gamma,\beta}\rangle = \frac{w_{uv}}{2}\left(1 + \frac{\sin(4\beta)\sin(\gamma w_{uv})}{2}\left[\prod_{x\neq u,v}\cos(\gamma w_{ux}) + \prod_{x\neq u,v}\cos(\gamma w_{vx})\right] + \right.$$
$$\left. -\frac{\sin^2(2\beta)}{2}\left[\prod_{x\neq u,v}\cos(\gamma(w_{ux} - w_{vx})) - \prod_{x\neq u,v}\cos(\gamma(w_{ux} + w_{vx}))\right]\right). \quad (2.14)$$

In this expression we realise that there is a close link between the value of the QAOA$_1$ angle γ and the weight of the edge w_{uv}, which is in accordance with what we observed in section 2.2.1.

Numerical results (noiseless simulations)

In the following, we present a performance analysis of the various procedures presented above. For max-cut at depth $p = 1$ we use the analytical formula of equation (2.14). For max-4-cut, MIS and max-cut at depth $p > 1$, we compute the mean cost by Monte-Carlo estimation, by simulating the quantum evolution either on the Atos Quantum Learning Machine [ato] or using the QuTIP library [JNN13] on the OCCIGEN supercomputer based in Montpellier, France.

Dataset

Data were driven from a set of 2250 loads performed during May 2017 on identical charging points of the Belib's network of load stations located in Paris, France [Bel17].

For both problems (SC1) and (SC2), an instance is a series of chronological loads characterised by their duration for (SC1) and starting/end times for (SC2). The instance size is the number of loads it contains. A dataset is a set of instances whose first load is randomly chosen among the 2250, according to a uniform law. Once an instance is built:

- for (SC1), a priority $p_i \in \mathbb{N}$ is randomly assigned to each load according to a Poisson's law
$$\mathbb{P}(p_i = k) = \frac{\lambda^k e^{-\lambda}}{k!},$$
enforcing a constant distribution of the different priority levels in each instance[4];
- for (SC2), the belonging to a group is randomly assigned to each load according to a uniform law parameterised by the number of groups and the number of loads in the instance.

---

[4] The "brute" weights from the data yield high values in the cost Hamiltonian, leading to a highly fluctuating energy landscape for $QAOA_1$ (cf. equation 2.14). A re-weighting was therefore done in order to obtain a smoother landscape for better optimisation as explained in section 2.2.1).

Minimisation of Total Weighted Load Completion Time (SC1)

First, we compare the performance of QAOA and the randomised algorithm on max-cut instances of different sizes $n \in [6, 8, 10, 15, 30, 50, 70, 100, 150]$.

The purpose of our performance analysis is not to demonstrate that QAOA beats the best classical algorithm but rather to compare it to a solution of the same nature. Indeed, both QAOA and randomised algorithm return a sample from a certain probability distribution (built by a quantum circuit and uniform distributions respectively). We leave for future work a more advanced comparison with local classical algorithms of bounded depth, that were suggested as fair competitors to QAOA in [Has19].

In order to compute the exact optimum C_{opt} we use brute-force search for max-cut on small instances (with up to 30 vertices) and the dynamic program algorithm presented in Ref. [Sah76] for the initial scheduling problem on bigger instances.

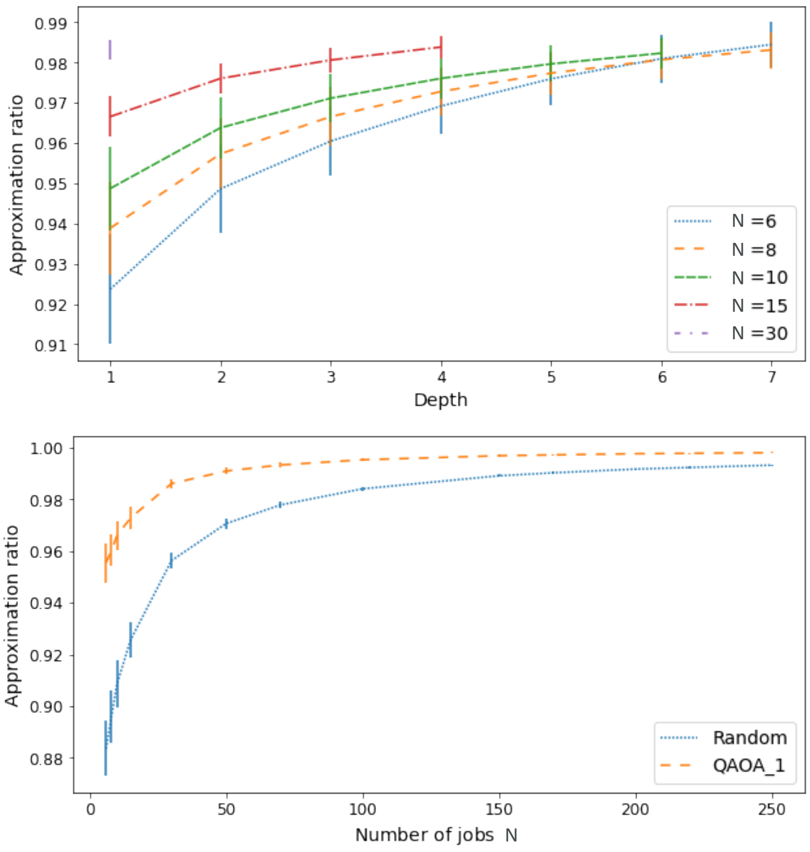


Figure 2.7: Top panel: Evolution of the approximation ratio of $QAOA$ with depth p for the Max-Cut problem. Bottom panel: Evolution of the average approximation ratio with the instance size for $QAOA$ at depth $p = 1$ (dashed orange line) and for the randomised algorithm on the *initial scheduling problem* (dotted blue line).

As expected, we observe on the top panel of Fig. 2.7 that, for a fixed value of n, the approximation ratio improves with the QAOA depth p.

Surprisingly, we also notice that the approximation ratio gets better with the size of the Max-Cut instance. Such behaviour is also observed for the randomised algorithm, as illustrated on the bottom panel of Figure 2.7. In this numerical experiment we observe that QAOA finds better solutions than the randomised algorithm.

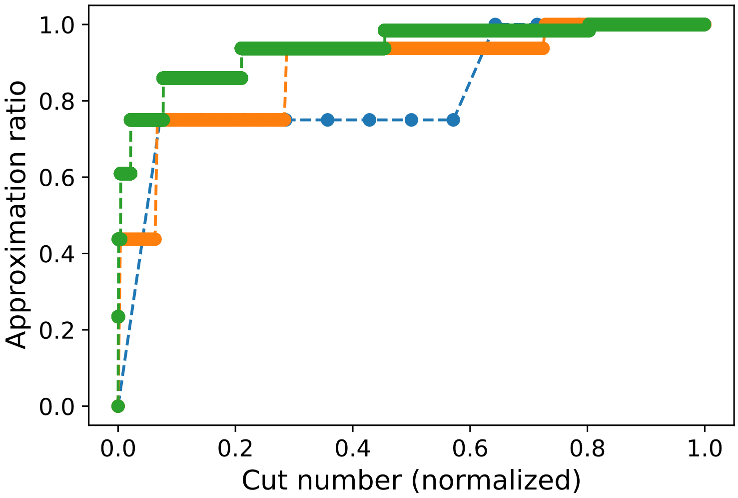


Figure 2.8: Approximation ratios of all possible cuts in the unweighted Max-Cut problem on complete graphs of size N, sorted by increasing values. The blue, orange and green dots correspond to $N = 4$, $N = 8$, and $N = 16$, respectively.

The good performances of the randomised algorithm shown on the bottom panel of Figure 2.7 suggest that the difficulty of the problem under consideration decreases with its size. To confirm this insight, let us consider the simpler case of unweighted Max-Cut on a complete graph. In this scenario, choosing a cut at random gives an approximation ratio of $\left[\sum_k \binom{n}{k} k(n-k)\right] / (n^2/4)$ which goes to 1 in the large n limit. This fact is illustrated in Fig. 2.8, where we show the approximation ratios of all possible cuts for different values of n. This plot shows that, for a fixed positive value of the normalised cut number, the corresponding approximation ratio approaches 1 in the large n limit. While our findings suggest that the presence of $\mathcal{O}(1)$ weights in our problems leads to the same behaviour, it does not extend to instances in which the magnitude of the weights would increase with n.

We now present numerical results for max-4-cut. Using the normalisation factor introduced in Section 2.2.1, we run QAOA for $p = 7$ layers and plot the statistical distribution of the approximation ratio achieved for 98 different instances (Figure 2.9). As expected, the approxi-

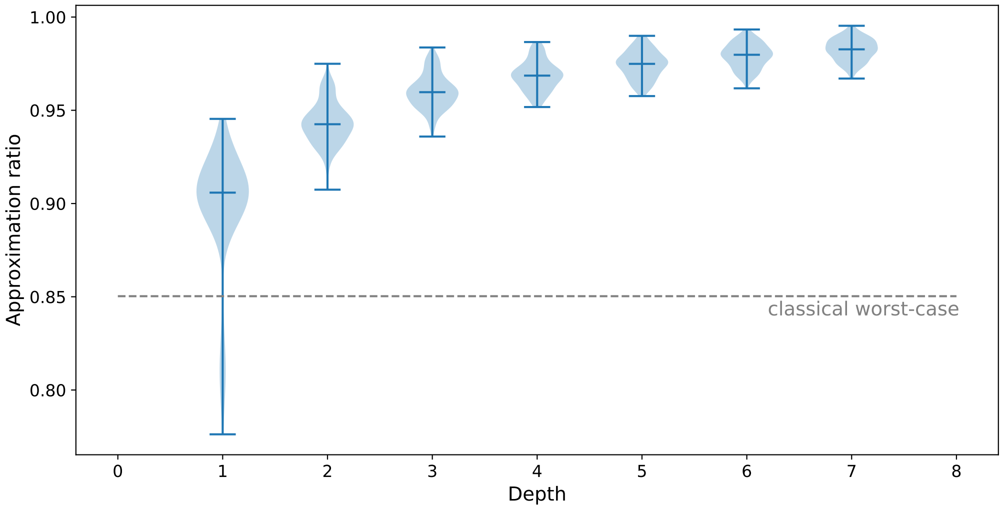
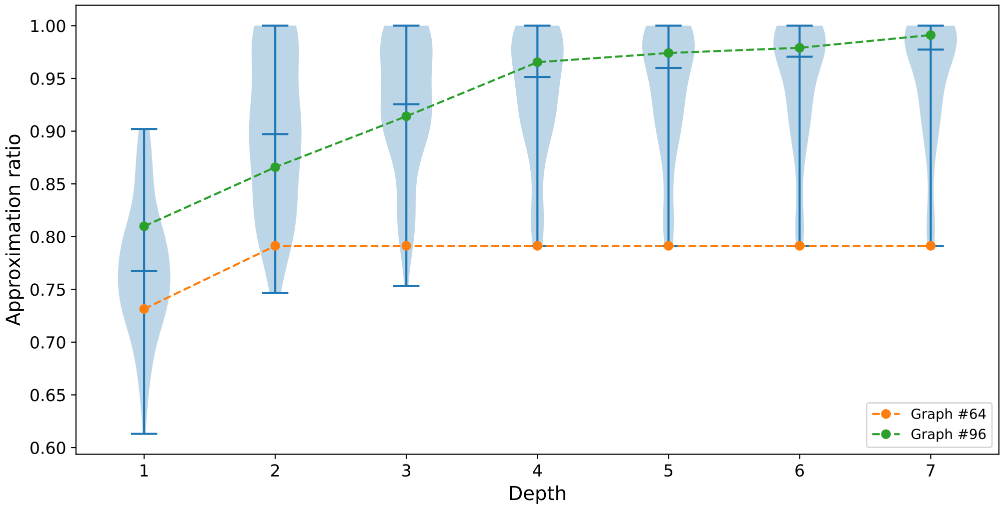


Figure 2.9: **Max-4-cut**. QAOA for $p = 7$ layers, and the statistical distribution of the approximation ratio achieved for 98 instances. Proper re-weighting of the graph uniforms the instances. The high value of the approximation ratio achieved is an encouraging result, showing that quantum approaches are comfortably higher than the classical approximation minimal guarantee of 0.857487 [dKPW04], even at low depths. While initial optimisation might not be ideal, as suggested by the tail of distribution on the first layer, it is corrected by the global smoothing at the next layer.

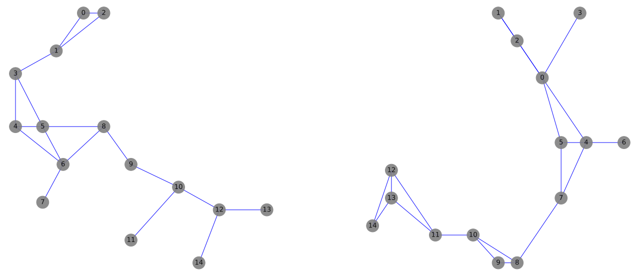
mation ratio increases to 1 with the number of layers. The high value of the approximation ratio achieved is an encouraging result, showing that quantum approaches are comfortably higher than the classical minimal approximation guarantee of 0.857487 [dKPW04], even at low depths. At $p = 1$, the tail of the distribution indicates that some instances have been poorly optimised. The fact however that this tail disappears in the next layers shows that initial poor optimisation can be corrected in the following layers. This might be explained by the fact that at the end of each layer our classical algorithm implements a rapid local optimisation on all parameters.

Optimal Scheduling of Load Time Intervals within Groups (SC2)

Thanks to Egg optimisation, the function evaluations can be done in parallel. Specifically, we ran our program on the OCCIGEN supercomputer where we used 28 CPU cores in parallel for our calculations. This reduced by a factor of 8.8 the time of derivation. In addition, we reduced the amount of phase space addressed by making educated guesses from layer $p-1$ to layer p, strongly inspired by Ref. [ZWC+20], and showcased precise results with much fewer function calls. The combination of educated guesses and parallel function evaluations made



(a) Evolution of the approximation ratio for 84 instances of Unit-Disk MIS ($n = 15$).



(b) Graph #96.

(c) Graph #64.

Figure 2.10: **(a)** For each layer, the median is plotted and the violin envelop represents the distribution of the approximation ratio achieved for each of the 84 points. While the approximation ratio grows rapidly to 1 in most cases, there seems to be some instances that are harder to optimise. We isolate a graph representative of the best-case bulk **(b)** and one for the worst-case bulk (**c**). Further understanding of the worst-case graphs is important in order to estimate the quality of a quantum approximation algorithm.

for a consequent speed-up, bringing down typical calculation times from half a day to an hour. Finally, the global optimiser (DE) has shown strong results in the context of noisy and changing landscape [ROM11], a typical behaviour of our noisy intermediate-scale quantum device. Using (DE) might prove robust in an experimental setup.

As can be seen in Fig. 2.10(a), the performances of QAOA for solving the UD-MIS problem are good in average, exceeding approximation ratios of 0.95 after seven layers. As can be noted in the figure, the distribution of the approximation ratio for each layer is rather wide. At the third layer, the distribution starts to separate in two bulks. The lower bulk stagnates by the fourth layer as the approximation ratio stays inferior to 0.85 until the last layer, while the approximation of the upper bulk increases to one as the depth grows. For completeness, we show on panel (b) and (c) of Fig. 2.10 a typical graph instance of each group. The instances in the lower bulk correspond to worst-case scenarios and represent 9.5% of the instances. Understanding the characteristics of the worst instances is of crucial importance to characterise the quantum approaches. Indeed in the approximation theory the quality of an algorithm is benchmarked on worst-case instances. The approximation ratio achieved by the algorithm on these particular instances is a guarantee from below for any other instance. Finding worst-case scenario has been investigated in the past for max-cut on uniform 3-regular graphs at depths $p = \{1, 2, 3\}$ [FGG14, WL20]. Obtaining a lower bound guarantee of QAOA on UD-MIS would enable us to compare the quantum approach to the classical approximation scheme to assess an eventual quantum advantage.

To conclude, we have shown numerical results for smart-charging problems, supposing a QAOA approach to find good solutions in a hardware-agnostic fashion. Because we seek to run these problems on actual devices, we explore in the following section the different characteristics that come into play when embedding the problems on a neutral-atom device. We focus on the implementation of (SC2), as the UD-MIS problem is naturally connected to the Ising Hamiltonian of Rydberg atoms.

2.4 Towards hardware implementation of unit-disk MIS

(Dalyac C., Leclerc L. et al. in preparation)

In the following section, we explore how one can implement UD-MIS on a neutral atom device and find good approximate solutions. The first step is to embed the graph with the atoms, which requires some work to insure that the interaction matrix resembles the adjacency matrix of the graph under study. Then, we present a batching technique that runs multiple graphs on the same SLM pattern, enabling a faster processing of the graphs on the device. Finally, we precise the sources of noises that the hardware is subject to and devise the adequate method to prepare the ground-state of the cost Hamiltonian associated to UD-MIS.

2.4.1 Interactions have to be faithful to the graph's adjacency matrix

In the previous section where we explored QAOA on 84 instances of Unit-Disk graphs, the respective positions of the vertices are given such that they respect the unit-disk constraints. In this scheme, two vertices are connected by an edge if their distance is inferior to the Rydberg blockade radius r_b. If the distance between two vertices is $r_b + \varepsilon$, where $\varepsilon > 0$, the vertices are considered to be not connected. Mathematically, this hard-sphere approximation cutoff is given by the following interaction function

$$h(r) = \begin{cases} \infty & \text{if } r \leq r_b, \\ 0 & \text{else} \end{cases} \tag{2.15}$$

where r is the distance between two vertices. In reality however, the atoms interact with each other continuously with a dependency in $1/r^6$: recall that the interaction term in the Ising Hamiltonian of equation (3.56) is given by

$$U = \sum_{j<i} \frac{C_6}{r_{ij}^6} \hat{n}_i \hat{n}_j. \tag{2.16}$$

which means that the interaction between atoms which are close to r_b is non-negligible as shown in Figure 2.11.

We therefore modify the positions of the atoms in order to make the hard-sphere approximation faithful. If this is the case then it ensures that the ground-state of the Ising Hamiltonian at $\Omega = 0$ and $\delta > 0$ effectively encodes the MIS.

Hard-sphere optimisation of the atomic positions

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a UD-graph with given positions $\{\mathbf{r}_i\}_{i \in 1,\ldots,|\mathcal{V}|}$ represented by an ensemble of atoms. Because all vertices have a fixed position, we can define for each pair of atoms an interaction energy $U_{ij} = \frac{C_6}{r_{ij}^6}$, where C_6 is the interaction constant related to the Rydberg level and r_{ij} is the Euclidean distance between atoms encoding vertices i and j. Recall that $\bar{\mathcal{E}}$ is the complementary set of edges of $\mathcal{G}$. The hard-sphere approximation requires that the largest distance between two atoms sharing a bond in the graph must be far less than the shortest distance between two atoms not sharing a bond. Formally, the interaction energy $U_{\min}^{\mathcal{E}} = \min\{U_{ij}, (i,j) \in \mathcal{E}\}$ should be an order of magnitude stronger than $U_{\max}^{\bar{\mathcal{E}}} = \max\{U_{ij}, (i,j) \in \bar{\mathcal{E}}\}$.

We therefore pre-process the position of the vertices to ensure that the hard-sphere approximation is valid. We developed a heuristic that locally modifies the positions of the vertices. The result of the optimisation procedure is presented in Figure 2.12. The key idea is to create a big gap between $U_{\min}^{\mathcal{E}}$, which is the minimal energy between two atoms representing connected vertices in the graph $\mathcal{G}$, and $U_{\max}^{\bar{\mathcal{E}}}$, which corresponds to the maximal energy between two atoms that do not represent vertices that are connected in the graph $\mathcal{G}$. Because unit-disk graphs are local, in the sense that vertices are only connected to other vertices which are close in

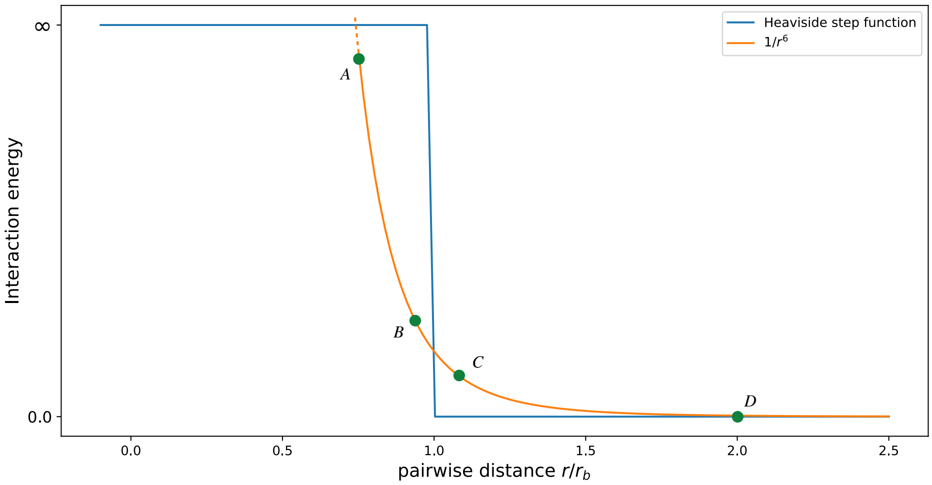


Figure 2.11: **Hard-sphere approximation of the Rydberg blockade radius**. The van der Waals interaction energy between two Rydberg atoms separated by a distance r follows a $1/r^6$ power law (orange). In the hard-sphere approximation scheme, or Heaviside approximation (blue), one considers that two atoms that are far from r_b have no interaction (point D), while atoms inside of the Rydberg blockade radius (point A) have infinite interaction energy. Sometimes however when embedding random graphs it might happen that the pairwise distance is close to r_b (point B or C). In this case, the approximation does not hold and the dynamics might be much different than expected.

distance, we devise a procedure that finds optimal positions for sub-groups of vertices, before concatenating the positions together. We describe the optimisation algorithm ?? below and the result is plotted in Figure(2.12):

Algorithm 2 OPTIMISING ATOMIC POSITIONS FOR HARD-SPHERE APPROXIMATION

```
Require: Unit-disk graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
Ensure: Optimal positions of the atoms $(x_1^*, \ldots, x_{|\mathcal{V}|}^*)$ that maximises the hard-sphere approximation.
Initial position $(x_1, \ldots, x_{|\mathcal{V}|})$
opt_pos = [ ], marked_vertices= [ ]
for $i$ in $\mathcal{V}$:
sub_vertices = $[i] + [j$  for j in $\mathcal{N}_{\mathcal{G}}(i)]$        ▷ $\mathcal{N}_{\mathcal{G}}(i)$ is the neighbourhood of $i$.
opt_pos ⟵ FindOptimalSubPos(sub_vertices, marked_vertices)
marked_vertices ⟵ marked_vertices + sub_vertices
Return opt_pos
```

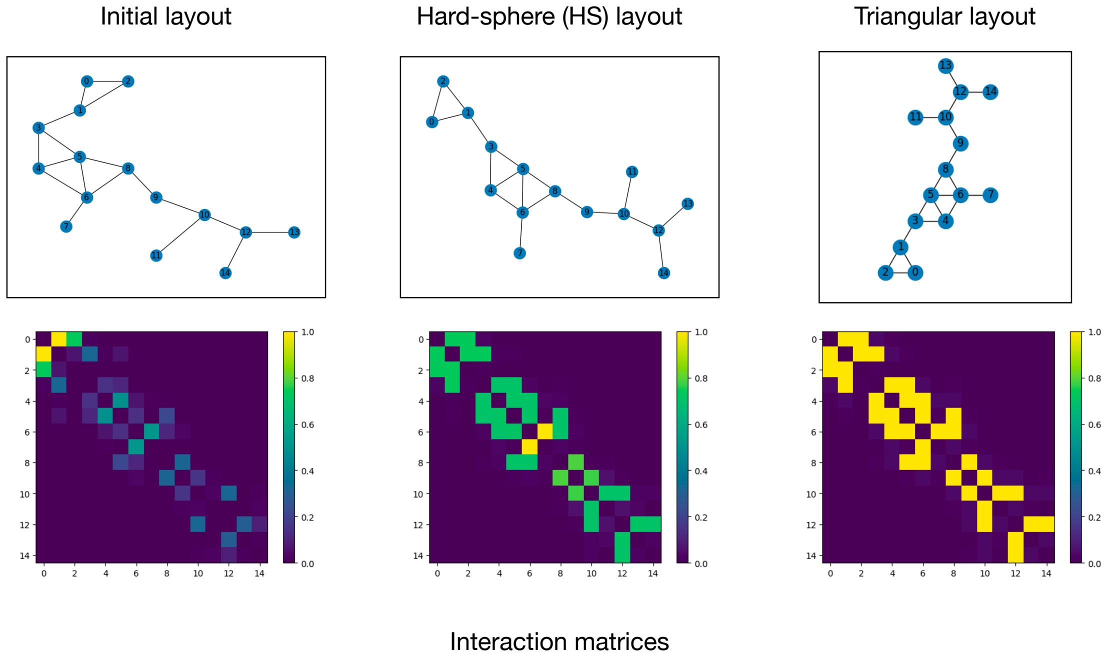


Figure 2.12: **Hard-sphere optimisation brings the interaction matrix close to the adjacency matrix of the graph.** In this example, the initial layout of the atoms induce an interaction matrix that is not representative of the adjacency matrix of the graph. This can induce dynamics that are not related to the graph under study. Thanks to the hard-sphere optimisation scheme, the atomic positions are modified such that the induced interactions between the atoms replicate as closely as possible the adjacency matrix of the graph. In some cases, the connectivity of the graph allows it to sit perfectly on an underlying triangular lattice, in which case the interaction matrix replicates perfectly the adjacency matrix as the pairwise distances are constant and next-nearest neighbour interaction is close to 0.

Algorithm 3 FindOptimalSubPos

```
Require: sub_vertices, marked_vertices, $\mathcal{G}$
Ensure: Optimal positions of the sub vertices that maximises hard-sphere approximation, knowing the marked vertices.
$\tilde{\mathcal{G}}$ : sub-graph of $\mathcal{G}$ with sub_vertices and marked_vertices.
pos $\longleftarrow \{x_i\}_{i \in \tilde{\mathcal{G}}}$
$\{x_i^*\}_{i \in \tilde{\mathcal{G}}} \longleftarrow$ minimise $U^{\bar{\tilde{\mathcal{E}}}}_{\max}/U^{\tilde{\mathcal{E}}}_{\min}$ by moving positions $\{x_i\}_{i \in \tilde{\mathcal{G}}}$
Return $\{x_i^*\}_{i \in \tilde{\mathcal{G}}}$
```

In this algorithm, we keep track of the vertices for which we have already found optimal positions that are called *marked vertices*. Then, we select a vertex i and its neighbourhood $\mathcal{N}(i)$ which are called *sub vertices* and build the subgraph $\tilde{\mathcal{G}}$ induced by the sub vertices. Given the subgraph $\tilde{\mathcal{G}}$, we calculate the values of $U^{\bar{\tilde{\mathcal{E}}}}_{\max}$ and $U^{\tilde{\mathcal{E}}}_{\min}$ given the positions of the vertices. We then minimise the ratio $U^{\bar{\tilde{\mathcal{E}}}}_{\max}/U^{\tilde{\mathcal{E}}}_{\min}$ using Sequential Least Squares Programming

(SLSQP) [BGLS06]. In this minimisation, we also minimise the variance of all pairwise distances between vertices which are connected in $\tilde{\mathcal{G}}$. This is done to ensure that the interaction matrix between the atoms is close to the adjacency matrix of the graph. The result of this optimisation is shown in Figure 2.12.

2.4.2 Batching graphs together

The optimised positions of the atoms that have been calculated previously are called the *layout*. In principle, the layout can be exactly respected by using the very versatile Spatial Light Modulator (SLM) pattern that creates the micro-traps on the hardware. In practice however the SLM calibration step can be quite time-consuming (of the order of the minute). We therefore seek to use the same SLM pattern for many graphs in order to reduce the number of calibrations needed for the whole dataset. We start by clustering the dataset of graphs according to similarities in their structures. The idea behind this is to superimpose all identical structures between graphs on the same micro-traps. We focus on retrieving the presence and multiplicity of cycles and build a similarity measure between the graphs. For pentagons for example, the similarity can be written under the form:

$$s(\mathcal{G}_1, \mathcal{G}_2) = 1 - \exp\Big(-\alpha |N_1^P - N_2^P|\Big), \tag{2.17}$$

where N^P represents the number of pentagons in $\mathcal{G}$ and α is a hyper-parameter. We then use a linear combination of similarity measures to build a similarity matrix between all graphs of the dataset. Then, we apply a k-means clustering algorithm [C.04] using the similarity matrix to separate the graphs into different batches. For each batch, we are going to use a single SLM pattern. Furthermore, since the laser power is distributed over all the traps, we want to reduce the total number of traps, in order to maximise the intensity provided to each trap. This ensures that the traps are deep enough to obtain a satisfying filling efficiency ($\sim 55\%$) over the whole pattern. For each batch, we thus apply the following mapping algorithm:

Algorithm 4 Creating a triangular SLM pattern by batching M graphs

```
Require: Graphs $\{\mathcal{G}_1, \dots, \mathcal{G}_M\}$ in sorted sizes and optimised positions $\{x_1, \dots, x_M\}$
Ensure: Single SLM pattern that embeds $M$ graphs with optimal positions on a triangular lattice.
traps $\longleftarrow \{\}$
for $i$ in range $1, \cdots, M$ :
find $r_{\mathcal{G}_i} = \{r_1, \dots, r_{|\mathcal{G}_i|}\}$ triangular grid points that best conserve the pairwise distances between points in $x_i$ and maximises overlap with existing traps.
traps $\leftarrow$ traps $+ r_{\mathcal{G}_i}$ \traps
if |traps| $< 2|\mathcal{G}_M|$, add additional random triangular grid points to guarantee the filling property for re-arrangement.
```

Once we have clustered the graphs into different batches, we embed them onto an underling grid structure. In the case of the Smart-Charging dataset, we select 33 graphs that can be

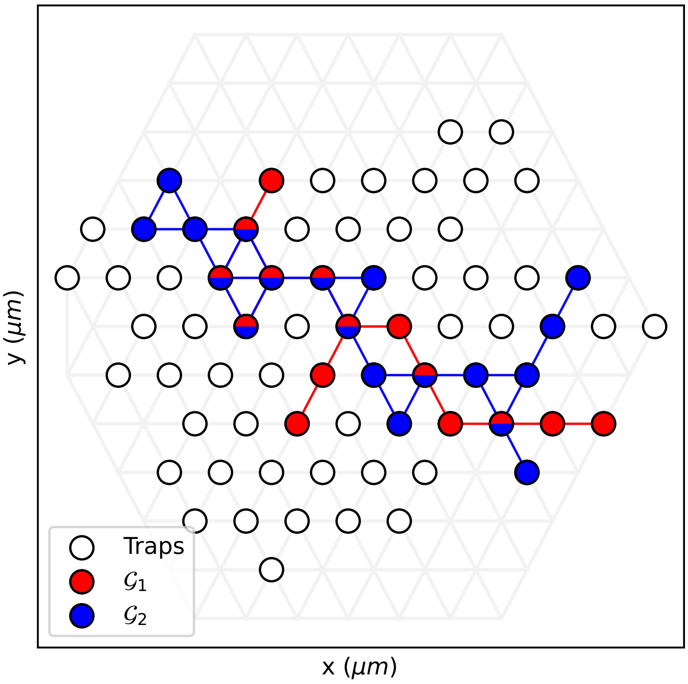


Figure 2.13: **Batching graphs on the same SLM pattern.** The batching method aims at superimposing graphs of a dataset on the same traps, created by a single SLM pattern. The graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are mapped to an underlying triangular lattice (grey intersection). By sharing common traps, a single SLM pattern (white circles) can be used to map many graphs; in this case, 16 graphs are mapped on the same underlying traps.

exactly embedded onto a triangular lattice such that the induced interaction matrix is a close to the adjacency matrix as it can be. Once all graphs have been embedded, the layouts are shifted and rotated to maximise the occupation of the same micro-traps. An example of overlapping graphs is given in Figure 2.13. In practice the graphs are ran sequentially, but all using the same SLM pattern.

To conclude this part, we have optimised the positions of the atoms in order to replicate as best as possible the adjacency matrix of the graphs under study. In a concern for hardware efficiency, we also grouped the different graphs in batches using the same SLM pattern. In this regime, the interaction matrix of the atoms replicate as closely as possible the topology of the embedded graph. This translates into the Rydberg blockade regime effectively blocking neighbouring atoms from being excited simultaneously. In other words, the quantum dynamics evolve only in the basis states that encode independent sets. This approximation, called the Truncated Hilbert Space (THS), enables us to simulate the dynamics of the ensemble of atoms only in the Hilbert sub-space spanned by the independent sets. The following step in our protocol is now to implement an evolution that guides the system towards the MIS of the graph. Before doing so, we have to go through the different sources of noise that can affect the hardware implementation.

2.4.3 Neutral-atom device noise model

To bring simulation closer to experience, the effects of various physical phenomena acting on the dynamics of atoms must be taken into account. In this section, we summarise the information contained in [Ldk18, DLBL+18]. The various noises can be categorised in different families. The first ensemble of errors are called state preparation and measurement errors (SPAM), which encompasses atom loading failure [Ldk18] but also false positive and false negative when measuring the quantum system at the end of the evolution. Once the atoms are trapped, the atoms are all prepared in the same state $|0\rangle$ by optically pumping them to one of the 5S hyper-fine energy level. During the dynamics, the tweezers are switched off but by the end of the process the measurement is done by switching on again the tweezers which effectively recaptures atoms in $|0\rangle$ and repels atoms in the Rydberg state $|1\rangle$. In this context, there are three types of errors that can occur:

1. **Optical pumping η.** This error accounts for the small inefficiency to load an atom in a specific 5S state. If the atom is loaded in another state, it will not follow the same dynamic as the others. A first approximation is to consider that it will not experience anything and will be measured at all times as being in $|0\rangle$. A second approximation would be to make it undergo the dynamics but with a consequent detuning to include the difference in energy levels. This error effect is quantified by η, which represents the fraction of inefficiency of the initial state preparation and is typically of the order $0.5-5\%$.

2. **False positive** ε. False positive error accounts for the probability of losing an atom while the dynamics is performed, due to collisions or thermal motion, or during the fluorescence imaging. This loss amounts to misinterpret a $|0\rangle$ for a $|1\rangle$ and is quantified by ε and is typically in the range $1-5\%$.

3. **False negatives ε'.** False negative error accounts for the probability that an atom in $|r\rangle$ quickly decays to $|g\rangle$ during the recapture duration. This effect depends on the Rydberg state chosen as $|r\rangle$. This unintended acquisition amounts to misinterpret a $|r\rangle$ for a $|g\rangle$, measuring a 1 instead of a 0, and is quantified by ε' and is typically in the range $3-7\%$.

These errors happen in the state preparation and the measurement steps, but there are also sources of noise that affect the dynamics when applying pulses to the ensemble of atoms. First, the pulses are applied using excitation lasers that have themselves several sources of noise. For example, they do not have a uniform profile: the electric field they shine on the ensemble of atoms decreases with the distance to the focal point. Therefore, the Rabi frequency applied to the ensemble of atoms has a spatial gradient and for a given atom at distance r from the focal point, the perceived Rabi frequency is given by:

$$\Omega(r) = \Omega_0 \exp\left(-\frac{r^2}{w^2}\right), \tag{2.18}$$

where w is called the *waist* of the laser and its typical value is $w = 170\mu m$ [Ldk18]. Furthermore, the centre of this Gaussian profile can be inaccurate and can change from one shot to another. This can be modelled as a centred Gaussian distribution $\mathcal{N}(0, \sigma_f)$ with a standard deviation $\sigma_f = 1\mu m$. Finally, the Rabi frequency the laser produces is not the same at each shot, considering that its typical variation time is greater than the duration of a shot. This effect can be modelled by multiplying the value Ω by a Gaussian distribution with mean 1 and standard deviation σ_Ω. This deviation is typically of order of a few percent with a lower bound of 1% achieved with feedback loops. Another source of noise comes from the finite temperature inside of the vacuum chamber. The temperature's effect on the dynamics is two-fold:

1. **Atom motion.** Atoms do not remain perfectly still in the tweezers because of finite temperature, which is related to their kinetic energy. In the trap plane, this random motion is usually modelled as a Gaussian law around the trap centre with a deviation given by $\sigma_r = \sigma_v/\omega$ where ω is the trapping angular frequency. In the orthogonal direction, according to experiment, the position uncertainty is said to be 4 times stronger. Thus $\mathbf{r} = (x, y, z) \sim \mathcal{N}(\boldsymbol{r_0}, \boldsymbol{\sigma})$ with $\boldsymbol{\sigma} = (\sigma_r, \sigma_r, 4\sigma_r)$. This motion affects differently for each shot the global Hamiltonian in two separate ways: the dipole-dipole interaction strength depends on the distance between atoms and the Rabi frequency has a spatial gradient. The first effect can be neglected if the inter-atomic distance R is such that $\sigma_r \ll R$. The second effect can be neglected if $\sigma_r \ll w$ where w is the waist of the Gaussian profile. Setting the trapping angular frequency at $\omega = 2\pi \times 0.1$ MHz gives $\sigma_r(T) = 0.12\ \mu m \times (\frac{T}{60\mu K})^{1/2}$.

2. **Doppler shift.** Given a temperature T, the atoms move with a velocity v that follows a centred Gaussian distribution with standard deviation $\sigma_v(T) = \sqrt{k_B T/m}$. The motion of the atoms then alters the lasers' frequency they "see". Defining the combined wave vector $\vec{k}_{eff} = \vec{k}_r + \vec{k}_b$, the Doppler shift induced on the two-photon detuning δ is distributed according to a Gaussian of parameters $\mathcal{N}(0, \sigma_\delta(T) = ||\vec{k_{eff}}||\sigma_v(T))$. Thus for each new shot, the lasers are slightly off resonant and the dynamics of the system differs by the value of the two-photon detuning perceived. The amplitude of these variations must be compared to the amplitude Ω.

All in all, three types of errors have to be taken into account when looking at the dynamics of an ensemble of atoms: SPAM errors, finite temperature effects and laser noises. All these errors are grouped in the Table 2.1

Taking into consideration all these noises is important when trying to estimate the quality of the hardware in solving the MIS problem on Unit-disk graphs. In the following part, we explicit the pulses that will be applied on the hardware in order to prepare a good ground-state to solve MIS.

SPAM errors				
ε	False positives	Probability to lose an atom in $\lvert g\rangle$ during sequence.	$\mathcal{P}_{1\to 0} = \varepsilon(1-\varepsilon)^{n_1-1}(1-\varepsilon')^{n_0}$	$1\% - 5\%$
ε'	False negatives	Probability to recapture an atom in $\lvert e\rangle$ before expelling it from the trap.	$\mathcal{P}_{0\to 1} = \varepsilon'(1-\varepsilon)^{n_1}(1-\varepsilon')^{n_0-1}$	$3\% - 7\%$
η	Optical pumping	Probability to initially not pump the atom in $\lvert g\rangle$.	$(1-\eta)^N\ H + \eta(1-\eta)^{N-1}\ \sum_i H_{/i}$	$0.5\% - 5\%$
Temperature				
σ_δ	Doppler shift	Detuning variation due to random motion of atoms and lasers orientation.	sts : $\delta + \mathcal{N}(0, \sigma_\delta)$ with $\sigma_\delta = \sqrt{k_B T/m}\ k_{eff}$	660 kHZ for $T = 60\ \mu K$
σ_r	Position	Atoms move randomly in traps due to their temperature.	sts : $pos + \mathcal{N}(0, \sigma_r, 2)$ with $\sigma_r = \sqrt{k_B T/m}/\omega_{trap}$	120 nm for $T = 60\ \mu K$
Laser spec				
σ_Ω	Amp. inaccuracy	Variation of the laser amplitude between shots	sts : $\Omega \times \mathcal{N}(1, \sigma_\Omega)$	$1\% - 5\%$
w	Amp. waist	Amplitude varies with position according to a Gaussian envelope maximal at the barycenter	$\Omega(r) = \Omega \exp(-\lvert\lvert r - r_0\rvert\rvert^2/w^2)$	$\sim 170\ \mu m$
σ_0	Focus inaccuracy	Variation of position of the focus point r_0 of the laser.	sts : $r_0 + \mathcal{N}(0, \sigma_f, 2)$	$1\ \mu m$

Table 2.1: **Main sources of noise in a neutral-atom device.**

2.4.4 Constraints on an adiabatic preparation

While the theoretical study in the noiseless case was done using QAOA, the lack of closed-loop optimisation on the current device makes it quite cumbersome to implement. Furthermore, continuously modifying over time the Rabi frequency $\Omega(t)$ and the detuning $\delta(t)$ is a natural parametrisation of the laser pulse sequence. For completeness, we recall the formulation of the Ising Hamiltonian that is parameterised by the Rabi frequency $\Omega(t)$ and the detuning $\delta(t)$:

$$H = \sum_{i=1}^{n} \frac{\hbar\Omega(t)}{2}\hat{\sigma}_i^x - \sum_{i=1}^{n} \frac{\hbar\delta(t)}{2}\hat{\sigma}_i^z + \sum_{j<i} \frac{C_6}{|\mathbf{r}_i - \mathbf{r}_j|^6}\hat{n}_i\hat{n}_j. \tag{2.19}$$

Recall that by embedding the atoms $\{\mathbf{r}_i\}_{i\in|\mathcal{V}|}$ onto a triangular grid, we have ensured that the interaction term of the Hamiltonian (third term) embeds the adjacency matrix of the graph. This means that when $\Omega(t) = 0$ and $\delta(t) > 0$, the ground-state of H corresponds to the MIS (we suppose it's unique for now). The next step is therefore to devise a protocol that prepares the ground-state of $H(\Omega = 0, \delta > 0)$. Inspired by the preparation of antiferromagnetic (AFM) states on square and triangular lattices presented in [SSW+21], we design with Pulser [SGD+22] an

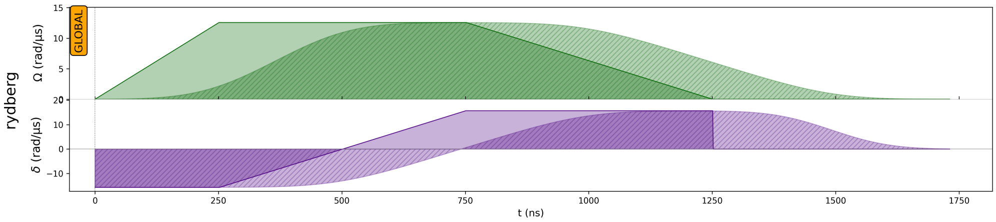


Figure 2.14: **Adiabatic ramp pulse for MIS preparation.** Let $\mathcal{G}$ be a graph of n vertices with a unique MIS represented in the computational basis state as $|S\rangle$. At $t = 0$, the groundstate of the Hamiltonian is $|0\rangle^{\otimes n}$. Then the Rabi frequency $\Omega(t)$ ramps up from 0 to $\Omega_{\max}$, adding excitations to the ground-state on its way. The value of $\Omega_{\max}$ is chosen so as to respect the THS approximation. At $t = t_f$, the Hamiltonian is diagonal and the value of the final detuning is chosen to be $\delta_f = \frac{d_{\max}}{d_{\max}-1} U_{NN}$, where $d_{\max}$ is the maximum degree of $\mathcal{G}$. This ensures that $|S\rangle$ is the ground-state of $H(t = t_f)$. Because the pulses are controlled by an Acoustico-Optic Modulator (AOM), they are subject to the finite bandwidth deformation effect depicted as the shaded area.

adiabatic ramp sequence. Our motivation for using this protocol is that AFM states correspond to Maximum Independent Sets on regular lattices. Interestingly in our case, we have pre-processed the graphs on the same type of regular lattice structure (triangular). The sequence is represented in Figure 2.14 and is built in the following way:

- linearly ramping up $\Omega(t)$ from 0 to $\Omega_{\max}$ while maintaining $\delta(t)$ at $\delta_0 < 0$ for t_1,
- linearly ramping up $\delta(t)$ from δ_0 to δ_f while maintaining $\Omega(t)$ at $\Omega_{\max}$ for t_2,
- linearly ramping down $\Omega(t)$ from $\Omega_{\max}$ to 0 while maintaining $\delta(t)$ at δ_f for t_3.

Some intuition can be built to understand why we choose this pulse. Suppose we have n atoms that encode a graph $\mathcal{G}$ with a single MIS. At $t = 0$, the negative detuning $\delta(0) = \delta_0 < 0$ and $\Omega(0) = 0$ ensures that the ground-state at $t = 0$ is $|0\rangle^{\otimes n}$, which corresponds to the state obtained after the optical pumping. At final time $t = t_f$, the detuning is now positive ($\delta_f > 0$) and $\Omega(t_f) = 0$ which means that the ground-state of $H(t_f)$ is the computational basis state that corresponds to the MIS if δ_f is in the correct range of values. Note that if δ_f is much greater than the van der Waals interaction between two atoms, then the excitation of the two atoms becomes the groundstate. We therefore have an easy-to-prepare groundstate at $t = 0$ and the wanted MIS encoded in the groundstate of the Hamiltonian at $t = t_f$. The pulse is called adiabatic because the parameters $\Omega(t), \delta(t)$ vary slowly enough to ensure that at all times the quantum state stays in the ground-state of the instantaneous Hamiltonian. If this adiabatic process is done slowly enough, one should be able to measure the MIS with high probability at $t = t_f$. In a following chapter, we will define more formally this adiabatic approach and try to understand in which cases we can expect a quantum speed-up when using this method.

The pulse is characterised by 6 parameters : $\Omega_{\max}$, $\delta_0, \delta_f, t_1, t_2$ and t_3. We want to have enough power (given by Ω) to overcome the strongest interaction between two no link-sharing atoms. We also want $\Omega_{\max}$ to be below the interaction between two atoms sharing an edge in $\mathcal{G}$ to ensure that we stay in the independent set regime. Because all graphs sit on an underlying triangular lattice with spacing $5\mu m$, we want $\Omega_{\max}$ to be below the nearest-neighbour (NN) interaction

$$U_{\mathcal{E}} = U_{NN} = C_6/5^6 \approx 2\pi \times 9\text{MHz}$$

but also want $\Omega_{\max}$ to be above the next-nearest neighbour (NNN) interaction given by

$$U_{\bar{\mathcal{E}}} \leq U_{NNN} = U_{NN}/\sqrt{3}^6 = 2\pi \times 0.3\text{MHz}$$

In the case of a triangular layout, we choose $\Omega_{\max}$ in the range

$$\Omega_{\max} \in [0.3, 9] \times 2\pi\,\text{MHz}, \tag{2.20}$$

We also want to choose the range of values for δ_f such that the MIS is the groundstate of our Hamiltonian at $t = t_f$. Let S be the MIS of size s represented by the computational basis state $|S\rangle$ such that $H|S\rangle = E(S)|S\rangle$ when $\Omega = 0$. In that case,

$$E(S) = -s\delta_f + \frac{1}{2}\sum_{i,j\in S} U_{i,j}, \tag{2.21}$$

is the energy associated to the MIS[5]. We now consider the change in energy if we add or delete an excitation i_0 from $|S\rangle$. Adding an excitation yields

$$E(S + i_0) = -(s+1)\delta_f + \frac{1}{2}\sum_{i,j\in S} U_{i,j} + \sum_{i\in S} U_{i,i_0}, \tag{2.22}$$

where the third term encodes the interaction energy between i_0 and the other excited atoms of S. In the case where we delete an excitation from S, the energy function reads

$$E(S - i_0) = -(s-1)\delta_f + \frac{1}{2}\sum_{i,j\in S} U_{i,j} - \sum_{i\in S, i\neq i_0} U_{i,i_0}. \tag{2.23}$$

To derive an upper bound we can use the fact that we want to ensure that for any vertex $i_0 \notin S$,

$$E(S) < E(S + i_0),$$

which implies therefore that $\delta_f < \min_{i_0\notin S}\left(\sum_{i\in S} U_{i,i_0}\right)$. In the worst case, i_0 is linked to only one atom of S and is far away from the others and we obtain the constraint

$$\delta_f < U_{NN}. \tag{2.24}$$

To derive a lower bound, we use the fact that we want

$$\forall i_0 \in S,\ E(S) < E(S - i_0),$$

---

[5] Note that in the hard-sphere approximation, the second term is equal to zero.

which gives us the constraint

$$\delta_f > \max_{i_0 \in S} \left(\sum_{i \in S, i \neq i_0} U_{i,i_0} \right) > 0. \tag{2.25}$$

On a triangular lattice, the worst case is to get 6 next-nearest neighbours, which corresponds to $\delta_f > 6U_{NNN} = 1.8 \times 2\pi\text{MHz}$. The bounds for δ_f are therefore

$$\delta_f \in [6U_{NNN}, U_{NN}]. \tag{2.26}$$

In practice, we want to take the highest value possible of δ_f in order to favour $|S\rangle$ as much as possible. For this reason, we decide to set

$$\delta_f = \frac{d_{\max}}{d_{\max}+1} U_{NN}, \tag{2.27}$$

where $d_{\max}$ is the maximum degree of $\mathcal{G}$. For δ_0, we take $\delta_0 = -\delta_f$.

