University of Trento

Physics Department

Ph.D. Thesis in Physics

Quantum algorithms for many-body structure and dynamics

Supervisor:
Prof. Francesco Pederiva
Co-supervisor:
Dr. Iacopo Carusotto

Candidate:
Francesco Turro

10 June 2022

Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy

Richard Feynman

Abstract

Nuclei are objects made of nucleons, protons and neutrons. Several dynamical processes that occur in nuclei are of great interest for the scientific community and for possible applications. For example, nuclear fusion can help us produce a large amount of energy with a limited use of resources and environmental impact. Few-nucleon scattering is an essential ingredient to understand and describe the physics of the core of a star.

The classical computational algorithms that aim to simulate microscopic quantum systems suffer from the exponential growth of the computational time when the number of particles is increased. Even using today's most powerful HPC devices, the simulation of many processes, such as the aforementioned nuclear scattering and fusion, is out of reach due to the excessive amount of computational time needed.

In the 1980s, Feynman suggested that quantum computers might be more efficient than classical devices in simulating many-particle quantum systems [1]. Following Feynman's idea of quantum computing, a complete change in the computation devices and in the simulation protocols has been explored in the recent years, moving towards quantum computations. Recently, the perspective of a realistic implementation of efficient quantum calculations was proved both experimentally and theoretically. [2, 3, 4, 5, 6]. Nevertheless, we are not in an era of fully functional quantum devices yet, but rather in the so-called ”Noisy Intermediate-Scale Quantum” (NISQ) era. As of today, quantum simulations still suffer from the limitations of imperfect gate implementations and the quantum noise of the machine that impair the performance of the device. In this NISQ era, studies of complex nuclear systems are out of reach. The evolution and improvement of quantum devices will hopefully help us solve hard quantum problems in the coming years. At present quantum machines can be used to produce demonstrations or, at best, preliminary studies of the dynamics of a few nucleons systems (or other equivalent simple quantum systems). These systems are to be considered mostly toy models for developing prospective quantum algorithms. However, in the future, these algorithms may become efficient enough to allow simulating complex quantum systems in a quantum device, proving more efficient than classical devices, and eventually helping us study hard quantum systems.

This is the main goal of this work, developing quantum algorithms, potentially useful in studying the quantum many body problem, and attempting to implement such quantum algorithms in different, existing quantum devices. In particular, the simulations made use of the IBM QPU's [7], of the Advanced Quantum Testbed (AQT) at Lawrence Berkeley National Laboratory (LBNL)[8], and of the quantum testbed recently based at Lawrence Livermore National Laboratory (LLNL) (or using a device-level simulator of this machine). The our research aims are to develop quantum algorithms for general quantum processors. Therefore, the same developed quantum algorithms are implemented in different quantum processors to test their efficiency. Moreover, some uses of quantum processors are also conditioned by their availability during the time span of my PhD.

The most common way to implement some quantum algorithms is to combine a discrete set of so-called elementary gates. A quantum operation is then realized in term of a sequence of such gates. This approach suffers from the large number of gates (depth of a quantum circuit) generally needed to describe the dynamics of a complex system. An excessively large circuit depth is problematic, since the presence of quantum noise would effectively erase all the information during the simulation. It is still possible to use error-correction techniques, but they require a huge amount of extra quantum register (ancilla qubits).

An alternative technique that can be used to address these problems is the so-called "optimal control technique". Specifically, rather than employing a set of pre-packaged quantum gates, it is possible to optimize the external physical drive (for example, a suitably modulated electromagnetic pulse) that encodes a multi-level complex quantum gate.

In this thesis, we start from the work of Ref. [9], where a quantum simulation of real-time neutron-neutron dynamics is proposed, in which the propagation of the system is enacted by a single dense multi-level gate derived from the nuclear spin-interaction at leading order (LO) of chiral effective field theory (EFT) through an optimal control technique.

Hence, we will generalize the two neutron spin simulations, re-including spatial degrees of freedom with a hybrid algorithm. The spin dynamics are implemented within the quantum processor and the spatial dynamics are computed applying classical algorithms. We called this method classical-quantum coprocessing. The quantum simulations using optimized optimal control methods and discrete get set approach will be presented.

By applying the coprocessing scheme through the optimal control, we have a possible bottleneck due to the requested classical computational time to compute the microwave pulses. A solution to this problem will be presented. Furthermore, an investigation of an improved way to efficiently compile quantum circuits based on the Similarity Renormalization Group will be discussed. This method simplifies the compilation in terms of digital gates.

The most important result contained in this thesis is the development of an algorithm for performing an imaginary time propagation on a quantum chip[10].

It belongs to the class of methods for evaluating the ground state of a quantum system, based on operating a Wick rotation of the real time evolution operator. The resulting propagator is not unitary, implementing in some way a dissipation mechanism that naturally leads the system towards its lowest energy state. Evolution in imaginary time is a well-known technique for finding the ground state of quantum many-body systems. It is at the heart of several numerical methods, including Quantum Monte Carlo techniques, that have been used with great success in quantum chemistry, condensed matter and nuclear physics. The classical implementations of imaginary time propagation suffer (with few exceptions) of an exponential increase in the computational cost with the dimension of the system. This fact calls for a generalization of the algorithm to quantum computers.

The proposed algorithm is implemented by expanding the Hilbert space of the system under investigation by means of ancillary qubits. The projection is obtained by applying a series of unitary transformations having the effect of dissipating the components of the initial state along excited states of the Hamiltonian into the ancillary space. A measurement of the ancillary qubit(s) will then remove such components, effectively implementing a ”cooling” of the system.

The theory and testing of this method, along with some proposals for improvements will be thoroughly discussed in the dedicated chapter.

The thesis is organized as follows:

- Chapter 1. We will introduce to the difficulty of using classical devices to simulate (nuclear) quantum systems
- Chapter 2. We will present the essential elements of quantum computing and we will show why it should be more efficient to use a quantum computer to simulate quantum systems
- Chapter 3. We will show the basic aspects of the hardware of quantum processors based on superconducting device
- Chapter 4 We will start showing how we can simulate the real time evolution through a quantum processor
- Chapter 5. This chapter will present the work on the coprocessing method.
- Chapter 6. We will discuss how complex it is to compile a generic quantum gate in elementary gates. We will present the SRG compiling algorithm to simply the compilation
- Chapter 7. We will present the quantum imaginary time propagation method
- Chapter 8. Here we will summarize the works and will give some future prospective.

Contents

Chapter 1

Introduction

This chapter will present the main problems in simulating quantum systems through a classical device (i.e., personal computers or supercomputers), mainly focusing on nuclear systems.

We will start discussing some interesting problems that we have in contemporary nuclear physics. Later we will briefly discuss some of the main shortcomings of simulating quantum systems in classical devices.

1.1 From quantum chromodynamics to nuclear physics

The Theory of strong interactions is described by quantum chromodynamics (QCD) [11, 12, 13]. QCD's degrees of freedom are quarks (the matter fields) and gluons (the force carriers). It is well-known that two or three quarks can form a bound state. These bound states describe the constituents of the nuclei (protons, neutrons, pions, ...). In particular, hadrons, such as the proton and neutrons, are made of three quarks and a quark and antiquark create a meson, which mediates the interaction between hadrons.

From a theoretical point of view, the QCD theory should predict all the properties of strongly interacting particles. Nevertheless, the equations of QCD are extremely complicated to decipher, even for a simple nucleon, proton or neutron. Indeed, in this regime, we cannot apply perturbation theory and, therefore, we should numerically solve all the equations through the Lattice QCD theory.

The QCD theory has some energy gaps between their different physics. Fig. 1.1 shows the main degrees of freedom one can come in contact with nuclear physics as a function of energy.

When a general theory has some energy gaps (like in our case), phenomena with energy below a specific energy cutoff can be described by an Effective Theory

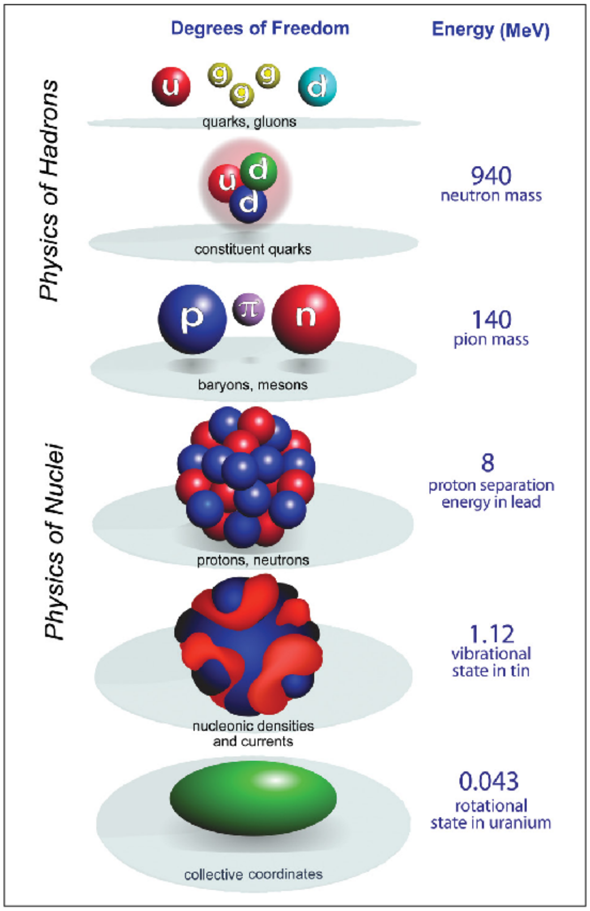


Figure 1.1: From the QCD elements to degrees of freedom of a nuclear calculation. Taken from Ref. [14]

(ET). The underlying interactions above a stabilized energy (cutoff) are described by some effective interaction and parameters (called running coupling constants that depend on the cutoff energy) [15]. Therefore, ET describes the system with some effective degrees of freedom. However, the DoFs depend on the resolution of our experimental microscope scale (in other words, on the cutoff). For the sub-nuclear and nuclear systems, experiments performed with nuclei at energies around 1000 MeV are sensitive to the internal quark-gluon structure of the nucleons. On the contrary, those performed at energies of 100 MeV will see a nucleus as an aggregation of nucleons interacting through complex forces (for instance, a one-pion exchange interaction). Sometimes the nucleons are described by point-like particles. At even lower energies, 10 MeV and below, other degrees of freedom, such as collective rotations and vibrations of the nucleus as a whole, come into play. In the end, if some high energy scale

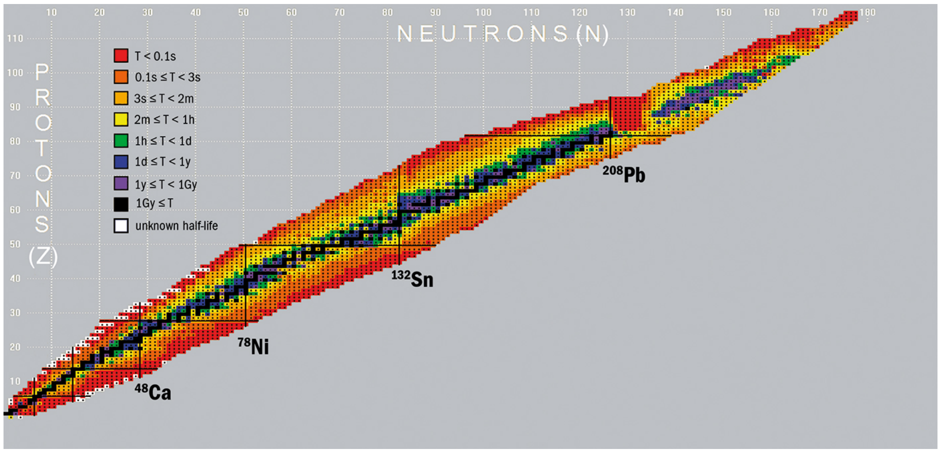


Figure 1.2: Nuclear landscape Chart of nuclides plotted with atomic number (Z) versus neutron number (N), with stable species indicated by black squares. The colours correspond to the decay half-life T. Credits: CERN

exists, we can describe nuclear systems with a smaller set of degrees of freedom than the complete ones coming from QCD.

1.1.1 Nuclear physics' studies

This work will focus on simulating nuclear systems formed by nucleons. Therefore, we are interested in the interaction between nucleons carried by pions, not in the underlying physics of quarks. Our future goal is to completely describe all the nuclei in the nuclides chart starting from the basic fundamental interactions between protons and neutrons. This nucleates chart illustrates the stable and unstable nuclei that we can get, as Fig. 1.2 shows.

Remaining at the level of the physics of nucleons, we can describe interesting and exciting nuclear systems. Many of them are nuclear reactions, which involve the real time dynamics of nucleons. In nuclear reactions, generally, we start from some incoming particles. From simulating their real time evolution, we can understand which particles and energy may be produced. These scattering processes are essential for producing in future green energy (in particular, nuclear fusion) and astrophysical studies.
It is well-known that nuclear reactions occur inside the core of stars (like the Sun). For example, the most known one is the pp-chain 1, the famous nuclear fusion process. When the life of a star ends, we have possible different stages. In one of them, the star becomes a Red Giant. The three-alpha process (or Helium burning reaction) occurs, producing Carbon. Fig. 1.3 shows these two nuclear reactions. These and other scattering processes are critical to understanding what happens inside the stars.

Nuclear reactions involve real time evolution (or dynamics); but, in nuclear physics, we are also interested in studying the structure of nuclei. For example, we want to compute the ground state of a nucleus. Regarding the ground state calculations, there is still an open problem about the composition of the inner core of a Neutron Star [16]. The Neutron Stars are the final stage of some massive stars. They are composed mostly of neutrons. As we said, their inner composition has not been established yet.
It is almost impossible with today's most powerful computers to describe a nuclear complex many-body system. It can even be observed in the difficulty of evaluating the ground state and ground energy of light nuclei with ab-initio calculations. Therefore, the majority of dynamic processes, for instance, nuclear reactions, are out of reach even with the most powerful exascale machines.

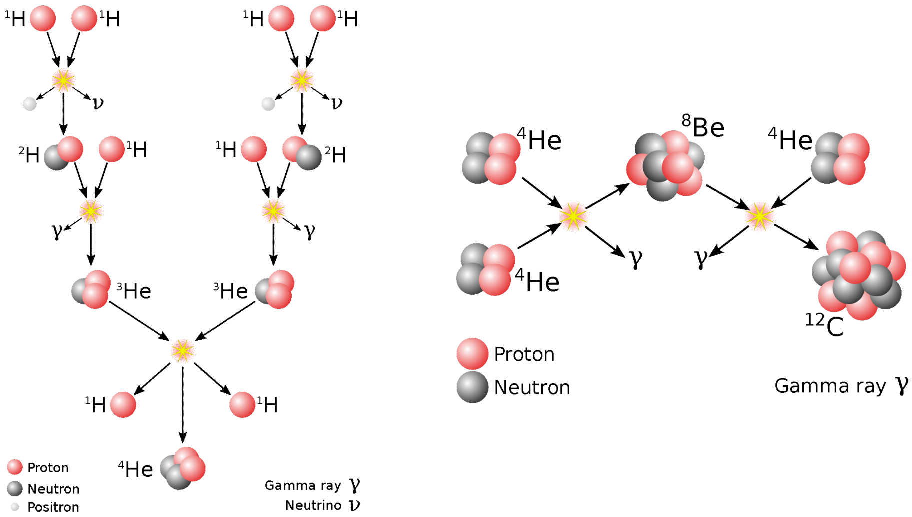


Figure 1.3: Example of nuclear reactions. Left the first pp-chain [17], right the Helium burning process [18]

1.2 Problems in simulating quantum systems in classical devices

The most simple reason for the complexity in simulating a quantum system is the exponential dependence of the Hilbert space on the number of particles. For example, in modeling nuclei, the number of states needed for describing the spin-isospin part depends on 4^A where A is the total number of nucleons. Furthermore, we should also consider the spatial evolution in a dynamical process or in ground state evaluations. Therefore, the number of paths increases enormously and the calculations become very complicated. So, if we increase

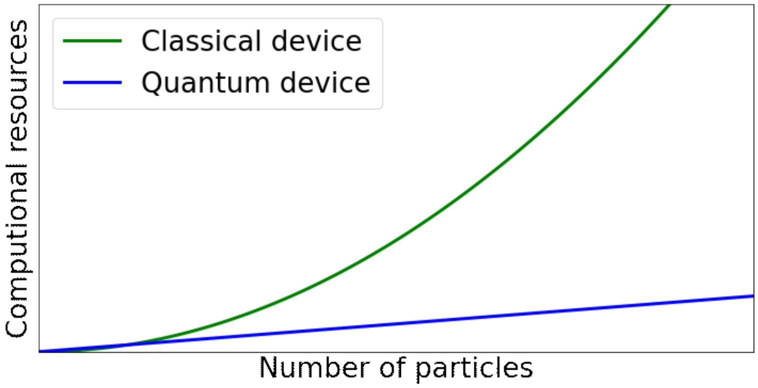


Figure 1.4: Example of dependence of computational resource (for instance, the computational time) of classical and quantum devices on the number of particles

the number of nucleons, we must request immense computational resources to reach an accurate result. Fig. 1.4 illustrates how depends the computational time as a function of the number of particles.

This dependence on enormous computational resources is connected to simulating entangled states. The entanglement in quantum mechanics is associated with correlations among particles. Mapping an entangled state in classical bits, we must consider all the correlations. It requires many powerful computational resources, overall, for highly correlated particles.

Another problem is connected to the nature of physical particles. Quantum particles are either Bosons or Fermions. Therefore, their wavefunction must be symmetric or antisymmetric under the exchange of particles, respectively. Considering that many DoFs are involved in nuclear systems (for instance, spatial, spin and isospin), it is hard to find general states that satisfy the corrected symmetry. Suppose that we compute the ground state of systems of Fermions, we should expand the Hamiltonian through all the antisymmetric states. It is very complicated to obtain all of them. Indeed, one should consider all the mixed antisymmetric combinations of degrees of freedom. Connected to the nature of Fermions, we have the so-called sign problem [19] typical of Monte Carlo methods [20, 21, 22]. These are standard and efficient algorithms to evaluate the ground state of a quantum system. The solution of this problem carries an exponential growth of the computational time due to statistical errors. Indeed, for example, for computing the ground energy, one should compute (see Eq. (9.122) of Ref. [22]):

$$\langle E\rangle = \frac{\langle\psi|\, H\, |\psi\rangle}{\langle\psi|\psi\rangle} = \frac{\int dR^{+}(\psi^{*}_{+}(R^{+})H\psi_{+}(R^{+}) - \psi^{*}_{-}(R^{-})H\psi_{-}(R^{-}))}{\int dR^{+}(\psi^{*}_{+}(R^{+})\psi_{+}(R^{+}) - \psi^{*}_{-}(R^{-})\psi_{(}R^{-}))}\,, \qquad (1.1)$$

where we split the wavefunction ψ when it is positive (R_{+}, ψ_{+}) and when it is negative (R_{-}, ψ_{-}). * symbol indicates the complex conjugate.

The problem arises from the fact that the two addends give the same solution for the antisymmetry of the wavefunction ($f(-x) = -f(x)$). Even though we have the correct energy value, the variance of this error can be high due to something that goes with $\frac{0}{0}$. In ref. [19] shows that the sign problem is a not deterministic polynomial-time hard (Hard NP) problem (unless $P = NP$). The hard NP class indicates the problems that could not be solved in polynomial time.

How can we simulate a complex physical system overcoming the exposed problems? In these years, different types of computers have been developed, the quantum computers. Its physical laws are those of quantum mechanics. Following the famous sentence of Feynman [1], we think quantum computers are more efficient in simulating quantum systems than classical devices. Therefore, using them, we can simplify and speed up our simulations and study complex quantum systems that are very hard or impossible to simulate in today's computers. A natural question can rise from the reader: why is a quantum computer more efficient than a classical computer? To answer that, we must discuss some fundamental concepts of quantum computing. We will answer this question at the end of Ch. 2.

Chapter 2

Elements of Quantum Computing

This chapter will introduce the fundamental concepts of quantum computing, from the definition of a qubit and which kind of operations we can implement in a quantum device. We will discuss DeVicenzo's criteria [23], the necessary conditions to have a working quantum computer. Showing the basis of quantum computing, we will explain why a quantum system simulation is more efficient using quantum processors than classical devices.
We start explaining qubits and their properties. After that, we will introduce the quantum gates and DeVicenzo's criteria. In the end, some quantum algorithms that have been proven to be more efficient than the classical counterparts will be shown.

2.1 Quantum bit

The fundamental unit of classical computational is the well-known bit that exists in two distinct values 0 and 1, usually stored in some part of the memory of the electronic device. A sequence of so-called logical gates describes the operations. A generic logical gate is defined by a Boolean function $\{0,1\}^k \rightarrow \{0,1\}^l$ where k (l) indicates the initial(final) number of bit. The most famous and familiar logical gates are the NOT, OR, AND, XOR, $NAND$ and NOR. These famous logical gates can be put together to create complex classical circuits performing complicated computational tasks. Moreover, one can prove that the combination of NOT, AND and OR gates can be used to describe a generic logical function. From this fact, we call this set of gates a universal gate set [24].
As for classical computational, we must define the fundamental units that carry the quantum information. This is commonly called *"qubit"* or "quantum bit" [25]. It is defined by a quantum system with two distinct levels. Examples

of experimental physical systems that might be used as qubits[1] are the $\frac{1}{2}$-spin particles, the vertical and horizontal (alternatively, left or right) linear polarization of photons and quantum superconducting devices.

According to quantum mechanics, the general state that describes the qubit is given by the superposition of the two states. Using the standard notation, the ground state of the two-level quantum system is named $|0\rangle$ and the excited state $|1\rangle$. Therefore, using the Dirac notation, the qubit state is given by

$$|\psi\rangle \,=\, \alpha\,|0\rangle + \beta\,|1\rangle \ , \tag{2.1}$$

where α and β are complex number and $|\alpha|^2 + |\beta|^2 \,=\, 1$.

A first difference respect to classical computations is that the bit can be found only in 0 or 1, the qubit is in a continuous mixture of such two states. In particular, according to equation (2.1) the probabilities to measure the $|0\rangle$ and $|1\rangle$ states are given by $|\alpha|^2$ and $|\beta|^2$, respectively.

We found that a two-level quantum system defines a qubit. Therefore, its Hilbert space is described by $\mathbb{C}^2$. More generally, we may also fabricate a quantum processor with $N > 2$ states. Hence, the Hilbert space of the simulation is described by $\mathbb{C}^N$, and the general state into the quantum processor would be in a superposition of these N states. In standard textbooks, we name the quantum processor with 3 states qutrit. Moreover, one with a number of states $\geq$ 4 is called qudit. The word qubit will generally describe N level system in this thesis. When we do not specify the number of states, a qubit will be composed of two states; otherwise, we will specify the number of states.

Moreover, one can couple N qubits through some connections or interactions (i.e. a electromagnetic signal). The corresponding state of this system is represented by the tensor product of the 2^N states:

$$|\psi\rangle \,=\, \otimes_i^N c_i\,|\psi_i\rangle = \sum_{i=0}^{2^N} \alpha_i\,|i\rangle \ , \tag{2.2}$$

where $|\psi_i\rangle$ indicates the i qubit state. On the right side of this equation, we write the Fock state with 2^N states, and the i index, representing the state i, can be written in decimal or binary notation. For example the state given by $|0\rangle \otimes |1\rangle \otimes |0\rangle$ can be written as $|2\rangle$ or $|010\rangle$.

The coupling of states can be advantageous in quantum computing because we obtain the entanglement property. Indeed, a general state in Eq. (2.2) can be an entangled state or a pure state. In elementary words, given a state that describes the two systems A and B, this is pure if a measure on A does not attach the information contained in the B state. Otherwise, it is an entangled state. The entanglement property is another difference with classical computation. It is very helpful in quantum computing because it can connect different qubits without any actual "wires" between them. In quantum mechanics, one finds that the entanglement is connected with the correlations of states.

[1] The necessary conditions when a generic quantum system can be a quantum processor will be described by Sec. 2.6

Here, we will show the entanglement concept with simple examples. Suppose we have two states of two system:

$$|\psi_1\rangle \;=\; \frac{(|0\rangle_A + |1\rangle_A)}{\sqrt{2}}\frac{(|0\rangle_B + |1\rangle_B)}{\sqrt{2}} \qquad\qquad |\psi_2\rangle \;=\; \frac{(|00\rangle + |11\rangle}{\sqrt{2}} \,. \tag{2.3}$$

If A measures its states, A obtains the 0 state with a probability of one-half. For $|\psi_1\rangle$ the final state is untouched by the measure, it still is in $\frac{(|0\rangle_B+|1\rangle_B}{\sqrt{2}}$. Instead, for $|\psi_2\rangle$, after the measure, the system collapses in the state $|00\rangle$, so the final state for B is the $|0\rangle$ state, not more than the $|0\rangle + |1\rangle$ state. A measure in a part of the sub-system changes the other sub-system parts.

The general mathematical definition of when a state is entangled or not is based on the Schmidt decomposition [26, 27]. Using it, we start from a general state $|w\rangle \in H_1 \otimes H_2$, where H_1 and H_2 are the two generating Hilbert spaces. We write it as a sum of tensor product of states of $|u_i\rangle \in H_1$ and $|v_j\rangle \in H_2$, so,

$$w \;=\; \sum_i c_i \, |u_i\rangle \otimes |v_j\rangle \;, \tag{2.4}$$

where $\{c_i\}$ is the set with the minimum number of these coefficients. Moreover, they are unique, real and positive.

The number of coefficients in the sum we need to write this equation is named Schmidt number or rank. If the rank is 1, the state is pure; otherwise, it is an entangled state. Therefore, an entangled state is always defined by a sum of at least two pure states.

2.1.1 Qubit representations

Here, we will describe how we can represent the qubit state.

We have defined the qubit as

$$|\psi\rangle \;=\; \alpha\,|0\rangle + \beta\,|1\rangle \;, \tag{2.5}$$

where α, $\beta \;\in\; \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 \;=\; 1$.

An other standard representation uses spinors. In particular, we define:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad\qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \,. \tag{2.6}$$

So, my generic qubit state is given by:

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \,. \tag{2.7}$$

We can notice that the number of free parameters of Eq. (2.5) is three, because the two complex numbers are defined by four numbers, but we have the probability condition, $|\alpha|^2 + |\beta|^2 = 1$. Therefore, we can rewrite Eq. (2.5) substituting

$$\begin{aligned} \alpha &= e^{i\delta} \cos\left(\frac{\theta}{2}\right) \\ \beta &= e^{i(\delta+\phi)} \sin\left(\frac{\theta}{2}\right) . \end{aligned} \tag{2.8}$$

So, we obtain

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) e^{i\phi} |1\rangle \ , \tag{2.9}$$

where we neglected a global phase δ because it does not have a physical significance. Indeed, it is the global phase and it can not be experimentally established.

Looking closer at Eq. (2.9), we can notice that the two angles, θ and ϕ, describe the qubit state. Therefore, a generic qubit state is represented by a point on a sphere of radius 1 that can be parametrized by two angles θ and ϕ. This sphere is called the Bloch sphere.

There are fundamental states in the Bloch Sphere. Two of them are $|0\rangle$ and $|1\rangle$, representing the two poles of the sphere. Other important states are the $|\pm x\rangle$ states. They are defined by Eq. (2.10). These states are shown in the Bloch sphere of Fig. 2.1. They are important in quantum cryptography.

$$|x\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \qquad |-x\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} . \tag{2.10}$$

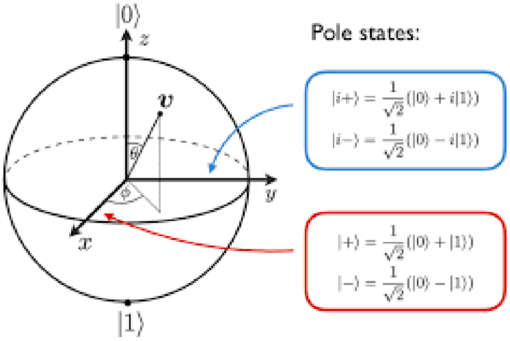

$|0\rangle$
z
v
θ
ϕ
y
x
$|1\rangle$
Pole states:
$|i+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$
$|i-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$
$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

Figure 2.1: Bloch sphere. Some important states are shown. The general state v is defined by the two angles θ and ϕ. Picture is taken from [28]

2.2 Quantum operations: Quantum gates

After defining qubits, we must explain the operations in quantum computing that manipulate the quantum information. A linear quantum operator describes a general operation in quantum mechanics, which transforms an initial state to another. There is a convenient representation of linear quantum operators through matrix forms like quantum states. Indeed, matrices transform vectors into other vectors.

A linear quantum operator must describe an operation between qubits from the point of view of quantum mechanics. It turns out there is solely another constraint required to define a quantum gate, the real operation implemented in a quantum processor. In the definition of the qubit, we have the constraint that the sum of the probabilities of measure $|0\rangle$ and $|1\rangle$ must always be equal to 1. Therefore, we have to preserve the total probability of the qubits during the operations. There is a class of operators in quantum mechanics that preserves the norm of a state, i.e. *unitary operators*. An operator U is unitary if and if only if

$$U^\dagger U = 1\,, \tag{2.11}$$

where $U^\dagger$ is the adjoint of U. Eq. (2.11) guarantees us that the total probability is preserved during the operations.

This unitary condition is the only constraint on a perfect quantum gate. In standard notation, we often want to specify onto which the gate is applied, so we also added a label indicating such number of qubits. For example, a unitary operator applied to a single qubit is called a single qubit gate, to two qubits a two-qubit gate and three qubits a three-qubit gate.

Mathematically, a quantum gate that manipulates n qubits is defined by an element of $SU(d = 2^n)$ group, where $SU(d)$ is the Special Unitary group.

Like for classical computation, we need to translate a generic quantum algorithm into a sequence of quantum gates, generally implementing single or two-qubits unitary operations. In the graphical representation of operations in classical computational, wires(lines) describe the bits story. To apply specific logical gates to a specific sub-set of bits, we draw specific boxes and symbols (that corresponds to the specific operation) to the lines of the bits of the sub-set. There is an analog representation of the quantum counterpart. We draw lines for the qubits and specific symbols (generally simple boxes) for the quantum gates. A quantum algorithm would be drawn as a sequence of lines and boxes that indicate the story of quantum operations. The sequence of quantum gates that describes a generic quantum algorithm belongs to the *quantum circuit* of the quantum algorithm. Like in classical computation, there may be different equivalent quantum circuits for the same quantum algorithms.

Having only the unitarity constraint, one can implement infinite possible gates in a quantum processor. Nevertheless, generally, one implements the standard quantum gates shown in Tab. 2.1. They are the most common and fundamental operations.

Pauli x-gate	$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$-\boxed{X}-$

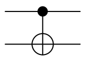
Pauli y-gate	$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$-\boxed{Y}-$
Pauli z-gate	$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	$-\boxed{Z}-$
Rotation along x axis	$R_x(\theta) = e^{-iX\theta/2} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right) \\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$	$-\boxed{R_x}-$
Rotation along y axis	$R_y(\theta) = e^{-iY\theta/2} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$	$-\boxed{R_y}-$
Rotation along z axis	$R_z(\theta) = e^{-iZ3\theta/2} = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix}$	$-\boxed{R_z}-$
Hadamard gate	$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	$-\boxed{H}-$
U_3 gate	$U_3(\theta, \phi, \lambda) = \begin{pmatrix} \cos(\theta) & -e^{i\lambda}\sin(\theta) \\ e^{i\phi}\sin(\theta) & e^{i(\phi+\lambda)}\cos(\theta) \end{pmatrix}$	$-\boxed{U_3}-$
CNOT	$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	

Table 2.1: Most common quantum gates

The last row of Tab. 2.1 shows the $CNOT$ gate. It is the simplest element of the so-called controlled operations family. The quantum gates of such class are essential in quantum computing because their action produce entangled states. A controlled quantum gate has at least two qubits as input and output. Some of these are the so-called ”control” qubits, and the others are called ”target” qubits. The operation that is applied to the target qubits depends on the state of the control qubits. Generally, a U-controlled operation (where U is a unitary operator) performs the U gate on the target qubits if all the states of the target qubits are in the $|1\rangle$ state. Otherwise, we apply the identity $\mathbb{1}$ on the target qubits. Specifically, one finds that the $CNOT$ gate is an X-controlled operation. Specifically, if the target qubit is $|0\rangle$ (first and second rows of the matrix of Tab. 2.1), we will apply an identity on the target state. Instead, we apply the X gate (third and fourth rows). We can also write the $CNOT$ as follows

$$CNOT = P_0 \otimes \mathbb{1} + P_1 \otimes X\,, \tag{2.12}$$

where P_0 and P_1 indicate the projection operator in the $|0\rangle$ and $|1\rangle$ state respectively. For a single qubit, they are defined by:

$$P_0 = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad P_1 = |1\rangle\langle 1| = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}. \tag{2.13}$$

A controlled operation is represented in a quantum circuit by a black dot for the controlled qubits. The dots of controlled qubits are connected by a vertical line and this line ends with a box on the target qubits where inside it is specified the U operations. For the CNOT gate, the box is replaced by a tensor sum symbol. Instead, in some cases, we want to apply the U operation when all the controlled qubits are in $|0\rangle$. For representing it, we change the dots in the controlled qubits in white circles.

One can observe that some gates in Tab. 2.1 can be obtained by other gates of the same table. For example, the X, Y and Z are obtained respectively from R_x, R_z and R_z with an angle $\theta = \pi$.

A generic single qubit quantum gate (i.e., a generic operation on the Bloch Sphere) can be described with a U_3 quantum gate, defined by

$$U_3(\theta, \phi, \lambda) = R_z(\phi)\, R_x\left(-\frac{\pi}{2}\right) R_z(\theta)\, R_x\left(\frac{\pi}{2}\right) R_z(\lambda)\,. \tag{2.14}$$

There is another expression for the U_3 gate called Euler Decomposition. Its decomposition is given by [25, 29]

$$U_3(\theta, \phi, \lambda) = R_z\left(\frac{\phi}{2}\right) R_y(\theta)\, R_z\left(\frac{\lambda}{2}\right). \tag{2.15}$$

Suppose that we have a generic quantum operator U and we translate it through a quantum circuit: how can we quantify if this circuit correctly reproduces the operator U? A quantitative estimate is usually made in terms of the ”fidelity” of the circuit. We will discuss the fidelity later in Sec. 2.5.

2.2.1 Quantum universality

In classical computing, we can decompose a generic logical gate to a specific finite combination of AND, OR and NOT gates. This primitive gate set is called universal.

One finds sets of primitive quantum gates in quantum computing whose finite sequence can describe all generic quantum gates. This statement is called quantum universality.

The set formed by R_x, R_z and $CNOT$ gates is the most common set of universal quantum gates. Indeed, these gates are the fundamental gates implemented in several quantum processors.

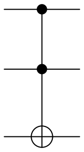
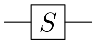
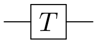
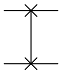
Toffoli gate	$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$	
Phase gate	$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	
$\pi/8$ gate	$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$	
SWAP	$SWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	

Table 2.2: Toffoli, $\pi/8$, phase and SWAP gates

In the literature, one can find other universal gate sets. One is given by Hadamard, $\pi/8$, phase and CNOT gates; another is obtained from Toffoli, Hadamard, phase and CNOT gates. The phase, $\pi/8$ and Toffoli gates are shown in Tab. 2.2.

Nevertheless, we have a problem with quantum universality. We know that a generic operator can be translated into a sequence of elementary gates. However, we do not know if the number of requested quantum gates is finite or not. There is a very useful theorem in the literature, the Solovay–Kitaev theorem [30, 31] that can solve this problem. It guarantees that a generic quantum gate applied to a constant number of qubits can be efficiently approximated by a sequence of elementary gates. In particular, its formulation of Ref. [32] says:

Theorem 2.2.1. *Let G be an instruction set for $SU(d)$, and let the desired accuracy $\epsilon > 0$ be given. There is a constant c such that for any $U \in SU(d)$ there exists a finite sequence of gates from G of length*

$$O\left(log^c\left(\frac{1}{\epsilon}\right)\right) \tag{2.16}$$

such that the operator distance $d(U, S) < \epsilon$.

In this case, the distance $d(U, S)$ between the two operators U and S is defined as $d(U, S) = \sup_{\|\psi\|=1} \|(U - S)\|$. The instruction set G indicates the universal set of quantum gates operating in the Hilbert space of dimension d. Specifically, Ref. [32] defines it as a finite set of quantum gates such that

1. All gates in G are $SU(d)$.
2. For each gate in G, its inverse gate is in G
3. G is a universal set for $SU(d)$.

The theorem states that the cost of approximating a quantum gate is a polynomial of the logarithm of the inverse of error. Ref. [32] shows that the upper bound of the exponent c of the logarithm in Eq.(2.16) is equal to $c = 3.97$.

2.2.2 Ancilla qubits

To give all the essential elements of quantum computing, we should deal with the concept of ancilla qubits, or simply ancillas. The use of ancillary qubits is widespread in quantum algorithms.

In the standard terminology, ancilla qubits indicate extra qubits that we add to the simulation to achieve a specific implementation or aim of a quantum algorithm. Practically, their purpose is to extend the Hilbert space of the system carrying the information. Their use helps us in the quantum calculation, simplifying the compilation or allowing for some particular operation.

