Universiteit Leiden
The Netherlands

Optimization of quantum algorithms for near-term quantum computers

Bonet Monroig, X.

1. Introduction

1.1. Preface

Quantum computation is a paradigm of computation and information processing in which the laws of quantum mechanics are used to perform calculations. The basic unit of information storage is called a quantum bit or qubit, and it has properties beyond those of its classical counter-part, the bit. The most notable of which is the fact that information on a qubit can be stored in a superposition of the 0 and 1 states,

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle, \alpha_i \in \mathbb{C} \tag{1.1}$$

$$|\alpha_0|^2 + |\alpha_1|^2 = 1. \tag{1.2}$$

The complexity to describe a (pure) N-qubit quantum state grows exponentially with the system size. This is because a complete description of such quantum state has $2^N - 1$ α_i (complex) coefficients or amplitudes

$$|\psi\rangle = \alpha_0|00\ldots0\rangle + \alpha_1|00\ldots1\rangle + \cdots + \alpha_{2^N-1}|11\ldots1\rangle. \tag{1.3}$$

A quantum computer is a collection of qubits, and thus a quantum computation is the use of such device to perform a calculation. As a specialized machine, quantum computers require dedicated algorithms that take advantage of quantum mechanical features to solve problems. In the early 90s, Deutsch and Josza [1, 2] showed that a quantum algorithm implemented on a quantum computer could solve a specific classical problem exponentially faster in time compared to the best-known classical algorithm for the same task. In 1997 a ground-breaking development by Peter Shor demonstrated the first quantum algorithm that had an exponential run-time advantage with respect to its classical counterpart for finding the prime factors of an integer [3]. Since then, the interest on quantum computation increased and mostly focused on developing quantum algorithms that show quantum speed-up in solving classically hard problems [5]. In recent years, research in quantum algorithms has gone beyond the search for speed-ups to a wider range of topics such as understanding what makes quantum computers powerful [4], studying quantum states that cannot be

accessed with a classical computer [6, 7] or designing fine-controlled quantum sensing protocols [8].

In the past two decades ground-breaking developments in experimental quantum physics has opened the door to study the power of quantum computation. Academic and industry laboratories world-wide have been devoted to manufacture prototypical quantum hardware, using a variety of platforms. In the process, researchers have observed that realistic quantum computers are highly sensitive to noise. The large number of imperfect and unwanted operations of current quantum hardware makes it unrealistic to run the originally proposed quantum algorithms. A solution to these problem is to design quantum error correction schemes such that with bigger devices and better control one can approach a noise-free quantum computer. Unfortunately, with the current state-of-the-art quantum hardware, a useful, fault-tolerant quantum computer will require millions of qubits [9, 10]. Until million-qubit devices are achieved we remain in an intermediate era, where the potential for application remains, but where low-cost error mitigation must replace provable error correction techniques.

This thesis studies various aspects of near-term quantum computing; from noise characterization and error mitigation schemes to information extraction of quantum states, as well as classical optimization methods for circuits. The results of this work cover a series of topics that are necessary to demonstrate beyond-classical computational advantage in noisy quantum hardware.

1.2. Basics of quantum algorithms

A quantum algorithm is a set of instructions given to a quantum computer to perform a task. It involves preparing a quantum state $|\psi\rangle$, evolving the state $|\psi\rangle \rightarrow U|\psi\rangle$ via unitary operations ($UU^\dagger = U^\dagger U = I$), and measure observables. A classical post-processing step is generally used to interpret the extracted information and/or to continue with a new algorithm iteration.

Typically, quantum registers are initialize in their 0 state, such that

$$|\psi_{\text{initial}}\rangle = |0\rangle^{\otimes N}. \tag{1.4}$$

This state is evolved applying a sequence of quantum (unitary) gates that transform the quantum state

$$|\psi_{\text{final}}\rangle = U_k U_{k-1} \dots U_0 |0\rangle^{\otimes N}. \tag{1.5}$$

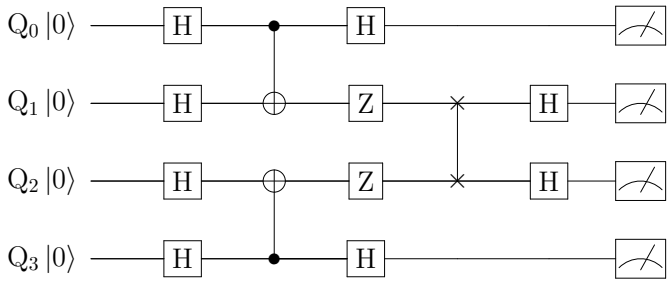

Figure 1.1.: An example of a 4-qubit quantum circuit. The initial state $|\psi_0\rangle = |0000\rangle$ is transformed by the use of one- and two-qubit gates to a different state. The boxes at the end of the circuit represent a measurement apparatus, where information about $|\psi_1\rangle$ is extracted.