2.4.5 Improvements using Bayesian optimisation

Pulses on the hardware can actually be shaped more precisely than simple ramping up or down the lasers in both intensity and frequency. With a bandwidth of few MHz, Acousto-Optical Modulators (AOM) allow to modulate the Rabi frequency $\Omega(t)$ and the detuning $\delta(t)$ at $\sim 100ns$ scale and run smoother pulses. To leverage this flexibility, we decide to optimise the adiabatic pulse by parametrising the timescale of the pulse at m different points in time. Precisely, we divide the total pulse duration T in m equal durations and define $2m$ control points $\{\Omega_{(t_i)}, \delta_{(t_i)}\}_{i=1\cdots m}$ where $t_i = T(i-1)/(m-1)$. The total time T can also be used as a tunable parameter. The control fields are finally derived using interpolation by monotonic cubic splines which directly maps a set of parameters to implementable pulses (see Fig.2.15). The parameter space is bounded by the same constraints that we presented in the previous section but are also hardware-related. For example, the maximum coherence time of the machine or the available laser power can fix the upper bounds on T and Ω. Given the constraints, we use an optimisation method called Bayesian optimisation [Fra18]. The Bayesian method seeks optimal parameters given a cost function C and is known for its robustness against noisy evaluations, its efficiency already with few iterations and its gradient-free approach. The algorithm consists of both a statistical model that simulates the unknown landscape function and a decision maker, the acquisition function which indicates where the next guess will be most likely to provide useful information [Fra18]. Bayesian optimisation methods have been used in many experimental set-ups to prepare quantum states with higher-fidelity than naive approaches [MSX+20].

On this specific dataset, we choose $m = 5$ and the following hardware constraints:

- $0 \leq \Omega/2\pi \leq 2$ MHz,

- $-5 \leq \delta/2\pi \leq 5$ MHz,
- $T_{min} \leq T \leq T_{max}$ with $T_{min/max} = 0.5, 1.25\mu s$.

We perform the optimisation over the reduced family of mapped graphs with $|\mathcal{V}| < 15$, which we call the training set, so that the optimisation can be done on a classical CPU. We initialise the optimiser with $n_r = 30$ random guesses using the Latin Hypercube sampling [Loh96] of the parameter space and set a maximum budget of $n_c = 300$ iterations. The optimised pulse lasts $T = 1.18\mu s$ ($1.70\mu s$ with finite bandwidth) and ends at $\delta_f/2\pi = 4$ MHz.

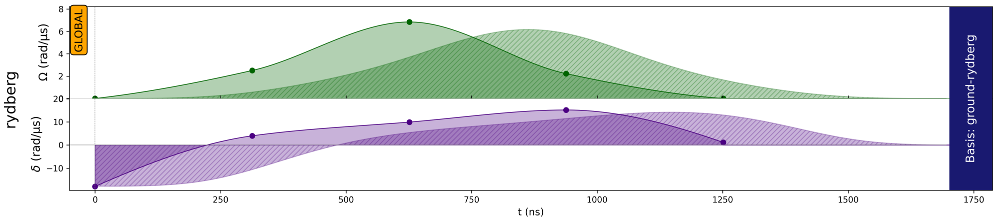


Figure 2.15: **Instance of interpolated sequence with $m = 5$.** A Bayesian optimisation outputs optimal value for the pulse $T*$ and the corresponding values of $\Omega(t)$ and $\delta(t)$ at m time-points of the pulse $t_i = T(i-1)/(m-1)$. The values of Ω are fixed initially and at the end to zero. The dinal control fields are derived using interpolation by monotonic cubic splines at the optimal points. As these sequences are also produced by the AOM, they are subject to the same finite bandwidth deformation effect that we saw for the ramp adiabatic pulse.

The optimised pulse is represented in Figure 2.16. It seems that the area under the Rabi frequency is the same, while the detuning profile is sharper in the case of the optimised pulse.

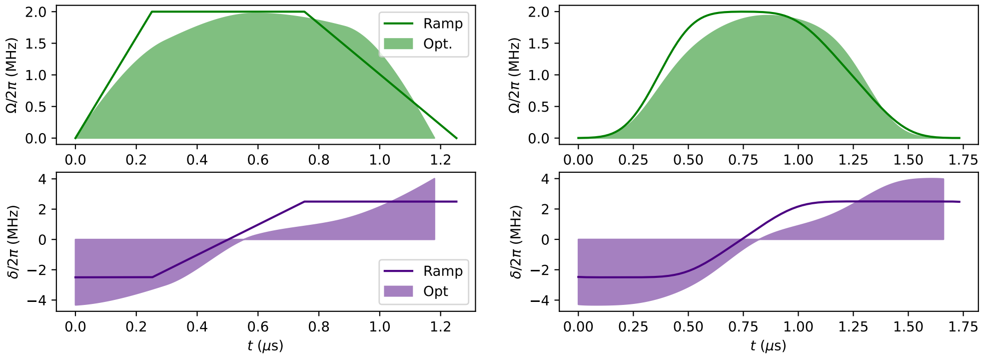


Figure 2.16: **Optimised pulse profile using Bayesian optimisation for MIS preparation on the training dataset** (left/right) Pulse before/after applying limited bandwidth. (line/filled) Ramp and optimised sequence with (top/bottom) amplitude/detuning.

2.4.6 Results

Considering the dataset of graphs mapped to a triangular underlying lattice, we compare the two protocols for the preparation of MIS states; the simulations are performed with the finite bandwidth effect. We measure the quality of each protocol by looking at the the probability of measuring the MIS in the final state. The two types of pulses yield high probabilities of measuring the Maximum Independent Set and the optimised pulse achieves remarkable results, with a mean 80% of measuring the MIS on the training dataset.

In Fig.2.17(a), the probability to measure a MIS state, $\mathbb{P}$(MIS) after applying the ramp (blue) or the optimised pulse (red) is shown for each graph. The dataset is split between graphs used or not for the optimisation, i.e. with $|\mathcal{G}| < 15$ (i) or $|\mathcal{G}| \geq 15$ (ii). Around the cut between (i) and (ii), we plot $\mu \pm \sigma$ for both protocols and for the 3 distributions (left) $|V| < 15$, (right) $|V| \geq 15$ and (middle) for all graphs in thee dataset. For all graphs, the optimised pulse increases $\mathbb{P}$(MIS), even for graphs it was not trained on. This counter-intuitive result stems from the shared topology (density, neighbourhood, mapping) between all graphs of the selected family. Thus, the optimised pulse is specifically tailored for MIS preparation on UD graphs lying on a triangular lattice. The improvement obtained with the optimised pulse compared to the ramp pulse is on average of 19% for all 3 distributions meaning that we are not losing efficiency by applying the trained pulse on the larger test part of the dataset.

In Fig.2.17(b), we shows the distributions of $\mathbb{P}$(MIS) in a more global form and we also analyse the effect of measurement errors on those distributions. On the first line, using the optimised pulse on the trained part instead of the ramp shifts the distribution towards values of $\mathbb{P}$(MIS) closer to 1 while including the whole dataset adds a smooth tail ending around $p_- := 36\%$. This value is the minimum probability of measuring a MIS on the dataset, meaning that performing $N_{\text{shots}} > \log(1-F)/\log(1-p_-) \approx 15.5$ assures to measure the MIS with fidelity $F = 99.9\%$. For the ramp, this leads to $N_{\text{shots}} > 31$.

On the second line, we modify the distributions by including *a posteriori* measurement errors. Those can be summarised by $\varepsilon = 3\%$ and $\varepsilon' = 8\%$, respectively the false positive and false negative probabilities. The probability of measuring the bitstring j instead of i is derived with:

$$P_{j|i} = \prod_{k}^{N} (1 - |i-j|_k) - (-1)^{|i-j|_k} \left[(1-i_k)\varepsilon + i_k\varepsilon'\right]. \tag{2.28}$$

The measurement errors mean that a prepared MIS can be detected as a MIS+1 if a bit is flipped from 0 to 1 with ε and can be detected as an IS if a bit is flipped from 1 to 0 with ε'. The global effect of the SPAM is to compress the distributions while shifting them towards lower values of $\mathbb{P}$(MIS). Indeed, the random bit-flips tend to bring different distributions closer together. Interestingly, the improvement from ramp to optimised pulse is less pronounced with

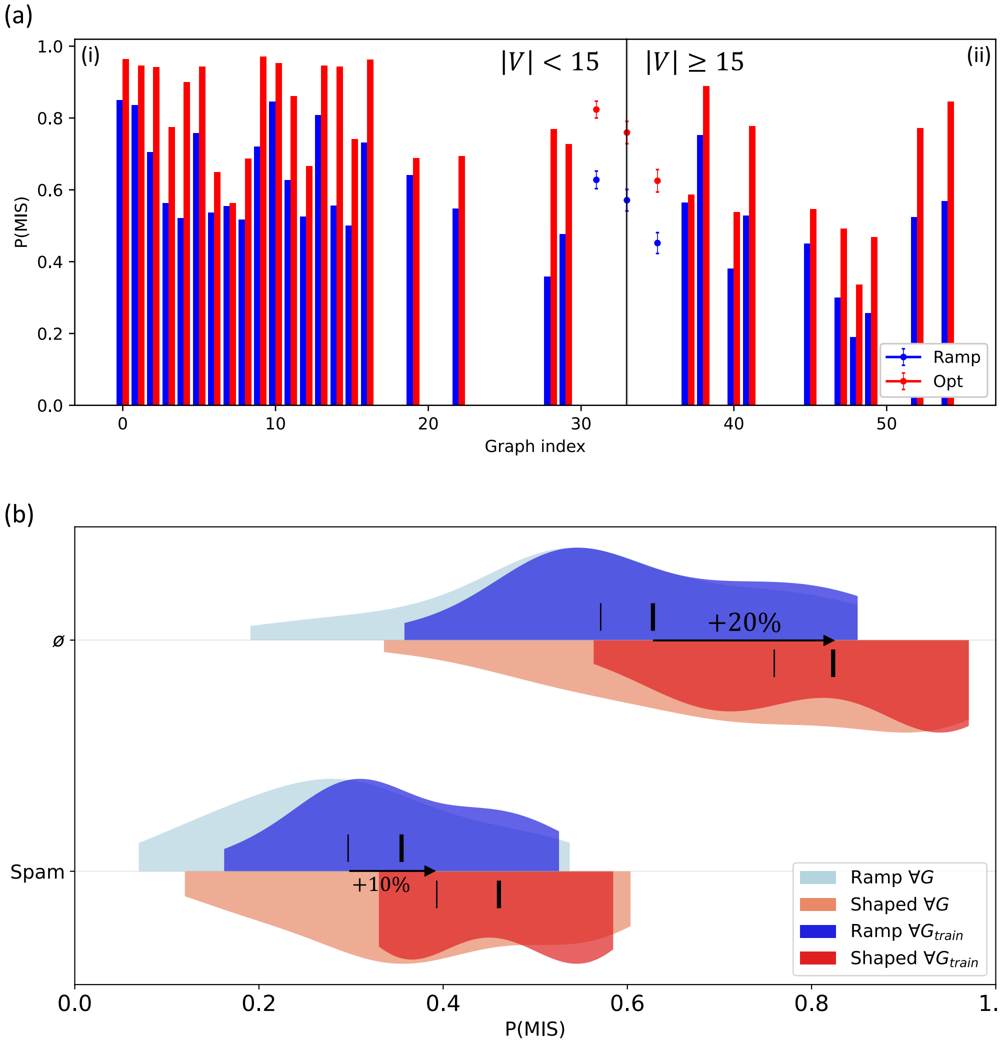


Figure 2.17: **Probability of measuring the MIS in the final state after a ramp adiabatic pulse and an optimised pulse on the EDF dataset** (a) Histograms of $\mathbb{P}$(MIS) for each graph, sorted by size. The mean value and standard deviation (dots) are plotted for all graphs with sizes $|\mathcal{G}| \leq 15$. The same pulses are then applied to the graphs with a higher number of vertices, achieving decent results. (b) The violin-plots of $\mathbb{P}$(MIS) show the distribution over the dataset. G_{train} corresponds to the graphs with size ≤ 15. The mean of each distribution are marked with black lines (thick = training dataset, thin = whole dataset). We see that the optimised pulses achieve an overall +20% score compared to the ramp adiabatic pulse. The impact of SPAM errors is also plotted and reveals that the optimised pulse yields in practice +10% in preparation quality.

only a 10% increase on average over the whole dataset. Crudely estimating that the MIS for each graph consists in exciting half the sites, we can estimate that $\mathbb{P}(\mathrm{MIS})$ will be altered by a factor $(1-\varepsilon)^{N/2}(1-\varepsilon')^{N/2}$. On average on the whole dataset, this quantity is around 53% with the spam values considered, explaining the factor 2 between the improvement ramp-optimised with and without SPAM. In a less expected way, it seems that some instances among the bigger graphs are less prone to SPAM errors.

2.5 Discussion and open problems

In the previous sections, we have demonstrated the capacity of neutral atom technology to prepare a quantum state that encodes the MIS of unit-disk graphs. First, we developed a method to map the atoms to positions that replicate as closely as possible the adjacency matrix of the underlying graph. Then, we implemented a batching method that enables to use the hardware efficiently regarding SLM calibration. Finally, we have shown two methods to prepare the ground-state of the Ising Hamiltonian inspired by adiabatic approaches that take into account the main noise sources of the hardware. Overall, we expect to see encouraging results on hardware and need to increase the dataset of UD-graphs and their size to gain intuition on the capacities of neutral-atom based methods. Impressive efforts in this direction have been achieved in reference [EKC+22] where UD graphs up to sizes $|\mathcal{V}| = 289$ have been achieved experimentally with neutral atoms. While it is already interesting to see that quantum methods can be used to address the instances of the UD-MIS problem, it is important to benchmark these results against classical algorithms to understand to what extent we can hope for a quantum speed-up using neutral atoms. In the following chapter, we define and compare classical approaches to solve **NP**-hard problems like max-k-cut and MIS on UD graphs.

A first challenge in comparing our approach to classical counterparts is the definition of an algorithmic success metric. Given the heuristic and probabilistic nature of the quantum method, a reasonable metric is to use the *expected approximation ratio* R of the output state defined by:

$$R = \frac{\langle \hat{C} \rangle}{C_{\text{opt}}} = \frac{\langle \psi_f | \hat{C} | \psi_f \rangle}{\min_{\mathbf{z}} C(\mathbf{z})}, \tag{2.29}$$

where $|\psi_f\rangle$ is the final quantum state and $\langle \hat{C} \rangle$ is an average of the cost function over the N_{samples} quantum measurements. We used this metric in the first part to showcase the quality of the QAOA approach in section 2.2. While this metric gives a good indication of the quality of the overall state in a noiseless setting, in practice SPAM errors can alter samples into poor solutions and artificially lower the average approximation ratio. Furthermore in practice one would return the best solution among all samples. We therefore define the *truncated approximation ratio* R_p as the approximation ratio when only considering the top p% best shots:

$$R_p = \frac{\langle \hat{C} \rangle_p}{C_{\text{opt}}}, \tag{2.30}$$

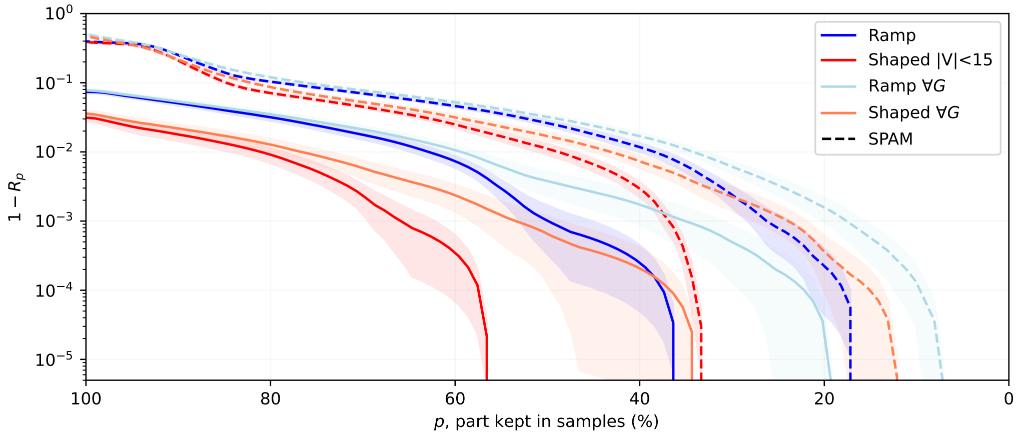


Figure 2.18: **Evolution of truncated approximation ratio with p.** $1 - R_p$ decreases with p towards 0 as only the best part of the samples are taken into consideration. The highest value p^* at which it hits 0 is the minimum proportion of samples to keep to only sample MIS states. Lines represent the mean over distributions while light areas represent the std.

where

$$\langle \hat{C} \rangle_p = \sum_{i=1}^{\lfloor pN_{\text{samples}} \rfloor} C(\mathbf{z}_i), \tag{2.31}$$

considering that the samples are sorted by cost quality. With this definition, the expected approximation ratio corresponds to keeping $p = 100\%$ of the samples. In Figure 2.18 we plot the evolution of the truncated approximation ratio for the EDF dataset as a function of p. We observe that truncating the samples to the best 50% when applying the optimised pulse to the whole dataset in presence of SPAM errors neglects these sources of error and enables to reach the same value as $R_1 = 96.5\%$ in the noiseless case.

Another interesting addition to the approximation ratio is to take into consideration the execution time of the algorithm. In this case, we can also look at the *maximum approximation ratio* defined in [SMA20] as

$$R(n_{\text{samples}}) \equiv \max_{i \in \{1,\dots,n_{\text{samples}}\}} \frac{C(z_i)}{C_{\text{opt}}}. \tag{2.32}$$

As our quantum approach is a heuristic and can yield sub-optimal results, it is natural to compare it to classical approximation schemes. In the following chapter, we will dive into more detail in the classical methods that have been developed to tackle these **NP**-hard problems.

Chapter 3

Unravelling the complexity of Maximum Independent Set: insights from statistical physics, quantum annealing and approximation schemes

This chapter embarks on a comprehensive exploration of the intricacies surrounding the challenging problem of Maximum Independent Set (MIS) from both computational complexity and approximation perspectives. We begin by delving into the theoretical foundations of MIS as an NP-hard problem in section 3.1 and leverage insights from statistical physics to gain a deeper understanding of its inherent complexity on random graphs. Thanks to their better understanding, physicists have proposed a heuristic solver called *simulated annealing* that we detail in section 3.2. Next, we delve into the realm of quantum annealing in Section 3.3 and its potential to address MIS. We present instances where exponential annealing time is expected, shedding light on the limitations of this quantum computational approach. Shifting our focus, we investigate the notion of approximation schemes in section 3.5 and their applicability to MIS on specific types of graphs. Finally, we propose in section 3.6 a method that utilises neutral atoms to map graphs that are out of reach for efficient classical approximation schemes. It is part of the paper

- [DHK+23] Dalyac C., Henry LP., Kim M., Ahn J., Henriet L. *Exploring the impact of graph locality for the resolution of MIS with neutral atom devices*, *arXiv preprint arXiv:2306.13373.* (2023)

Through this interdisciplinary examination of MIS, statistical physics, quantum annealing and approximation schemes, this chapter offers insights into the challenges and potential breakthroughs in solving MIS, ultimately paving the way for future advancements in quantum computing with neutral atoms.

Computational complexity is a branch of computer science that reveals many fascinating results about the hardness of solving problems and the types of algorithms that can be used to do so efficiently. In 1945, the first digital computer named the Electronic Numerical Integrator and Computer (ENIAC) came to use [Mah88]. The ENIAC was a large, room-sized machine that used vacuum tubes to perform calculations. It was capable of performing a variety of complex computations, including ballistics calculations for the US military. Weighing in at 30 tons the machine contained over 18,000 vacuum tubes, 1,500 relays, and hundreds of thousands of resistors, capacitors and inductors [Wei55]. Functioning thanks to 200 kW of electric power, the ENIAC regularly caused blackouts in the city of Philadelphia. It was able to add or subtract 5000 times a second, making it a thousand times faster than any other machine at that time: before the development of the ENIAC, calculations were performed by human "computers" (typically women) using mechanical calculators. For tasks such as predicting the trajectory of a shell, these calculations had to take into account variables such as air density, temperature or wind and a single trajectory took between 20 to 40 hours to compute by hand [GH20]. With the introduction of ENIAC however, the same calculations took under a minute. As it could be reprogrammed for many different problems, the ENIAC and its successors were quickly overwhelmed with demands of calculations from scientists across the world. Physical limits however came into play: it could take several days to reprogram the ENIAC to run another algorithm and the reliability was quite poor because of heating fluctuations. Indeed, many vacuum tubes burned out almost every day, leaving the device nonfunctional about half the time. In order to make the most out of the up-time of the ENIAC, scientists tried to find out more efficient algorithms that could solve the calculations faster. For some problems quicker algorithms were found easily, but other problems seemed to resist and seemed intrinsically hard. After many unsuccessful attempts, it seemed that finding the most efficient algorithm for a given problem was not so trivial, and started to bother scientists like Alan Cobham who mentioned this struggle during a conference in 1964 [Cob65]:

> *The subject of my talk is perhaps most directly indicated by simply asking two questions: first, is it harder to multiply than to add? and second, why? ... I (would like to) show that there is no algorithm for multiplication computationally as simple as that for addition, and this proves something of a stumbling block.*

In this particular context, a significant area of focus involves looking at combinatorial optimisation problems [PY91]. Instances of these problems are defined with a cost function C taking a finite number of values defined over n variables. Combinatorial optimisation seeks to determine whether families of such cost functions can be categorised in a way that allows algorithms to find their global minimum through polynomial many steps in n. The dominant conjecture as of today is that certain families of cost functions exist for which no polynomial-time algorithm can achieve this objective, a notion commonly referred to as the well-known **P≠NP** conjecture [Coo23]. While **P** optimisation problems can be solved in polynomial-time, the two problems we have looked at in the previous chapter, namely max-k-cut and maximum independent set (MIS), are two problems that are believed to be **NP**-hard [Spi99, GJ78]. Let us build some intuition on why these problems are hard to solve. First of all, not all instances of NP problems are hard: solving MIS for example on path graphs or rooted-tree graphs takes polynomial time [CV86]. For other families of graphs however like Erdos-Renyi random graphs, it seems very hard to find solutions in polynomial time [Fri90].

3.1 MIS on random graphs is hard

Let $\mathcal{G}(n,m)$ be a graph on n vertices obtained by choosing m edges uniformly at random (without replacement). We call *independence number* $\alpha(\mathcal{G}(n,m))$ of the graph the size of its maximum independent set. If we let $d = 2m/n$ denote the average degree, then it can be shown that for $1 \ll d \ll \sqrt{n}$ [Fri90]:

$$\alpha(\mathcal{G}(n,m)) \simeq 2n\frac{\ln d}{d}. \tag{3.1}$$

Astonishingly, there is no known polynomial algorithm that can find an independent set of size $(1+\varepsilon)n\ln(d)/d$ for any $\varepsilon > 0$ in the worst case. In other words, it seems that it is computationally difficult to find an independent set that is bigger than half the size of the MIS. An amusing fact is that a simple greedy algorithm that constructs an inclusion-maximal independent set over $\mathcal{G}$ yields with high probability an independent set of size $n\ln(d)/d$ [GM75] and has never been beaten after almost fifty years of research on the subject. Coja-Oghlan and Eftymiou have put forward an explanation to this apparent hardness: there is a phase transition in the structure of the problem that occurs when the size of the independent sets passes the point $n\ln(d)/d$ [COE15]. For a graph $\mathcal{G}$ and an integer $k \in \mathbb{N}$, we let $\mathcal{S}_k(\mathcal{G})$ denote the set of all independent sets in G that have size exactly k. For each size k, we create an auxiliary graph where each independent set is represented by a node, and two independent sets are connected by an edge if their Hamming distance is inferior to a constant (that does not depend on n). Mathematically, we define for two independent sets A and B the symmetric difference $A\Delta B := A\cup B - A\cap B$ such that $d_H(A,B) = |A\Delta B|$ where d_H is the Hamming distance between the two ensemble seen as vectors of size n. We say that $\mathcal{S}_k$ is $\gamma-$connected if for any two independent sets $\sigma_A, \sigma_B \in \mathcal{S}_k$ there exists a path $\sigma_1, \ldots, \sigma_N$ such that $\sigma_1 = \sigma_A$ and $\sigma_N = \sigma_B$ and $d_H(\sigma_i, \sigma_{i+1}) \leq \gamma$ for all $1 \leq i < N$. Conversely, we can also define the inverse notion called *shattering*, which describes an ensemble $\mathcal{S}_k$ that can be partitioned into unconnected subsets. Precisely,

Definition 3.1. *A set $\mathcal{S}_k(\mathcal{G}(n,m))$ shatters if there exists constants $\gamma, \zeta > 0$ such that with high probability the set $\mathcal{S}_k(\mathcal{G}(n,m))$ admits a partition into subsets such that*

1. *Each subset contains at most $e^{-\gamma n}|\mathcal{S}_k(\mathcal{G}(n,m))|$ independent sets*
2. *If A, B belong to different subsets, then $d_H(\sigma_A, \sigma_B) \geq \zeta n$.*

Given these definition, the authors prove the following theorem:

Theorem 3.1 ([COE15]). *There exists $\varepsilon_d, \varepsilon'_d \to 0$ and for any d a number $C_d > 0$ (independent of n) such that*

(i) $\mathcal{S}_k(\mathcal{G}(n,m))$ is C_d-connected with high probability for any

$$k \leq (1-\varepsilon_d)n\frac{\ln d}{d}, \tag{3.2}$$

(ii) $\mathcal{S}_k(\mathcal{G}(n,m))$ shatters for all k such that

$$(1+\varepsilon'_d)n\frac{\ln d}{d} \leq k \leq (2-\varepsilon'_d)n\frac{\ln d}{d}. \tag{3.3}$$

This theorem shows that there is a critical point in k when looking at the structure of $\mathcal{S}_k(\mathcal{G}(n,m))$: if $k < n\ln d/d$, then S_k is well connected and local algorithms like simple Markov chains are ergodic. However at the critical point $k = n\ln d/d$ the auxiliary graph associated to $\mathcal{S}_k$ shatters to exponentially many unconnected subgraphs of exponentially small sizes. While these results give information on the structure of $\mathcal{S}_k(\mathcal{G}(n,m))$, it is also interesting to look at how much an independent set of size k can be expanded into a bigger independent set. The authors prove that with high probability, almost all independent sets of size $k = (1-\varepsilon)n\frac{\ln d}{d}$ are contained in some bigger independent set of size $(1+\varepsilon)n\frac{\ln d}{d}$, above the transition point. However, with high probability an independent set of size $k = (1+\varepsilon)n\frac{\ln d}{d}$ cannot be expanded to an independent set of size $(1+\gamma)k$, where $\gamma > \varepsilon_d$ without first reducing its size below the critical point $k^* = n\frac{\ln d}{d}$. All in all, these results demonstrate that before the critical point k^*, most independent sets can be expanded to sizes $k^* + \varepsilon$, but with high probability these new independent sets are not expandable. This means that the cost function landscape associated to MIS on random graphs is full of local minima around half the size of the MIS, with exponentially few of them being expandable to higher sizes. We illustrate this property in Figure 3.1.

Statistical physics have had a prominent role in understanding the complexity of the landscape of random **NP**-hard problems. The emergence of phase transition in classical combinatorial optimisation problems naturally connects them to transitions in mean-field spin glasses [BFK+13], as physical systems defined via random constructions like the Anderson localisation model [And58] have been studied for decades in physics. Furthermore, optimisation problems are intimately related to low-temperature statistical physics.

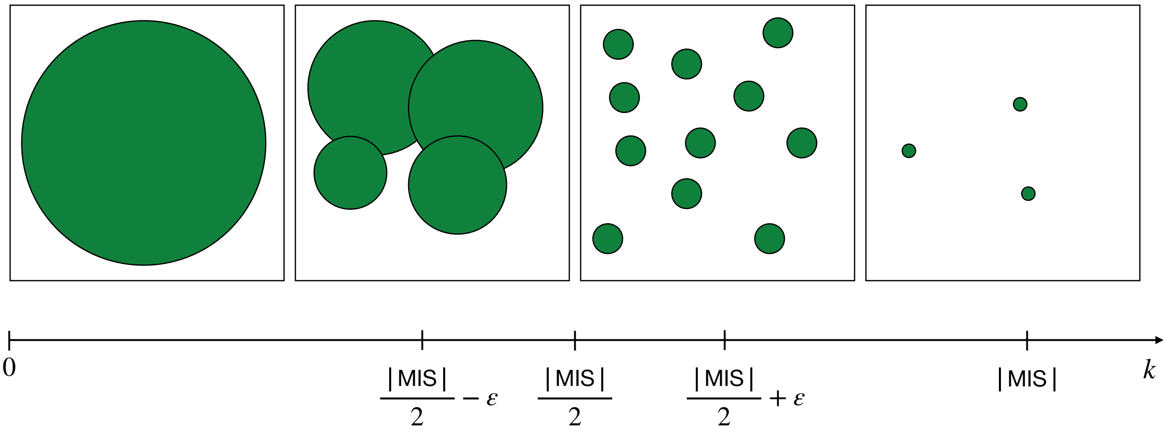


Figure 3.1: **Shattering of the independent sets in random graphs.** The x-axis corresponds to the size of the independent sets of a random graph $\mathcal{G}$. At small sizes, the independent sets are adjacent and it is quite easy to navigate from one to another. At the critical size $k^* = |\mathrm{MIS}|/2$, the independent sets shatter to an ensemble of exponentially small ensemble of independent sets. Tragically, the abundant local minima at size $k^* + \varepsilon$, where $\varepsilon > 0$ are *dead-ends* in the sense that they are not expandable to bigger independent sets (these are also called *maximal* independent sets as opposed to maximum). These minima make it very difficult to find the global MIS using local algorithms.

3.2 Simulated annealing: a statistical physics approach to combinatorial problems

Let us consider an optimisation problem defined by a cost function C over the domain $\{0,1\}^n$. While optimisation consists in finding the optimal configuration $\mathbf{z}^* \in \{0,1\}^n$, one can introduce a Boltzmann probability measure on the space of configurations where each configuration $\mathbf{z}$ is assigned a probability p_β defined as

$$p_\beta = \frac{1}{\mathcal{Z}(\beta)} e^{-\beta C(\mathbf{z})}, \quad \text{where}\, \mathcal{Z}(\beta) = \sum_{\mathbf{z}\in\{0,1\}^n} e^{-\beta C(\mathbf{z})}. \tag{3.4}$$

In this formulation the parameter β plays the role of an inverse temperature and the partition function $\mathcal{Z}(\beta)$ ensure the normalisation of the probability law. In the low temperature regime ($\beta \longrightarrow \infty$) the probability distribution concentrates on the configurations of minimum energy which is the relevant limit for optimisation problems. This approach led to the creation, among other methods, of *simulated annealing*, a statistical physics-inspired method to solve combinatorial optimisation problems. To discover the lowest values of the cost function C, we first define an annealing schedule of the form

$$\{(\beta_1, N_1), (\beta_2, N_2), \ldots, (\beta_L, N_L)\}, \tag{3.5}$$

where $\beta_i \in [0, +\infty[$ is the inverse temperatures and $N_i \in \mathbb{N}$. Starting from a random configuration $\mathbf{z_0}$, the algorithm executes N_1 a random-walk in the configuration space $\{0,1\}^n$ with transition probabilities respecting the detailed-balance equations with respect to the Boltzmann probability measure at temperature β_1. After the N_1 steps, the final state is used as a starting point for the next cycle of N_2 steps at a temperature β_2. Usually, the temperature schedule slowly decreases towards zero and if done slowly enough, ensures that the final state is the optimal solution. An intuitive way of understanding the idea of the algorithm is that at high temperatures the random-walk equilibrates across the configuration space, and as the temperature decreases it focuses on low-cost regions of the space. The finite temperature ensures that with some non-zero probability a configuration with higher cost can be accepted by the algorithm. This temperature can be related to a kinetic energy term which can help the system escape local minima in a *thermal hopping* fashion. By the end of the annealing schedule, the very low temperature forces the system to the minimum energy cost. An important remark is that this method is guaranteed to converge to the optimum if for every temperature β_i the random walk, seen as a Markov chain, converges to its stationary distribution. Unfortunately, there are instances of graphs for which the number of steps required to converge to the stationary distribution increases dramatically. For the random Erdos-Renyi graphs $\mathcal{G}(n, m)$ that we have studied earlier, it has been shown that the mixing time of the Markov chain becomes exponential with n when trying to access independent sets bigger than $(1 + \varepsilon_d)n \ln d/d$ [COE15].

In the following section, we will describe quantum annealing, which is the quantum version of the simulated annealing, where the kinetic term is replaced by a transverse field. The idea is then to confront both methods and try to understand in which cases we can expect quantum annealing to have an advantage over simulated (thermal) annealing.

3.3 Adiabatic quantum algorithms

For any given cost function C, we define the associated cost Hamiltonian H_C, diagonal in the computational basis such that

$$H_C \left|\mathbf{z}\right\rangle = C(\mathbf{z}) \left|\mathbf{z}\right\rangle, \tag{3.6}$$

In the following, we omit for clarity the bold font for $\mathbf{z}$ and simply write a computational basis state as $\left|z\right\rangle$. To perform a quantum annealing scheme, we choose another Hamiltonian H_M, called the *mixing Hamiltonian* that does not commute with H_C. A typical choice for H_M is

$$H_M = -\sum_{i=1}^{n} \hat{\sigma}_i^x, \tag{3.7}$$

that can be seen as a transverse field acting on the qubit system. A quantum annealing procedure consists in initialising the system in the ground-state of H_M, here $|\psi(0)\rangle = |+\rangle^{\otimes n}$ and applying a time-dependent Hamiltonian $H(s)$ according to:

$$\frac{i}{\mathcal{T}}\frac{d}{ds}|\psi(s)\rangle = H(s)|\psi(s)\rangle, \tag{3.8}$$

where $H(s)$ is a linear interpolation between the initial and the final Hamiltonian, $s = t/\mathcal{T}$ is the interpolation parameter where $\mathcal{T}$ is the total duration of the evolution. If the adiabatic condition holds, then for all times s the state $|\psi(s)\rangle$ is close to the instantaneous ground-state. Therefore at the end of the annealing, $|\psi(s=1)\rangle$ corresponds to the ground-state of H_C and a measurement of the qubit system returns an optimal solution to the cost function C.

The adiabatic condition

Let $|z_0(s)\rangle, |z_1(s)\rangle$ be the instantaneous ground-state and first excited state of $H(s)$, with respective energies $E_0(s)$ and $E_1(s)$. According to the adiabatic condition [Ami09], a quantum system remains in its instantaneous ground-state if the following condition on the total time $\mathcal{T}$ is satisfied:

$$\mathcal{T} \gg \frac{1}{\varepsilon} \max_{s\in[0,1]} \frac{|\langle z_0(s)|\frac{d}{ds}H(s)|z_1(s)\rangle|}{|E_0(s) - E_1(s)|^2}. \tag{3.9}$$

This condition ensures that the probability of not finding the system in the ground-state is at most ε^2. More formal versions of the adiabatic conditions exist and we refer the interested reader to the review by Albash and Lidar [AL18], but this version is sufficient to put forward the important condition for adiabaticity. Note that the the term $|\langle z_0(s)|\frac{d}{ds}H(s)|z_1(s)\rangle|$ can be seen as the difference in slopes between the ground-state and the first excited state. Because it has no singular scaling with the system size n [BFK+13], the adiabatic condition of equation (3.9) can be replaced by the simpler form

$$\mathcal{T} \gg \mathcal{O}(n g_{\min}^{-2}), \tag{3.10}$$

where $g_{\min} = \min_{s\in[0,1]} |E_0(s) - E_1(s)|$. This means that the total time of the adiabatic protocol is governed by the minimum gap $g_{\min}$ and its scaling with n. Typically, we do not expect to find $g_{\min} = 0$ for non-degenerate Hamiltonians which would correspond to a level crossing between the ground-state and the first excited state during an adiabatic evolution. The intuition behind this can be built by looking at the adiabatic process on a simple two-level generic Hamiltonian that depends on s

$$H(s) = \begin{pmatrix} a(s) & c(s) + id(s) \\ c(s) - id(s) & b(s) \end{pmatrix}, \tag{3.11}$$

where a, b, c and d are real-valued functions. In this case, the level crossing corresponds to having $a(s) = b(s) = x$ and simultaneously $c(s) = d(s) = 0$. The parameterised curve $(a(s), b(s), c(s), d(s))$ in $\mathbb{R}^4$ does not possess any specific characteristics or constraints that would cause it to cross $(x, x, 0, 0)$. On the other hand, if $H(s)$ commutes with $\hat{\sigma}_x$, then $a(s) = b(s)$ and $d(s) = 0$, and it seems more probable that for some s, $c(s) = 0$ which would correspond to level-crossing. These arguments can be generalised to Hamiltonians acting on n qubits and the authors of [FGGS00a] argue that in the absence of symmetries, levels typically do not cross during an adiabatic evolution. Although levels do not cross, the *anti-crossing* phenomenon can create exponentially small gaps that make the evolution time impractically large.

Avoided level-crossing and adiabaticity

Also known as the von Neumann-Wigner theorem, an avoided level-crossing corresponds to levels getting extremely close but not equal during an evolution under an external perturbation. The following Hamiltonian describes the phenomenon on a reduced two-level system. Let

$$H = \begin{pmatrix} b(s-s^*) & \gamma \\ \gamma & -b(s-s^*) \end{pmatrix}, \tag{3.12}$$

where γ represents an external perturbation. After diagonalising H, the spectral gap reads $g(s) = 2\sqrt{\gamma^2 + b^2(s-s^*)^2}$ which means that even when the diagonal values are equal the spectral gap is non zero as $g_{\min} = g(s^*) = 2\gamma$.

In the following section, we will explicit instances of graphs for which solving the MIS with a typical adiabatic path yields exponentially closing gaps. Before doing so, we would like to conclude by recalling the main differences between classical and quantum annealing schemes. The main difference comes from the resource that is used to escape local minima; finite temperature in the classical system enables thermal hopping *over* energy barriers while quantum tunnelling allows the system to go *through* energy barriers. Therefore, the height and width of typical barriers in the energy landscape of a combinatorial optimisation problem affect the quality of each method. We illustrate in Figure 3.2 the optimal landscape for each method.

3.4 Hard instances of MIS for quantum adiabatic algorithms

In this section, we build hard instances of the MIS problem by adapting the proof from Farhi *et al.* [FGG+09] where they build hard instances of 3SAT for which the spectral gap vanishes exponentially by the end of the adiabatic path.

We propose a family of random bipartite graphs. In this construction, we build a bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \mathcal{A} \cup \mathcal{B}$, with $|\mathcal{A}| = |\mathcal{B}| = \frac{n}{2}$, $\mathcal{A} \cap \mathcal{B} = \emptyset$ and $\mathcal{E} = \{(a, b), a \in \mathcal{A}, b \in \mathcal{B}\}$. To begin with, we initially connect nodes from $\mathcal{A}$ and $\mathcal{B}$ as an even cycle graph. This insures that the only two solution MIS are all nodes in A or all nodes in B. From the $n(n-1)$ available edges, we select each one randomly with probability $p = \log n/n$ (grey edges in Figure 3.3).

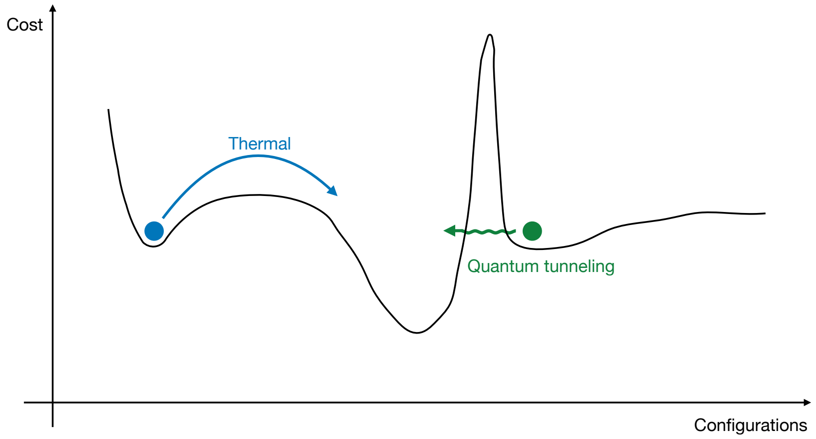


Figure 3.2: **Walls and traps for annealing schemes.** Simulated annealing and quantum annealing can escape local minima in different fashion and their efficiency depends on the cost function landscape. Precisely, the height and width of a barrier between a local minima and a global solution affect differently simulated thermal annealing or quantum annealing. Simulated annealing works fine if the cost barriers are short. If the cost barrier is too high, the probability of accepting the configuration on top of the barrier is very low or requires high temperatures and does not converge. On the other hand, quantum annealing tunnels through energy barriers. It works great if the energy barrier is thin, and can tunnel through high-valued energy peaks.

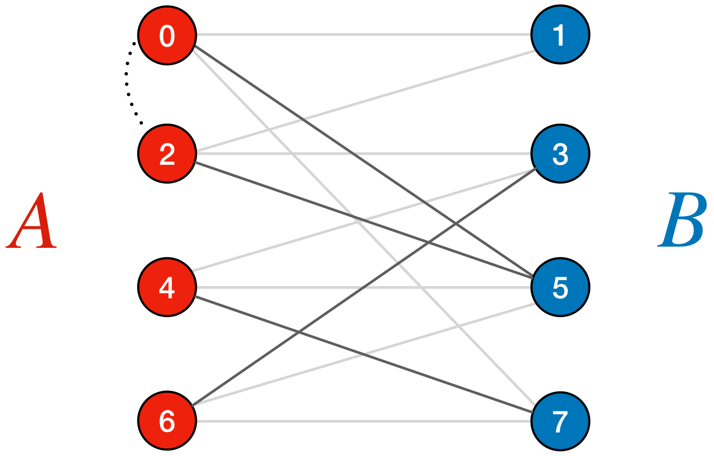


Figure 3.3: An instance of a random bipartite graph. While by nature A and B have the same total out-degree, the distribution over the nodes is different: $d_A = \{3, 3, 3, 3\}$ while $d_B = \{2, 3, 4, 3\}$. In order to have a non-degenerate ground-state for H_C, we add an artificial E_l link between nodes 0 and 2, insuring that B is the ground-state and A is right above. Taking $E_l = \frac{1}{2}\Delta$ ensures that A is the first excited state of H_C (lower than B with one less node).

We consider a cost Hamiltonian of the following form

$$\tilde{H}_c = -\Delta \sum_{i=1}^{n} \hat{\sigma}_i^z + U \sum_{i<j} A_{ij} \hat{n}_i \hat{n}_j, \tag{3.13}$$

where $\Delta > 0$ and A_{ij} are the elements of the adjacency matrix of the graph $\mathcal{G}$ and $\hat{n}_i = (\mathbb{I} + \hat{\sigma}_z)/2$. We then add an artificial E_l link between nodes 0 and 2, insuring that B is the ground-state and A is the non-degenerate first excited state. Taking $E_l = \frac{1}{2}\Delta$ ensures that A is lower in energy than B with one less node. We obtain the cost Hamiltonian

$$H_c = \tilde{H}_c + h = -\Delta \sum_{i=1}^{n} \hat{\sigma}_i^z + U \sum_{i<j} A_{ij} \hat{n}_i \hat{n}_j + E_l \hat{n}_0 \hat{n}_1. \tag{3.14}$$

Furthermore, we choose the following mixing Hamiltonian

$$H_M = -\Omega \sum_{i=1}^{n} \hat{\sigma}_i^x. \tag{3.15}$$

Note that Ω and Δ are positive constants which physically correspond respectively to the Rabi frequency and the detuning in a quantum Ising Hamiltonian. The ground-state of H_M is the equal superposition of all vectors of the computational basis representing independent sets in the graph. Then, the idea is to slowly vary from H_M to H_C using the adiabatic evolution:

$$H(s) = (1-s)H_M + sH_c, \quad s \in [0,1], \tag{3.16}$$

where s controls the rate at which $H(s)$ varies.

Location of the minimum spectral gap

We show in this subsection the existence of an exponentially closing gap at the end of the adiabatic path. Recall that

$$H(s) = (1-s)H_M + sH_c = s\left[\frac{1-s}{s}H_M + H_c\right]. \tag{3.17}$$

When $s \to 1$, we can view H_M as a perturbation term of $H(s)$. We will use perturbation theory to estimate the location of the minimum spectral gap.

Let z be a bit-string of size n. In the perturbation approximation, the energy associated to z can be re-written as:

$$E_z(s) = s\left[E_z^{(0)} + \frac{1-s}{s}E_z^{(1)} + \frac{(1-s)^2}{s^2}E_z^{(2)} + \dots\right], \tag{3.18}$$

where

$$E_z^{(0)} = \langle z|H_c|z\rangle = E_z, \tag{3.19}$$

$$E_z^{(1)} = \langle z|H_M|z\rangle = 0, \tag{3.20}$$

$$E_z^{(2)} = \sum_{y \neq z} \frac{|\langle y|H_M|z\rangle|^2}{E_z^{(0)} - E_y^{(0)}}, \tag{3.21}$$

and $|y\rangle$ is a computational basis state.