For example, the ancilla qubits are used in error-correcting algorithms, which correct the results from the contribution of noise presented in quantum processors to the simulation. We will utilize ancillary qubits in the quantum algorithm of Ch. 7.

2.3 Measurement

After implementing the desired quantum algorithm on some qubits, we would extract some information from the qubits. To do that, we must employ a (quantum) measurement on the quantum processor. Generally, in quantum computing, measurement gives us back the probability distribution of the qubits on a computational basis. A measurement by its nature is a not-unitary and irreversible operation. After that, we cannot go back to the state before the measurement.

The symbol of a measurement in the graphical representation is represented by a "meter". Fig. 2.2 shows it.

The general definition of measurement comes from the postulates of quantum mechanics. Specifically, the measurement is defined by a set $\{M_m\}$ of observables such that after the measurement, we get the resulting outcome m. Some examples of observables of a quantum system are energy, momentum and spatial

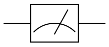


Figure 2.2: Meter symbol for the quantum measure

distribution. If the state before the measurement is $|\psi\rangle$, after it, we may obtain as a result for the outcome m with probability

$$p_m = \langle\psi| M_m^\dagger M_m |\psi\rangle \,. \tag{2.17}$$

The final state would be the collapse of $|\psi\rangle$ in

$$|\psi_m\rangle = \frac{M_m |\psi\rangle}{\langle\psi| M_m^\dagger M_m |\psi\rangle} \,. \tag{2.18}$$

The operators $\{M_m\}$ must satisfy

$$\sum_m M_m^\dagger M_m = \mathbb{1} \,. \tag{2.19}$$

There are two different measurements in quantum mechanics: the Projection Valued Measurement (PVM) and Positive Operator-Valued Measure (POVM).

2.3.1 Projection Valued Measurement(PVM)

We can write the set $\{M_m\}$ of a generic projective measurement described as

$$M_m = P_m \,, \tag{2.20}$$

where m indicates the outcome of M_m and P_m is the projection operator to the subspace of M with the m eigenvalue. The projector operator is defined by the operator such that $P_m^2 = P_m$.

The set of projection operators P_m forms an orthogonal and complete set. Therefore, for a discrete set $\{P_i\}$, we have $P_m\, P_n = \delta_{nm}$ and $\sum_m P_m = \mathbb{1}$. After measuring, suppose we obtain the outcome m, iterating measurements with the same observable we always obtain the m outcome. This fact is due to the orthogonal condition of the projector.

The PVM definition describes the standard measurement that one finds in quantum mechanics and quantum computation. Specifically, the general measurement implemented in all the QPUs is to measure the probability distribution of qubits in the computational basis $|0\rangle\,, |1\rangle\,, ...$. In some QPUs, there is the possibility of changing the measurement's computational basis. For example, a useful computational basis in quantum cryptography is described by $|\pm\, x\rangle$ states.

2.3.2 Positive Operator-Valued Measure(POVM)

For many applications, we are interested in the probabilities of the outcomes, not in the post-selected state. A mathematical tool named POVM describes a measurement procedure where we have the probability of outcome without entirely collapsing the state.
The measurement operators of POVM set must obey solely to Eq. 2.19. The POVM is a generalization of PVM where we lose the orthogonality condition.

2.4 Density matrix formulation

In the previous sections, the basic elements of quantum computing were presented in the state vector formulation. In many cases, it is convenient to use a different formulation, the density matrix one. This is very useful when we include the noise of the quantum processor because it involves not-unitary and irreversible processes.
We start from the link between density matrix and vector states. The density matrix ρ of a pure state $|\psi\rangle$ is given by

$$\rho = |\psi\rangle \langle\psi| \, . \tag{2.21}$$

The generalization of the formulation of a density matrix is given by Eq. (2.22) considering an ensemble of pure states $\{p_i, |\psi_i\rangle\}$.

$$\rho = \sum_i p_i \, |\psi_i\rangle \langle\psi_i| \, . \tag{2.22}$$

A general density matrix ρ is defined by an operator if and only if it satisfies

1. $\mathrm{Tr}(\rho) = 1$.
2. ρ is a positive operator

The quantum gates are still represented by unitary operators. Their action on the density matrix ρ is given by

$$\rho \xrightarrow{U} \rho_1 = U \rho U^\dagger \, . \tag{2.23}$$

The measurement in density matrix formulation becomes for the probability

$$p_m = \mathrm{Tr}\left\{M_m \rho \, M_m^\dagger\right\} = \mathrm{Tr}\left\{M_m^\dagger \, M_m \rho\right\} \tag{2.24}$$

and the final density matrix

$$\rho_1 = \frac{M_m \rho M_m^\dagger}{\mathrm{Tr}\left\{M_m \rho M_m^\dagger\right\}} \, . \tag{2.25}$$

In the last two equations M_m and m represent the collection of measurement operators and their experimental outcomes, respectively.
Studying complex quantum systems, the density matrix formulation is very convenient for distinguishing between pure and entangled states. Considering a general density matrix ρ, one finds that $Tr\left[\rho^2\right] \leq 1$. Moreover, one proves that $Tr\left[\rho^2\right] = 1$ if and if only if ρ is a pure state. Therefore, if the result of $Tr\rho^2$ is 1, we have a pure state, otherwise, ρ is a entangled state if $Tr\left[\rho^2\right] < 1$.
The last operation about density matrix formulation that we will discuss is the reduced density operator. Considering two systems A and B, we have the density matrix ρ^{AB} that describes the two systems. The reduced density matrix of A is defined by

$$\rho_A = \mathrm{Tr}_B\left[\rho^{AB}\right] = \sum_{\psi_B} \langle\psi_B|\,\rho^{AB}\,|\psi_B\rangle \ , \tag{2.26}$$

where $|\psi_B\rangle$ are generic generators of B. Applying this operation, we focus on the system A regardless of the state of B.

2.5 Fidelity

Fidelity is a quantum property that quantifies how two states or two operators are close. One possible definition for states in the Hilbert space is given by

$$F(\sigma,\rho) = \left(Tr\left[\sqrt{\sigma^{\frac{1}{2}}\rho\sigma^{\frac{1}{2}}}\right]\right)^2 , \tag{2.27}$$

where ρ and σ are density matrices of two states. We should apply a different definition for computing the closeness of two operators.
One can prove that the fidelity has the following properties.

- Symmetry: $F(\sigma,\rho) = F(\rho,\sigma)$.
- It has a lower and an upper bound: $0 \leq F(\rho,\sigma) \leq 1$.
- For two pure states ψ and ϕ: $F(\psi,\phi) = |\langle\phi|\psi\rangle|^2$
- Unitary invariance: $F(\rho,\sigma) = F(U\rho U^\dagger, U\sigma U^\dagger)$
- $F(\rho,\sigma) = 1$ if and only if $\rho = \sigma$. This is a consequence of Uhlmann's theorem [33].

The definition of fidelity to measure the closeness of two operators is given by [34]

$$F(U_1,U_2) = \langle U_1|U_2\rangle\,\langle U_2|U_1\rangle \ , \tag{2.28}$$

where $\langle U_1|U_2\rangle$ is defined as:

$$\langle U_1|U_2\rangle = \frac{1}{N}\,Tr(U_1^\dagger U_2)\,, \tag{2.29}$$

with N the dimension of the two operators and Tr the trace operation.
One can prove that the definition of Eq. (2.28) is invariant under a global phase change of two operators. Finally, using this definition we recover the the last properties of the state fidelity in the bullet points. If two operators U and V are the same if and only if $F(U, V) = 1$. The translation of a unitary operator O in a quantum circuit represented by an operator Q gets better if the fidelity $F(O, Q)$ tends to 1.

2.6 Necessary conditions to get a quantum computer

A common question concerns whether a generic quantum system can easily ask when a generic quantum system can be used as a quantum processor (QPU). There are different answers to this questions. DiVincenzo et al. in Ref. [23] found five essential criteria when a quantum system can become a quantum processor. Here, we will show and discuss them.

1. A scalable physical system with well-characterized qubits

This first criterion seems very obvious. The first requirement of a quantum computer is to have its fundamental units, the qubits implemented by a physical system. This simple criterion can be experimentally very problematic in the words "well-characterized". Indeed, we must know the entire Hamiltonian that rules the quantum systems with its internal parameters. For example, we must characterize all the interactions and couplings among all quantum states. A quantum gate is implemented in all quantum machines by coupling a subpart or the whole qubits with a time-dependent external field (for instance, an electromagnetic pulse). The resulting operator that describes the evolution of the quantum processor would be the desired quantum gate. A further discussion is found in Sec. 3.2. Moreover, one should also characterize the quantum machine's noise contribution to get a full characterization. This experimental characterization can be very challenging.

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

Suppose the initial state of a simulation is in some wrong state. In that case, the result of the application of our transformation will not be the expected one. Many error-correcting codes, which are routines to correct the simulations from the contributions of noise sources, are based on the assumption that the initial state can be exactly reached. Therefore, we must prepare the quantum processor at least in an initial state that can be precisely characterized.

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

This criterion indicates we cannot have accurate results from a quantum algorithm if the quantum noise in the quantum processor destroy the coherence of the physical qubit. Furthermore, a good knowledge of how the system loses its coherence can be very useful in finding quantum error-correcting code that can help in the quantum simulation. The simulations of this work are implemented in quantum processors based on superconducting devices. They have a decoherence time of order of 10-100 μs currently; the time for applying a quantum gate is $10 - 100$ ns. From this fact, we can be sure that a small number of quantum gates can be implemented in today's quantum machines. We will discuss deeper some aspects of noise source contribution in Ch. 3.

4. A "universal" set of quantum gates

This criterion declares that we need to translate a quantum algorithm in some quantum operations that the quantum machine can perform. There are two approaches in implementing a general gate: the analog quantum simulator and the digital quantum simulator. The analog quantum simulation is based on a quantum processor whose dynamics look like the action of the desired implemented quantum gate. We will find a deeper discussion in Sec. 3.2. Instead, the digital quantum simulator is based on compiling operations using universal gate sets.

5. A qubit-specific measurement capability

Also, this requirement is self-evident: after performing the operations, we would read out the results. Therefore, we request the ability to measure the probabilities of qubits. This measuring process must be independent of the states of qubits, and it must not change the rest of the quantum system.

2.7 Is quantum computing more efficient than classical computation?

Unluckily, we cannot answer this question with mathematical proof. Indeed, Preskill declares in Ref. [4] *We cannot yet prove this claim, either mathematically or experimentally, but we have reason to believe it is true; arguably, it is one of the most interesting distinctions ever made between quantum and classical. It means that well-controlled large quantum systems may "surpass understanding," behaving in ways we find surprising and delightful.*

Some quantum algorithms in the literature have been proven to be more efficient than their classical counterpart. The most famous are those of Shor [6] and Grover [5]. Furthermore, an experiment proving this statement was done by a research team based on Google [2]. We will discuss these algorithm in the following subsections. In the end, another simple example of quantum efficiency is presented, the position in a sphere.

Shor's algorithm

Shor's algorithm [6] describes an efficient way to compute the integer factorization on a quantum computer. The integer factorization involves finding the prime factor of an integer number N. Using a quantum computer, one finds the order of the computational time (that is connected to the number of quantum gates) of Shor's algorithm is given by

$$O(\log(N)^2 \log(\log(N)) \log(\log(\log(N)))) \, , \tag{2.30}$$

where N is the integer to factorize.
On the contrary, in classical computation the most efficient algorithm [35] requests a computational time of order:

$$O(e^{1.9 \log(N)^{\frac{1}{3}} \log(\log(N))} \, . \tag{2.31}$$

We can see that the quantum computational time is exponentially shorter than one of classical computational.

Grover's algorithm

Another quantum algorithm that is proved to be much faster than a classical one is Grover's one [5]. He developed a method for the unstructured search of a database. The quantum computational time is of order $O(\sqrt{N})$. Instead, it is $O(N)$ on a classical device.

Experiment of Google

The work of Ref. [2] is the most famous experimental proof of quantum supremacy. We call quantum supremacy the tasks implemented by quantum devices that classical computers cannot perform in any feasible amount of time and memory. Google's authors used 53 superconducting qubits to compare the computational time obtained between quantum and classical computers in sampling the output of a pseudo-random quantum circuit. The quantum computational time takes 200 seconds instead of 10 000 years on a supercomputer in its experiment.

Point on a sphere

A trivial example of quantum efficiency is the determination of the position of a point in a sphere (for example, a position on the Earth). With a single qubit, we can describe this position. It is simply the position of the qubit on the Bloch sphere. On the contrary, we must describe the coordinates at least with two angles in a classical computer. So, at least we must duplicate the computational power.

Some evidence of quantum efficiency have just been discussed. Going back to the original problem of overcoming the present issues of classical device in simulating quantum systems, we have some signs that a quantum simulation in a quantum processor should solve some problems. However, we must be attention to declare that quantum computers are the solutions because so far there is not a clear proof. Moreover, this topic is still being debating in the scientific community.

2.7.1 Quantum simulations

We have seen in Ch. 1 that the most powerful supercomputers have many problems in simulating quantum systems. Specifically, the main one is the exponential growth of computational resources with the number of particles for describing their high-entanglement

From a theoretical point of view, a possible solution of simulating efficiently quantum systems may be found on what Richard Feynman stated [1]. He declared that a quantum machine should be more efficient in simulating a quantum system than a classical device because the quantum processor applies the same physical laws of the systems and nature. Indeed, it is very straightforward to obtain entangled states in a quantum processor. For example, it is sufficient to apply a $CNOT$ gate to a generic pure state. Instead, in a classical device, we should spend a lot of computational resources to describe the same entangled state.

Along this direction, Ref. [36] says that a quantum processor with 50 qubits represents a milestone of quantum simulations between classical and quantum computers. Indeed, it states that we cannot simulate by brute force using the most powerful supercomputer quantum systems with 50 qubits. Quantum computers would easily have more than 50 qubits in the future, overcoming the most advanced classical resources.

In Ch. 1 we discussed that the nature of particles could be a problem in simulating quantum systems. In particular, this issue is represented by the Fermion sign problem. This problem in quantum simulations may be eliminated by mapping only the antisymmetric states in the qubit states.

However, a theoretical or mathematical proof is not present in the literature [4]. We should also consider that we could have the possibility to have a classical algorithm that may simulate a quantum system more efficiently than all quantum algorithms in the future. Despite this, we are at a point in which it is becoming absolutely worth studying a quantum system in a quantum processor. Indeed, we can find situations and physical problems were it is better to use the quantum processor and in others classical devices. In any case, quantum computing is a fresh way of rethinking problems, which might lead to better performing classical algorithms.

Chapter 3

Basic aspects of quantum hardware

This chapter will review some fundamental hardware elements of quantum computers based on superconducting devices. The quantum processors of IBM's[7] and the quantum testbed of LLNL [37] are made of transmon devices [38].
Transmons are essentially LC-like circuits where the inductance is replaced by a Josephson Junction [39, 40], a non-linear element. We will investigate this circuit showing some of its fundamental aspects. Some devices based on Josephson Junctions will be presented. However, we will mostly focus more on the transmon.
After showing that the transmon is a good candidate for a qubit, we will discuss how we can experimentally implement a quantum gate. At this point, we will also discuss noises sources. At the end of this chapter, we will see the readout procedure. It is the method to obtain the probabilities from a quantum processor.
This chapter is based mostly on Ref. [41, 42].

3.1 From LC circuit to transmon

We start from the Lagrangian of LC circuit, where L and C are an inductor and a capacitor, respectively:

$$L_{LC} = \frac{1}{2}C\left(\frac{d\Phi}{dt}\right)^2 - \frac{1}{2\,L}\Phi^2\,. \tag{3.1}$$

From this equation using the Legendre transformation we can get the Hamiltonian of the LC circuit:

$$H_{LC} = \frac{1}{2}\,C\,V^2 + \frac{1}{2}\,L\,I^2 \tag{3.2}$$

where

$$Q = \frac{\partial L}{\partial \frac{d\Phi}{dt}} = C \frac{d\Phi}{dt} \tag{3.3}$$

indicates the charge operator and is the momentum conjugate to the flux.
We can notice that H_{LC} is a harmonic oscillator Hamiltonian with $m = C$ and $\omega = \frac{1}{\sqrt{LC}}$. To write the quantum version of this Hamiltonian, we must compute the Poisson bracket:

$$\{\Phi, Q\} = \frac{\partial \Phi}{\partial \Phi}\frac{\partial Q}{\partial Q} - \frac{\partial Q}{\partial \Phi}\frac{\partial \Phi}{\partial Q} = 1 - 0 = 1 \tag{3.4}$$

and associate the commutator relation

$$\left[\hat{\Phi}, \hat{Q}\right] = i\,\hbar\,, \tag{3.5}$$

where now $\hat{\Phi}$ and $\hat{Q}$ are quantum operators. From now on, we will drop the hat symbol for simplicity and clearness.
We define the reduced flux $\phi = \frac{2\pi\Phi}{\Phi_0}$ and the reduced charge $n = \frac{Q}{2\,e}$ where Φ_0 is the quantum superconducting magnetic flux $\Phi_0 = \frac{h}{2e}$ and e is the electric charge. We used the value $2\,e$ because in these regime we can assume that the electrons form Cooper-pairs. Hence, we can rewrite eq. (3.2) using these new definitions as:

$$H = 4\,E_C\,n^2 + \frac{1}{2}E_L\,\phi^2\,, \tag{3.6}$$

where $E_C = \frac{e^2}{2\,C}$ is the energy required to add a Cooper pair to the capacitor, and $E_L = \frac{\phi_0^2}{(2\pi)^2\,L}$ is the inductive energy. We have that $[\phi,\,n] = i$.

As we said, the Hamiltonian of eq. (3.6) is the same of one of 1 dimensional Harmonic Oscillator with $\omega = \frac{1}{\sqrt{LC}}$. Therefore, using creation and annihilation operators, we get:

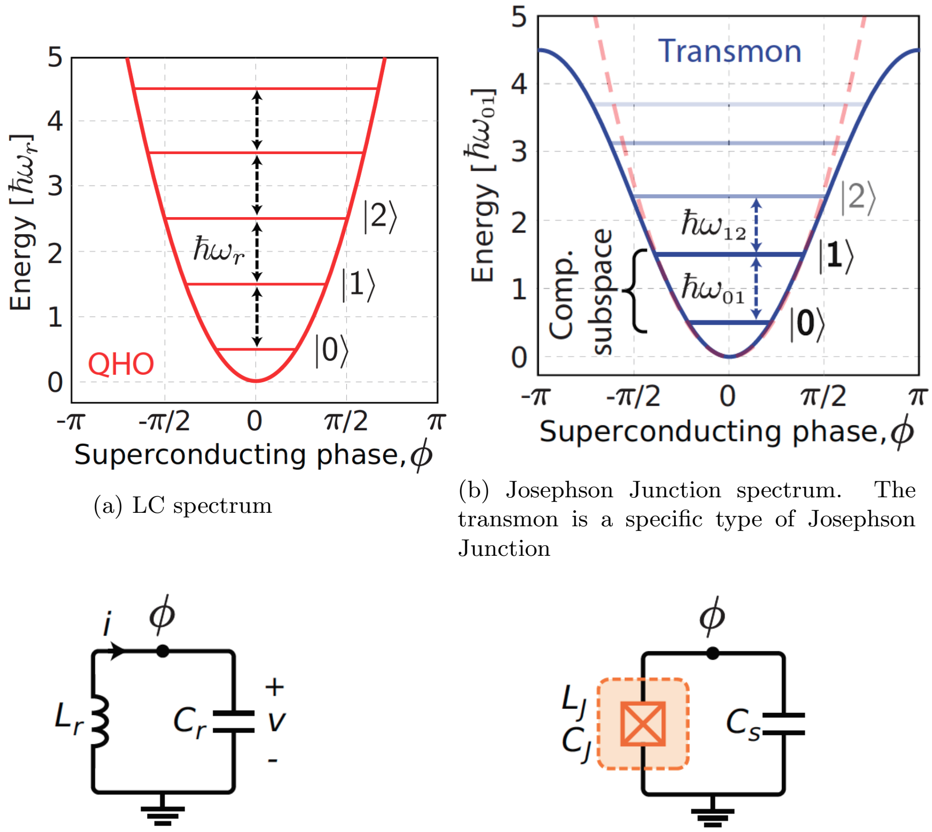
$$H = \hbar\omega_r\left(a^\dagger\,a + \frac{1}{2}\right) \tag{3.7}$$

where

$$n \;=\; i\,n_0\,(a - a^\dagger) \tag{3.8}$$

$$\phi \;=\; \phi_0(a + a^\dagger) \tag{3.9}$$

with $n_0 = \left(\frac{E_L}{32E_C}\right)^{\frac{1}{4}}$.
Eq. (3.7) is the quantum Hamiltonian for the LC circuit. This system cannot be a qubit because it does not respect the first DeVicenzo's criteria (see Sec. 2.6). We know from 1-d Harmonic oscillator theory that the energy difference between levels i and $i+1$ is always given by ω_r. This is a real problem in performing controllable operations. Consider that we want to flip the state from $|0\rangle$ to $|1\rangle$ of a N-state qubit. For doing that, we will send a photon with energy ω. But, for the degeneracy of the level differences, we could also excite different levels



(a) LC spectrum

(b) Josephson Junction spectrum. The transmon is a specific type of Josephson Junction

(c) LC circuit

(d) Josephson junction circuit

Figure 3.1: Spectrum of LC circuit (Harmonic oscillator) (3.1a) vs one with a Josephson junction (3.1b). We have also the graphical representation of these two circuits (3.1c) and (3.1d)

with the same transition probability, for example, from the first to second or from the second to the third.

We must add a non-linear element to the quantum circuit to modify the spectrum. In standard CQEd (Circuit Quantum Electrodynamics), one replaces the inductor with a Josephson Junction [39, 40].

The Hamiltonian of the quantum circuit including a Josephson Junction becomes:

$$H = 4\,E_C n^2 - E_j\,\cos(\phi)\,, \tag{3.10}$$

where $E_C = \frac{e^2}{2C_\Sigma}$ with $C_\Sigma = C + C_J$ (C_J indicates self-capacitance of the junction) and $E_j = \frac{I_c \Phi_0}{2\pi}$, the Josephson energy.

Fig. 3.1 shows the difference between the spectra of the LC circuit and the one

with a Josephson Junction. Moreover, in the same figure, the pictorial representations of their circuits are shown. We should observe how the energy differences change.
The quantum circuit formed by a Josephson Junction and a capacitor may be a good candidate for being a qubit.

3.1.1 Josephson Junction Hamiltonian

The Josephson Junction (JJ) is made of two (or more) layers of a superconducting material separated by a thin insulating film. This sandwich structure can explain why the JJ has intrinsic capacitance C_J of eq. (3.10). Fig. 3.2 shows an example of JJ.
Due to the thin insulator layers, a Cooper pair, a bound state of two electrons with opposite spin, can coherently tunnel through the insulator giving raise to a superconducting current. This is called DC Josephson effect.

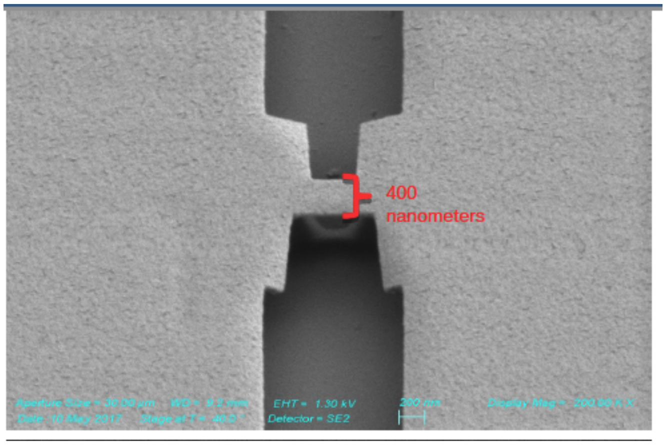


Figure 3.2: A Josephson junction. The two superconducting sides of JJ are this figure's darkest grey parts. In the middle, there is an insulator of a thickness of 400 nm. Figures is taken from Ref. [41]

We can write the Schrödinger equations for the two superconducting parts, where ψ_1 and ψ_2 describes the wavefunction on each side [43] :

$$\frac{d\psi_1}{dt} = \frac{eV}{2}\psi_1 + K\psi_2 \qquad\qquad \frac{d\psi_2}{dt} = \frac{eV}{2}\psi_2 + K\psi_1 \tag{3.11}$$

where V indicates the potential across the JJ. K is a characteristic constant of the JJ and it describes the coupling between the two superconducting sides.
Using a simple description of the wavefunction of two superconducting parts,

$$\psi_i = \sqrt{\rho_i}e^{i\theta_i} \qquad i = 1, 2\,, \tag{3.12}$$

one can get:

$$\frac{d\rho_1}{dt} = \frac{2}{\hbar} K\sqrt{\rho_1\rho_2}\,\sin(\phi)\,, \tag{3.13}$$

$$\frac{d\rho_2}{dt} = -\frac{2}{\hbar} K\sqrt{\rho_1\rho_2}\,\sin(\phi)\,, \tag{3.14}$$

$$\frac{d\theta_1}{dt} = -\frac{K\sqrt{\rho_2}}{\hbar\sqrt{\rho_1}}\cos(\phi) - \frac{qV}{2\hbar}\,, \tag{3.15}$$

$$\frac{d\theta_2}{dt} = -\frac{K\sqrt{\rho_1}}{\hbar\sqrt{\rho_2}}\cos(\phi) + \frac{qV}{2\hbar}\,, \tag{3.16}$$

where $\phi = \theta_2 - \theta_1$.
The first two equations tell us how the charge densities change. Therefore, they describe which kind of current flows in the JJ. The current from side 1 to side 2 is simply $\frac{d\rho_1}{dt}$.
Usually, the two layers of the JJ are connected by wires to a battery that makes the voltage V constant across it. Including it in the JJ's equation, one has that the two densities $\rho_1 = \rho_2 = \rho_0$ do not change during time. (If the two densities change, they will produce further voltage). Then, we have found the first relation of JJ:

$$I = I_c\,\sin(\phi) \tag{3.17}$$

where we have recalled ϕ as the phase different of two superconducting materials $\phi = \theta_1 - \theta_2$ and $I_C = \frac{2K\rho_0}{\hbar}$.
Subtracting eq. (3.16) with eq. (3.15), we get:

$$\frac{d\phi}{dt} = \frac{d(\theta_2 - \theta_1)}{dt} = \frac{2e\,V}{\hbar} \tag{3.18}$$

This equation is called the second relation of JJ.
The energy of the JJ is given by:

$$E = \int V\,I\,dt = \int I_c\sin(\phi)\frac{1}{2e}\frac{d\phi}{dt}dt = \int I_c\sin(\phi)\frac{1}{2e}d\phi = -\frac{I_c}{2e}\cos(\phi) \tag{3.19}$$

We have proved that the energy depends on the cosine of the phase difference.This final result proves the term in eq. (3.10).

3.1.2 Quantum circuits with Josephson Junction

Eq. (3.10) gives us:

$$H = 4\,E_C n^2 - E_j\,\cos(\phi)\,, \tag{3.20}$$

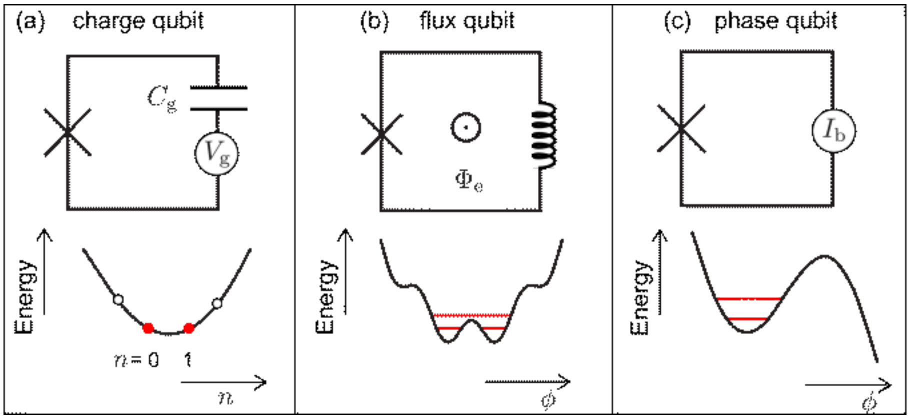


Figure 3.3: Scheme of three basic superconducting quantum circuits and their potential energies. (a) Charge qubit (b) flux qubit and (c) phase qubit. It is taken from Ref. [42]

Considering the effect of offset charge noise [44] $n_g = \frac{Q_r}{2\,e} + \frac{C_j V}{2e}$, with Q_r is the charge induced by the environment, V_g and C are the gate voltage and capacitor respectively, the general Hamiltonian is given by:

$$H = 4\,E_C(n^2 - n_g^2) - E_j\,\cos(\phi)\,. \tag{3.21}$$

We have three types of Josephson Junction and they are determined by the values of two parameters E_C and E_j. Considering that the n and ϕ obey to the Heisenberg principle $\left[e^{i\phi}, n\right] = e^{i\phi}$ or $\Delta\phi\,\Delta n \geq 1$, when $E_C > E_J$ the circuit is dominated by the capacitance and n is a good quantum number (ϕ has a large quantum fluctuations). Instead for $E_J > E_C$, the JJ dominates the circuit with a well-defined ϕ but with a large fluctuations for n_g.
Therefore, the classification of Josephson Junctions depends on the ratio $\frac{E_J}{E_C}$. The standard rank for JJ circuit is

- When $\frac{E_J}{E_C} \ll 1$. (in particular $\frac{E_C}{E_J} \sim 10$). The device in this regime is called charge-qubit. The Hamiltonian of eq. (3.21) becomes:

  $$H = \sum_n 4\,E_C\,(n - n_g)^2\;|n\rangle\,\langle n| - \frac{1}{2}E_J\,(|n+1\rangle\,n + |n\rangle\,\langle n+1|) \tag{3.22}$$

  This device at first order is insensitive to offset charge noise.

- When $\frac{E_J}{E_C} \sim 1 - 50$. The circuit in this regime is called the flux-qubit circuit. We must consider all the terms of the Hamiltonian and Fig. 3.3(b) shows its spectrum.

- When $\frac{E_J}{E_C} \gg 50$. This device is called a phase-qubit circuit. Its spectrum is very similar to one of the harmonic oscillators with a small anharmonicity. This circuit is very insensitive to offset-charge noise.

Fig. 3.3 shows the quantum circuits for the three basic types of superconducting quantum circuits and their potential energies.

Several improvements of the three presented devices are proposed. A complete list can be found in Refs. [41, 42]. The modifications are realized to obtain a long coherence time, an easy connectivity between qubits or a full control of the circuit. The main one are the quantronium [45, 46],the four-junction flux-qubit circuits [47, 48, 49, 50], the tunable-gap flux-qubit circuit [51, 52, 53], the transmon [44, 54], the fluxonium-qubit [55, 56], the Xmon-qubit [57], the Gmon-qubit [58, 59].
We currently limit our focus on the transmon circuit that better approximates the LLNL quantum testbed [37]. The transmon works with a ratio $\frac{E_J}{E_C} \sim 10 - 100$, so we are in the regime of phase qubit.

3.1.3 Transmon's Hamiltonian

In the regime when $\frac{E_J}{E_C} \gg 1$, the superconducting phase ϕ becomes a good quantum number. We have that the N lowest states, which we will use as states of the N qubit, are localized inside the well as Fig. 3.1b shows. Therefore, we can expand the cosine part of eq. (3.10) around the minimum ($\phi = 0$), obtaining:

$$E_J \cos(\phi) = \frac{E_j}{2}\phi^2 - \frac{1}{24}E_J\phi^4 + O(\phi^6)\,. \tag{3.23}$$

The constant is just an energy shift that we can neglect in our calculation because as we will see it is fundamental to the transition energies. At the second order, we obtain a Harmonic oscillator term ϕ^2, but the fourth-order term gives us an anharmonicity. We need the anharmonicity to well-control the transmon's states as the first DiVicenzo criterium [23] requests.
Therefore, the general transmon Hamiltonian becomes

$$\begin{aligned} H &= 4\,E_C(n^2 - n_g^2) - E_j\,\cos(\phi) \simeq 4\,E_C(n^2 - n_g^2) + \frac{E_j}{2}\phi^2 - \frac{1}{24}E_J\phi^4 \\ &= \omega_q a^\dagger a + \frac{\alpha}{2}a^\dagger a^\dagger a a\,, \end{aligned} \tag{3.24}$$

where α is the anharmonicity $\alpha = \omega_{21} - \omega_{10}$ and a ($a^\dagger$) is the annihilation (creation) operator of the eigenstates of Harmonic oscillator states defined by eq. (3.9)). The transmon devices work when $\alpha \ll \omega_q$. Hence, we recognize that the final transmon Hamiltonian is the weakly anharmonic oscillator one.
Fig. 3.4 shows the transmon energies as a function of offset charge n_g for different ratios $\frac{E_J}{E_C}$. The actual transmon range is when $\frac{E_J}{E_C} = 50$. From the figure we observe that the energies are insensitive to the offset charge n_g. Furthermore, the spectrum is very similar to one of weak Anharmonic oscillator.

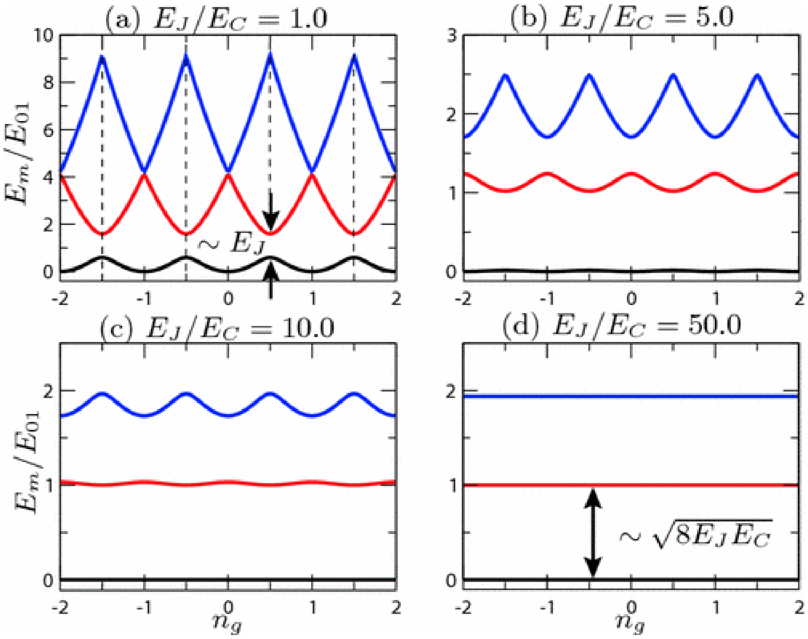


Figure 3.4: First three energies E_m of transmon device as a function of offset charge n_g for different ratios $\frac{E_J}{E_C}$. The energies are scaled by the factor E_{10}. Taken from Ref. [44]

3.2 Operations in real hardware

We have described how we can build a qubit with a superconducting device. Now, we are interested in implementing our quantum gates that are indicated by a quantum algorithm.

We start from the fact that the (transmon) qubit system can be made to interact with a time-dependent microwave pulse. The evolution of the quantum processor interacting with this pulse after some time T_m is a way to implement a particular quantum gate. Its action is obtained from solving the Schrodinger equation for the real time dynamics. Tailoring the pulses, we get different quantum gates. Therefore, for a set of quantum operations or gates, we should optimize the time dependent shape of the pulses.

Ch. 2 introduced analog and digital quantum computation for implementing a general quantum gate. In the analog scheme, we need to optimize the microwave pulse, such that the whole evolution of the quantum processor after a particular time T_m gives the desired quantum gate. In the digital scheme, instead, we translate and optimize the general quantum gate in a sequence of elementary gates. However, these digital elementary gates are implemented via specific time-dependent microwave pulses. An example can be found in IBM digital gates that are implemented through specific pulse shapes.