The information is extracted from the quantum device by measuring measuring one or more qubits. From the 2^N bits of information encoded in the quantum state, only a maximum of N bits of information can be obtained after a measurements operation. This is a huge bottleneck when using a quantum computer.

The system is projected through the Born's rule [11] to the computational state described by these measurements, and the rest of the information is lost. An example of a quantum circuit with single-qubit (Hadamard, Z-gate) and two-qubit (CNOT, SWAP) gates is depicted in Figure 1.1. In this figure the time goes from left to right, sequentially applying unitary operations, the U_k in eq. 1.5, at each time-step. The left side of the figure indicates the initial state $|\psi_0\rangle = |0000\rangle$. After the evolution the quantum state is measured, action depicted by the measurements apparatus at the end of the figure.

1.2.1. Quantum state tomography

In the preface of this thesis we stated that a (pure) N-qubit quantum state is described by $2^N - 1$ complex coefficients. Quantum state tomography is the process of obtaining the full description of a given quantum state $|\psi\rangle$ as described in eq. 1.3. One can describe a pure quantum state by its

density matrix representation,

$$\rho = |\psi\rangle\langle\psi| = \sum_{i,j=0}^{2^N-1} \beta_{i,j}|i\rangle\langle j|. \tag{1.6}$$

The density matrix formalism of quantum states is strictly more descriptive as one can also represent non-pure (mixed) quantum state,

$$\rho_{\text{mixed}} = \frac{1}{2}[|0\rangle\langle 0| + |1\rangle\langle 1|]. \tag{1.7}$$

One might think that the equal superposition state,

$$|\psi\rangle = \frac{1}{\sqrt{2}}[|0\rangle + |1\rangle], \tag{1.8}$$

also has a probability $\frac{1}{2}$ of returning 0 or 1. By writing its density matrix representation

$$\rho_{\text{superposition}} = |\psi\rangle\langle\psi| = \frac{1}{2}[|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| + |1\rangle\langle 1|], \tag{1.9}$$

and comparing eq. 1.7 with respect to eq. 1.9 one can see that the mixed state is not the same as the equal superposition state. The density matrix ρ is a complex $2^N \times 2^N$ matrix in the same Hilbert space of the qubit states [43] with entries $\beta_{i,j} = \alpha_i \cdot \alpha_j$. ρ can be expressed as a linear combination of Pauli operators, $\hat{P}_i \in \{I, X, Y, Z\}^{\otimes N}$, as they form a basis of the Hilbert space of the qubits,

$$\rho = \frac{1}{2^N} \sum_{i=0}^{4^N-1} c_i \hat{P}_i. \tag{1.10}$$

In this form, quantum state tomography is reduced to simply estimate the coefficients

$$c_i = \frac{\text{Trace}[\rho P_i]}{2^N}. \tag{1.11}$$

On a quantum computer, eq. 1.11 cannot be exactly computed. After every measurement of an N-qubit quantum state ρ only a maximum of N bits of information can be retrieved. This is formalized by the so-called Holevo bound [20], and is one of the greatest bottlenecks when performing quantum computations. One can only obtain an estimate of c_i by using

M copies of ρ and performing a projective measurement on the basis of the operator P_i (see Fig. 1.2). The estimator of c_i

$$\mathbb{E}[c_i] = \langle P_i \rangle, \tag{1.12}$$

with variance

$$\mathrm{Var}[\mathbb{E}[c_i]] = \frac{1 - \langle P_i \rangle^2}{M}. \tag{1.13}$$

The probability of $\mathbb{E}[c_i]$ being $\epsilon > 0$ close to the true value c_i is given by the Chebyshev's inequality

$$\mathbb{P}[|\mathbb{E}[c_i] - c_i| \geq \epsilon] \leq \frac{\mathrm{Var}[c_i]}{M\epsilon^2}. \tag{1.14}$$

This equation provides a bound on the number of copies of ρ needed to estimate a given coefficient to a certain precision

$$M \sim \frac{1}{\epsilon^2}. \tag{1.15}$$

A full characterization of an N-qubit quantum state requires one to estimate each of the 4^N c_i coefficients. With the method described here the time complexity of running quantum state tomography will scale as

$$\mathrm{Time} = \mathcal{O}(\epsilon^2 \cdot 4^N). \tag{1.16}$$

Newer quantum state tomography methods [21] have been able to reduce the time-complexity to

$$\mathrm{Time} = \epsilon \cdot \exp \mathcal{N}. \tag{1.17}$$

thus making quantum state tomography unfeasible even for a small-sized quantum computers.