Note that all odd powers of the expansion are null: this is because $\langle y|H_M|z\rangle \neq 0$ if and only if y and z are at a Hamming distance of 1. For an odd power expansion term, there is therefore always a term in the product that is equal to zero. We can therefore re-write the perturbation approximation of the energy associated to z, defining $\lambda := \frac{1-s}{s}$:

$$E_z(s) = s \sum_{m=0}^{\infty} \lambda^{2m} E_z^{(2m)}, \tag{3.22}$$

Furthermore, we only need to explicit the second order to find the location of the minimum spectral gap. By re-writing $\hat{\sigma}_x^{(i)}|z\rangle = |z \oplus e_i\rangle$ where $|e_i\rangle$ is the $i-$th computational basis state, we can write the second order correction as:

$$\begin{aligned} E_z^{(2)} &= \sum_{y\neq z} \frac{|\langle y|H_M|z\rangle|^2}{E_z^{(0)} - E_y^{(0)}}, \\ &= |\Omega|^2 \sum_{i=1}^{n} \frac{|\langle z\oplus e_i|\sigma_x^{(i)}|z\rangle|^2}{E_z^{(0)} - \langle z\oplus e_i|H_c|z\oplus e_i\rangle}, \\ &= |\Omega|^2 \sum_{i=1}^{n} \frac{1}{E_z^{(0)} - \langle z\oplus e_i|H_c|z\oplus e_i\rangle}. \end{aligned} \tag{3.23}$$

The two MIS solutions on either even vertices or the odd vertices will respectively be called the upper state and the lower state according to their place in the energy spectrum of H_C. We will denote by z_U and z_L the corresponding bitstrings ($z_U = 1010...0$ and $z_L = 0101.....1$). We can write the energy difference between these two states as a function of s:

$$E_U(s) - E_L(s) = s\left[E_l + \frac{(1-s)^2}{s^2}(E_U^{(2)} - E_L^{(2)})\right]. \tag{3.24}$$

We can then estimate the second order difference

$$E_U^{(2)} - E_L^{(2)} = |\Omega|^2 \sum_{i=1}^{n} \frac{1}{E_U^{(0)} - \langle z_U\oplus e_i|H_c|z_U\oplus e_i\rangle} + |\Omega|^2 \sum_{i=1}^{n} \frac{1}{E_L^{(0)} - \langle z_L\oplus e_i|H_c|z_L\oplus e_i\rangle}. \tag{3.25}$$

We can now rewrite Equation (3.25) as

$$E_U^{(2)} - E_L^{(2)} = |\Omega|^2 \sum_{i=1}^{n} \frac{1}{E_U^{(0)} - \langle z_U\oplus e_i|H_c + E_l n_0 n_2|z_U\oplus e_i\rangle} + \frac{1}{E_U^{(0)} - \langle z_L\oplus e_i|H_c + E_l n_0 n_2|z_L\oplus e_i\rangle}. \tag{3.26}$$

For $i > 2$, the link E_l is conserved and the energy difference can be written, for $i \in [\![3, n]\!]$:

$$E_U^{(2)} - E_L^{(2)} = |\Omega|^2 \sum_{i=3}^{n} \frac{1}{E_U^{(0)} - \langle z_U \oplus e_i | H_c | z_U \oplus e_i \rangle} - \frac{1}{E_L^{(0)} - \langle z_L \oplus e_i | H_c | z_L \oplus e_i \rangle}, \tag{3.27}$$

$$= |\Omega|^2 \sum_{i=3}^{n} \frac{1}{E_U^{(0)} - \langle z_U \oplus e_i | H_c | z_U \oplus e_i \rangle} - \frac{1}{E_L^{(0)} - \langle z_L \oplus e_{i+1 \pmod n} | H_c | z_L \oplus e_{i+1 \pmod n} \rangle}, \tag{3.28}$$

$$= |\Omega|^2 \sum_{i \text{ odd}}^{n} \frac{1}{E_U^{(0)} - \langle z_U \oplus e_i | H_c | z_U \oplus e_i \rangle} - \frac{1}{E_L^{(0)} - \langle z_L \oplus e_{i+1 \pmod n} | H_c | z_L \oplus e_{i+1 \pmod n} \rangle}. \tag{3.29}$$

We only keep the terms in the sum that increase the Hamming weight of the bitstring, which corresponds to adding a node from the opposite class. In that case,

$$\langle z_U \oplus e_i | H_c | z_U \oplus e_i \rangle = (n+1)\Delta + U \times d_i + E_l, \tag{3.30}$$

and

$$\langle z_L \oplus e_i | H_c | z_L \oplus e_i \rangle = (n+1)\Delta + U \times d_i, \tag{3.31}$$

where d_i is the degree of node i. Recall that:

$$E_L^{(0)} = n\Delta \quad \text{and} \quad E_U^{(0)} = n\Delta + E_l. \tag{3.32}$$

and we can simplify the equation to

$$E_U^{(2)} - E_L^{(2)} = |\Omega|^2 \sum_{i \text{ odd}}^{n} \frac{1}{-\Delta - U \times d_i} - \frac{1}{\Delta - U \times d_{i+1}}. \tag{3.33}$$

By construction, the degree of each node is a random variable that follows a binomial distribution $\mathcal{B}(n, \log n/n)$ with $\mathbb{E}(d_i) = O(\log n)$ and variance $\mathbb{V}(d_i) = \log n(1 - \frac{\log n}{n}) = O(\log n)$.

Useful results

We recall a few useful results that will be used in the estimation of the energy difference.

Let X be a random variable following a binomial distribution $X = \mathcal{B}(n, p)$.

$$\mathbb{E}\left(\frac{1}{X}\right) \simeq \frac{1}{\mathbb{E}(X)} + \frac{1}{\mathbb{E}(X)^3}\mathbb{V}(x), \quad \text{and} \tag{3.34}$$

$$\mathbb{E}(X^c) = \sum_{k=0}^{c} S(c,k) n^{\underline{k}} p^k, \tag{3.35}$$

where $S(c,k)$ is the Stirling numbers of the second kind and $n^{\underline{k}}$ is the k-th falling power of n defined as

$$n^{\underline{k}} = n(n-1)...(n-k+1). \tag{3.36}$$

Back to estimating the second order difference

We recall that the second order difference is given by eq.(3.33):

$$E_U^{(2)} - E_L^{(2)} = |\Omega|^2 \sum_{i \text{ odd}}^{n} \frac{1}{-\Delta + U \times d_{i+1}} - \frac{1}{-\Delta + U \times d_i}. \tag{3.37}$$

By symmetry of the degree distribution, the expectation value of the difference is zero. We now estimate the variance of the difference. In order to do this, we are interested in the quantity

$$\mathbb{V}\left(\frac{1}{d_i}\right) = \mathbb{E}\left(\frac{1}{d_i^2}\right) - \mathbb{E}\left(\frac{1}{d_i}\right)^2. \tag{3.38}$$

On the one hand, we have

$$\mathbb{E}\left(\frac{1}{Ud_i - \Delta}\right) = \frac{1}{U\mathbb{E}(d_i) - \Delta} + U^2 \frac{1}{(U\mathbb{E}(d_i) - \Delta)^3} \mathbb{V}(d_i), \tag{3.39}$$

$$\simeq \frac{1}{\log n} + \frac{1}{\log^2 n}, \tag{3.40}$$

using equation 3.34, and on the other hand

$$\mathbb{E}\left(\frac{1}{d_i^2}\right) \simeq \frac{1}{\mathbb{E}(d_i^2)} + \frac{1}{\mathbb{E}(d_i^2)^3} \mathbb{V}(d_i^2), \tag{3.41}$$

and

$$\mathbb{E}(d_i^2) = n(n-1)p^2 + np = (\log n)^2 + \log n, \tag{3.42}$$

and

$$\mathbb{V}(d_i^2) = -2(\log n)^3, \tag{3.43}$$

$$= O((\log n)^3), \tag{3.44}$$

and therefore

$$\mathbb{E}\left(\frac{1}{d_i^2}\right) = \frac{1}{(\log n)^2} + \frac{1}{(\log n)^2}. \tag{3.45}$$

Finally, we find that:

$$\mathbb{V}\left(\frac{1}{d_i}\right) = \frac{2}{(\log n)^2} - \frac{1}{(\log n)^2} - \frac{2}{(\log n)^3} - \frac{1}{(\log n)^4}, \tag{3.46}$$

$$= O\left(\frac{1}{\log^2 n}\right). \tag{3.47}$$

Now, the variance of the sum is of order $\sqrt{n} \times \mathbb{V}(\frac{1}{d_i}) = O(\frac{\sqrt{n}}{\log^2 n})$. We can know find s^* such that:

$$\begin{aligned} & E_U(s^*) - E_L(s^*) = 0, && (3.48) \\ \iff & s^* \left[(E_U^{(0)} - E_L^{(0)}) + \frac{(1-s^*)^2}{s^{*2}} (E_U^{(2)} - E_L^{(2)}) \right] = 0, && (3.49) \\ \iff & \frac{1}{s^*} = 1 \pm \sqrt{\Theta(\frac{\Delta^2}{|\Omega|^2} \frac{\log^2 n}{\sqrt{n}})}, && (3.50) \\ \iff & s^* = \frac{1}{1 + \Theta(\frac{\Delta \log^2 n}{|\Omega| n})}, && (3.51) \end{aligned}$$

where we keep the solution such that $s^* < 1$.

We see that for theses instances, in the same fashion as in Ref. [FGGS00b], the position of the spectral gap goes to 1 as $n \to \infty$. Let us remind that the energy difference expression can be written as

$$E_{UL}(s) = s \sum_{m=0}^{\infty} \lambda^{2m} E_{UL}^{(2m)}. \tag{3.52}$$

where $E_{UL}^{(2m)} := E_U^{(2m)} - E_L^{(2m)}$. Because the two solutions U and L are at a Hamming distance of order n, the non-zero term appears at the n^{th} order correction as $\langle U | H_M^n | L \rangle \neq 0$.

We can write the Hamiltonian at the $n-$th order correction, reduced to the two levels $|U\rangle$ and $|L\rangle$ as

$$H(s) = \begin{pmatrix} \frac{E}{2}(s - s*) & \lambda^n \\ \lambda^n & \frac{E}{2}(s - s*) \end{pmatrix}, \tag{3.53}$$

where $E = E_U - E_L$. By diagonalisation of the matrix, we see that the instantaneous gap is

$$E_{UL}(s) = 2\sqrt{\lambda^{2n} + \frac{E^2}{4}(s - s^*)}, \tag{3.54}$$

and when $s = s^*$, the energy anti-crossing correction is

$$E_{12}(s^*) = 2\lambda^n, \tag{3.55}$$

which is exponentially small in n.

To summarise, we built random instances of MIS that reveal an interesting mechanism that can happen in quantum annealing, resembling Anderson localisation on the hypercube [AKR10]. Indeed, the avoided level crossing becomes exponentially small mainly because the two solutions have been artificially planted at the antipodes of the hypercube at a Hamming distance of order n. Therefore, they can only be connected by a transverse field matrix element of order $|\Omega|^n$, which becomes exponentially small in the thermodynamic limit $n \to \infty$.

Another structure of the cost function landscape that can give rise to an exponentially closing gap has been explored in Ref [AC09]. In their work, the authors build a graph instance where there is a single global narrow minimum, and a secondary local minimum that has higher

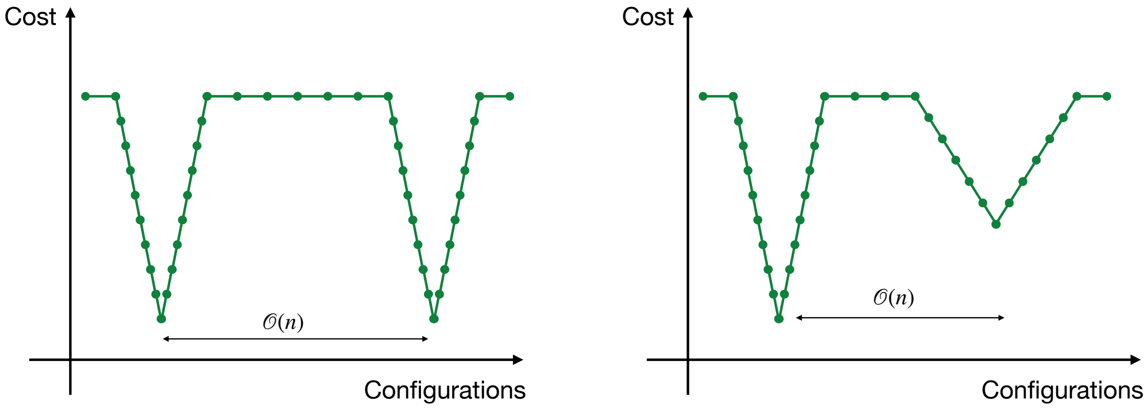


Figure 3.4: **Anderson localisation makes quantum annealing fail.** This visualisation is inspired from [BFK+13]. We represent the cost function of two problems that put forward Anderson localisation using two different mechanisms that can create anti-crossing. The configurations are represented as point linked to their neighbours in Hamming distance. On the left, we depict our built instances for which two global solutions are far apart in Hamming distance. On the right, the authors of Ref [AC09] build an instance with a local minimum which is wider than the global minimum. Both these cost functions explicit an exponential closing of the spectral gap close to $s = 1$ between localised states.

energy but is much wider (see Figure 3.4). In this case, the denominator in the perturbation equation 3.21 is much smaller for the secondary minimum and therefore under the effect of the transverse field the energy level of the secondary minimum is lowered faster than the global one and a crossing eventually appears. For both types of instances, the position of the minimal spectral gap goes to $s = 1$ asymptotically and has therefore been called a perturbative anti-crossing in the literature [AKR10, Dic11] between two states that are almost classical. While it exhibits an exponentially closing gap, these instances are toy-models which were carefully crafted to showcase these possible anti-crossings. Our findings are in accordance with a very interesting recent result [CCL+23] that also proves exponential closing gaps close the end of the adiabatic path. In this paper, the authors put forward a mechanism of (de)localisation in the configuration space. They show that the QAA performance is determined by (de)localisation of the low-energy eigenstates right above the MIS in configuration space. They show that when low-energy eigenstates are delocalised over the configuration space, QAA can achieve a quadratic speedup over a wide class of SA and parallel tempering algorithms. While these results are promising, it is still an open-question to know if these phenomenons appear in real data. Indeed, the problems we built are not representative of general combinatorial optimisation problems: typical hard instances of exact cover for example (the problem depicted in [AKR10]) have been shown to have exponentially-many solutions [KS10] for which the non-degenerate perturbation theory is not valid. Some quantum Monte-Carlo numerical results have rather shown the appearance of exponentially closing gaps at the middle of the adiabatic path [YKS10]. All in all, it is interesting to see that the structure of the (classical) energy landscape associated to a combinatorial optimisation problem affects the quality of a classical or quantum algorithm. All NP-complete problems seem to share the characteristic of an energy landscape with many local minima, which explains the difficulty to solve them exactly. However, the number of configurations around a local minima can change from one NP-complete problem to another, as well as the steepness of the energy around a local minimum can depend on the minimum itself [BFK$^+$13]. Different NP-Complete problems can have very different energy landscapes that have been studied in detail thanks to developments in the analysis of classical spin glasses [MPV87, BMW00, MZ02]. The structure of the cost landscape is also a good indicator of the hardness in finding approximate solutions. In the random Erdos-Renyi graphs that we studied in Section 3.1, the shattering of the independent sets at half the size of the MIS explained why it is so hard to find even an approximate MIS of size $\frac{|\mathrm{MIS}|}{2} + \varepsilon$, $\varepsilon > 0$. In the following section, we will leave the study of the energy landscape and dive into the theory of approximate solutions with *a priori* guarantees. In this fascinating topic, researchers have discovered that although finding an exact solution is NP-hard, some problems can be approximated much more easily than other ones. This includes approximate solutions to the MIS on Unit-disk graphs.

3.5 Approximation schemes for MIS

Unit-disk graphs are a class of graphs which are inherently local in the sense that two nodes v and w are connected by an edge if and only if the distance between the two nodes is inferior to a given threshold. These graphs are part of the larger family of intersection graphs, that appear in concrete problems such as wireless networking [Gil61], computational biology [XB06], map labelling [EHJ$^+$10] or to maximise yield when cutting chips from a large chip wafer [HM85]. An example of a broadcasting instance is depicted in Figure 3.5. Although these graphs seem to present a little more structure than the random graphs we looked at previously, solving MIS on unit-disk graphs is also **NP**−hard [CCJ90]. Because it is strongly believed that it is not possible to find a polynomial time algorithm to solve optimally NP-hard problems, operational researchers have been strongly motivated in finding approximate solutions. The idea is to find a good trade-off between time complexity and solution quality. The simulated annealing or quantum annealing that we saw previously are part of this family as well as other well-known schemes such as polyhedral methods [CCZ09]. The approximation algorithms of interest in this

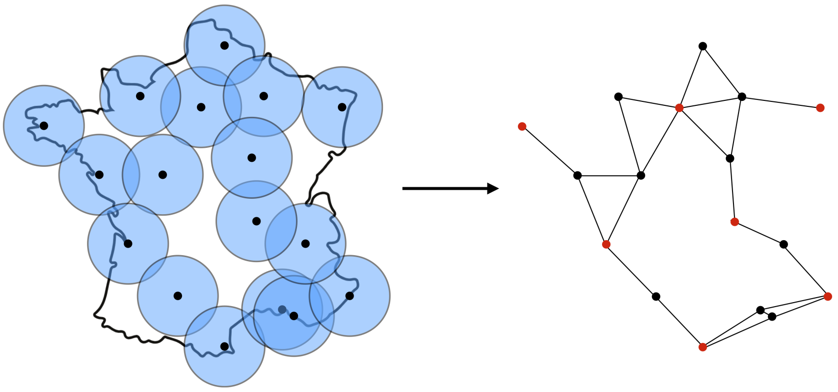


Figure 3.5: **Unit-disk graphs represent well broadcasting problems.** Suppose that an ensemble of identical radio transmitters over French cities have the same radius of transmission. They can be naturally modelled by a unit-disk graph, where each tower is represented by a vertex and two vertices are connected if their respective towers transmit on an overlapping territory. Then, a typical reason one would want to solve the MIS problem on this graph is that two transmitters with close or equal frequencies can interfere with one another, hence the necessity to assign non-interfering frequencies to overlapping transmitting towers. Because of the limited amount of bandwidth space, some towers have to be assigned the same or close frequencies. The MIS of a graph of towers indicate the maximum number of towers that can have close or equal given frequency (red points).

section are called *polynomial approximation schemes with a priori performance guarantees.*

Polynomial-time approximation schemes

One example of such an approximation algorithm is called the Polynomial Time Approximation Scheme (PTAS) that provides near-optimal solutions to complex optimisation problems in polynomial time. A more formal definition [Vaz10] is:

Definition 3.2 (PTAS)**.** *Let* Π *be an NP-hard optimisation problem with objective function* f_Π*.* OPT *will denote the objective function value of an optimal solution to instance* I*. Algorithm* $\mathcal{A}$ *is an approximation scheme for* Π *if on input* (I, c)*, where* I *is an instance of* Π *and* $c > 0$ *is an error parameter, it outputs a solution* s *such that:*

1. $f_\Pi(I, s) \leq (1 + c) \cdot OPT$ *if* Π *is a minimisation problem.*

2. $f_\Pi(I, s) \geq (1 - c) \cdot OPT$ *if* Π *is a maximisation problem.*

$\mathcal{A}$ will be said to be a PTAS if for each fixed $c > 0$, its running time is bounded by a polynomial in the size of instance I.

The great advantage of this method is that it returns feasible solutions with an *a priori* guarantee on the quality of the solution, in polynomial time. Most examples of PTAS appear for problems that have an underlying structure over which a divide-and-conquer approach is applied. This approach consists in cutting the problem in smaller pieces, solving locally the independent parts and then concatenating the results to find a global feasible solution. The local solving is done exactly, which can take exponential time but on a sub-instance of size $m \ll n$. A significant characteristic of a PTAS is that its running time $\mathcal{A}$ can depend arbitrarily on $c > 0$. In some cases, this dependence is exponential. In Appendix A.4, we describe in detail how a PTAS can be used to tackle the MIS problem on Unit-disk graphs, using the *shifting strategy*. A shifting lemma bounds the error of the shifting strategy and therefore offers a guarantee for the approximation ratio that depends on the required precision.

Another type of PTAS exists for solving approximately (weighted) UD-MIS called the breadth-first search sphere based scheme [NHK05]. The authors of Ref [SMA20] draw inspiration from this method to build a classical heuristic of local nature. Precisely, a vertex $u \in \mathcal{V}$ of the graph is selected at random, and given a parameter d the subgraph $\mathcal{S}_d(u)$ of radius d is built. The MIS of $\mathcal{S}_d(u)$ is calculated and the subgraph and its neighbours are deleted. A random vertex is selected in the new graph and the steps are repeated until the graph is empty. This method, although it does not guarantee an approximation ratio and therefore cannot be considered as a PTAS, has nonetheless the advantage of processing each node exactly once and more importantly resembles finite quantum correlation distances that appear during a quantum annealing scheme [LdLB+18]. The authors of [SMA20] estimated the spin-spin correlation length for various sizes and noise levels. In a noiseless setting, the longest the annealing time, the longer the correlation length gets through the system and hence a better solution can be found. In the case of noise however, the decoherence introduces defaults in the correlation between distant atoms, thereby limiting the median size of the correlation. Therefore for each noise level there exists an optimal annealing time, independent of the size of the graph under study. This allows to extrapolate without simulating the expected approximation ratio of quantum annealing on UD-MIS for large sized graphs. Compared to their heuristic, they find a break-even point for quantum advantage estimated to 8,000 atoms for a time budget of 2 seconds, or 1,000 - 1,200 atoms for a time budge of 0.2 seconds, given the actual repetition rate of a neutral atom quantum processor and that the noise level can be improved by a factor 10.

More general graphs are harder to approximate

Interestingly, the quality of the approximation depends on the type of graph under study: for Unit-Disk graphs, we have seen previously that classical algorithms can leverage the locality of the edges to efficiently estimate an approximation of the MIS. In the case of planar graphs, another PTAS exists that also guarantees high approximation ratios. In the same flavour as for the UD case, it relies on dividing the graph into subgraphs with $k-$outerplanar forms [Bak94].

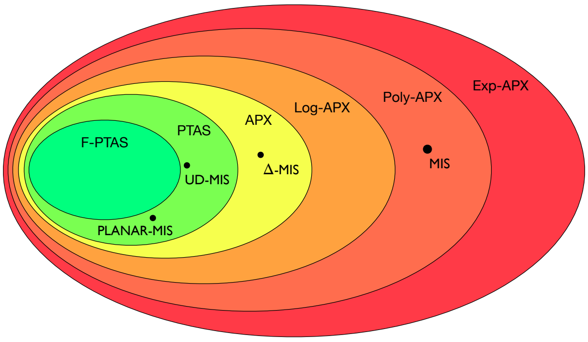


Figure 3.6: **Standard approximability classes of NP-hard problems** (under the assumption $\mathbf{P} \neq \mathbf{NP}$). While solving MIS exactly is **NP**-Hard, finding an approximate solution of useful quality can be easy, depending on the nature of the graph at hand. The complexity class **NPO** corresponds to the class of optimisation problems whose underlying decision problem is in **NP**. A polynomial-time approximation scheme (PTAS) is a family of $\varepsilon-$parameterised algorithms that output approximate optimal solutions that are $\varepsilon > 0$ away from the exact solution in polynomial time. When the family of algorithms is also polynomial in the parameter ϵ, it is a Full-PTAS (F-PTAS) and is very efficient. The **APX** class corresponds to problems for which a polynomial-time algorithm can only achieve a constant approximation ratio. For MIS, if the graph is Unit-Disk or planar, there exists a PTAS. In our work, we propose a polynomial embedding of general bounded-MIS (Δ-MIS) problems which are known to be in **APX**. Δ-MIS is even **APX**-Complete, meaning that no PTAS exists for it unless $\mathbf{P} = \mathbf{NP}$. In the case of a general graph with no underlying structure, it was proven that for any $\varepsilon > 0$, MIS is inapproximable within approximation ratio $n^{\varepsilon-1}$, corresponding to the complexity class poly-**APX**.

However, more general graphs such as bounded-degree graphs do not present enough structure for classical algorithms to $\varepsilon-$approximate maximum independent sets in polynomial time. In the case of a graph with bounded-degree Δ, finding an approximation solution to the MIS problem is known to be APX-complete [PY91]. In other words, this means that the best approximation ratio that can be achieved by a classical algorithm is constant; to the best of our knowledge this approximation ratio is $r = \frac{5}{\Delta+3}$ [BF94]. We expect that this ratio cannot be improved without adding an exponential time overhead for a classical algorithm. The approximation guaranteed by a polynomial-time approximation scheme worsens in the case of a general graph. It was proven that for any $\varepsilon > 0$, MIS is inapproximable within approximation ratio $n^{\varepsilon-1}$, unless $\mathbf{P} = \mathbf{NP}$ [Zuc06]. A summary of the approximability classes of MIS on these specific classes of graphs is shown in Figure 3.6. The key take-away is that the presence of structure (locality, planarity, bounded-degree) in a graph can provide enough information for a classical algorithm to find good approximations in polynomial time.

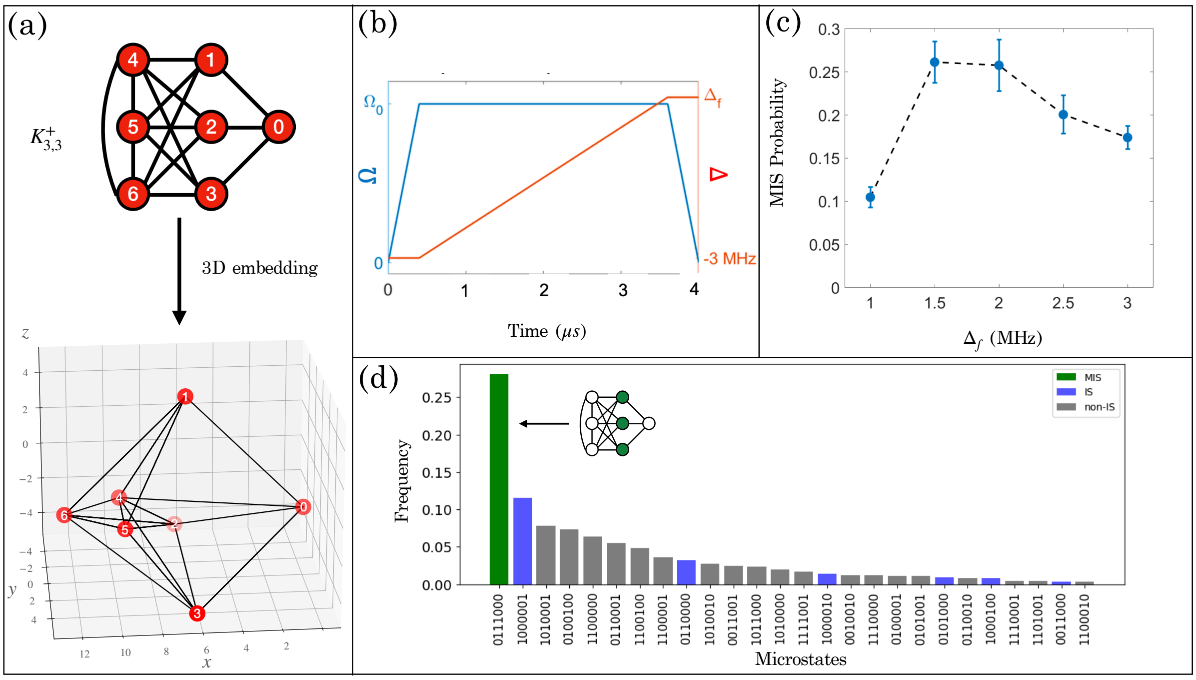


Figure 3.7: **Solving MIS on unit-ball graphs with Rydberg atoms.** The graph $K_{3,3}^{+}$ represented in **(a)** is the smallest example of a unit-ball graph for which the Greedy algorithm fails. This graph is embedded with a 3D array of atoms. **(b.)** The adiabatic evolution path corresponds to a linear detuning Δ from $\Delta_i = 2\pi \times 0.7\text{MHz}$ to Δ_f, while the Rabi frequency Ω is ramps up to $\Omega_0 = 2\pi \times 0.7\text{MHz}$. **(c.)** The experiment is tested by changing the final detuning Δ_f. **(d.)**The microstates probability histogram is presented for $\Delta_f = 2\pi \times 1.5\text{MHz}$. The SPAM error $P(g|r) = 0.15$ and $P(r|g) = 0.05$ is corrected. The MIS solution probability is improved up to $P_{\text{MIS}} = 0.25$, in stark contrast to the other microstates.

The difference in the approximability of these problems motivates the need for efficient hardware embedding of graphs which present less geometrical structure than UD or planar graphs. Unit-disk graphs can be extended to higher dimension where they are referred to as unit-ball graphs of dimension d. It can be shown that any $n-$vertex graph can be embedded as a unit-ball graph of dimension $d = n - 1$[Mae84]. The dimension of the embedding is therefore a parameter of hardness for MIS approximation, since for $d = 2$ there exists a PTAS but for $d = n - 1$ it is inapproximable within approximation ratio $n^{\varepsilon - 1}$ for any $\varepsilon > 0$. A natural step in finding a hard approximation zone is therefore to increase the dimension d, which we do in the following section by embedding unit-ball graphs where $d = 3$.

3.6 Breaking the approximation barrier: mapping harder-to-approximate graphs on a neutral atom device

3.6.1 Unit-ball graphs

A common strategy employed to find an approximate solution to the MIS problem is to use a greedy algorithm. This iterative method involves local optimal choices at each step and yields a valid solution in polynomial time. For example, a greedy algorithm on MIS selects a random vertex in G at each step, deletes its neighbours and repeats this step until the graph is empty. By construction this method guarantees that the selected vertices form an independent set. The greedy algorithm therefore provides an $1/n-$approximation to MIS as in the worst-case it returns a single vertex. Surprisingly, the inapproximability result for general MIS below $n^{\varepsilon-1}$ implies that the greedy algorithm is an optimal approximation algorithm [Hås99]. It is also proven to be optimal for bounded-degree graphs [HR97]. An improvement to the greedy algorithm is to select at each step the vertex with the lowest degree as it deletes less neighbouring vertices. As the greedy algorithm runs in polynomial time and MIS is an NP-complete problem, there exists a class of graphs for which the greedy algorithm fails completely. The smallest graph of this class [Mar17] is $K_{3,3}^{+}$ and is represented in Figure 3.7. It corresponds to the bipartite graph with 3 vertices per class, called $K_{3,3}$, augmented with edges such that one class forms a clique (in our figure it corresponds to vertices $(4, 5, 6)$) and a single vertex (0) is connected to the other class $(1, 2, 3)$. In this case, the degree-informed greedy algorithms fails. Interestingly, $K_{3,3}^{+}$ is a unit-ball graph, for which we give an embedding on the right of Figure 3.7. We therefore embed $K_{3,3}^{+}$ experimentally in 3D using an array of atoms and follow a quantum annealing scheme to prepare its MIS.

We remind that the reader that for Rydberg atoms, the Ising Hamiltonian reads

$$\hat{H}_{\theta}(\mathbf{r};t) = \hat{H}_c(t;\theta) + \hat{H}_{vdW}(\mathbf{r}) = \hbar\sum_{i=1}^{N}\left(\frac{\Omega_{\theta}(t)}{2}\hat{\sigma}_x^i - \delta_{\theta}(t)\hat{n}_i\right) + \sum_{j<i} U(||r_i - r_j||)\hat{n}_i\hat{n}_j \tag{3.56}$$

The control part is composed of two parameterized control fields, Ω_θ and δ_θ and the atoms positions $\mathbf{r}$ set the distance-dependent interaction part $U(d) \propto d^{-6}$. The trial adiabatic evolution path is tuning the detuning $\delta_\theta(t)$ linearly from δ_i to δ_f, while the Rabi frequency Ω is fixed to Ω_0. The experiment is tested by changing the final detuning Δ_f. The full 27 microstates probability histogram is presented for $\Delta_f = 2\pi \times 1.5MHz$. The SPAM error $P(g|r) = 0.15$ and $P(r|g) = 0.05$ is corrected. The MIS solution probability is improved up to $P_{\text{MIS}} = 0.25$, in stark contrast with the other microstates. These very encouraging results show that it is possible to prepare the MIS of K_{33}^{+} using a quantum annealing scheme on a 3D array of neutral atoms. On the classical side, K_{33}^{+} is the smallest example for which a greedy algorithm, optimal in approximation, fails. Other classical methods however known as *slicing* [vL06] can return the optimal solution in polynomial time on such instances. We therefore need a way to embed

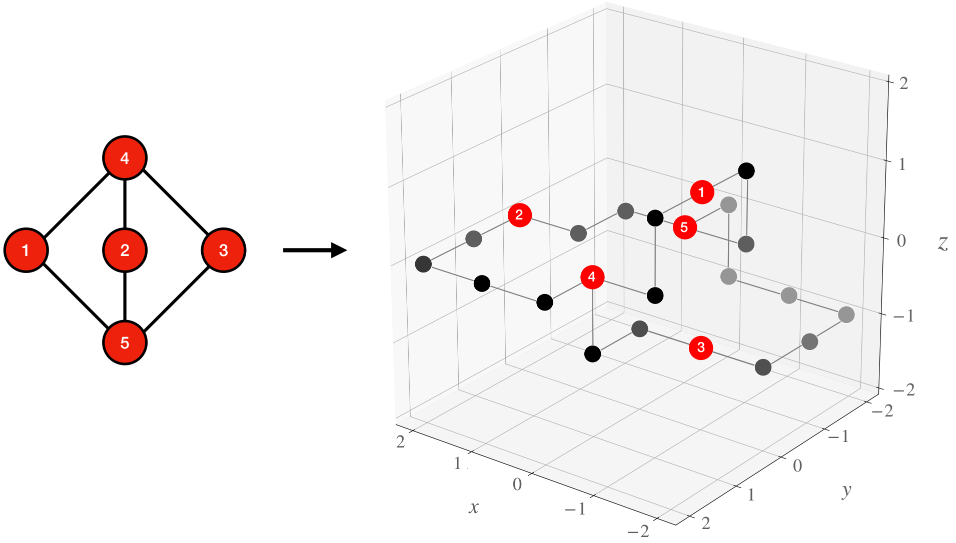


Figure 3.8: **Mapping a non-UD graph with Rydberg atoms using 3D quantum wires.** The graph on the left is the smallest example of a non-Unit-Disk graph (otherwise the vertex 2 would be connected to 1 and 3). This graph is embedded in a 3D array of atoms (right). The red atoms correspond to the original graph vertices, while the chains of black atoms encode the edges of the original graph. The procedure enables to map any general graph, including non-UD and non-planar graphs with a bounded-degree up to $\Delta = 6$.

classes of graphs for which the inapproximability is stronger. This is the case for bounded-degree graphs and we will demonstrate how such graphs can be embedded on a neutral atom device in the following section.

3.6.2 Bounded-degree graphs

Recent and previous works propose to represent non-local edges of graphs with chains of ancillary atoms [PWZ+18, KKH+21], in 2D and 3D respectively. Building upon this idea, we present an efficient and automatic method to represent any graph of degree at most 6 with a 3D array of atoms. An illustration of our method is given in Figure 3.8. It runs in polynomial time and numerical simulations suggest a low overhead in the number of added ancillary atoms (linear).

We define the *drawing* of a graph as the realisation or layout of a graph in a 3D space, where no two vertices overlap and no vertex-edge intersection occurs unless its incidence exists in the original graph. We say that a drawing is *crossing-free* if no two edges cross. A growing interest emerged in 3D drawings of graphs for circuit designs [LR86] or for information visualisation [WF94, WM08]. In our case, we focus on 3D orthogonal grid drawings (OGD) of a graph $G = (V, E)$ for which the vertices of G are represented as distinct points of the grid $\mathbb{Z}^3$, while all edges $E = (u, v) \in V^2$ are restricted to being drawn on lines parallel to one of the three axes. This restriction implies that only graphs with maximum degree six can have such a drawing; given a vertex at coordinates $(x, y, z) \in \mathbb{Z}^3$, the only authorised directions for an edge are $(x \pm 1, y \pm 1, z \pm 1)$. It is proven that every graph of bounded-degree Δ admits a crossing-free OGD if $\Delta \leq 6$:

Theorem 3.2 ([ESW96])**.** *Let G be a graph with n vertices and maximum degree $\Delta \leq 6$. Then G has a three-dimensional orthogonal grid drawing such that no pair of edges cross.*

The general idea behind our method is to construct an OGD for the original graph and replace the edges by chains of ancillary atoms. The advantage of an OGD is that two distinct edges only intersect at common endpoints, thereby preventing ancillary atoms from interacting if they are not part of the same chain. Ideally, we would like to find an OGD that minimises the edge lengths in order to have as little ancillary atoms as possible. Unsurprisingly, it is **NP**-hard to find an OGD that minimises the total length of the edges [ESW96]. While many different algorithms have been proposed to optimise the total volume [BJSW02] or the average number of bends per edge [PT97] of an OGD, we present a simple heuristic to construct an OGD with a small total length of edges.

Given a general Δ-bounded graph $G = (V, E)$ with $\Delta \leq 6$, we place the vertices of V in $\mathbb{R}^3$ using the Fruchterman-Reingold algorithm (FR) [FR91b] that runs efficiently in $\mathcal{O}(|V|^3)$ steps. Note that other algorithms could be used at this step [STT81, HGK10, Shn92] but FR yielded the best results in our simulations. The vertices are then moved to the closest grid point in $\mathbb{Z}^3$, insuring that no two vertices get the same coordinates.

We then use optimal path-finding algorithms [D$^+$59, HNR68] to find the shortest route between two vertices, restricted to the underlying grid. The resulting path is transformed into a chain of ancillary vertices. We previously ensure that two distinct edges are separated by at least a 2-grid-point distance, in order to avoid any unwanted interaction between ancillary atoms of two distinct chains. Finally, if the path length is odd, we add an ancillary vertex at each $\mathbb{Z}^3$ coordinate of the path. If the path is of even length p, we add $p + 1$ evenly spaced ancillary vertices along the path. After this procedure, one obtains an augmented graph $G_+ = (V_+, E_+)$ of size $|V_+| = N_+$.

By representing all vertices with atoms, one can encode a Maximum Independent Set of G in the ground state of an Ising Hamiltonian on Rydberg atoms [BBL16b] over the augmented graph G_+ :

$$H_+ = \sum_{j=1}^{N_+} \frac{\hbar\Omega}{2} \sigma_j^x - \sum_{j=1}^{N_+} \frac{\hbar}{2} (\delta + \delta_j) \sigma_j^z + \sum_{j<i} \frac{C_6}{|\mathbf{r}_i - \mathbf{r}_j|^6} n_i n_j. \tag{3.57}$$

where δ_j represents the local detuning applied to each atom and $n_i = (\sigma_i^z + \mathbb{I})/2$.

In Figure 3.9, we show explicitly on a single augmented edge how one can choose the values of local detuning on ancillary atoms to ensure that the MIS of the edge corresponds to the groundstate of the Hamiltonian H_+.

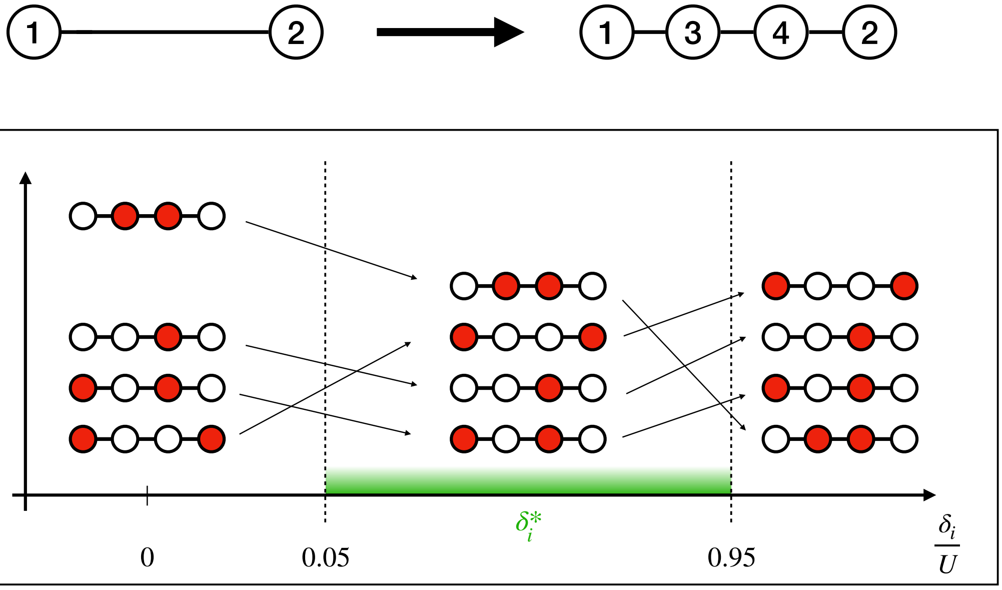


Figure 3.9: **Local detuning on ancillary atoms.** In the top, an edge between vertex 1 and 2 is augmented with two ancillary atoms 3 and 4. With no local detuning, the ground-state of the augmented edge corresponds to exciting $\{1, 2\}$. This however is inconsistent in the initial graph (1 and 2 cannot be simultaneously in the MIS). We therefore add local detuning to all ancillary atoms to ensure that the ground-state corresponds to the anti-ferromagnetic state. (b) Energy diagram of the spectrum as a function of the local detuning δ_i applied to the ancillary atoms. A red atom corresponds to an excited state. The independence condition of the original link is respected if $0.05 \times U \leq \delta_i \leq 0.95 \times U$, where U is the interaction between two neighbouring atoms. In these values of local detuning, the ground-state corresponds to the anti-ferromagnetic state.

In this example, there are 3 MISs in the augmented graph : $\{1, 4\}, \{2, 3\}$ which are acceptable solutions, but also $\{1, 2\}$ which does not correspond to a MIS of the original graph. With a reasonable global detuning $\delta > 0$, this latter state is actually the ground state of the chain. In order to guarantee that $\{1, 4\}, \{2, 3\}$ correspond to the ground-states of the Hamiltonian associated to the augmented path, we apply an additional local detuning δ_i to each ancillary atom with $\delta_i = J/2$, where $U = C_6/r^6$ is the interaction energy between two closest atoms of the augmented graph.

Keeping $0.5 \times U \leq \delta_i \leq 0.95 \times U$ ensures that the MIS returned by the algorithm preserves initial constraints.

We estimate in this proof the lower and upper bound for the local detuning on the two ancillary atoms of an augmented edge. Let $E_{i_1 i_3 i_4 i_2}$ be the energy associated to the bit-string $i_1 i_3 i_4 i_2$, where $i_k \in \{0, 1\}$ and k is the label of the atom $(1, 2$ are the main atoms, and $3, 4$ the ancillas. We want to ensure the following inequalities:

$$\begin{cases} E_{1010} < E_{1001} \\ E_{1010} < E_{0110} \end{cases} . \tag{3.58}$$

We show calculations in the case $\delta_i = 0$ for original atoms (1 and 2). We therefore have that

$$\begin{aligned} &\begin{cases} -2\delta - \delta_i + \frac{1}{2^6}U < -2\delta + \frac{1}{3^6}U, \\ -2\delta - \delta_i + \frac{1}{2^6}U < -2(\delta + \delta_i) + U, \end{cases} \\ \iff \ &\delta_i \in [(\frac{1}{2^6} - \frac{1}{3^6}) \times U, (1 - \frac{1}{2^6}) \times U]. \end{aligned} \tag{3.59}$$

Taking $\delta_i = U/2$ for all ancillary atoms is a safe spot. In the general case, we determine for each edge the corresponding value for the detuning that will be applied on all the ancillary atoms of that chain. This ensures that the ground state of the augmented Hamiltonian encodes a Maximum Independent Set of the original graph.

Overhead

The procedure described above enables us to replicate the connectivity of any input graph G of maximum degree six in three dimensions, at the expense of adding ancillary vertices. In order to assess the overhead incurred by this embedding, we test the procedure of general graphs of maximum degree 6. For each size, 20 Erdos-Renyi graphs are generated and the size N_+ of the augmented graph is recorded. Our simulations seem to indicate a linear overhead in the number of ancillary atoms, as illustrated in Figure 3.10 where we show the number of vertices in the augmented graph N_+ with respect to the size of the original graph N. Our method would be impractical if the number of ancillary atoms exploded or if the size of the edges grew exponentially with respect to the graph size. Indeed, the Lieb-Robinson bounds [LR72a] would imply that information could not propagate efficiently through the ancillary paths. Luckily, the necessary volume to draw the OGD of a graph was proven to be polynomially bounded [ESW96]. Precisely, every N-vertex degree-6 graph admits an OGD in $\mathcal{O}(N^{3/2})$ volume and that bound is best possible for some degree-6 graphs. The authors give an explicit algorithm that places all vertices on a $\mathcal{O}(N) \times \mathcal{O}(N)$ grid in a $2D$ plane and draws each edge with at most 16 bends. The growing number of atoms that can be experimentally trapped in recent experimental setups [**?**] is an encouraging sign that the overhead in the number of atoms required in our method is reasonable.

We described here the versatility of neutral atom platforms at solving hard graph problems. Previous implementations of MIS solutions using atoms trapped in optical tweezers were either limited to UD graphs or using a large number of ancillary qubits [NLW+23]. We have successfully illustrated that this approach can be extended to a larger class of graphs, by solving the MIS

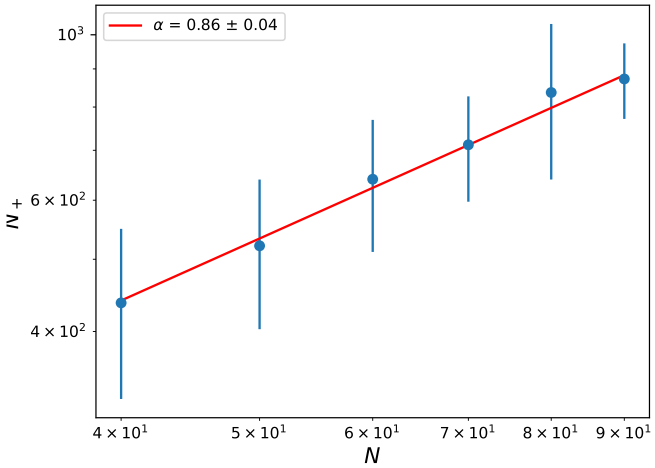


Figure 3.10: **Scaling of additional atoms in our method**. For each size, 20 random Erdos-Renyi graphs of maximum degree 6 are generated. Our method is then applied and we plot the mean size N_+ of the augmented graphs and the standard deviation of the sample. The power-law fit suggests a sub-linear growth of the required additional atoms.

problem on a minimal non-unit disk hard graph using a 3D array of atoms. The results showcase the validity of the approach for unit-ball graphs.

Furthermore, we described a method embedding the MIS problem over non-local graphs as Unit-Ball graphs in 3D space. This procedure is guaranteed to run in a time growing polynomially with the input graph size, and can be used, in particular, to encode bounded-degree graphs of maximum degree 6, for which no PTAS exists unless $\mathbf{P} = \mathbf{NP}$. As neutral atom platforms develop and benefit from additional hardware features, we expect our method to become more and more relevant. In particular, the validity of the OGD relies on the ancillary chains being in an antiferromagnetic state, which can be enforced using local detunings. A very intriguing and exciting prospect is to qualify both theoretically and experimentally the capabilities of quantum devices in approximately solving the MIS on this classes of hard graphs like the bounded-degree graphs. We believe that understanding from a theoretical perspective the capabilities of quantum approaches in finding guaranteed performance ratio for NP-complete problems is an important question in the quest for practical quantum advantage.