In order to determine the pulses suitable to the desired operation, we start from the lab Hamiltonian that describes a generic interaction between a microwave pulse with amplitude $\epsilon(t)$ and a multi-level system quantum system [60],

$$\begin{aligned} H_{int} = u(t)H(t) = (c + c^{\dagger})\ \left[\epsilon(t)e^{-i\omega_d t} + \epsilon(t)^{*}e^{i\omega_d t}\right] = \\ (c + c^{\dagger})\,(2\ \mathrm{Re}(\epsilon(t))\cos(\omega_d\, t) + 2\ \mathrm{Im}(\epsilon(t))\sin(\omega_d\, t))\ , \end{aligned} \tag{3.25}$$

where c and $c^{\dagger}$ are the annihilation and creation operators for transmon's levels. ω_d indicates the frequency of the electric field. The free-transmon Hamiltonian is given by $H_0 = \hbar \sum \omega_k \left|k\right\rangle \left\langle k\right|$ where ω_k is the energy of k-th level. In standard notation, the real part of electric signal, $\epsilon_I(t)\ = \mathrm{Re}\left[\epsilon(t)\right]$, is called in-phase component and the imaginary part, $\epsilon_Q(t)\ = \mathrm{Im}\left[\epsilon(t)\right]$, the quadrature component .
To facilitate the tailoring of the time-dependent microwave pulse, one can change the reference frame into an easier one, called Rotating Frame (RF). This is defined by applying the following time-dependent unitary operator to the system:

$$R(t)\ =\ \exp\left[i\omega_d t \sum_{k=0}^{N-1} k \left|k\right\rangle \left\langle k\right|\right]\ , \tag{3.26}$$

where N is the total number of states of the transmon device.
The action of the unitary operator $R(t)$ to the transmon Hamiltonian is described by:

$$H_{rot} = R(t)(H_0 + H_{int})R(t)^{\dagger} - i\, R(t)\, \frac{d}{dt} R(t)^{\dagger}\ . \tag{3.27}$$

Leading to the rotating-frame Hamiltonian H_{rot}:

$$H_{rot} = \sum_{k}^{N-1} (\omega_k - k\,\omega_d) \left|k\right\rangle \left\langle k\right| + \epsilon_I(t)\,(a + a^{\dagger}) - i\,\epsilon_Q(t)\,(a - a^{\dagger}) \tag{3.28}$$

where a and $a^{\dagger}$ are the lowering and raising operator in the rotating frame. We impose that ω_d is equal to the difference between the first two states of the transmon, $\omega_d = \omega_{10} = \omega_1 - \omega_0$. Moreover, using the rotating wave approximation we can also neglect terms oscillating with frequency $\omega_k + k\omega_d$.
Finally, the RF Hamiltonian is given by:

$$\begin{aligned} H\ =&\ \sum_{k=0}^{N-1} (\omega_k - k\omega_d) \left|k\right\rangle \left\langle k\right| + \epsilon_I(t) H_1^c - i\epsilon_Q(t)\, H_2^c \\ =&\ H_0\ +\ \epsilon_I(t) H_1^c - i\epsilon_Q(t)\, H_2^c \end{aligned} \tag{3.29}$$

where $H_1^c = a + a^{\dagger}$ and $H_2^c = -i(a - a^{\dagger})$.
In the end, for a desired quantum gate U_{wanted}, one should tailor the microwave pulse to drive the evolution of the quantum processor according to the gate itself. The formal general real time evolution operator for the transmon device

interacting with a time-dependent microwave pulse is given by the right side of the following equation:

$$U_{wanted} \sim \mathbb{T}\exp\left\{-\frac{i}{\hbar}\int_0^{T_m} d\tau\left[H_0+\hbar\epsilon_I(t)(a^\dagger+a)+i\hbar\epsilon_Q(t)(a^\dagger-a)\right]\right\} \tag{3.30}$$

where H_0 indicate the free Hamiltonian of the qubits, the terms $a^\dagger \pm a$ describe the control Hamiltonian. The operator $\mathbb{T}$ is the time order operator.

Therefore, one must optimize for a fixed machine period T_m the in-phase and quadrature components of the pulse such the previous equation (eq. (3.30)) is valid within a small error.

How do we tailor the pulse and how do we quantify whether the evolution operator implements correctly the desired gate or not? We have already discussed that in Ch. 2 one must compute the fidelity between two operators to measure the closeness between the experimental evolution operator and desired gate. Recapping a fundamental property of the fidelity, two operators are the same if and if only if the fidelity is equal to 1.

In conclusion, in order to implement a quantum gate U_{wanted} in a transmon device, one has to tailor a time-dependent pulse with real and imaginary parts, named ϵ_I and ϵ_Q, such that we maximize the fidelity between the two operators of Eq. (3.30). The right side operator describes the actual evolution of the quantum processor, and the left side represents the desired gate. Generally, we stop optimizing the pulse when we reach a preset value of fidelity. Instead of maximizing the fidelity, one may rather minimize the error fidelity defined by:

$$E_f = 1 - F(U_1, U_2) \tag{3.31}$$

where F is the fidelity between the operators U_1 and U_2.

The exposed procedure is very general. Indeed, it is also valid when we couple different transmons with different microwave pulses $\{\epsilon_j\}$. In this case, one should compute the general Hamiltonian that describes the quantum processor interacting with this set $\{\epsilon_j\}$ of microwave pulses.

3.3 Noise sources in quantum processors

Random and uncontrollable physical processes occur in devices for quantum computing s in any other experimental setup. We have them even in quantum computing. These noise sources are classified into errors in the readout process, the gate implementation (gate infidelity), or the qubit's interaction with the environment. All of them cause loss of coherence of qubit states destroying the quantum simulation.

The read-out errors and gate infidelity can be corrected a posteriori by some algorithms. Nevertheless, it is very hard to deal with the most physical source of noise, the interaction of the quantum processor with the environment[1]. In this

[1] In this thesis we will often refer to such interaction as "quantum noise

section, we will discuss a simple characterization of its effects with the so-called Bloch-Redfield model [61, 62, 63]. In this model, the effect of noise on a single qubit state,

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\ , \tag{3.32}$$

can be described by the following density matrix for the qubit after some time t_m:

$$\begin{pmatrix} 1 + (|\alpha|^2 - 1)e^{-\frac{t_m}{T_1}} & \alpha\beta^* e^{-\frac{t_m}{T_2}}\, e^{-\frac{t_m}{2\,T_1}}\, e^{i\omega t} \\ \alpha^*\beta e^{-\frac{t_m}{T_2}}\, e^{-i\omega t} e^{-\frac{t_m}{2\,T_1}} & |\beta|^2 e^{-\frac{t_m}{T_1}} \end{pmatrix}\ , \tag{3.33}$$

where the time T_1 and T_2 describe the so-called relaxation and dephasing time, respectively. We can describe the effect of noise with these two parameters. The third parameter $e^{i\omega\, t}$ with $\omega = \omega_{10} - \omega_d$ describes the case when the frequency qubit ω_{10} differs from the rotating-frame frequency ω_d (it describes the frequency of oscillations of the electric field signal). In our work, we will neglect this contribution.

3.3.1 Relaxation process

We start with the parameter T_1, called relaxation time. The process related to T_1 is called relaxation process or longitudinal relaxation. It indicates the decay from the excited state (the logical state $|1\rangle$) to the ground state ($|0\rangle$) and vice versa. Formally, it is given by:

$$\frac{1}{T_1} = \frac{1}{T_1^{ge}} + \frac{1}{T_1^{eg}} \tag{3.34}$$

where T_1^{eg} (T_1^{ge}) indicates the typical time of decay from excited state (ground state) to the ground state (excited state).
For superconducting qubits one can estimate with the Boltzmann distribution $T^{ge} = T_{eg} e^{-\frac{\omega_{10}}{k_B T}}$ where T is the temperature ($T \sim 2 - 20$ mK) and $\omega_{10} = E_1 - E_0$. In this case, one can easily demonstrate that the second contribution (ground to excited state) is exponentially suppressed, hence, $T_1 \sim T_1^{eg}$. Experimentally, to measure T_1, one prepares the qubit in the $|1\rangle$ state and samples the probability to find the qubit in $|1\rangle$ state as a function of time. An example with the first device at LLNL is shown in Fig. 3.5. The function that interpolates the points of this process is well-fitted by a single exponential. The parameter in the exponent is connected to T_1.

3.3.2 Dephasing process

The other very important contribution to the noise is due to T_2, the dephasing time. This is the main contribution to noise in superconducting devices. It formally describes the depolarization of the Bloch sphere in the $x - y$ plane. In

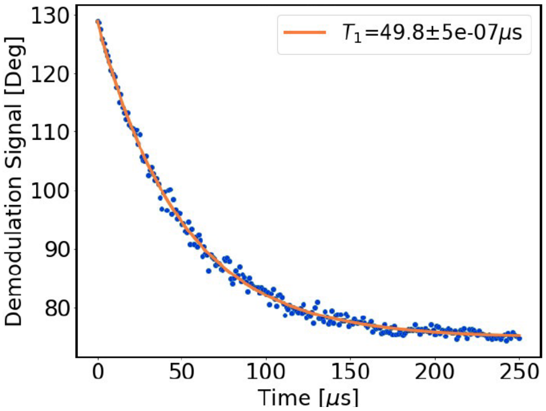


Figure 3.5: Result of T_1 estimation of LLNL testbed. Taken from Ref. [64]

other words, this process describes the loss of the relative phase between the $|0\rangle$ and $|1\rangle$ states. The relative phase of a qubit is described by the off-diagonal terms of the diagonal matrix. According to the Bloch-Redfield model, the T_2 contribution kills these terms exponentially (see eq. (5.23)).
To visualize what this process describes, one could imagine to have the initial state in x-state, $|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. This represents the point along the x axis in the Bloch sphere. The dephasing noise starts to rotate the qubit state randomly along the z-axis. Therefore, after some time t, we have a cloud of states around the equatorial plane of the Bloch sphere. Fig. 3.6 plots this process.
Starting from the state $|x\rangle$ and waiting for a long time, the result of this noise source is a mixture of states equally distributed around the equatorial plane. We lose the relative phase between the two states entirely in this situation. For the T_2 estimation one can use the Ramsey interferometry [66]. First, the initial state is obtained applying a $X_{\pi/2} = R_x(\frac{\pi}{2})$ gate (i.e., a rotation along the x-axis of angle $\pi/2$) to $|0\rangle$). After some time τ, one applies a $X_{\pi/2}$ pulse. Repeating this process for different values of τ, one gets some oscillation of Fig. 3.7. Interpolating the exponential decay gives the time T_2.
Spin Echo experiments [67, 68] are another way to estimate T_2. This applied a $Y_\pi = R_y(\pi)$ pulse in the middle of the two $X_{\pi/2}$ pulses. This variation helps to reduce the sensitivity of qubits to other sources of noise (for example, relaxation processes).

3.3.3 Markovian Lindblad master equation

We have seen that the quantum noise source can be parameterized with the two times T_1 and T_2 describing the relaxation and dephasing processes. The Bloch-Redfield model is too simple to compute the actual effect of quantum noise in a real device simulation of a quantum algorithm.

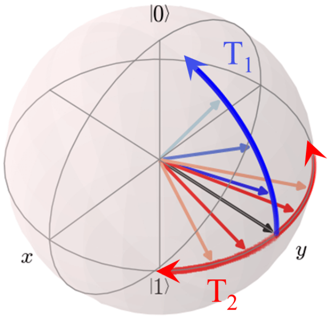


Figure 3.6: Bloch spheres for different sources of noise. We should notice how the dephasing process (T_2) and the relaxation process (T_1) may move the initial state creating a mixture of states. Taken from Ref. [65]

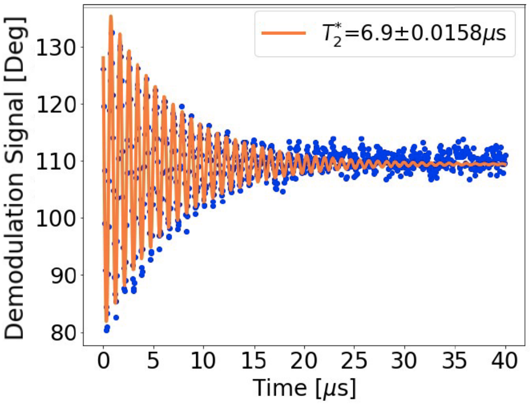


Figure 3.7: Result of T_2 estimation of LLNL testbed.Taken from Ref. [64]

A better description is provided by the Markovian Lindblad master equation [69] describing the real time evolution of a quantum system (in our case, the qubits) with some noise sources. As previously mentioned the noise sources are the relaxation process, parameterized by T_1 and by the operator a, and dephasing one, parameterized by T_2 and by the operator $a^\dagger a$. The operator a and $a^\dagger$ describe the annihilation and creation operator on the qubit state, respectively. The Markovian Lindblad master equation for a qubit system is given by

$$\frac{d\rho}{dt} = -\frac{i}{\hbar}\left[H_{QPU}, \rho\right] + \left(\frac{1}{T_1}\hat{D}[a] + \frac{1}{T_2}\hat{D}[a^\dagger a]\right)\rho \tag{3.35}$$

$$H_{QPU} = H_0 + H_c \tag{3.36}$$

$$D[O] = O\,\rho\,O - \frac{1}{2}\left\{O^{\dagger}O, \rho\right\} \,, \tag{3.37}$$

where ρ is the density matrix of N-state qubit. H_{QPU} is the Hamiltonian in the exponent of eq. (3.30), the sum of free Hamiltonian (H_0) and the control Hamiltonian (H_c).

3.4 Qubit Readout

The qubit readout is the procedure that allows us to extract the probabilities of the states. Generally, a quantum measurement can be described as an interaction between a quantum system (qubit) with a classical variable of a classical probe. Then the probe is classically measured to have the information about the observable of the quantum system.
In CQEd, the qubit is entangled with a classical resonator. By measuring the electric field properties of the cavity, for instance, the number of photons, one can have information about the probability of the qubits.
The simple model that describes the dynamics of an atom or a transmon with two states with a resonator is the James-Cummings model [70]. In this model, the Hamiltonian is given by:

$$H = \omega_r (a^{\dagger}a + \frac{1}{2}) + \frac{\omega_q}{2}\sigma_z + \frac{\Omega_0}{2}(a\sigma_+ + a^{\dagger}\sigma_-)\,, \tag{3.38}$$

where ω_q and ω_r indicates the frequency of qubit and resonator respectively and $\Omega_0 = 2\,g$ with g is the vacuum Rabi frequency. σ_+ and σ_- are the qubit ladder operators.
The energy and eigenstates of these Hamiltonian are:

$$E_n^{\pm} = \omega_r\,(n + \frac{1}{2}) \pm \frac{1}{2}\sqrt{\Delta^2 + \Omega_n^2} \tag{3.39}$$

$$|n, +\rangle = \cos\left(\frac{\theta_n}{2}\right)|e\rangle\,|n\rangle + \sin\left(\frac{\theta_n}{2}\right)|g\rangle\,|n+1\rangle \tag{3.40}$$

$$|n, -\rangle = \cos\left(\frac{\theta_n}{2}\right)|g\rangle\,|n+1\rangle - \sin\left(\frac{\theta_n}{2}\right)|e\rangle\,|n\rangle\ , \tag{3.41}$$

where $|g\rangle$ and $|e\rangle$ indicate the two state of the qubit, the ground state and excited one respectively, and the $|n\rangle$ describes the state with n photons in the cavity. The parameters Δ, Ω_n and θ_n are given by:

$$\Delta = \omega_q - \omega_r \qquad \tan(\theta_n) = \frac{\Omega_n}{\Delta} \qquad \Omega_n = \Omega_0\,\sqrt{n+1} \tag{3.42}$$

A useful regime of the JC Hamiltonian for a readout procedure is when $g \ll |\omega_q - \omega_r|$. This regime is called the dispersive regime. If the ratio $\frac{g}{\Delta}$ is small, we can expand the Hamiltonian as follows,

$$
\begin{aligned}
H \sim \left(a^\dagger a + \frac{1}{2}\right)\left(\omega_r + \frac{g^2}{\Delta}\sigma_z\right) + \frac{1}{2}\left(\omega_q + \frac{g^2}{\Delta}\right)\sigma_z \\
\sim \left[\omega_r + \frac{g^2}{\Delta}\sigma_z\right] a^\dagger a + \frac{1}{2}\left(\omega_q + \frac{g^2}{\Delta}\right)\sigma_z \, . \qquad (3.43)
\end{aligned}
$$

This Hamiltonian is at the first order in g/Δ. It has been obtained using the Schrieffer-Wolff transformation in which we eliminate the linear-order term $a\sigma_+ + a^\dagger\sigma_-$ [71].

Eq. (3.43) shows an essential property of transmons, the *Quantum non-demolition* (QCD) readout. The outcomes of the qubit measurement (the qubit probabilities) are not altered from the acts of measuring the photon number. Indeed, if the number of photons is small, the σ_z operator commutes with the interaction term in eq.(3.43).

The most important result from Eq. 3.43 is that the resonator frequency takes a energy shift depending on the qubit state. The shift magnitude is given by $\frac{g^2}{\Delta}\langle\sigma_z\rangle$. Our readout procedure is derived from a two-state system, but it can be generalized for N state qubit. Therefore, we can infer the qubit probabilities by measuring the resonator frequency.

Note that the qubit frequency due to the vacuum fluctuations of the resonator picks up a term $\frac{g^2}{\Delta}$. This is called the Lamb shift. There is another significant effect, called ac-Stark shift, coming from the term $2\frac{g^2}{\Delta}a^\dagger a\sigma_z$. This effect contributes to the qubit dephasing.

3.4.1 Readout in I-Q plane

We have just described the physics of the readout. The system is set up in order to be in the dispersive regime, where the cavity photon picks a contribution due to the qubit state. Moreover, this also prevents the interaction of the measurement process on the qubit.

The standard way to read the cavity states is to use the in-phase and quadrature components plotted in the so-called I-Q plane of the voltage coming from the reflection or transmission of the cavity.

The cavity is coupled with an electric signal of shape $s_{ini}(t) = A\cos(\omega_{RO}\, t)$, where ω_{RO} is the carrier frequency used to probe the resonator. This signal after the interaction with the resonator has the form :

$$
s(t) = A_{RO}\cos(\theta_{RO} + \omega_{RO} t)\,, \qquad (3.44)
$$

where A_{RO} and θ_{RO} are respectively the amplitude and phase due to the contribution of the qubit states. Fig. 3.8 shows the results for a single qubit. In the figure, the amplitude and phase are shown as a function of the difference between signal frequency (ω_{RF}) and the frequency of the resonator (ω_r). When

$\omega_{RF} = \omega_r$, one can notice that two phases have the most significant difference. The I-Q plane is a standard method to find the amplitude and phase of an electric signal [72]. Employing it, one rewrites for each frequency ω_{RO} the signal $s(t)$ of (3.44) in this way

$$\begin{aligned} s(t) &= A_{RO}\cos(\theta_{RO})\cos(\omega_{RO}t) + iA_{RO}\sin(\theta_{RO})\sin(\omega_{RO}t) \\ &= I\cos(\omega_{RO}t) + iQ\sin(\omega_{RO}t)\,. \end{aligned} \tag{3.45}$$

where the functions I and Q functions are called "in-phase" and "quadrature" components respectively.
We use the I-Q plane because it is easy to read the measurement results. Indeed, after some analysis, one may obtain the plot shown in Fig. 3.9. Each point represents a measure in a specific qubit state and the big spots represent the qubit states. Classifying and counting the number of data for each spot and dividing them by the total number of data, one obtains the qubit probabilities.

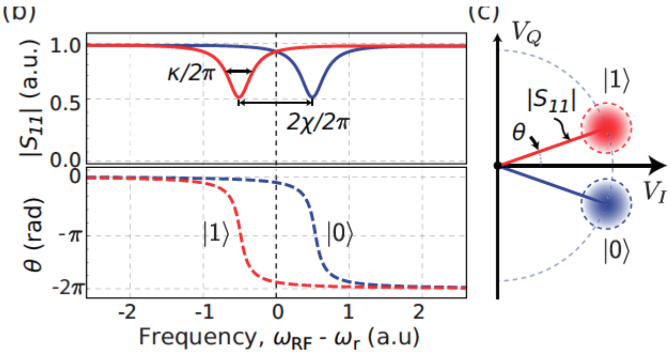


Figure 3.8: Readout example for the signal (absolute value and phase) due to reflection outcome when the qubit is the $|0\rangle$ or in the $|1\rangle$ state. (c) I-Q plane for the qubit. Taken from Ref. [41]

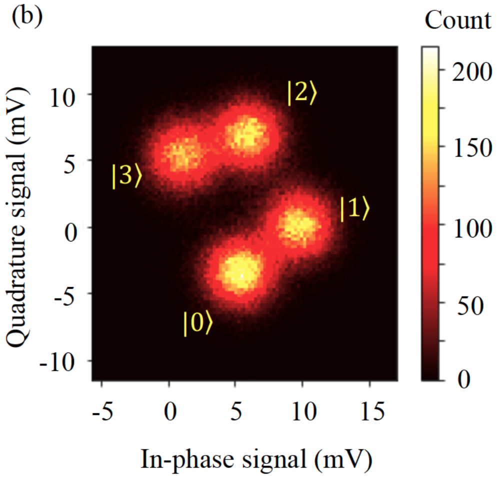


Figure 3.9: Readout plane I-Q for the LLNL quantum testbed for a qudit of 4 states. Taken from Ref. [37]

Chapter 4

Determination of the spectra from a real time evolution of a quantum system

This chapter will focus on the work of Ref. [9]. The paper presents a study of the real time evolution (or dynamics) of two interacting neutrons through a level device quantum simulation. The unitary time propagation of the nuclear dynamics is efficiently encoded into a single multi-level quantum device present at the Lawrence Livermore National Laboratory (LLNL).
We will begin illustrating the nuclear problem of Ref. [9]. Then, the fundamental steps to simulate the dynamics of a quantum system will be illustrated. After that, we will discuss numerical algorithms that tailor the experimental microwave pulse for implementing the desired quantum gate according to the analog quantum simulation approach.
The device level simulation of Ref. [9] and the actual data from the LLNL quantum testbed [37] will be shown.
In the end, the simulations applying the same method for a different quantum system, the Hydrogen atom, will be presented. They were done before the actual run on the LLNL testbed.

4.1 Nuclear system

Ref. [9] presents a simulation for computing the real time evolution (or dynamics) of two neutrons interacting with a simple potential derived by the leading order(LO) of chiral effective field theory (χ-EFT), derived in details in App. C.

The Hamiltonian of two neutrons is written as

$$H = T + V_{LO}(\vec{r})\,, \tag{4.1}$$

where the potential is (see eq. (C.33))

$$\begin{aligned} V_{LO}(\vec{r}) \;&=\; \frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1\cdot\tau_2)\left[T_{m_\pi}(r)(1-e^{-\frac{r^4}{R_0^4}})\vec{S}_{12} + (1-e^{-\frac{r^4}{R_0^4}})Y_{m_\pi}(r)\sigma_1\cdot\sigma_2\right] + \\ &+ c_s\alpha e^{-\frac{r^4}{R_0^4}} + c_t\sigma_1\cdot\sigma_2\alpha e^{-\frac{r^4}{R_0^4}}\,, \end{aligned} \tag{4.2}$$

and T indicates the Kinetic energy operator.
The Hamiltonian can be split in two terms, a spin-dependent (SD) (V_{SD}) and spin-independent (SI) (H_{SI}) components. We then write:

$$H = V_{SD} + H_{SI} \tag{4.3}$$

with

$$H_{SI} = T + V_{SI} = T + c_s\alpha e^{-\frac{r^4}{R_0^4}}\,, \tag{4.4}$$

$$\begin{aligned} V_{SD} \;=\; \frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1\cdot\tau_2)&\left[T_{m_\pi}(r)(1-e^{-\frac{r^4}{R_0^4}})\vec{S}_{12} + (1-e^{-\frac{r^4}{R_0^4}})Y_{m_\pi}(r)\sigma_1\cdot\sigma_2\right] \\ &+ c_t\sigma_1\cdot\sigma_2\alpha e^{-\frac{r^4}{R_0^4}}\,. \end{aligned} \tag{4.5}$$

The evolution at time t of a generic state of this system, $|\psi(t)\rangle$, is obtained from

$$|\psi(t)\rangle = e^{-i\,\Delta t\,H}\;|\phi\rangle = e^{-i\,\Delta t\,(H_{SI}+V_{SD})}\;|\phi\rangle\;, \tag{4.6}$$

where $|\phi\rangle$ is the state at time $t = 0$.
If we consider the time step Δt in the limit of $\Delta t \to 0$, we can apply the Trotter decomposition to the propagator splitting the spin-dependent part from the spin-independent. Therefore, we get:

$$e^{-i\,\Delta t\,H} \simeq e^{-i\,\Delta t\,(T+V_{SD})}e^{-i\,\Delta t\,V_{SI}}\;|\phi\rangle\;. \tag{4.7}$$

Looking closer at the spin dependent potential, whose operators are $\vec{S}_{12}$ and $\sigma_1\cdot\sigma_2$, one can rewrite this potential in term of Pauli matrices[22]

$$V_{SD} = \sum_{\alpha,\beta\,=\,x,y,z} A_{\alpha\beta}(r)\sigma^1_\alpha\cdot\sigma^2_\beta\,, \tag{4.8}$$

where $A_{\alpha,\beta}(r)$ indicates the projection of the spin-dependent potential in the α and β-axis. For example, one has:

$$\begin{aligned} A_{xx} \;=\; \frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1\cdot\tau_2)&\left[T_{m_\pi}(r)(1-e^{-\frac{r^4}{R_0^4}})(3r_x^2-1) + (1-e^{-\frac{r^4}{R_0^4}})Y_{m_\pi}(r)\right] \\ &+ c_t\alpha e^{-\frac{r^4}{R_0^4}}\,. \end{aligned} \tag{4.9}$$

$$A_{xy} = \frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1 \cdot \tau_2) T_{m_\pi}(r)(1 - e^{-\frac{r^4}{R_0^4}})\, 3\, r_x\, r_y\,, \tag{4.10}$$

where $r_x = \vec{r} \cdot \vec{x}$ and $r_y = \vec{r} \cdot \vec{y}$.
In Ref. [9] a further approximation was introduced in order to simplify the calculations. The authors considered the two neutrons "frozen" in their positions. In this condition the spin-independent component is the only one participant in the dynamics, while the radial wavefunction is described by two delta functions in the position of two neutrons, $\delta(\vec{r} - \vec{r}_1)\delta(\vec{r} - \vec{r}_2)$. Therefore, the two neutron dynamics is restricted to a spin evolution, where the distance $\vec{r} = \vec{r_1} - \vec{r_2}$ becomes a constant parameter of the spin-independent potential.
Now, we will discuss how to implement the nuclear spin dynamics. After that, we will continue showing the results of Ref. [9].

4.2 Fundamental steps for Quantum Simulations

This section will explain the fundamental steps to perform a quantum simulation of a physical quantum system, referring to the particular case mentioned above.
In quantum simulation, we want to evaluate some properties of a physical quantum system (for example, its ground state or its real time evolution). In the Hilbert space of this system, we start from an initial physical state, $|\phi\rangle$ and we perform some operations U_t. After their application, the final state will become $|\phi_t\rangle = U_t\,|\phi\rangle$. Our goal is to measure properties from $|\phi_t\rangle$.
To extract the desired information using a quantum processor, we map the physical states in the states of the quantum processor. The requested operation U_t is transformed in unitary operations U_g. We start from $|\psi\rangle$, the initial state of the processor that emulates $|\phi\rangle$. Then, U_g is applied to obtain the final state of quantum processor $|\psi_t\rangle = U_g\,|\psi\rangle$. This final state is equivalent to $|\phi_t\rangle$. Fig. 4.1 shows a scheme of the quantum simulation.
In order to do a quantum simulation correctly, we must employ the following steps. The derived two spatial-blocked spin neutrons will be taken as an example .

1. *Map the Hilbert space on quantum processor.* To effectively simulate a physical quantum system, we need to map its states onto the states of quantum processors. We must have a biunivocal correspondence between the states of physical systems and quantum processors. Furthermore, one should choose the most efficient map that cancels possible errors, for instance, removing the contribution of the quantum noise.
   An easy map of the 2-spin neutron spins consists of:

$$|0\rangle \Longleftrightarrow |\downarrow\downarrow\rangle\ ,$$

$$|1\rangle \Longleftrightarrow |\downarrow\uparrow\rangle\ ,$$

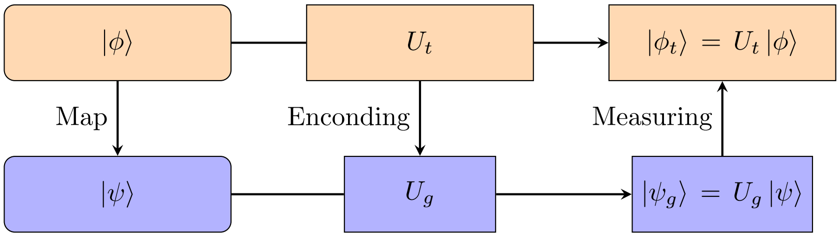

$|\phi\rangle$
U_t
$|\phi_t\rangle = U_t |\phi\rangle$
Map
Enconding
Measuring
$|\psi\rangle$
U_g
$|\psi_g\rangle = U_g |\psi\rangle$

Figure 4.1: Scheme of quantum simulation. The upper row represents the physical system; the lower one indicates the quantum simulation performed in a quantum processor. From left to right: the first boxes indicate the initial states, ones with U_t and U_g the performed operations and the final boxes the obtained final states. If the map and encoding are correct, extracting information from the final state of the quantum processor gives the same properties of the physical system.

$$|2\rangle \Longleftrightarrow |\uparrow\downarrow\rangle \ ,$$
$$|3\rangle \Longleftrightarrow |\uparrow\uparrow\rangle \ ,$$

where $0, 1, 2, 3$ indicate the states of the 4-level transmon qubit (or the states of two qubits). Different mappings can be used, assigning, for example, the triplet and singlet states to the levels of transmon qubit.

2. *Encoding the Quantum Operator*. After mapping physical states to states of the QPU, one has to translate the quantum algorithm into quantum gates (unitary operations). There are two main approaches to translating a general quantum gate into physical hardware gates.
   In the *digital quantum simulation* approach, the gate is translated in a sequence of elementary gates, a quantum circuit. This is the most common approach. This method suffers from the possible very large requested depth of quantum circuits; in other words, we would have a massive number of basic gates in the circuit. The depth of a quantum circuit is the quantum version of the depth of a classical circuit[73]. The depth of a circuit is defined as an integer number that describes the number of gates of the longest path from the input to the output qubits, moving forward in time along the wires. We can simply define the depth as the integer number that minimizes the times for running the entire quantum circuit counting each gate 1 in unit time. Having a massive number of gates causes that the requested machine time to implement them is longer than the decoherence time of the qubits. Moreover, having many gates, we should consider the contribution of the gate infidelity. These problems can be mitigated by techniques for efficiently compiling a quantum gate into a quantum circuit.
   The other approach is *the analog quantum simulation or Optimal Control* (OC). We compute the microwave pulse for implementing the desired operation in a single quantum gate application using the theory presented

in App. 3.2. This approach addresses in part the problem of noise since it reduces the required machine time. However, this approach also presents some problems with this approach. The main ones are the amount of classical computational resources required to compute the pulses. Indeed, one finds that the computational time grows exponentially with the number of states and consequently with the number of degrees of freedom in the system. App. A will discuss some details of the optimal control approach and it will present numerical algorithms that compute pulses. A method for reducing the classical computational resources will be discussed in App. B. In our case, the quantum operator U, which we want to implement, is the real-time evolution operator of the two neutron spins, $U = e^{-i\,\Delta t\,H}$. Ref. [9] implemented it through the optimal control technique.

3. *Initial State.* One needs to decide the initial state of the quantum simulation. Most simulations start for convenience in the $|0\rangle$ state, because it is the default state of quantum processors. Ref. [9] prepared the initial state in $|1\rangle$ representing the nuclear state $|\downarrow\uparrow\rangle$.

4. *Extract the information from the simulation.* We should decide what information we can extract from the quantum simulation. We measure the probability distribution from the quantum processor on some computational basis. We must link the occupation probability to observables of the physical system that we are interested in.
For example, in the real time evolution of two neutron spin systems, we are interested in the probability distribution of spins at different time steps.

5. *Execution.* Finally, we run on the quantum processor.

4.3 Real time evolution of two blocked neutrons

We listed all the elements to simulate the nuclear spin evolution of two interacting neutrons through a quantum computer.
Now, we will focus on implementing it in quantum processor available at LLNL [37]. Tab. 4.1 summarizes the fundamental elements of the quantum simulations of Ref. [9] applies for studying the real time evolution of two blocked neutrons interacting with a potential derived by a LO Chiral EFT.
The numerical optimization problem for finding the pulse for the dynamics was done by the python software *Quantum Toolbox in Python* (QuTiP) [74] with the *optimize_pulse_unitary* function implementing the GRAPE algorithm (see App. A for algorithm's details). Specifically, Ref. [9] uses six states for the transmons. The authors used six states instead of four to improve the accuracy of the numerical optimization of the pulse. The quantum gate implements the real time propagator for the four lowest levels of the transmon (using as map one of Tab. 4.1). The fourth and fifth levels are evolved through an identity gate. The calculations of in Ref. [9] are device simulations with parameters tailored on the physical machine. The results are expected to be very similar to ones

Map	$\vert 0\rangle \Longleftrightarrow \vert\downarrow\downarrow\rangle\,,\, \vert 1\rangle \Longleftrightarrow \vert\downarrow\uparrow\rangle\,,\, \vert 2\rangle \Longleftrightarrow \vert\uparrow\downarrow\rangle\,,\, \vert 3\rangle \Longleftrightarrow \vert\uparrow\uparrow\rangle$
Unitary gate	Real time evolution $e^{-i\Delta t V_{SD}(r)}$
Implementation of the quantum gate	Optimal control computed with GRAPE (see App. A)
Measure	We measure the probability of the states of the qudit after the implementation of k real time propagators. The obtained values of probability correspond to the spin probability of the two neutrons at time $t = \ k\,\Delta t$

Table 4.1: Elements for simulating nuclear physics

gotten from actual LLNL quantum testbed. Tab. 4.2 shows the experimental parameters used for emulating the quantum processor[1].
The real (ϵ_I) and imaginary part (ϵ_Q) of the obtained optimal pulse is shown in Fig. 4.2 (a). Fig. 4.2 (b) presents the Discrete Fourier Transform of this amplitude, highlighting the transition frequency of the transmon.
Fig. 4.3 presents the results obtained from the simulation of Ref. [9]. The circles represent the simulated probability output for the transmon states solving the Markovian Lindblad master equation (see eq. 3.35). The solid lines indicate the exact evolution of two neutrons scaled by device noise terms.

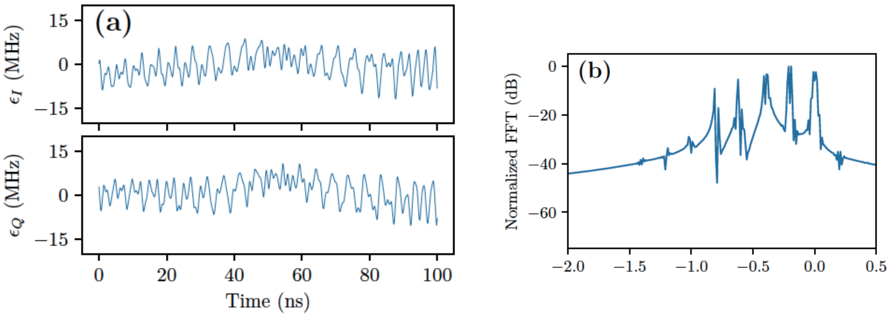


Figure 4.2: The two (a) figures show the optimized time dependent real (ϵ_I) and imaginary (ϵ_Q) amplitude pulse for the nuclear propagators. The (b) panel shows the Discrete Fourier Transform of the pulse amplitude. Plots taken from Ref. [9]

The discrete wavefunction $|\psi(t = j\,\Delta t)\rangle$, which represents the spin state at time $t = j\,\Delta t$ ($j \in N$) of the two neutrons, is obtained by repeating j-times the nu-

[1] This fact was confirmed in later experiments

3D transmon anharmonicity α_T	200 MHz
Transmon states	6
Relaxation time T_1	30 μs
Dephasing time T_2	50 μs
Total decoherence time $T^{-1} = T_1^{1} + T_2^{-1}$	27 μs
Pulse duration (pd)	100 ns
Pulse sampling frequency (s_{rate})	32 Gsamples
Number of points of the pulse	3200
Fidelity error	10^{-4}
Initial pulse for optimization	Gaussian function with height amplitude smaller than 2 MHz
Maximum Drive strength	20 MHz

Table 4.2: Parameters of the quantum testbed. The first four parameters indicate the hardware parameters (for more information see Ch. 3). The others describe the parameters for the pulse calculation

clear propagator,

$$
\begin{aligned}
|\psi(t = j\Delta t)\rangle = \prod_{i=0}^{j} e^{-i\Delta t V_{SD}} |\psi(t = 0)\rangle \\
= \sum_k c_k e^{-itV_{SD}} |\phi_k\rangle = \sum_k c_k e^{-it\lambda_k} |\phi_k\rangle \ , \quad (4.11)
\end{aligned}
$$

where we have expanded the wavefunction in eigenstates of the Hamiltonian, $V |\phi_k\rangle = \lambda_k |\phi_k\rangle$ and $c_k = \langle \phi_k | \psi(t = 0\rangle$.

Introducing the computational basis $|\zeta = 0, 1, 2, 3\rangle$ for measuring the quantum processor, the i-th occupation probability is given by:

$$
\begin{aligned}
P_i(t) = |\langle\zeta|\psi(t)\rangle|^2 = \left|\sum_k c_k e^{-it\lambda_k} \langle\zeta_i|\phi_k\rangle\right|^2 = \left|\sum_k c_k e^{-it\lambda_k} b_{ik}\right|^2 \\
= \sum_{kl} e^{-i\,t(\lambda_k - \lambda_j)} c_k c_j^* b_{ki} b_{ji}^* \ , \quad (4.12)
\end{aligned}
$$

where $b_{ik} = \langle\zeta|\phi_k\rangle$.