In practice, we don't need to estimate every expectation value of a quantum state. Given an operator

$$\hat{O} = \sum_i o_i P_i, \tag{1.18}$$

where P_i are Pauli operators, we need only measure those $\langle P_i \rangle$ that have non-zero o_i coefficients. For instance, the electronic structure problem, a central problem in quantum chemistry, only requires one to characterize the state on $\mathcal{O}(N^4)$ operators where N refers to the number of spin-orbitals. This is an exponential reduction in the number of measurements with respect to characterizing the full quantum state [108, 119].

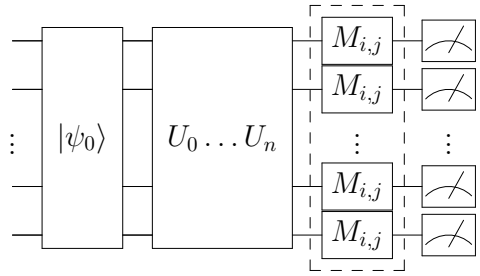

Figure 1.2.: Quantum state tomography is done by measuring the state in different basis rotations before extracting the information the qubits, $M_{i,j}$ gate in the figure. The boxed gates in the figure are the tomography rotations that allow to extract the information from the quantum state in different bases. In general, there are 4 possible basis rotations, namely the Pauli matrices I, X, Y and Z. Recover the full quantum state requires to prepare all possible combinations of the Pauli matrices in all possible qubits. However, typically we are only interested in a subset of these $M_{i,j}$ combinations to extract the most important information of the quantum state.

1.2.2. Variational quantum algorithms

Variational quantum algorithms are a family of quantum algorithms in which a set of classical parameters are used to control the quantum state of a device [39–41, 87]. These parameters are included during the evolution step of the algorithm such that some of the quantum gates are parametrized. Therefore, the parametrized quantum state is $|\psi(\vec{\theta})\rangle = U(\vec{\theta})|0\rangle^{\otimes N}$. Figure 1.3 shows an example of a parametrized quantum circuit, where single-qubit rotation gates can be modified based on the classical parameters $\vec{\theta} = \theta_0 \dots \theta_n$.

Variational quantum algorithms require an additional element, namely a real-valued cost function $\mathcal{C}(\vec{\theta})$ to optimize the parameters of the quantum state. $\mathcal{C}$ is typically the expectation value of a quantum observable $\hat{O}$, a Hermitian operator in the Hilbert space of the qubits. On a quantum computer Hermitian observables can be measured by expressing them as a sum of Pauli operators, e.g. $\hat{P}_i \in \{I, X, Y, Z\}^{\otimes N}$,

$$\hat{O} = \sum_i c_i \hat{P}_i \rightarrow \mathcal{C}(\vec{\theta}) = \langle \hat{O} \rangle = \sum_i c_i \langle \hat{P}_i \rangle, \tag{1.19}$$

with $c_i \in \mathbb{R}$. In Sec. 1.2.1 we described how to characterize a N-qubit quantum state through its Pauli decomposition (eq. 1.10). We can use

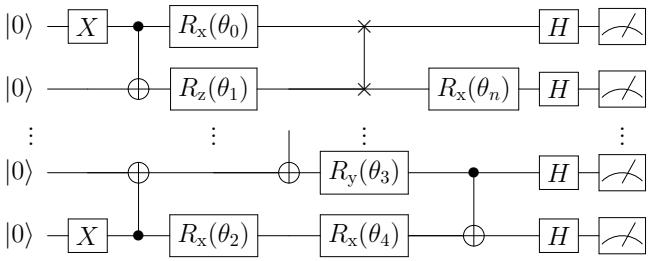

Figure 1.3.: A parametrized quantum circuit is formed by combining fixed and variable gates. In this example we show fixed gates Hadamard (H), X, CNOT or SWAP gates and variable rotation gates around specified axes (X, Y or Z). Although here every parametrized gate has its own parameter, it is possible that multiple gates share the same parameter.

(partial) quantum state tomography can be used to estimate the cost function $\mathcal{C}$ of a variational quantum algorithm. However, the number of operators to estimate now need not be exponential in the system size. In fact, most of the problems of interest only need poly(N) operators. While the Pauli decomposition is a widely use strategy to measure any quantum observable $\hat{O}$ on a quantum computer there exist other methods to estimate $\hat{O}$ such as low-rank factorization [132] or classical shadows of quantum states [65–68].