3.7 Perspective on many-body dynamics for combinatorial optimisation problems

Seeking quantum advantage on a combinatorial optimisation problem is a high risk/reward mission. While the applications are numerous and concern real-world practical problems, no one has convincingly proven a speed-up on combinatorial task with a quantum computer in practice. First, current quantum computers are obviously too small to map real-world hard problems: with a maximum of few hundreds of qubits individually mapped to a single variable, the problem can usually be solved by classical means. In the past years, the hope for quantum advantage on these tasks relied on the fact that at some point the classical computer has to brute-force the solution which corresponds to unstructured search. Grover's algorithm [Gro96] which promises a quadratic speed-up for unstructured search could therefore be used as a sub-routine to accelerate the computation. In practice however achieving Grover speed-up is very unlikely. With fault-tolerant quantum computers for example, an asymptotic quadratic speed-up will suffer from constant overheads considering the slow clock-time, lack of parallelisability and error-correction overhead compared to classical computers [DPCB23]. Indeed, studies have estimated that an error-corrected quantum computer with quadratic speed-up will surpass classical computers on combinatorial problems only if:

(i) the calculation time is an entire day with 10^5 logical qubits [CKM19]

(ii) the calculation time is one entire year with 512 logical qubits [SBC+20].

In other words, the time required for the quadratic speed-up to be faster in practice than a classical computer is on timescale much bigger than expected. These studies take into account the large overhead in error-correction schemes and the fact that a single CPU can run a billion operation per second. Nonetheless, these estimations become much more optimistic if the polynomial degree of the speed-up increases. For example, a quartic ($T^{\frac{1}{4}}$) speed-up would yield a quantum advantage after just 5 hours of calculations (in comparison to a year for quadratic speed-up) [SBC+20]. Following this direction, a recent paper has presented rigorous super-Grover speed-ups for combinatorial optimisation problems [DPCB23], paving the way towards more reachable quantum advantage. Excitingly, the authors use a twisted version of a quantum annealing algorithm to prove their speed-up. As of today, the most promising schemes to achieve a practical speed-up rely on variations of annealing procedures [EKC+22, CL21, YONL19], a very natural type of quantum evolution to be run on neutral atom quantum processors. Known to have long coherence times and low overheads, neutral atom platforms have successfully explored many-body dynamics that are unreachable for classical simulations including quantum quenches [Lab16], topological exotic phases [SLK+21] or continuous-symmetry breaking phenomenon [CBB+23]. Note that even a digital quantum computer using Trotter-Suzuki decomposition would require itself 10^6 qubits and an hour run-time to replicate just 100 steps of the dynamic on 256 atoms of a neutral atom device [SBC+20]. In my opinion, neutral atoms platforms have demonstrated clear advantage in simulating quantum dynamics of spin glasses in excited regimes and we should therefore pursue efforts in this direction to find traces for quantum speed-ups on combinatorial tasks. Recent works assess the use of such quantum effects like many-body localisation [WYK22] or quantum critical dynamics [KRL+23] to achieve quantum speed-ups; I believe these are very exciting research directions.

Part II

Quantum supervised machine learning with Rydberg atoms

Chapter 4

A gentle introduction to quantum machine learning (QML)

Machine learning is the art and science of making computers learn how to solve problems instead of being explicitly programmed by learning on data [SP21]. The well-known problem of hand-written digit classification is a good example to showcase the capacity of machine learning algorithms to learn patterns and extract features from complex data. In this problem, a *dataset* consisting of hand-written digits and their associated value are given to the algorithm. The raw input consists of pixel values representing the intensity of each pixel, and traditional rule-based algorithms would struggle to define precise rules for classifying each possible variation of a digit. Machine learning however can learn and discern relevant features that differentiate one digit from another without relying on explicit classification rules.

Mathematically, given an input domain $\mathcal{X} = \{x_1, \ldots, x_M\}$ and an output domain $\mathcal{Y}$, the dataset consists of M point $\mathcal{D} = \{(x_1, y_1), \ldots, (x_M, y_M)\}$. Given a new unclassified input $x \in \mathcal{X}$, the goal is to predict the corresponding output $y \in \mathcal{Y}$. The idea to do so is to build a model family $\{f(x, \theta) : \mathcal{X} \times \Theta :\rightarrow \mathcal{Y}\}$, parameterised by θ, and pick the model that minimises the error of the model over the dataset. If we consider $\mathcal{X}$ and $\mathcal{Y}$ as Euclidean spaces, a typical error function that is widely used is the *empirical risk* that reads

$$E(\theta) = \frac{1}{M} \sum_{i=1}^{M} \|f(x_i, \theta) - y_i\|^2, \tag{4.1}$$

and represents the mean of the sum of the square of errors between the predictions $f(x_i, \theta)$ and the corresponding target value y_i. The *training* of the model minimises the empirical risk by using gradient descent techniques [DHS11, Lem12]. Once the model is trained it can be tested to predict the value of unseen datapoints in a process called *generalisation*. The ability to generalise is crucial because the ultimate purpose of a machine learning model is to make predictions on unseen data, rather than merely memorising the training examples. Ideally, a well-generalised model will exhibit similar performance on both the training and test sets, indicating that it has learned the underlying patterns and can make accurate predictions on new data.

The field of ML attracted increasing attention with the development of deep learning, where artificial neural networks with multiple layers demonstrated the capacity to learn hierarchical representations of data, allowing for more complex and abstract feature extraction. In 2017, the introduction of transformers [VSP+17] revolutionised the field of natural language processing (NLP): unlike traditional sequential models, transformers use a self-attention mechanism that allows them to capture long-range dependencies in a sequence of data. Transformers have become the backbone of many state-of-the-art models in NLP, including the popular GPT, BERT and Llama models [Ope23, DCLT19, TLI+23]. All of these impressive tools have impacted a diversity of fields including image and speech recognition [KSH17, THC23], autonomous systems [BDTD+16, ABC+20], recommendation systems [CAS16] or protein folding [JEP+21].

The field of experimental quantum physics has also been transformed by machine learning and the most impressive example took place in Germany in 2017. At that time, quantum optics researcher Mario Krenn and his team were banging their heads against the blackboard as they had been trying to come up with an experimental set-up to observe high-dimensional multipartite-entanglement. After several weeks, Krenn developed a machine learning model named Melvin [KMF+16] that was given access to a toolbox containing typical optical devices that are accessible in a quantum optics lab: beam splitters, phase shifters, wave plates, holograms, dove prisms and more complex devices such as effective single-photon filters. Melvin was free to place any elements on the virtual optical table, at the end of which the resulting state was calculated and compared to the target state. Considering approximately 120 different optical devices in the toolbox and space for 15 of them on the optical table, the number of possible set-ups amounted to $120^{15} \simeq 10^{31}$ different configurations. Astonishingly, Melvin identified within a few hours a protocol that seemed able to prepare the state. A couple of years later, Melvin's protocol was refined and realised experimentally, showing in practice the long-sought-after high-dimensional entangled state [EMKZ18]. Melvin is an impressive example of how machine learning can help quantum scientists and as such fits the definition of quantum ML.

Quantum ML (QML) is an umbrella term that groups many different approaches, depending on the way one decides to mix quantum with classical. An interesting distinction was proposed in Ref [SP21], where the authors separate the field in four according to the nature of the data being examined and the nature of the device processing the data. If we focus on the type of device and data that is used in QML, there are actually four different sub-fields that can be distinguished.

1. **(CC)** The first one corresponds to classical data being analysed with a classical device (CC class) but with a *quantum-inspired* method. Examples of QML that fit this class are applications of tensor networks for neural network training [GPC18] and also (amusingly) quantum algorithms that have been dequantised, *i.e.* that can be implemented efficiently classically with conserved speed-ups [ADBL19].

2. **(QQ)** On the other side of the spectrum, we can define QML tasks where the data is itself quantum and analysed with a quantum device (QQ class). We could imagine quantum simulations preparing ground-states and, feeding those quantum states to a quantum computer, use specific quantum methods to analyse them [CCL19a]. Note that this approach requires to store and extract a quantum state, which is a very tricky task due to decoherence and as such QQ ML is seen as a long-term perspective.

3. **(QC)** The third class corresponds to analysing quantum states using classical ML tools. With the recent emergence of available quantum hardware, scientists can prepare ground-states of interest but are often limited in the number of measurements they can do of the system. ML scientists have come up with a scheme to nonetheless predict many properties of a quantum system from very few measurements [HKP20]. Furthermore, experimentalists can also face hardware constraints that limit the exploration of phase space. Once again, classical ML techniques have been successful in predicting from few samples the whole phase space, putting forward interesting transitions locations for physicists [HKT+21].

4. **(CQ)** Finally, the class of interest in this thesis corresponds to classical data being analysed with a quantum computer (CQ class). In the following chapters, our definition of QML will be focused on classical data being processed on a quantum device. Historically, it was of theoretical interest to understand if quantum computing could speed-up ML sub-routines in the same fashion than Shor's algorithm speeds-up factorisation [Sho94]. Dominated by a complexity theory mindset, scientists worked to prove quadratic or exponential speed-up on linear algebra routines, the most famous one being the Harrow-Hassidim-Lloyd (HHL) algorithm [HHL09] that promises an exponential speed-up over classical methods for solving a specific linear system of equations. Precisely, provided a N-linear system is sparse and has a low condition number κ and that the user is interested in the result of a scalar measurement on the solution vector, the HHL algorithm has a run-time of $\mathcal{O}(\log N\kappa^2)$ compared to the classical run-time $\mathcal{O}(N\kappa^2)$, offering an exponential speedup. Although the exponential speed-up is impressive, this algorithm has been criticised over the assumptions made on data access and the matrix being “well-behaved" [Aar15].

While the first wave of QML was dominated by a complexity theory point of view, the development and availability of Noisy-Intermediate Scale Quantum (NISQ) devices in the past years shifted the focus of QML towards a *hands-on* approach. Indeed, scientists started focusing on the capabilities of quantum devices to act as ML models themselves rather than being sub-routines accelerating classical ML methods. Very interesting results have emerged in the understanding of the types of problems that can be embedded on NISQ devices by taking into consideration hardware capabilities and limitations. Examples on IBM, Google, Xanadu and other devices have attracted quite some attention.

In the following chapter, we will explore in the same fashion the capabilities of Rydberg-atom QPUs in QML tasks. Then, we will try to understand in our final chapter where resides the advantage of quantum expressivity.

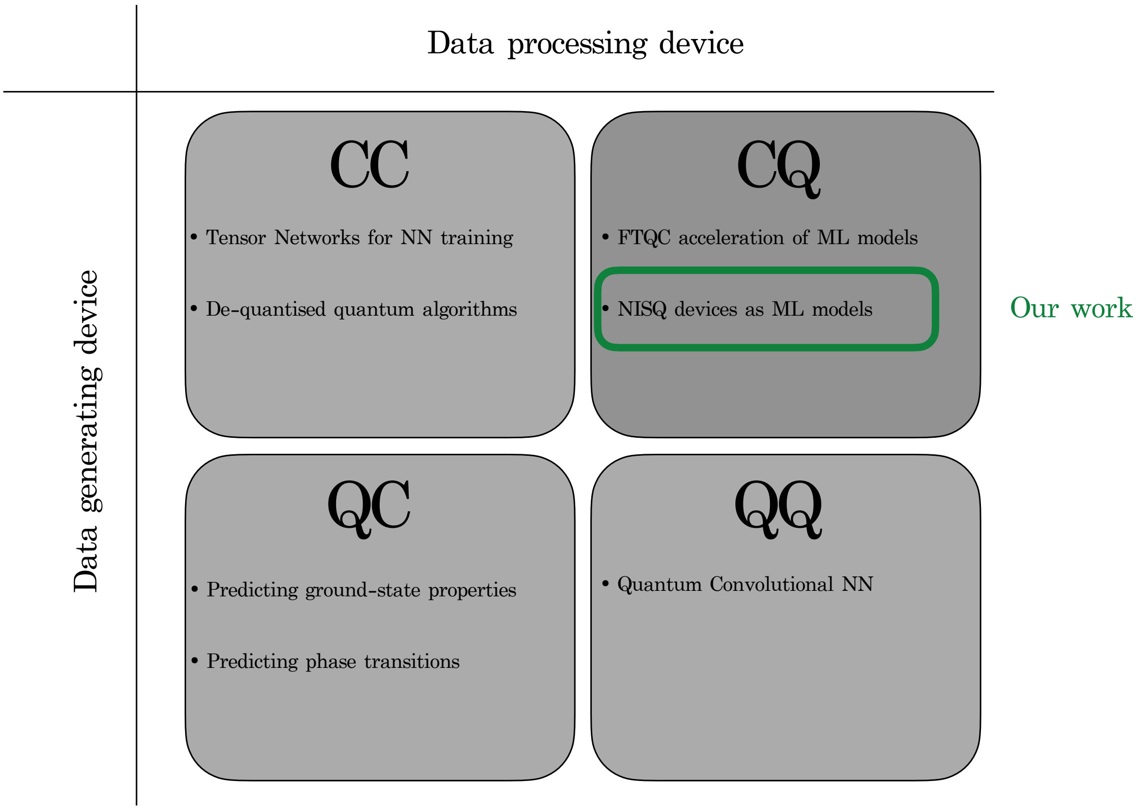


Figure 4.1: **The four quadrants of quantum machine learning (QML).** Depending on the device generating the data and the one treating it, QML can mean very different things. In this thesis, we focus on using a neutral-atom quantum processor as a machine learning model to treat classical data. The goal is to understand what types of ML problems can be embedded on the device and where and why we can expect some quantum advantage.

CHAPTER 5

GRAPH MACHINE LEARNING WITH NEUTRAL ATOM QPUS

As we have seen in the first part of this thesis, it is possible to replicate to some extent the adjacency matrix of a graph with the interaction matrix of Rydberg atoms. Graphs are not only suited to represent combinatorial problems, but they are also a natural representation of data in many domains of sciences. They naturally describe relationships in social networks [Fre00], characterise interactions of proteins and genes [TvDEF09] and can also represent the structure of sentences in linguistics [S^+01]. Many impactful applications arise from efficient graph-based methods, such as predicting potential edges in recommendation systems [SKR01], detecting frauds in communication networks [POKB20], or for protein function prediction [MOB20].

While graphs offer a rich structure for manipulating complex data, it is unfortunately difficult to apply linear algebra tools directly to these kinds of datasets: for example two graphs might have the same adjacency matrix but different labels on the nodes and on the edges. The high level of freedom that graph-structured data presents therefore comes with a resource-consuming data analysis. An important research topic is to therefore find good graph embedding techniques [GF18], which refer to finding a representation of a graph or of its individual nodes in a vector space. By finding node representatives which preserve different types of relational information from the graph, node embedding can be used for prediction tasks at the node-level, such as node classification [BCM11] or link prediction [LNK03]. Embeddings can also be done at the graph-level to distinguish graphs of different nature: using the exponentially large Hilbert space accessible to a quantum computer in order to generate graph embeddings is an appealing idea, with many proposals and theoretical studies over the past few years [SK19, HCT+19, SBI+20, KSR+21]. With the recent advances in geometric quantum machine learning, works have shown how graph-structured data could be encoded into quantum states and manipulated for classification, clustering or regression tasks. These efforts started with quantum convolutional neural networks [CCL19b, ZGL21], and attempts were made to translate classical Graph Neural Network (GNN) architectures to Quantum Neural Networks (QNN) [VML+19a]. Theoretical studies of geometrical quantum machine learning and their invariant properties include [LSS+22a, SCY+23], the latter studying applications to weighted graphs. More recently, in-depth theoretical studies of equivariant and geometric quantum machine learning aspects were presented [RBN+22, NSB+22, SLSC22].

In this chapter, we will first show formally how a neutral-atom QPU can be seen as a learning model for graph-based data (section 5.1). Then, we will explore in section 5.2 the expressivity of the feature map created by our learning model. Finally, we will implement our method on hardware on a practical classification task of protein toxicity in section 5.3. This chapter is summarised in the paper

- [ADL+23] Dalyac, C., Albrecht, B., Leclerc, L., Ortiz-Gutiérrez, L., Thabet, S., D'Arcangelo, M., ... Henriet, L. (2023). *Quantum feature maps for graph machine learning on a neutral atom quantum processor.* Physical Review A, 107(4), 042615.

5.1 A parameterised Hamiltonian evolution is a learning model

In the introduction to Part II, we defined a machine learning model as a family of parameterised function $\{f(x,\theta) : \mathcal{X} \times \Theta :\rightarrow \mathcal{Y}\}$. A variational analog quantum processing scheme can easily fit this definition, if we look at it in the following way:

Starting from a graph $\mathcal{G}$ reproduced in the array of tweezers with qubits in state $|0\rangle$, we apply a parameterized laser pulse onto the atoms in order to generate a wavefunction $|\psi_{\mathcal{G}}\rangle$ of the form

$$|\psi_{\mathcal{G}}\rangle = U(\mathcal{G};t)\,|0\rangle^{\otimes|\mathcal{G}|}, \tag{5.1}$$

where we define the time-evolution operator $U(\mathcal{G};t)$ as

$$U(\mathcal{G};t) := \mathcal{T}\left[\exp\left(-i/\hbar\int_{s=0}^{t}\hat{\mathcal{H}}_{\mathcal{G}}(s)ds\right)\right]. \tag{5.2}$$

The time-evolution operator $U(\mathcal{G};t)$ depends on the data graph $\mathcal{G}$ and on the parametrisation of $\hat{\mathcal{H}}_{\mathcal{G}}(s)$ over time. Then, a quantum observable $\hat{O}$ is measured and the function

$$f(\mathcal{G},\theta) = \langle\psi_{\mathcal{G}}|\,\hat{O}\,|\psi_{\mathcal{G}}\rangle\,, \tag{5.3}$$

defines a deterministic quantum learning model in the sense of Ref [Sch21]. We will call *quantum feature map* the feature map that is yielded by our quantum learning model for graph-structured data. Note that we can use this feature map as an input for building more complex feature

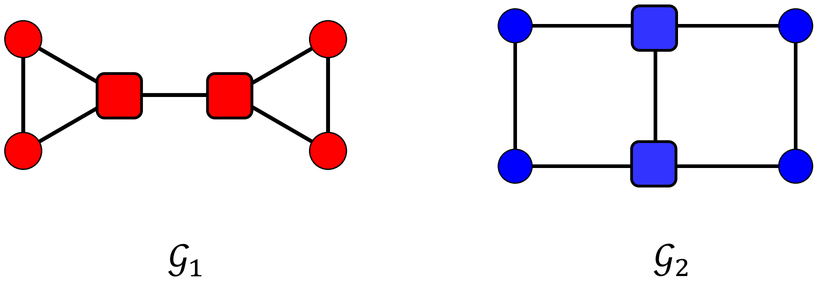


Figure 5.1: **Two graphs that are locally equivalent but non-isomorphic.** For each graph, the nodes can be separated in two classes; the border ones (circle nodes) and the centre ones (square nodes). Interestingly, while it is easy to see that these two graphs are not isomorphic, nodes have the same neighbourhood in both graphs. Indeed, in each graph the circle nodes have degree two and square nodes have degree three. Moreover, circle nodes are adjacent to a circle node and a square node, and each square node is adjacent to another square node and two circle nodes. All-in-all, they are indistinguishable by their local structure but non-isomorphic.

maps, as we can decide for example to add post-processing blocks by looking at the distribution of an observable, an idea that will be explored in a future section. As a first step, we explore in the following section the expressive power of our graph quantum feature map.

5.2 Insights on the expressive power of the graph quantum feature map

The graph quantum feature map already shows interesting properties when associated with single-body observables $\langle \hat{O}_i \rangle$ for $i \in \{1, \dots, |\mathcal{G}|\}$. The measured values are not only affected by local graph properties such as node degrees, but also by more global ones such as the presence of cycles. This enrichment provided by the quantum dynamics contrasts with the locality of node representations in many classical graph machine learning schemes. This key feature comes from the fact that the quantum dynamics of a given spin model is significantly influenced beyond short times by the complete structure of the graph as the quantum correlation length grows with time [LR72b, TGS$^+$19].

We illustrate experimentally this behaviour for two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ that are represented in Figure 5.1. In these graphs, nodes can be separated into two equivalence classes according to their neighbourhood: border nodes B have one degree-3 neighbour and one degree-2 neighbour, while centre nodes C have two degree-2 neighbours and one degree-3 neighbour. We will see that the presence of interactions will enable us to discriminate between $\mathcal{G}_1$ and $\mathcal{G}_2$ by comparing the dynamics of local observables on border and centre nodes.

These graphs are quite famous as they are one of the smallest example of non-isomorphic graphs for which the Weisfeiler-Lehman (WL) test [WL68] fails. The WL test is a polynomial-time graph isomorphism algorithm that works most of the time. It is known for its local nature, as it directly relies on the neighbourhood of each vertex in the graph. Precisely, the algorithm starts by assigning each vertex a unique label based on its immediate neighbourhood, considering the vertices it is directly connected to. Then, in subsequent iterations, the labels are refined by incorporating information from the labels of the neighbouring vertices and their respective neighbourhoods as well. This iterative process continues until either a distinguishing feature is found or the labels stabilise [1]. By focusing on the local neighbourhood information, the Weisfeiler-Lehman test captures the local structure of the graphs being compared. It exploits the idea that if two graphs are isomorphic, their vertices and their respective neighbourhoods should exhibit similar label patterns throughout the iterations. Conversely, if the graphs are not isomorphic at some point in the iterations the labels will capture distinguishing characteristics that reveal the structural differences. The WL test is also related to standard message-passing neural networks (MPNN) [GSR+17] architectures where information is only propagated along the graphs edges. In particular, these successful models [MLLK22] have been proven to be at most as powerful in distinguishing graph structures [XHLJ19]. While the WL test is quite powerful, it fails on the two graphs we represent in Figure 5.1. It is therefore interesting to see what happens when embedding these graphs with atoms and letting them evolve under a quantum Ising Hamiltonian.

Experiment

The first step is to embed the graphs on the hardware. To do so, we prepare a tweezers array with a nearest-neighbour (NN) distance of $r_{NN} = 5.3\,\mu$m. Once the atoms are trapped and re-arranged to respect the layout of the graphs, we apply the Ising Hamiltonian with a constant pulse with $\Omega/2\pi = 1.0\,$MHz and $\delta/2\pi = 0.7\,$MHz. We then measure the mean Rydberg excitation $\langle n_i \rangle$ for varying pulse duration $t \in [0, 2.5]\,\mu$s (Figure 5.2). Because the dynamics only depend on being a centre or border node, we limit our study to one representative of each class, for each graph. As illustrated in Figure 5.2, a qualitative difference in the dynamics of both graph appears after $t \sim 0.25\,\mu$s.

Precisely, the excitation of the border nodes is initially increasing with indistinguishable behaviour between the two graphs before a distinction appears. The mean density for the border qubits of $\mathcal{G}_1$ exhibits damped oscillations around $\langle n_B \rangle \sim 0.15$ with period of the order of $0.5\,\mu$s while for $\mathcal{G}_2$ it exhibits flatter oscillations centred around 0.25 with period around $1\,\mu$s. We can observe a comparable distinction between the two graphs for the centre qubits.

When restricted to the mean-field approximation, the qubit dynamics on either graphs are far more similar as illustrated in the insets of Figure 5.2. We still observe distinct dynamics between the two graphs, which is due to next nearest neighbours (NNN) interactions. If we neglected those NNN interactions, the mean-field equations governing the dynamics of each

---

[1] An excellent visualisation and introduction can be found in Ref [BBCV21].

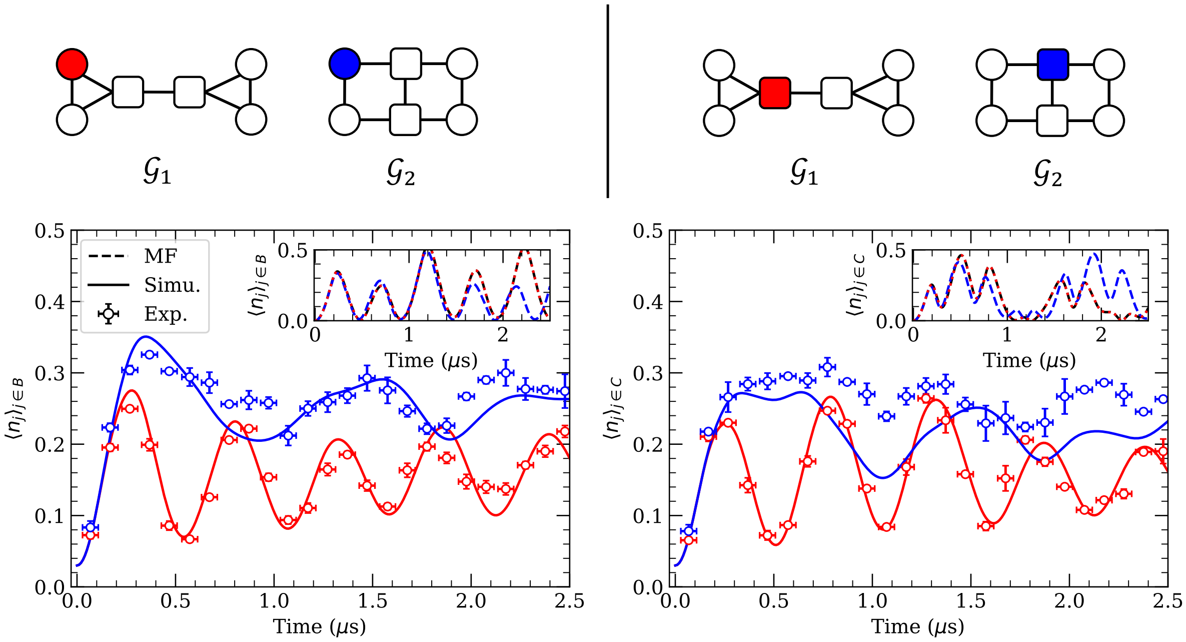


Figure 5.2: **Different Rydberg dynamics for locally-equivalent nodes.** We measure the mean Rydberg excitation $\langle n_i \rangle$ for varying pulse duration $t \in [0, 2.5]\,\mu$s. Because the dynamics only depend on being a centre or border node, we limit our study to one representative of each class, for each graph. Precisely, we plot the evolution of the mean occupation $\langle n_i \rangle$ of the two regions B (left) and C (right) for both graphs $\mathcal{G}_1$ (red) and $\mathcal{G}_2$ (blue). The dots represent the experimental results while the full curves show noisy simulation results. Horizontal error bars account for the sequence-trigger uncertainty ($\approx$ 40ns) while the vertical ones account for the sampling noise. The insets show the corresponding mean field dynamics (dashed) with only NN (black) or full (coloured) interactions.

qubit would only depend on its direct neighbourhood and as a result the whole dynamics would solely depend on the local structure of the graph. In that case, the qubits dynamics for $\mathcal{G}_1$ and $\mathcal{G}_2$ obey the exact same equations.

To conclude this analysis, we have seen experimentally that evaluating local observables in the presence of interactions enables to discriminate between two graphs that are nonetheless locally equivalent. We can also quantify these difference by looking at a global observable.

Global observable

In the following part, we choose the general observable

$$\hat{O} = \sum_{i=1}^{6} \hat{n}_i, \tag{5.4}$$

to quantify the difference in the dynamics between the two graphs. In order to do so, we first compute the histogram $\mathcal{P}_i$ of number of excitations observed in each shot on graph $\mathcal{G}_i$. The

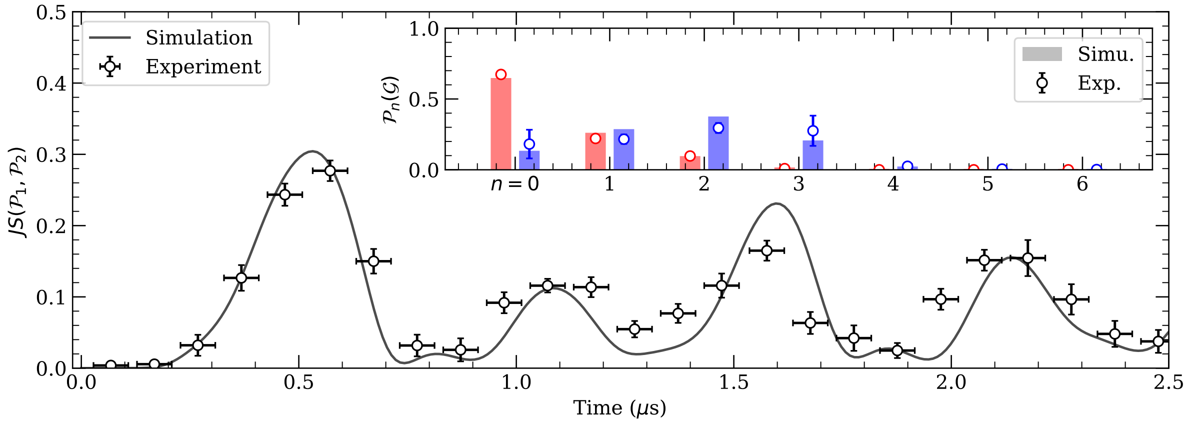


Figure 5.3: **Evolution of the Jensen-Shannon divergence of a global observable**. The experimental values (dot) are compared to the noisy simulation (plain). At each point in time, $JS(\mathcal{P}_1, \mathcal{P}_2)$ is computed using the excitation distributions $\mathcal{P}_{1/2} = \{\mathcal{P}_n(\mathcal{G}_{1/2})\}_{n=0\ldots6}$ obtained either numerically (bar) or experimentally (dot). The inset depicts $\mathcal{P}_{1/2}$ obtained at $t \approx 0.57$ μs which yields the maximum value $JS_{max} \approx 0.28$ reached.

difference between those graphs is then estimated via the Jensen-Shannon divergence JS of their respective histograms [BH13], a commonly-used distance measure between probability distributions, which is defined as

$$JS(\mathcal{P}_1, \mathcal{P}_2) = H\left(\frac{\mathcal{P}_1 + \mathcal{P}_2}{2}\right) - \frac{H(\mathcal{P}_1) + H(\mathcal{P}_2)}{2}. \tag{5.5}$$

Here $H(\mathcal{P}) = -\sum_k p_k \log p_k$ is the Shannon entropy of $\mathcal{P} = (p_1, \ldots p_{|\mathcal{G}|})$. The Jensen-Shannon divergence is minimal $JS(\mathcal{P}, \mathcal{P}) = 0$ if $\mathcal{P}_1 = \mathcal{P}_2 = \mathcal{P}$ and is maximal if $\mathcal{P}_1$ and $\mathcal{P}_2$ have disjoint supports, with $JS(\mathcal{P}_1, \mathcal{P}_2) = \log 2$. This is illustrated in

Figure 5.3 where the largest difference $JS_{max} \approx 0.28$ is achieved (roughly 40% of the maximal value) at a time $t \sim 0.57\mu s$. At this duration, the distribution for $\mathcal{G}_1$ is sharply peaked at $n = 0$ while that of $\mathcal{G}_2$ is wider and peaks around $n = 2$, as illustrated in the inset. We note that the local observables $\langle n_j \rangle_{j \in B/C}$ exhibit maximal deviation at this same duration t, indicating direct correspondence between measurements at the node and graph levels. On a more general aspect, the dependency of local observables evaluated after the application of the quantum feature map on global graph structures have been considered in quantum-enhanced versions of GNNs [VML+19b, MMC22, TFH22].

Noise model

Despite the precise calibration of the control devices which enable to monitor quantities such as the SLM pattern spacing or the pulse shapes, several experimental imperfections may alter the data measured on the experiment. Among the many different noise sources that were detailed in Part I, the most affecting ones are statistical sampling noise, SPAM errors and decoherence.

First, due to the nature of the quantum state and the limited budget of shots, measurements are subject to sampling noise. For instance, on average, each of the 25 experimental points on Figure 5.3 is obtained using 600 shots and the uncertainty related to this effect (vertical error bars) is estimated using the Jackknife resampling method [ST95].

The finite sampling is also inherently flawed by several physical processes like atoms thermal motion, background-gas collisions or Rydberg state finite lifetime, whose effects can all be encompassed as first approximation into the two SPAM parameters ε and ε'. On our device, we measure $\varepsilon \approx 3\%$ and $\varepsilon' \approx 8\%$; thus as an example, we can compute $P_{1001|0101} = \varepsilon\varepsilon'(1-\varepsilon)(1-\varepsilon') \approx 0.2\%$. Those detection errors can deeply modify the measured excitation distributions, with a noticeable effect shown on Fig. 5.3 at $t = 0$ where the simulated $\langle n_j \rangle$ does not start at 0 despite $|\psi(t=0)\rangle = |0\ldots 0\rangle$.

Additional errors can also lead to decoherence in the system [DLBL+18], affecting the atom dynamics in ways costly to emulate. In order to replicate the experimental data presented in Figure 5.3 we resort to an effective decoherence model in the form of solving the Master equation with a relaxation rate of $2\pi \times 0.06$ MHz [BVC+13]. This value was obtained by fitting with the above model damped Rabi oscillations measured on the same device. Thus, reaching similar behaviour within error bars between numerically simulated and experimentally obtained $JS(\mathcal{P}_1, \mathcal{P}_2)$ was achieved with no free parameter.

In the following section, we recall the link between a feature map and a kernel and expose many variants of classical graph kernels. This helps to understand the particularities of our quantum evolution kernel (QEK).

5.3 From a quantum feature map to a kernel: the Quantum evolution kernel (QEK)

5.3.1 Classification in ML using feature maps and kernels methods

Suppose that we are given a dataset of M points $\{(x_i, y_i)\}_{i=1,\ldots,M}$ where $x_i \in \mathbb{R}^2$ and $y_i \in \{-1, +1\}$ are two distinct classes represented in Figure 5.4. Because this dataset is distributed in concentric circles, blue and red datapoints are not linearly separable in $\mathbb{R}^2$.

To make the data linearly separable, a good feature map is

$$\phi(z) = (z_1, z_2, z_1^2 + z_2^2), \tag{5.6}$$

where $z = (z_1, z_2) \in \mathbb{R}^2$ is a data-point. This feature map makes the data linearly separable in $\mathbb{R}^3$. The goal of our classification task now consists in defining regions of the feature space that correspond to different classes. In the case where the data is linearly separable, we can

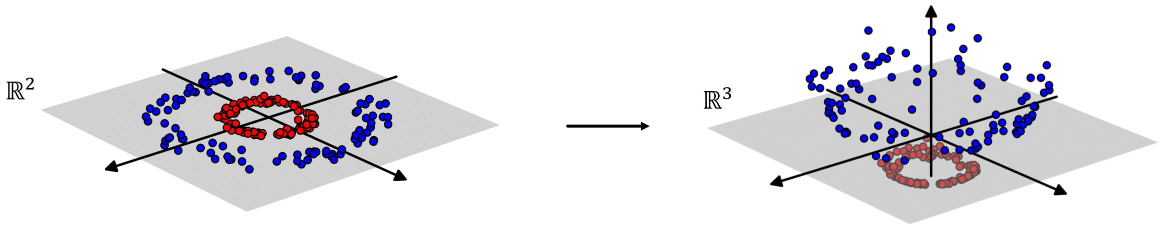


Figure 5.4: **A non-linear separable dataset can be linearly separated when mapped to a higher dimensional space.** A dataset of 2 two-dimensional concentric circles is impossible to separate by a linear decision boundary (right): using the feature map $\phi(z) = (z_1, z_2, z_1^2 + z_2^2)$ transforms the data into a linear separable dataset (left).

use a *linear discriminant* for which the decision boundary between the different regions is a hyperplane. The simplest representation for a linear discriminant function is given by

$$y(x) = \mathbf{w}^T \phi(x) + \mathbf{b}, \tag{5.7}$$

where $\mathbf{w}$ is a weight vector of size 2 and $\mathbf{b}$ is sometimes called a *bias*, an offset that we set to zero by supposing that the data is centred. Training the model consists in finding the vector $\mathbf{w}*$ that minimises a regularised sum-of-squares error function given by

$$J(\mathbf{w}) = \frac{1}{2M} \sum_{i=1}^{M} |\mathbf{w}^T \phi(x_i) - y_i|^2 + \frac{\lambda}{2} ||\mathbf{w}||^2, \tag{5.8}$$

$$= \frac{1}{2M} ||\mathbf{\Phi} \mathbf{w} - \boldsymbol{y}||^2 + \frac{\lambda}{2} ||\mathbf{w}||^2, \tag{5.9}$$

where $\mathbf{\Phi}$ is a matrix of size $M \times 2$ with each row i corresponds to $\phi(x_i)^T$ and $\boldsymbol{y}$ is the vector of all labels y_i. Note that the first term corresponds to the Mean Square Error (MSE) while the second term represents the ridge regularisation and prevents the weights from exploding. The magnitude of the regularisation is controlled by a hyperparameter $\lambda > 0$.

Here, an analytic solution to the problem is given by setting the gradient of $J(\mathbf{w})$ with respect to $\mathbf{w}$ equal to zero and reads

$$\mathbf{w}^* = (\mathbf{\Phi}^T \mathbf{\Phi} + \lambda I)^{-1} \mathbf{\Phi}^T \boldsymbol{y}. \tag{5.10}$$

The dual formulation of this problem is given by expressing $\mathbf{w}$ as a linear combination of the data points $\mathbf{w} = \mathbf{\Phi}^T \boldsymbol{\alpha}$. The minimisation on $\mathbf{w}$ become a minimisation on $\boldsymbol{\alpha}$ and can be expressed as

$$\boldsymbol{\alpha}^* = \underset{\boldsymbol{\alpha}}{\text{argmin}} \frac{1}{M} ||\mathbf{\Phi}\mathbf{\Phi}^T \boldsymbol{\alpha} - \boldsymbol{y}||^2 + \lambda \boldsymbol{\alpha}^T \mathbf{\Phi}\mathbf{\Phi}^T \boldsymbol{\alpha}. \tag{5.11}$$

$$\tag{5.12}$$

In this dual formulation, the solution to the problem reads

$$\boldsymbol{\alpha}^* = (\boldsymbol{\Phi}\boldsymbol{\Phi}^T + \lambda I)^{-1}\boldsymbol{y}. \tag{5.13}$$

Note that the dual solution only depends on the matrix of scalar products between feature vectors $\boldsymbol{\Phi}\boldsymbol{\Phi}^T$, whereas the covariance matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ appears in the primal version. We can now define the Gram matrix

$$K := \boldsymbol{\Phi}\boldsymbol{\Phi}^T, \tag{5.14}$$

which is an $M \times M$ symmetric matrix with elements $K_{ij} = k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$. In machine learning, k is called a kernel function and is a similarity measure between data points. The kernel function is designed in such a way that it corresponds to the inner product of the data points in the high-dimensional feature space. Formally, it is defined as following:

Definition 5.1 ([Sch21])**.** *Given a data domain $\mathcal{X}$, a kernel is a positive semi-definite bivariate function $k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$. Positive semi-definiteness means that for any dataset of points $\mathcal{D} = \{x_1, \ldots, x_M\} \subset \mathcal{X}$, the Gram matrix K with entries*

$$K_{m,m'} = k(m, m'), \tag{5.15}$$

is positive semi-definite. As a consequence, $k(m, m') \geq 0$ and $k(m, m') = k(m, m')^*$.

The interesting property of kernel functions is that they can be used directly in the linear regression model instead of the explicit feature vectors. Indeed, we can now re-write the linear regression model (equation 5.7, taking $\mathbf{b} = 0$) as

$$y(x) = \mathbf{w}^{*T}\phi(x) = \boldsymbol{\alpha}^{*T}\boldsymbol{\Phi}\phi(x) = k(x)^T(K + \lambda I)^{-1}\boldsymbol{y}. \tag{5.16}$$

where $k(x) = (k(x, x_0), \ldots, k(x, x_M))$. As such, the dual formulation allows the solution to the least-squares problem to be expressed entirely in terms of the kernel function k. In some cases, it can be interesting to come up directly with a kernel function rather than an explicit feature map. Because a kernel is a similarity measure between datapoints, it might happen that for some classification tasks one has a good intuition for a kernel function. For example, the Gaussian kernel is given by

$$k(x_i, x_j) = e^{-\gamma\|x_i - x_j\|^2}, \tag{5.17}$$

where $\gamma > 0$ is a hyperparameter. Using the series expansion of the exponential we can write this kernel as

$$\begin{aligned} k(x_i, x_j) &= e^{-\gamma\|x_i - x_j\|^2}, & (5.18)\\ &= \sum_{j=0}^{\infty} \frac{(x_i^T x_j)^k}{k!} e^{\frac{1}{2}\|x_i\|^2} e^{-\frac{1}{2}\|x_j\|^2}, & (5.19)\\ &= \langle \phi(x_i), \phi(x_j) \rangle. & (5.20) \end{aligned}$$

where $\phi(x_i)$ is a feature vector of infinite size. Another great advantage of kernels is that they can handle symbolic objects, *i.e.* not only vectors but more general objects... like graphs. In the following section, we describe a number of classical graph kernels that will be confronted to our quantum graph kernel. Then, we will describe how classifications for graphs can be done using graph kernels using the well-known Support Vector Machine (SVM) algorithm.

5.3.2 Classical graph kernels

In this subsection, we describe a couple of classical kernels that will be used in the main text to benchmark the quality of our quantum kernel.

Size kernel

A very naive kernel but that will give us interesting insights is called the size kernel. Given two graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, the size kernel is defined as:

$$K_{\text{size}}(\mathcal{G}_1, \mathcal{G}_2) := e^{-\gamma(|\mathcal{V}_1|-|\mathcal{V}_2|)^2}, \tag{5.21}$$

with a choice of hyperparameter $\gamma > 0$. It echoes the Gaussian kernel presented in the previous section and considers that two graphs are similar if their sizes are close.

SVM-ϑ kernel

The Lovasz-ϑ kernel is a graph kernel that is based on the Lovasz extension of graph isomorphisms. It measures the similarity between graphs by comparing their induced subgraphs. However, computing the Lovasz-ϑ kernel can be computationally intensive, especially for large graphs, which may limit its practicality in some scenarios. The SVM-ϑ kernel was proposed as an alternative to the more computationally intensive Lovasz-ϑ kernel. Both ϑ kernels leverage the so-called orthogonal representation of a graph. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the orthogonal representation is an assignment of unit vectors $\{\mathbf{u}_i\}$ to each node of the graph, subject to the constraint that unit vectors associated to vertices that are not joined by an edge are orthogonal: $\langle \mathbf{u}_i, \mathbf{u}_j \rangle = 0$ if $\{i, j\} \notin \mathcal{E}$.

Orthogonal representations are not unique, but there is a particular representation associated with the ϑ number [Lov79] of a graph. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with n vertices, denote $U_{\mathcal{G}}$ an orthogonal representation of $\mathcal{G}$, and C the space of unit vectors in $\mathbb{R}^n$. The ϑ number is defined as:

$$\vartheta(\mathcal{G}) := \min_{\mathbf{c}\in C} \min_{U_{\mathcal{G}}} \max_{\mathbf{u}_i \in U_{\mathcal{G}}} \frac{1}{\langle \mathbf{c}, \mathbf{u}_i \rangle^2}. \tag{5.22}$$

From now on, we will always be referring to the particular orthogonal representation $U_{\mathcal{G}}$ that minimizes (5.22).

Now consider a subset of vertices $B \subset \mathcal{V}$, and call $U_{\mathcal{G}|B}$ the orthogonal representation obtained from $U_{\mathcal{G}}$ by removing the vectors that are not in B:

$$U_{\mathcal{G}|B} := \{\mathbf{u}_i \in U_{\mathcal{G}} : \ i \in B\}. \tag{5.23}$$

Note that $U_{\mathcal{G}|B}$ preserves the global properties encoded in $U_{\mathcal{G}}$ through the orthogonal constraint, and that $U_{\mathcal{G}|B}$ is not in general the orthogonal representation of the subgraph of $\mathcal{G}$ containing only the vertices in B. Define the ϑ_B number:

$$\vartheta_B(\mathcal{G}) := \min_{\mathbf{c} \in C} \max_{\mathbf{u}_i \in U_{\mathcal{G}|B}} \frac{1}{\langle \mathbf{c}, \mathbf{u}_i \rangle^2}. \tag{5.24}$$

We are ready now to give the definition of the Lovasz-ϑ kernel. Given two graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$, $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, define:

$$K_{\text{Lo}}(\mathcal{G}_1, \mathcal{G}_2) := \sum_{B_1 \subset V_1} \sum_{B_2 \subset V_2} \delta_{|B_1|,|B_2|} \frac{1}{Z} k\left(\vartheta_{B_1}, \vartheta_{B_2}\right), \tag{5.25}$$

where $Z = \binom{|V_1|}{|B_1|}\binom{|V_2|}{|B_2|}$, δ is the Kronecker delta, and k is a freely specifiable kernel (called base kernel) from $\mathbb{R} \times \mathbb{R}$ to $\mathbb{R}$.

The SVM-ϑ kernel is defined as (5.25), but it uses an approximation for the ϑ numbers. Consider a graph $\mathcal{G}$ with n vertices and adjacency matrix A, and let $\rho \geq -\lambda$, where λ is the minimum eigenvalue of A. The matrix

$$\kappa := \frac{1}{\rho} A + I, \tag{5.26}$$

is positive semi-definite. Define the maximization problem:

$$\max_{\alpha_i \geq 0} \ 2\sum_{i=1}^{n} \alpha_i - \sum_{i,j=1}^{n} \alpha_i \alpha_j \kappa_{ij}. \tag{5.27}$$

If $\{\alpha_i^*\}$ are the maximizers of (5.27), then it can be proven that on certain families of graphs the quantity $\sum_i \alpha_i^*$ is with high probability a constant factor approximation to $\vartheta(\mathcal{G})$:

$$\vartheta(\mathcal{G}) \leq \sum_{i=1}^{n} \alpha_i^* \leq \gamma \vartheta(\mathcal{G}), \tag{5.28}$$

for some γ. The SVM-ϑ kernel then replaces the ϑ_B numbers on subgraphs with:

$$\vartheta_B(\mathcal{G}) \to \sum_{j \in B} \alpha_j^*. \tag{5.29}$$

The SVM-ϑ kernel requires a choice of base kernel $k : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. We choose a translation invariant universal kernel [MXZ06] $k(x, y) = (\beta + ||x - y||^2)^{-\alpha}$, where α and β are two trainable hyperparameters.

Graphlet Sampling kernel

This kernel is designed to capture the local structural patterns, known as graphlets, within a graph. Graphlets are small connected subgraphs that can be used as building blocks to represent the structural properties of a larger graph. The Graphlet Sampling kernel measures the similarity between two graphs by counting the occurrences of different graphlets in both graphs and comparing their frequencies.

Formally, let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $H = (\mathcal{V}_H, \mathcal{E}_H)$ be two graphs. We say that $\mathcal{H}$ is a subgraph of $\mathcal{G}$ if there exists an injective map $\alpha : \mathcal{V}_{\mathcal{H}} \to \mathcal{V}$ such that $(u, v) \in \mathcal{E}_H \iff (\alpha(u), \alpha(v)) \in \mathcal{E}$. In general it might be possible to map $\mathcal{H}$ into $\mathcal{G}$ in several different ways, *i.e.* the mapping α, if it exists, is not necessarily unique.

Given two graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, the idea behind the Graphlet kernel is to pick an integer $k < \min\{|\mathcal{V}_1|, |\mathcal{V}_2|\}$, enumerate all possible graphs of size k and find the number of ways they can be mapped to $\mathcal{G}_1$ and $\mathcal{G}_2$. Denote by $f^{(k)}_{\mathcal{G}_i}$ the vector where each entry counts the way a specific graph of size k can be mapped as a subgraph of $\mathcal{G}_i$. A kernel can then be defined as the dot product $f^{(k)}_{\mathcal{G}_1} \cdot f^{(k)}_{\mathcal{G}_2}$ between the two vectors.