Eq. (4.12) shows that the occupation probability depends by the difference of

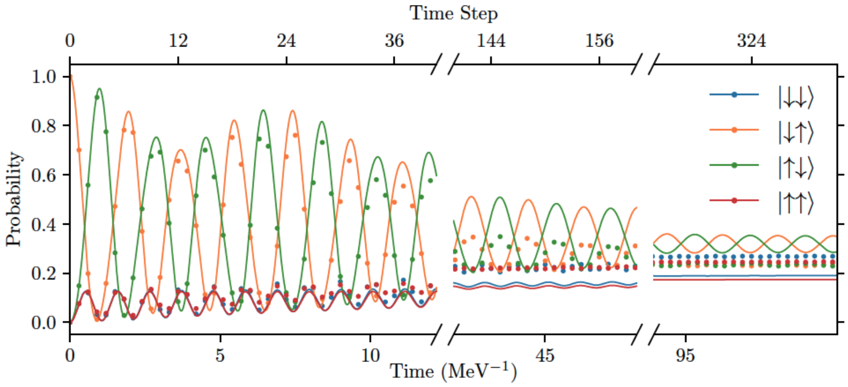


Figure 4.3: Occupation probabilities as a function of time in the quantum simulation of the spin evolution of the two interacting spins. Colored circles indicate the output probability as a function of the time steps. Each point is obtained by solving the Lindblad master equation, including dissipation and dephasing terms. Lines indicate the analytical evolution. They are obtained by integrating the LO Chiral EFT potential scaled by dephasing and relaxation processes. The collapses of the four probabilities at the latest time steps are due to device decoherence. Taken from Ref. [9]

the energies. Therefore, we can compute its its Fourier Transform, obtaining

$$P_i(\omega) = \int dt e^{i\,t\,\omega} P_i(t) = \sum_{kl} c_k c_j^* b_{ki} b_{ji}^* \int dt e^{-it(\lambda_k - \lambda_j) + i\,\omega\,t}$$

$$= \sum_{kl} c_k c_j^* b_{ki} b_{ji}^* \delta(\omega - (\lambda_k - \lambda_j))\,. \quad (4.13)$$

We can see from Eq. (4.13) that the Fourier transform of the occupation probability for each transmon level will show peaks centered at the difference between the energies. The overlap factors give the height of these peaks, $c_k c_j^* b_{ki} b_{ji}^*$. Looking closer at eq. (4.13), we can note that number of expected peaks is $\frac{d(d-1)}{2}$. This value is equal to all possible combinations between the two different energies.

Fig. 4.4 shows the Discrete Fourier Transform of occupation probability for the $|1\rangle$ state of Fig. 4.3. By the analysis described in App. B of Ref. [9], one can obtain the energy differences shown in Tab. 4.3.

This procedure is not sufficient to fully determine the spectrum. One needs at least one energy value, for instance, the lowest one λ_0, to reconstruct all the energy values.

Ref. [9] carries a very clever solution to this problem. One should do the same analysis for the real time evolution again, but changing the Hamiltonian of the propagator from V_{SD} to V_{SD}^3. Thereby, we will obtain the difference between

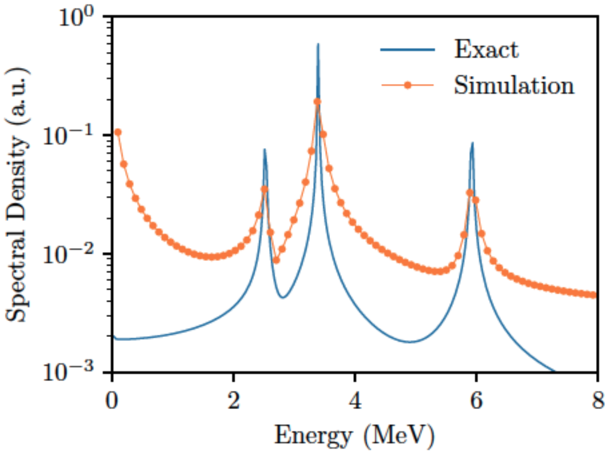


Figure 4.4: Energy Spectra. The circles are obtained from the Fourier Transform of the probability distribution of $|1\rangle$ state in Fig. 4.3. The line indicates the Fourier Transform of the analytical evolution for the $|1\rangle$ state without including noise. Taken from Ref. [9]

$\lambda_k^3 - \lambda_i^3$ energy. The first eigenvalue will then be determined by solving the linear system with all the results. The use of the cube of the Hamiltonian guarantees the correct sign of the eigenvalue.

The results obtained for the energies of the two neutron system are shown in Tab. 4.4. We can note that such results are compatible with the exact ones. The introduced scheme will be updated in Ch. 5 using a hybrid scheme with a classical device to approximately evolve the spatial degrees of freedom. We named this method *coprocessing*.

At the end of this chapter, the applicability of this method for a different quantum system, the Hydrogen atom, expanded by the STO-2G Hamiltonian, is shown.

4.3.1 Experimental results using LLNL testbed

This subsection will present the actual results obtained from the LLNL quantum testbed.

Primarily, the map between physical states and transmon levels was changed representing the three triplet states of the neutron spins by the three levels of the transmon.

The same procedure described before was performed, but there is a variation: the real time evolution is computed from a sequence of a set of real time propagators. For example, suppose that the real time propagators for final intervals

λ_{ij}	Exact (MeV)	Simulated (MeV)
$\lambda_2 - \lambda_1$	2.5254	2.55(2)
$\lambda_1 - \lambda_0$	3.3951	3.41(3)
$\lambda_2 - \lambda_0$	5.9205	5.93(1)

Table 4.3: Difference between energies ($\lambda_i - \lambda_j$) computed from the Discrete Fourier Transform of the occupation probability (Simulated column) and the analytical ones (Exact column). Note that two states are degenerate. Taken from Ref. [9]

Energy	Exact (MeV)	Simulated (MeV)
$\lambda_0 = \lambda_1$	-2.329	-2.3(2)
λ_2	1.066	0.9(6)
λ_3	3.592	3.6(2)

Table 4.4: Exact and simulated energies for the operator V_{SD} obtained from the quantum simulation. Taken from Ref. [9]

$\Delta t = 0.1, 0.2, 0.5, 1.0$ MeV^{-1} are available. The various time steps are computed, optimizing the sequence between these basic propagators. For example, the time $t = 8.9$ MeV^{-1} would be computed applying eight times the propagators with $\Delta t = 1.0$ MeV^{-1}, one with $\Delta t = 0.5$ MeV^{-1}, two with $\Delta t = 0.2$ MeV^{-1} and one $\Delta t = 0.1$ MeV^{-1}.
Fig. 4.5 shows the results of the actual simulation. The circles are the actual occupation probabilities obtained from the quantum testbed. The lines indicate the analytic evolution of the two neutrons.

4.4 Results for the Hydrogen atom

We have discussed how we can extract the spectrum of a Hamiltonian from the occupation probability. Ref. [9] proves it with a numerical simulation. However, the actual results with the LLNL quantum testbed were obtained in 2021. Before that, we need a single qubit to describe the states. Using the procedure described in App. C.1.1, one derives a 2×2 Hamiltonian that can be simulated in a quantum computer with the correct spectra of Hydrogen atom system.
We mapped the first Gaussian to the $|0\rangle$ and the second one to $|1\rangle$. We implemented as quantum gate the propagator e^{-itH} in order to extract the Hamiltonian spectra as we discussed in the previous sections.
We used the same quantum simulator for a transmon qubit of the previous sec-

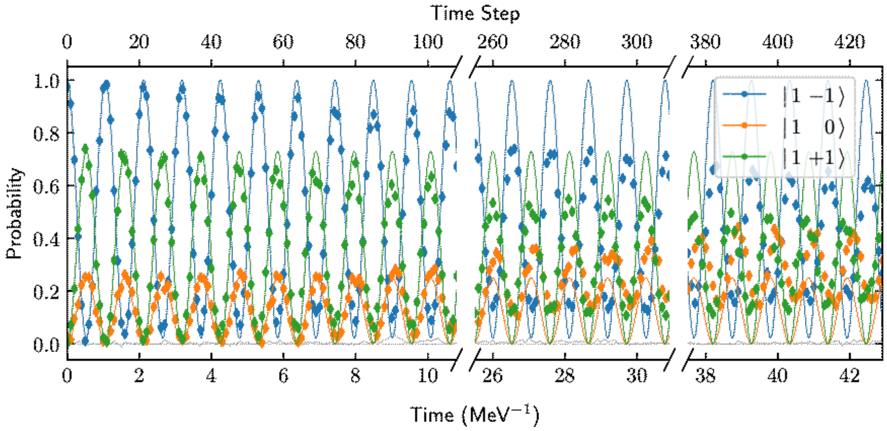


Figure 4.5: Occupation probability for the two neutron spin evolution from the LLNL quantum testbed. Circles indicate the real data and lines the analytic evolution. Taken from Ref. [75]

tion with parameters shown in Tab. 4.5 and the (*Rigetti Aspen−4*) quantum processor [76]. We computed the optimal pulse through the GRAPE algorithm for a transmon device (see App. A for algorithm's details). For the Righetti QPU, the logical gates were translated as a sequence of its elementary gates (R_x, R_z and CZ) through the PyQuil Software Development Kit [77]. The output of the translation optimizes the quantum circuit. So at each time step, the quantum circuit of Righetti has the same depth but with different gates.

The occupation probabilities as a function of time for the two different simulations are shown in Fig. 4.6a. The lines indicate the analytical evolution. The squares and circles represent respectively the results of the device level simulations of qubit transmon and Righetti, respectively.

The Righetti simulation follows approximately the analytical evolution. In the qubit transmon data, we instead observe decoherence. Indeed, in the qubit transmon simulation, the evolution is implemented through a sequence of propagators. On the contrary, we do not have this issue in the Righetti calculation because the software internally optimizes the gate, shrinking the sequence of gates into a single one. The parameter of qubit transmon simulation through optimal control is shown in Tab. 4.5.

Fig. 4.6b shows the Discrete Fourier Transformations of these occupation probabilities. We see that the obtained peaks are compatible with the exact peaks obtained from the Hamiltonian spectrum. Iterating the same algorithm changing the Hamiltonian in the propagator in H^3, we get the results of Fig. 4.7.

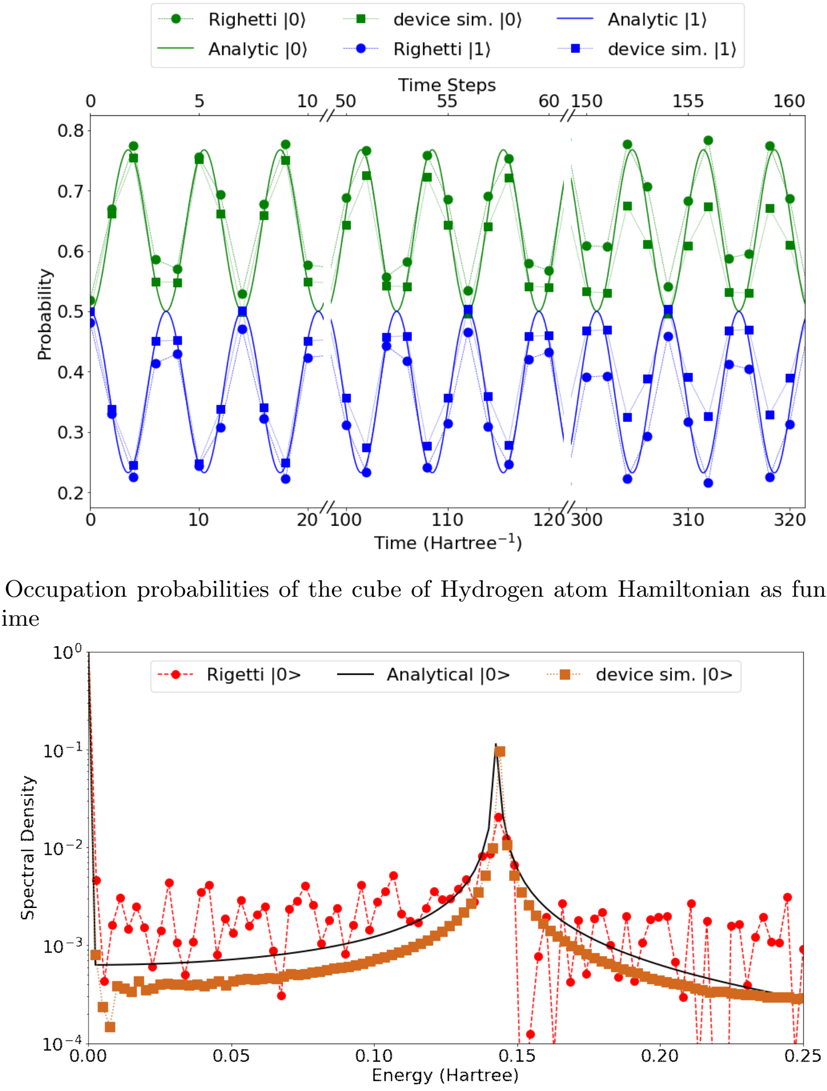
Tab. 4.6 presents the obtained results of two simulations with the analytic values. The energies in the table are expressed in Hartree units, where 1 Hartree is defined as twice the binding energy of Hydrogen atom, i.e., 1 Hartree $\simeq$ 27.21 eV.

Transmon parameters	3D transmon anharmonicity α_T	200 MHz
	Transmon states	2
Noise parameters	Relaxation time T_1	7 μs
	Dephasing time T_2	50 μs
GRAPE parameters	Pulse duration (pd)	50 ns
	Pulse sampling frequency (s_{rate})	32 Gsamples
	Number of points of the pulse	1600
	Fidelity error	10^{-8}
	Initial pulse	A zero-pulse
	Maximum Drive strength	20 MHz

Table 4.5: Parameters of the superconducting transmon qubit, noise of the quantum machine and of optimal control

	Righetti (Hartree)	transmon qubit (Hartree)	Exact (Hartree)
$\lambda_1 - \lambda_0$	0.1432(8)	0.1439(8)	0.14285
$\lambda_1^3 - \lambda_0^3$	0.0309(8)	0.0303(8)	0.02924
λ_0	-0.516(15)	-0.508(17)	-0.48199292
λ_1	0.383(5)	0.389(5)	0.415579

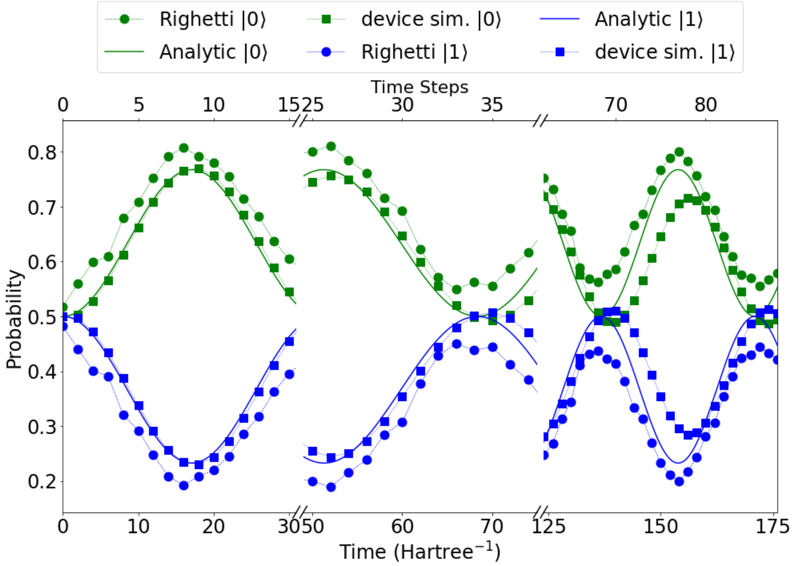
Table 4.6: Analytic (Exact column) and simulated results of Righetti and the device level simulation of transmon qubit for the Hydrogen STO-2G Hamiltonian. In bracket, the uncertainties of the Righetti and the transmon qubit results are shown.



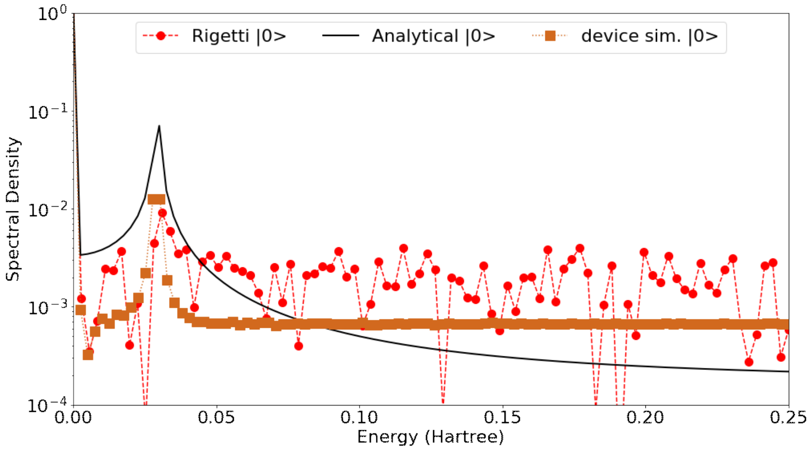
(a) Occupation probabilities of the cube of Hydrogen atom Hamiltonian as function of time

(b) the Discrete Fourier Transform of $|0\rangle$ probability

Figure 4.6: Results for real time evolution and the Fourier transform of probability for Hydrogen atom. The lines indicate the analytical evolution. The circles and squares represent respectively the results of the device level simulations of qubit transmon and Righetti, respectively



(a) Occupation probabilities of the cube of Hydrogen atom Hamiltonian as function of time.



(b) the Discrete Fourier Transform of $|0\rangle$ probability

Figure 4.7: Results for real time evolution and the Fourier transform of the measured probability for the Hydrogen atom for H^3. The lines indicate the analytical evolution. The circles and squares represent respectively the results of the device level simulations of qubit transmon and Righetti, respectively

Chapter 5

Quantum-classical simulations in time evolution

The first simulation presented in Ch. 4 was the real time evolution of a spin system of two spatial blocked neutrons. This chapter generalizes it, (re-)including the spatial evolution. The exposed method, named *Coprocessing*, employs a hybrid calculation between a classical device and a quantum one. Specifically, the spin of neutrons will be computed through a quantum processor and the space dynamics through a classical device. The main aim of this method is to computationally simulate a scattering process starting from incoming particles and obtaining the probability (or the final state) of the outgoing particles through real time evolution.
We will start exposing the coprocessing scheme and showing the results obtained from the device level simulation on the transmon qudit of Tab. 4.5. Then, we will present the results obtained using the Advanced Quantum Testbed (AQT) testbed [8] in different approaches. In the end, we will show a method based on a reinitializing procedure to possibly improve the noisy results. Results will be reported.

5.1 Coprocessing scheme for nuclear scattering simulation

In the first sections of Ch. 4 we have dealt with the nuclear system studied in Ref. [9]. Its authors simulated the real time dynamics of the spin system of two neutrons that are fixed in their position.
We will briefly recap the fundamental steps to simulate the two neutrons' real

time evolution. The entire Hamiltonian of Ref. [9] is given by:

$$H = T + V_{SI} + V_{SD} \tag{5.1}$$

where T indicates the kinetic energy, V_{SI} the spin independent potential, V_{SD} the spin dependent one. (see App. C.2 for details)
The quantum gate to study the real time evolution defined by:

$$U(t) = e^{-i\Delta t H} \simeq \exp\{-i\Delta t\,(V_{SI} + T)\}\,\exp\{-i\Delta t\, V_{SI}\} + O(\Delta t^2)\,, \tag{5.2}$$

where we have used the Trotter decomposition to split the propagator in the limit of $\Delta t \to 0$.
The red factor evolves the spatial degrees of freedom and the blue operator the spin ones. In Ch. 4 and Ref. [9] only the blue propagator is considered because the two neutrons are deemed blocked in space.
A fundamental problem remains: how can we include the spatial degrees of freedom (red operator of Eq. (5.2)) to simulate a real scattering process of nuclear particles?
We have two possible solutions. The more formal method would treat the whole propagator as a true quantum operator, expanding the states of the system in a basis set that includes all the spatial, spin and isospin degrees of freedom. Then the states would be mapped on the levels of a quantum processor. We will compute the quantum circuit that implements the real short-time evolution described by this Hamiltonian. This method can require more qubits than those presented in today's quantum machine and also an excellent compiler to translate the propagator in gates. Therefore, we can conclude that this approach cannot be implemented in today's quantum processors. A further discussion will be presented in Ch.6.
The other approach uses a hybrid algorithm to study the spatial spin-isospin dynamics. The idea is to simulate the spin-isospin evolution in a quantum processor and the spatial component in a classical device. We name it coprocessing scheme. Therefore, the real time evolution of two neutrons would be obtained by the following scheme:

1. Map the spin states of two neutrons in the quantum processor levels.
2. At each time step $\Delta\, t$:
   (a) Evolve spatial coordinates with standard algorithms in classical devices.
   (b) Compute propagators that evolves the spin from time t to $t + \Delta\, t$ using the following equation:

   $$U_{spin}(t) = \mathbb{T}\,\exp\left(-i\int_t^{t+\Delta t} dt H_{SD}(r(t))\right) \tag{5.3}$$

   where $\mathbb{T}$ is the time order operator and H_{SD} is the spin-dependent Hamiltonian that usually depends on the spatial components.

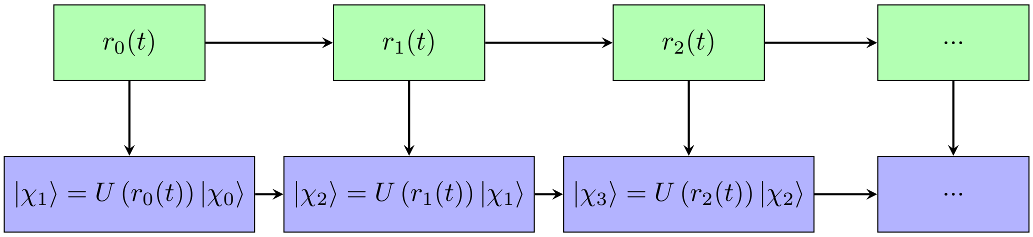

$r_0(t)$
$r_1(t)$
$r_2(t)$
$\cdots$
$|\chi_1\rangle = U\left(r_0(t)\right)|\chi_0\rangle$
$|\chi_2\rangle = U\left(r_1(t)\right)|\chi_1\rangle$
$|\chi_3\rangle = U\left(r_2(t)\right)|\chi_2\rangle$
$\cdots$

Figure 5.1: Coprocessing scheme. The spatial evolution is computed by solving the Newton equation. For the spin evolution, we must update the new relative position to compute the spin propagator at each step. The top row represents the spatial evolution at each time step, bottom row the spin evolution. The time arrow goes from left to right

(c) Translate the spin propagators in a quantum circuit through digital gates or the optimal control approaches.

(d) Implement the quantum circuit.

(e) If requested to evolve the spatial dynamics, identify the spin state.

This chapter will present numerical device level and actual results using this scheme for the study of two neutrons dynamics using the potential derived by Leading Order chiral EFT (see App. C.2).
The results shown in this chapter are obtained by solving the Newton equation for spatial dynamics. This simplification was not done to reach realistic results but solely to test the coprocessing method. However, from a quantum point of view, the obtained classical trajectory describes the most probable path that the two neutrons could do as the saddle point approximations guarantees [78].
Looking closer at the potential, the spin-independent operator (that evolves the spatial DoFs) does not depend explicitly on the spin state. Therefore, we can further simplify the calculations by computing the spatial evolution before the quantum simulation. Fig. 5.1 shows a scheme of the simulations.

5.2 Coprocessing scheme using optimal control approach with pulse fitting

This section will focus on the coprocessing method where we performed device level simulation on the transmon device of Tab. 4.5 implementing the Optimal Control (OC) approach through the Gradient Ascent Pulse Engineering (GRAPE) algorithm (see App. A.2 for details). The obtained results from these simulations would be very close to ones that we would get running on LLNL quantum testbed [37].
We start from the spatial dynamics of the two neutrons. As we said before,

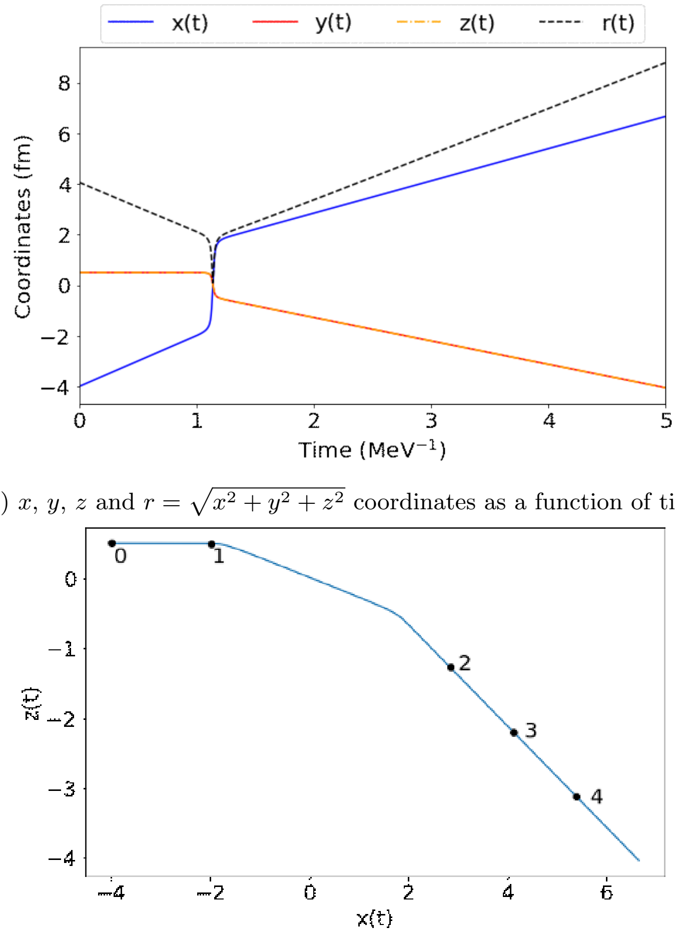
the trajectory can be computed previously or at the same time of the spin evolution. We solved the Newton equation of two neutrons through the velocity Verlet integration [79, 80]. Their initial position and velocity in their relative frame system[1] are $x(t = 0) = [4., 0.5, 0.5]$ fm and $v(t = 0) = [0., -5.0, 0.]$ fm MeV. Specifically, our integration method at each time step t_i is based on:

1. $v(t_i + \frac{\Delta t}{2}) = v(t_i) + \frac{\Delta t}{2} a_i$ with $a_i = \frac{F(x(t_i), v(t_i))}{\mu}$ (F is the radial force in Newton equation and μ the two neutron reduced mass);
2. $x(t_{i+1}) = x(t_i) + v(t_i + \frac{\Delta t}{2})\Delta t$;
3. $v(t_{i+1}) = v(t_i + \frac{\Delta t}{2}) + \frac{\Delta t}{2} a_{i+\frac{1}{2}}$ with $a_{i+\frac{1}{2}} = \frac{F(x(t_{i+1}), v(t_{i+\frac{1}{2}}))}{\mu}$.

Fig. 5.2 shows the results of the three spatial components evolution as a function of time. This simulation was computed used a short-time step $\Delta\, t = 0.0025$ MeV^{-1} with a total number of time step of $N = 2000$.
According to the described coprocessing scheme, the spin evolution is simulated in a quantum processor. We numerically perform device level simulations using a 4 level transmon (using the parameters of Tab. 4.5). The implemented map is shown in Sec. 4.2. We apply the Fourier Transform interpolation algorithm described in App. B for obtaining the experimental pulses that drive the transmon dynamics in the desired nuclear real time evolution. In this simulation, the time step is $\Delta t = 0.01$ MeV^{-1} and the number of time steps is $N = 100$.
The obtained results are shown in Fig. 5.3. The circles represent the data without including the quantum noise. Instead, the squares show the data obtained from solving the Lindblad Master equation with realistic noise sources. Lines represent the analytical evolution. We can observe that the results obtained in the absence of the quantum noise closely follow the analytic evolution. A very light shift can be observed at the end of the simulation; it may be due to some numerical errors or small infidelity contributions. Nevertheless, this figure shows that our interpolation model of the pulses works. Instead, the numerical data computed from the Master equation present a substantial problem. After some time step, the noise ruins the simulation. Therefore, we could not extract any desired physical information in this time interval anymore. For instance, if we are interested in obtaining the final spin probability, through the simulated results of Fig. 5.3 we will get wrong values due to the contribution of noise.

As we have observed, simulating the short-time spin evolution for the total number of time steps requires clean qubits (with a very long decoherence time). This experimental constraint would avoid the quantum noise killing the simulation in today's quantum machine. Unluckily, today's quantum processors do not have this condition .
A simple solution is to group the short-time propagators in a small set that describes the same evolution but with a longer time step $\{T_i\}$. Using this coarse set, we can study the entire scattering process. Hence, the spin propagator from

[1] r and v variables are defined from $\vec{r} = \vec{r_2} - \vec{r_1}$ and $\vec{v} = \vec{v_1} - \vec{v_2}$



(a) x, y, z and $r = \sqrt{x^2+y^2+z^2}$ coordinates as a function of time

(b) Trajectory in relative frame in the plane $x-z$. The enumerated points correspond to the position at time t = 0, 1, 2, 3, 4 MeV^{-1}

Figure 5.2: Spatial evolution

time T_j to T_{j+1} is computed from the short-time propagators as it follows:

$$\begin{aligned} U(T_j) &= \mathbb{T}\prod_{k=0}^{K} e^{-i\Delta t H_{SD}(r(t_k))} = \\ &e^{-i\Delta t H_{SD}(r(t_K))}\, e^{-i\Delta t H_{SD}(r(t_{k_1}))} \dots e^{-i\Delta t H_{SD}(r(t_1))} e^{-i\Delta t H_{SD}(r(t_0))}\,, \end{aligned} \tag{5.4}$$

where $\mathbb{T}$ is the time order operator, $e^{-i\Delta t H_{SD}(r(t_k))}$ are the discretized short-time propagator at time k.
We computed the set of coarse propagator $\{U(T_j)\}$ and through GRAPE algorithm (see App. A.2) we tailor their pulses. We perform a device level simula-

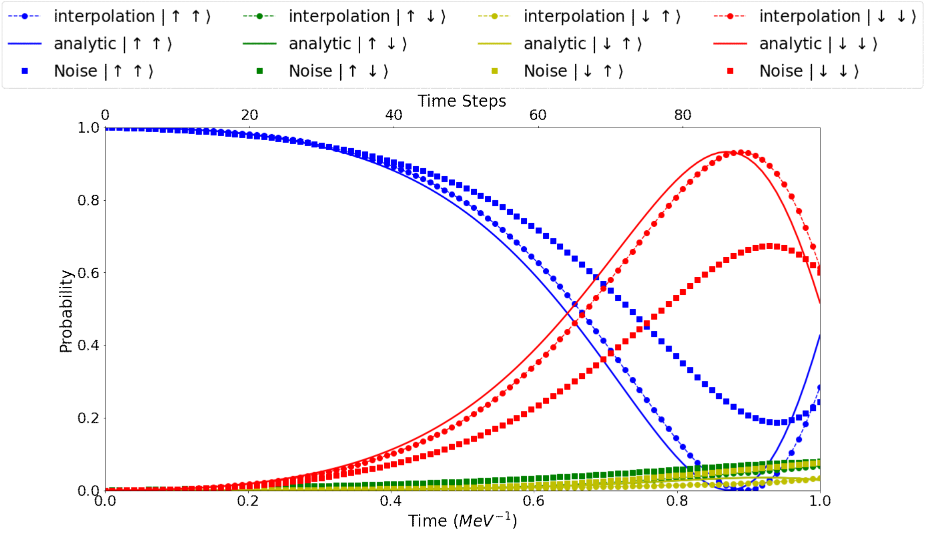


Figure 5.3: Results of the spin evolution of two neutrons in the coprocessing approach. The simulation is computed using a four-level transmon with the interpolation of the pulse of App. B. The squares and circles represent the results with and without the inclusion of noise, respectively. The lines indicate the analytic evolution.

tion solving the Lindblad Master equation. The parameters of the simulation are shown in Tab. 4.5.
Fig. 5.4 shows the obtained numerical results. The circles indicates the numerical results and the line the exact spin evolution. We observe that the simulated evolution is very close to the analytic one.

5.3 Coprocessing scheme using discrete gate sets on the LNBL Advance Quantum Testbed (AQT)

This part of the work pertains to a collaboration with the Advance Quantum Testbed (AQT) team in the Berkeley Lawrence Berkeley National Laboratory(LNBL), established at the beginning of 2021. In this Laboratory, a quantum processor is currently developed [8]. The Trento/LLNL group participated in the collaboration providing the theory and codes for the simulation of the two neutron problem described in the previous section. The experimental group at LBNL ran the quantum simulations on their device.
The difference between the AQT and the LLNL quantum testbeds is in the implementation of the gates. The quantum testbed at LBNL is based on applying

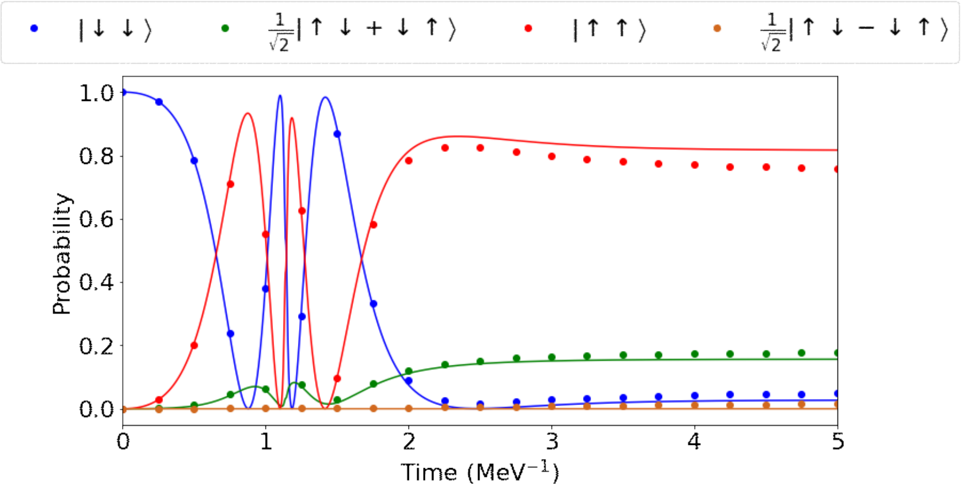


Figure 5.4: Results of the numerical simulation of LLNL testbed of the spin real time evolution of two neutrons using the coprocessing scheme with the OC method. Lines represent the analytic evolution and circles the numerical results

a sequence of digital gates. In the LLNL device, we apply the Optimal Control method.
The quantum codes needed to be adopted in terms of a decomposition of the long real-time evolution propagators (Eq. (5.4)) in elementary gates. In our case, the fundamental digital gates are R_x, R_z and $CNOT$ ones (see Sec. 2.2 and Tab. 2.1). The propagators are computed from Eq. (5.4) and they are translated with *decompose* function of *QISKit* open-source software development kit [81].
In calculations, different strategies will be implemented on the AQT QPU to simulate the quantum evolution of a simple scattering process. They will be presented in the following subsections as they were implemented in chronological order.
The applied approaches study the same two neutron evolution of Sec. 5.2. Fig. 5.2 recaps the spatial evolution of the two neutrons.
Like as we did in Sec. 5.2, our simulation starts from computing the entire classical trajectory solving the Newton equation. The short time spin propagators are computed and they are grouped using Eq. (5.4). Hence, we will have $N_{tot} = 20$ coarse operators $U_{TOT}(t_j)$.

5.3.1 Simulation using a full-optimized quantum circuit

The first implemented quantum circuits are composed of two quantum gates. Fig. 5.5 shows a scheme of such quantum circuit. The first implements a quan-

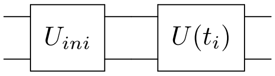


Figure 5.5: Scheme of gates for the quantum circuit of the full-optimized quantum circuit. The first gate represents the reinitializing gate and the second gate evolves from time t_i to t_{i+1}. In the actual simulation each two qubit gates is decomposed by three $CNOT$ gates and 10 U_3 gate

tum operation that evolves the state of the quantum processor from the default state $|0\rangle$[2] to the state at time t_i, that is given by

$$U_{ini}(t_i) = \exp\left[-i\int_0^{t_i} dt H_{SD}(r(t))\right] = \mathbb{T}\prod_{j=0}^{t_i} U_{TOT}(t_j)\,. \tag{5.5}$$

The second gate applies the coarse propagator $U(t_i)$ that evolves the spins from time t_i to t_{i+1}.

We call this scheme a fully optimized approach. Indeed, for the time t_i all the sequence of propagators before t_i are grouped in a single gate, called reinitializing gate, $U_{reini} = U(t_i)\cdot ...U(t_1)\cdot U(t_0)$. In other words, we have optimized the quantum circuit of the sequence of propagators in a single two-qubit gate. In App. E, an actual implementation of a quantum circuit is shown.

We should highlight that each quantum circuit does not depend on the others because the reinitializing operators are computed from the theoretical propagators. This approach was our first test of the coprocessing method with discrete gates.

Fig. 5.6 presents the obtained results. The circles indicate the real results of the AQT quantum processor and lines represent the analytic evolution. We can see good compatibility between data and theoretical evolution.

However, this approach carries no information from a physicist point of view. We added further information to simulate the spin evolution. Specifically, the reinitializing gate is computed from the analytic propagators.