An evaluation of the cost function involves preparing and measuring the quantum state multiple times to build statistics. Therefore, we only have access to a sampled cost function that depends on the parameters $\vec{\theta}$ and the number of measurements M

$$\bar{\mathcal{C}}(\vec{\theta}, M_i) = \sum_i c_i \big[\langle \hat{P}_i \rangle + \epsilon_i(M_i)\big]. \tag{1.20}$$

The sampling noise manifests in the form of a random variable ϵ_i drawn from a binomial distribution with variance $\sigma_i^2 \sim \frac{1}{M_i}$ (see eq. 1.13).

In a variational quantum algorithm one attempts at finding the set of parameters $\vec{\theta}$ of a fixed parametrized quantum circuit that minimize $\mathcal{C}(\vec{\theta})$. Thus, a classical optimization algorithm is used to update $\vec{\theta}$ after every evaluation of $\mathcal{C}$. The optimization loop continues until some convergence or hard-stop criteria have been met. At the end of the algorithm we are left with a set of parameters $\vec{\theta}_{\text{opt}}$ that minimize $\bar{\mathcal{C}}$. The state generated by the parametrized quantum circuit with $\vec{\theta}_{\text{opt}}$ is an approximation to the

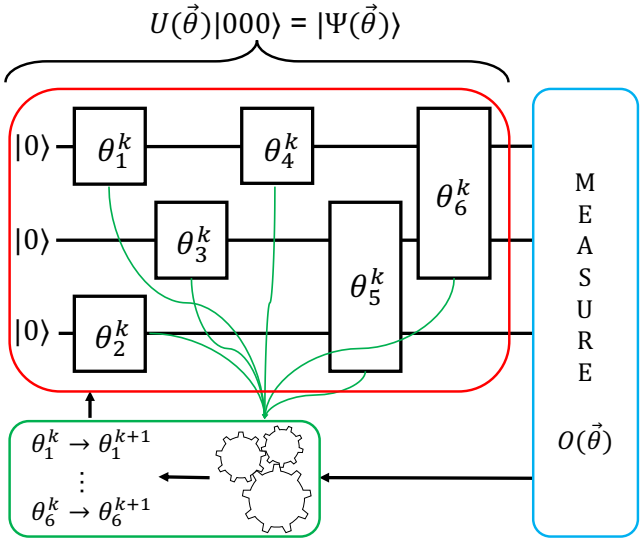

Figure 1.4.: A schematic representation of a variational quantum algorithm. First the quantum device is initialized in a known quantum state $|000\rangle$, followed by a sequence of parametrized quantum operations that prepare a final quantum state $|\Psi(\vec{\theta})\rangle$ (red box). The quantum state is then measured to obtain the observable $\hat{O}(\vec{\theta})$ (blue box), which is fed into a classical optimization loop. A new set of angles $\vec{\theta}^{k+1}$ is given by the optimizer (green box). The new parameters are then used in the red box step, and the process is repeated until some convergence criteria is met. Figure taken with permission from ref. [19]

ground-state of the observable used as a cost function $\hat{O}$. An schematic representation of a variational quantum algorithm is depicted in fig. 1.4.

1.2.3. Classical optimization of variational quantum algorithms

Quantum computers promise to solve problems that are out-of-reach for the largest and fastest existing supercomputers. However such quantum advantage does not imply that quantum computers will replace their classical counter-parts. In fact, classical computers are a crucial part of a quantum computer architecture for its control [30]. Similarly, in variational quantum algorithms a classical optimization method is used to find the optimal values of the parametrized quantum circuit with respect to

the cost function (Sec. 1.2.2). An optimization algorithm is a piece of software that attempts to find the parameters $\vec{x}$ that minimize (or maximize) some cost function $\mathcal{F}(\vec{x})$. To find such optimal parameters, an optimizer suggests candidates $\vec{x}_{1\ldots n}$ to be evaluated on the cost function $\mathcal{F}(\vec{x}_{1\ldots n})$ sequentially. From those evaluations the optimizer updates a set of rules such that a new candidate with a lower (or higher) $\mathcal{F}(\vec{x}_{n+1})$ is found. The search for candidate parameters continues until some convergence criterion is met (e.g. no improvement in $\mathcal{F}(\vec{x})$, changes in parameters smaller than some threshold, etc.) or some hard stop criterion is reached (e.g. total number of function evaluations, maximum time allowed, etc.). It is clear that a successful implementation of a variational quantum algorithm strongly depends on the ability of the classical computer to optimize the given cost function.