The complexity of computing such a kernel scales as $O(n^k)$, as there are $\binom{n}{k}$ size-k subgraphs in a graph of size n. For this reason it is preferable to resort to sampling rather than complete enumeration [SVP+09]. Given a choice of integer N, graphs $g_1, \ldots, g_N$ of size between 3 and k are randomly sampled. The number of ways each g_i can be mapped as a subgraph of $\mathcal{G}_j$ is computed and stored in a vector $f_{\mathcal{G}_j}$, and the Graphlet Sampling kernel is defined as the dot product:

$$K_{\mathrm{GS}}(\mathcal{G}_1, \mathcal{G}_2) := f_{\mathcal{G}_1} \cdot f_{\mathcal{G}_2}. \tag{5.30}$$

To account for the different size of $\mathcal{G}_1$ and $\mathcal{G}_2$, each vector can be normalized by the total number of its subgraphs.

Random Walk kernel

The Random Walk kernel is one of the oldest and most studied graph kernels [GFW03]. Given two graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$, the idea is to measure the probability of simultaneous random walks of a certain length between two vertices in $\mathcal{G}_1$ and $\mathcal{G}_2$.

Simultaneous random walks can be conveniently encoded in powers of the adjacency matrix on the product graph. The product graph $\mathcal{G}_1 \times \mathcal{G}_2 = \mathcal{G}_\times = (\mathcal{V}_\times, \mathcal{E}_\times)$ is defined as follows:

$$\begin{aligned}
\mathcal{V}_\times &:= \{(u_i, u_r) \mid u_i \in \mathcal{V}_1,\ u_r \in \mathcal{V}_2\}, && (5.31)\\
\mathcal{E}_\times &:= \{((u_i, u_r), (v_j, v_s)) \mid (u_i, v_j) \in \mathcal{E}_1, \\
&\qquad\qquad (u_r, v_s) \in \mathcal{E}_2\}. && (5.32)
\end{aligned}$$

In other words, an edge in the product graph indicates that an edge exists between the endpoints in both $\mathcal{G}_1$ and $\mathcal{G}_2$. If $A_\times$ is the adjacency matrix of the product graph, then the entries of

$A_\times^k$ indicate the probability of a simultaneous random walk of length k between two vertices $u_i, v_j \in \mathcal{V}_1$ and $u_r, v_s \in \mathcal{V}_2$.

If $p, q \in \mathbb{R}^{|\mathcal{V}_\times|}$ are vectors representing the probability distribution of respectively starting or stopping the walk at a certain node of $\mathcal{V}_\times$, the first idea for a kernel would be to compute the sum $\sum_k q^{\mathrm{T}} A_\times^k p$, which however may fail to converge. A simple modification to make the sum convergent is to choose an appropriate length-dependent weight $\mu(k)$:

$$K(\mathcal{G}_1, \mathcal{G}_2) := \sum_{k=0}^{\infty} \mu(k)\ q^{\mathrm{T}} A_\times^k p. \tag{5.33}$$

The Geometric Random Walk kernel is obtained by choosing the weights to be the coefficients of a geometric series $\mu(k) = \lambda^k$, and p, q to be uniform. If λ is tuned in such a way as to make the series convergent, the kernel reads:

$$K_{\mathrm{RW}}(\mathcal{G}_1, \mathcal{G}_2) := \sum_{k=0}^{\infty} \lambda^k\ e^{\mathrm{T}} A_\times^k e = e^{\mathrm{T}} \left(I - \lambda A_\times\right)^{-1} e, \tag{5.34}$$

where e denote vectors with all the entries equal to 1.

The cost of matrix inversion scales as the cube of the matrix size. If $|\mathcal{V}_1| = |\mathcal{V}_2| = n$, then the cost of the algorithm scales as $O(n^6)$, as it involves the inversion of an adjacency matrix of size $n^2 \times n^2$. Several methods are proposed in [VSKB10] to make the computation faster. The Spectral Decomposition method in particular allows to reduce the complexity for unlabeled graphs to $O(n^3)$. Essentially, one exploits the fact that the adjacency matrix of the product graph can be decomposed in the tensor product of the individual adjacency matrices:

$$A_\times = A_1 \otimes A_2, \tag{5.35}$$

which allows to diagonalize each $n \times n$ adjacency matrix in $O(n^3)$ time and perform the inversion only on the diagonal components.

Shortest Path kernel

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, an edge path between two vertices $u, v \in \mathcal{V}$ is a sequence of edges $(e_1, \ldots, e_n)$ such that $u \in e_1$, $v \in e_n$, e_i and e_{i+1} are contiguous (*i.e.* they have one of the endpoints in common) and $e_i \neq e_j$ for $i \neq j$. Computing the shortest edge path between any two nodes of a graph can be done in polynomial time with the Dijkstra [Dij59] or Floyd-Warshall [Flo62] algorithms, which makes it a viable feature to be probed by a graph kernel.

The first step of the Shortest Path kernel is to transform the graphs into shortest path graphs. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the shortest path graph $\mathcal{G}^S = (\mathcal{V}^S, \mathcal{E}^S)$ associated to $\mathcal{G}$ is defined as:

$$\mathcal{V}^S = \mathcal{V}, \tag{5.36}$$

$$\begin{aligned}\mathcal{E}^S = \{(u, v) \mid \exists \text{ an edge path } (e_1, \ldots, e_n) \\ \text{between } u \text{ and } v \text{ in } \mathcal{G}\}.\end{aligned} \tag{5.37}$$

In addition, to each edge $e \in \mathcal{E}^S$ a label $l(e)$ is assigned given by the length of the shortest path in $\mathcal{G}$ between its endpoints. The Shortest Path kernel is then defined as:

$$K_{\mathrm{SP}}(\mathcal{G}_1, \mathcal{G}_2) := \sum_{e \in \mathcal{E}_1^S} \sum_{p \in \mathcal{E}_2^S} k(e, p), \tag{5.38}$$

with k being a kernel between edge paths such as the Brownian bridge kernel:

$$k(e, p) := \max\{0,\ c - |l(e) - l(p)|\}, \tag{5.39}$$

for a choice of c.

To conclude, we presented several classical graph kernels. As we mentioned previously, although graphs can be very versatile to represent data they have the burden of being harder to analyse. For this reason, different graph kernels have been introduced to capture different aspects of graph structures. In the following section, we will detail our quantum evolution kernel. A main interest is to understand if it can capture graph features that are not accessible to classical graph kernels.

5.3.3 The quantum evolution kernel (QEK)

We now present the quantum evolution kernel (QEK), using the dynamics of an interacting quantum system as a tool to characterise graphs. As we have seen in the previous sections, the idea is to encode the adjacency matrix of a graph $\mathcal{G}$ in the interaction term of a quantum Hamiltonian. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, one can take for example an Ising Hamiltonian of the form $\hat{\mathcal{H}}_{\mathcal{G}} = \hat{\mathcal{H}}_I = \sum_{(i,j)\in\mathcal{E}} \hat{\sigma}_i^z \hat{\sigma}_j^z$, or the XY Hamiltonian $\hat{\mathcal{H}}_{\mathcal{G}} = \hat{\mathcal{H}}_{XY} = \sum_{(i,j)\in\mathcal{E}} (\hat{\sigma}_i^+ \hat{\sigma}_j^- + h.c.)$. Those two Hamiltonians are analysed here because they are ubiquitous spin models that can be easily implemented on a neutral-atom processor. Depending on the problem at hand and the features of the graph one is trying to take into account, different Hamiltonians could be used.

The most general time-evolution of the system under a parameterised Hamiltonian $\hat{\hat{\mathcal{H}}}(t)$ is given by:

$$|\psi_f\rangle = \mathcal{T}\exp\left[-\mathrm{i}\int_0^{t_f} dt\, \hat{\mathcal{H}}(t)\right] |\psi_0\rangle\,, \tag{5.40}$$

where $\mathcal{T}$ is the time-ordering operator and $|\psi_0\rangle$ is the initial state. Once the system has been prepared in the final state $|\psi_f\rangle$, an observable $\hat{\mathcal{O}}$ is measured and we can build a histogram of the obtained values. In other words, we reconstruct the components of $|\psi_f\rangle$ in the eigenbasis of the operator. Let us note $\{\lambda_1, \dots \lambda_K\}$, the eigenvalues of $\hat{\mathcal{O}}$ (*i.e.* the possible outcomes of the measure), and $\{|o_1\rangle, \dots |o_K\rangle\}$ the corresponding eigenstates. The normalised histogram of measured values approaches in the large M limit the following probability distribution

$$\mathcal{P}_{\mathcal{G}}^{\hat{\mathcal{O}}}(\Lambda) = (p_1, \dots p_K),\ \text{where } p_k = |\langle o_k|\psi_f\rangle|^2\,. \tag{5.41}$$

Note that if some eigenvalues are degenerate, one would get instead

$$p_k = \sum_i \delta(\lambda_i - \lambda_k) \left|\langle o_i|\psi_f\rangle\right|^2 , \tag{5.42}$$

where p_k is restricted to the $\tilde{K} < K$ distinct eigenvalues of $\hat{\mathcal{O}}$.

In practice, if K is large, one would resort to binning the values of λ_i by defining a set of $K' < K$ intervals $\{I_k = [\tilde{\lambda}_k, \tilde{\lambda}_{k+1}]\}_{k=1\ldots,K'}$, with $\tilde{\lambda}_1 \leq \min_k \lambda_k$ and $\tilde{\lambda}_{K+1'} \geq \max_k \lambda_k$, such that $\mathcal{P}_{\mathcal{G}}^{\hat{\mathcal{O}}}(\Lambda) = (\tilde{p}_1, \ldots \tilde{p}_{K'})$, where

$$\tilde{p}_k = \frac{|\{m_i | m_i \in I_k\}|}{M} \underset{M\to\infty}{\equiv} \sum_{i|\lambda_i \in I_k} \left|\langle o_i|\psi_f\rangle\right|^2 . \tag{5.43}$$

Finally, once we have associated a probability distribution to $\mathcal{G}$ and $\mathcal{G}'$, we can naturally define a graph kernel by computing the distances between the probability distributions $\mathcal{P}$ and $\mathcal{P}'$. There are many choices of distances between probability distributions. We will here use the Jensen-Shannon divergence that we used in section 5.2. We recall that given two probability distributions $\mathcal{P}$ and $\mathcal{P}'$, the Jensen-Shannon divergence is defined as

$$JS(\mathcal{P}, \mathcal{P}') = H\left(\frac{\mathcal{P} + \mathcal{P}'}{2}\right) - \frac{H(\mathcal{P}) + H(\mathcal{P}')}{2}, \tag{5.44}$$

where $H(\mathcal{P}) = -\sum_k p_k \log p_k$ is the Shannon entropy of $\mathcal{P}$. $JS(\mathcal{P}, \mathcal{P}')$ takes values in $[0, \log 2]$. In particular $JS(\mathcal{P}, \mathcal{P}) = 0$, and $JS(\mathcal{P}, \mathcal{P}') = \log 2$ is maximal if $\mathcal{P}$ and $\mathcal{P}'$ have disjoint supports.

For two graphs $\mathcal{G}$ and $\mathcal{G}'$, and their respective probability distributions $\mathcal{P}$ and $\mathcal{P}'$ we define the graph kernel as

$$\mathcal{K}_\mu(\mathcal{G}, \mathcal{G}') = \exp\left[-\mu\, JS(\mathcal{P}, \mathcal{P}')\right] \in \left[2^{-\mu}, 1\right] . \tag{5.45}$$

The kernel is then positive by construction. Throughout this chapter we set $\mu = 1$, but it might be helpful to adjust this value to improve the results. The parameter Λ is determined through training on a dataset containing graphs whose class is known.

All-in-all, the approach we propose consists in associating each graph $\mathcal{G}$ with a probability distribution $\mathcal{P}_{\mathcal{G}}$ obtained by the measurement of an observable on a quantum system whose dynamics is driven by the topology of $\mathcal{G}$. Then, the QEK is computed using the Jensen-Shannon divergence.

Depending on the type of Hamiltonian evolution or observable, QEK can be related to other kernels, a subject detailed in Ref [HTDH21]. In the following section, we describe a machine learning algorithm that uses a kernel to classify datapoints. Specifically, one can use QEK inside of SVM to do classification tasks on graphs.

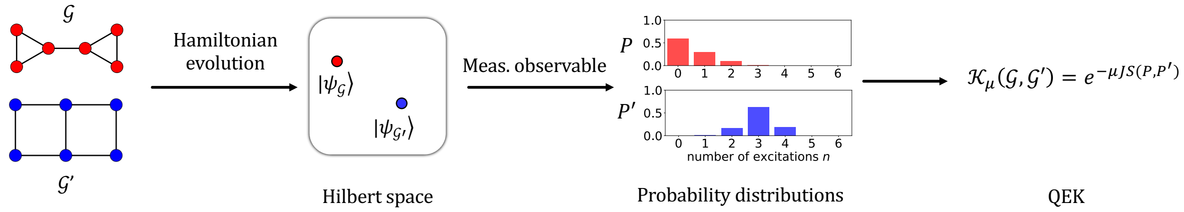


Figure 5.5: **Building blocks of the Quantum Evolution Kernel (QEK).** For each graph, a quantum system is evolved under a Hamiltonian that encodes $\mathcal{G}$ (resp. $\mathcal{G}'$) to a specific state of the Hilbert space. Then, an observable is measured and is converted into a probability distribution $\mathcal{P}$ (resp. $\mathcal{P}'$). Finally, QEK consists in exponentiating the Jensen-Shannon divergence of the two distributions $\mathcal{P}, \mathcal{P}'$.

The Support Vector Machine (SVM) algorithm

The SVM algorithm aims at splitting a dataset into two classes by finding the best hyperplane that separates the data points in the feature space, in which the coordinates of each data point (here each graph) is determined according to the kernel K.

Suppose we have access to a training graph dataset $\{\mathcal{G}_i\}_{i=1\dots M}$ with a set of labels $\mathbf{y} = \{y_i\}_{i=1\dots M}$ where $y_i = \pm 1$ depends on which class the graph $\mathcal{G}_i$ belongs to. Mathematically, the dual formulation of the SVM problem consists in finding $\tilde{\boldsymbol{\alpha}} \in \mathcal{A}_C(\mathbf{y}) = \left\{\boldsymbol{\alpha} \in [0, C]^M \middle| \boldsymbol{\alpha}^T \mathbf{y} = 0\right\}$ such that

$$\frac{1}{2}\tilde{\boldsymbol{\alpha}}^T Q \tilde{\boldsymbol{\alpha}} - \mathbf{e}^T \tilde{\boldsymbol{\alpha}} = \min_{\boldsymbol{\alpha} \in \mathcal{A}_C(\mathbf{y})} \left\{ \frac{1}{2}\boldsymbol{\alpha}^T Q \boldsymbol{\alpha} - \mathbf{e}^T \boldsymbol{\alpha} \right\}, \tag{5.46}$$

where $\mathbf{e}$ is the vector of all ones, Q is a $M \times M$ matrix such that $Q_{ij} = y_i y_j K(\mathcal{G}_i, \mathcal{G}_j)$, and $C > 0$ is an adjustable penalty hyperparameter [2]. The data points for which $\tilde{\alpha}_i > 0$ are called support vectors (SV). Once the α_i are trained, the class of a new graph $\mathcal{G}$ is predicted by the decision function, given by:

$$y(\mathcal{G}) = \operatorname{sgn}\left\{ \sum_{i \in SV} y_i \tilde{\alpha}_i K(\mathcal{G}, \mathcal{G}_i) \right\}. \tag{5.47}$$

Once the kernel is trained, the prediction relies only on estimate the kernel between the unseen graph and the support-vector graphs. The choice of the kernel function determines the nature of the decision boundary that SVM can learn: in the next section, we benchmark SVM when used with different underlying kernels, namely the classical graph kernels and QEK.

5.4 Experimental results on a toxicity classification task

In this section, we tackle a binary classification task on a dataset of chemical compounds called the Predictive Toxicity Challenge on Female Mice (PTC-FM) [HKKS01, SNL+18]. The objective

---

[2] Setting C to a large value increases the range of possible values of α and therefore the flexibility of the model. On the other hand, it also increases the training time and overfitting risk.

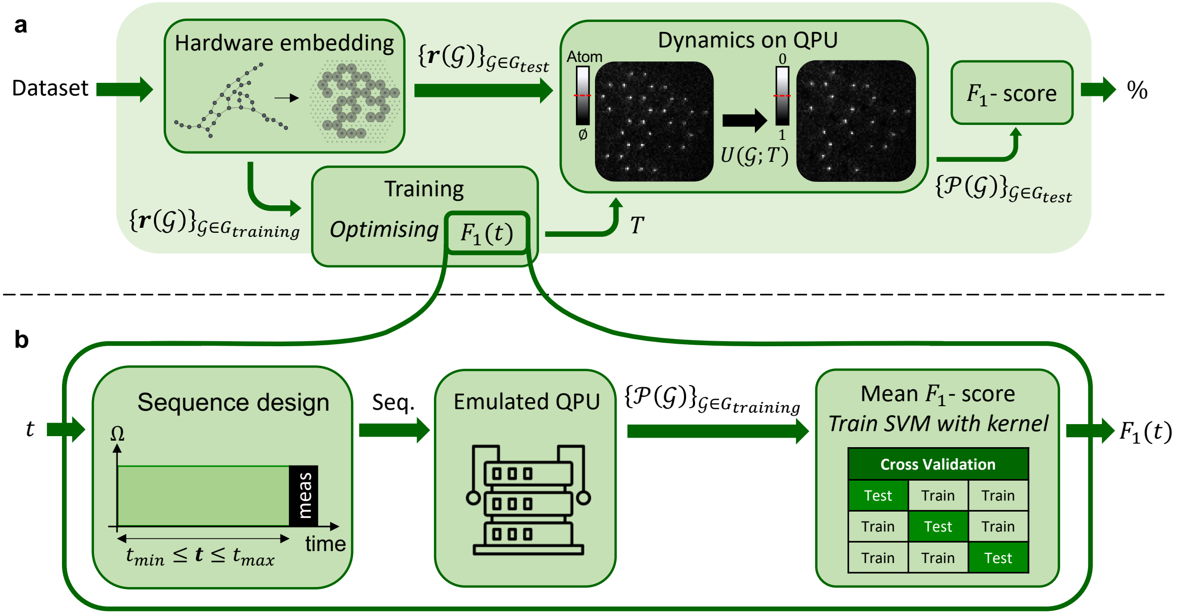


Figure 5.6: **Overview of QEK implementation.** A dataset of graphs $\mathcal{G}$ is first mapped onto atomic registers $\mathbf{r}(\mathcal{G})$ implementable on the QPU, and separated between a training set $\mathcal{G}_{training}$ and a test set $\mathcal{G}_{test}$(**a**). We use the training set to determine numerically the optimal pulse sequence to be applied on the hardware using a grid search algorithm for optimising $F_1(t)$ (see **b**). This training phase outputs the optimal parameter T used to design the laser-pulse sequence applied experimentally on each register of the test set. The resulting dynamics performed on the QPU generates $U(\mathcal{G};T)$, driving the system from $|0\rangle^{\otimes\mathcal{G}}$ to $|\psi_{\mathcal{G}}\rangle$. F_1 is then derived from the measured probability distributions $\{\mathcal{P}(\mathcal{G})\}_{\mathcal{G}\in G_{test}}$. **b.** The optimisation of the score function F_1 during the training includes several steps. The input t, taken from the parameter space $[t_{min}, t_{max}]$ defines a laser sequence with Ω and δ fixed parameters followed by a measurement. The dynamics of the system is emulated and enables us to compute the probability distributions associated to this particular value of t for the whole training part of the dataset. Finally, $F_1(t)$ is obtained by fitting the SVM with the kernel constructed from those probability distributions.

is to accurately predict the toxicity of chemical compounds based on their structural properties. Indeed in many cases poisonous proteins act as enzyme inhibitors, where the geometry of the protein fits to the binding site of an enzyme and perturbs its usual functioning [BUBVO20, SV91]. To realise the classification task on this dataset, we test classical kernels and QEK inside of an SVM algorithm. In the next sections, we will detail the hardware implementation protocol for QEK. The whole process is illustrated in Figure 5.6.

5.4.1 Mapping the dataset on hardware

In the original PTC-FM dataset, the 349 molecules are represented under the form of graphs where each node is labeled by atomic type and each edge is labeled according to its bond type. We first truncate the dataset to small graph sizes in order to be able to train the kernel in reasonable time, and discard larger molecules. For the $M = 286$ remaining graphs of this dataset, we take into account the adjacency matrix of the graphs representing the compounds and discard the nodes and edges labels. Note that the results of our implementation are therefore not directly comparable to kernel results in the literature which take into account edge and node labels as in Ref. [KJM20] for example.

Each node of a graph will be represented by a qubit in the QPU. In the same fashion as for the UD-graphs in combinatorial optimisation schemes, we first need to determine the positions of these atoms in order to implement an interaction term that effectively reflects the graph topology. To this end we use the same mapping and batching technique as presented in Part I of this thesis. For completeness, we just recall the main idea of our protocol. Starting from a Reingold-Fruchterman layout [FR91a], our optimiser minimises the average distance between two connected nodes while maximising the distances between unconnected nodes. Then taking advantage of our ability to tailor the spatial disposition of the tweezers generated by a Spatial Light Modulator (SLM) to fit the optimised layout, we can replicate the graph in the hardware.

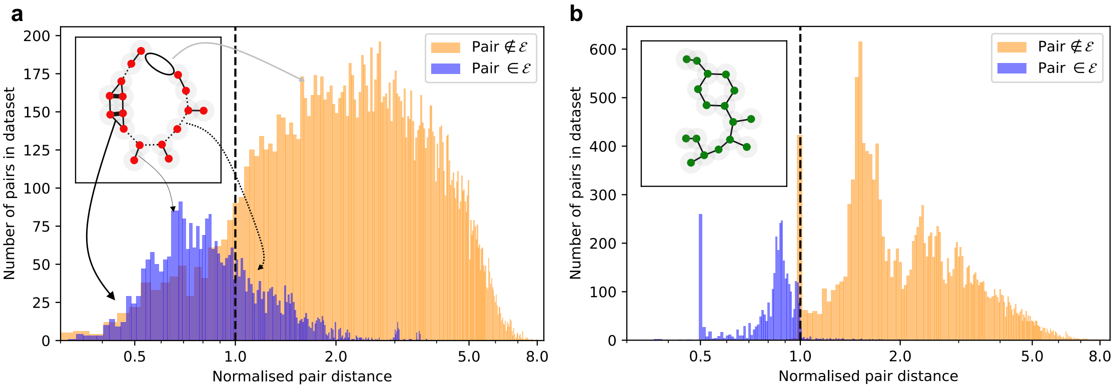


Figure 5.7: **Optimising positions for embedding graph topology in atomic registers.** Histograms of normalised pairwise distances between atoms in the 286 graphs of the truncated dataset when performing the embedding with **a.** only a Fruchterman-Reingold layout or **b.** when adding a local optimisation step afterwards. For a given graph (insets), two atoms forming a pair $\in \mathcal{E}$ (blue) can be close enough to form a bond via interaction (plain) or too far, creating a missing bond (dotted). Likewise, two atoms forming a pair $\notin \mathcal{E}$ can be placed too close and form a fake edge (thick line).

We assess the benefit of this approach by comparing the distributions of distance of pairs $\in \mathcal{E}$ and pairs $\notin \mathcal{E}$ before and after the optimisation (Figure 5.7). While some defects such as fake

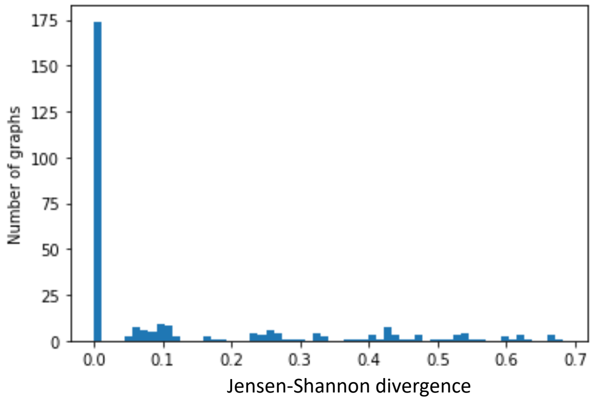


Figure 5.8: **Distribution of Jensen-Shannon divergences between measured and expected histograms** for a pulse of duration $T = 0.66\mu s$. We see that a majority of the graphs are exactly mapped, while others unfortunately are further away than expected.

or missing bonds frequently appear in the pre-optimisation embedding, the optimised positions are constrained in such a way that a clear cut is visible between the two distributions. One can also characterise the effect of these defects by analysing their impact on the measurement histograms. For each graph, we first compute the histogram that would have been obtained with a perfect embedding. We then compute the Jensen-Shannon divergence between this histogram and the one measured in the QPU (Figure 5.8).

Once the mapping is deemed good enough, we apply the batching method to the dataset and successfully map the entire dataset of 286 graphs into only 6 SLM patterns. For example, we batch 66 graphs together onto the 71-trap SLM pattern presented in Figure 5.9. On average, the 6 SLM patterns use 70 traps each to encode 48 graphs each.

5.4.2 Model training

To test the performance of our implementation, we perform a standard procedure called cross-validation. Cross-validation consists in dividing the dataset in 5 equal parts called ‘splits’, and using each split for testing while the rest of the dataset is used for training. During the training phase, we construct for each pulse duration t the corresponding kernel and train a SVM model with it. We estimate the quality of the classification by using the F1 score defined as

$$F_1 = \frac{t_p}{t_p + (f_p + f_n)/2}, \tag{5.48}$$

where t_p, f_p and f_n are respectively the number of true positives, false positives and false negatives of the predicted distribution. Correctly predicting the toxicity of a compound leads to a better F_1 score. On the other hand, classifying a toxic compound as harmless or a harmless compound as toxic results in a lower score. We then evaluate the F_1-score on the part of the dataset that was left as a test set. We repeat the splitting 10 times, and the cross-validation

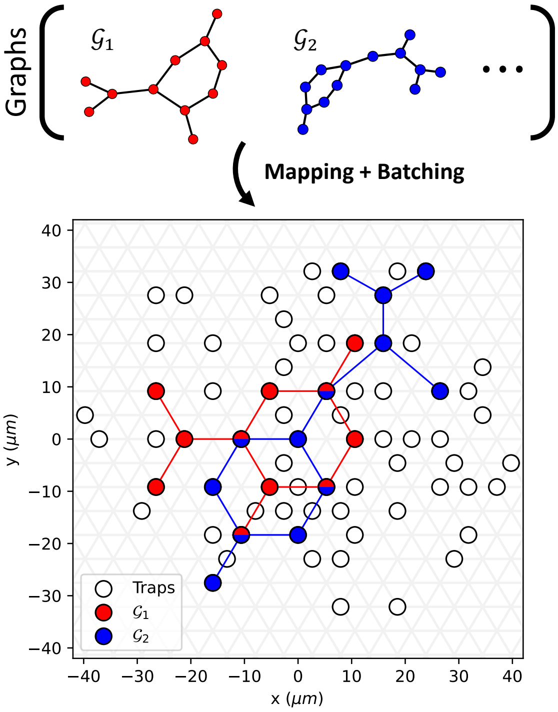


Figure 5.9: **Batching graphs to the same SLM pattern.** A family of 66 graphs, ranging in sizes from 4 to 19 nodes, is mapped and batched to the same SLM pattern (white dots) over a triangular grid with spacing 5.6 μm. The traps used when implementing $\mathcal{G}_1$ ($\mathcal{G}_2$) are colored in red (blue). The bi-colored traps are those used for both graphs.

score is defined as the average of the F_1-score of each split (50 splits in total). We perform a grid search on the penalty hyperparameter C of the SVM on the range $[10^{-3}, 10^3]$ such that the final score of a given pulse is the best cross-validation score among the tested values.

Including graphs with sizes $|\mathcal{V}| \leq 20$, we numerically compute the score for a nearest-neighbour distance $r_{NN} = 5.6\,\mu$m and a resonant constant pulse with fixed $\Omega/2\pi = 1$ MHz. We vary its duration between $t_{min} = 0.1\,\mu$s and $t_{max} = 2.5\,\mu$s to select the duration that exhibits the maximum F_1-score (namely $T = 0.66\,\mu s$) before implementing the pulse directly on the QPU.

5.4.3 Classification results

After a training of our model, we experimentally obtain an F_1-score of $60.4 \pm 5.1\%$. We show in Table 5.1 the performances of QEK and all the classical kernels we exposed previously, namely the Graphlet Sampling (GS), the Random Walk (RW), the Shortest Path (SP) and the SVM-ϑ kernels. The obtained scores range from $49.8 \pm 6.0\%$ up to $58.2 \pm 5.5\%$, showing that QEK is on-par with standard classical kernels on this dataset.

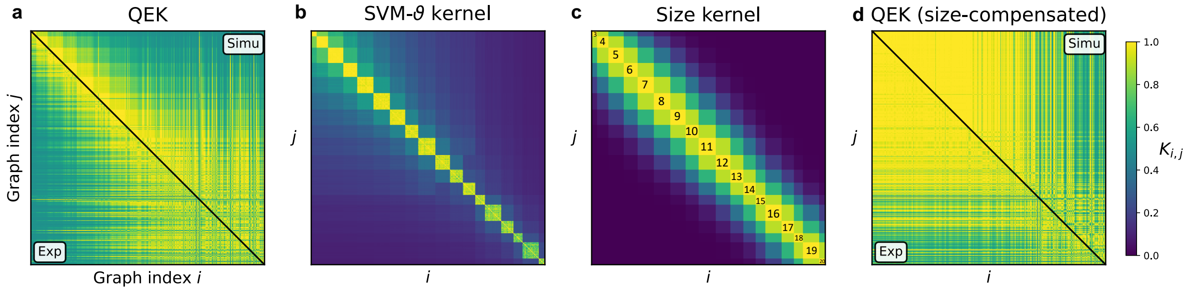


Figure 5.10: **Kernel matrices obtained numerically or experimentally.** Each kernel is represented by a $M \times M$ matrix where $K_{i,j} = K(\mathcal{G}_i, \mathcal{G}_j)$. The graph indices are sorted by increasing size. A separation (black line) is drawn between numerically simulated (top right) and experimentally measured (bottom left) QEK matrices. **a.** QEK kernel obtained using directly the raw distributions $\mathcal{P}_i$ and $\mathcal{P}_j$. **b.** Kernel obtained via SVM-ϑ method. **c.** Size kernel obtained with $K^{\text{size}}(\mathcal{G}_i, \mathcal{G}_j) = \exp(-\gamma(|\mathcal{G}_i| - |\mathcal{G}_j|)^2)$ with $\gamma = 0.1$. **d.** QEK kernel obtained using modified distributions $\tilde{\mathcal{P}}_i$ and $\tilde{\mathcal{P}}_j$, where graphs of smaller sizes are convoluted with binomial distributions when compared to larger graphs.

Kernel	F_1-score (%)
QEK	60.4 ± 5.1
QEK (size-compensated)	45.1 ± 3.7
SVM-ϑ	58.2 ± 5.5
Size	56.7 ± 5.6
Graphlet Sampling	56.9 ± 5.0
Random Walk	55.1 ± 6.9
Shortest Path	49.8 ± 6.0

Table 5.1: F_1-score reached experimentally on the PTC-FM dataset by QEK (± std. on the splits). In addition, the scores reached numerically by the classical kernels SVM$-\vartheta$, Size, Graphlet Sampling, Random Walk and Shortest-Path. The values reported are the average over a 5-fold cross-validation repeated 10 times.

Because the dataset is of acceptable size, we can plot directly the Gram matrix associated to each kernel (Figure 5.10). Using the same noise model as for the expressivity test (Section 5.2), we find adequate agreement between the numerical noisy simulation and experimental results.

Size *is* important

An interesting feature of both QEK and SVM-ϑ kernel matrices is the emergence of size-related diagonal blocks (Figure 5.10**b**). This means that the models capture the size of the graphs as an important feature for classification. Examining more closely the dataset, we found out that indeed our dataset was significantly size imbalanced, as illustrated in Figure 5.11. Since the graph size seems to be a relevant feature for this particular dataset, we reach an F_1-score

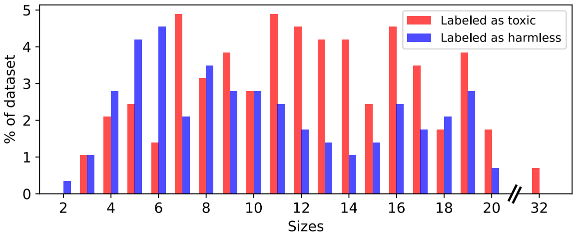


Figure 5.11: **Dataset imbalance**. The PTC-FM dataset exhibits a strong size imbalance. For small number of nodes ($\lesssim 10$) more graphs are labeled as harmless (blue) while it is the opposite for larger graphs, more prone to be labeled as toxic (red).

of $56.7 \pm 5.6\%$ with the simplest size kernel. The corresponding kernel matrix is displayed in Figure 5.10**c** and exhibits a block-diagonal shape with a Gaussian tail.

It is interesting to note that the quantum model was able to identify size as a relevant parameter for this dataset, leading to classification results which are on par with the best classical kernels.

Getting rid of the size feature

In this part, we want to understand if our quantum kernel can capture other features rather than the size. In order to do so, we modify the QEK procedure to incorporate a convolution operation. Mathematically, let us consider two graphs $\mathcal{G}_i$ and $\mathcal{G}_j$ of respective sizes N_i and $N_j = N_i + \Delta N > N_i$. We note their respective observable distributions $\mathcal{P}_i$ and $\mathcal{P}_j$. From $\mathcal{P}_i$ we construct $\tilde{\mathcal{P}}_i = \mathcal{P}_i \star b_{\Delta N}^{(i/j)}$ which is the convolution of $\mathcal{P}_i$ and a binomial distribution:

$$b_{\Delta N}^{(p)}(n) = \binom{\Delta N}{n} p^n (1-p)^{\Delta N - n}. \tag{5.49}$$

A physical intuition is to see $\tilde{\mathcal{P}}_i$ as the distribution one would get by adding ΔN non-interacting nodes to the graph. When embedded in the atomic array, each of these isolated atoms undergoes Rabi oscillations induced by the applied pulse sequence. They are therefore measured either in $|0\rangle$ with probability p or in $|1\rangle$ with probability $1-p$, where $p = \sin^2(\pi\Omega T)$ (≈ 0.768 here). We finally define the modified graph kernel as

$$K_{conv}(\mathcal{G}_i, \mathcal{G}_j) = \exp\left[-JS(\tilde{\mathcal{P}}_i, \mathcal{P}_j)\right]. \tag{5.50}$$

Using this procedure on the data obtained experimentally, we obtain the kernel matrix shown in Figure 5.10**d** with a corresponding $F1$-score of $45.1 \pm 3.7\%$. If this size-compensated version of QEK had been implemented without interaction between atoms, its score would be 42%, which is the lowest score reachable by any kernel. We therefore see that this version of QEK cannot capture useful features beyond the graph size, meaning that the presence of interactions by itself is not sufficient to produce an interesting kernel for the task at hand. Nonetheless, although the size-compensated QEK does not give results that are comparable with classical kernels, it is still interesting to study the geometric structure that is induced by this kernel. In the next part we will show that the geometry induced by this method is hardly reproducible by a classical kernel. In other words, QEK creates an associated feature space with highly complex geometry that is unfortunately not necessary for the classification task at hand.

5.5 Geometric test with respect to classical kernels and maximum quantum-classical separation

In order to obtain an advantage over classical approaches it is not sufficient to implement a quantum feature map based on quantum dynamics that are hard to simulate classically. As shown in [HBM+21a], classical ML algorithms can in certain instances learn efficiently from intractable quantum evolution if they are allowed to be trained on data. The authors consequently propose another metric between kernels in the form of an asymmetric metric function called the geometric difference g_{12}. It compares two kernels K_1 and K_2 in the following way:

$$g_{12} = \sqrt{||\sqrt{K_2}\,(K_1)^{-1}\,\sqrt{K_2}||_\infty}, \tag{5.51}$$

where $||.||_\infty$ is the spectral norm. Intuitively, g_{12} measures the difference between how kernels K_1 and K_2 perceive the relation between data. Precisely, it characterises the disparity regarding how each of them maps data points to their respective feature spaces. In our case, we take K_1 to be the size-compensated QEK K_{conv}, and K_2 is selected from a set of classical kernels. If the geometric difference is small, it means that there exists no underlying function mapping the data to the targets for which K_{conv} outperforms the classical kernel. On the other hand, a high geometric difference between a quantum and a classical kernel guarantees that there exists such a function for which the quantum model outperforms the classical one. Estimating the geometric difference is therefore a sanity check stating that the encoding of data to the feature space through the Quantum Evolution Kernel could not be closely replicated by a classical model.

We compute the geometric difference between QEK and various classical kernels over the PTC-FM dataset and report the results in Table 5.2. The threshold for a high geometric difference is typically taken to be $\sqrt{M}$, where M is the size of the dataset. Here, the obtained g_{12} is always far beyond $\sqrt{M} \sim 10^1$, indicating that the embedding of data through our quantum-enhanced kernel is not trivial and cannot be replicated by a classical machine learning algorithm.

To summarize, while the F_1-score on PTC-FM is rather similar using quantum or classical models, we see nonetheless that the geometry created by our quantum model is non-trivial. A possible interpretation of the non-superiority of quantum approaches on PTC-FM would be that the relationship between the data and the targets is not better captured by our quantum model, although its feature space is not reproducible by classical means. To further confirm this understanding, we find a function that increases and even maximizes the utility of our rich quantum feature space. We build such a function by artificially relabelling the targets according to the following procedure.

Relabelling the targets

If M is the size of the dataset, a value of g_{CQ} of order $\sqrt{M}$ or greater indicates that the geometry of the feature space induced by the quantum kernel is rich enough to be hard to learn classically, and the quantum kernel can potentially perform better than classical kernels. In that case, it is possible to artificially relabel the dataset in order to maximally separate the kernels' performance. Such a relabelling process is a constructive proof of the existence of a certain dataset on which one kernel performs much better than the other. If v is the eigenvector of $\sqrt{K_2}\,(K_1)^{-1}\sqrt{K_2}$ corresponding to the eigenvalue g_{12}^2, the vector of new labels is given by $y_{\text{new}} = \sqrt{K_2}v$. When dealing with a finite amount of training data, equation (5.51) should be regularised in order to stabilize the inversion of K_1. The regularised expression reads:

$$g_{12}(\lambda) = \sqrt{||\sqrt{K_2}\sqrt{K_1}\,(K_1 + \lambda I)^{-2}\,\sqrt{K_1}\sqrt{K_2}||_\infty}, \tag{5.52}$$

where λ is the regularisation parameter. The geometric difference $g_{12}(\lambda)$ has a plateau for small λ, when the regularisation parameter becomes smaller than the smallest eigenvalue of K_1, and decreases for increasing λ. The effect of λ is to introduce a certain amount of training error. The training error can be upper bounded by a quantity proportional to:

$$g_{\text{tra}}(\lambda)^2 = \lambda^2||\sqrt{K_2}\,(K_1 + \lambda I)^{-2}\,\sqrt{K_2}||_\infty. \tag{5.53}$$

Practically, one should look at the regime where g_{12} has not plateaued but the training error is still small enough.

A regularisation should be introduced also in the relabelling procedure. The new labels are taken to be $y_{\text{new}} = \sqrt{K_Q}v$, where v is the eigenvector of the regularised matrix

$$\sqrt{K_Q}\sqrt{K_C}\,(K_C + \lambda I)^{-2}\,\sqrt{K_C}\sqrt{K_Q},$$

corresponding to the eigenvalue $g_{12}(\lambda)^2$.

We observe that QEK, without retraining, retains an F_1-score of around 99% on the relabelled dataset, while the closest classical kernel reaches a score of at most 82% even after retraining it on the new labels. The results are summarised in Table 5.3, where the difference in F_1-score between QEK and the various classical kernels is shown.

In light of the geometric difference assessment and the observed gap of F_1-score between QEK and classical kernels on an artificial function, it remains an open question to generally

Geometric Difference w.r.t. QEK	
SVM-ϑ	10^3
Size	10^5
Graphlet Sampling	10^4
Random Walk	10^5
Shortest Path	10^5

Table 5.2: Order of magnitude of the geometric difference between QEK and various classical kernels.

F_1-score gap (%) w.r.t. QEK (relabelled)	
SVM-ϑ	17.2 ± 4.5
Size	17.8 ± 4.2
Graphlet Sampling	20.1 ± 4.5
Random Walk	17.3 ± 4.3
Shortest Path	18.2 ± 4.4

Table 5.3: Gap in F_1-score between QEK and various classical kernels after relabelling the dataset.

characterize which types of dataset naturally offer a structure that better exploits the geometry offered by our quantum model, without requiring artificial tweaking of the labels. In the following section, we present a synthetic dataset on which QEK is able to outperform classical methods without any relabelling.

5.6 Synthetic dataset

This binary classification dataset is created by sampling weighted random walks on a triangular lattice. In class A, sites belonging to a honeycomb-type sublattice are favoured. They are explored with a weight $p_0 = 1$ while the rest of the triangular lattice sites are explored with a weight $p < 1$. Class B is constructed in a similar fashion, but taking a kagome instead of a honeycomb sublattice. The construction of this artificial dataset is illustrated in Figure 5.12. In the case where $p = 0$, the differences in their local structure make the two classes easily distinguishable. However, with increasing p, their local structure becomes more and more similar, as additional triangular lattice sites are incorporated. When p is large enough, a lot of triangular local substructures are shared by the two classes, rendering them potentially hard to distinguish by classical methods. At $p = 1$, the underlying triangular lattice is explored uniformly, rendering the datasets indistinguishable.

Building on our ability to distinguish between graphs with similar local structure but globally distinct, we apply QEK on this synthetic dataset. We expect our method to be hardly affected

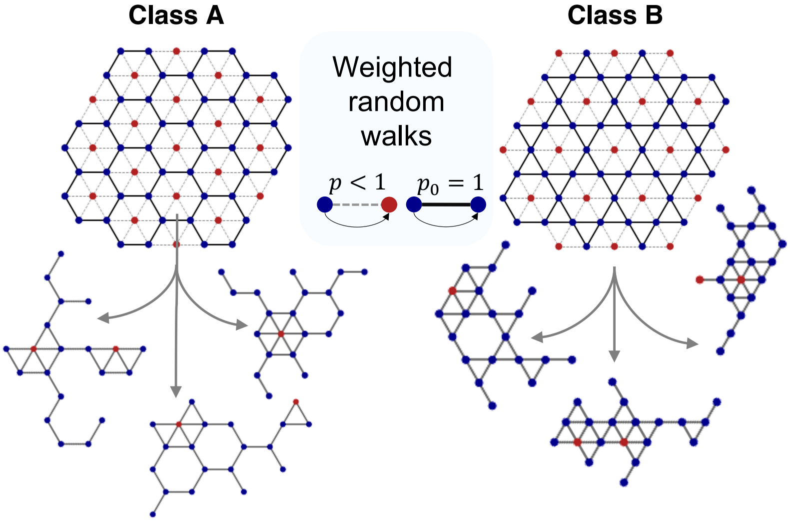


Figure 5.12: **Synthetic dataset.** Graphs in Class A contains honeycomb sites (blue) with inclusions of non-honeycomb sites (red) with probability p. Graphs in Class B contains kagome sites (blue) with inclusions of non-kagome sites (red) with probability p. We show examples of generated graph with the aforementioned process.

by the presence of sparse defects and therefore be able to outperform classical approaches.

We investigate numerically this assumption for several values of p. In each case, we create a balanced dataset of 200 graphs of 20 nodes each with 100 graphs in each class. The graphs are mapped to a triangular lattice with 5 µm spacing. Here, we consider two alternative schemes of pulse sequences. The first one remains almost the same as the experimentally implemented one, *i.e.* a unique resonant pulse of $\Omega/2\pi = 2$ MHz with parameterized duration up to $8\,\mu$s. The second one is an alternate layer scheme with 4 parameters as described in [HTDH21], where we evaluate 500 random values of the parameters and select the best one. The procedure is designed in such way that it would be directly implementable on the hardware, as we did for the PTC-FM dataset. We then compare the F_1-score reached by QEK to those reached by other classical kernels, namely: SVM-ϑ, GS, RW and SP. The results are summarised in Figure 5.13. With decreasing proportion of defects, all methods perform increasingly better, as expected. Overall, regarding the mean F_1-score reached, the two QEK schemes outperform the four other classical kernels tested for all $p \leq 0.5$. Noticeably, at $p = 0.1$ (*resp* $p = 0.2$), the mean gap in F_1-score between the QEK scheme and the best classical scheme is 4.5% (*resp* 7.1%) while the mean gap obtained with the alternate QEK scheme is even larger with 13.7% (*resp* 21%), thus showing that QEK can significantly surpass classical approaches on certain types of datasets. When adding too many defects, *i.e.* $p = 0.5$, our Quantum Evolution Kernel exhibits similar performance to the SVM-ϑ.

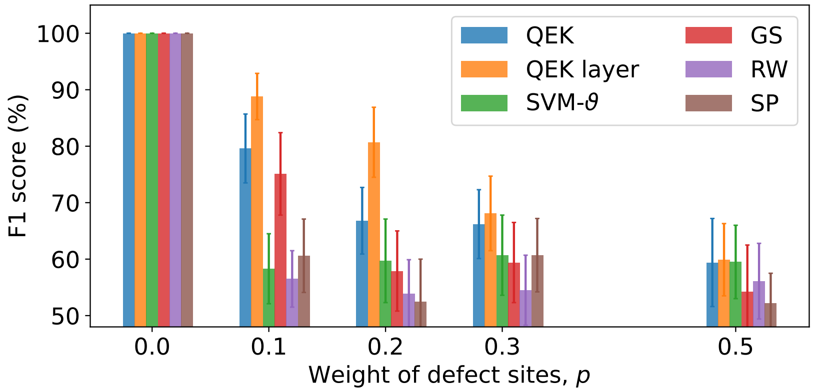


Figure 5.13: F1-score (%) reached on the synthetic dataset for different probabilities p of including non-sublattice sites, by the Quantum Evolution Kernel (the alternate scheme is noted QEK layer) as well as by the best SVM-ϑ, GS, RW and SP kernels. The values reported are the average over a 5-fold cross-validation repeated 10 times. Each kernel reaches an F_1-score of 100% when $p = 0$.

5.7 Conclusion

In this chapter, we reported the implementation of a quantum feature map for graph-structured data on a neutral atom quantum processor with up to 32 qubits. We experimentally showed that this embedding was not only sensitive to local graph properties but was also able to probe more global structures such as cycles. This property offers a promising way to expand the capabilities of standard GNN architectures, which have been shown to have the same expressiveness as the Weisfeiler-Lehman (WL) Isomorphism test in terms of distinguishing non-isomorphic graphs [MRF+19, XHLJ19]. Some properties of quantum-enhanced version of GNNs have been explored in [TFH22] by some of my collegues.