5.3.2 Simulation using a sequence of propagators

After applying the full-optimized approach, the same scheme of Sec. 5.2 was implemented. The two neutron scattering was simulated using a sequence of the real time coarse propagators $\{U_j\}$ (computing from Eq. (5.4)). This strategy is more realistic than a full-optimized approach.

Fig. 5.7 shows an example of such quantum circuit. Once more, in App. E the actual quantum circuit for the time step $t = 8$ is shown. One can observe that the presented quantum circuit is very deep. The very large number of gates

[2] if one wants to start from a different state, one must implement another two-qubit gate that moves from $|0\rangle$ to the desired state

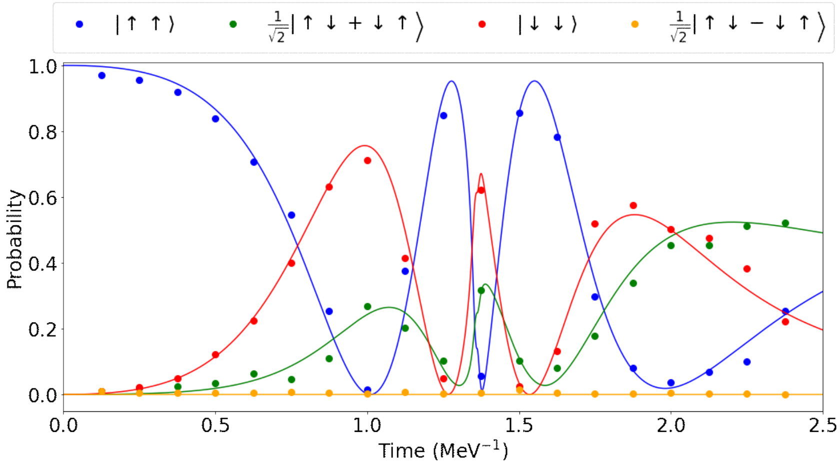


Figure 5.6: Results from the simulation of coprocessing approach using the full-optimized quantum circuits. The spin evolution is plotted as a function of time. For the spin evolution, circles indicate the simulated results, the lines the analytic evolution. In this figure, the simulated spin evolution time is one half of the time shown for the coordinates

should be noticed.
The obtained results are shown in Fig. 5.8. Circles indicate the actual results of the AQT quantum processor, and lines represent the analytic evolution. After a few time steps, we observe that the noise impairs the simulation.
The problem of noise in the simulation is an issue to deal with. Indeed, for studying the real-time evolution of complex systems, all the physical information may be lost in very few steps using today's quantum processor.
We studied possible solutions from the software side to increase the accuracy of the quantum simulations. One should implement error mitigation procedures at the hardware level to remove some or all contribution of the noise present. The experimentalists employed error mitigation algorithms to reduce the dephasing contribution from the AQT quantum processor. The results can be seen later in Fig. 5.11. Even though the accuracy is increased, the obtained results are not close to the analytic evolution yet.
We also study an algorithm that tries to overcome noise problems. It is based on the reinitializing procedure and is reported in the following sections.

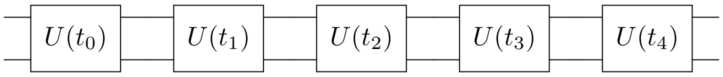


Figure 5.7: Schematic quantum circuit for a sequence of real time propagators

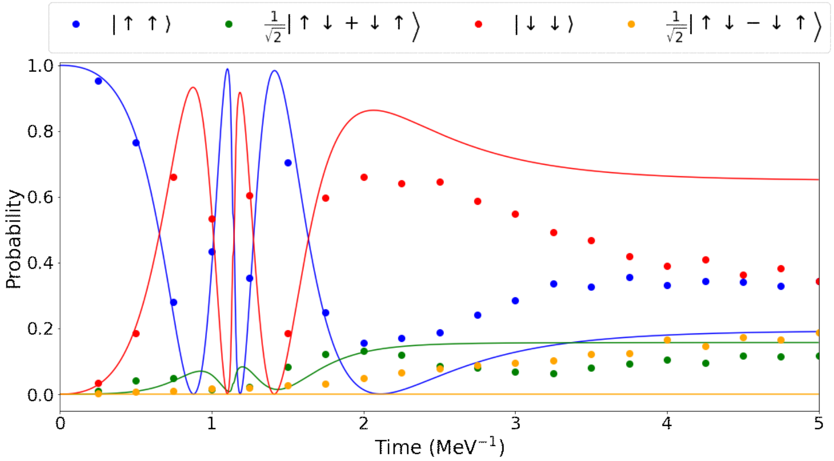


Figure 5.8: Obtained results using coprocessing results through the quantum circuits given a sequence of real time propagators. The coordinate evolution as a function of time are shown in Fig. 5.2

5.3.3 Reinitializing procedure

Our solution to overcome the noise problem is to merge the two approaches presented in previous subsections.
We can observe that after a few time steps, the actual results of the quantum processor in Fig. 5.8 are very close to the analytic evolution and have a small contribution to the noise. Specifically, we can conclude that we still have good compatibility after $3-4$ time steps. Therefore, if we could stop here, the state of the quantum processor, $|\phi\rangle$ would be very close to the analytical one. Hence, the evolution of the next time steps would be much closer to the analytic one if the initial state of the quantum processor could be reinitialized on the state $|\phi\rangle$. Indeed, we reduce the actual machine time for which the noise starts destroying the simulation.
The quantum circuit in this approach is based on the first gate that reinitializes the state at time t and the following gates that evolve the spin from time t. An example of this quantum circuit is shown in Fig. 5.9, where the gate U_{ini} evolves the initial state, $|0\rangle$, in the state previously obtained at time $t=2$. Then, we implement the real time evolution propagators that evolves from $t=2$ to $t=4$. In App. E an actual quantum circuit is reported.

Results of reinitializing approach are shown in Figs. 5.10 and 5.11. The two figures differ from the time step where we reinitialize. However, we should consider that the reinitializing gate is computed from the exact propagators. Fig. 5.11 also shows results that have been corrected through error mitigation algorithms. Specifically, the experimentalists corrected readout error and gate infidelity.
Looking at the figures, we observe that we start to reduce the contribution of

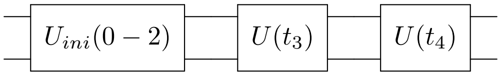


Figure 5.9: Scheme for the quantum circuit in the merged approach. We took the quantum circuit shown in Fig. 5.7 and the three gates ($U(t_0)$, $U(t_1)$ and $U(t_2)$) are grouped in one single two qubit $U_{ini}(0-2)$. The $U_{ini}(0-2)$ represents the reinitilizing gate

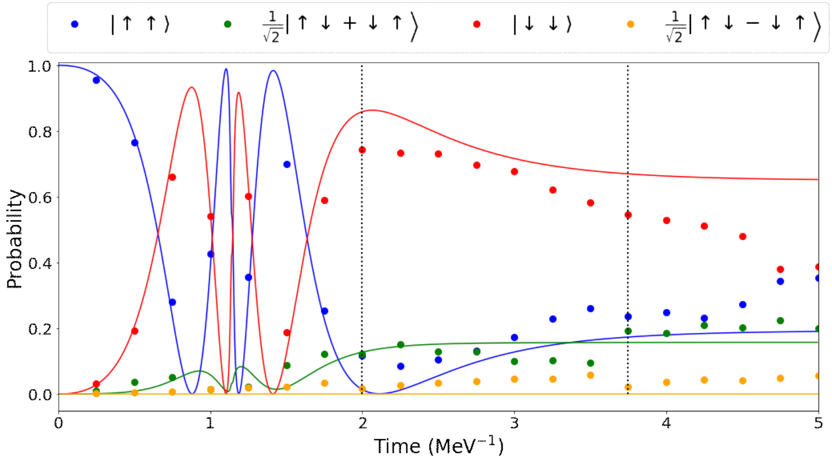


Figure 5.10: Coprocessing results with $T_{reini} = 3$. The dashed lines indicate the steps when there is a reinitializing process.

the noise using this approach.

At the end of this section, we conclude that this merged approach, named reinitializing procedure, solves the quantum noise problem. However, here, we added more information to have accurate results, since the reinitializing gates are determined from the analytical propagators. In the following sections, we will answer the following question: can the real time evolution be simulated using the exposed reinitializing procedure without using analytic propagators in the reinitializing gates? In other words, can the state of the quantum processor be reinitialized using only the experimental data?

5.4 State tomography and reinitialization algorithm

In the results shown before, we have seen that after some application of real time propagators for studying the spin evolution, we have several issues with

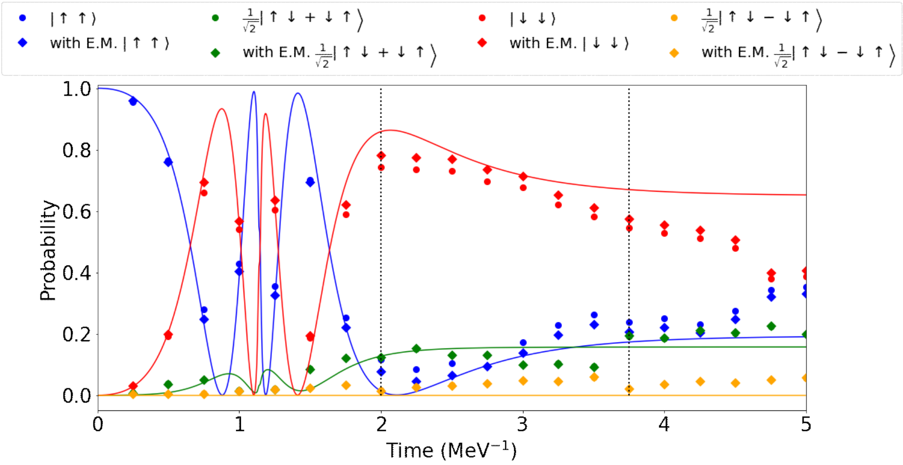


Figure 5.11: Coprocessing results with $T_{reini} = 7$. The dashed lines indicate the steps when there is a reinitializing process. The E.M. results (diamonds) indicate the data corrected through error mitigation algorithms.

the quantum noise that kills the accuracy of our results.

A simple solution to reach a good precision on the evolution is to perform a state tomography process [25] to identify the state at some early time step, when the noise's contribution is not so significant to do. After that, we continue to study real time evolution, but we start from the state that we identified with the state tomography. This initial state should be very close to the final state of the previous run. Iterating this method, we can study long-time dynamics with reasonable accuracy. We have called it reinitializing procedure. It is similar in spirit to the "restarting" procedure proposed in Ref. [82], where the state after one step of dynamical evolution is approximated by optimizing a variational circuit before performing the next time step. We will see that, in general, the state tomography procedure proposed here is not scalable to more than a few qubits but can provide interesting benchmarks on small scale near-term devices.

State tomography is a process that uniquely identifies a given state ρ (pure or mixed) computing some observables. These observables have to be a complete operator basis of the Hilbert space to provide all the state's information.

Let us do a simple example to better clarify. The general state of a single qubit can be written as [25]:

$$\rho = \frac{1}{2}\mathbf{1} + \frac{1}{2}\left(c_x\,\sigma_x + c_y\,\sigma_y + c_z\,\sigma_z\right) \qquad\qquad c_x^2 + c_y^2 + c_z^2 \,\leq 1\,, \tag{5.6}$$

where ρ indicates its density matrix. If $c_x^2 + c_y^2 + c_z^2 \;= 1$ the state is pure, otherwise is a mixed state.

The most common way to identify the state of Eq. (5.6) is to measure the expectation values of the Pauli matrices, X, Y and Z. Their expectation values are linked to the coefficient of the density matrix. In particular, one can find

$c_x = \langle X\rangle$, $c_y = \langle Y\rangle$ and $c_z = \langle Z\rangle$.
For a generic N-qubit system, its density matrix can be decomposed as:

$$\rho = \frac{1}{2^N}\sum_{\vec{v}} \mathrm{Tr}\{O_{v_1}\otimes ...\otimes O_{v_n}\,\rho\}\,O_{v_1}\otimes ...\otimes O_{v_n}\;, \tag{5.7}$$

where $\vec{v} = (v_1,\, ...,\, v_n)$ with entries v_i are chosen from $\{0,1,2,3\}$. The most popular basis set for $\{O\}$ are given by $\{I,\, X,\, Y,\, Z\}$ (identity and Pauli matrices).
The general decomposition of the density matrix of Eq. (5.7) shows the main problem of the state tomography process that is its scalability. In the Hilbert space of n qubit one should measure 4^n observables to uniquely identify the desired state. There are experimental tricks that one can use to reduce the number of quantum circuits to perform a quantum tomography (generally, one implements 3^n quantum circuits). For instance, for two qubits one should run 9 different quantum circuits because the expectation value of XI and IX can be computed from the simulation XX.
Our research developed a different state tomography algorithm. Through it the state is identified reducing the number of implemented quantum circuits. The essential idea of this new state tomography process is to approximate the state inside the quantum processor with a pure state. The rationale of this choice is that reinitialization can be easily achieved by unitary gates.
For example, the general parametrizations of pure states are shown in Eq. (5.8) for a single qubit, and in Eq. (5.9) for two qubits. In these equations, P_i indicates the i state probability and ϕ_i represents the relative phase of the $|i\rangle$ state. We can neglect the phase of the first state because it represents the global phase of the state. A global phase cannot be evaluated through any quantum gates in this Hilbert space.
Our state tomography process will be shown starting from the case of single qubits and two qubits. Then, it will be generalized in the case of n qubit. At the end of this section, we will expose how to build the reinitializing gate from the results of the state tomography.

$$|\psi\rangle \;=\; \begin{pmatrix} \sqrt{P_0} \\ \sqrt{P_1}\,e^{i\,\phi_1} \end{pmatrix} \tag{5.8}$$

$$|\psi\rangle \;=\; \begin{pmatrix} \sqrt{P_0} \\ \sqrt{P_1}\,e^{i\,\phi_1} \\ \sqrt{P_2}\,e^{i\,\phi_2} \\ \sqrt{P_3}\,e^{i\,\phi_3} \end{pmatrix} \tag{5.9}$$

5.4.1 Tomography for a single qubit

Our aim is to evaluate P_0, P_1 and $\phi_r = \phi_1 - \phi_0$ of the following qubit:

$$\psi = \begin{pmatrix} a + i\,b \\ c + i\,d \end{pmatrix} \;=\; e^{i\,\phi_0}\begin{pmatrix} \sqrt{P_0} \\ \sqrt{P_1}e^{i(\phi_1-\phi_0)} \end{pmatrix}\;, \tag{5.10}$$

where we can consider $\phi_0 = 0$ because it represents a global qubit phase. We will start by presenting the state tomography for the state formulation. After that, it will be generalized for the density matrix formulation.
Our state tomography process for a single qubit is based on the following steps.

1. Measure the probabilities of the bare circuit. We thereby obtain P_0 and P_1.
2. Measure the probabilities of the bare circuit with a final $R_y(-\frac{\pi}{2})$ rotation.
3. Measure the probabilities of the bare circuit with a final $R_x(-\frac{\pi}{2})$ rotation.

Why do we need the two rotations? They establish the relative phase uniquely. Applying R_y to the state of Eq. (5.8), we obtain:

$$R_y\left(-\frac{\pi}{2}\right)|\psi\rangle \;=\; \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} \sqrt{P_0} \\ \sqrt{P_1}\, e^{i\,\phi_1} \end{pmatrix} \;=\; \begin{pmatrix} \frac{1}{\sqrt{2}}\left(\sqrt{P_0}+\sqrt{P_1}e^{i\phi_1}\right) \\ \frac{1}{\sqrt{2}}\left(\sqrt{P_1}e^{i\phi_1}-\sqrt{P_0}\right) \end{pmatrix}. \tag{5.11}$$

In particular, computing the probability for the $|0\rangle$ state one gets

$$\begin{aligned} P_0^y &= \frac{1}{2}\left|\sqrt{P_0}+\sqrt{P_1}e^{i\phi_1}\right|^2 = \frac{1}{2}\left(P_0+P_1+\sqrt{P_0\,P_1}\,e^{i\phi_1}+\sqrt{P_0\,P_1}\,e^{-i\phi_1}\right) \\ &= \frac{1}{2}\left(P_0+P_1+2\sqrt{P_0\,P_1}\,\cos(\phi_1)\right) \end{aligned}. \tag{5.12}$$

Therefore, we have:

$$\cos(\phi_1) = \frac{1}{\sqrt{P_0 P_1}}\left(P_0^y - \frac{1}{2}(P_0+P_1)\right). \tag{5.13}$$

We have connected the relative phase with the experimental probabilities. However, we have a problem with the cosine because we will have two possible solutions for ϕ_1. Therefore, another operation should be implemented, for example $R_x\left(-\frac{\pi}{2}\right)$ to univocally determine the angle.
Similarly, implementing $R_x\left(-\frac{\pi}{2}\right)$ to the desired state, we obtain:

$$R_x\left(-\frac{\pi}{2}\right)|\psi\rangle \;=\; \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix}\begin{pmatrix} \sqrt{P_0} \\ \sqrt{P_1}\, e^{i\,\phi_1} \end{pmatrix} \;=\; \begin{pmatrix} \frac{1}{\sqrt{2}}\left(\sqrt{P_0}+i\sqrt{P_1}e^{i\phi_1}\right) \\ \frac{1}{\sqrt{2}}\left(i\sqrt{P_0}+\sqrt{P_1}e^{i\phi_1}\right) \end{pmatrix}. \tag{5.14}$$

Like before, computing the probability for the $|0\rangle$ state one gets

$$\begin{aligned} P_0^x &= \frac{1}{2}\left|\sqrt{P_0}+i\sqrt{P_1}e^{i\phi_1}\right|^2 = \frac{1}{2}\left(P_0+P_1+i\sqrt{P_0\,P_1}\,e^{i\phi_1}-i\sqrt{P_0\,P_1}\,e^{-i\phi_1}\right) \\ &= \frac{1}{2}\left(P_0+P_1-2\sqrt{P_0\,P_1}\,\sin(\phi_1)\right). \end{aligned} \tag{5.15}$$

Therefore, we have:

$$\sin(\phi_1) = \frac{1}{\sqrt{P_0 P_1}} \left(-P_0^x + \frac{1}{2}(P_0 + P_1) \right) . \tag{5.16}$$

With these two equations ((5.13) and (5.16)) we can establish uniquely the angle ϕ_1. Hence, the relative phase ϕ_1 is given by

$$\phi_1 \; = \; \arctan \left(\frac{-P_0^x + \frac{1}{2}(P_0 + P_1)}{P_0^y - \frac{1}{2}(P_0 + P_1)} \right) , \tag{5.17}$$

where the function arctan must be considered as the function which returns a correct and unambiguous value for the angle ϕ_1 (in other words, the numerical $arctan2$ function). In the following, all the arctan will indicate this specific kind of function.

Density matrix formulation

Here, we will show what our state tomography algorithm connects in the density matrix formulation.
The general density matrix for a single qubit can be expressed as follows:

$$\rho = \frac{1}{2}(\mathbb{1} + c_x \, X + c_y \, Y + c_z \, Z) \, . \tag{5.18}$$

Applying $R_x(-\frac{\pi}{2})$ and $R_x(-\frac{\pi}{2})$ gates to ρ, one gets:

$$\rho_x = R_x(-\frac{\pi}{2})\rho R_x(-\frac{\pi}{2})^\dagger = \frac{1}{2}(1 + c_x \, X - c_y \, Z + c_z \, Y) \tag{5.19}$$

and

$$\rho_y = R_y(-\frac{\pi}{2})\rho R_y(-\frac{\pi}{2})^\dagger = \frac{1}{2}(1 + c_x \, Z + c_y \, Y - c_z \, X) \, . \tag{5.20}$$

Measuring the $|0\rangle$ state for both the two density matrices, one obtains:

$$P_0^x \; = \; \frac{1}{2}(1 - c_y) \qquad \text{and} \qquad P_0^y \; = \; \frac{1}{2}(1 + c_x) \, . \tag{5.21}$$

Using the formula in eqs. (5.13) and (5.16) in the case of a general density matrix, one gets

$$\begin{aligned} \phi_1 &= \arctan \left(\frac{(-P_0^x + \frac{1}{2}(P_0 + P_1))}{(P_0^y - \frac{1}{2}(P_0 + P_1))} \right) \\ &= \arctan \left(\frac{-\frac{1}{2}(1 - c_y) + \frac{1}{2}}{\frac{1}{2}(1 + c_x) - \frac{1}{2}} \right) \\ &= \arctan \left(\frac{c_y}{c_x} \right) . \end{aligned} \tag{5.22}$$

We have linked the relative phase to the two parameters c_x and c_y.
To show how the proposed state tomography procedure is robust respect to quantum noise, we use the same Bloch-Redfield model for density matrix [61, 62, 63] used in Sec. 3.3 to write the noise contribution to the qubit state. Starting from the pure qubit state, $\alpha\left|0\right\rangle + \beta\left|1\right\rangle$, the contribution of noise sources can be described by the following density matrix

$$\rho = \begin{pmatrix} 1 + (|\alpha|^2 - 1)e^{-\frac{t}{T_1}} & \alpha\beta^* e^{-\frac{t}{T_2}} \\ \alpha^*\beta e^{-\frac{t}{T_2}} & |\beta|^2 e^{-\frac{t}{T_1}} \end{pmatrix} . \tag{5.23}$$

Looking at the dephasing contribution $e^{-\frac{t}{T_2}}$, we can conclude that our state tomography is insensible to this noise source. Indeed, the dephasing process changes c_x in $c_x^1 = c_x e^{-\frac{t}{T_2}}$ and c_y in $c_y^1 = c_y\, e^{-\frac{t}{T_2}}$ according to Bloch-Redfield density matrix. But, when ϕ_1 is computed, we have the ratio between c_x^1 and c_y^1. The result is the true value for ϕ_1.
Tab. 5.1 shows the expectation values of $\langle X\rangle$, $\langle Y\rangle$ and $\langle Z\rangle$ and the fidelity with the state without noise obtained using the exposed tomography method with the Bloch Redfield density matrix (see Eq. (5.23)) for the some random states. We can conclude that our purification and tomography procedure resists the dephasing error but not the relaxation process because it changes the values of P_0 and P_1.
App. D reports further studies about differences between standard and presented state tomography processes.

5.4.2 Tomography for 2 qubits

For the tomography of two qubits, we can find a similar algorithm to the single qubit case.

1. Measure the probabilities of the bare circuit
2. Measure the probabilities of the bare circuit with a final $\mathbb{1} \otimes R_y(-\frac{\pi}{2})$ and $\mathbb{1} \otimes R_x(-\frac{\pi}{2})$ rotations.
3. Measure the probabilities of the bare circuit with a final $R_y(-\frac{\pi}{2}) \otimes \mathbb{1}$ and $R_x(-\frac{\pi}{2}) \otimes \mathbb{1}$ rotations.

According to calculations of the previous subsection, one can prove that the second steps given us

$$\phi_1 \;=\; \arctan\left(\frac{-P_0^x + \frac{1}{2}(P_0 + P_1)}{P_0^y - \frac{1}{2}(P_0 + P_1)}\right) , \tag{5.24}$$

$$\phi_3 - \phi_2 \;=\; \arctan\left(\frac{-P_2^x + \frac{1}{2}(P_2 + P_3)}{P_2^y - \frac{1}{2}(P_2 + P_3)}\right) . \tag{5.25}$$

	$\langle X\rangle$	$\langle Y\rangle$	$\langle Z\rangle$	fidelity
$0.9177\,\lvert 0\rangle + (0.1479 + 0.3687\,i)\,\lvert 1\rangle$				
no noise	0.2714	0.6767	0.6844	1.0000
$\frac{1}{T_1} = 0$ $\frac{1}{T_2} = 1$	0.2714	0.6767	0.6844	1.0000
$\frac{1}{T_1} = 1$ $\frac{1}{T_2} = 1$	0.1741	0.4341	0.8839	0.9864
$0.1252\,\lvert 0\rangle + (0.6997 + 0.7034\,i)\,\lvert 1\rangle$				
no noise	0.1751	0.1761	-0.9687	1.0000
$\frac{1}{T_1} = 0$ $\frac{1}{T_2} = 1$	0.1751	0.1761	-0.9687	1.0000
$\frac{1}{T_1} = 1$ $\frac{1}{T_2} = 1$	0.6779	0.6815	0.2758	0.6970
$0.6689\,\lvert 0\rangle + (0.7396 + -0.0747\,i)\,\lvert 1\rangle$				
no noise	0.9894	-0.0999	-0.1051	1.0000
$\frac{1}{T_1} = 0$ $\frac{1}{T_2} = 1$	0.9894	-0.0999	-0.1051	1.0000
$\frac{1}{T_1} = 1$ $\frac{1}{T_2} = 1$	0.8008	-0.0809	0.5935	0.9322
$0.6038\,\lvert 0\rangle + (-0.2969 + -0.7397\,i)\,\lvert 1\rangle$				
no noise	-0.3586	-0.8933	-0.2708	1.0000
$\frac{1}{T_1} = 0$ $\frac{1}{T_2} = 1$	-0.3586	-0.8933	-0.2708	1.0000
$\frac{1}{T_1} = 1$ $\frac{1}{T_2} = 1$	-0.3153	-0.7855	0.5325	0.9139

Table 5.1: Results of $\langle X\rangle$, $\langle Y\rangle$, $\langle Z\rangle$ and fidelity with the state without noise implementing the exposed tomography method with the Bloch Redfield density matrix using randomic qubit states

The probabilities of the third step link the relative phases as follows

$$\phi_2 = \arctan\left(\frac{-P_0^x + \frac{1}{2}(P_2 + P_0)}{P_0^y - \frac{1}{2}(P_2 + P_0)}\right) , \tag{5.26}$$

$$\phi_3 - \phi_1 = \arctan\left(\frac{-P_1^x + \frac{1}{2}(P_1 + P_3)}{P_1^y - \frac{1}{2}(P_1 + P_3)}\right) , \tag{5.27}$$

From these equation we can evaluate all the phase solving the linear system composed by equations (5.24), (5.26) and (5.26). This algorithm will be implemented for simulating the real time spin evolution of two neutrons in Sec. 5.5.

Generalization for n qubit

For the case of n qubit $2\,n + 1$ quantum circuits should be implemented to evaluate the state. The previous algorithm can be generalized as it follows:

1. Measure the bare probability

2. For $i = 0$, 1, ..., n.
    - Measure the probability after the application of R_x and R_y to qubit i and identities to the others.

For instance, the first two operations are described by $R^0_x = \mathbb{1} \otimes .... \otimes \mathbb{1} \otimes R_x(-\frac{\pi}{2})$ and $R^0_y = \mathbb{1} \otimes .... \otimes \mathbb{1} \otimes R_y(-\frac{\pi}{2})$. Their actions gives us the following relation:

$$\phi_{i+1} - \phi_i = \arctan\left(\frac{-P^x_i + \frac{1}{2}(P_i + P_{i+1})}{P^y_i - \frac{1}{2}(P_i + P_{i+1})}\right) \qquad i = 0, 2, 4, ..., 2^n - 2\,. \tag{5.28}$$

The second step of rotations, given by $R^1_x = \mathbb{1} \otimes .... \otimes R_x \otimes \mathbb{1}$ and $R^1_y = \mathbb{1} \otimes .... \otimes R_y \otimes \mathbb{1}$ connects the relative phases as it follows:

$$\phi_{i+2} - \phi_i = \arctan\left(\frac{-P^x_i + \frac{1}{2}(P_i + P_{i+2})}{P^y_i - \frac{1}{2}(P_i + P_{i+2})}\right) \qquad i = 0, 2, 4, ..., 2^n - 2\,. \tag{5.29}$$

To better visualize the action of the tomography operations are doing, we look at the spinor of Eq. (5.30). It describes the relative phases of the state. It is written in the "binary" decomposition of the relative phases. The first relative phase ϕ_0 is neglected because it represents the global phase.

$$\begin{pmatrix} 1 \\ e^{i\phi_1} \\ e^{i\phi_2} \\ e^{i\phi_3} \\ e^{i\phi_4} \\ e^{i\phi_5} \\ e^{i\phi_6} \\ ... \end{pmatrix} = \begin{pmatrix} 1 \begin{pmatrix} 1 \begin{pmatrix} 1 \begin{pmatrix} 1 \\ e^{i\phi_1} \end{pmatrix} \\ e^{i\phi_2} \begin{pmatrix} 1 \\ e^{i\phi_3 - i\phi_2} \end{pmatrix} \end{pmatrix} \\ e^{i\phi_4} \begin{pmatrix} 1 \begin{pmatrix} 1 \\ e^{i\phi_5 - i\phi_4} \end{pmatrix} \\ e^{i\phi_6 - i\phi_4} \begin{pmatrix} 1 \\ e^{i\phi_7 - i\phi_6} \end{pmatrix} \end{pmatrix} \end{pmatrix} \\ ... \end{pmatrix} \tag{5.30}$$

At step $i = 0, ..., n-1$, the two rotations are implemented to the qubit i. From these probabilities the relative phase inside the binary decomposition i can be evaluated. Specifically, the rotations to qubit 0 gives ϕ_1, $\phi_3 - \phi_2$, ... The rotations to 1 qubit linked ϕ_2, $\phi_6 - \phi_4$, ... The third would give us ϕ_4, $\phi_8 - \phi_4$. Iterating for the N qubits we would have all the equations for the relative phases. It is important to mark that this procedure moves the exponentially increasing cost from the number of quantum circuits needed (for a fixed number of shots) in standard tomography to the number of required measurements for a fixed number of quantum circuits. Therefore, we should highlight again that the state tomography process is not scalable for a quantum simulation with a huge number of qubits.

5.4.3 Reinitializing operator

With the presented tomography, the state in a quantum processor can be identified. After doing that, evaluating for instance the state of the previous time step, we must reinitialize the state to continue our study of the real-time evolution. This subsection will discuss how we can build the reinitializing operator.
Again we must highlight that the most common default initial state of QPUs is the $|0\rangle$ state (the ground state of the quantum processors). It is for IBM QPUs [7] and AQT QPU [8]. Therefore, $|0\rangle$ will be considered as the initial state for our reinitializing operator.
We suppose the following pure state with N states has been identified:

$$|\psi\rangle = \begin{pmatrix} P_0 \\ P_1\, e^{i\,\phi_1} \\ ... \\ P_n\, e^{i\,\phi_N} \end{pmatrix} . \tag{5.31}$$

First of all to build the reinitializing gate that moves $|0\rangle$ to this state, we start by moving the default state $|0\rangle$ to one with the right probabilities distribution. To do that we observe that the $R_y(\theta)$ gate moves $|0\rangle$ to

$$|\psi_1\rangle \;=\; \cos\!\left(\frac{\theta}{2}\right)\,|0\rangle + \sin\!\left(\frac{\theta}{2}\right)\,|1\rangle\ . \tag{5.32}$$

Choosing θ correctly, $|\psi_1\rangle$ can be in the state with correct probability distribution. Specifically, for a single qubit $\theta \;=\; 2\,\arcsin\!\big(\sqrt{P_1}\big)$.
Suppose to work in a generic Hilbert space of dimension N, we can implement the rotation in Eq. (5.33) with $\theta_1 \;=\; 2\,\arcsin\!\big(\sqrt{P_1}\big)$.

$$R_1 \;=\; \begin{pmatrix} \cos\big(\frac{\theta_1}{2}\big) & -\sin\big(\frac{\theta_1}{2}\big) & 0 & 0 & ... & 0 \\ \sin\big(\frac{\theta_1}{2}\big) & \cos\big(\frac{\theta_1}{2}\big) & 0 & 0 & ... & 0 \\ 0 & 0 & 1 & 0 & ... & 0 \\ 0 & 0 & 0 & 1 & ... & 0 \\ ... & ... & ... & ... & ... & \\ 0 & 0 & 0 & 0 & ... & 1 \end{pmatrix} \tag{5.33}$$

The action of this rotation moves the state $|0\rangle$ to a new one with the correct probability for the state $|1\rangle$. The obtained state would be described by $|\psi_1\rangle = (\cos\big(\frac{\theta_1}{2}\big),\, \sin\big(\frac{\theta_1}{2}\big),\, 0,\, ..)^T$.
To obtain the correct probability distribution for the state $|2\rangle$, we can apply the

following rotation matrix,

$$R_2 = \begin{pmatrix} \cos\left(\frac{\theta_2}{2}\right) & 0 & -\sin\left(\frac{\theta_2}{2}\right) & 0 & ... & 0 \\ 0 & 1 & 0 & 0... & 0 & \\ \sin\left(\frac{\theta_2}{2}\right) & 0 & \cos\left(\frac{\theta_2}{2}\right) & 0 & ... & 0 \\ 0 & 0 & 0 & 1 & ... & 0 \\ ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & 1 \end{pmatrix} . \tag{5.34}$$

This evolves $|\psi_1\rangle$ to $|\psi_2\rangle = (\cos\left(\frac{\theta_1}{2}\right)\cos\left(\frac{\theta_2}{2}\right), \sin\left(\frac{\theta_1}{2}\right), \cos\left(\frac{\theta_1}{2}\right)\sin\left(\frac{\theta_2}{2}\right), 0, ...)^T$. To get the probability for the second state equal to P_2, we should choose $\theta_2 = 2\ \text{arcsin}\left(\frac{\sqrt{P_2}}{\cos\left(\frac{\theta_1}{2}\right)}\right)$.

Iterating this algorithm for the state k, we should y-rotate the $|k-1\rangle$ and $|k\rangle$ state with an angle of

$$\theta_k = 2\ \text{arcsin}\left(\frac{\sqrt{P_k}}{\prod_i^{k-1}\cos\left(\frac{\theta_i}{2}\right)}\right) \tag{5.35}$$

The general rotation matrix that moves the state $|0\rangle$ to the state $(\sqrt{P_0}, \sqrt{P_1}, ..., \sqrt{P_N})^T$ is obtained from

$$R_{TOT} = R_N\, R_{N-1}\, R_{N-2} ...\, R_2\, R_1\, R_0\,. \tag{5.36}$$

The last step to reach the state of Eq. (5.31) is to implement the relative phases. To do that we can apply after R_{TOT}, the following phase gate:

$$Ph = \begin{pmatrix} 1 & 0 & 0 & ... & 0 \\ 0 & e^{i\,\phi_1} & 0 & ... & 0 \\ 0 & 0 & e^{i\,\phi_2} & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & e^{i\,\phi_N} \end{pmatrix} . \tag{5.37}$$

The action of this gate implements the right phases to the state $R_{TOT}\ |0\rangle$. Finally, the final reinitializing operator is given by

$$U_{reinit} = Ph\, R_{TOT}\,, \tag{5.38}$$

where R_{TOT} and Ph is given by eqs. ((5.36)) and ((5.37)) respectively.

5.5 Coprocessing scheme using discrete gate sets implementing the reinitialization procedure

In this section, the presented state tomography procedure and the reinitializing operator will be used to simulate the same nuclear system of Sec. 5.3.

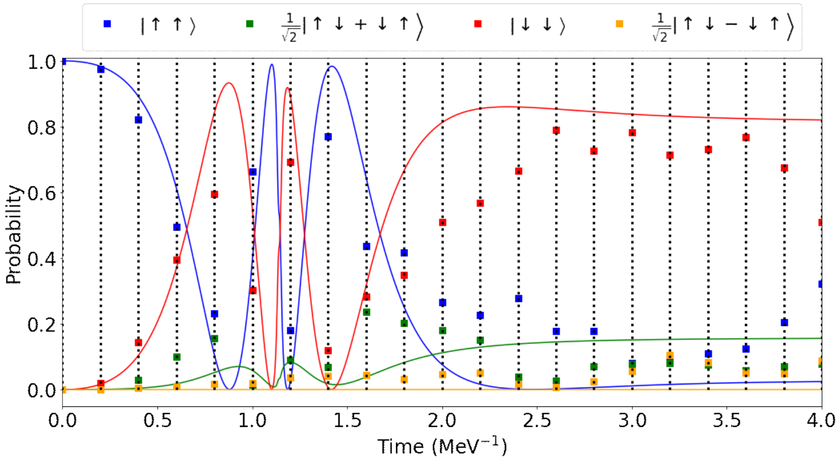


Figure 5.12: Coprocessing results with $T_{reini} = 1$. The dashed lines indicate the steps when there is a reinitializing process.

The same algorithm used at the end of Sec. 5.3 was employed. Instead, the obtained experimental data will be used to reinitialize the state.
Different simulations were performed using the quantum processor of AQT (see Ref. [8]) using different time steps where we reinitialize the state. This time step index is represented in this work with the variable T_{reini}. The results of simulations are shown in Figs. 5.12, 5.13 and 5.14 respectively with $T_{reini} = 1$, 3, 7. Squares represent the obtained data, lines the analytic spin evolution and the vertical dashed lines when we applied the reinitializing gate.