Quantum devices need the assistance of a classical optimization method for their characterization and calibration [26–30]. However, the question of how optimal these methods for quantum-optimal control tasks is uncertain and currently a very active research field. Similarly, the optimization of variational quantum algorithms has largely been with off-the-shelf optimizers. Typically optimizers assume that the cost function is deterministic, but in variational quantum algorithms we only have access to a probabilistic output of it, $\bar{\mathcal{C}}$. For this reason, the results in the literature do not show a clear trends, thus making difficult to asses the optimality of commonly used optimizers for variational quantum algorithms.

For decades, computer scientists have been developing strategies to systematically compare and benchmark optimization algorithms for classical cost functions [22–25]. From the field of classical optimization algorithms we have learnt that there is not a universal optimizer: every problem is better optimized with a particular method and for every instance there are optimal hyper-parameters of the optimizer. This reflects the heuristic nature of the optimization algorithms. Progress in variational quantum algorithms requires us to extend these systematic analysis to the newly accessible cost functions through optimizer benchmark and landscape analysis. Recently, some work has been put into the design of optimizers specifically for variational quantum algorithm specific optimizers [138–141] but many questions remain unknown about their capabilities to optimize larger problems.

1.2.4. Applications to quantum chemistry and material science

Quantum computers have been marked as a tool to study problems in the domain of quantum chemistry and material science. Typically, we are interested in calculated the spectrum and/or some properties (e.g. energy-gradients, polarization, magnetization) of the physical systems described by a Hamiltonian. One of such problems in quantum chemistry the electronic structure problem, which describes the behaviour of the electrons around the nuclei, under the Born-Oppenheimer approximation which assumes that the nuclei are fixed particles. The Hamiltonian that describes this systems is

$$\hat{H} = \sum_{pq} h_{pq} \hat{a}_p^\dagger \hat{a}_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} \hat{a}_p^\dagger \hat{a}_q^\dagger \hat{a}_r \hat{a}_s, \tag{1.21}$$

where $\hat{a}$ and $\hat{a}^\dagger$ are the fermion creation and annihilation operators acting on the fermionic modes p, q, r, s. The coefficients h_{pq} and h_{pqrs} are the set of one- and two-electron integrals evaluated classically using a fixed basis set functions [31]. The Fermi-Hubbard model is another example of a system of interest for material science [32, 33],

$$H_{\text{Hubbard}} = H_{\text{t}} + H_{\text{U}} = -t \sum_{(i,j),\sigma} \left(a_{i\sigma}^\dagger a_{j\sigma} + a_{j\sigma}^\dagger a_{i\sigma} \right) + U \sum_i n_{i\uparrow} n_{i\downarrow}. \tag{1.22}$$

It describes the interaction between particles on a lattice of n_x x n_y sites. Particles can tunnel to the nearest-neighbor site hopping strength t but they are prevented to tunnel to an occupied site of the same spin σ with on-site potential strength U.

In quantum chemistry and material science most of the relevant properties of the systems are described by the eigenvalues and eigenstates of the Hamiltonian (e.g. the ground- and low-lying energy states of the system). Therefore, the goal is to accurately compute these energies. Within the variational paradigm, this can be achieved by preparing approximate ground states via variational circuits (Sec. 1.2.2), and extracting energies or other expectation values using the techniques described in Sec. 1.2.1.

Particles in nature are observed to be of two types: fermions or bosons. Bosonic systems have a symmetric wave-function

$$\Psi_{\text{boson}}(\vec{x}_2, \vec{x}_1) = \Psi_{\text{boson}}(\vec{x}_1, \vec{x}_2), \tag{1.23}$$

and thus the exchange of two bosons leaves the system invariant. The symmetric nature of bosons implies that multiple particles can occupy

the same space. In contrast, fermionic systems have an anti-symmetric wave-function

$$\Psi_{\text{fermion}}(\vec{x}_2, \vec{x}_1) = (-1)\Psi_{\text{fermion}}(\vec{x}_1, \vec{x}_2), \tag{1.24}$$

and an exchange of two fermions changes the sign of the wave-function [12]. Typically, many-body quantum systems are studied through the *creation* ($a^\dagger$) and *annihilation* ($a^\dagger$) or second-quantization formalism. The action of these operators on the vacuum state $|0\rangle$ is as follows:

$$a_i|\vec{0}\rangle = 0 \tag{1.25}$$

$$a_i^\dagger|\vec{0}\rangle = |00100\rangle, \tag{1.26}$$

where a particle at the i-th position has been created. The creation and annihilation operators of a bosonic system obeys the commutation relation

$$[a_i, a_j^\dagger] = a_i a_j^\dagger - a_j^\dagger a_i = \delta_{ij}. \tag{1.27}$$

However a fermionic system follows the anti-commutation relation

$$\{a_i, a_j^\dagger\} = a_i a_j^\dagger + a_j^\dagger a_i = \delta_{ij}. \tag{1.28}$$