We then used the quantum graph feature map for a toxicity screening procedure on a standard bio-chemistry dataset comprising 286 graphs of sizes ranging from 2 to 32 nodes. This procedure achieved an F_1-score of $60.4 \pm 5.1\%$, on par with the best classical kernels. We intentionally did not include GNNs in the benchmark, as they belong to another distinct family of models. Beyond this pure performance assessment, we showcased the potential advantage of using a quantum feature map through the computation of geometric differences with respect to said classical kernels, which are metrics evaluating the degree of similarity between the kernels' feature spaces. We showed that the Quantum Evolution Kernel captured features that are invisible to the classical kernels we considered. An artificial relabelling of the data enabled us to create a synthetic dataset for which the performances of the Quantum Evolution Kernel could not be matched. We also identified another dataset made of bipartite 2D lattices, for which the quantum procedure exhibited superior performances.

This proof-of-concept illustrates the potential of quantum-enhanced methods for graph machine learning tasks. Our study paves the way for the incorporation of quantum-enhanced algorithms with standard ML solutions, aiming at constructing better tools for graph data analysis and prediction. Further work on more diverse datasets will be required to assess the viability of the approach compared to powerful state-of-the-art GNN architectures [GSR+17, YCL+21, RGD+22, KBH+21]. Additionally, our results showcase the power and versatility of neutral atom QPUs, with their ability to change the register geometry from run to run. Going forward, the implementation of similar methods on non-local graphs could be envisaged by embedding them into three-dimensional registers using our method in Chapter 3 or moving the qubits throughout the course of the computation [BLS+21].

Chapter 6

Classical approximation of variational quantum models

In the previous chapter, we showcased a new quantum kernel (QEK) that can be applied to classification and regression tasks on graph datasets. Interestingly, we have shown that QEK creates a rich feature space that cannot be learned by other classical kernels. It is however quite hard to gain insight on the power of quantum machine learning models when, differently from classical models, they can only be benchmarked on datasets of small sizes and few hundreds of instances.

To overcome the limitations of current quantum devices, researchers have been exploring theoretical properties of NISQ learning models. The theoretical study of variational quantum circuits (VQC) as quantum models mainly focuses on three aspects: expressivity, trainability and generalisation. Expressivity refers to the capabilities of quantum models to represent and approximate a wide range of functions or data distributions. The capacity of a VQC to succeed in finding the correct underlying model relies on the construction of a suited parameterised quantum circuit, also called an *ansatz*. The type of ansatz can depend on the task at hand: the coupled-cluster ansatz [BM07] for example is usually used for quantum chemistry. More simply, the ansatz can be constrained by the hardware capabilities (these are called *hardware-efficient* ansatz) [KMT+17]. Ideally, the ansatz should be expressive enough in order to learn the best model, but should also be trainable. A trainability issue that is proper to quantum models is called the *barren plateau* phenomenon: quantum circuits with a 2-design structure have exponentially vanishing gradients, making them terribly untrainable [MBS+18]. As a matter of fact, a fundamental link has been discovered between ansatz expressibility and trainability, showing that highly expressive ansatz exhibit flatter cost landscapes and are therefore harder to train [HSCC22]. Recent results have also connected the dynamical Lie algebra of an ansatz to the barren plateau phenomenon [LCS$^+$22]. In the past years, researchers have focused on perfecting the structure of the ansatz and have developed the theory of geometric quantum learning models based on symmetries of the problem being solved [LSS$^+$22b, MMGF$^+$23]. Finally, the third important component of QML along expressivity and trainability is the generalisation, which refers to the ultimate goal of a machine learning model of making accurate predictions on unseen data. Interesting and encouraging results can be found in Refs [CGFM$^+$21a, HBM$^+$21b, CHC$^+$22, BPP21].

In this chapter we focus on the first pillar of QML, namely expressivity of quantum models emanating from VQCs. The common intuition behind QML is to leverage the exponential Hilbert space to learn a complex model. In the following sections, we explore how a classical method called random Fourier features (RFF) [RR09] can be used to approximate models emanating from VQCs. Broadly, the idea is to build a tractable model that mimics the un-simulatable quantum model with good approximation. Understanding how and why certain quantum models can be effectively approximated classically is an important step in our comprehension of quantum models in machine learning. In section 6.1, we recall the spectral properties of a family of quantum models. We then show in section 6.2 that this quantum model gives rise to a shift-invariant kernel. We then explore in section 6.3 the RFF strategies used to approximate quantum models, accompanied with theoretical bounds on the quality of approximation. Finally, we run numerical experiments in section 6.4.

This chapter is summarised in the paper accepted at ICLR 2023, a classical machine learning conference (2,500+ attendees)

- [LTD$^+$22] Landman, J., Thabet, S., Dalyac, C., Mhiri, H., Kashefi, E. (2022). *Classically Approximating Variational Quantum Machine Learning with Random Fourier Features.* ICLR 2023

6.1 Variational quantum circuits (VQCs) and spectral properties

Let us consider a standard ML task where a model f is optimised to map data points to their target values. The data used for the training consists of M points $x = (x_1, \dots, x_M)$ in $\mathcal{X} = \mathbb{R}^d$ along with their target values y in $\mathcal{Y} = \mathbb{R}$. Recall from the previous chapter that a quantum model is defined as the family of parametrized functions $f : (\mathcal{X}, \Theta) \longrightarrow \mathcal{Y}$, such that

$$f(x;\theta) = \langle 0|U(x;\theta)^\dagger O U(x;\theta)|0\rangle, \tag{6.1}$$

where $U(x;\theta)$ is a unitary that represents the parametrized quantum circuits, θ represents the trainable parameters from a space Θ, and O is an observable. In the following, we consider a unitary $U(x;\theta)$ that can be decomposed as an alternation of encoding and trainable gates.

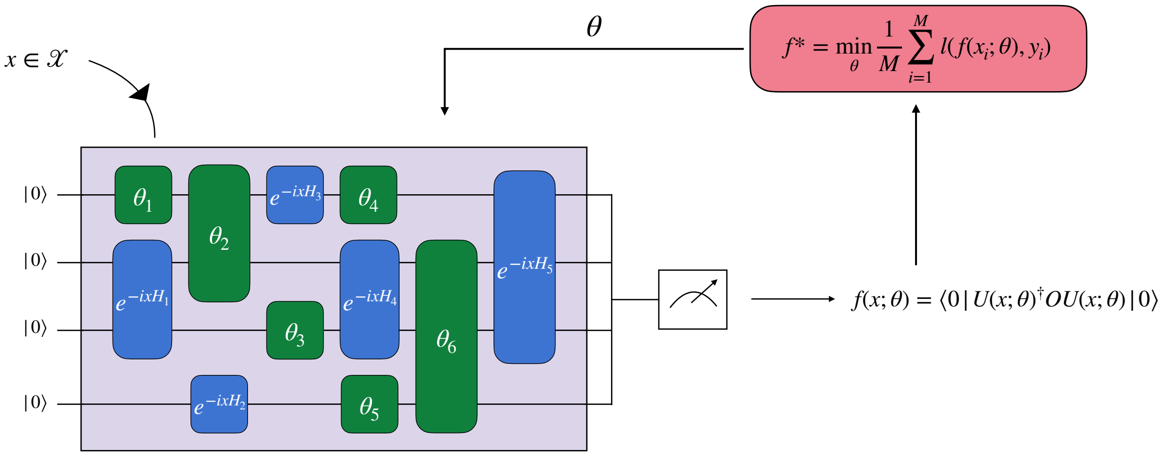


Figure 6.1: **A typical variational quantum circuit (VQC).** A data-point $x \in \mathcal{X}$ is encoded in a quantum circuit with encoding gates (blue) and trainable gates (green). Once the quantum circuit is built, it is prepared on a quantum computer, measured, and the optimal parameters θ^* are found by minimising the empirical risk. This yields a quantum model $f(x;\theta^*)$.

Precisely, we study the same type of structure that is presented in Ref. [SSM21] where the unitary is decomposed as

$$U(x;\theta) = \prod_{k=1}^{L} U_k(\theta_k) S_k(x), \tag{6.2}$$

where

$$S_k(x) = e^{-ix_1 H_1^k} \otimes \cdots \otimes e^{-ix_d H_d^k}, \tag{6.3}$$

is the $k-$th data encoding layer with fixed Hamiltonians H_i^k and $U_k(\theta_k)$ is a trainable unitary parametrized by the vector θ_k. A typical instance of $U(x;\theta)$ is illustrated in Figure 6.1. Note that many models in the quantum machine learning literature exist such as quantum circuit born machines [LW18], variational imaginary time evolution [ZLW21], differentiable quantum circuits [KPE21] or quantum orthogonal neural networks [KLM21] that are also variational but are not considered here.

Quantum models are large Fourier series

The expressivity of the quantum model we defined previously has been studied in Ref [SSM21]. The authors showed that the quantum model can be rewritten as the following Fourier series

$$f(x;\theta) = \sum_{\omega \in \Omega} c_\omega e^{i\omega x}, \tag{6.4}$$

where the spectrum Ω of frequencies is determined by the ensemble of eigenvalues of the encoding Hamiltonians and the coefficients c_ω depend on the parameterised ansatz (Figure 6.2).

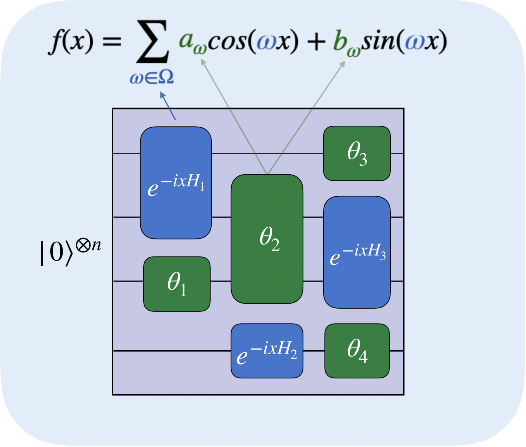


Figure 6.2: **VQCs as Fourier series.** The function of a quantum model can be written in a Fourier basis. Interestingly in our context the frequencies $\omega \in \Omega$ depend only on the eigenvalues of the encoding Hamiltonians (blue gates). On the other hand, the coefficients a_ω, b_ω are correlated to the trainable gates (green).

Let us build some intuition by explicitly building Ω in the case of a one-dimensional data input ($\mathcal{X} = \mathbb{R}$) and with a variational circuit containing exactly L encoding gates. In simple terms, the frequency spectrum Ω is the ensemble of all the differences between all possible sums of the eigenvalues of the encoding gates. Mathematically, let us note λ_ℓ^k as the k^{th} eigenvalue of the ℓ^{th} encoding Hamiltonian H_ℓ having d_ℓ eigenvalues. We use the multi-index $\boldsymbol{i} = (i_1, \dots, i_L)$ indicating which eigenvalue is taken from each encoding Hamiltonian. We define $\Lambda_{\boldsymbol{i}}$ as

$$\Lambda_{\boldsymbol{i}} = \lambda_1^{i_1} + \cdots + \lambda_L^{i_L}. \tag{6.5}$$

Finally, we can express Ω the set of frequencies as

$$\Omega = \left\{ \Lambda_{\boldsymbol{i}} - \Lambda_{\boldsymbol{j}}, \boldsymbol{i}, \boldsymbol{j} \in \prod_{\ell=1}^{L} [|1, d_\ell|] \right\}, \tag{6.6}$$

This is easier to understand schematically, as represented in Figure 6.3.

The simple example of Paul encoding

A simple case to explain the spectrum is to look at Pauli encoding where all encoding Hamiltonians are Pauli matrices (e.g. encoding gates $R_Z(x) = e^{-i\frac{x}{2}\sigma_Z}$). This encoding appears in [SSM21, CGFM+21b]. In this case all the eigenvalues are $\lambda = \pm 1/2$ and therefore the $\Lambda_{\boldsymbol{i}}$ are all the integers (or half-integers, if L is odd) in $[-L/2, L/2]$. It follows that the set of distinct values in Ω is simply the set of integers in $[\![-L, L]\!]$. In this case, there are many redundant frequencies, due to the fact that all Pauli eigenvalues are the same: only $2L + 1$ distinct frequencies appear among the 2^{2L} possible values of $\Lambda_{\boldsymbol{i}} - \Lambda_{\boldsymbol{j}}$. We therefore see that various eigenvalues between encoding Hamiltonians creates more distinct frequencies. An interesting

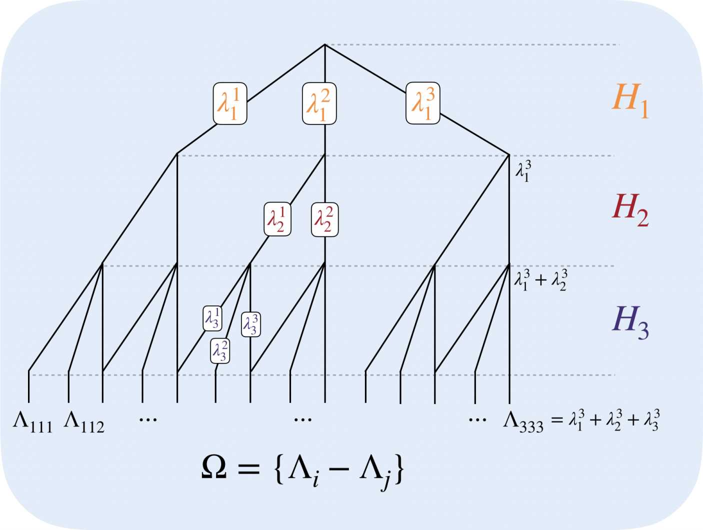


Figure 6.3: **From encoding Hamiltonians to frequencies.** The frequencies composing the VQC model (on one dimensional input) come from all the combinations of eigenvalues from each encoding Hamiltonians. By selecting one eigenvalue per Hamiltonian and adding them together we obtain a leaf value Λ. The whole spectrum corresponds to all possible differences between the leaves of this tree graph. In this figure, we consider $L = 3$ Hamiltonians. We also see potential redundancy in the leaves.

approach developed in Ref [STJ22] consists in adding scaling factors in the Pauli encoding gate to modify their eigenvalues and avoid redundancy: it results in an exponential number of integer frequencies with respect to L.

General case

Let us now suppose that $\mathcal{X} = \mathbb{R}^d$, such that we encode a vector $x = (x_1, \ldots, x_d)$ in our quantum model. In this case, Ω becomes the $d-$dimensional Cartesian product

$$\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_d, \tag{6.7}$$

where each $\Omega_{i=1,\ldots,d}$ is defined in equation (6.6) on its own set of encoding Hamiltonians.

As such, the size of the spectrum $|\Omega|$ can grow exponentially with the number of encoding gates and the dimension of the input data. For example, a d-dimensional vector x and L Pauli-encoding gates for each dimension results in a spectrum Ω that scales as $O(L^d)$. Supposing $L = 20$ and $d = 16$, it would require more than one hundred times the world's storage data capacity available in 2007 to store the entire spectrum [HL11].

The main task that we tackle in this chapter is to understand if such a large Fourier series can be approximated by classical methods. Precisely, we seek to build a classical approximator

$\tilde{f}$ as

$$\tilde{f}(x) = \sum_{\omega \in \tilde{\Omega}} \tilde{c}_\omega e^{i\omega x}, \tag{6.8}$$

such that $\tilde{\Omega}$ is of tractable size and the two solution are close, *i.e.* for a given $\varepsilon > 0$ we guarantee

$$\forall x \in \mathcal{X}, \left\| f(x) - \tilde{f}(x) \right\| \leq \varepsilon. \tag{6.9}$$

In the cases where the construction of such a model is possible, it would imply that although classically simulating the encoding VQC might not be possible, the quantum model that emerges from it can be efficiently approximated in a classical way. Our approximator relies on random Fourier Features [RR09], and requires that the kernel associated to our quantum model is shift-invariant which we prove in the next section.

6.1.1 Quantum models are shift-invariant kernel methods

As the quantum model is a real-valued function, it follows that $\omega \in \Omega$ implies $-\omega \in \Omega$ and $c_\omega = c^*_{-\omega}$. We express the Fourier series of the quantum model as a sum of trigonometric functions by defining for every $\omega \in \Omega$:

$$a_\omega := c_\omega + c_{-\omega} \in \mathbb{R}, \tag{6.10}$$

$$b_\omega := \frac{1}{i}(c_\omega - c_{-\omega}) \in \mathbb{R}, \tag{6.11}$$

such that

$$\begin{aligned} f(x;\theta) &= \sum_{\omega \in \Omega_+} c_\omega e^{i\omega x} + c_{-\omega} e^{-i\omega x}, \\ &= \sum_{\omega \in \Omega_+} a_\omega \cos(\omega x) + b_\omega \sin(\omega x), \end{aligned} \tag{6.12}$$

where Ω_+ contains only the positive frequencies from Ω. Considering only Pauli gates, if $d = 1$, we simply have $\Omega = [\![-L, L]\!]$ and $\Omega_+ = [\![0, L]\!]$. In dimension d, we have $\Omega = [\![-L, L]\!]^d$ and Ω_+ is built by keeping half of the frequencies (after removing those of opposite sign), plus the null vector. We therefore have

$$|\Omega_+| = \frac{(2L+1)^d - 1}{2} + 1. \tag{6.13}$$

Without loss of generality, we consider only Ω_+ and conveniently drop the + subscript.

Furthermore, the quantum model can be written as a linear model

$$f(x;\theta) = \langle \psi(x;\theta) | O | \psi(x;\theta) \rangle = \mathbf{w}(\theta)^T \phi(x), \tag{6.14}$$

where $\phi(x)$ is the feature vector and $\mathbf{w}$ is the trainable weight vector expressed as

$$\phi(x) = \frac{1}{\sqrt{|\Omega|}} \begin{bmatrix} cos(\omega^T x) \\ sin(\omega^T x) \\ \vdots \end{bmatrix}_{\omega \in \Omega} \quad , \quad \mathbf{w}(\theta) = \begin{bmatrix} a_\omega \\ b_\omega \\ \vdots \end{bmatrix}_{\omega \in \Omega} \quad . \tag{6.15}$$

Recall from the previous chapter that finding the optimal quantum model translates to finding the optimal weight vector $\mathbf{w}^*$. This can be done using Linear Ridge Regression [BN06]. As shown in the previous chapter, there exists a dual formulation to the regression task using the kernel associated to the feature vector. In our case, the related kernel function is defined as

$$\begin{aligned} k(x, x') &= \langle \phi(x), \phi(x') \rangle , \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \cos(\omega x) \cos(\omega x') + \sin(\omega x) \sin(\omega x'), \\ &= \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \cos(\omega(x - x')), \end{aligned} \tag{6.16}$$

which is a shift-invariant kernel, meaning that $k(x, x') = k(x - x')$.

Due to the nature of Ω, this kernel is high-dimensional which can make it hard to simulate classically in practice. But thanks to its shift-invariance, we can leverage Random Fourier Features (RFF), a seminal method known to be powerful approximator of high-dimensional kernels [RR09].

6.2 Random Fourier Features: approximating high-dimensional shift-invariant kernels

In this section we present the Random Fourier Features (RFF) method [RR09, LTOS19, SS15]. We will use this method to create several classical sampling algorithms for approximating VQCs.

Let $\mathcal{X} \subset \mathbb{R}^d$ be a compact domain and $k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$ be a kernel function. We assume k is shift-invariant, meaning

$$k(x, y) = \overline{k}(x - y) = \overline{k}(\delta), \tag{6.17}$$

where $\overline{k} : \mathcal{X} \longrightarrow \mathbb{R}$ is a single variable function, and we will note $\overline{k} = k$ to simplify the notation.

Bochner's theorem [Rud17] insures that the Fourier transform of k is a positive function and we can write

$$k(\delta) = \int_{\omega \in \mathcal{X}} p(\omega) e^{-i\omega^T \delta} d\omega. \tag{6.18}$$

If we assume k is also normalised, then the Fourier transform $p(\omega)$ of k can be assimilated to a probability distribution. With a dataset of M points, fitting a Kernel Ridge Regression (KRR) model with the kernel k necessitates M^2 operations to compute the kernel matrix and $\mathcal{O}(M^3)$ to invert it. This becomes impractical when M reaches high value in modern big datasets.

The original idea of the RFF method was therefore to sample frequencies from the shift-invariant kernel and build an approximate kernel. Mathematically, the kernel k is approximated by

$$\tilde{k}(x,y) \simeq \tilde{\phi}(y)^T\tilde{\phi}(x), \quad \text{where} \quad \tilde{\phi}(x) = \frac{1}{\sqrt{D}}\begin{bmatrix} cos(\omega_i^T x) \\ sin(\omega_i^T x)\end{bmatrix}_{i\in[\![1,D]\!]}. \tag{6.19}$$

The frequencies $\omega_{i=1,\ldots,D}$ are sampled *i.i.d.* from the frequency distribution $p(\omega)$. Formally, it is a Monte-Carlo estimate of k. In some cases like Gaussian or Cauchy kernel, $p(\omega)$ can be analytically found [RR09] which makes the sampling efficient.

Finally, instead of fitting a KRR for k, one will solve a Linear Ridge Regression (LRR) with $\tilde{\phi}$. In this case, the number of operations required is $\mathcal{O}(MD^2 + D^3)$. If D is much smaller than M, it is much cheaper than solving the KRR directly. The output of the LRR or gradient descent is simply a weight vector $\tilde{\mathbf{w}}$ that is used to create the approximate function

$$\tilde{f} = \tilde{\mathbf{w}}^T\tilde{\phi}(x). \tag{6.20}$$

To summarise, the general idea is to sample D frequencies from Ω to build a classical approximator of the quantum model. We explicitly show how to do so in the following section.

6.3 RFF Methods for Approximating VQCs

As described above, the idea behind our method is to sample frequencies from Ω and train a classical model from them. We first introduce some related work that was explored in parallel by Pr. Eisert's group [SEM22]. Then, we present three different strategies to sample frequencies from Ω to build classical models in section 6.3.1. Furthermore, we provide theoretical bounds on the number of samples required in section 6.3.2.

Related Work

A recent work independently proposed a similar approach where classical surrogate methods approximate quantum models emanating from VQCs. The difference with this work is the necessity of having access to all Ω, the totality of the frequencies of the VQC considered, without sampling from them.

Indeed, if Ω is known, the coefficients a_ω and b_ω of the VQC function (see Eq.6.12) can be easily fitted by solving the classical least square problem. Namely, one determines $\mathbf{w}^*$ such that

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{M}\sum_{i=1}^{M} |\mathbf{w}^T\phi(x_i) - y_i|^2 + \frac{\lambda_0}{M}||\mathbf{w}||^2, \tag{6.21}$$

where $\phi(x) = \begin{bmatrix} \cos(\omega x) \\ \sin(\omega x)\end{bmatrix}_{\omega\in\Omega}$, and λ_0 is the regularisation parameter. As explained in the previous section, with a dataset of M points, this can be solved exactly using matrix inversion in $\mathcal{O}(M|\Omega|^2 + |\Omega|^3)$ operations if $M \geq 2|\Omega|$. If the inequality is not fulfilled or if $|\Omega|^3$ is too big, one would use stochastic gradient descent instead of matrix inversion.

However, this method assumes that Ω is known, and is not too large, which will usually be the case as we show in Section 6.1. One should also be able to enumerate all individual frequencies $\omega \in \Omega$. For completeness, we note from the seminal work [Sch21] that the author briefly mentions the idea of approximating kernels with RFF. Similarly, in a more recent work [PS22] the authors mention RFF as a sampling strategy on VQCs with shift-invariant kernels.

6.3.1 RFF Sampling strategies

We propose three types of strategies to sample from the spectrum Ω, depending on the knowledge of the quantum model at hand. Precisely, we will explain in which case these strategies are suited according to the type of ansatz, the dimension of the input vectors and the number of training points. The three sampling strategies are named *distinct* sampling, *tree* sampling and *grid* sampling.

RFF with distinct sampling

In this sampling strategy, we assume that all the frequencies of Ω can be explicitly calculated. We then select uniformly at random D frequencies. The complete method is describe hereafter in Algorithm 5.

Algorithm 5 RFF with Distinct Sampling

```
Require: the VQC of the quantum model $f$, and $M$ datapoints $\{x_j\}_{j\in[M]}$
Ensure: Approximate function $\tilde{f}$
Diagonalize the Hamiltonians of the VQC's encoding gates.
Use their eigenvalues to obtain all frequencies $\omega \in \Omega$, as in Equation (6.6)
Sample $D$ frequencies $(\omega_1, \cdots, \omega_D)$ from $\Omega$
Construct the approximated kernel $\tilde{k}(x,y) = \tilde{\phi}(y)^T\tilde{\phi}(x)$ with $\tilde{\phi}(x) = \frac{1}{\sqrt{D}}\begin{bmatrix} cos(\omega_i^T x) \\ sin(\omega_i^T x) \end{bmatrix}_{i\in[\![1,D]\!]}$
Solve the Linear Ridge Regression problem by matrix inversion or stochastic gradient descent and obtain an optimal weight vector $\tilde{\mathbf{w}}^*$.
Obtain the approximated function $\tilde{f}(x) = \tilde{\mathbf{w}}^{*^T}\widetilde{\phi(x)}$
```

RFF with Tree sampling

The distinct sampling requires building all of Ω which can be computationally expensive. In some cases, redundancies occur in the final frequencies. As we will showcase in numerical experiments, the frequencies with more redundancy tend to obtain larger coefficients in the quantum model. Conversely, isolated frequencies are very likely to have small coefficients in comparison, making them 'ghost' frequencies in Ω. Rather than sampling with uniform probability all frequencies of Ω, it is worth drawing the redundant frequencies with a higher probability. In our tree sampling method, we propose to directly sample from the tree shown in Figure 6.3 when knowing the eigenvalue decomposition of each encoding Hamiltonian. In practice, we select uniformly at random each encoding eigenvalue. This naturally tends to sample redundant frequencies with higher probability. Furthermore, it does not require computing the whole set Ω, but only D paths through the tree (which can be used to generate up to $\binom{D}{2}+1$ positive frequencies, with potential redundancy).

Algorithm 6 RFF with Tree Sampling

```
Require: a VQC model $f$, and $M$ datapoints $\{x_j\}_{j\in[M]}$
Ensure: Approximate function $\tilde{f}$
1: Diagonalize the Hamiltonians of the VQC's encoding gates.
2: Sample $D$ paths from the tree shown in Figure 6.3, obtain $D$ frequencies $(\omega_1, \cdots, \omega_D)$ from $\Omega$
3: Follow steps 4-6 of Algorithm 5.
```

RFF with Grid sampling

The two above methods suffer from a common caveat: if one or more of the encoding Hamiltonians are hard to diagonalise, we cannot add encoding eigenvalues together to sample the final frequencies.

If the eigenvalues are unknown, we can guess an upper bound on the maximum value $\omega_{\max}$ and sample from a grid of frequencies regularly disposed between zero and the upper bound $\omega_{\max}$. In practice, the value of $\omega_{\max}$ can be chosen as the largest frequency learnable by the Shannon criterion or by using Gershgorin circle theorem [Wei03] to upper-bound the eigenvalue of each encoding Hamiltonian. Letting $s > 0$ be the step on this grid, the number of frequencies on a single dimension is given by $\omega_{\max}/s$. Over all dimensions, there are $\lceil(\omega_{\max}/s)\rceil^d$ frequency vectors. We will show later on that the error between the quantum model f and the approximation $\tilde{f}$ can be bounded by s. When s is small enough, the number D of samples necessary to reach an error $\epsilon > 0$ grows like $1/\epsilon^2 \log(1/s)$.

Algorithm 7 RFF with Grid Sampling

```
Require: Assumption on the highest frequency $\omega_{\max}$, a step $s > 0$ and $M$ datapoints $\{x_j\}_{j\in[M]}$
Ensure: Approximate function $\tilde{f}$
1: Create a regular grid in $[0, \omega_{\max}]^d$ with step $s$.
2: Sample $D$ frequencies $(\omega_1, \cdots, \omega_D)$ from the grid.
3: Follow steps 4-6 of Algorithm 5.
```

We now describe the approximation error that is achieved with this model as a function of the number of samples used. We derive several versions according to the sampling method used.

6.3.2 Approximation Error and number of samples

In classical ML, researchers have bounded the approximation error in the RFF method [SS15]. Precisely, the following theorem bounds the actual prediction error when using RFF compared to the kernel ridge regression (KRR) model estimate:

Theorem 6.1 ([SS15])**.** *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$. We consider a training set $\{(x_i, y_i)\}_{i=1}^{M}$. Let f be the KRR model obtained with the true kernel k and regularisation $\lambda = M\lambda_0$ for $\lambda_0 > 0$, and $\tilde{f}$ the KRR model obtained with the approximate kernel and the same regularisation. Then we can guarantee $|f(x) - \tilde{f}(x)| \leq \epsilon$ with probability $1 - \delta$ for a number D of samples given by:*

$$D = \Omega\left(d\left(\frac{(\lambda_0+1)\sigma_y}{\lambda_0^2\epsilon}\right)^2\left[log(\sigma_p|\mathcal{X}|) + log\frac{(\lambda_0+1)\sigma_y}{\lambda_0^2\epsilon} - log\delta\right]\right), \tag{6.22}$$

with $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^{M} y_i^2$ and σ_p, and $|\mathcal{X}| = \max_{x,x'\in\mathcal{X}}(\|x - x'\|)$ is the diameter of $\mathcal{X}$. Note that in Eq.6.22 the notation Ω stands for the computational complexity "Big-Ω" notation.

Theorem 6.1 enables to derive a lower bound on the number of necessary samples D for the approximation error to be less than $\epsilon > 0$. Remarkably, the bound on D grows linearly with the input dimension d and logarithmically with σ_p, which is linked to the variance of the frequency distribution $p(\omega)$. In our case, the continuous distribution $p(\omega)$ will be replaced by the actual set of frequencies Ω

$$p(\omega) = \frac{1}{|\Omega|}\sum_{\omega\in\Omega}\delta_\omega, \tag{6.23}$$

where δ_ω represents the Dirac distribution at ω. As a result, we can write the discretised variance as

$$\sigma_p = \sum_{\omega\in\Omega} p(\omega)\omega^T\omega. \tag{6.24}$$

From this, we can explicit the link between the number D of samples necessary and the size of Ω or the number L of encoding gates per dimension.

In the general case, we consider that σ_p is the average value of $\omega^T\omega$, that is to say the trace of the multidimensional variance: the more the frequencies are spread, the higher σ_p will be.

Finally, note that it is important to take into account the Shannon criterion, stating that one needs at least $2\omega_{\max}$ training points to estimate the coefficients of a Fourier series of maximum frequency $\omega_{\max}$. In practice, it puts some limitation on the largest frequency one can expect to learn on a dataset. For example, exponentially large frequencies within VQCs as depicted in [STJ22] would have a limited interest when only a low number of training points are available.

In the following sections, we provide the minimum bound in the case of Paul encoding.

Pauli encoding

We provide here a bound on the minimum of samples required to achieve a certain error between the RFF model and the complete model in the case of Pauli encoding in the distinct sampling strategy.

Theorem 6.2. *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$. We consider a training set $\{(x_i, y_i)\}_{i=1}^M$. Let f be a VQC model with L encoding Pauli gates on each of the d dimensions and full freedom on the associated frequency coefficients, trained with a regularization λ. Let $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^M y_i^2$ and $|\mathcal{X}|$ the diameter of $\mathcal{X}$. Let $\tilde{f}$ be the RFF model with D samples in the distinct sampling strategy trained on the same dataset and the same regularization. Then we can guarantee $|f(x) - \tilde{f}(x)| \leq \epsilon$ with probability $1-\delta$ for a number D of samples given by:*

$$D = \Omega\left(\frac{dC_1(1+\lambda)^2}{\lambda^4\epsilon^2}\left[log(dL^2|\mathcal{X}|) + log\frac{C_2(1+\lambda)}{\epsilon\lambda^2} - log\delta\right]\right), \tag{6.25}$$

with C_1, C_2 being constants depending on σ_y, $|\mathcal{X}|$. We recall that in Eq.6.25 the notation Ω stands for the computational complexity "Big-Ω" notation.

Proof. We know that $\Omega = [\![-L, L]\!]^d$. In one dimension, we simply have $\sigma_p = 1/L\sum_{\ell=-L,\cdots,L}\ell^2 = O(L^2)$. In dimension d, a frequency ω is given by its values on each dimension $(j_1, \cdots, j_d)$ with $j_k \in [|-L, L|]$. We similarly have

$$\sigma_p = \frac{1}{(2L+1)^d}\sum_{j_1,\cdots,j_d} j_1^2 + \cdots + j_d^2. \tag{6.26}$$

Note that $\sum_{j_1,\cdots,j_d} j_1^2 + \cdots + j_d^2$ is $d(2L+1)^{d-1}$ times the sum of all squares,

$$\begin{aligned}\sigma_p = \frac{d(2L+1)^{d-1}}{(2L+1)^d}\sum_{\ell=-L}^{L}\ell^2 &= \frac{d}{2L+1}\,\frac{2L(L+1)(2L+1)}{6},\\ &= O(dL^2) = O(d|\Omega|^{2/d}).\end{aligned} \tag{6.27}$$

The expression is then obtained by replacing the value of σ_p in Theorem 6.1. □

We can conclude that the number D of samples grows linearly with the dimension d, and logarithmically with the size of Ω. Furthermore, the result can be extended to scaled Pauli encoding by replacing L with a term growing as c^L where c is a constant. D would therefore grow linearly in L rather than logarithmically. Note however that the scaling in ϵ and λ, respectively in $\Omega(1/\epsilon^2)$ and $\Omega(1/\lambda^4)$, is not favourable and can limit in practice the use of the RFF method.

Note that this proof changes if we take into account the redundancy of each frequency when sampling from Ω. This will be the case in the *Tree Sampling* strategy (see Section 6.3.1). In that case, the variance becomes even smaller since some frequencies are more weighted than others, in particular for Pauli encoding.

Grid sampling

We provide here a bound on the minimum number of samples required to achieve a certain error between the RFF model and the complete model in the case of a general encoding in the gird sampling strategy. The proof for this theorem has been done by my clever colleague Slimane Thabet and is given in the Appendix for completeness.

Theorem 6.3. *Let* $\mathcal{X}$ *be a compact set of* $\mathbb{R}^d$*, and* $\epsilon > 0$*. We consider a training set* $\{(x_i, y_i)\}_{i=1}^{M}$*. Let* f *be a VQC model with any Hamiltonian encoding, with a maximum individual frequency* $\omega_{\max}$ *and full freedom on the associated frequency coefficients, trained with a regularisation* λ*. Let* $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^{M} y_i^2$ *and* $|\mathcal{X}|$ *the diameter of* $\mathcal{X}$*. Let* $\tilde{f}$ *be the RFF model with* D *samples in the grid strategy trained on the same dataset and the same regularisation. Let* $C = |f|_\infty |\mathcal{X}|$ *and* s *the sampling rate defined in the grid sampling strategy. Then we can guarantee* $|f(x) - \tilde{f}(x)| \leq \epsilon$ *for* $0 < s < \frac{1}{C}$ *with probability* $1 - \delta$ *for a number* D *of samples given by:*

$$D = \Omega\left(\frac{dC_1(1+\lambda)}{\lambda^4(\epsilon - sC)^2}\left[\log(\omega_{\max}|\mathcal{X}|) + \log\frac{C_2(1+\lambda)}{\lambda^2(\epsilon - sC)} - \log\delta\right]\right), \tag{6.28}$$

with C_1 *and* C_2 *being constants depending on* σ_y *and* $d(X)$*. We recall that in Equation (A.23) the notation* Ω *stands for the computational complexity 'Big-*Ω*' notation.*

Here, we see that quite intuitively the number of samples will depend inversely on the size of the spacing. In the following section, we run numerical experiments to check our theoretical findings.

6.4 Numerical experiments

In this section we put to practice our method and aim to assess the accuracy and efficiency of RFF to approximate quantum models. Each VQC was analysed using noiseless simulations using PyQtorch[1], an efficient large-scale emulator designed for quantum machine learning integrated with a PyTorch backend.

Approximating random quantum models

A typical random VQC instance is built from a list of general encoding Hamiltonians $\{H_1, \cdots, H_k\}$ applied to randomly selected qubits (Figure 6.4). Note that for simulation purposes the number of qubits is fixed to five.

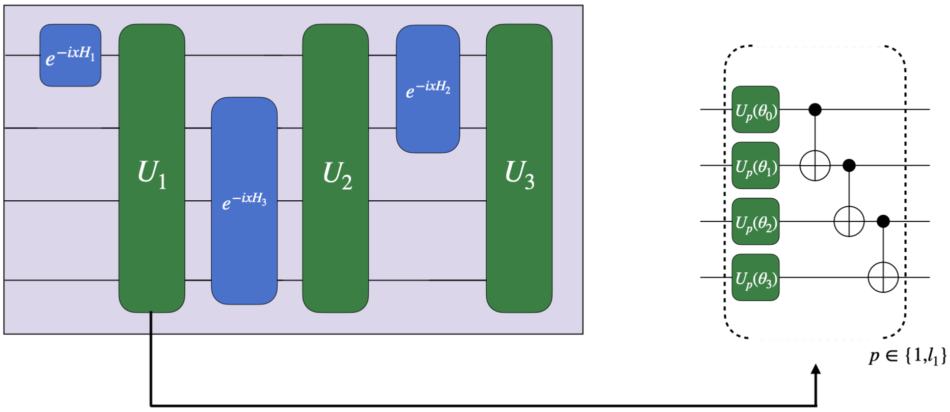


Figure 6.4: **Random instance of a VQC.** In this example, three encoding Hamiltonians $\{H_1, H_2, H_3\}$ are randomly assigned over four qubits, and load a 1-dimensional vector x. Following each encoding gate H_i, an ansatz with trainable parameters and a ladder of CNOTs is applied, l_i times in a row.

The dataset consists of M vectors $\{x_1, \ldots, x_M,\ x_i \in \mathbb{R}^d\}$. It is prepared by considering a grid of evenly distributed points over

$$X_{\text{grid}} = \prod_{i=1}^{d} [0, x_{\max_i}], \tag{6.29}$$

and the associated labels Y_{grid} corresponds to the evaluation of the quantum circuit over the input points. According to the sampling method, we sample D frequencies, build the approximated kernel and obtain an approximated function $\tilde{f}$ by solving the Linear Ridge Regression (LRR) (Algorithm 5). We use the Mean Square Error (MSE) of the LRR as a metric for the approximation which effectively corresponds to the train loss value, defined as

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^{M} |\mathbf{w}^T \phi(x_i) - y_i|^2. \tag{6.30}$$

---

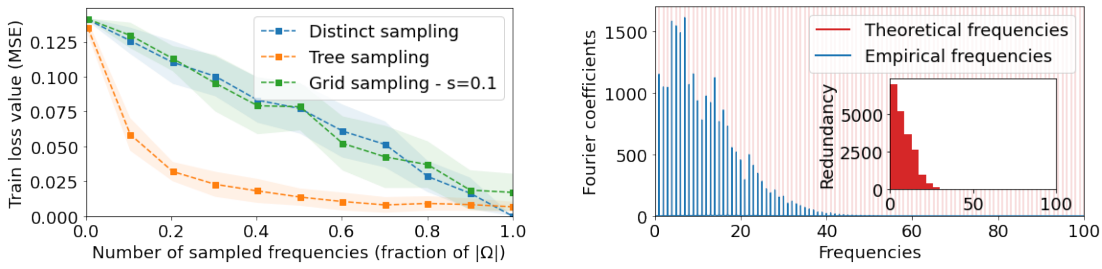
[1] https://github.com/pasqal-io/PyQ

We benchmark each RFF sampling strategy (distinct, tree, grid) against several types of encoding gates: Pauli-encoding, exponential Pauli encoding and a XYZ Hamiltonian.

Pauli encoding

We first consider a quantum model with $L = 200$ Pauli encoding gates for a uni-dimensional input vector $x \in \mathbb{R}$. The resulting spectrum consists of integer-frequency and reads $\Omega = [\![0, L]\!]^d$. In this case, the quantum model is a periodic function of period $T = 2\pi$ and we can therefore limit the grid to $X_{\text{grid}} = [0, 2\pi]$.

We estimate the MSE of our RFF model as a function of the fraction of the full spectrum size $|\Omega|$, and average over 10 different random initialisation (Figure 6.5(a)). On average, the RFF training error for distinct and grid sampling is a linear function of the number D of samples taken from Ω. On the other hand, the error using tree sampling exhibits a faster decay, reaching relatively low errors with only 20% of the spectrum size. An interpretation of the tree sampling being more efficient is that it naturally follows the redundancy of the spectrum frequencies. Indeed, we note that the Fourier coefficients of the VQC are correlated on average to the frequency redundancy in the empirical quantum spectrum (Figure 6.5(b)). Frequencies above a certain threshold $\omega_{\text{effective}}$ are cut from the quantum model empirical spectrum. The effective spectrum of the VQC is therefore smaller than the one predicted in theory.



(a) Evolution of RFF train loss as a function of the relative number of frequencies sampled.

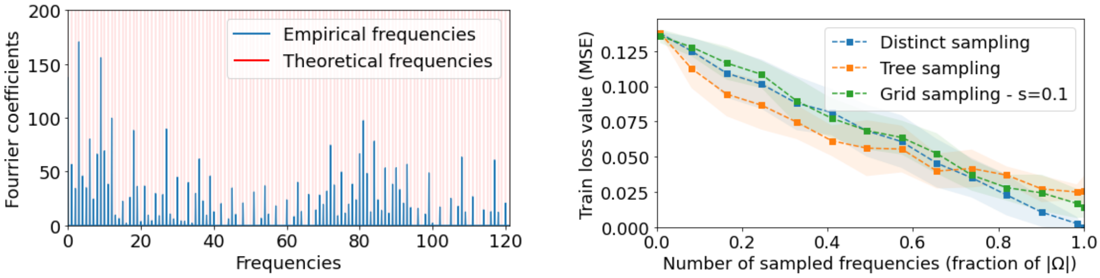
(b) Average Fourier Transform of the VQC's quantum models.

Figure 6.5: **RFF *vs.* 1D Pauli encoding.** In subplot (a) we plot the MSE associated to RFF models relative to a quantum model consisting of $L = 200$ Pauli gates. following the three different sampling schemes. The Tree sampling strategy takes advantage of the high redundancy to sample less frequencies to reach a good approximation (subplot (b)). The frequencies with high coefficients are the ones with high redundancy in Ω (inner red histogram). Frequencies over 100 have negligible coefficients and redundancy, and therefore are not shown.

6.4.0.1 Scaled Pauli encoding

In order to obtain VQCs with a large number of frequencies but low redundancy, we follow the exponential encoding scheme proposed in Ref [STJ22] resulting in a uniform probability distribution over integers. In this encoding strategy, encoding Pauli gates are enhanced with a scaling coefficient β_{kl} for the l^{th} Pauli rotation gate encoding the component x_k. This gives us a total of 3^{Ld} positive and negative frequencies. These frequencies can be all distinct with the particular choice of $\beta_{kl} = 3^{l-1}$ resulting in an exponentially large and uniform spectrum Ω.

We test our classical RFF approximation in Figure 6.6. This time we observe that all three strategies have a linear scaling, showing that the absence of redundancy in the spectrum makes tree sampling on par with the other sampling methods.



(a) Spectrum Ω for an exponential encoding VQC.

(b) RFF approximation performance

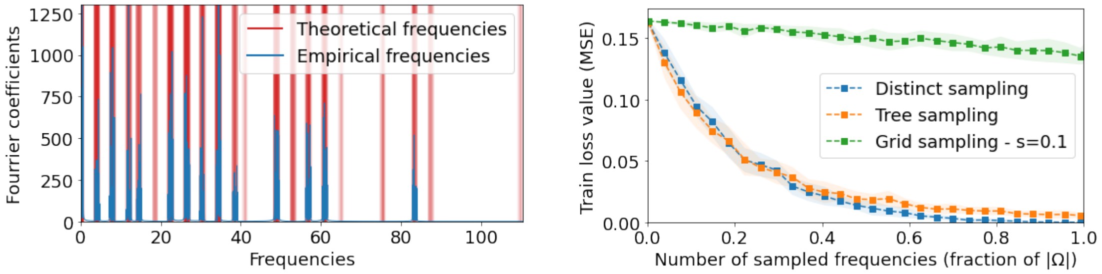
Figure 6.6: **RFF *vs.* exponential Pauli encoding** with $L = 5$ layers. Here, the Pauli gates are scaled in order to suppress the redundancy of the spectrum Ω. The empirical frequencies are therefore spread across the whole theoretical spectrum, making the model more expressive. The three sampling methods are on par and show a linear decrease with the number of sampled frequencies.

6.4.0.2 XYZ Hamiltonian encoding

We now consider a quantum model where the encoding Hamiltonian is a more general Hamiltonian. Precisely, we consider a general two-body XYZ interaction Hamiltonian that reads

$$H_{XYZ} = \sum_{\langle i,j \rangle} \alpha_{ij} \hat{\sigma}_x^i \hat{\sigma}_x^j + \beta_{ij} \hat{\sigma}_y^i \hat{\sigma}_y^j + \gamma_{ij} \hat{\sigma}_z^i \hat{\sigma}_z^j + \sum_i \delta_i \hat{\sigma}^i, \tag{6.31}$$

where $\langle i, j \rangle$ indicates a pair of connected qubits and $\hat{\sigma}^i = \{\hat{\sigma}_x^i, \hat{\sigma}_y^i \text{ or } \hat{\sigma}_z^i\}$. An example of the resulting spectrum is illustrated in Figure 6.7(a). With this complex encoding, the frequencies of the spectrum Ω are distributed in concentrated packets. When training the RFF, the grid sampling is not effective as the spacing is too big and the error seems to be in line with the theoretical bounds given in Theorem 6.3. The distinct and tree sampling on the other hand show a power-law decay of the error as the number of sampled frequencies increases. A possible interpretation is that because the frequencies are concentrated in packets, sampling an effective frequency ω_{eff} per packets allows the RFF to find a good approximate model. This could point towards the fact that many frequencies that are too close to each other yield quantum models that are not so hard to approximate with an RFF method.



(a) Spectrum Ω

(b) RFF train loss.

Figure 6.7: **RFF *vs.* 3-qubits H_{XYZ} Hamiltonian**. We depict the spectral distribution of frequencies for a VQC consisting of a XYZ Hamiltonian and 4 scaled Pauli-gates.The intensity of the vertical red lines indicates the concentration of the theoretical frequencies in Ω. The distinct sampling benefits from the concentration of frequencies in packets to approximate with less samples.

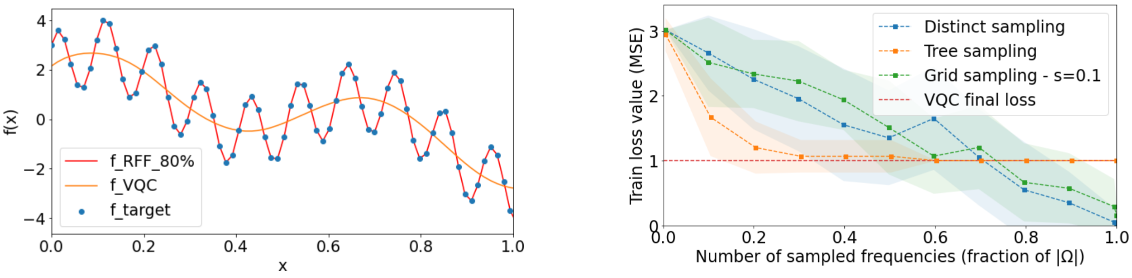
To conclude this first experimental part, we observe that when some frequencies in the spectrum Ω have much redundancy (Pauli encoding), these frequencies are empirically the ones with higher coefficients. In such case, the tree sampling strategy is able to approximate the quantum model with fewer samples than the other methods as expected. With more complex Hamiltonians, concentrated packets of frequencies appear and even without much redundancy both tree and distinct sampling require fewer frequency samples to cover these packets. According to these experiments, the worst case scenario for the RFF is a uniform probability distribution where all the three sampling techniques will be equivalent. Nonetheless, the theoretical bounds prove that the number of Fourier features will scale linearly with respect to the spectrum size that scales itself exponentially. The poor results of grid sampling in the case of a general Hamiltonian also points towards the fact that for a large many-body Hamiltonian that cannot be diagonalised we can expect the RFF method to poorly approximate the quantum model.