Observing the residuals between the obtained values and the analytical evolution, we can notice that the tomography results with the smaller T_{reini} get good accuracy. For example, this is very evident looking at the asymptotic limit, where for $T_{reini} = 1$ we have a slight discrepancy; instead, for $T_{reini} = 7$, a significant error is obtained. These results are obtained from the bare simulation without employing any error mitigation procedure. Using error-correcting algorithms, one might expect an improvement of the results.
Furthermore, Fig. 5.15 shows the fidelities between the state computed from the analytic evolution and experimental one obtained from our state tomography for different T_{reini} values.
We obtained another proof that the results with small T_{reini} get good compatibility with the analytic evolution. Nevertheless, the cost of performing this kind of tomography is exponentially huge. One should implement five different quantum circuits for studying a single time step for two qubits. As we said earlier, studying a complex system with n qubits, one should implement $2\,n+1$ quantum circuits. However, one must implement an exponential number of measures causing the scalability problem.
The last figure of this chapter Fig. 5.16 shows how the experimental reinitial-

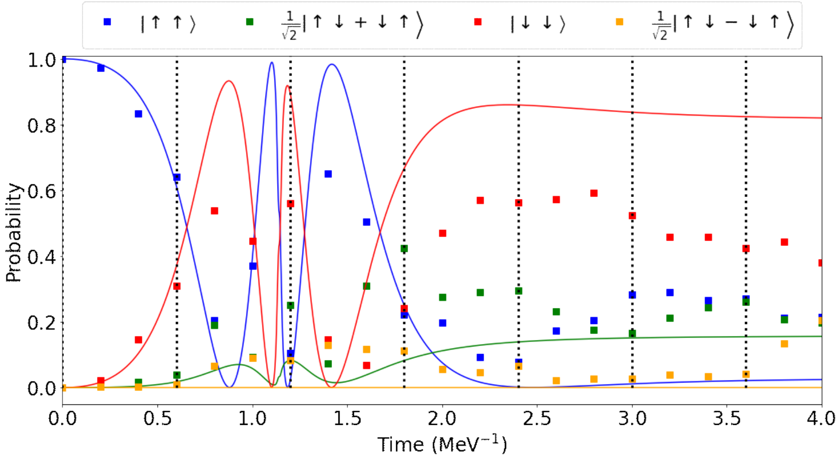


Figure 5.13: Coprocessing results with $T_{reini} = 3$. The dashed lines indicate the steps when there is a reinitializing process.

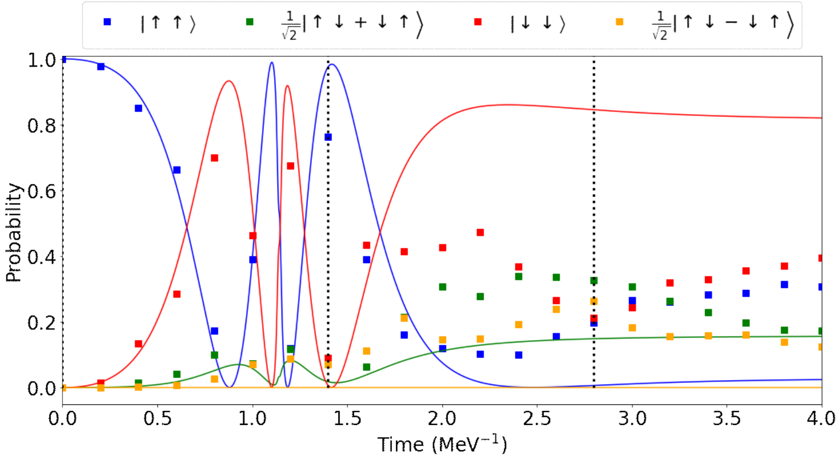


Figure 5.14: Coprocessing results with $T_{reini} = 7$. The dashed lines indicate the steps when there is a reinitializing process.

izing procedure increases the accuracy of the simulated data compared to one obtained applying the full sequence of propagators, plotting the residuals between the obtained data and the analytical evolution. In Fig. 5.16, squares and circles indicate the results with the reinitializing procedure and those applying the full sequence, respectively.

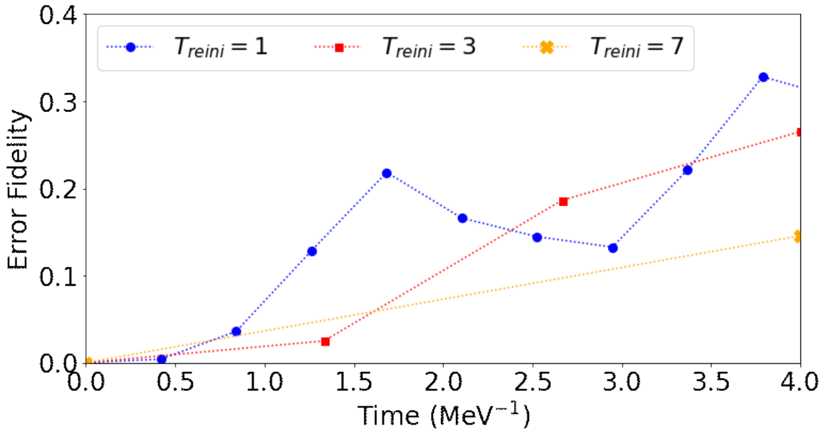


Figure 5.15: Error fidelity between the obtained results using the partial tomography and the analytic states for the different T_{reini}

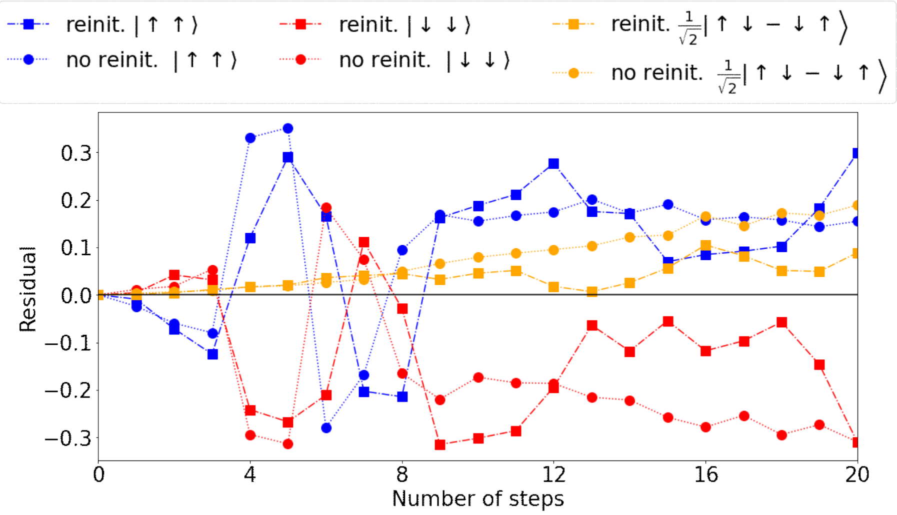


Figure 5.16: Residuals between results and analytical evolution. The squares(circles) represent the data with(without) applying the reinitializing procedure with the exposed state tomography process. The black solid line shows the 0 value.

Chapter 6

Application of Similarity Renormalization Group to the compilation of a quantum gate in digital gates

This chapter will focus on how to efficiently translate a generic N qubit gate into a quantum circuit with digital gates. The generic quantum operation to be compiled will be the real time propagator of a deuteron system, the nucleus formed by a neutron and a proton.
We will start from the deuteron Hamiltonian. Then, we will show why it is difficult to compile a generic quantum operator with the known algorithms efficiently. In the end, a possible strategy to simplify the compilation of this kind of operators by applying the Similarity Renormalization Group (SRG) theory will be shown.

6.1 Full quantum simulation of a deuteron system

We are interested in simulating the real time evolution of a deuteron system. Indeed, we want to compute from the real time probabilities its ground energy (following the method of Ch. 4).
To simulate this quantum system in a quantum processor, we may use the co-processing method of Ch. 5. But, using a classical method the fermionic nature

of the two constituents may be lost. Therefore, we may have a problem with its energies.

The solution is to employ a full quantum simulation. We will expand the Hamiltonian in a basis set that includes all spatial, spin and isospin degrees of freedom. Each element of the basis set will be mapped into the levels of the quantum processor. We will encode the real time propagator in a quantum circuit. This allows us to simulate the real time evolution of the deuteron through a quantum computer.

We start from the nuclear Hamiltonian derived from a LO chiral EFT (for details, see App. C.2). Being interested in the ground energy of deuteron, we know that it has isospin $T = 0$ and angular momentum $L = 0, 2$. We also know that this kind of Hamiltonian has isospin symmetry. It is also invariant under the total angular momentum $J = L + S$ (S indicates the total spin) and J_z. Hence, we have degenerate states with different J and J_z, and they cannot interact with other states with different J and J_z. These symmetries are used to simplify the simulation. Therefore, we will restrict the calculation to states with isospin $T = 0$ and $J = 1$ $J_z = 0$ with $L = 0$ and $L = 2$. Calculations with $J_z = 1, -1$ will give the same result.

The general basis set is given by:

$$|\psi\rangle \;=\; R(r)\; |J,\, J_z,\, L\rangle \;, \tag{6.1}$$

where the function $|J,\, J_z,\, L\rangle$ connects the spherical harmonic $Y_{l,m}$ and the spin states with same J, J_z and L. $R(r)$ indicates the spatial distribution. We expand the spatial basis set $R(r)$ with the Laguerre polynomials.

Expanding the Hamiltonian, we should choose when we truncate the expansion with Laguerre polynomials. This number in this thesis is indicated with N_{max}. In Fig. 6.1 the obtained ground energy of the deuteron Hamiltonian is shown as function of N_{max}.

For simulating the deuteron dynamics in a quantum computer, we choose as truncation for the basis $N_{max} = 3$ since it is sufficient to describe the ground energy with reasonable accuracy. Therefore, we need at least 4 qubits (or a transmon qudit with $N = 16$ levels) to simulate this system through a quantum processor.

6.2 Translating a n qubit gate in elementary gates

We are using a quantum processor based on the elementary gate approach. Given a generic n qubit quantum gate U, how may we compile it in elementary gates (for instance, in terms of R_x, R_z and $CNOT$)? In our case, U is the real time evolution of the deuteron using 4 qubits.

Ch. 2 presented the quantum universality and the Solovay–Kitaev theorem. The former declares that it is always possible to translate a n qubit in terms

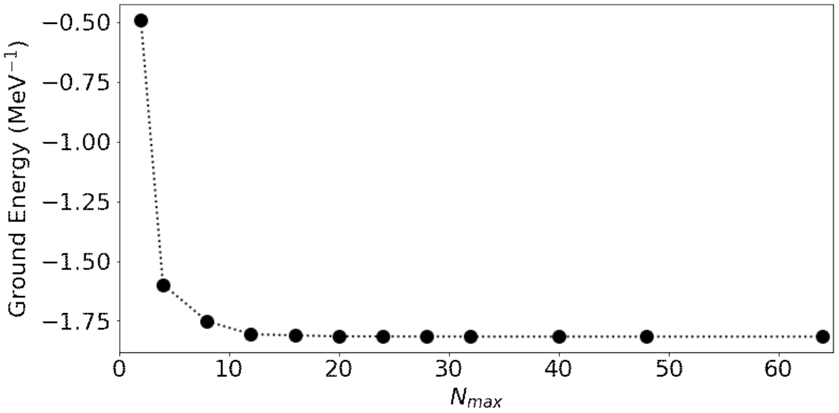


Figure 6.1: Deuteron ground energy as a function of N_{max}

of elementary gates. The latter demonstrates that we can efficiently approximate a quantum operator with a quantum circuit with a gate depth[1] of order $O(log(\epsilon^{-1})^c)$ with $c = 3.97$ and ϵ the error of the approximation. However, these two statements do not tell us how to do that numerically.

Many algorithms translate a generic operator in a quantum circuit in the literature. The most common are the sine-cosine decomposition (CSD) [83] and a method that uses the so-called Gray code [84, 85].

Applying these analytical compiling methods, the obtained quantum circuit is made of a huge number of elementary gates. For instance, a general 3 qubit gate using the Grey code requires 196 elementary gates and CSD compiling algorithm 131 elementary gates. The found quantum circuit can not be implemented in a real machine because of its immense depth. The machine's noise would kill the quantum simulation employing this number of gates. Furthermore, we also have to consider the contribution of gate infidelity.

Our goal is to evaluate the ground energy of deuteron. According to Ref. [9] (see Ch. 4), simulating its real time evolution and computing the Fourier Transform of the obtained real time probability, we will obtain its spectra. The problem is how we compile the real time evolution operator of a deuteron system mapped in 4 qubits in a quantum circuit based on digital gates avoiding the immense depth of quantum circuit.

We investigate different alternatives to compile a quantum gate efficiently. The most exciting result is employing the Similarity Renormalization Group (SRG) method [86, 87, 88]. The next section will present it.

[1] The depth of a quantum circuit is the quantum version of the depth of a classical circuit[73]. The depth of a circuit is defined as an integer number that describes the number of gates of the longest path from the input to the output qubits, moving forward in time along the wires. We can simply define the depth as the integer number that minimizes the times for running the entire quantum circuit counting each gate 1 in unit time.

6.3 Compiling Similarity Renormalization Group method

6.3.1 Theory of Similarity Renormalization Group

We start presenting the basic aspects of the general theory of Similarity Renormalization Group (SRG).
Let H_0 be a very complicated Hamiltonian. The SRG algorithm aims to decompose H_0 into simpler Hamiltonians with the same spectra. More specifically, through the SRG method, we transform H_0 in a new Hamiltonian H_n with a block structure.
To conserve the spectrum, we must transform the Hamiltonian with unitary operators:

$$H(s) = U^\dagger H_0 U \,, \tag{6.2}$$

This procedure is iterated and s identifies which Hamiltonian we have.
In the SRG method, the U operator is obtained from

$$U = \exp\left\{ - \int ds \, \eta(s) \right\} . \tag{6.3}$$

The operator $\eta(s)$ in eq. (6.3) is named generator of SRG evolution. How do we choose η correctly to simplify the calculations? First of all, choosing that the operator η is anti-hermitian, $\eta^\dagger = -\eta$, we ensure that the operator U is unitary. In standard SRG applications, one defines $\eta = [T, H(s)]$, a commutator of another generator T. This other commutator ensures that the operator η is anti-hermitian. If T is a block matrix, the magic power of SRG transforms the Hamiltonian H to a block matrix with the same structure of T. In nuclear physics, one generally selects T equal to the kinetic energy (working in momentum space it is diagonal).
Differentiate eq. (6.2), one obtains the SRG flow equation.

$$\frac{d}{ds} H(s) = [\eta(s), H(s)] = \left[[T, H(s)] \,, H(s) \right] \,. \tag{6.4}$$

In practical calculations, one solves the SRG flow equation to obtain a block Hamiltonian.

6.3.2 Quantum compilation with SRG for real time evolution

Let H be the (deuteron) Hamiltonian of the quantum system in n qubit, we want simulate its evolution with $U(t) = e^{-i\,t\,H}$. Computing the Fourier transform, the spectrum will be computed. We have problems compiling U in digital gates for number qubits greater than 2. Having 2 qubit gates, we have efficient methods to translate them, for example, using the *decompose* function of Qiskit [81].
We have seen that, using SRG, a complex Hamiltonian will be simplified in one

with block diagonal structure. The idea is to use the SRG method to compile complex unitary operators in digital gates using ancilla qubits. Specifically, we start from the Hamiltonian $H_0 \in 2^n \times 2^n$. Then, the SRG is used with generator $T = \sigma_z^n$ (the z-Pauli matrix of qubit n). The obtained Hamiltonian would be block $2^{n-1} \times 2^{n-1}$ matrices. Using ancilla qubits, the quantum circuits can be split into smallest ones.

Now, the fundamental steps will be presented.

As we said, we start from some Hamiltonian H_0, a hard Hamiltonian mapped to n qubit (as an example, the deuteron Hamiltonian mapped on 4 qubits). We evolve the Hamiltonian through the SRG flow equation with $T = \sigma_z^n$. The obtained final Hamiltonian would be given by:

$$H_{SRG} = \begin{pmatrix} H_1^1 & 0 \\ \\ 0 & H_2^1 \end{pmatrix}, \tag{6.5}$$

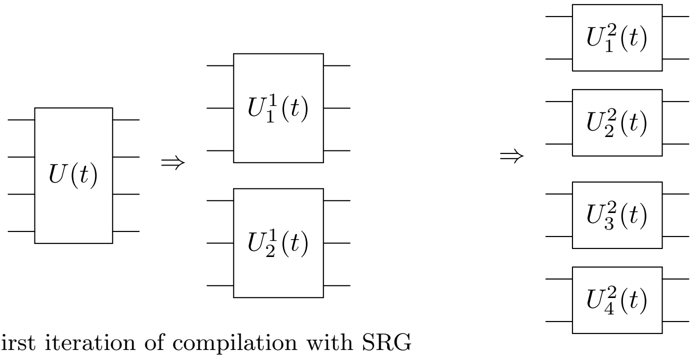
where H_1^1 and H_2^1 are Hamiltonian mapped on $n-1$ qubit. The lower and upper indices indicate the number of resulting Hamiltonian and the step of iteration with SRG respectively. We can notice that if we compute the real time evolution operator it has the same block structure of the Hamiltonian, specifically:

$$U_{SRG}(t) = \begin{pmatrix} U_1^1(t) = e^{-i\,t\,H_1^1} & 0 \\ \\ 0 & U_2^1(t) = e^{-i\,t\,H_2^1} \end{pmatrix}. \tag{6.6}$$

The structure of $U_{SRG}(t)$ would allow add $n-2$ ancilla qubits for spliting the real time evolution into two distinct quantum circuits, one for $U_1^1(t)$ and the other for $U_2^1(t)$.

For example, in our case, we start from a Hamiltonian with 4 qubit, and through the SRG method, we arrive at two Hamiltonians (quantum circuits) of 3 qubits. We know that a 2 qubit gate can efficiently be translated with a maximum of 3 $CNOT$ gates. Therefore, the SRG compiling method would be iterated until we arrive to 2 qubit gates.

Fig. 6.2 shows the scheme for compiling a 4-qubit real evolution operator. In conclusion, through the SRG compilation algorithm, we can have a simpler quantum circuit. However, the price is to have many more qubits than the ones initially required. Luckily, some quantum circuits can be removed. For the SRG properties, the obtained quantum circuits describes the same spectra of Hamiltonian. For instance, if we are interested in computing the properties of the deuteron using the potential derived by the LO chiral interaction, we know that there is a limit in the validity of the theory for energies >400 MeV). Therefore, one has to neglect in general the quantum circuits with energies greater than some cutoff E_c.



(a) First iteration of compilation with SRG

(b) Second iteration of compilation with SRG

Figure 6.2: An example of employing SRG to simplify the compilation of 4 qubit gates in 4 quantum circuits with 2 qubit gates. For SRG theory, the obtained spectra of the real time evolution are equal to the initial one. Panel (a) represents the first iteration (from four qubit gate to two three qubit gates), panel (b) the second one (from three qubit gate to two two qubit gates)

6.4 Deuteron result through the SRG compilation

We want to compute the ground energy of a deuteron system through real time evolution. We start from the Hamiltonian described in Sec. 6.1 with $N_{max} = 8$. To simulate this Hamiltonian, at least 4 qubits are requested. 8 states describes the spatial component of the deuteron Hamiltonian expanded in the $n = 0, 1, .., 7$ Laguerre polynomials with $|J = 1, J_z = 0, L = 0\rangle$; the others describe the same with $|J = 1, J_z = 0, L = 2\rangle$.

We start from the 4 qubit gate representing the real time evolution with a generic time step t. The SRG compiling method is iterated two times. The scheme of the obtained quantum circuits is shown in Fig. 6.2. Fig. 6.3 presents the actual quantum circuits. The circuits are sorted according to the relative eigenvalues in decreasing order. Being interested in evaluating the ground energy of deuteron, we can just implement the quantum circuit with the lowest eigenvalue. Tab. 6.1 presents the initial energy of the Hamiltonian and the final obtained energies of four quantum circuits. We cannot observe any difference between the original and obtained spectra.

The final quantum circuits were implemented in the IBM *iqmq_quinto* processor with a number of shots $N_{shot} = 4000$ where we reduced fully optimized the circuits.

Fig. 6.4a shows the obtained results employing this quantum circuit. Lines

and circles indicate the analytical and obtained results, respectively. Fig. 6.4b presents the Fourier Transform of the real time probabilities of Fig. 6.4a. Again lines and circles indicate the analytical and obtained results, respectively. The vertical dashed line represents the value given by $\frac{E_1-E_0}{2\pi}$.
The experimental value of $\frac{E_1-E_0}{2\pi}$ is 0.52(4). Iterating the same algorithm for H^3, the experimental data for $\frac{E_1^3-E_0^3}{2\pi}$ is 1.5(4). The obtained value of ground energy is -1.96 ± 0.09 MeV. This result is compatible with 3 sigmas with the lowest eigenvalue of the Hamiltonian, -1.75 MeV.

Original Hamiltonian	[-1.75, 1.37, 2.92, 5.52, 8.20, 13.44, 17.95, 29.63, 37.50, 64.80, 82.03, 173.38, 217.16, 718.30, 818.02, 8650.24] MeV
From quantum circuits	[217.16, 718.30, 818.02, 8650.24] MeV [37.50, 64.80, 82.03, 173.38]MeV [8.15, 13.44, 17.95, 29.63] MeV [-1.75, 1.37, 2.92, 5.57] MeV

Table 6.1: Energies of original deuteron Hamiltonian and ones obtained from the 4 quantum circuits.

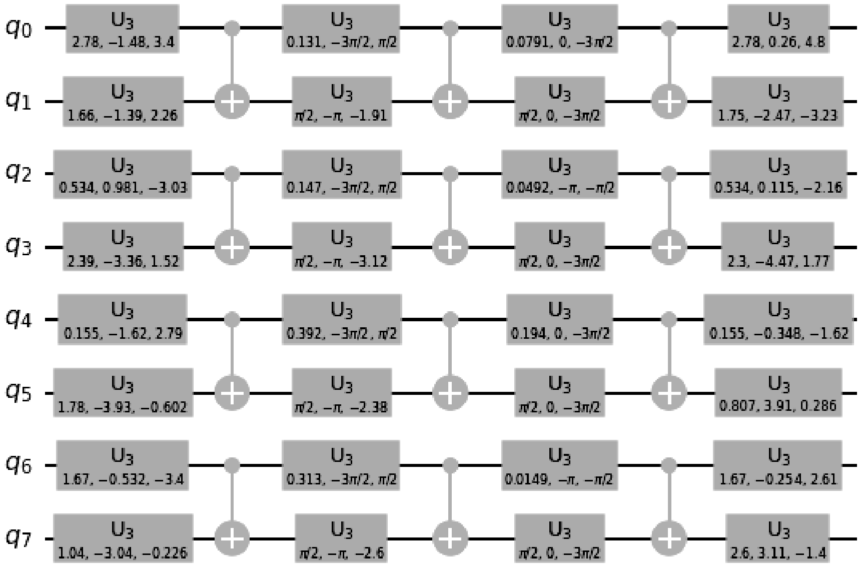
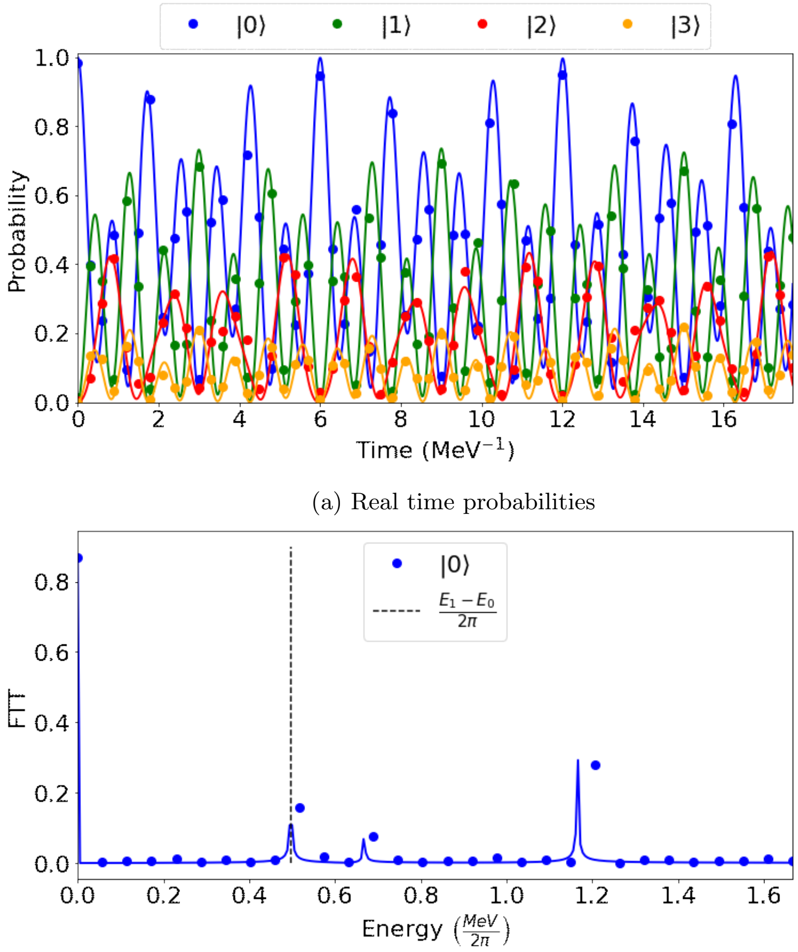


Figure 6.3: Actual obtained quantum circuits of the real time evolution of deuteron system with $t = 0.30$ MeV^{-1}. The corresponded energies of each quantum circuits are shown in Tab. 6.1



(a) Real time probabilities

(b) Fourier transform of the probability for $|0\rangle$ state. The dashed line indicates the analytical difference between the energy of first excited and the ground energy

Figure 6.4: Theoretical results of real time probabilities and their Fourier Transform after applying the SRG compiling method to simplify the calculation. Lines and circles indicate the analytical and obtained results respectively.

Chapter 7

Quantum Imaginary Time Propagation

In the previous chapters, we focused on studying the real time evolution of a quantum system. Instead, this chapter will focus on evaluating or preparing the ground state(GS) of a quantum system in a quantum processor.
In standard computation, one possibility for computing ground state properties of a given system is to operate a Wick rotation on the real time evolution operator. The resulting propagator is not unitary, implementing a dissipation mechanism. Evolution in imaginary time is a well-known technique for finding the ground state of quantum many-body systems [22].
A possible way to introduce dissipation in a quantum algorithm through an imaginary time method will be presented. It is implemented by expanding the Hilbert space of the system under investigation by introducing ancillary qubits. The projection is obtained by applying a series of unitary transformations to evolve the components of the initial state along excited states of the Hamiltonian H to the ancillary space. A measurement of the ancillary qubits should then remove such components.
We will begin reviewing the Imaginary Time Propagation (ITP) in classical computation. Then, its quantum version will be presented. We will speak about some improvements and results. In the end, we will discuss how we can use Trotter decomposition. This chapter is based on Ref. [10].

7.1 Standard Imaginary Time Propagation

The time-dependent Schrödinger equation reads:

$$i\hbar\frac{d}{dt}\left|\psi(t)\right\rangle = H\left|\psi(t)\right\rangle \ , \tag{7.1}$$

where $|\psi(t)\rangle$ is a state and H is the Hamiltonian of a system.
If we perform a Wick rotation $t \to -i\,\hbar\,\tau$, the Schrödinger equation becomes:

$$-\frac{\partial}{\partial\tau}\,|\psi(\tau)\rangle = H\psi(\tau)\,. \tag{7.2}$$

If H does not depend on time, we can compute the state $|\psi(\tau)\rangle$ by:

$$|\psi(\tau)\rangle = e^{-H\tau}\,|\psi(\tau_0)\rangle\ , \tag{7.3}$$

where $e^{-H\tau}$ is called imaginary-time propagator (ITP) and $|\psi(\tau_0)\rangle$ indicates the initial state.
Writing the initial state $|\psi(\tau_0)\rangle$ as a linear composition of the eigenstates of the Hamiltonian, $H\,|\phi_n\rangle = E_n\,|\phi_n\rangle$,

$$\psi(\tau_0) = \sum_n c_n\ |\phi_n\rangle\ , \tag{7.4}$$

with $c_n\ =\ \langle\phi_n|\psi(t_0)\rangle$, the imaginary time propagation becomes

$$|\psi(t)\rangle = \sum_{n=0}^{\infty} c_n\, e^{-H\tau}\,|\phi_n\rangle = \sum_{n=0}^{\infty} c_n e^{-E_n\tau}\,|\phi_n\rangle\ , \tag{7.5}$$

If we evolve Eq. (7.5) in the limit $\tau \to \infty$ the term $c_0\, e^{-E_0\tau}$ becomes dominant because the other terms, $c_n\, e^{-E_n\tau}$, are exponentially suppressed. Hence, in the limit of $\tau \to \infty$ the wave function tends to the ground state (GS) of H:

$$|\psi(\tau)\rangle \xrightarrow{\tau\to+\infty} c_0 e^{-\tau E_0}\ |\phi_0\rangle\ , \tag{7.6}$$

where $|\phi_0\rangle$ indicates the ground state.
Therefore, if we propagate along imaginary time any arbitrary state, which must be not orthogonal to the ground state, the obtained state converges to the mathematical ground state of H. Notice that the physical ground state needs to account for the symmetry of particles and might not coincide with the ground state of H in the mathematical sense. For instance, the mathematical ground state of Fermions is not the actual physical GS because one should consider the Pauli principle that requires the state to be antisymmetric.
We must notice that the imaginary-time propagator is hermitian but not unitary, and this causes the normalization of the projected state to be not guaranteed. Generally, the state norm drops to 0 for $\tau\ \to \infty$. It is possible to preserve the normalization at the projected state redefining the propagator as:

$$|\psi(\tau)\rangle = e^{-\tau(H-E_T))}\,|\psi(\tau_0)\rangle\ , \tag{7.7}$$

where E_T is called trial energy.
E_T is generally chosen equal to the ground energy E_0. Indeed, it is easy to prove that if $E_T = E_0$ the final state has a lower bound in the normalization given by $|c_0|^2$. In standard applications, it is not strictly necessary to know in advance the exact ground state E_0, since it is sufficient to evaluate an upper bound $E_T = E_0 + \epsilon$ (for instance, from a variational method). If the error ϵ is small enough, the ITP method is stable and convergent.

7.2 Quantum Imaginary Time algorithm (QITP)

In the previous section, we have seen that the standard ITP algorithm is based on the application of the ITP propagator $e^{-(H-E_T)\tau}$. If one wants to translate this method on a quantum computer, one should fight with the non-unitarity of the ITP propagator.

Indeed, quantum operations must be described by unitary operators, but the ITP propagator is not unitary:

$$\left(e^{-\tau\,(H-E_T)}\right)^\dagger e^{-\tau(H-E_T)} = e^{-2\,\tau\,(H-E_T)} \neq \mathbb{1}\,. \tag{7.8}$$

A possible solution is to extend the Hilbert space of the simulation by adding ancilla qubits, external qubits [89, 90]. Working with ancilla qubits is at the basis of several algorithms [91, 92, 93, 94].

We start mapping the states of the physical system that we are interested in computing the ground state in some qubits. For instance, all the $N = 2^n$ states are mapped in n qubits. We start from an initial wavefunction $|\psi_s(\tau_0)\rangle$ describing a specific initial state of the physical state. Then, an ancilla is added to the simulation extending the Hilbert space from 2^n to 2^{n+1}. This external qubit is prepared in the state $|0\rangle$. Hence, the total wavefunction into the quantum processor will be given by:

$$|\Psi_{\text{init}}\rangle = |0\rangle \otimes |\psi_s\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes |\psi_s\rangle\ , \tag{7.9}$$

where in our notation for the tensor product the left state represents the ancilla and the right one the physical state.

Now, we must look for a unitary operator in this extended Hilbert space that implements a dissipation mechanism with the ITP method. Eq. (7.10) shows a good candidate for our aims:

$$\begin{aligned} \hat{U}(\tau) &= \begin{pmatrix} \frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} & \frac{1}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} & -\frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} \end{pmatrix} \\ &= \begin{pmatrix} \hat{Q}_{\text{ITP}}(\tau) & \frac{1}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}} & -\hat{Q}_{\text{ITP}}(\tau) \end{pmatrix}\,. \end{aligned} \tag{7.10}$$

In this equation $Q_{ITP}(\tau) = \frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2(\hat{H}-E_T)\tau}}}$ and $\mathbb{1}$ are operators acting in the Hilbert space of the physical systems and $\mathbb{1}$ indicates the identity matrix. One can observe that the operators in the main diagonal are variations of the standard ITP propagator. Specifically, they are given by multiplying the standard ITP operator by a normalization operator. The normalization operator is needed to have a final unitary operator. Indeed, computing $U^\dagger U$ one can find:

$$
\begin{aligned}
U^\dagger U &= \left[\begin{pmatrix} \frac{e^{-\tau\,(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & -\frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \end{pmatrix}\right]^\dagger \begin{pmatrix} \frac{e^{-\tau\,(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & -\frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \end{pmatrix} \\
&= \begin{pmatrix} \frac{e^{-\tau\,(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & -\frac{e^{\tau(-(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \end{pmatrix}^2 \\
&= \begin{pmatrix} \left(\frac{e^{-\tau\,(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\right)^2 + \left(\frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\right)^2 & \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} - \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} - \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} & \left(-\frac{e^{\tau(-(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\right)^2 + \left(\frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\right)^2 \end{pmatrix} \\
&= \begin{pmatrix} \frac{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}} & 0 & \\ 0 & \frac{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}} & 0 \end{pmatrix} = \begin{pmatrix} \mathbb{1} & 0 \\ 0 & \mathbb{1} \end{pmatrix} = \mathbb{1} \otimes \mathbb{1}
\end{aligned}
\tag{7.11}
$$

From the first line to the second one, we used that the identity and ITP operator are hermitian.
One can also rewrite the QITP operator $U(\tau)$ as:

$$
U(\tau) = \sigma_z \otimes Q_{ITP}(\tau) + \sigma_x \otimes \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\,, \tag{7.12}
$$

where $\hat{\sigma}_z$ and $\hat{\sigma}_x$ are the Pauli Z and X operators acting on the ancillary qubit. We can notice from this equation that $U(\tau)$ entangles the ancilla and physical states.
The application of $U(\tau)$ to the initial state of Eq. (7.9) yields:

$$
\begin{aligned}
|\Psi(\tau)\rangle &= |0\rangle \otimes \hat{Q}_{\mathrm{ITP}}(\tau)\,|\psi_s\rangle + |1\rangle \otimes \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\,|\psi_s\rangle \\
&= \begin{pmatrix} Q_{\mathrm{ITP}}(\tau)\,|\psi_s\rangle \\ \frac{1}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\,|\psi_s\rangle \end{pmatrix}.
\end{aligned}
\tag{7.13}
$$

We can notice that the physical state when the ancilla qubit is in $|0\rangle$ is given by the application of ITP times the normalization to the initial state. In Sec. 7.2.3 we will prove that the final state is always closer to ground state than starting one.
Therefore, performing a (partial) measurement along the ancilla qubit and obtaining $|0\rangle$ (with some probability), the state becomes:

$$
|\Psi_{\mathrm{fin}}\rangle = C\,|0\rangle \otimes \hat{Q}_{\mathrm{ITP}}(\tau)\,|\psi_s\rangle = C\,|0\rangle \otimes \frac{e^{-\tau(H-E_T)}}{\sqrt{1+e^{-2\tau(H-E_T)}}}\,|\psi_s\rangle\,, \tag{7.14}
$$

where the constant C is introduced in order to normalize the state. This final state is analogous of applying the standard imaginary time propagator. However, here in the QITP, the parameter C is really fundamental because it represents the success probability of this algorithm (called also P_s).
The following subsections will deeply investigate some essential properties of the QITP method. We will study the asymptotic limits of the QITP algorithm for $\tau \to 0$. After that, the success probability and fidelity with the GS will be discussed. The subsection of the fidelity will also demonstrate that the final state of the QITP algorithm is always closer to GS than initial state. In the end, a numerical test that proves the results of fidelity and success probability will be shown.

7.2.1 Asymptotic behavior of QITP for small time step

This subsection will discuss the asymptotic limit of QITP algorithm when $\tau \to 0$. It is very crucial when one wants to study many-body systems because generally their Hamiltonian is a sum of two- or three body interactions. Hence, we should prove that the Trotter decomposition can be used to simplify the QITP algorithm.
We look closer at the Q_{ITP} operator, and the two exponentials are expanded for $\tau \to 0$. We obtain

$$Q_{ITP} = \frac{\sum_n \frac{(-\tau(H-E_T))^n}{n!}}{\sqrt{1+\sum_n \frac{(-2\tau(H-E_T))^n}{n!}}} \,. \tag{7.15}$$

and we get at first order in τ

$$\begin{aligned} Q_{ITP} = \frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}} &\simeq \frac{1-\tau(H-E_T)}{\sqrt{1+1-2\tau(H-E_T)}} + O(\tau^2) \\ &= (1-\tau(H-E_T))\frac{1}{\sqrt{2}}\left(1+\frac{1}{2}\tau(H-E_T)\right) + O(\tau^2) \\ &= \frac{1}{\sqrt{2}}\left(1-\frac{1}{2}\tau(H-E_T)\right) + O(\tau^2)) \\ &= \frac{1}{\sqrt{2}}e^{-\frac{\tau}{2}(H-E_T)} + O(\tau^2)\,. \end{aligned} \tag{7.16}$$

We have applied $(1+x)^\alpha = \sum_n \begin{pmatrix} \alpha \\ n \end{pmatrix} x^n$ for $|x| < 1$ to pass from the first line to the second. one
Therefore, we have obtained $Q_{ITP} \xrightarrow{\tau\to 0} e^{-\frac{\tau}{2}(H-E_T)}$. It is equivalent to applying the standard ITP propagator but halving time step. The ancilla probability to

measure $|0\rangle$ is obtained from:

$$P_s = \left\| \frac{1}{\sqrt{2}} e^{-\tau(H-E_T)} |\psi_{ini}\rangle \right\|^2 \xrightarrow{\tau\to 0} \frac{1}{2} \|(1-\frac{1}{2}H\tau)\,|\psi_{ini}\rangle\,\|^2 \sim \frac{1}{2} + O(\tau)\,. \quad (7.17)$$

If P_s is plotted as a function of the time step for a particular value of E_T, the curve always starts from $\frac{1}{2}$.