The anti-symmetric nature of interacting fermionic systems can pose a problem when studying them via e.g. classical Monte-Carlo methods. The so-called fermionic sign problem appears because one must track all fermionic exchanges to account for the correct sign in the wave-function. This sign problem has been proven NP-hard [13], which is strong evidence that calculating ground states of interacting fermionic systems scales exponentially in the system size. On the contrary a qubit can store the information of a fermionic site in a one-to-one correspondence. Polynomial-time quantum algorithms under fair assumptions to solve systems of interacting fermions have been discovered [14–17, 31, 76, 78, 79, 95, 187]. This translates into an exponential speed-up in computing ground states of interacting fermions using a quantum computers with respect to classical methods.

Variational quantum algorithms are suited to approximate the energy spectrum of such fermionic interacting problems in the near-term [18, 19]. However, a fermionic representation is non-native on a quantum computer; the native operations are in terms of qubit (Pauli) operators. Luckily there exist methods to transform fermionic operators onto a polynomial number of Pauli operators. The so-called fermion-to-qubit mappings [34–38] ensure that the anti-commutation relations of the fermionic operators is respected by the Pauli operators.

In the past decade we have seen a large number of experimental demonstration of variational quantum algorithms for quantum chemistry and material science problems. Variational quantum eigensolvers were originally proposed for the task of approximating the ground-state of the Hydrogen molecule in a photonic quantum device [39, 40]. Many more experiments in a wide variety of platforms have followed [41, 42, 44, 45]. The largest variational quantum algorithm for chemistry was run in 2021 with 12 qubits to compute the Hartree-Fock energy of several molecules [46].

1.3. Noisy intermediate-scale quantum computers

In the roadmap towards error-free quantum computers based on quantum error correction schemes there will exist error prone intermediate-scale hardware sufficiently large to challenge the limits of classical computers. This idea led to the definition of noisy intermediate-scale quantum (NISQ) computation by Preskill [47] to explore the capabilities of quantum hardware pre-quantum error correction. As a newly introduced form of quantum computation, NISQ era needs tailored quantum algorithms that take into account the noise limitations of the hardware. A whole new research field has opened to understand, improve and develop quantum hardware and algorithms for noisy intermediate-scale quantum computers [48].

Quantum computation with NISQ devices requires tailored software and hardware to minimize the noise effects. The first necessary step is to understand as accurately as possible the limitations of the hardware. Only then it is possible to explore algorithms and problems that are suited to them. An alternative solution is to construct hardware which is specifically design for a task or problem, namely building analog quantum computers. In this way, the capabilities of the hardware are maximized (for the specific problem) but their usability is largely constrained.

Regardless of the approach, it is likely that NISQ computing will be best suited for highly specialized problems in the field of quantum mechanics. The simplest approach is to use the quantum hardware to prepare a quantum state $|\psi\rangle$, and extract relevant information using quantum state tomography described in Sec. 1.2.1. Ideally, the quantum state is not accessible with a classical computer, and thus we can obtain information which is otherwise impossible to get.

1.3.1. Noise in quantum hardware

A quantum computation involves storing and transforming quantum information on a quantum system. Unfortunately quantum information is fragile and extremely sensitive to interactions between the quantum system and its surroundings. Some level of noise in any quantum processing unit will be unavoidable because we need to interact with it to perform a computation. Therefore, it is important to study which errors occur during the execution and which physical process might have caused such error. By understanding these effects we can construct noise models that reproduce the errors of a quantum device accurately and reliably, allowing us to benchmark the quality of the quantum hardware.

The simplest noise models that reliably describe errors on a quantum processing unit are the so-called bit- and phase-flip errors. A bit-flip error is the process by which the state of a single qubit $|\psi\rangle$ changes from $|\psi\rangle = |0\rangle \to |1\rangle$ (or vice-versa) with probability p_{bit} and remains unchanged with probability $1 - p_{\text{bit}}$. Similarly, a phase-flip error describes the change in the relative phase of the quantum state $|\psi\rangle = |0\rangle + |1\rangle \to |\psi\rangle = |0\rangle - |1\rangle$ of a single qubit with probability p_{phase}. The bit- and phase-flip error models can be combined to a single noise model such that with probability p_{bp} a single qubit undergoes a bit and phase flip error, and with $1-p_{\text{bp}}$ the qubit remains unchanged. Although these noise models describe the errors at a single-qubit level one can use them to approximate the errors of a multi-qubit device by extending the error model to all qubits of the system with the same probability. It is possible to make the bit-phase flip model more accurate by assigning different probabilities of error to each individual qubit. Despite its simplicity the bit- and phase-flip errors are a good approximation to the errors occurring to a quantum device. These noise models are particularly relevant for quantum error correction [60, 85], where repeated measurement of a system projects coherent sources of error to discrete events.