Comparing VQC and RFF on artificial target functions

Another interesting numerical comparison is to measure the efficiency of both a quantum model and a RFF trained from the same frequencies on a common target function. The goal of this experiment is to understand if quantum models, while having access theoretically to given frequencies, can be trained to "use" them. Recall that in the previous section we saw that for VQCs with Pauli encoding, the Fourier coefficients are rapidly decreasing, cutting out frequencies higher than $\omega_{\text{effective}}$ from the empirical spectrum. For this reason, we have chosen a particular synthetic target function: we create a sparse Fourier series as a target function $s(x)$ defined as

$$s(x) = \sum_{\omega \in \{4,10,60\}} \cos(\omega x) + \sin(\omega x), \tag{6.32}$$

and we train a quantum model with $L = 200$ 1D-Pauli encoding gates. Interestingly, the tree sampling, which shadows the redundancy in the spectrum, cannot fit the artifical function and plateaus at the same level as the quantum model's final loss value (Figure 6.8(b)) as they miss on the higher frequency. On the other hand, the RFF models based on distinct and grid sampling can effectively learn the target function with enough frequency samples. This result shows that even when a VQC with Pauli encoding is trained, it cannot reach all of its theoretical spectrum thus questioning the expressivity of such a quantum model.



(a) Predicting a target function.

(b) RFF train loss.

Figure 6.8: **Fitting an artifical target function** with a VQC architecture of $L = 200$ Pauli gates compared to an RFF model. The predictions of the target function $s(x)$ are plotted in Figure (a) and shows that the quantum model f_{VQC} cannot fit the highest frequency, although it exists in its theoretical spectrum. Clearly, the RFF with distinct and grid sampling are able to outperform the quantum model, while the tree sampling shadows the redundancy in the VQC and hence achieves the same loss.

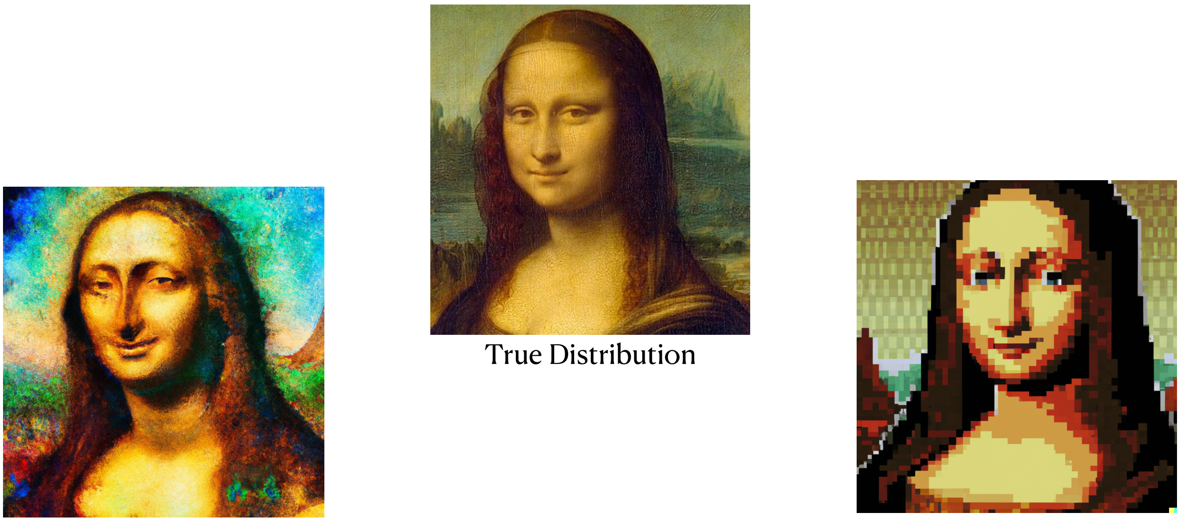
Numerical experiments discussion

To conclude these numerical experiments, we summarise a few interesting results. First, the simulations confirm that the spectrum of VQCs are predictable from their encoding gates. We observe however that the effective set of frequencies does not cover the whole spectrum, questioning the effective expressivity of VQCs in practice. It also confirms the intuition that frequencies that appear with high redundancy in the spectrum tend to have larger coefficients. Furthermore, the three RFF alternatives we propose to approximate quantum models on several tasks revealed to be efficient. The number D of samples seen as a fraction of the spectrum size grows favourably and allows for good approximation. With a highly redundant spectrum, the tree sampling method approximate quantum models with fewer samples. It interestingly inherits the same drawbacks as VQCs and are sometimes not capable of learning less redundant frequencies in the spectrum. On the other hand, RFF with distinct and grid sampling can outperform the quantum model in such cases.

6.5 Conclusion

In this chapter, we have adapted the RFF sampling strategy to approximate quantum models emanating from variational quantum circuits with Hamiltonian encoding. Our three sampling methods use the fact that sampling few random frequencies from a large dimensional kernel can be enough to provide a good approximation. It is worth noting that this can be done given only the description of the circuit and without requiring running it on a quantum computer. We studied in depth the number of samples and its dependence to different properties of the quantum circuit (input dimension, encoding Hamiltonians, circuit depth, spectrum, number of training points). On the theoretical side, the number of necessary samples grows favourably with the size of the spectrum, but suffers from potentially high overhead from the regularisation parameter. Experimentally, we have tested our classical approximators on several use cases and the approximation power of RFFs of practically relevant quantum circuits will have to be assessed in further experiments. Interestingly, Ω increases as well when the encoding Hamiltonians have distinct eigenvalues and are acting on many qubits. Therefore, quantum computers that can natively implement Hamiltonians acting on many qubits would be a plus for enlarging the spectrum.

As the Hamiltonians become larger we could reach a limit where it becomes impossible to diagonalise them. In such a case, without sampling access to Ω, the *Distinct* and *Tree* sampling strategies would be unavailable. The *Grid* sampling scheme would suffice until suffering from the high dimensionality or other factors detailed above. Overall, some limits for our classical methods can be guessed and observed already, but the main ones remain to be measured on real and larger scale quantum computers. We leave this research for future work. On another hand, one could want to understand better the relation between the available frequencies and their amplitude in practice, to find potential singularities that could help, or not, the VQCs. Finally, we want to insist on the fact that the assumptions on VQCs are crucial on the whole construction that we propose, and that some of them could be questioned, especially concerning the encoding. For instance, when encoding vectors $x = (x_1, \cdots, x_d)$, not having encoding gates expressed as $\exp(-x_i H)$ could potentially change the expression of $f(x;\theta)$ and therefore would change the associated kernel. For instance, in [KPE21], the authors use $\exp(-\arcsin(x_i)H)$ to encode data, resulting in f being expressed in the Chebyshev basis instead of the Fourier one. More generally, understanding what happens with an encoding of the form $\exp(-g(x_i)H)$, and whether we can still use our classical approximation methods, remain an open question. Similar questions arise if we use simultaneous components encoding $\exp(-x_i x_j H)$, or other alternative schemes. We believe that our work showed that somewhat surprisingly some quantum models can be (theoretically) approximated in a classical fashion although running the corresponding quantum circuit cannot be done classically. We hope that this work will pave the way towards designing quantum models that cannot be approximated classically with RFFs in a pursuit of quantum advantage in QML.



True Distribution

Quantum Model

random Fourier Features

Figure 6.9: **Visualisation of our method.** The true distribution to find is the Mona Lisa (centre). A quantum model (left) can be highly complex: see how detailed the painting is and the range of colours across the quantum Mona Lisa! This complex quantum model reproduces somewhat the original painting. The RFF method on the other side samples and creates a model with much less complexity (pixelised Mona Lisa) but still yields a good approximation of the painting. Our main message is that quantum models with high complexity can be approximated with lower-dimensional classical kernels. These images where generated by Leonardo da Vinci and with the assistance of DALL · E 2.

CONCLUSION AND OUTLOOK

We have shown in this thesis that neutral atom QPUs, historically used to simulate intractable quantum dynamics, can also be used to tackle combinatorial optimisation problems and machine learning tasks. Thanks to our facilitated access to the device, we had the chance of running our tailored algorithms directly on hardware.

To summarise, we demonstrated in Part I that industrial problems such as Smart-Charging can be embedded as the Maximum Independent Set problem on unit-disk graphs (UD-MIS), which itself can be encoded as the Ising Hamiltonian over Rydberg atoms. Then, we used tools from computational complexity theory and statistical physics to shed light on the inherent difficulty of solving UD-MIS with classical methods and the quantum annealing algorithm (QAA). Furthermore, we proposed a new scheme to embed more general classes of graphs using 3D registers of atoms. We obtained experimental results for unit-ball graphs, and expect experimental feasibility in the future for bounded-degree graphs of maximum degree 6. These results open many exciting perspectives for tackling combinatorial problems with neutral atoms.

First, quantum annealing has only been considered as a quantum algorithm to find *exact* solutions. Bottlenecks have therefore been identified as exponentially closing gaps appearing at the very end of the annealing path [AKR10, AC09, CCL$^+$23]. It would be very interesting however to consider an annealing scheme that stops before the bottleneck and study the quality of the obtained state. If the exponentially closing gap is indeed asymptotically close to the end, then stopping right before could lead to a solution of high quality, which is of great importance in practice. By bounding the quality of this approximate solution, one could open a new paradigm of quantum approximation algorithms with *a priori* guaranteed performances. Linked to this idea, it is also important to understand how and when solutions of good quality appear during quantum annealing. Experiments on disordered alloys indeed suggest that spin glasses can be brought into low-energy states faster by annealing with quantum fluctuations than by conventional thermal annealing [KRL$^+$23]. One could then imagine using quantum annealing as an accelerator towards low-energy states in combinatorial optimisation problems, yielding non-trivial approximate solutions. Approximation algorithms in the quantum complexity setting have been explored in the past [Has12, GK12a, GK12b] but QAA has been very much left aside. There are however many mechanisms that have been put forward by physicists (Kibble-Zurek mechanism and quantum phase transitions [Dzi10]) that could be used to better qualify quantum approaches for approximate solutions.

In Part II, we showcased how quantum dynamics over ensemble of Rydberg atoms can be used as a quantum feature map for graphs. We then built upon the quantum feature map a quantum kernel named Quantum Evolution Kernel (QEK). We implemented our method on hardware and showed that the geometry induced by QEK cannot be learned by classical means. Surprisingly however we did not get substantially higher results on the classification challenge at hand. Our interpretation is that the dataset was not mapped in a particularly useful way in this rich quantum feature space. To confirm this intuition, we created an artificial dataset inspired by spin-glass physics for which quantum effects yielded discriminating dynamics. In this case, QEK performed much better than all other classical kernels tested. A naive argument for quantum machine learning is to expect better results because the quantum dynamics are intractable and give rise to a high-dimensional complex feature space. We discovered however with our experiment that it is not always the case. This triggered our interest in understanding if a variational quantum circuit, while being classically hard to simulate, can give rise to a quantum model that can actually be approximated by classical means. We answered positively this question with the use of random Fourier features (RFF) and showed that many proposed models of variational quantum circuits as quantum models can be approximated with reasonable quality. This brings new guidelines to the construction of quantum models, for example applying a global Hamiltonian on many qubits yields one of the hardest model to approximate with RFFs.

The field of QML is growing strongly thanks to the increasing availability of quantum devices and encouraging theoretical results. An interesting question is trying to guess where the quantum advantage can come from. For classical datasets, it is hard to prove a quantum advantage as classical models perform already extraordinarily well. Although we have shown interesting results at a small scale, a fair benchmark requires comparing classical ML models with quantum ones on very big datasets, which cannot unfortunately be encoded efficiently on quantum devices just yet. However, it seems quite natural to assume that QML will beat its classical counterpart on quantum datasets. As we have seen in this thesis, the quality of QEK changes depending on the dataset at hand and performs well on the artificial dataset. This is because the graphs where labelled according to their underlying lattices that yield distinct frustrations during quantum dynamics. We can therefore expect that QML will demonstrate advantage over classical methods for datasets that are naturally biased with respect to quantum properties. With the development of quantum memories [CHQ+20] quantum computers will hopefully have access to quantum states directly, making QML the most suited method to analyse and learn on quantum datasets directly.

APPENDIX

A

APPENDIX

A.1 Integer linear reformulation of the (QP) problem

The (QP) problem can be solved by a classical reformulation-linearisation of non-convex quadratic programs into binary/integer ones, by considering variables x_i and y_i as integers and replacing them by their binary expansions in constraints(A.16) and (A.17), then adding binary variables and appropriate constraints to represent the products between them [PB17, EL19, BSA08]. The resulting binary/integer linear model is as follows:

$$\text{Find}\{(x_i, y_j)\} \tag{A.1}$$

$$s.t.$$

$$\forall i \in V, x_i = \sum_{k=0}^{k=log(\overline{L})} 2^k bx_i^k; \quad y_i = \sum_{k=0}^{k=log(\overline{L})} 2^k by_i^k \tag{A.2}$$

$$\forall i \in V, x_i < \overline{L}; y_i < \overline{L} \tag{A.3}$$

$$\forall \langle i, j \rangle \in E :$$

$$\rho^2 \leq X_jX_j - 2X_iX_j + X_iX_i + Y_jY_j - 2Y_iY_j + Y_iY_i \leq r^2 \tag{A.4}$$

$$X_iX_j = \sum_{k=0}^{k=log(\overline{L})} \sum_{k'=0}^{k'=log(\overline{L})} 2^{(k+k')} wx_{i,j}^{k,k'} \tag{A.5}$$

$$Y_iY_j = \sum_{k=0}^{k=log(\overline{L})} \sum_{k'=0}^{k'=log(\overline{L})} 2^{(k+k')} wy_{i,j}^{k,k'} \tag{A.6}$$

$$wx_{i,j}^{k,k'} \leq bx_i^k; \quad wx_{i,j}^{k,k'} \leq bx_j^{k'}; \quad wx_{i,j}^{k,k'} \geq bx_i^k + bx_j^{k'} - 1 \tag{A.7}$$

$$(QPRLT) \qquad wy_{i,j}^{k,k'} \leq by_i^k; \quad wy_{i,j}^{k,k'} \leq by_j^{k'}; \quad wy_{i,j}^{k,k'} \geq by_i^k + by_j^{k'} - 1 \tag{A.8}$$

$$\forall \langle i, j \rangle \in \overline{E} :$$

$$X_jX_j - 2X_iX_j + X_iX_i + Y_jY_j - 2Y_iY_j + Y_iY_i > r^2 \tag{A.9}$$

$$X_iX_j = \sum_{k=0}^{k=log(\overline{L})} \sum_{k'=0}^{k'=log(\overline{L})} 2^{(k+k')} wx_{i,j}^{k,k'} \tag{A.10}$$

$$Y_iY_j = \sum_{k=0}^{k=log(\overline{L})} \sum_{k'=0}^{k'=log(\overline{L})} 2^{(k+k')} wy_{i,j}^{k,k'} \tag{A.11}$$

$$wx_{i,j}^{k,k'} \leq bx_i^k; \quad wx_{i,j}^{k,k'} \leq bx_j^{k'}; \quad wx_{i,j}^{k,k'} \geq bx_i^k + bx_j^{k'} - 1 \tag{A.12}$$

$$wy_{i,j}^{k,k'} \leq by_i^k; \quad wy_{i,j}^{k,k'} \leq by_j^{k'}; \quad wy_{i,j}^{k,k'} \geq by_i^k + by_j^{k'} - 1 \tag{A.13}$$

$$bx_i^k, by_i^k, wx_{i,j}^{k,k'}, wy_{i,j}^{k,k'} \text{all binary.} \tag{A.14}$$

Once rewritten this way, and assuming that it is feasible, (QP) can be solved by any conventional techniques dedicated to linear discrete constraint satisfaction problems. However, it is wort noting that this formulation quickly leads to a large amount of binary variables, since for each couple (x_i, y_i) of continuous variables modeling a vertex in the original problem, we introduce $2log(\overline{L})$ binary variables (bx_i, by_i) to expand its integer representation, and $(log(\overline{L}))^2$ binary variables $wx_{i,j}^{k,k'}/wy_{i,j}^{k,k'}$ to express each product in the constraints related to the edges.

A.2 From Interval Graphs to Unit Disk Graphs

Graphs such as the ones used to model our Job Interval Selection problem (SC2) do not correspond to UD graphs as they are unions of *interval intersection graphs* (the edges corresponding to the overlapping tasks) and of *cluster graphs* (the set of complete disjoint cliques corresponding to the groups) and thus *one-dimensional* intersection graphs, rather than *two-dimensional* ones. Hence, we have to transform our scheduling graphs in order to implement the search of their MIS on the quantum machine.

The mapping between the original problem graph and the locations of the Rydberg atom/qubits in the machine can be obtained by solving the following continuous quadratic constraint satisfaction problem (QP):

$$\text{Find}\{(x_i, y_j)\} \tag{A.15}$$

$$s.t.$$

$$\forall \langle i,j \rangle \in E, \rho^2 \leq (x_j - x_i)^2 + (y_j - y_i)^2 \leq r_b^2 \tag{A.16}$$

$$\forall \langle i,j \rangle \in \overline{E}, (x_j - x_i)^2 + (y_j - y_i)^2 > r_b^2 \tag{A.17}$$

$$\forall i \in V, x_i < L;\ y_i < L \tag{A.18}$$

where:

1. $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is the original interval graph of the problem;
2. $\overline{\mathcal{G}} = (\mathcal{V}, \overline{\mathcal{E}})$ is $\mathcal{G}$'s complementary, i.e. the graph whose vertices (i, j) are connected *iff* they are not connected in $\mathcal{G}$;
3. r is the Rydberg blockade radius, i.e. the upper bound on the distance between two connected vertices, $\rho \sim r/3$ a given factor defining the lower bound on their distance, illustrating the minimal spacing between separated atoms that is experimentally feasible;
4. (x_i, y_i) are the coordinates in the Euclidean plan of the atom/qubit representing the node $i \in V$;
5. L defines the maximum square area available to place atoms on the machine, while imposing the above constraints.

Experimentally, the Rydberg blockade radius r_b can be around 15μm, for minimal distances between atoms of the order of 5μm, and the atoms are contained in a region characterised by $L \sim 100\mu$m.

Ideally, the transformation of the original (SC2) graph into one respecting the constraints of the quantum device, i.e. the solving of the (QP) problem above, should be **(i)** always feasible, and **(ii)** sufficiently efficient to preserve the quantum advantage expected when solving the UD MIS problem on the quantum machine, thanks to the "natural" embedding of this problem on it, as shown in the section 2.1.2.1 above. Unfortunately these two conditions cannot be met in all cases. First, depending on the machine characteristics (r_b, ρ, L) and the structure of the original interval graph G, (QP) does not necessarily have feasible solutions. For example, strongly connected graphs make it difficult to satisfy the constraint of bringing together the atoms corresponding to connected nodes in the rings delimited by (r_b, ρ).

Furthermore due to the lower bound in constraints (A.16) and to constraints (A.17), (QP) is not convex, and is thus NP-hard in the general case. Classical reformulation-linearisations of non-convex quadratic programs are available for (QP)-like problems [PB17, EL19, BSA08], leading to integer/binary linear constraint satisfaction problems (CSP) that can be tackled by well-studied conventional techniques. The drawback however of this approach is the large number of binary variables. Such a formulation for (QP) is presented in Section A.1. It was tested on our graph instances using the Ibm Cplex solver. As an alternative, we used a more compact linear formulation which, although not providing a guarantee to find an existing solution in the elapsed time allowed, did perform well on the majority of smart-charging graphs tested. The idea is to replace the quadratic constraints by linear ones but expressing the inclusion in the rings defined by (r, ρ) by means of a given set of parametrized radius. The resulting model is as follows:

$$
(\mathrm{QPR}_\Phi) \qquad
\begin{aligned}
& \mathrm{Find}\{(x_i, y_j)\} && \text{(A.19)} \\
& s.t. && \\
& \forall i \in V, x_i < L;\ y_i < L && \text{(A.20)} \\
& \forall \langle i,j \rangle \in \mathcal{E} : \bigvee_{\phi \in \Phi} f_\phi(x_i, y_i, x_j, y_j) && \text{(A.21)} \\
& \forall \langle i,j \rangle \in \overline{\mathcal{E}} : (|x_j - x_i| > r) \vee (|y_j - y_i| > r) && \text{(A.22)}
\end{aligned}
$$

where f_ϕ is a set of clauses enforcing the belonging of the point (x_j, y_j) to the radius defined by the angle ϕ intersecting the ring of center (x_i, y_i) defined by (r, ρ). For example:

$$
f_{\phi\in[0,\frac{\pi}{2}[} = (\rho\cos\phi \leq x_j - x_i) \wedge (x_j - x_i \leq r\cos\phi) \wedge (\rho\sin\phi \leq y_j - y_i) \wedge (y_j - y_i \leq r\sin\phi)
$$

Because of the "or" and "absolute value" terms, this formulation is still combinatorial, but much more efficient than the previous one.

This formulation was tested with Ibm Cplex solver. It provided 84 UD graphs to be implemented for UD-MIS search on the Rydberg atom based quantum machine, starting from 100 instances of real smart-charging graphs of 12 to 15 nodes each, limiting the (QPR_Φ) CSP resolution time to 60s for each instance.

This approach to reformulate (SC2) problems into UD-MIS tractable on Pasqal's quantum machine seems thus valid. However, in the experiments conducted up to now and presented here, we had to solve the (QP) problem this way only for graph instances whose number of vertices was limited. Tackling larger load scheduling problems on larger quantum processors with this method might prove vain, due to the cost of computing the corresponding UD graph of Rydberg atoms, assuming it exists.

Our method should thus be considered as a hybrid classical/quantum *heuristic* for solving (SC2)-like smart-charging problems –NP-Hard with no PTAS– by first trying to quickly find a UD representation of the graph, then aiming to exploit the expected quantum advantage gained when solving the UD-MIS problem on a Rydberg atom quantum device. Clearly, any improvement of the "mapping" between the original scheduling problem and its representation on the quantum processor will benefit to the method. This is an important research objective that we leave for future work.

A.3 Grid sampling with a general Hamiltonian

We provide here a bound on the minimum of samples required to achieve a certain error between the RFF model and the complete model in the case of a general encoding in the gird sampling strategy.

Theorem A.1. *Let $\mathcal{X}$ be a compact set of $\mathbb{R}^d$, and $\epsilon > 0$. We consider a training set $\{(x_i, y_i)\}_{i=1}^{M}$. Let f be a VQC model with any Hamiltonian encoding, with a maximum individual frequency $\omega_{\max}$ and full freedom on the associated frequency coefficients, trained with a regularisation λ. Let $\sigma_y^2 = \frac{1}{M}\sum_{i=1}^{M} y_i^2$ and $|\mathcal{X}|$ the diameter of $\mathcal{X}$. Let $\tilde{f}$ be the RFF model with D samples in the grid strategy trained on the same dataset and the same regularisation. Let $C = |f|_\infty |\mathcal{X}|$ and s the sampling rate defined in the grid sampling strategy. Then we can guarantee $|f(x) - \tilde{f}(x)| \leq \epsilon$ for $0 < s < \frac{1}{C}$ with probability $1 - \delta$ for a number D of samples given by:*

$$D = \Omega\left(\frac{dC_1}{\lambda^4(\epsilon - sC)^2}\left[log(\omega_{\max}|\mathcal{X}|) + log\frac{C_2}{\lambda^2(\epsilon - sC)} - log\delta\right]\right) \tag{A.23}$$

with C_1 and C_2 being constants depending on σ_y, $d(X)$. We recall that in Eq.A.23 the notation Ω stands for the computational complexity "Big-Ω" notation.

Proof. The following theorem bounds the approximation between a function defined by its Fourier series and another function with frequencies distant by at most a constant s of the original frequencies.

Let $\mathcal{X}$ a compact set of $\mathbb{R}^d$ with diameter $|\mathcal{X}|$ and Ω a finite subset of $\mathcal{X}$. Let $f(x) = \sum_{\omega\in\Omega} a_\omega \cos\left(\omega^T x\right) + b_\omega \sin\left(\omega^T x\right)$. Let Ω' a subset of $\mathcal{X}$ and $s > 0$ such that $\forall\omega \in \Omega$, $\exists\omega' \in \Omega$, $|\omega - \omega'| \leq s$.

Let $\mathcal{F}_{\Omega'} = \left\{\sum_{\omega\in\Omega'} a_\omega \cos\left(\omega^T x\right) + b_\omega sin(\omega^T x), a_\omega, b\omega \in \mathbb{R}\right\}$.

Theorem A.2. *It exists f' $\in \mathcal{F}_{\Omega'}$ such that*

$$\sup_{x\in\mathcal{X}} |f'(x) - f(x)| \leq sC \tag{A.24}$$

with $C = |\mathcal{X}||f|_\infty$.

Proof. For each $\omega \in \Omega$ let $b(\omega) \in \Omega'$ be such that $|\omega - b(\omega)| \leq s$. Such element exists by definition but is not necessarily unique. Let $f'(x) = \sum_{\omega\in\Omega} a_\omega \cos\Big(b(\omega)^T x\Big) + b_\omega \sin\Big(b(\omega)^T x\Big)$. The $b(\omega)$s are not necessarily different therefore there might be less frequencies in f' than in f.

$$|f(x) - f'(x)| = 2\Bigg|\sum_{\omega\in\Omega} \sin\left(\frac{(\omega - b(\omega))^T}{2}x\right) \tag{A.25}$$

$$[b_\omega \sin\left(\frac{(\omega + b(\omega))^T}{2}x\right) - a_\omega \cos\left(\frac{(\omega + b(\omega))^T}{2}x\right)]\Bigg| \tag{A.26}$$

$$\leq 2\sum_{\omega\in\Omega} |\frac{(\omega - b(\omega))^T}{2}||x|[|b_\omega| + |a_\omega|] \tag{A.27}$$

$$\leq s|x|\sum_{\omega\in\Omega} |b_\omega| + |a_\omega| \tag{A.28}$$

$$\leq s|\mathcal{X}||f|_\infty \tag{A.29}$$

□

We shall here extend the proof where we sample from the grid described above. Let us note $\hat{f}_s$ the RFF model with the whole grid and $\tilde{f}$ the RFF model with D samples from the grid below. For all $x \in \mathcal{X}$ we have

$$|\tilde{f}(x) - f(x)| \leq |\tilde{f}(x) - \hat{f}_s| + |\hat{f}_s - f(x)| \tag{A.30}$$

$$\leq |\tilde{f}(x) - \hat{f}_s| + sC \tag{A.31}$$

Then

$$\mathbb{P}(|\tilde{f}(x) - f(x)| \geq \epsilon) \leq \mathbb{P}(|\tilde{f}(x) - \hat{f}_s| \geq \epsilon - sC) \tag{A.32}$$

for $s < \epsilon/C$.

In this case $|\Omega| = (\omega_{\max}/s)^d$ Using the expression of Section **??**, we can guarantee that $|f(x) - \tilde{f}(x)| \leq \epsilon$ with probability $1 - \delta$ if

$$D = \Omega\left(d\frac{1}{(\epsilon - sC)^2}\left[\log(\omega_{\max}/s) + \log\frac{1}{\epsilon - sC} - \log\delta\right]\right) \tag{A.33}$$

□

A.4 PTAS for unit-disk graphs

The graphs that can be mapped onto a neutral-atom device are called $\lambda-$*precision unit-disk graphs.*

Definition A.1 ([HIMR+98])**.** *For any fixed* $\lambda > 0$*, consider a finite set of unit disks in the plane where the centers of any two disks are at least* λ *apart. A* $\lambda-$*precision unit-disk graph* $G(V, E)$ *is a graph where each vertex of* G *is in one-to-one correspondence with the set of unit-disks and two vertices are joined by an edge i.f.f. the corresponding disks intersect.*

In the case of a neutral-atom quantum processor, λ corresponds to the minimal distance between atoms, of the order of $5\mu m$.

We describe the algorithm presented by Hunt [HIMR+98] that guarantees an approximate solution to the MIS problem on a λ-UD graph G for which an embedding in $\mathbb{R}^2$ is given.

The quality of the approximation ratio is guaranteed thanks to the *shifting technique* [HM85]. This general approach uses a set of evenly spaced separators to decompose the problem into smaller, more manageable sub-problems whose solutions are merged to obtain a solution to the global problem. The process is repeated with different placements of the separator set, and the best solution across these placements is selected as an approximation of the optimal solution. The term “shifting technique" arises from the concept of moving the separator set through the problem. The key idea of the technique is represented in Figure A.1.

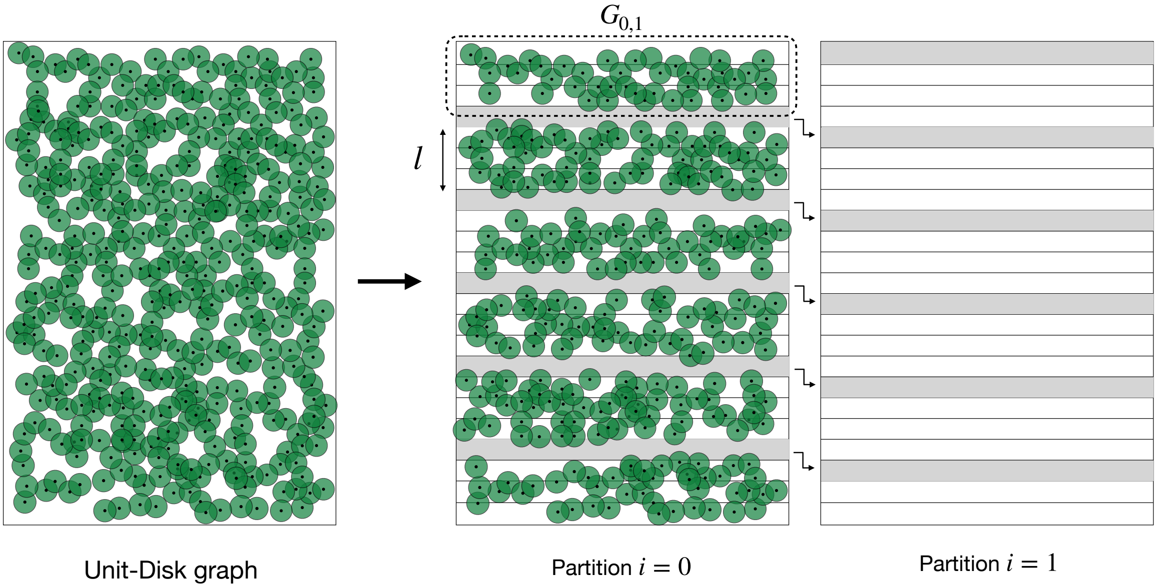


Figure A.1: **Shifting technique for UD-MIS**. Let G be a $\lambda-$UD graph where two nodes u, v are connected if $\|u, v\| \leq 1$ and all nodes are separated by a minimal distance $\lambda > 0$. The area I in which G is embedded is decomposed in horizontal strips of width 1. Given a shifting parameter l, G is partitioned in independent subgraphs of width l. They are all made independent by deleting the nodes in the grey strips. The MIS is solved exactly on the individual subgraphs and the IS of the whole graph is the union of all IS sub-graphs. Because of the independence of the sub-graphs, such a union of IS is a feasible solution on the global problem. We can then create another partition of G by moving all grey strips one level down. After l partitions, we recover the partition $i = 0$. From the l possible partitions, the best solution is chosen and has a guaranteed approximation ratio of $(l/(l+1)) \cdot |OPT(G)|$ nodes, where $OPT(G)$ is the optimal solution. This guarantee comes from the shifting lemma.

We now describe the algorithm in pseudo-code. Let $OPT(G)$ denote (one of) the optimal solution, and let $IS(G)$ be the output solution of the algorithm:

This algorithm guarantees a solution $IS(G)$ such that $IS(G) \geq (\frac{l}{l+1}) \cdot |OPT(G)|$. This is given by the following lemma:

Lemma A.1 (**Shifting technique**).

$$\max_{0 \leq i \leq l} |OPT(G_i)| \geq \frac{l}{l+1} |OPT(G)|. \tag{A.34}$$

Proof. For each partition $i, 0 \leq i \leq l$, let S_i be the set of unit disks which were **not** considered in iteration i. Given the optimal independent set $OPT(G)$, we denote $IS_{\text{opt}}(S_i) = \{v \in V / v \in OPT(G) \cap S_i\}$ the vertices of the optimal set that are in S_i. The family $\{S_0, \cdots, S_l\}$ forms a partition of the graph and therefore

Algorithm 8 PTAS for λ-UD MIS

```
Require: $l \in \mathbb{N}$
Ensure: $IS(G) \geq (\frac{l}{l+1}) \cdot |OPT(G)|$
Divide the plane into horizontal strips of width 1.
for $i$ in $0, \cdots, l$ do                                                  ▷ Shifting technique
    Partition the disks into $r$ disjoint sets $G_{i,1}, \ldots, G_{i,r}$ by removing all disks in every horizontal strip congruent to $i \mod (l+1)$.
    $G_i = \cup_{1 \leq j \leq r} G_{i,j}$ and $\forall j, k, G_{i,j} \cap G_{i,k} = \emptyset$
    for $j$ in $1, \cdots, r$ do
        Solve MIS exactly and obtain $OPT(G_{i,j})$.                          ▷ See theorem.
    end for
    $OPT(G_i) = \cup_{1 \leq j \leq r} OPT(G_{i,j})$
end for
$IS(G) = \max_{0 \leq i \leq l} OPT(G_i)$
```

$$\sum_{i=0}^{l} |IS_{\text{opt}}(S_i)| = |OPT(G)|, \tag{A.35}$$

as $S_i \cap S_j = \emptyset$ for any $i \neq j$. Therefore,

$$\min_{0 \leq t \leq l} |IS_{\text{opt}}(S_t)| \leq \frac{1}{l+1} \cdot |OPT(G)|. \tag{A.36}$$

Also, note that

$$\max_{0 \leq i \leq l} |OPT(G_i)| \geq |OPT(G)| - \min_{0 \leq t \leq l} |IS_{\text{opt}}(S_t)|, \tag{A.37}$$

and using the upper bound of equation (A.36) we obtain

$$\max_{0 \leq i \leq l} |OPT(G_i)| \geq |OPT(G)| - \frac{1}{l+1} \cdot |OPT(G)| \tag{A.38}$$

$$\geq \frac{l}{l+1} \cdot |OPT(G)|. \tag{A.39}$$

□

The shifting lemma bounds the error of the shifting strategy and therefore offers a guarantee of the approximation ratio that depends on the integer $l \in \mathbb{N}$. In the algorithm, step 6 requires to solve MIS exactly on the subgraph $G_{i,j}$. Note that when looking at $G_{i,j}$ over a rectangular area of $r_b \times lr_b$, where r_b is the Rydberg blockade radius. Then, because the vertices of a $\lambda-$UD graph cannot be closer than λ, the maximum number of vertices inside of the area is $l(\frac{r_b}{s})^2$. This ensures that the tree decomposition of $G_{i,j}$ has a treewidth of size $l(\frac{r_b}{s})^2$ (see proof of Theorem 5.1 in [HIMR⁺98]), taken as $\mathcal{O}(l)$ as r_b and s are fixed. In this case, it is possible to find exactly the MIS of a graph with bounded treewidth in time $\mathcal{O}(2^l n)$, which is exponential with the precision parameter (as $\varepsilon = l/(l+1)$) but linear in n.

To conclude this section, we have detailed the workings of a PTAS for solving MIS on $\lambda-$UD graphs. The shifting technique is the core strategy of this approximation algorithm that guarantees a lower-bound on the quality of the solution returned. To the best of my knowledge, a quantum version of PTAS has not been formalised yet. An immediate approach would be to build the oracle that returns the quality of a partition i and therefore get a quadratic speed-up *à la* Grover to find the best partition from all possible partitions.

BIBLIOGRAPHY

[AAB$^+$19] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, et al.
Quantum supremacy using a programmable superconducting processor.
Nature, 574(7779):505–510, 2019.
(page 6)

[Aar15] S. Aaronson.
Read the fine print.
Nature Physics, 11(4):291–293, 2015.
(page 81)

[AB09] S. Arora and B. Barak.
Computational complexity: a modern approach.
Cambridge University Press, 2009.
(page 10)

[ABC$^+$20] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al.
Learning dexterous in-hand manipulation.
The International Journal of Robotics Research, 39(1):3–20, 2020.
(page 80)

[AC09] M. H. Amin and V. Choi.
First-order quantum phase transition in adiabatic quantum computation.
Physical Review A, 80(6):062326, 2009.
(pages 63, 64, and 131)

[ADBL19] J. M. Arrazola, A. Delgado, B. R. Bardhan, and S. Lloyd.
Quantum-inspired algorithms in practice.
arXiv preprint arXiv:1905.10415, 2019.
(page 81)

[ADL$^+$23] B. Albrecht, C. Dalyac, L. Leclerc, L. Ortiz-Gutiérrez, S. Thabet, M. D'Arcangelo, J. R. Cline, V. E. Elfving, L. Lassablière, H. Silvério, et al.
Quantum feature maps for graph machine learning on a neutral atom quantum processor.
Physical Review A, 107(4):042615, 2023.
(pages 2 and 84)

[AH23] S. Aaronson and S.-H. Hung.
Certified randomness from quantum supremacy.
arXiv preprint arXiv:2303.01625, 2023.
(page 6)

[AKR10] B. Altshuler, H. Krovi, and J. Roland.
Anderson localization makes adiabatic quantum optimization fail.
Proceedings of the National Academy of Sciences, 107(28):12446–12450, 2010.
(pages 63, 64, 65, and 131)

[AL18] T. Albash and D. A. Lidar.

Adiabatic quantum computation.
Reviews of Modern Physics, 90(1):015002, 2018.
(page 56)

[Ami09] M. H. Amin.
Consistency of the adiabatic theorem.
Physical review letters, 102(22):220401, 2009.
(page 56)

[And58] P. W. Anderson.
Absence of diffusion in certain random lattices.
Physical review, 109(5):1492, 1958.
(pages 9 and 53)

[ato] Quantum learning machine.
URL `https://atos.net/en/solutions/quantum-learning-machine`.
(page 27)

[Bak94] B. S. Baker.
Approximation algorithms for np-complete problems on planar graphs.
J. ACM, 41(1):153–180, jan 1994.
doi:10.1145/174644.174650.
(page 68)

[BBCV21] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković.
Geometric deep learning: Grids, groups, graphs, geodesics, and gauges.
arXiv preprint arXiv:2104.13478, 2021.
(page 86)

[BBL16a] A. Browaeys, D. Barredo, and T. Lahaye.
Experimental investigations of dipole–dipole interactions between a few rydberg atoms.
Journal of Physics B: Atomic, Molecular and Optical Physics, 49(15):152001, 2016.
(page 7)

[BBL16b] A. Browaeys, D. Barredo, and T. Lahaye.
Experimental investigations of dipole–dipole interactions between a few Rydberg atoms.
Journal of Physics B: Atomic, Molecular and Optical Physics, 49(15):152001, Aug. 2016.
doi:10.1088/0953-4075/49/15/152001.
(page 72)

[BCM11] S. Bhagat, G. Cormode, and S. Muthukrishnan.
Node Classification in Social Networks, pages 115–148.
Springer US, Boston, MA, 2011.
doi:10.1007/978-1-4419-8462-3_5.
(page 83)

[BDTD$^+$16] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al.
End to end learning for self-driving cars.
arXiv preprint arXiv:1604.07316, 2016.
(page 80)

[Bel17] Belib réseau parisien de bornes de recharges pour véhicules électriques.
`www.data.gouv.fr/fr/datasets/belib-reseau-parisien-de-bornes-de-recharges-accelerees-22-kw-ac-dc-pour-vehicules-`
2017.
(page 27)

[BF94] P. Berman and M. Fürer.
Approximating maximum independent set in bounded degree graphs.
In *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA ’94, page 365–371, USA, 1994. Society for Industrial and Applied Mathematics.
(page 68)

[BFK+13] V. Bapst, L. Foini, F. Krzakala, G. Semerjian, and F. Zamponi.
The quantum adiabatic algorithm applied to random optimization problems: The quantum spin glass perspective.
Physics Reports, 523(3):127–205, 2013.
(pages 53, 56, 64, and 65)

[BGLS06] J.-F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal.
Numerical optimization: theoretical and practical aspects.
Springer Science & Business Media, 2006.
(page 36)

[BH91] E. Boros and P. L. Hammer.
The max-cut problem and quadratic 0–1 optimization; polyhedral aspects, relaxations and bounds.
Annals of Operations Research, 33(3):151–180, 1991.
(page 11)

[BH13] L. Bai and E. R. Hancock.
Graph kernels from the jensen-shannon divergence.
Journal of mathematical imaging and vision, 47(1):60–69, 2013.
(page 88)

[BJSW02] T. Biedl, J. R. Johansen, T. Shermer, and D. R. Wood.
Orthogonal drawings with few layers.
In *PROC. 9TH INTERNATIONAL SYMP. ON GRAPH DRAWING (GD '01*, pages 297–311. Springer, 2002.
(page 72)

[BKKT20] S. Bravyi, A. Kliesch, R. Koenig, and E. Tang.
Hybrid quantum-classical algorithms for approximate graph coloring.
arXiv e-prints, page arXiv:2011.13420, Nov. 2020, 2011.13420.
(page 18)

[BL98] B. Berger and T. Leighton.
Protein folding in the hydrophobic-hydrophilic (hp) is np-complete.
In *Proceedings of the second annual international conference on Computational molecular biology*, pages 30–39, 1998.
(page 11)

[BL20] A. Browaeys and T. Lahaye.
Many-body physics with individually controlled rydberg atoms.
Nature Physics, 16(2):132–142, Feb 2020.
doi:10.1038/s41567-019-0733-z.
(page 8)

[BLL+18] D. Barredo, V. Lienhard, S. d. Léséleuc, T. Lahaye, and A. Browaeys.
Synthetic three-dimensional atomic structures assembled atom by atom.
Nature, 561(7721):79–82, Sept. 2018.
doi:10.1038/s41586-018-0450-2.
(page 10)

[BLS+21] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang, S. Ebadi, M. Kalinowski, A. Keesling, N. Maskara, H. Pichler, M. Greiner, V. Vuletic, and M. D. Lukin.
A quantum processor based on coherent transport of entangled atom arrays.
arXiv e-prints, page arXiv:2112.03923, Dec. 2021, 2112.03923.
(page 110)

[BLS+22] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang, S. Ebadi, M. Kalinowski, A. Keesling, N. Maskara, H. Pichler, M. Greiner, et al.
A quantum processor based on coherent transport of entangled atom arrays.
Nature, 604(7906):451–456, 2022.
(page 9)

[BM07] R. J. Bartlett and M. Musiał.
Coupled-cluster theory in quantum chemistry.
Reviews of Modern Physics, 79(1):291, 2007.
(page 111)

[BMW00] G. Biroli, R. Monasson, and M. Weigt.
A variational description of the ground state structure in random satisfiability problems.
The European Physical Journal B-Condensed Matter and Complex Systems, 14(3):551–568, 2000.
(page 65)

[BN06] C. M. Bishop and N. M. Nasrabadi.
Pattern recognition and machine learning, volume 4.
Springer, 2006.
(page 117)

[BPP21] L. Banchi, J. Pereira, and S. Pirandola.
Generalization in quantum machine learning: A quantum information standpoint.
PRX Quantum, 2(4):040321, 2021.
(page 112)

[BSA08] A. Billionnet, E. S., and L. A.
Linear reformulations of integer quadratic programs.
Modelling, Computation and Optimization in Information Systems and Management Sciences, 14, 2008.
(pages 133 and 136)

[BUBVO20] L. Boon, E. Ugarte-Berzal, J. Vandooren, and G. Opdenakker.
Protease propeptide structures, mechanisms of activation, and functions.
Critical Reviews in Biochemistry and Molecular Biology, 55(2):111–165, 2020, https://doi.org/10.1080/10409238.2020.1742090.
doi:10.1080/10409238.2020.1742090.
PMID: 32290726.
(page 99)

[BVC$^+$13] L. Beguin, A. Vernier, R. Chicireanu, T. Lahaye, and A. Browaeys.
Direct measurement of the van der waals interaction between two rydberg atoms.
Physical review letters, 110(26):263201, 2013.
(pages 6 and 89)

[C.04] M. D. J. C.
An Example Inference Task: Clustering.
Cambridge University press, 2004.
(page 36)

[CAB$^+$20] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles.
Variational Quantum Algorithms.
arXiv e-prints, page arXiv:2012.09265, Dec. 2020, 2012.09265.
(page 21)

[CAB$^+$21] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, et al.
Variational quantum algorithms.
Nature Reviews Physics, 3(9):625–644, 2021.
(page 24)

[CAS16] P. Covington, J. Adams, and E. Sargin.
Deep neural networks for youtube recommendations.
In *Proceedings of the 10th ACM conference on recommender systems*, pages 191–198, 2016.
(page 80)

[CBB$^+$23] C. Chen, G. Bornet, M. Bintz, G. Emperauger, L. Leclerc, V. S. Liu, P. Scholl, D. Barredo, J. Hauschild, S. Chatterjee, et al.