Looking at the result of (7.16), we can also notice that at the first order, our quantum imaginary time step can be decomposed according to the Trotter formula. For example, this is the case when the Hamiltonian is a sum of two-body interactions, $H = \sum_n h_n$. Indeed, we can write at first order in τ

$$Q_{ITP} \sim \frac{1}{\sqrt{2}} e^{-\tau(H-E_T)} \sim \frac{1}{\sqrt{2}} e^{-\tau(\sum_n h_n - E_T)} \sim \frac{1}{\sqrt{2}} e^{-\tau E_T} \prod_n e^{-\tau h_n} + O(\tau^2)\,. \quad (7.18)$$

Sec. 7.7 will explore formally the Trotter decomposition of QITP.

7.2.2 Success probability

This subsection will analyze the success probability (P_s) of the QITP algorithm. We have seen that after implementing $U(\tau)$ and measuring the ancilla qubit in $|0\rangle$, the resulting physical state will be closer to the ground state. The success probability of the presented QITP is equal to the probability of measuring the ancillary qubit in $|0\rangle$ state.

The unnormalized state after the application of $U(\tau)$ when the ancilla qubit is found in $|0\rangle$ is given by:

$$\left|\Psi^0\right\rangle = \frac{e^{-\tau(H-E_T)}}{\sqrt{1+e^{-2\tau(H-E_T)}}} |\phi_s\rangle\,. \quad (7.19)$$

The success probability P_s is equal to:

$$P_s = \left\|\,\left|\Psi^0\right\rangle\right\|^2 = \left\langle \Psi^0 \middle| \Psi^0 \right\rangle\,. \quad (7.20)$$

We start decomposing $\left|\Psi^0\right\rangle$ with the eigenvectors of the Hamiltonian H, $H|\phi_n\rangle = E_n|\phi_n\rangle$, getting

$$\left|\Psi^0\right\rangle = \frac{e^{-\tau(E_0-E_T)}}{\sqrt{1+e^{-2\tau(E_0-E_T)}}} c_0 |\phi_0\rangle + \sum_{n\neq 0} c_n \frac{e^{-\tau(E_n-E_T)}}{\sqrt{1+e^{-2\tau(E_n-E_T)}}} \left|\phi_0^\perp\right\rangle\,. \quad (7.21)$$

where $c_0 = \langle\phi_0|\psi_s\rangle$ represents the initial overlap with the GS and the other $\{c_n\}$ are the overlaps with the other eigenstates of H.

The success probability P_s becomes:

$$P_s = \frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}} |c_0|^2 + \sum_{n\neq 0} |c_n|^2 \frac{e^{-2\,\tau(E_n-E_T)}}{1+e^{-2\tau(E_n-E_T)}}\,, \quad (7.22)$$

where we used the eigenstates of H are orthogonal. Therefore, the probability is obtained by the sum of the squares of absolute values of the single eigenstate contributions.

From Eq. (7.22) an upper bound for P_s can be obtained. Fixing c_0, the upper bound is given when we have a slow convergence to ground state. The slowest decay to the ground state is obtained when the state is a mixture of the ground state (we fixed c_0) and the first excited state with overlap $c_1 = \sqrt{1-|c_0|^2}$. Hence, the upper bound of P_s, P_s^u, is given by

$$P_s \leq P_s^u = \frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\,|c_0|^2 + \left(1-|c_0|^2\right)\frac{e^{-2\,\tau(E_1-E_T)}}{1+e^{-2\tau(E_1-E_T)}}\,. \tag{7.23}$$

Instead, to obtain the lower bound we may use that the spectral norm of the Hamiltonian $\|H\|_\infty$[1]. Hence, the success probability can be bounded from below with:

$$P_s \geq P_s^b = \frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\,|c_0|^2 + \left(1-|c_0|^2\right)\frac{e^{-2\,\tau(\|H\|_\infty-E_T)}}{1+e^{-2\tau(\|H\|_\infty-E_T)}}\,. \tag{7.24}$$

Therefore, we can conclude that our exact success probability is between:

$$\begin{aligned}\frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\,|c_0|^2 + \left(1-|c_0|^2\right)\frac{e^{-2\,\tau(\|H\|_\infty-E_T)}}{1+e^{-2\tau(\|H\|_\infty-E_T)}} \leq \; P_s \\ \leq \frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\,|c_0|^2 + \left(1-|c_0|^2\right)\frac{e^{-2\,\tau(E_1-E_T)}}{1+e^{-2\tau(E_1-E_T)}}\,.\end{aligned} \tag{7.25}$$

In the limit $\tau \to \infty$, we can notice that the trial energy E_T, a parameter of the QITP algorithm, becomes very crucial. Indeed, tuning E_T, the success probability can be increased or decreased. One finds that the fundamental threshold of the success probability is given for $E_T = E_0$.

In the limit $E_T < E_0$, the upper bound given by Eq. (7.24) drops to 0 because all the exponentials have a negative exponent. Therefore, tuning $E_T < E_0$, a large number of measurements must be performed to obtain the ancilla qubit in the $|0\rangle$ state. This can also be seen that in the limit of $\tau \to \infty$ the QITP operator converges to the X gate for ancilla. Therefore, the action of $U(\tau)$ is just to flip the states of the ancillary qubit.

Instead, for $E_T \geq E_0$, it is easy to show that the lower bound of the ancilla qubit at least goes to a constant value. Specifically, it is given by

$$P_{min} = \frac{e^{-2\,\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\,|c_0|^2\,. \tag{7.26}$$

This contribution does not drop to 0. In particular when $\tau \to \infty$ we obtain that $P_{min} = |c_0|^2$ for $E_T > E_0$ and $\frac{1}{2}\,|c_0|^2$ for $E_T = E_0$. An other important range for E_T is when $E_T > \|H\|_\infty$. In this interval, the lower bound of the ancilla probability goes to 1. However, in the next subsection, we will see that in this range we have a very slow convergence to the GS.

[1] $\|H\|_\infty = max_i\,|E_i|$ with E_i energy of H

7.2.3 Fidelity between the projected state and Ground State

This subsection will compute the fidelity with the ground state and it will prove that the final state of QITP algorithm is always closer to the Ground State (GS) than the initial one.
We start from the final measured state

$$|\Psi_{fin}\rangle = \frac{1}{\sqrt{P_s}}\,|0\rangle \otimes \frac{e^{-\tau(H-E_T)}}{\sqrt{\mathbb{1}+e^{-2\left(\hat{H}-E_T\right)\tau}}}\,|\psi_{ini}\rangle\ , \tag{7.27}$$

where P_s is the success probability, the probability of measuring the ancilla qubit in $|0\rangle$.
The fidelity (see Sec. 2.5) between the state of Eq. (7.27) and the GS is given by:

$$F = |\langle\phi_0|\Psi_{fin}\rangle|^2 = \left|\frac{c_0 e^{-\tau(E_0-E_T)}}{\sqrt{1+e^{-2\tau(E_0-E_T)}}}\frac{1}{P_s}\right|^2 = \frac{|c_0|^2\, e^{-2\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\frac{1}{P_s^2}\ , \tag{7.28}$$

where the final state is decomposed using the eigenvectors of the Hamiltonian H (see Eq. (7.21)).
Eq. (7.22) gives the probability of measuring the ancilla in $|0\rangle$,

$$P_s^2 = \sum_n \frac{|c_n|^2\, e^{-2\tau(E_n-E_T)}}{1+e^{-2\tau(E_n-E_T)}}\ . \tag{7.29}$$

Like for the success probability, a lower bound for the fidelity can be computed. The lower fidelity can be computed minimizing the whole function of Eq. (7.28) or maximizing its denominator, that is the success probability P_s. Hence, we use the upper bound for P_s (see Eq. (7.23)), obtaining:

$$F \geq \frac{|c_0|^2\, e^{-2\tau(E_0-E_T)}}{1+e^{-2\tau(E_0-E_T)}}\frac{1}{\frac{|c_0|^2 e^{-\tau(E_0-E_T)}}{\sqrt{1+e^{-2\tau(E_0-E_T)}}}+\frac{|c_1|^2 e^{-\tau(E_1-E_T)}}{\sqrt{1+e^{-2\tau(E_1-E_T)}}}}\ , \tag{7.30}$$

where E_1 is the energy of the first excited state.
Rewriting this last equation, we get

$$F \geq F_b = \frac{1}{1+\frac{1-|c_0|^2}{|c_0|^2}\frac{1+e^{2\tau(E_0-E_T)}}{1+e^{2\tau(E_1-E_T)}}} = \frac{|c_0|^2}{|c_0|^2+(1-|c_0|^2)\frac{1+e^{2\tau(E_0-E_T)}}{1+e^{2\tau(E_1-E_T)}}}\ . \tag{7.31}$$

Eq. (7.31) shows the lower of the fidelity of the final state with GS.
Now, we will demonstrate the following lemma.

Lemma 7.2.1. *The final state of the QITP algorithm is* ***always*** *closer to the ground state that the initial state. Therefore, given the initial fidelity $|c_0|^2$ to the GS and the final fidelity F, we have*

$$F \,\geq\, |c_0|^2\,. \tag{7.32}$$

Proof. We demonstrate this lemma proving that the lower bound fidelity of Eq. (7.31) is greater than $|c_0|^2$.
Therefore, we have:

$$F_b = \frac{|c_0|^2}{|c_0|^2 + (1 - |c_0|^2)\frac{1+e^{2\tau(E_0-E_T)}}{1+e^{2\tau(E_1-E_T)}}} \geq |c_0|^2 \tag{7.33}$$

Dividing both sides by $|c_0|^2 > 0$, we obtain

$$\begin{aligned}
&\frac{1}{|c_0|^2 + (1 - |c_0|^2)\frac{1+e^{2\tau(E_0-E_T)}}{1+e^{2\tau(E_1-E_T)}}} \geq 1 \\
&\frac{1}{|c_0|^2 + (1 - |c_0|^2)\frac{1+e^{2\tau(E_0-E_T)}}{1+e^{2\tau(E_0-E_T)}}} - 1 \geq 0 \\
&\frac{1 + e^{2\tau(E_1-E_T)} - \Big((1 + e^{2\tau(E_1-E_T)})\,|c_0|^2 + (1 - |c_0|^2)(1 + e^{2\tau(E_0-E_T)})\Big)}{|c_0|^2\,(1 + e^{2\tau(E_1-E_T)}) + (1 - |c_0|^2)(1 + e^{2\tau(E_0-E_T)})} \geq 0\,.
\end{aligned} \tag{7.34}$$

The denominator is always positive, the sign is given by the numerator:

$$\begin{gathered}
1+e^{2\tau(E_1-E_T)}-|c_0|^2-e^{2\tau(E_1-E_T)}\,|c_0|^2-1-e^{2\tau(E_0-E_T)})+|c_0|^2+e^{2\tau(E_0-E_T)}\,|c_0|^2 \geq 0 \\
e^{2\tau(E_1-E_T)} - e^{2\tau(E_1-E_T)}\,|c_0|^2 - e^{2\tau(E_0-E_T)} + e^{2\tau(E_0-E_T)}\,|c_0|^2 \geq 0 \\
(1 - |c_0|^2)\left(e^{2\tau(E_1-E_T)} - e^{2\tau(E_0-E_T)}\right) \geq 0 \\
(1 - |c_0|^2)\,e^{2\tau(E_0-E_T)}\left(e^{2\tau(E_1-E_T)-2\tau(E_0-E_T)} - 1\right) \geq 0
\end{gathered} \tag{7.35}$$

The last equation is valid if and only if $E_1 \geq E_0 \;\forall\, E_T$ because $\tau \geq 0$. Since E_1 has been defined as the energy of the first excited state; we always have $E_1 \geq E_0$ by definition. Therefore, we can conclude that the fidelity of the final state of the QITP algorithm is always closer to GS than the initial one.

□

Like for the success probability, we should find the optimal range of values of E_T that maximizes the fidelity. We consider for simplicity the limit of $\tau \to \infty$. Looking at Eq. (7.31), we have three possible intervals of E_T, $E_T < E_0$, $E_0 \leq E_T \leq E_1$ and $E_T > 1$.
Starting from $E_T < E_0$, all the exponents are positive, therefore we can write

$$F_b \xrightarrow{\tau\to\infty} \frac{|c_0|^2}{|c_0|^2 + (1 - |c_0|^2)\frac{e^{2\tau(E_0-E_T)}}{e^{2\tau(E_1-E_T)}}} \sim \frac{|c_0|^2}{|c_0|^2 + (1 - |c_0|^2)e^{2\tau(E_0-E_1)}} \sim 1\,. \tag{7.36}$$

In this case the fidelity goes to 1. The other interval, $E_0 \leq E_T \leq E_1$, we have

$$F_b \xrightarrow{\tau\to\infty} \frac{|c_0|^2}{|c_0|^2 + (1-|c_0|^2)\frac{1+0}{e^{2\tau(E_1-E_T)}}} \sim \frac{|c_0|^2}{|c_0|^2 + (1-|c_0|^2)e^{-2\tau(E_1-E_T)}} \sim 1\,. \tag{7.37}$$

In the last range $E_T > E_1$, all the exponents are negative. Therefore, we obtain:

$$F_b \xrightarrow{\tau\to\infty} \frac{|c_0|^2}{|c_0|^2 + (1-|c_0|^2)\frac{1+0}{1+0}} \sim |c_0|^2 \tag{7.38}$$

Finally, we conclude that the best tuning of E_T for maximizing the fidelity is when $E_T < E_1$.

Recapping also the result for the success probability, $E_T > E_0$, we conclude that the best interval is given by $E_0 \leq E_T < E_1$.

Moreover, if $E_T < E_1$, the ground state of the system can be prepared just applying once the QITP operator $U(\tau)$ tuning $\tau \to \infty$. This comes from the fidelity results. Furthermore, in the optimal interval $E_0 \leq E_T < E_1$ the success probability is $\neq 0$ in this limit.

7.2.4 A numerical test for fidelity and success probability

We tested the final results for fidelity and success probability (Eq. (7.22)) for different values of the trial energy E_T. The taken test system was very simple, specifically, it was a system with spectra described by $\left[0,\, 1,\, \frac{\pi}{2}\right]$. The overlaps with the ground state, first and second excited state were $c_0 = 0.1$, $c_1 = 0.4$ and $c_2 = \sqrt{1 - c_0^2 - c_1^2}$ respectively.

Fig. 7.1 shows for different values of E_T the exact fidelity (Eq. (7.28)) and its bound limit (Eq. (7.31)) as function of the imaginary time step τ. The dashed horizontal line indicates the initial fidelity with the GS. We took a wide range of values of E_T covering all the possible combinations.

Furthermore, the success probability (Eq. (7.22)) and its lower bound (Eq. (7.24)) were also evaluated as a function of the imaginary time τ. Fig. 7.2 presents our results. As we expected, all curves of the fidelity and its lower bounds are greater than the initial fidelity for all the E_T values. Hence, we have had another "experimental" proof of the statement about the closeness of the QITP final state to the GS.

We notice that when $E_T < E_1$ the obtained fidelity and its lower bound monotonously grow as a function of τ. In the end, they converge to the value 1 for $\tau \to \infty$ as we expected from eqs. (7.36) and (7.37).

Instead, for $E_T \geq E_1$ the fidelity may show a peak for some value of τ. Moreover, it is interesting to observe the asymptotic limit of the fidelity for $\tau \to \infty$. For $E_T < E_2$ (and in general for $E_T < \|H\|_\infty$) the fidelity tends to a constant value between $|c_0|^2$ and 1. In this range, all contributions of eigenstates of H with energies $E_n > E_T$ have been dissipated. The resulting final state would be in a normalized superposition of the rest of the eigenstates.

In the limit $\tau \to \infty$ and for $E_T \geq E_2$ (in general, $E_T > \|H\|_\infty$), the fidelity

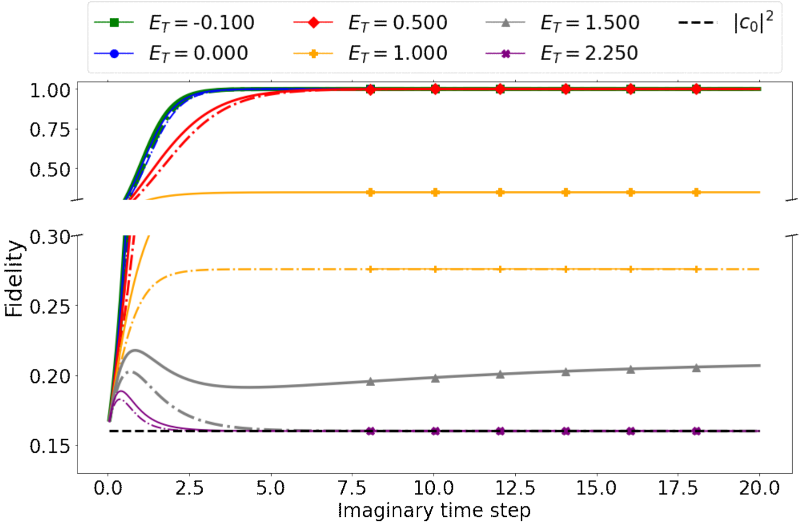


Figure 7.1: Fidelity with the GS for different values of E_T as a function of τ. The initial fidelity with the GS is shown by the black dashed horizontal line (with value $|c_0|^2$). Solid and dashed lines indicate the exact and its lower bound values respectively

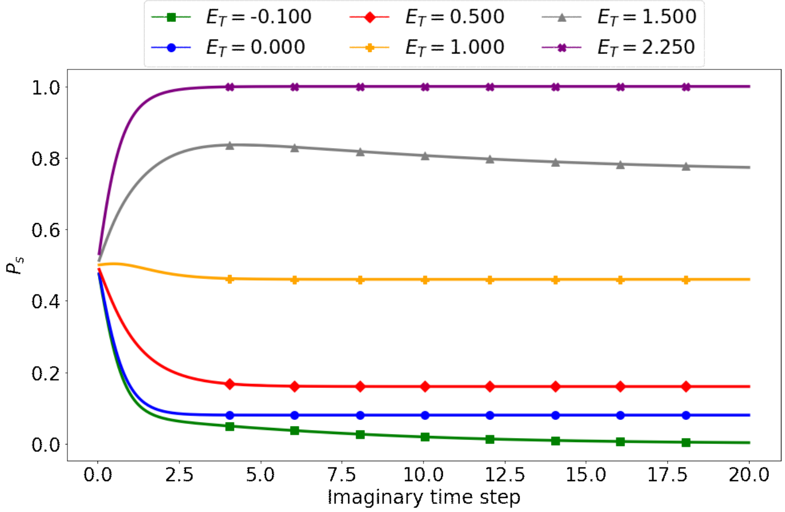


Figure 7.2: Probability to measure the ancilla in $|0\rangle$ (P_s) for different values of E_T as a function of τ.

E_T	Ancilla probability	Fidelity with GS
$E_T < E_0$	0	1
$E_T = E_0$	$\frac{\|c_0\|^2}{2}$	1
$E_0 < E_T < E_1$	$\|c_0\|^2$	1
$E_T = E_1$	$\|c_0\|^2 + \|c_1\|^2$	$\frac{\|c_0\|^2}{\|c_1\|^2+\|c_0\|^2}$
$E_1 < E_T \leq \\|H\\|_\infty$	$\sum_{E_n<E_T} \|c_n\|^2$	$\sum_{E_n<E_T} \frac{\|c_0\|^2}{\|c_n\|^2}$
$E_T > \\|H\\|_\infty$	1	$\|c_0\|^2$

Table 7.1: Ancilla probability and fidelity for different ranges of E_T in the limit $\tau \to \infty$

converges to the initial fidelity. In this range, the energy of the system is not dissipated.

Fig. 7.2 shows the expected behavior for the success probability. As previously discussed, for $E_T < E_0$, it drops to 0. Instead, for $E_0 \leq E_T < E_2$ it goes to a constant value. While for $E_T > E_2$ it converges to 1.

In conclusion, from the two figures, we also demonstrate that the obtained optimal interval to tune E_T is when $E_0 \leq E_T < E_1$. In this interval, we have a good tradeoff for the convergence to GS and reliable success probability.

Tab. 7.1 summarizes the different ranges of E_T for the fidelity and success probability in the limit of $\tau \to \infty$.

7.3 Possible improvements of QITP algorithm: Quantum Amplitude Amplification

This section will present a possible improvement of the QITP algorithm. Indeed, the most significant problem of this method is the limitation due to the probability of measuring the ancilla qubit in $|0\rangle$.

One possible strategy is to use the quantum Amplitude Amplification (AA) algorithm [95, 96, 97] that raises the success probability.

The following subsection will illustrate the basic theory of the AA method.

7.3.1 Theory of Quantum Amplitude Amplification

Let us consider the Hilbert space $\mathbb{H}$ of dimension n and a quantum algorithm $A : \mathbb{H} \to \mathbb{H}$ that has success probability P_g to reach the desired subspace (or state) (called good subspace in Ref. [97]). The failure probability is then $P_b = 1 - P_g$. The Amplitude Amplification (AA) method increases the probability of obtaining the desired subspace.

We start from the state $|\psi\rangle$ that is given by the application of A to an initial state $|\phi\rangle$, $|\psi\rangle = A\,|\phi\rangle$. We can write $|\psi\rangle$ as linear combination of the generators of good ($|\psi_g\rangle$) and not-desired ($|\psi_b\rangle$) subspaces as it follows:

$$|\psi\rangle = cos(\theta)\,|\psi_b\rangle + \sin(\theta)\,|\psi_g\rangle\ , \tag{7.39}$$

where the angle θ is connected to the success probability. Specifically, it is obtained from $\theta = \arcsin\big(\sqrt{P_g}\big)$ and it belongs to $\big[0, \frac{\pi}{2}\big]$.
We define the following amplitude amplification operator

$$Q = -(1\ - 2\,P_\psi)\,(1\ - 2\,P_g)\,, \tag{7.40}$$

where P_ψ and P_g are projector to the state $|\psi\rangle$ and to good subspace spanned by $|\psi_g\rangle$ respectively. We can write $P_\psi = |\psi\rangle\,\langle\psi|$ and $P_g = |\psi_g\rangle\,\langle\psi_g|$. The operator $(1 - P_\psi(P_g))$ flips the sign of the state $|\psi\rangle$ ($|\psi_g\rangle$).
Applying Q to the two subspace one obtains:

$$Q\,|\psi_b\rangle\ =\ (2\cos(\theta)^2 - 1)\ |\psi_b\rangle + 2\ \sin(\theta)\ \cos(\theta)\ |\psi_g\rangle \tag{7.41}$$

and

$$Q\,|\psi_g\rangle\ =\ -2\ \sin(\theta)\ \cos(\theta)\ |\psi_b\rangle\ +\ (1 - 2\ \sin(\theta)^2)\ |\psi_g\rangle\ . \tag{7.42}$$

From these equations, writing the generators of good and not-desired subspace in spinors terms as it follows

$$|\psi\rangle = \begin{pmatrix} |\psi_b\rangle \\ |\psi_g\rangle \end{pmatrix} \tag{7.43}$$

one can notice the action of the operator Q is a rotation between the two subspaces. Therefore, we can write the operator Q as follows:

$$Q = \begin{pmatrix} \cos(2\,\theta) & -\sin(2\,\theta) \\ \sin(2\,\theta) & \cos(2\,\theta) \end{pmatrix}\ . \tag{7.44}$$

Applying n times the operator Q to the state $|\psi\rangle$ we obtain

$$Q^n\,|\psi\rangle = \cos\left((2n+1)\,\theta\right)|\psi_b\rangle\ +\ \sin\left((2n+1)\theta\right)\ |\psi_g\rangle\ . \tag{7.45}$$

Soon we can conclude that if n is chosen such that $\sin\left((2n+1)\theta\right) = 1$, applying n times Q to $|\psi\rangle$ the good subspace ($|\psi_g\rangle$) is selected with probability 1.
The optimal strategy to find the AA operator $Q_{TOT} = Q^n$, the optimal way is to find n_1 that is given by $(2n_1 + 1)\theta = \frac{\pi}{3}$ and then n_2 that finally rotates to $\frac{\pi}{2}$. With this optimization one fixes some numerical issues.
A typical case when we have problem of increasing the success probability is when $P_s\ =\ \frac{1}{2}$. The integer number n such that $\sin\big((2n+1)\frac{\pi}{4}\big)\ =\ 1$ (where $\theta = \arcsin\Big(\sqrt{\frac{1}{2}}\Big)$) is given by solving $(2n+1)\frac{\pi}{4} = \frac{\pi}{2}k\pi$ with $k \in \mathbb{Z}$. The solution is $n = k - \frac{1}{2}$. But, there is not any values of k such that n becomes integer.

Therefore, in this case, the success probability cannot be increased to 1 with a single rotation. Instead, two intermediate steps can be used to raise P_s. First, the success probability is increased to $\frac{\sqrt{3}}{2}$ corresponding of having an angle $\theta = \frac{\pi}{3}$. After that, we can find an other $Q_1^{n_2}$[2] such that the action of $Q_1^{n_2} Q^{n_1}$ increases the success probability to 1.

7.3.2 Amplitude amplification applied to QITP

In order to increase the probability of measuring the ancilla in $|0\rangle$, we can study how to implement the AA algorithm in the QITP method.

First one needs to identify the good and not-desired subspace defined in the AA method. It is easy to see that the ancilla spans the desired subspace in $|0\rangle$ (because we are closer to the GS). The other spanned by $|1\rangle$ is the undesired subspace.

The initial state for the AA algorithm is given by the QITP final state, $|\Psi_{fin}\rangle = U(\tau)\, |\Psi_{ini}\rangle$.

Hence, the amplitude amplification operator Q for the QITP algorithm is obtained from:

$$Q = -(\mathbb{1} - 2\,|\Psi_{fin}\rangle\langle\Psi_{fin}|)\,(\mathbb{1} - 2\,P_A^0) \;=\; -\,(\mathbb{1} - 2\,U(\tau)\,|\Psi_{ini}\rangle\langle\Psi_{ini}|\,U(\tau)^\dagger)\,(\mathbb{1} - 2\,P_A^0)\,, \tag{7.46}$$

where P_A^0 is the projector to the subspace spanned by the ancilla in $|0\rangle$. In our notation, it is given by:

$$P_A^0 \;=\; |0\rangle\langle 0| \otimes \mathbb{1} \;=\; \begin{pmatrix} \mathbb{1} & 0 \\ 0 & 0 \end{pmatrix} . \tag{7.47}$$

The other parameter of AA algorithm, the angle θ, is computed from the probability P_s of measuring the ancilla in $|0\rangle$. In particular,

$$\theta \;=\; \arcsin\left(\sqrt{P_s}\right) . \tag{7.48}$$

Through this AA method the success probability of QITP can be raised to 1 and the GS would be reached more easily than the raw QITP algorithm.

The following section will show some results of QITP using or not the AA method.

7.4 Results of QITP algorithm for the STO-2G Hydrogen atom Hamiltonian

As a first application, we consider the determination of the ground state of Hydrogen atom described in the STO-2G Gaussian basis (for details, see App. C.1).

[2] the amplification operator is different from one of the first step due to a different initial state

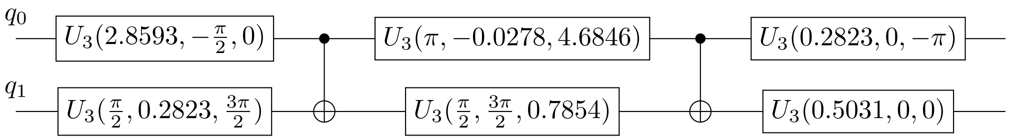

q_0
$U_3(2.8593, -\frac{\pi}{2}, 0)$
$U_3(\pi, -0.0278, 4.6846)$
$U_3(0.2823, 0, -\pi)$
q_1
$U_3(\frac{\pi}{2}, 0.2823, \frac{3\pi}{2})$
$U_3(\frac{\pi}{2}, \frac{3\pi}{2}, 0.7854)$
$U_3(0.5031, 0, 0)$

Figure 7.3: Gate set for implementing $U(\tau)$ for the Hydrogen atom Hamiltonian with $\tau = 15$ Hartree^{-1}. q_1 represents the ancilla qubit and q_0 the system qubit

We begin with computing the STO-2G Hamiltonian and transforming it with the algorithm presented in App. C.1.1. In the end, we obtain a Hamiltonian with the correct spectrum.

We have previously seen that a useful way to implement the QITP method is to use $\tau \to \infty$ to obtain the GS with a single application. Specifically, this limit is obtained choosing $\tau \gg \frac{1}{E_1 - E_0}$. In our case the energy difference is about ~ 1 Hartree[3], therefore, a good candidate is $\tau = 15$ Hartree^{-1}. For the trial energy, we chose $E_T = E_0$.

For the implementation of the propagator $U(\tau = 15\,\text{Hartree}^{-1})$ we can use the two approaches: decomposing the QITP operator in elementary gates (the digital one) and the optimal control technique (the analog one).

We use two qubits for the whole simulation, one is for the system, and the other is for the ancilla. The initial state was prepared in the $|x\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$ state via the Hadamard gate. This state has an overlap probability with the ground state of 0.361.

For decomposing in a digital gate set R_x, R_z and $CNOT$ are used as elementary gates. Moreover, the function *decompose()* of qiskit [81] is applied to compute the quantum circuit with these gates set. The result of this decomposition is shown in Fig. 7.3 where the gates U_3 are given by the specific combination of R_x and R_z.

We implemented and ran the quantum circuit of Fig. 7.3 on the *ibmq_santiago* free-access IBM QPU [7] sampling 8192 values for the probability. A partial measurement of the ancillary qubit on the IBM quantum processors cannot be performed. Instead, all probability distributions of the four states can be obtained. We want the normalized probabilities when the ancilla in $|0\rangle$ (according to Eq. (7.14)). Therefore, one can compute them from experimental occupancies when the ancilla qubit is measured in the $|0\rangle$ state, $p_{0\beta}$, according to:

$$p_\beta(\tau_{\text{tot}}) = \frac{p_{0\beta}(\tau_{\text{tot}})}{\sum_{i \in \{\beta\}} p_{0i}(\tau_{\text{tot}})}\,, \tag{7.49}$$

where β indicates the index of the physical state (for n qubit β runs from 0 to $2^n - 1$), in our case $|\beta\rangle = |0\rangle\,, |1\rangle$. In other words, we are normalizing the state neglecting the contribution coming from the ancillary qubit in $|1\rangle$.

[3] 1 Hartree$\simeq$ 27.21 eV

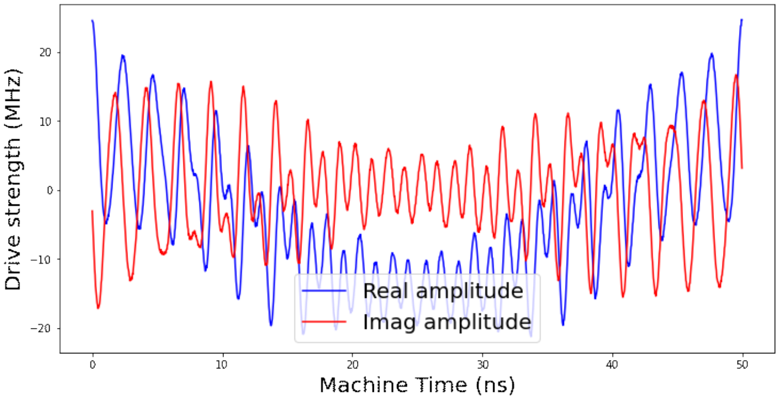


Figure 7.4: Real and Imaginary part of the optimal amplitude computed with GRAPE algorithm

Hydrogen atom system	
Simulation	Fidelity
Simulator of transmon qudit with noise[1]	0.9978
Simulator of transmon qudit without noise[1]	1.00000000
IBM ITP[2]	0.942(4)
IBM ITP+AA[2]	0.9968(4)

Table 7.2: Fidelity results for the different applications of the ITP algorithm. [1] indicates the fidelity is computed with the density matrix obtained solving the Master equation. [2] indicates indicates the fidelity is computed with the state estimated with the tomography described in Sec. 5.4.

For the implementation via optimal control, we compute the pulse for $U(\tau)$ for a 4 level superconducting transmon (see parameters in Tab. 4.5). The optimal obtained amplitude is shown in Fig. 7.4. We perform a device level simulation solving the Lindblad Master equation, including the dephasing and decaying of qudit. Fig. 7.5 shows the obtained results. We also report the results after implementing the Amplitude Amplification method on the IBM QPU.

The blue bars of the histogram indicate the initial probability distribution (given by $|x\rangle$ state), the green the results after the raw QITP, the red the obtained values of QITP with the AA, the orange the numerical results through OC on the transmon, and the gray the exact ground state probability. The black boxes are the uncertainties for the IBM results. The uncertainties of AA simulation are presented, but there are very small).

It is important to highlight two main facts. First, in the simulations on IBM,

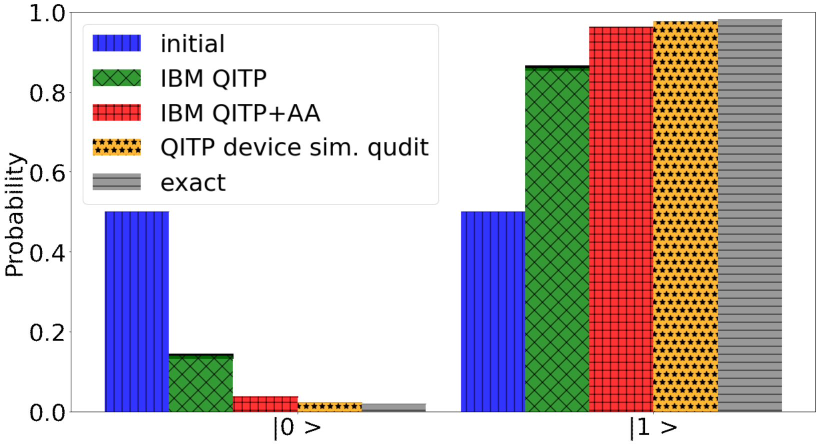


Figure 7.5: Normalized occupation probabilities (p_β) for the computed wave function of the Hydrogen atom using the STO-2G basis set at $\tau = 0$ (blue bar with vertical lines) and $\tau = 15.0$ Hartree^{-1} in two different approaches: the IBM result (green with rhombus grid and red bars with a square grid) and the device level simulations using optimal control with inclusion of hardware noise (dots orange bars) compared to the exact ground-state distribution (gray bar with horizontal line). The black bar represents the uncertainties. The green IBM results are obtained applied the QITP algorithm, and the red results applied the QITP operator with the amplitude amplification (AA) algorithm

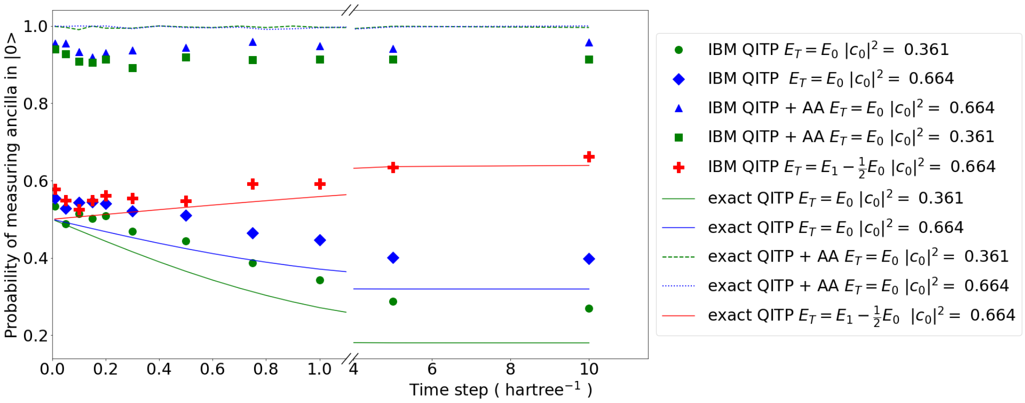


Figure 7.6: Ancilla probability in the $|0\rangle$ state as a function of time step τ and for different overlap with the ground state (GS) applying the ITP operator and ITP operator with the amplitude amplification(AA) with different values of the trial energy E_T. All uncertainties are smaller than 5 10^{-3}.

we did not apply any error mitigation. Therefore both results of the two approaches are bare. The second important thing to consider is that the results

QPUs	GS Fidelity	R. q_0	R. q_1	Cnot	$\sqrt{\sigma_x}(q_0)$	$\sqrt{\sigma_x}(q_1)$
ibmq_lima	0.885(9)	0.0207	0.0169	0.0051	0.00021	0.00023
ibmq_manila	0.882(14)	0.020	0.023	0.0062	0.00016	0.00017
ibmq_santiago	0.950(3)	0.014	0.015	0.0072	0.00023	0.00022
ibmq_quito	0.889(4)	0.036	0.018	0.017	0.00077	0.00031

QPUs	$T_1(q_0)\,(\mu s)$	$T_2(q_0)\,(\mu s)$	$T_1(q_1)\,(\mu s)$	$T_2(q_1)\,(\mu s)$
ibmq_lima	71.88	116.87	98.84	126.62
ibmq_manila	184.94	131.68	100.90	97.51
ibmq_santiago	113.05	64.23	202.74	80.14
ibmq_quito	78.81	67.26	114.10	154.79

Table 7.3: Obtained fidelities between the GS and the normalized state obtained after quantum ITP simulation with $\tau = 15.0$ Hartree^{-1} for different IBM QPUs. The parameters (Readout error of qubit 0 and 1, of Cnot and $\sqrt{\sigma_x}$ implementations) of IBM QPUs are also shown. The initial fidelity was 0.361

for the OC method are obtained from a numerical simulation. Hence, in a real simulation using the OC approach, one can obtain worse results due to readout errors and possible quantum error sources not included in the numerical analysis. However, we can reasonably expect that the realistic experimental results would not move away from what we obtained because a realistic model (with realistic parameters) is used for the quantum machine. Nevertheless, we expect that the results obtained via the OC method would be more accurate than those via elementary gates due to the shorter experimental time machine and less gate infidelity of the OC approach.