Another important source of error in quantum hardware is the depolarizing noise channel. It describes the process by which a quantum state becomes a maximally mixed state with probability p_{dep}. This process can be seen as the quantum state losing its "quantumness" because the maximally mixed state is a classical state with equal probability of measuring 0 or 1 in a single qubit in the computational basis 1.7.

In reality, noise in a quantum device is not discrete, and its quantum state is continuously modified in time by the action of the environment. Noise models that account for the time-dependent nature of these interactions are needed to describe the errors that occur during the execution

of a quantum algorithm in a noisy device more accurately. An example of such time-dependent interaction is the loss of energy of the quantum device to the environment also known as amplitude damping. The rate at which the energy (or amplitude) of the quantum state is dissipated to the surroundings is given by T_1; after time t, the probability that a system prepared in the excited state has decayed to the ground state is $1 - e^{\frac{-t}{T_1}}$. Another source of error in a quantum processing unit is the loss of information without energy dissipation. This is the phase-flip channel, and it is characterized by the pure dephasing rate T_ϕ; the rate at which the phase decays for a given time t. It is common to use the total loss of phase information rate T_2 instead of the pure dephasing rate. T_2 is the rate at which the phase of a quantum state is damped. It defines the time-scale at which the probability of the phase damping happening is given by $p_{\text{phase}}(t) = 1 - e^{\frac{-t}{T_2}}$. It can be shown that the coherence times T_1 and T_2 are related such that [43]

$$T_1 \geq 2T_2. \tag{1.29}$$

The T_1 and T_2 coherence times are relevant figures to characterize a quantum processing unit. First, they set an upper bound on the number of operations that can be performed on the device before a quantum computation breaks down due to errors. Secondly, they are a standard metric to compare across different qubit platforms. Beyond the errors models introduced in this section, many more error sources are known to affect quantum hardware (see ref. [30] for an overview of noise sources in superconducting qubits): leakage to higher-energy states, idle cross-talk, pulse cross-talk, measurement cross-talk.

Depending on the level of accuracy desired to assess the quality of a quantum program on a specific device we can build highly accurate noise models by running characterization experiments. Such experiments provide researchers with information about coherent over-rotations, thermal fluctuations or leakage to spurious states. All of them can be combined in a device-specific noise model that can reproduce the experimental data to high accuracy. In this thesis we develop one of such errors models in a two-qubit superconducting quantum device, modelling all known sources of noise.

1.3.2. Quantum error correction

Since early in the development of quantum computation it was known that quantum states are fragile, and that a successful quantum algorithm will

require the quantum states to be stable for long periods of time. Quantum error correction was developed to overcome such a challenge. Building upon knowledge from classical error correction where additional bits of information are used to store redundant information used to detect errors, quantum error correction codes use additional qubits with the same goal. The first quantum error correction scheme was proposed independently by Shor [57] and Steane [58]. Subsequently, the stabilizer formalism was developed by Gottesman [59, 60] to investigate properties of quantum error correcting codes.

Outstanding progress has been done in the domain of quantum error correction in the past decade. In particular, several experimental realizations of the surface code [61] have been done with superconducting quantum processors [62, 63] and the largest surface code experiment to date [64]. Quantum error correction is a very active research field, constantly making progress towards efficient implementations of quantum error correcting codes for fault-tolerant universal quantum computers. Unfortunately at the time of this thesis the theoretical predictions on the error rates and/or number of qubits required to implement fault-tolerant quantum algorithms are orders of magnitude away from existing state-of-the-art quantum devices. Therefore, it is expected that fully-fledged quantum error corrected machines will be available in the upcoming decades.

1.3.3. Quantum error mitigation

Quantum error mitigation defines a family of techniques that aim at reducing the error on a noisy quantum computation by detecting errors by adding qubits and/or measurements without the ability to correct them. In order to develop a successful quantum error mitigation method one must consider the capabilities of the hardware such that the cost of running the error mitigation does not worsen the unmitigated quantum computation. Recently much progress has been done in developing and improving the performance of noisy quantum algorithm with low-cost overhead error mitigation strategies. Despite the effort in reducing the computational overhead of quantum error mitigation strategies they all require some steps that will take large polynomial or even exponential computational resources with the system size. Therefore, these strategies are thought to be used in computational tasks involving a moderate number of qubits where the cost will still be within the capabilities of the hardware.