Continuous symmetry breaking in a two-dimensional rydberg array.
Nature, 616(7958):691–695, 2023.
(page 76)

[CCJ90] B. N. Clark, C. J. Colbourn, and D. S. Johnson.
Unit disk graphs.
Discrete Mathematics, 86(1):165–177, 1990.
doi:https://doi.org/10.1016/0012-365X(90)90358-O.
(page 65)

[CCL19a] I. Cong, S. Choi, and M. D. Lukin.
Quantum convolutional neural networks.
Nature Physics, 15(12):1273–1278, 2019.
(page 81)

[CCL19b] I. Cong, S. Choi, and M. D. Lukin.
Quantum convolutional neural networks.
Nature Physics, 15(12):1273–1278, Dec 2019.
doi:10.1038/s41567-019-0648-8.
(page 84)

[CCL$^+$23] M. Cain, S. Chattopadhyay, J.-G. Liu, R. Samajdar, H. Pichler, and M. D. Lukin.
Quantum speedup for combinatorial optimization with flat energy landscapes, 2023, 2306.13123.
(pages 64 and 131)

[CCZ09] M. Conforti, G. Cornuéjols, and G. Zambelli.
Polyhedral approaches to mixed integer linear programming.
In *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*, pages 343–385. Springer, 2009.
(page 65)

[CGFM$^+$21a] M. C. Caro, E. Gil-Fuster, J. J. Meyer, J. Eisert, and R. Sweke.
Encoding-dependent generalization bounds for parametrized quantum circuits.
Quantum, 5:582, 2021.
(page 112)

[CGFM$^+$21b] M. C. Caro, E. Gil-Fuster, J. J. Meyer, J. Eisert, and R. Sweke.
Encoding-dependent generalization bounds for parametrized quantum circuits.
Quantum, 5:582, Nov. 2021.
doi:10.22331/q-2021-11-17-582.
(page 114)

[CHC$^+$22] M. C. Caro, H.-Y. Huang, M. Cerezo, K. Sharma, A. Sornborger, L. Cincio, and P. J. Coles.
Generalization in quantum machine learning from few training data.
Nature communications, 13(1):4919, 2022.
(page 112)

[CHQ$^+$20] M. Cao, F. Hoffet, S. Qiu, A. S. Sheremet, and J. Laurat.
Efficient reversible entanglement transfer between light and quantum memories.
Optica, 7(10):1440–1444, 2020.
(page 132)

[CKM19] E. Campbell, A. Khurana, and A. Montanaro.
Applying quantum algorithms to constraint satisfaction problems.
Quantum, 3:167, 2019.
(page 76)

[CL21] E. Crosson and D. Lidar.
Prospects for quantum enhancement with diabatic quantum annealing.
Nature Reviews Physics, 3(7):466–489, 2021.
(page 76)

[Cob65] A. Cobham.
The intrinsic computational difficulty of functions.
In Y. Bar-Hillel, editor, *Logic, Methodology and Philosophy of Science: Proceedings of the 1964 International Congress (Studies in Logic and the Foundations of Mathematics)*, pages 24–30. North-Holland Publishing, 1965.
(page 51)

[COE15] A. Coja-Oghlan and C. Efthymiou.
On independent sets in random graphs.
Random Structures & Algorithms, 47(3):436–486, 2015.
(pages 52, 53, and 55)

[Col13] E. Collini.
Spectroscopic signatures of quantum-coherent energy transfer.
Chemical Society Reviews, 42(12):4932–4947, 2013.
(page 8)

[Coo23] S. A. Cook.
The complexity of theorem-proving procedures.
In *Logic, Automata, and Computational Complexity: The Works of Stephen A. Cook*, pages 143–152. 2023.
(page 52)

[CV86] R. Cole and U. Vishkin.
Deterministic coin tossing with applications to optimal parallel list ranking.
Information and Control, 70(1):32–53, 1986.
(page 52)

[D$^+$59] E. W. Dijkstra et al.
A note on two problems in connexion with graphs.
Numerische mathematik, 1(1):269–271, 1959.
(page 72)

[DCLT19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova.
Bert: Pre-training of deep bidirectional transformers for language understanding, 2019, 1810.04805.
(page 80)

[DHJ$^+$21] C. Dalyac, L. Henriet, E. Jeandel, W. Lechner, S. Perdrix, M. Porcheron, and M. Veshchezerova.
Qualifying quantum approaches for hard industrial optimization problems. a case study in the field of smart-charging of electric vehicles.
EPJ Quantum Technology, 8(1):12, 2021.
(pages 2 and 14)

[DHK$^+$23] C. Dalyac, L.-P. Henry, M. Kim, J. Ahn, and L. Henriet.
Exploring the impact of graph locality for the resolution of mis with neutral atom devices, 2023, 2306.13373.
(pages 2 and 50)

[DHS11] J. Duchi, E. Hazan, and Y. Singer.
Adaptive subgradient methods for online learning and stochastic optimization.
Journal of machine learning research, 12(7), 2011.
(page 79)

[Dic11] N. G. Dickson.
Elimination of perturbative crossings in adiabatic quantum optimization.
New Journal of Physics, 13(7):073011, 2011.
(page 64)

[Dij59] E. W. Dijkstra.
A note on two problems in connexion with graphs.
Numerische Mathematik, 1(1):269–271, Dec 1959.
doi:10.1007/BF01386390.
(page 95)

[dKPW04] E. de Klerk, D. V. Pasechnik, and J. P. Warners.
On approximate graph colouring and max-k-cut algorithms based on the θ-function.
Journal of Combinatorial Optimization, 8(3):267–294, 2004.
(page 30)

[DLBL+18] S. De Léséleuc, D. Barredo, V. Lienhard, A. Browaeys, and T. Lahaye.
Analysis of imperfections in the coherent optical excitation of single atoms to rydberg states.
Physical Review A, 97(5):053803, 2018.
(pages 39 and 89)

[DPCB23] A. M. Dalzell, N. Pancotti, E. T. Campbell, and F. G. Brandão.
Mind the gap: Achieving a super-grover quantum speedup by jumping to the end.
In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1131–1144, 2023.
(page 76)

[DZAG21] M. Dupret, J.-P. Zimmermann, N. Andreau, and M. Guernevel.
Panel usages electrodomestiques - consommations électrodomestiques françaises basées sur des mesures collectées en continu dans 100 logements, Mar 2021.
URL https://librairie.ademe.fr/cadic/5443/panel-usages-electrodomestiques-2021_rapport-condense.pdf.
(page 15)

[Dzi10] J. Dziarmaga.
Dynamics of a quantum phase transition and relaxation to a steady state.
Advances in Physics, 59(6):1063–1189, 2010.
(page 131)

[EHJ+10] T. Erlebach, T. Hagerup, K. Jansen, M. Minzlaff, and A. Wolff.
Trimming of graphs, with application to point labeling.
Theory of Computing Systems, 47:613–636, 2010.
(page 65)

[Ehr08] T. Ehret.
Old brass brains - mechanical prediction of tides.
ACSM Bulletin, pages 41–44, 2008.
URL https://web.archive.org/web/20110728172441/http://www.webmazine.org/issues/bull233/documents/oldBrassBrains.pdf.
(page 5)

[EKC+22] S. Ebadi, A. Keesling, M. Cain, T. T. Wang, H. Levine, D. Bluvstein, G. Semeghini, A. Omran, J.-G. Liu, R. Samajdar, X.-Z. Luo, B. Nash, X. Gao, B. Barak, E. Farhi, S. Sachdev, N. Gemelke, L. Zhou, S. Choi, H. Pichler, S.-T. Wang, M. Greiner, V. Vuletic, and M. D. Lukin.
Quantum optimization of maximum independent set using rydberg atom arrays.
Science, 0(0):eabo6587, 2022, https://www.science.org/doi/pdf/10.1126/science.abo6587.
doi:10.1126/science.abo6587.
(pages 48 and 76)

[EL19] S. Elloumi and A. Lambert.
Global solution of non-convex quadratically constrained programs.
Optimization Methods and Software, Taylor & Francis, 34(1):98–114, 2019.
(pages 133 and 136)

[EMKZ18] M. Erhard, M. Malik, M. Krenn, and A. Zeilinger.
Experimental greenberger–horne–zeilinger entanglement beyond qubits.
Nature Photonics, 12(12):759–764, 2018.
(page 80)

[EMW20] D. J. Egger, J. Marecek, and S. Woerner.
Warm-starting quantum optimization, 2020, 2009.10095.
(page 25)

[ESW96] P. Eades, C. Stirk, and S. Whitesides.
The techniques of komolgorov and bardzin for three-dimensional orthogonal graph drawings.
Information Processing Letters, 60(2):97–103, 1996.
doi:https://doi.org/10.1016/S0020-0190(96)00133-0.
(pages 72 and 74)

[Eul24] L. Euler.
Institutionum calculi integralis, volume 1.
impensis Academiae imperialis scientiarum, 1824.
(page 4)

[Fey82] R. P. Feynman.
Simulating physics with computers.
International Journal of Theoretical Physics, 21(6-7):467–488, jun 1982.
doi:10.1007/bf02650179.
(page 6)

[FGG+09] E. Farhi, J. Goldstone, D. Gosset, S. Gutmann, H. B. Meyer, and P. Shor.
Quantum adiabatic algorithms, small gaps, and different paths.
arXiv preprint arXiv:0909.4766, 2009.
(page 57)

[FGG14] E. Farhi, J. Goldstone, and S. Gutmann.
A quantum approximate optimization algorithm.
arXiv:1411.4028, 2014.
(pages 14, 21, 22, and 32)

[FGGS00a] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser.
Quantum computation by adiabatic evolution.
arXiv:quant-ph/0001106., 2000.
(page 57)

[FGGS00b] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser.
Quantum computation by adiabatic evolution, 2000.
doi:10.48550/ARXIV.QUANT-PH/0001106.
(page 63)

[Flo62] R. W. Floyd.
Algorithm 97: Shortest path.
Commun. ACM, 5(6):345, jun 1962.
doi:10.1145/367766.368168.
(page 95)

[FR91a] T. M. J. Fruchterman and E. M. Reingold.
Graph drawing by force-directed placement.
Software: Practice and Experience, 21(11):1129–1164, 1991, https://onlinelibrary.wiley.com/doi/pdf/10.1002/spe.4380211102.
doi:https://doi.org/10.1002/spe.4380211102.
(page 100)

[FR91b] T. M. Fruchterman and E. M. Reingold.
Graph drawing by force-directed placement.
Software: Practice and experience, 21(11):1129–1164, 1991.
(page 72)

[Fra18] P. I. Frazier.
A tutorial on bayesian optimization.
arXiv preprint arXiv:1807.02811, 2018.
(page 44)

[Fre00] L. C. Freeman.
Visualizing social networks.
J. Soc. Struct., 1, 2000.

(page 83)

[Fri90] A. M. Frieze.
On the independence number of random graphs.
Discrete Mathematics, 81(2):171–175, 1990.
(page 52)

[GF18] P. Goyal and E. Ferrara.
Graph embedding techniques, applications, and performance: A survey.
Knowledge-Based Systems, 151:78–94, 2018.
(page 83)

[GFW03] T. Gärtner, P. Flach, and S. Wrobel.
On graph kernels: Hardness results and efficient alternatives.
In B. Schölkopf and M. K. Warmuth, editors, *Learning Theory and Kernel Machines*, pages 129–143, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
(page 94)

[GH20] R. Guerraoui and L. N. Hoang.
Turing à la plage: L'intelligence artificielle dans un Transat.
Dunod, 2020.
(pages 5 and 51)

[Gil61] E. N. Gilbert.
Random plane networks.
Journal of the society for industrial and applied mathematics, 9(4):533–543, 1961.
(page 65)

[GJ78] M. R. Garey and D. S. Johnson.
“strong”np-completeness results: Motivation, examples, and implications.
Journal of the ACM (JACM), 25(3):499–508, 1978.
(pages 11 and 52)

[GK12a] S. Gharibian and J. Kempe.
Approximation algorithms for qma-complete problems.
SIAM Journal on Computing, 41(4):1028–1050, 2012.
(page 131)

[GK12b] S. Gharibian and J. Kempe.
Hardness of approximation for quantum problems.
In *International Colloquium on Automata, Languages, and Programming*, pages 387–398. Springer, 2012.
(page 131)

[GLLRK79] R. L. Graham, E. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan.
Optimization and approximation in deterministic sequencing and scheduling: A survey.
Annals of Discrete Mathematics, 5:287–326, 1979.
(page 16)

[GM75] G. R. Grimmett and C. J. McDiarmid.
On colouring random graphs.
In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 77, pages 313–324. Cambridge University Press, 1975.
(page 52)

[GMW19] P. Gill, W. Murray, and M. Wright.
Practical Optimization.
Classics in Applied Mathematics. SIAM, Society for Industrial and Applied Mathematics, 2019.
URL `https://books.google.fr/books?id=G-vEDwAAQBAJ`.
(page 25)

[GOS22] T. Graham, E. Oh, and M. Saffman.
Multi-scale architecture for fast optical addressing and control of large scale qubit arrays.

arXiv preprint arXiv:2212.12428, 2022.
(pages 7 and 9)

[GPC18] I. Glasser, N. Pancotti, and J. I. Cirac.
Supervised learning with generalized tensor networks.
arXiv preprint arXiv:1806.05964, page 50, 2018.
(page 81)

[Gro96] L. K. Grover.
A fast quantum mechanical algorithm for database search.
In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.
doi:10.1145/237814.237866.
(page 76)

[GS17] G. G. Guerreschi and M. Smelyanskiy.
Practical optimization for hybrid quantum-classical algorithms, 2017, 1701.01450.
(page 25)

[GSR$^+$17] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl.
Neural message passing for quantum chemistry.
In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR, 06–11 Aug 2017.
URL `https://proceedings.mlr.press/v70/gilmer17a.html`.
(pages 86 and 110)

[GSS21] C. Gschwendtner, S. R. Sinsel, and A. Stephan.
Vehicle-to-x (v2x) implementation: An overview of predominate trial configurations and technical, social and regulatory challenges.
Renewable and Sustainable Energy Reviews, 145:110977, 2021.
(page 15)

[Hås99] J. Håstad.
Clique is hard to approximate.
Acta Mathematica, 182(1):105–142, 1999.
(page 70)

[Has12] M. B. Hastings.
Trivial low energy states for commuting hamiltonians, and the quantum pcp conjecture.
arXiv preprint arXiv:1201.3387, 2012.
(page 131)

[Has19] M. B. Hastings.
Classical and quantum bounded depth approximation algorithms, 2019, 1905.07047.
(page 28)

[HBM$^+$21a] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean.
Power of data in quantum machine learning.
Nature Communications, 12(1):2631, May 2021.
doi:10.1038/s41467-021-22539-9.
(page 105)

[HBM$^+$21b] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean.
Power of data in quantum machine learning.
Nature communications, 12(1):1–9, 2021.
(page 112)

[HBS$^+$20] L. Henriet, L. Beguin, A. Signoles, T. Lahaye, A. Browaeys, G.-O. Reymond, and C. Jurczak.
Quantum computing with neutral atoms.
Quantum, 4:327, 2020.
(page 9)

[HCT+19] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta.
Supervised learning with quantum-enhanced feature spaces.
Nature, 567(7747):209–212, 2019.
doi:10.1038/s41586-019-0980-2.
(page 84)

[Hen13] L.-P. Henry.
Classical and quantum two-dimensional ice: Coulomb and ordered phases.
PhD thesis, Ecole normale supérieure de lyon-ENS LYON, 2013.
(page 9)

[HGK10] Y. Hu, E. R. Gansner, and S. Kobourov.
Visualizing graphs and clusters as maps.
IEEE Computer Graphics and Applications, 30(6):54–66, 2010.
(page 72)

[HHL09] A. W. Harrow, A. Hassidim, and S. Lloyd.
Quantum algorithm for linear systems of equations.
Physical review letters, 103(15):150502, 2009.
(page 81)

[HIMR+98] H. B. Hunt III, M. V. Marathe, V. Radhakrishnan, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns.
Nc-approximation schemes for np-and pspace-hard problems for geometric graphs.
Journal of algorithms, 26(2):238–274, 1998.
(pages 139 and 141)

[HKKS01] C. Helma, R. D. King, S. Kramer, and A. Srinivasan.
The Predictive Toxicology Challenge 2000–2001 .
Bioinformatics, 17(1):107–108, 01 2001, https://academic.oup.com/bioinformatics/article-pdf/17/1/107/576297/170107.pdf.
doi:10.1093/bioinformatics/17.1.107.
(page 98)

[HKP20] H.-Y. Huang, R. Kueng, and J. Preskill.
Predicting many properties of a quantum system from very few measurements.
Nature Physics, 16(10):1050–1057, 2020.
(page 81)

[HKT+21] H.-Y. Huang, R. Kueng, G. Torlai, V. V. Albert, and J. Preskill.
Provably efficient machine learning for quantum many-body problems.
arXiv preprint arXiv:2106.12627, 2021.
(page 81)

[HL11] M. Hilbert and P. López.
The world's technological capacity to store, communicate, and compute information.
Science, 332(6025):60–65, 2011, https://www.science.org/doi/pdf/10.1126/science.1200970.
doi:10.1126/science.1200970.
(page 115)

[HM85] D. S. Hochbaum and W. Maass.
Approximation schemes for covering and packing problems in image processing and vlsi.
Journal of the ACM (JACM), 32(1):130–136, 1985.
(pages 65 and 139)

[HNR68] P. E. Hart, N. J. Nilsson, and B. Raphael.
A formal basis for the heuristic determination of minimum cost paths.
IEEE Transactions on Systems Science and Cybernetics, 4(2):100–107, 1968.
doi:10.1109/TSSC.1968.300136.
(page 72)

[HR97] M. M. Halldórsson and J. Radhakrishnan.
Greed is good: Approximating independent sets in sparse and bounded-degree graphs.

Algorithmica, 18(1):145–163, 1997.
doi:10.1007/BF02523693.
(page 70)

[HSCC22] Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles.
Connecting ansatz expressibility to gradient magnitudes and barren plateaus.
PRX Quantum, 3(1):010313, 2022.
(page 112)

[HTDH21] L.-P. Henry, S. Thabet, C. Dalyac, and L. Henriet.
Quantum evolution kernel: Machine learning on graphs with programmable arrays of qubits.
Phys. Rev. A, 104:032416, Sep 2021.
doi:10.1103/PhysRevA.104.032416.
(pages 2, 97, and 108)

[HWO+17] S. Hadfield, Z. Wang, B. O'Gorman, E. G. Rieffel, D. Venturelli, and R. Biswas.
From the quantum approximate optimization algorithm to a quantum alternating operator ansatz.
arXiv e-prints, page arXiv:1709.03489, 9 2017.
(page 22)

[JEP+21] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, et al.
Highly accurate protein structure prediction with alphafold.
Nature, 596(7873):583–589, 2021.
(page 80)

[JNN13] J. Johansson, P. Nation, and F. Nori.
Qutip 2: A python framework for the dynamics of open quantum systems.
Computer Physics Communications, 184(4):1234–1240, Apr 2013.
doi:10.1016/j.cpc.2012.11.019.
(page 27)

[KBH+21] D. Kreuzer, D. Beaini, W. Hamilton, V. Létourneau, and P. Tossou.
Rethinking graph transformers with spectral attention.
Advances in Neural Information Processing Systems, 34:21618–21629, 2021.
(page 110)

[KGAR04] G. A. Kochenberger, F. Glover, B. Alidaee, and C. Rego.
A unified modeling and solution framework for combinatorial optimization problems.
Or Spectrum, 26(2):237–250, 2004.
(page 11)

[KGJV83] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi.
Optimization by simulated annealing.
science, 220(4598):671–680, 1983.
(page 11)

[KJM20] N. M. Kriege, F. D. Johansson, and C. Morris.
A survey on graph kernels.
Applied Network Science, 5(6), 2020.
doi:10.1007/s41109-019-0195-3.
(page 100)

[KKH+21] M. Kim, K. Kim, J. Hwang, E.-G. Moon, and J. Ahn.
Rydberg quantum wires for maximum independent set problems with nonplanar and high-degree graphs, 2021.
doi:10.48550/ARXIV.2109.03517.
(page 71)

[KLM21] I. Kerenidis, J. Landman, and N. Mathur.
Classical and quantum algorithms for orthogonal neural networks.
arXiv preprint arXiv:2106.07198, 2021.

(page 113)

[KLPS07] A. W. Kolen, J. K. Lenstra, C. H. Papadimitriou, and F. C. Spieksma.
Interval scheduling: A survey.
`www.interscience.wiley.com`, 2007.
(page 19)

[KMF+16] M. Krenn, M. Malik, R. Fickler, R. Lapkiewicz, and A. Zeilinger.
Automated search for new quantum experiments.
Phys. Rev. Lett., 116:090405, Mar 2016.
doi:10.1103/PhysRevLett.116.090405.
(page 80)

[KMT+17] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta.
Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets.
nature, 549(7671):242–246, 2017.
(page 111)

[KPE21] O. Kyriienko, A. E. Paine, and V. E. Elfving.
Solving nonlinear differential equations with differentiable quantum circuits.
Physical Review A, 103(5):052416, 2021.
(pages 113 and 129)

[KRL+23] A. D. King, J. Raymond, T. Lanting, R. Harris, A. Zucca, F. Altomare, A. J. Berkley, K. Boothby, S. Ejtemaee, C. Enderud, et al.
Quantum critical dynamics in a 5,000-qubit programmable spin glass.
Nature, pages 1–6, 2023.
(pages 77 and 131)

[KS10] S. Knysh and V. Smelyanskiy.
On the relevance of avoided crossings away from quantum critical point to the complexity of quantum adiabatic algorithm.
arXiv preprint arXiv:1005.3011, 2010.
(page 65)

[KSH17] A. Krizhevsky, I. Sutskever, and G. E. Hinton.
Imagenet classification with deep convolutional neural networks.
Communications of the ACM, 60(6):84–90, 2017.
(page 80)

[KSR+21] K. Kishi, T. Satoh, R. Raymond, N. Yamamoto, and Y. Sakakibara.
Graph kernels encoding features of all subgraphs by quantum superposition, 2021, 2103.16093.
(page 84)

[KZI+10] C. Knoernschild, X. L. Zhang, L. Isenhower, A. T. Gill, F. P. Lu, M. Saffman, and J. Kim.
Independent individual addressing of multiple neutral atom qubits with a micromirror-based beam steering system.
Applied Physics Letters, 97(13):134101, 2010.
(page 9)

[Lab16] H. Labuhn.
Rydberg excitation dynamics and correlations in arbitrary 2D arrays of single atoms.
PhD thesis, Université Paris-Saclay (ComUE), 2016.
(pages 7 and 76)

[LCS+22] M. Larocca, P. Czarnik, K. Sharma, G. Muraleedharan, P. J. Coles, and M. Cerezo.
Diagnosing barren plateaus with tools from quantum optimal control.
Quantum, 6:824, 2022.
(page 112)

[Ldk18] S. d. Leseleuc de kerouara.
Quantum simulation of spin models with assembled arrays of Rydberg atoms.

Theses, Université Paris Saclay (COmUE), Dec. 2018.
URL `https://pastel.archives-ouvertes.fr/tel-02088297`.
(pages 6, 39, and 40)

[LdLB+18] V. Lienhard, S. de Léséleuc, D. Barredo, T. Lahaye, A. Browaeys, M. Schuler, L.-P. Henry, and A. M. Läuchli.
Observing the space-and time-dependent growth of correlations in dynamically tuned synthetic ising models with antiferromagnetic interactions.
Physical Review X, 8(2):021070, 2018.
(page 67)

[Lem12] C. Lemaréchal.
Cauchy and the gradient method.
Doc Math Extra, 251(254):10, 2012.
(page 79)

[LKS+19] H. Levine, A. Keesling, G. Semeghini, A. Omran, T. T. Wang, S. Ebadi, H. Bernien, M. Greiner, V. Vuletić, H. Pichler, et al.
Parallel implementation of high-fidelity multiqubit gates with neutral atoms.
Physical review letters, 123(17):170503, 2019.
(page 9)

[LNK03] D. Liben-Nowell and J. Kleinberg.
The link prediction problem for social networks.
In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, page 556–559, New York, NY, USA, 2003. Association for Computing Machinery.
doi:10.1145/956863.956972.
(page 83)

[Loh96] W.-L. Loh.
On latin hypercube sampling.
The annals of statistics, 24(5):2058–2080, 1996.
(page 45)

[Lov79] L. Lovasz.
On the shannon capacity of a graph.
IEEE Transactions on Information Theory, 25(1):1–7, 1979.
doi:10.1109/TIT.1979.1055985.
(page 92)

[LR72a] E. H. Lieb and D. W. Robinson.
The finite group velocity of quantum spin systems.
In *Statistical mechanics*, pages 425–431. Springer, 1972.
(page 74)

[LR72b] E. H. Lieb and D. W. Robinson.
The finite group velocity of quantum spin systems.
Communications in Mathematical Physics, 28(3):251 – 257, 1972.
doi:cmp/1103858407.
(page 85)

[LR86] F. T. Leighton and A. L. Rosenberg.
Three-dimensional circuit layouts.
SIAM Journal on Computing, 15(3):793–813, 1986.
(page 71)

[LSS+22a] M. Larocca, F. Sauvage, F. M. Sbahi, G. Verdon, P. J. Coles, and M. Cerezo.
Group-invariant quantum machine learning.
PRX Quantum, 3:030341, Sep 2022.
doi:10.1103/PRXQuantum.3.030341.
(page 84)

[LSS+22b] M. Larocca, F. Sauvage, F. M. Sbahi, G. Verdon, P. J. Coles, and M. Cerezo.
Group-invariant quantum machine learning.
PRX Quantum, 3(3):030341, 2022.
(page 112)

[LTD+22] J. Landman, S. Thabet, C. Dalyac, H. Mhiri, and E. Kashefi.
Classically approximating variational quantum machine learning with random fourier features, 2022, 2210.13200.
(pages 2 and 112)

[LTOS19] Z. Li, J.-F. Ton, D. Oglic, and D. Sejdinovic.
Towards a unified analysis of random fourier features.
In *International conference on machine learning*, pages 3905–3914. PMLR, 2019.
(page 117)

[Luc14] A. Lucas.
Ising formulations of many np problems.
Frontiers in physics, 2:5, 2014.
(page 11)

[LW18] J.-G. Liu and L. Wang.
Differentiable learning of quantum circuit born machines.
Physical Review A, 98(6):062324, 2018.
(page 113)

[Mae84] H. Maehara.
Space graphs and sphericity.
Discrete Applied Mathematics, 7(1):55–64, 1984.
doi:https://doi.org/10.1016/0166-218X(84)90113-6.
(page 69)

[Mah88] M. S. Mahoney.
The history of computing in the history of technology.
Annals of the History of Computing, 10(2):113–125, 1988.
(page 51)

[Mar17] M. Mari.
Study of greedy algorithm for solving maximum independent set problem, 2017.
URL `https://www.di.ens.fr/~mmari/content/papers/rapport.pdf`.
(page 70)

[MBS+18] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven.
Barren plateaus in quantum neural network training landscapes.
Nature communications, 9(1):1–6, 2018.
(pages 23 and 111)

[MHC+22] A. J. Menssen, A. Hermans, I. Christen, T. Propson, C. Li, A. J. Leenheer, M. Zimmermann, M. Dong, H. Larocque, H. Raniwala, et al.
Scalable photonic integrated circuits for programmable control of atomic systems.
arXiv preprint arXiv:2210.03100, 2022.
(page 9)

[Mit18] A. Mitra.
Quantum quench dynamics.
Annual Review of Condensed Matter Physics, 9:245–259, 2018.
(page 9)

[MKW21] C. O. Marrero, M. Kieferová, and N. Wiebe.
Entanglement-induced barren plateaus.
PRX Quantum, 2(4):040316, 2021.
(page 24)

[MLLK22] S. Maskey, Y. Lee, R. Levie, and G. Kutyniok.
Stability and generalization capabilities of message passing graph neural networks.
CoRR, abs/2202.00645, 2022, 2202.00645.
URL `https://arxiv.org/abs/2202.00645`.
(page 86)

[MM09] M. Mezard and A. Montanari.
Information, physics, and computation.
Oxford University Press, 2009.
(page 11)

[MMC22] P. Mernyei, K. Meichanetzidis, and I. I. Ceylan.
Equivariant quantum graph circuits.
In *Proceedings of the 39th International Conference on Machine Learning, ICML 2022, Baltimore, Maryland, USA*, 2022.
URL `https://arxiv.org/pdf/2112.05261.pdf`.
(page 88)

[MMGF+23] J. J. Meyer, M. Mularski, E. Gil-Fuster, A. A. Mele, F. Arzani, A. Wilms, and J. Eisert.
Exploiting symmetry in variational quantum machine learning.
PRX Quantum, 4(1):010328, 2023.
(page 112)

[MOB20] G. Muzio, L. O'Bray, and K. Borgwardt.
Biological network analysis with deep learning.
Briefings in Bioinformatics, 22(2):1515–1530, 11 2020, https://academic.oup.com/bib/article-pdf/22/2/1515/36655309/bbaa257.pdf.
doi:10.1093/bib/bbaa257.
(page 83)

[MPV87] M. Mézard, G. Parisi, and M. A. Virasoro.
Spin glass theory and beyond: An Introduction to the Replica Method and Its Applications, volume 9.
World Scientific Publishing Company, 1987.
(page 65)

[MRF+19] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe.
Weisfeiler and leman go neural: Higher-order graph neural networks.
Proceedings of the AAAI Conference on Artificial Intelligence, 33(01):4602–4609, Jul. 2019.
doi:10.1609/aaai.v33i01.33014602.
(page 109)

[MSX+20] R. Mukherjee, F. Sauvage, H. Xie, R. Löw, and F. Mintert.
Preparation of ordered states in ultra-cold gases using bayesian optimization.
New Journal of Physics, 22(7):075001, 2020.
(page 44)

[MXZ06] C. A. Micchelli, Y. Xu, and H. Zhang.
Universal kernels.
Journal of Machine Learning Research, 7(95):2651–2667, 2006.
URL `http://jmlr.org/papers/v7/micchelli06a.html`.
(page 93)

[MZ02] M. Mézard and R. Zecchina.
Random k-satisfiability problem: From an analytic solution to an efficient algorithm.
Physical Review E, 66(5):056126, 2002.
(page 65)

[NHK05] T. Nieberg, J. Hurink, and W. Kern.
A robust ptas for maximum weight independent sets in unit disk graphs.
In *Graph-Theoretic Concepts in Computer Science: 30th International Workshop, WG 2004, Bad Honnef, Germany, June 21-23, 2004. Revised Papers 30*, pages 214–221. Springer, 2005.
(page 67)

[NLW+23] M.-T. Nguyen, J.-G. Liu, J. Wurtz, M. D. Lukin, S.-T. Wang, and H. Pichler.
Quantum optimization with arbitrary connectivity using rydberg atom arrays.
PRX Quantum, 4:010316, Feb 2023.
doi:10.1103/PRXQuantum.4.010316.
(page 74)

[NM65] J. A. Nelder and R. Mead.
A simplex method for function minimization.
Computer Journal, 7:308–313, 1965.
(page 25)

[NSB+22] Q. T. Nguyen, L. Schatzki, P. Braccia, M. Ragone, P. J. Coles, F. Sauvage, M. Larocca, and M. Cerezo.
Theory for equivariant quantum neural networks.
arXiv preprint arXiv:2210.08566, 2022.
(page 84)

[Ope23] OpenAI.
Gpt-4 technical report, 2023, 2303.08774.
(page 80)

[PB17] J. Park and S. Boyd.
General heuristics for nonconvex quadratically constrained quadratic programming.
arXiv:1703.07870, 2017.
(pages 133 and 136)

[PNS+19] A. Pillepich, D. Nelson, V. Springel, R. Pakmor, P. Torrey, R. Weinberger, M. Vogelsberger, F. Marinacci, S. Genel, A. van der Wel, et al.
First results from the tng50 simulation: the evolution of stellar and gaseous discs across cosmic time.
Monthly Notices of the Royal Astronomical Society, 490(3):3196–3233, 2019.
(page 5)

[POKB20] T. Pourhabibi, K.-L. Ong, B. H. Kam, and Y. L. Boo.
Fraud detection: A systematic literature review of graph-based anomaly detection approaches.
Decision Support Systems, 133:113303, 2020.
doi:https://doi.org/10.1016/j.dss.2020.113303.
(page 83)

[Pow94] M. J. D. Powell.
A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation, pages 51–67.
Springer Netherlands, Dordrecht, 1994.
doi:10.1007/978-94-015-8330-5_4.
(page 25)

[PS22] E. Peters and M. Schuld.
Generalization despite overfitting in quantum machine learning models.
arXiv preprint arXiv:2209.05523, 2022.
(page 119)

[PSTZ+18] H. Pichler, W. Sheng-Tao, L. Zhou, S. Choi, and M. Lukin.
Quantum optimization for maximum independent set using rydberg atom arrays.
arXiv:1808.10816v1, 2018.
(pages 14, 19, and 20)

[PT97] A. Papakostas and I. G. Tollis.
Incremental orthogonal graph drawing in three dimensions.
In G. DiBattista, editor, *Graph Drawing*, pages 52–63, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.
(page 72)

[PWZ+18] H. Pichler, S.-T. Wang, L. Zhou, S. Choi, and M. D. Lukin.
Quantum optimization for maximum independent set using rydberg atom arrays, 2018.
doi:10.48550/ARXIV.1808.10816.

(pages 11 and 71)

[PY91] C. H. Papadimitriou and M. Yannakakis.
Optimization, approximation, and complexity classes.
Journal of Computer and System Sciences, 43(3):425–440, 1991.
doi:https://doi.org/10.1016/0022-0000(91)90023-X.
(pages 51 and 68)

[QZB10] C. Quinn, D. Zimmerle, and T. H. Bradley.
The effect of communication architecture on the availability, reliability, and economics of plug-in hybrid electric vehicle-to-grid ancillary services.
Journal of Power Sources, 195(5):1500–1509, 2010.
(page 15)

[QZQ$^+$18] Z. Qin, H. Zhang, X. Qin, K. Xu, K. N. A. Dimitrov, G. Wang, W. Yu, Z. Qin, H. Zhang, X. Qin, et al.
The history of computing.
Fundamentals of Software Culture, pages 1–36, 2018.
(page 4)

[RB20] M. A. H. Rafi and J. Bauman.
A comprehensive review of dc fast-charging stations with energy storage: Architectures, power converters, and analysis.
IEEE Transactions on Transportation Electrification, 7(2):345–368, 2020.
(page 15)

[RBH01] J.-M. Raimond, M. Brune, and S. Haroche.
Manipulating quantum entanglement with atoms and photons in a cavity.
Reviews of Modern Physics, 73(3):565, 2001.
(page 6)

[RBN$^+$22] M. Ragone, P. Braccia, Q. T. Nguyen, L. Schatzki, P. J. Coles, F. Sauvage, M. Larocca, and M. Cerezo.
Representation theory for geometric quantum machine learning.
arXiv preprint arXiv:2210.07980, 2022.
(page 84)

[RGD$^+$22] L. Rampášek, M. Galkin, V. P. Dwivedi, A. T. Luu, G. Wolf, and D. Beaini.
Recipe for a general, powerful, scalable graph transformer.
arXiv preprint arXiv:2205.12454, 2022.
(page 110)

[ROM11] P. Rocca, G. Oliveri, and A. Massa.
Differential evolution as applied to electromagnetics.
IEEE Antennas and Propagation Magazine, 53(1):38–49, 2011.
(page 32)

[RR09] A. Rahimi and B. Recht.
Random features for large-scale kernel machines.
Advances in Neural Information Processing Systems 20 - Proceedings of the 2007 Conference, pages 1–10, 2009.
(pages 112, 116, 117, and 118)

[RRG18] D. R.-J. G.-J. Rydning, J. Reinsel, and J. Gantz.
The digitization of the world from edge to core.
Framingham: International Data Corporation, 16, 2018.
(page 5)

[Rud17] W. Rudin.
Fourier analysis on groups.
Courier Dover Publications, 2017.
(page 117)

[S$^+$01] R. V. Solé et al.

The small world of human language.
Proceedings. Biological Sciences, 268(1482):2261–2265, 2001.
(page 83)

[ŠA18] N. Šibalić and C. S. Adams.
Rydberg physics.
IOP Publishing, 2018.
(page 7)

[Sah76] S. K. Sahni.
Algorithms for scheduling independent tasks.
J. ACM, 23(1):116–127, Jan. 1976.
doi:10.1145/321921.321934.
(page 28)

[SBC+20] Y. R. Sanders, D. W. Berry, P. C. Costa, L. W. Tessler, N. Wiebe, C. Gidney, H. Neven, and R. Babbush.
Compilation of fault-tolerant quantum heuristics for combinatorial optimization.
PRX Quantum, 1(2):020312, 2020.
(page 76)

[SBI+20] M. Schuld, K. Brádler, R. Israel, D. Su, and B. Gupt.
Measuring the similarity of graphs with a gaussian boson sampler.
Physical Review A, 101(3):032314, 2020.
(page 84)

[Sch21] M. Schuld.
Supervised quantum machine learning models are kernel methods.
arXiv preprint arXiv:2101.11020, 2021.
(pages 84, 91, and 119)

[SCY+23] A. Skolik, M. Cattelan, S. Yarkoni, T. Bäck, and V. Dunjko.
Equivariant quantum circuits for learning on weighted graphs.
npj Quantum Information, 9(1):47, 2023.
(page 84)

[SEM22] F. J. Schreiber, J. Eisert, and J. J. Meyer.
Classical surrogates for quantum learning models.
arXiv preprint arXiv:2206.11740, 2022.
(page 118)

[SGD+22] H. Silvério, S. Grijalva, C. Dalyac, L. Leclerc, P. J. Karalekas, N. Shammah, M. Beji, L.-P. Henry, and L. Henriet.
Pulser: An open-source package for the design of pulse sequences in programmable neutral-atom arrays.
Quantum, 6:629, 2022.
(pages 3 and 41)

[Shn92] B. Shneiderman.
Tree visualization with tree-maps: 2-d space-filling approach.
ACM Transactions on graphics (TOG), 11(1):92–99, 1992.
(page 72)

[Sho94] P. W. Shor.
Algorithms for quantum computation: discrete logarithms and factoring.
In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.
(page 81)

[SK19] M. Schuld and N. Killoran.
Quantum machine learning in feature hilbert spaces.
Phys. Rev. Lett., 122:040504, Feb 2019.
doi:10.1103/PhysRevLett.122.040504.
(page 84)

[SKR01] J. B. Schafer, J. A. Konstan, and J. Riedl.
E-commerce recommendation applications.
Data Min. Knowl. Discov., 5(1–2):115–153, jan 2001.
doi:10.1023/A:1009804230409.
(page 83)

[Sku98] M. Skutella.
Semidefinite relaxations for parallel machine scheduling.
In *39th Annual Symposium on Foundations of Computer Science Palo Alto, CA, USA*, pages 472–481, 1998.
(page 17)

[SLK+21] G. Semeghini, H. Levine, A. Keesling, S. Ebadi, T. T. Wang, D. Bluvstein, R. Verresen, H. Pichler, M. Kalinowski, R. Samajdar, et al.
Probing topological spin liquids on a programmable quantum simulator.
Science, 374(6572):1242–1247, 2021.
(page 76)

[SLSC22] L. Schatzki, M. Larocca, F. Sauvage, and M. Cerezo.
Theoretical guarantees for permutation-equivariant quantum neural networks.
arXiv preprint arXiv:2210.09974, 2022.
(page 84)

[SMA20] M. F. Serret, B. Marchand, and T. Ayral.
Solving optimization problems with rydberg analog quantum computers realistic requirements for quantum advantage using noisy simulation and classical benchmarks.
arXiv:2006.11190, 2020.
(pages 49 and 67)

[Smi56] W. E. Smith.
Various optimizers for single–stage production.
Naval Research and Logistics Quarterly, 3:59–66, 1956.
(page 16)

[SNL+18] G. Siglidis, G. Nikolentzos, S. Limnios, C. Giatsidis, K. Skianis, and M. Vazirgiannis.
Grakel: A graph kernel library in python, 2018.
doi:10.48550/ARXIV.1806.02193.
(page 98)

[SP97] R. Storn and K. Price.
Differential evolution –a simple and efficient heuristic for global optimization over continuous spaces.
Journal of Global Optimization, 11(4):341–359, 1997.
(pages 24 and 25)

[SP21] M. Schuld and F. Petruccione.
Machine learning with quantum computers.
Springer, 2021.
(pages 79 and 80)

[Spi99] F. C. R. Spieksma.
On the approximability of an interval scheduling problem.
J. of Scheduling, 2(5):215–227, 1999.
(pages 19 and 52)

[SRPG01] N. Schlosser, G. Reymond, I. Protsenko, and P. Grangier.
Sub-poissonian loading of single atoms in a microscopic dipole trap.
Nature, 411(6841):1024–1027, 2001.
(page 7)

[SS15] D. J. Sutherland and J. Schneider.
On the error of random fourier features.
arXiv preprint arXiv:1506.02785, 2015.

(pages 117 and 121)

[SSM21] M. Schuld, R. Sweke, and J. J. Meyer.
Effect of data encoding on the expressive power of variational quantum-machine-learning models.
Physical Review A, 103(3):1–16, 2021, 2008.08605.
doi:10.1103/PhysRevA.103.032430.
(pages 113 and 114)

[SSW+21] P. Scholl, M. Schuler, H. J. Williams, A. A. Eberharter, D. Barredo, K.-N. Schymik, V. Lienhard, L.-P. Henry, T. C. Lang, T. Lahaye, et al.
Quantum simulation of 2d antiferromagnets with hundreds of rydberg atoms.
Nature, 595(7866):233–238, 2021.
(page 41)

[ST95] J. SHAO and D. TU.
The Jackknife and Bootstrap.
SPRINGER, 1995.
(page 89)

[STJ22] S. Shin, Y. Teo, and H. Jeong.
Exponential data encoding for quantum supervised learning.
arXiv preprint arXiv:2206.12105, 2022.
(pages 115, 121, and 125)

[STT81] K. Sugiyama, S. Tagawa, and M. Toda.
Methods for visual understanding of hierarchical system structures.
IEEE Transactions on Systems, Man, and Cybernetics, 11(2):109–125, 1981.
(page 72)

[SV91] R. Shapiro and B. L. Vallee.
Interaction of human placental ribonuclease with placental ribonuclease inhibitor.
Biochemistry, 30(8):2246–2255, 02 1991.
doi:10.1021/bi00222a030.
(page 99)

[SVP+09] N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt.
Efficient graphlet kernels for large graph comparison.
In *JMLR Workshop and Conference Proceedings Volume 5: AISTATS 2009*, pages 488–495, Cambridge, MA, USA, Apr. 2009. Max-Planck-Gesellschaft, MIT Press.
(page 94)

[SWB+22] P. Scholl, H. J. Williams, G. Bornet, F. Wallner, D. Barredo, L. Henriet, A. Signoles, C. Hainaut, T. Franz, S. Geier, et al.
Microwave engineering of programmable x x z hamiltonians in arrays of rydberg atoms.
PRX Quantum, 3(2):020303, 2022.
(page 8)

[SZF+15] P. Schauß, J. Zeiher, T. Fukuhara, S. Hild, M. Cheneau, T. Macrì, T. Pohl, I. Bloch, and C. Groß.
Crystallization in ising quantum magnets.
Science, 347(6229):1455–1458, 2015.
(page 9)

[TFH22] S. Thabet, R. Fouilland, and L. Henriet.
Extending Graph Transformers with Quantum Computed Aggregation.
arXiv e-prints, page arXiv:2210.10610, Oct. 2022, 2210.10610.
(pages 88 and 109)

[TGS+19] M. C. Tran, A. Y. Guo, Y. Su, J. R. Garrison, Z. Eldredge, M. Foss-Feig, A. M. Childs, and A. V. Gorshkov.
Locality and digital quantum simulation of power-law interactions.
Phys. Rev. X, 9:031006, Jul 2019.
doi:10.1103/PhysRevX.9.031006.
(page 85)

[THC23] S. Tukra, F. Hoffman, and K. Chatfield.
Improving visual representation learning through perceptual understanding.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14486–14495, 2023.
(page 80)

[Tho81] S. W. Thomson.
The tide gauge, tidal harmonic analyser, and tide predicter.
In *Minutes of the Proceedings of the Institution of Civil Engineers*, volume 65, pages 2–25. Thomas Telford-ICE Virtual Library, 1881.
(page 4)

[TLI+23] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al.
Llama: Open and efficient foundation language models.
arXiv preprint arXiv:2302.13971, 2023.
(page 80)

[TvDEF09] A. Theocharidis, S. van Dongen, A. J. Enright, and T. C. Freeman.
Network visualization and analysis of gene expression data using biolayout express3d.
Nature Protocols, 4:1535–1550, 2009.
(page 83)

[Vaz10] V. V. Vazirani.
Approximation Algorithms.
Springer Publishing Company, Incorporated, 2010.
(page 66)

[vL06] E. J. van Leeuwen.
Better approximation schemes for disk graphs.
In *Algorithm Theory–SWAT 2006: 10th Scandinavian Workshop on Algorithm Theory, Riga, Latvia, July 6-8, 2006. Proceedings 10*, pages 316–327. Springer, 2006.
(page 70)

[VML+19a] G. Verdon, T. McCourt, E. Luzhnica, V. Singh, S. Leichenauer, and J. Hidary.
Quantum graph neural networks.
arXiv preprint arXiv:1909.12264, 2019.
(page 84)

[VML+19b] G. Verdon, T. McCourt, E. Luzhnica, V. Singh, S. Leichenauer, and J. Hidary.
Quantum Graph Neural Networks.
arXiv e-prints, page arXiv:1909.12264, Sept. 2019, 1909.12264.
(page 88)

[VSKB10] S. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt.
Graph kernels.
Journal of Machine Learning Research, 11(40):1201–1242, 2010.
URL `http://jmlr.org/papers/v11/vishwanathan10a.html`.
(page 95)

[VSP+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin.
Attention is all you need.
Advances in neural information processing systems, 30, 2017.
(page 80)

[Wei55] M. H. Weik.
Ballistic research laboratories report no. 971: A survey of domestic electronic digital computing systems.
S Department of Commerce, Tech. Rep, 1955.
(page 51)

[Wei03] E. W. Weisstein.
Gershgorin circle theorem.

https://mathworld. wolfram. com/, 2003.
(page 120)

[WF94] C. Ware and G. Franck.
Viewing a graph in a virtual reality display is three times as good as a 2d diagram.
In *Proceedings of 1994 IEEE Symposium on Visual Languages*, pages 182–183. IEEE, 1994.
(page 71)

[WHJR18] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel.
Quantum approximate optimization algorithm for maxcut: A fermionic view.
Physical Review A, 97(2), Feb 2018.
doi:10.1103/physreva.97.022304.
(page 27)

[WL68] B. Weisfeiler and A. Leman.
The reduction of a graph to canonical form and the algebra which appears therein.
nti, Series, 2(9):12–16, 1968.
(page 85)

[WL20] J. Wurtz and P. J. Love.
Bounds on maxcut qaoa performance for p>1, 2020, 2010.11209.
(page 32)

[WM08] C. Ware and P. Mitchell.
Visualizing graphs in three dimensions.
ACM Trans. Appl. Percept., 5(1), jan 2008.
doi:10.1145/1279640.1279642.
(page 71)

[WYK22] H. Wang, H.-C. Yeh, and A. Kamenev.
Many-body localization enables iterative quantum optimization.
Nature communications, 13(1):5503, 2022.
(page 76)

[XB06] J. Xu and B. Berger.
Fast and accurate algorithms for protein side-chain packing.
Journal of the ACM (JACM), 53(4):533–557, 2006.
(page 65)

[XHLJ19] K. Xu, W. Hu, J. Leskovec, and S. Jegelka.
How powerful are graph neural networks?
In *International Conference on Learning Representations*, 2019.
URL `https://openreview.net/forum?id=ryGs6iA5Km`.
(pages 86 and 109)

[YCL$^+$21] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu.
Do transformers really perform badly for graph representation?
Advances in Neural Information Processing Systems, 34:28877–28888, 2021.
(page 110)

[YKS10] A. Young, S. Knysh, and V. Smelyanskiy.
First-order phase transition in the quantum adiabatic algorithm.
Physical review letters, 104(2):020502, 2010.
(page 65)

[YONL19] Y. Yamashiro, M. Ohkuwa, H. Nishimori, and D. A. Lidar.
Dynamics of reverse annealing for the fully connected p-spin model.
Physical Review A, 100(5):052321, 2019.
(page 76)

[YY03] H. Yang and J. Ye, Yinyu. Zhang.
An approximation algorithm for scheduling two parallel machines with capacity constraints.

Discrete Applied Mathematics, 130:449–467, 2003.
(page 17)

[ZGL21] J. Zheng, Q. Gao, and Y. Lü.
Quantum graph convolutional neural networks.
In *2021 40th Chinese Control Conference (CCC)*, pages 6335–6340, 2021.
doi:10.23919/CCC52363.2021.9550372.
(page 84)

[ZLW21] C. Zoufal, A. Lucchi, and S. Woerner.
Variational quantum boltzmann machines.
Quantum Machine Intelligence, 3(1):1–15, 2021.
(page 113)

[Zuc06] D. Zuckerman.
Linear degree extractors and the inapproximability of max clique and chromatic number.
In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 681–690, 2006.
(page 68)

[ZWC$^+$20] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin.
Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices.
Phys. Rev. X, 10:021067, Jun 2020.
doi:10.1103/PhysRevX.10.021067.
(pages 24 and 30)