A broad range of quantum error mitigation strategies have been proposed following developments of variational quantum algorithm for near-

term quantum hardware. Quantum error mitigation protocols can be split into two groups depending on which errors they target. The first class involves reducing errors at the hardware-level, for example:

- *Zero-noise (Richardson) extrapolation* [49, 50]: a zero-noise extrapolated observable is found by artificially increasing the noise level of the gates and calculating the value of the observable. Then, its zeroth order value is computed by fitting a polynomial function to the measured points.

- *Quasi-probabilistic gate decomposition* [50, 51]: gate-level noise is averaged-out by randomly adding noisy gates to the target circuit such that some of the errors are cancelled. The observables are measured from different random circuits, thus finding an average value of all of them.

Alternatively one can remove errors at the problem-level by exploiting previous knowledge of the target system. Two of the main quantum error mitigation of this type are

- *Symmetry verification* [52–54]: the quantum state at the end of a noisy quantum circuit is checked to verify if it respects the symmetries of the target problem. The state is then projected to the correct sub-space that respects these symmetries.

- *Virtual state distillation* [55, 56]: M copies of the noisy quantum state are used to approximate the pure state by distillation of the leading eigenvalues of the noisy copies. As M increases the quantum state approaches the pure state exponentially fast.

In the upcoming years it is expected that the quantity and quality of the qubits on quantum hardware will be significantly larger. At that point we hope to achieve quantum advantage in specific problems by carefully designing algorithms that reduce the noise on these devices together with powerful quantum error mitigation protocols.

1.4. Outline of this thesis

In the rest of the introduction we provide an overview of this thesis where a brief description of every chapter is presented.

Chapter 2

In this chapter of this thesis we cover the topic of quantum error mitigation to improve the results of faulty quantum computations. We developed a quantum error mitigation strategy that uses the inherent symmetries of a target problem as a way to detect errors during a variational quantum algorithm run. To perform the symmetry-verification step we design shallow quantum circuits that signal if an error occurred by measuring one or more qubits. However such a process incurs on a cost in terms of additional gates and qubits, and might lead to less error mitigation power. To overcome the additional cost we develop a cost-free approach in which symmetry-verification can be done as a post-processing step, and only requires a polynomial number of additional observables.

Chapter 3

This chapter covers the work done in collaboration with an experimental laboratory to demonstrate the power of symmetry-verification on a two superconducting qubit experiment. First we perform an in-depth study of the noise sources affecting the experiment and provide a theoretical prediction of the experiment. Then we study the power of symmetry-verification for the experiment, with a theoretical prediction of the experiment after symmetry-verification. Finally, we compare our predictions with the experimental data finding an outstanding agreement between them.

Chapter 4

The next part of the thesis continues in the domain of near-term quantum algorithms. We focus on how to extract the necessary relevant information of a quantum state without reaching exponentially large times for the task. First we find analytical lower-bounds on the number of independent measurement circuits required to extract elements of k-reduced density matrices in both spin-$\frac{1}{2}$ and fermions. Then, we find a binary partition scheme strategy to design the measurement circuits that is asymptotically optimal for spin-$\frac{1}{2}$ up to logarithmic factors, and matches the best-ever strategy. A similar method is then used to devise measurement scheduling of fermionic 1-, and 2-RDMs, finding an exact solution for the former and an asymptotically optimal solutions for the latter (up to a constant factor).

Chapter 5

In this chapter we focus our attention on the third part of any computation using noisy intermediate-scale quantum hardware, the optimization of variational quantum algorithms. The goal of this chapter is to assess the limitation of existing classical optimization algorithms for the task of optimization a variational quantum algorithm under noise. First, we devise a set of numerical experiments that allow us to compare between several off-the-shelf optimizers across multiple target problems, showing that only two of these methods are able to find accurate results under sampling noise conditions. These two methods are then compared with their best hyper-parameters, finding a comparable performance among them. Our final result is the formal definition of the sampling noise floor: any value within a region defined by the sampling level can be measured as the best-ever result. The sampling noise floor is an artifact that can lead to erroneous parameters during an optimization, greatly affecting the final result of the computation.

Chapter 6

In this chapter we describe a user case test on the Dutch quantum computing facilities Quantum Inspire. The goal of this research is to assess the limitations of the current infrastructure to perform NISQ computations. Our results and conclusion are currently being used to develop the new generation of Quantum Inspire.

Chapter 7

In the last chapter of this thesis we develop quantum algorithms to calculate energy derivatives for quantum chemistry. The theory of this work covers a large amount of resource estimates for several quantum algorithms in both noisy intermediate-scale quantum and fault-tolerance regime. A small test example of how to calculate gradients using Newton's method is shown using simulated and experimental data with an outstanding agreement.