However, the probabilities shown in Fig. 7.5 are not the most accurate indications to understand whether this method works or not. Indeed, one should quantify the convergence to the GS with the fidelity between the final state and the GS itself. We calculated the fidelity of the final state and GS using the tomography method presented in Sec. 5.4 to identify the final state. Tab. 7.2 shows the experimental fidelities.

The same quantum circuit of Fig. 7.3 was run on different free access IBM Quantum Experience machines [7].

We kept the same algorithm parameters of the previous execution of IBM results, and using 8192 shots starting from the $|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ state whose fidelity with the GS is 0.361. We chose the trial energy $E_T = E_0$. The numerical results are shown in Tab. 7.3 for different IBM QPUs. The main errors of the QPUs were reported in the same table. In Fig. 7.3, the quantum circuit is based on the R_x, R_z and $CNOT$, but on IBM QPU the R_x gate is not standard. Indeed, in IBM QPUs the elementary gates are $\sqrt{X} = \sqrt{\sigma_x}$ (square root of the

x-pauli matrix), R_z and $CNOT$. Like for U_3 gate, the R_x can be decomposed in a sequence of R_z and $\sqrt{\sigma_x}$ gates.
Furthermore, we also simulated on *ibmq_manilla* IBM QPU the ancilla probability in $|0\rangle$ as a function of time step τ in the propagator for different initial overlap with the GS and different values of the trial energy E_T. As previously, we also obtained the results using amplitude amplification.
Fig. 7.6 shows the obtained results. The green circles and squares are computed from an initial state with an overlap probability of 0.361 with the GS. The blue diamonds and triangles represent the calculation with initial GS fidelity of 0.639. The dashed and continuous lines represent the exact results. Applying amplitude amplification, the probability to measure the ancilla in $|0\rangle$ is significantly improved. Moreover, the success probability when $E_T = E_0 + \frac{1}{2}E_1 > E_0$ was computed. These results are indicated by the red plus symbols and line. As we expected, the ancilla probability is higher than one with $E_T = E_0$.
In the end, we observe that more or less all the points follow their exact curves. The discrepancies are due to the quantum noise present in the processor. As for Fig. 7.5 we did not employ any error mitigation technique to improve the results.

7.5 QITP applied to 2 neutron spin system

The second test of the presented QITP algorithm is the evaluation of the ground state of two "frozen"-neutron systems studied in Ch. 4.
Briefly, our system is composed of two blocked neutrons in their spatial positions. Their dynamics are reduced to the spin part and, the Chiral EFT potential at Leading Order described in Sec. C.2 describes their interactions.
We studied the QITP only via optimal control technique on the transmon device in this test. The used parameters are the same as Tab. 4.5. Fig. 7.7 shows the obtained optimal pulse through the GRAPE algorithm (see App. A.2 for details).
Like before, the required states are mapped on the transmon levels. In this case, four states are needed to describe the spin states couple with an ancilla qubit. Therefore, the total number of states is 8. Our mapping to the quantum processor is as follows: the $|0\rangle$ state corresponds to the tensor product of the ancilla qubit in-state $|0\rangle$ with the uncoupled spin state of $|0\rangle \otimes |\downarrow\downarrow\rangle$, $|1\rangle$ to $|0\rangle \otimes |\downarrow\uparrow\rangle$, $|2\rangle$ to $|0\rangle \otimes |\uparrow\downarrow\rangle$ and $|3\rangle$ o $|0\rangle \otimes |\uparrow\uparrow\rangle$. The next four Fock states have a similar mapping, except for the ancilla qubit being in the $|1\rangle$ state.
Like before, we want to run the QITP algorithm to find the GS with a single application. To do that, we chose as time step $\tau = 1$ MeV^{-1} and $E_T = E_0$. The initial state was chosen as the state with maximum overlap between all the spin states. Specifically, it is given by

$$|\psi_s(\tau = 0)\rangle = \frac{1}{2}(|0\rangle + |1\rangle + |2\rangle + |3\rangle)\,. \tag{7.50}$$

We find that the initial overlap with the GS is 0.407.
Fig. 7.8 and Tab. 7.4 present the obtained normalized probability and final

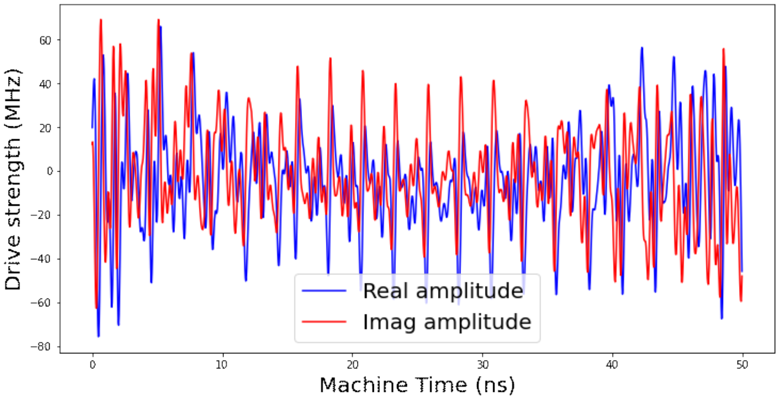


Figure 7.7: Optimal pulse for the QITP application for the two spin neutron system. It is computed through the GRAPE algorithm

Nuclear spin system	
Simulation	Fidelity
Simulator of transmon qudit with noise	0.9919
Simulator of transmon qudit without noise	0.999999996

Table 7.4: Fidelity results for the different applications of the ITP algorithm.

fidelity with the GS, respectively. In Fig. 7.8 the blue, orange and gray bars represent the initial state, the results from the device level simulation and the ground state distribution, respectively. Once again, the results show that the GS is approximated by the projected state with a great accuracy. As before, these results can be affected by readout errors, but we expect that the final experimental data might still be very close to the simulated ones.

7.6 Sequential applications of QITP

In standard calculation employing the Imaginary Time Propagation method, one usually does not employ the ITP operator with a long time step. Instead, we split the imaginary time into small time steps. Applying a sequence of these short-time ITP propagators, one reaches the ground state. The short-time limit can be as usual expressed by the Trotter decomposition to break the propagator into a product of propagators containing simpler Hamiltonian (generally two- or three- body ones).

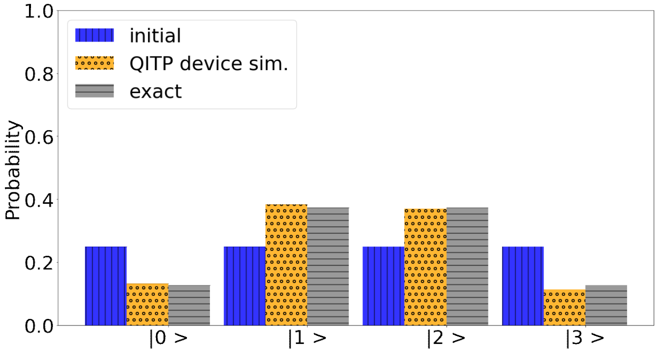


Figure 7.8: Normalized occupation probabilities (p_β) for the two spin neutron system with $\tau = 1$ MeV^{-1} in the optimal control approach. Same legend of Fig. 7.5

Specifically, writing this concept in formula, the whole evolution is given by

$$e^{-\tau(H-E_T)}\left|\psi_0\right\rangle = \left(\prod_i^N e^{-\delta\tau(H-E_T)}\right)\left|\psi_0\right\rangle \ , \tag{7.51}$$

where $\delta\tau = \frac{\tau}{N}$.
Also in QITP, we can implement the imaginary evolution through a sequence of short-time propagators. At each time step, the ancilla is prepared in the $|0\rangle$ state. Then, the $U(\delta\tau)$ operator will be implemented. In the end, the ancillary state will be measured. If the $|0\rangle$ state is obtained, the evolution will be continued. After some time steps the ground state will be reached. All this concept is summarized by the following equation,

$$\left[P_0\, U_{QITP}\left(\frac{\tau}{r}\right)\right]^r |0\rangle \otimes |\psi_{init}\rangle \xrightarrow[r\to\infty]{} c\;|0\rangle \otimes |\phi_0\rangle \ , \tag{7.52}$$

where P_0 describes the projection operator of the ancillary state in the $|0\rangle$ state. A drawback of the presented algorithm is the necessity of measuring the ancilla qubit in the $|0\rangle$ state at each time step. This requirement slows down the efficiency of QITP. Sec. 7.7 will study better the scaling of QITP algorithm.
Here possible solutions will be investigated to avoid measuring the ancilla qubit increasing the efficiency of the presented method.
The first solution is to use the Amplitude Amplification method (see Sec. 7.3.2). The second solution is to add to the simulation many ancilla qubits. The idea is to implement the same QITP gate, changing only the ancillary qubit. The closest final state to GS for the physical system is obtained measuring all the ancilla qubits in $|0\rangle$ states. For instance, we implement a quantum circuit for $U(\tau)$ using the first ancilla qubit; then, we copy and paste the same quantum circuit for $U(\tau)$ changing from the first to the second the ancillary qubits. If we have N ancilla qubits, we will continue to do until we finish the ancilla qubits.

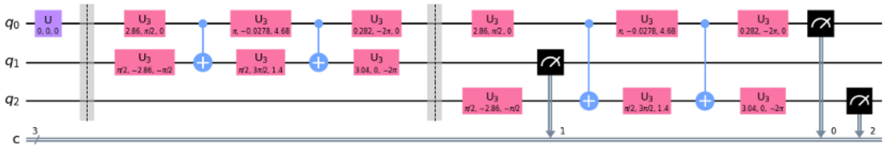


Figure 7.9: Quantum circuit using two ancilla qubits to further propagate in imaginary time step. The first qubit q_0 represents the physical system, in this case, the STO-2G Hydrogen atom, q_1 and q_2 the ancilla qubits

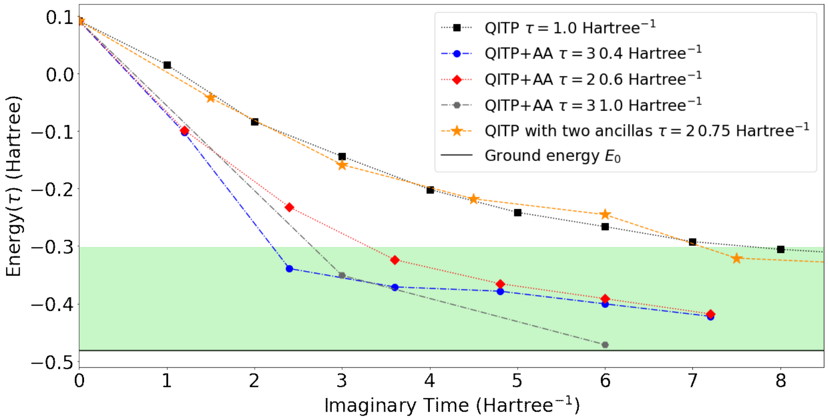


Figure 7.10: Results for the expectation value of the energy as a function of imaginary time for different time steps τ with a reinitializing procedure. The light-green color bar indicates the energy range with fidelity with GS greater than 80%. The cases indicated with AA are computed with Amplitude Amplification. The case labeled with "2nd ancilla" represented the simulation done with two ancilla qubits. The integer factor in front of the time step indicates the number of repetitions of the ITP quantum circuit before the tomography procedure.

Implementing the QITP with N ancillary qubits allow us to propagate longer the imaginary evolution reaching a time step of $N\,\tau$. Fig. 7.9 shows an example of a quantum circuits.

In addition, the last third possible solution is to employ a reinitializing procedure (see Ch. 5). We perform a state tomography to identify the state of the physical system. Then, we would reinitialize the quantum circuit into the evaluated state. For example, we propagate an initial state employing the QITP algorithm, reaching some final state for the physical state. This final state which can be identified by tomography is closer to the GS. Now, starting from this identified state and applying the QITP method again, we keep converging towards the GS. As we said in Sec. 5.4, state tomography is not a scalable method.

However, it can be applied to study physical systems with few qubits.
Fig. 7.10 shows the obtained results of the ground state of STO-2G Hamiltonian through a sequence of short-imaginary time QITP employing the three discussed solutions. The simulations were implemented through the *manilla* IBM quantum processor [7] with 8192 shots. The plotted energies indicate the Hamiltonian's expectation value as a function of the imaginary time evolution. To evaluate these energy values, the state tomography of Sec. 5.4 (for a single qubit case) was employed, obtaining a pure state that approximates the state in the quantum processor. With this state, we evaluated its energy. The results are plotted in the figure.
To continue the evolution after identifying the state, we reinitialize the quantum circuit through a U_3 gate (the general single-qubit gate). This U_3 gate moved the $|0\rangle$ state of the processor to one we had identified before.
In Fig. 7.10, squares indicate the results obtained applying the reinitializing procedure after implementing a single QITP operator. Stars represent the results obtained by the imaginary time propagation through two ancilla qubits. Specifically, we implemented the circuit of Fig. 7.9 where the QITP operator is given by $U(\tau = 0.75\,\mathrm{Hartree}^{-1})$. The state tomography was employed identifying the state when the two ancillary qubits are in the $|00\rangle$ state.
The diamonds, circles and hexagons of the figure Fig. 7.10 represent the obtained results implementing the AA method. In the legend, the time step is written as a multiplication of an integer number and a real number. The integer factor indicates how many times the quantum circuits of $U(\tau)$ operator with the corresponding amplification operator are implemented. The real number indicates the value of τ in the QITP operator.
The green bar of Fig. 7.10 indicates the energies of states that have fidelity with GS greater than 0.80. It is shown to highlight the accuracy of our propagation. Indeed, if the final state gets a fidelity larger than 0.80, we can conclude that the ground state is obtained. This threshold value is chosen because the performed simulations have run through a noisy free access IBM quantum processor.
Observing Fig. 7.10, we observe that the results from amplitude amplification reach final energies very close to the ground energy. The others slowly converge to the ground energy. Nevertheless, we conclude that we obtain the ground state from all the final experimental states because they get fidelity higher than 80%.

7.7 Scalability and Trotter decomposition of QITP algorithm

This section will focus on the limit for $\tau \to 0$ to show the scalability of the QITP algorithm implemented with the Trotter decomposition.
Specifically, let H be the total Hamiltonian of the physical system and $H = \sum_{l=0}^{L} H_l$ where H_l are Hamiltonians acting on a small fraction of qubits. For example, the nuclear Hamiltonian is a sum of two- and three-body interactions;

therefore, H_l would be the single term of the one-, two- or three-body Hamiltonian.

We start building the short-time imaginary propagators. Given the final imaginary time τ, we want to apply r times the $U(\delta\tau)$ where $\delta\tau = \frac{\tau}{r}$ and r describes the total number of time steps. The QITP algorithm is based on measuring the ancilla qubit in $|0\rangle$. Therefore, in the case of L Hamiltonians and r Trotter time step, this can be done with $r\,L$ ancillary qubits and post-selecting their state in $|0\rangle^{\otimes_i^{Lr}}$. It is the second presented solution of Sec. 7.6. Equivalently we can use a single ancilla qubit where we post-selecting it in $|0\rangle$ state for $r\,L$ times.

We start generalizing the $U(\tau)$ operator with the following

$$\hat{U}(\tau,\eta) = \begin{pmatrix} \frac{e^{-\tau(\hat{H}-E_T)}}{\sqrt{\eta^2+e^{-2\tau(\hat{H}-E_T)}}} & \frac{\eta}{\sqrt{\eta^2+e^{-2\tau(\hat{H}-E_T)}}} \\ \frac{\eta}{\sqrt{\eta^2+e^{-2\tau(\hat{H}-E_T)}}} & -\frac{e^{-\tau(\hat{H}-E_T)}}{\sqrt{\eta^2+e^{-2\tau(\hat{H}-E_T)}}} \end{pmatrix}\,, \tag{7.53}$$

where η is a real positive number. The operator $U(\tau)$ exposed in this chapter is given by $\eta = 1$.

We are interested in implementing the Trotter decomposition through r time steps. The generalized QITP operator of r time step is given by

$$\left(\hat{P}_0\hat{U}(\tau,\eta)\right)^r = \begin{pmatrix} \left(\frac{e^{-\tau(\hat{H}-E_T)}}{\sqrt{\eta^2+e^{-2\tau(\hat{H}-E_T)}}}\right)^r & 0 \\ 0 & 0 \end{pmatrix}\,, \tag{7.54}$$

where $P_0 = |0\rangle\langle 0|$ is the projection operator of the ancilla qubit in the $|0\rangle$ state. Appendix F of Ref. [98] shows an upper bound of the error using the Trotter decomposition for a standard real or imaginary time propagation. Summarizing the results, the error grows as $\frac{(\Lambda\delta t L)}{r}e^{\Lambda\delta t L}$. (for details, see App. F)

We generalize the results of the paper for the QITP algorithm obtaing the following Lemma.

Lemma 7.7.1. *Let H be a Hermitian operator expressed as the sum of L Hermitian operators $\{\hat{H}_1, .., \hat{H}_L\}$ as $\hat{H} = \sum_l^L \hat{H}_l$ and $\Lambda_T = \max_j \left\|\hat{H}_j - E_T\right\|_\infty$. Then,*

$$\left\|\hat{Q}_{ITP}(\eta,\tau) - \prod_k^L \hat{Q}^{(k)}_{ITP}(\eta,\tau)\right\| \leq L^2\Lambda_T^2\tau^2\,, \tag{7.55}$$

where

$$\hat{Q}^{(k)}_{ITP}(\eta,\tau) = \frac{e^{-\tau\left(\hat{H}_k - \frac{E_T}{L}\right)}}{\sqrt{\eta^2 + e^{-2\tau\left(\hat{H}_k - \frac{E_T}{L}\right)}}}. \tag{7.56}$$

Proof. We first introduce the notation $\mathcal{R}_k(f)$ to denote the remainder of the truncated Taylor series of an analytic function f. Explicitly, for a generic function $f(x)$ it is given by:

$$R_k(f) = \sum_{i=k+1}^{\infty} \frac{1}{i!}\frac{d^i f}{dx^i}(a)(x-a)^i \; = \sum_{i=k+1}^{\infty} \frac{f^i(a)}{i!}(x-a)^i \,. \tag{7.57}$$

In the literature, one can find that different forms of the remainder, the Langrange form [99],

$$R_k(x) = \frac{f^{k+1}(\epsilon)}{k+1!}(x-a)^k + 1 \tag{7.58}$$

with $\epsilon \in [x,\, a]$ and or the integral form [99]

$$R_k(x) = \int_a^x \frac{f^{k+1}(t)}{k!}\,(x-t)^k\, dt \,. \tag{7.59}$$

We are interested in the following simplified form of the $QITP$ operator,

$$f(x) = \frac{e^{-x}}{\sqrt{\eta^2 + e^{-2\,x}}}\ , \tag{7.60}$$

with $\eta > 0$. This is analytic for any real x and using Langrange's expression for the Taylor series remainder with $a = 0$ we can write

$$\mathcal{R}_k(f;x) = \sum_{m=k+1}^{\infty} \frac{f^m(x)}{m!} x^m = \frac{f^{k+1}(\xi)}{(k+1)!} x^{k+1} \tag{7.61}$$

for some $\xi \in [0, x]$. We want a bound for the full QITP operator. Therefore we must study the case when $k = 1$.

Computing the second derivative of $f(x)$, we have

$$\begin{aligned} \left|f^2(x)\right| &= \left| \frac{e^{-x}}{\sqrt{\eta^2 + e^{-2\,x}}} - 4\frac{e^{-3x}}{(\eta^2 + e^{-2\,x})^{3/2}} + 3\frac{e^{-5x}}{(\eta^2 + e^{-2\,x})^{5/2}} \right| \\ &= \left| \frac{e^{-x}}{\sqrt{\eta^2 + e^{-2\,x}}} \left(1 - \frac{e^{-2x}}{\eta^2 + e^{-2x}} \left(4 - 3\frac{e^{-2x}}{\eta^2 + e^{-2x}}\right)\right) \right| \\ &\le \left| \frac{e^{-x}}{\sqrt{\eta^2 + e^{-2\,x}}} \left(1 - \frac{e^{-2x}}{\eta^2 + e^{-2x}}\right) \right| \\ &\le \left| \frac{e^{-x}}{\sqrt{\eta^2 + e^{-2\,x}}} \right| \le 1 \,. \end{aligned} \tag{7.62}$$

The last two ineequalities come from the fact that $f(x)$ is between 0 and 1. Fig. 7.11 shows $f(x)$, $f^1(x)$ and $f^2(x)$ as a function of x. Consider now the generalization of $f(x)$ with an operator function

$$\begin{aligned} f(t\hat{H}) &= \sum_n |n\rangle\langle n| f(tE_n) \\ &= \sum_{m=0}^{\infty} \sum_n |n\rangle\langle n| \frac{f^m(tE_n)}{m!}(tE_n)^m \,, \end{aligned} \tag{7.63}$$

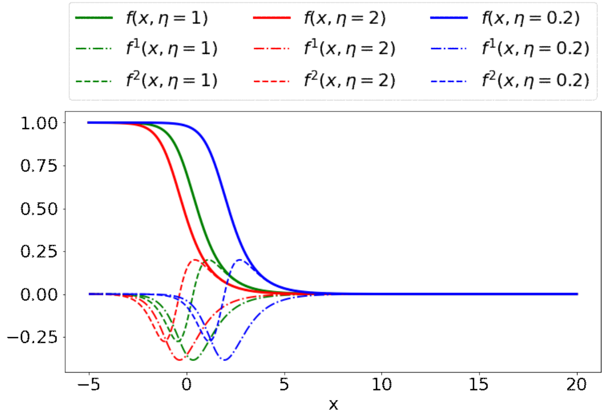


Figure 7.11: $f(x)$, $f^1(x)$ and $f^2(x)$ as a function of x

where $|n\rangle$ indicates the eigenstate of H with eigenvalue E_n.
Noting that $f(x)$ is positive definite we can bound the remainder for $k=1$ with

$$\left\|\mathcal{R}_1(f,t\hat{H})\right\| = \left\|\frac{f^2(t\hat{H})}{2}(t\hat{H})^2\right\| \leq \frac{t^2\|\hat{H}\|^2}{2}\,. \tag{7.64}$$

Consider now the remainder of the product

$$f_L(t,\vec{x}) = \prod_{l=1}^{L} f(tx_l)\,, \tag{7.65}$$

written in terms of derivatives with respect to the scalar variable t (for bookkeeping purposes)

$$f_L^1(t,\vec{x}) = \sum_{m=1}^{L} x_m f^1(tx_m) \prod_{l\neq m}^{L} f(tx_l)\,, \tag{7.66}$$

and

$$\begin{aligned} f_L^2(t,\vec{x}) = & \sum_{m=1}^{L} x_m^2 f^2(tx_m) \prod_{l\neq m}^{L} f(tx_l) \\ & + \sum_{m=1}^{L}\sum_{k\neq l}^{L} x_m x_k\, f^1(tx_m) f^1(tx_k) \prod_{l\neq m,k}^{L} f(tx_l) \\ & \leq \sum_{m=1}^{L} x_m^2 f^2(tx_m) + \sum_{m=1}^{L}\sum_{k\neq l}^{L} x_m x_k f^1(tx_m) f^1(tx_k) \\ & \leq L^2 \max_k [x_k^2]\,, \end{aligned} \tag{7.67}$$

since $|f^1(x)| \leq 1$ and $|f^2(x)| \leq 1$ for any real x. This follows the same argument used to derive Eq. (7.62).
We now get for the remainder:

$$\begin{aligned} \mathcal{R}_1\left[\prod_{l=1}^{L} f(t\hat{H}_l)\right] &= \left\| \frac{f_L^2(t,\hat{H})}{2} t^2 \right\| \\ &\leq \frac{t^2 L^2 \max_l[\|\hat{H}\|_l^2]}{2} \,. \end{aligned} \tag{7.68}$$

In order to simplify the notation we will use $Q = Q_{ITP}(\eta,\tau)$ and $Q^{(k)} = Q_{ITP}^{(k)}(\eta,\tau)$ whenever there is no risk of confusion. Using the triangular inequality, we get:

$$\begin{aligned} \left\| Q - \prod_k^L Q^{(k)} \right\| &= \left\| \mathcal{R}_1\left(Q - \prod_k^L Q^{(k)}\right) \right\| \\ &= \left\| \mathcal{R}_1\left(Q\right) - \mathcal{R}_1\left(\prod_k^L Q^{(k)}\right) \right\| \\ &\leq \|\mathcal{R}_1\left(Q\right)\| + \left\| \mathcal{R}_1\left(\prod_k^L Q^{(k)}\right) \right\| \,. \end{aligned} \tag{7.69}$$

Using the results obtained above we have

$$\begin{aligned} \|\mathcal{R}_1\left(Q_{ITP}(\eta,\tau)\right)\| &\leq \frac{\tau^2 \|\hat{H} - E_T\|^2}{2} \\ &\leq \frac{\tau^2 L^2 \Lambda_T^2}{2} \,, \end{aligned} \tag{7.70}$$

while for the product

$$\left\| \mathcal{R}_1\left(\prod_k^L Q^{(k)}(\eta,\tau)\right) \right\| \leq \frac{\tau^2 L^2 \Lambda_T^2}{2} \,. \tag{7.71}$$

Note that neither bound explicitly depends on the choice of the η parameter. The results easily follow by summing these two contributions. From Eq. (7.70), we get:

$$\begin{aligned} \left\| Q - \prod_k^L Q^{(k)} \right\| &\leq \|\mathcal{R}_1\left(Q\right)\| + \left\| \mathcal{R}_1\left(\prod_k^L Q^{(k)}\right) \right\| \\ &\leq \frac{\tau^2 L^2 \Lambda_T^2}{2} + \frac{\tau^2 L^2 \Lambda_T^2}{2} \leq \tau^2 L^2 \Lambda_T^2 \,. \end{aligned} \tag{7.72}$$

□

We have proved that the error of the Trotter decomposition is of order $\epsilon \sim O(\tau^2 L^2 \Lambda_T^2)$. Considering $\tau = r\delta\tau$, we need that $\epsilon \ll 1$ to obtain a good approximation of the total QITP operator. Therefore, we need that

$$\delta\tau = \mathcal{O}\left(\sqrt{\frac{\epsilon}{r}}\frac{1}{L\Lambda_T}\right) \ll 1 \ . \tag{7.73}$$

Applying r times the $\hat{U}(\delta\tau, \eta)$ with the measuring the ancilla qubit in $|0\rangle$ gives that the fidelity with GS becomes:

$$F = \left(1 + \sum_n \frac{|c_n|^2}{|c_0|^2} \frac{\left(\eta^2 e^{2\,\delta\tau\,(E_0 - E_T)} + 1\right)^r}{\left(\eta^2 e^{2\,\delta\tau\,r\,(E_n - E_T)} + 1\right)^r}\right)^{-1} \tag{7.74}$$

Using the result of Eq. (7.31) for the fidelity bound, considering a generic η and $r \in \mathbb{N}$ we get:

$$F \geq \left(1 + \frac{1 - |c_0|^2}{|c_0|^2}\left(\frac{\eta^2 e^{2\delta\tau(E_0 - E_T)} + 1}{\eta^2 e^{2\delta\tau(E_1 - E_T)} + 1}\right)^r\right)^{-1} . \tag{7.75}$$

In order to bound the behavior in the parenthesis we use

$$\begin{aligned}\frac{\eta^2 e^{2\delta\tau(E_0 - E_T)} + 1}{\eta^2 e^{2\delta\tau(E_1 - E_T)} + 1} &= \frac{\eta^2 + e^{-2\delta\tau(E_0 - E_T)}}{\eta^2 e^{2\delta\tau\Delta} + e^{-2\delta\tau(E_0 - E_T)}} \\ &= 1 - \eta^2 \frac{e^{2\delta\tau\Delta} - 1}{\eta^2 e^{2\delta\tau\Delta} + e^{-2\delta\tau(E_0 - E_T)}} \ ,\end{aligned} \tag{7.76}$$

where $\Delta = E_1 - E_0$.
Now assume then we choose $E_1 > E_T > E_0$. This means that $E_T - E_0 < \Delta$, and we have:

$$\begin{aligned}\frac{\eta^2 e^{2\delta\tau(E_0 - E_T)} + 1}{\eta^2 e^{2\delta\tau(E_1 - E_T)} + 1} &< 1 - \frac{\eta^2}{1 + \eta^2}\frac{e^{2\delta\tau\Delta} - 1}{e^{2\delta\tau\Delta}} \\ &\leq 1 - \frac{\eta^2}{1 + \eta^2}\frac{2\delta\tau\Delta}{1 + 2\delta\tau\Delta} \ .\end{aligned} \tag{7.77}$$

Considering an error ϵ sufficiently small or r sufficiently large to be in the limit $2\delta\tau\,\Delta < 1$, one can prove that:

$$\begin{aligned}\left(\frac{\eta^2 e^{2\delta\tau(E_0 - E_T)} + 1}{\eta^2 e^{2\delta\tau(E_1 - E_T)} + 1}\right)^r &\leq \left(1 - \frac{\eta^2}{1 + \eta^2}\frac{2\delta\tau\Delta}{1 + 2\delta\tau\Delta}\right)^r \\ &\leq \left(1 - \frac{\eta^2}{1 + \eta^2}\delta\tau\Delta\right)^r \\ &\leq \exp\left(-\frac{\eta^2}{1 + \eta^2} r\delta\tau\Delta\right) \ .\end{aligned} \tag{7.78}$$

Hence, we found that the fidelity is given by

$$F \sim \left[1 + \frac{1 - |c_0|^2}{|c_0|^2}\mathcal{O}\left(\exp\left(-\frac{\eta^2}{1 + \eta^2}\frac{\Delta}{L\Lambda_T}\sqrt{r\epsilon}\right)\right)\right]^{-1} . \tag{7.79}$$

Looking at the result of the last equation, we can conclude that the fidelity approaches sub-exponentially 1 in the $r \to \infty$ limit.
We must also compute the success probability as a function of r. We have

$$\begin{aligned} P(0;r) &= \langle\Psi|\left(\hat{M}_0\hat{U}(\delta\tau,\eta)\right)^{2r}|\Psi\rangle \\ &= \langle\Psi|\hat{Q}_{ITP}(\delta\tau)^{2r}|\Psi\rangle \\ &\geq |c_0|^2\langle\phi_0|\hat{Q}_{ITP}(\delta\tau)^{2r}|\phi_0\rangle \\ &\geq \frac{|c_0|^2}{(\eta^2 e^{2\delta\tau(E_0-E_T)}+1)^r} \end{aligned} \tag{7.80}$$

which, for the choice $E_1 > E_T > E_0$ can be rewritten more clearly as

$$P(0;r) \geq \frac{|c_0|^2}{(\eta^2 e^{-2\delta\tau|E_0-E_T|}+1)^r} \geq \frac{|c_0|^2}{(\eta^2+1)^r}, \tag{7.81}$$

which exponentially decays with the number of steps. By tuning η one can make also this behavior to be sub-exponential, at the expense of a slower convergence of the fidelity with step number r. This results holds only for the exact short-time unitary $\hat{U}(\delta\tau,\eta)$ but using the approximation in Eq. (7.55) of the Lemma above we still have that the difference in probabilities is bounded as:

$$\begin{aligned} \delta P &= \left|\langle\Psi|\hat{Q}_{ITP}(\delta\tau)^{2r}|\Psi\rangle - \langle\Psi|\prod_k^L \hat{Q}^{(k)}_{ITP}(\eta,\tau)^{2r}|\Psi\rangle\right| \\ &= \left|\langle\Psi|\left(\hat{Q}_{ITP}(\delta\tau)^{2r} - \prod_k^L \hat{Q}^{(k)}_{ITP}(\eta,\tau)^{2r}\right)|\Psi\rangle\right| \\ &\leq \left\|\hat{Q}_{ITP}(\delta\tau)^{2r} - \prod_k^L \hat{Q}^{(k)}_{ITP}(\eta,\tau)^{2r}\right\| \\ &\leq 2rL^2\Lambda_T^2\delta\tau^2 \ . \end{aligned} \tag{7.82}$$

In other words, if we use an approximation for the full sequence of r steps with error bounded by ϵ, the success probability will be at most 2ϵ smaller. Since the decay of $P(0;r)$ is exponential in r, the lower bound in Eq. (7.81) will go to zero at some finite number of steps. This is possibly a consequence of the looseness of the bound for δP given above.

7.7.1 Case for standard QITP

The obtained results of this section are general. This subsection will present them in the case of $\eta = 1$. When $\eta = 1$, we obtain the QITP operator discussed in this chapter.
Therefore, the Lemma 7.7.1 becomes

Lemma 7.7.2. *Let H be a Hermitian operator expressed as the sum of L Hermitian operators $\{\hat{H}_1,..,\hat{H}_L\}$ as $\hat{H} = \sum_l^L \hat{H}_l$ and $\Lambda_T = \max_j \left\|\hat{H}_j - E_T\right\|_\infty$.*

Then,

$$\left\| \hat{Q}_{ITP}(\eta,\tau) - \prod_{k}^{L} \hat{Q}^{(k)}_{ITP}(\eta,\tau) \right\| \leq L^2 \Lambda_T^2 \tau^2 \,, \tag{7.83}$$

where

$$\hat{Q}^{(k)}_{ITP}(\eta,\tau) \;=\; \frac{e^{-\tau\left(\hat{H}_k - \frac{E_T}{L}\right)}}{\sqrt{\eta^2 + e^{-2\tau\left(\hat{H}_k - \frac{E_T}{L}\right)}}}. \tag{7.84}$$

In order to use the Trotter decomposition in r steps, we need

$$\delta\tau = \mathcal{O}\left(\sqrt{\frac{\epsilon}{r}}\frac{1}{L\Lambda_T}\right) \ll 1 \,. \tag{7.85}$$

The order of the fidelity bound becomes

$$F \;\leq\; \frac{1}{1 + \frac{1-|c_0|^2}{|c_0|^2}\,\mathcal{O}\left(\exp\left(-\frac{1}{2}\frac{\Delta}{L\Lambda_T}\sqrt{r\epsilon}\right)\right)} \,. \tag{7.86}$$

The success probability after r steps is given by:

$$P(0;r) \geq \frac{|c_0|^2}{(e^{2\delta\tau(E_0 - E_T)} + 1)^r}, \tag{7.87}$$

which, for the choice $E_1 > E_T > E_0$ can be rewritten more clearly as

$$P(0;r) \geq \frac{|c_0|^2}{(e^{-2\delta\tau|E_0 - E_T|} + 1)^r} \geq \frac{|c_0|^2}{2^r}, \tag{7.88}$$

that decays exponentially with the number of steps.

In conclusion, we summarize the obtained results for scalability. We proved that the probability decays exponentially with r, and the fidelity goes to 1 subexponentially. The first fact can be a massive problem for a long imaginary time propagation. After some time steps, the success probability is very close to 0. However, we have proved that we increase the fidelity with the ground state. We may also improve the convergence to GS with high order Trotter expansions [98, 100] or we can raise the success probability. A further study should be done.

The presented method, though not scaling perfectly, could still be used as a preconditioner for some quantum algorithms to efficiently increase the initial fidelity with the GS. Indeed, Ref. [91] presents one of the most efficient quantum algorithms to prepare the ground state. The efficiency is given by (taken from Tab. 1 of Ref. [91])

$$O\left(\frac{\alpha}{\gamma\Delta}\log\frac{1}{\epsilon}\right) \,, \tag{7.89}$$

where Δ, γ and ϵ indicate the gap between first excited and ground states, the initial fidelity with GS and the final fidelity error, respectively (α is another

parameter of the algorithm, for more details, see the reference). We notice a possible problem of this algorithm: it depends explicitly on the initial fidelity $\sim \frac{1}{\gamma}$.
Suppose we want to prepare/study the ground state of some complex Hamiltonian where its GS is unknown. In that case, we usually start with a state that has a very small overlap with GS. If we use just the algorithm of Ref. [91] we will be in a situation where the efficiency of the calculations is reduced due to the low overlap with the GS. Applying the QITP algorithm for a few time steps, a new state with greater fidelity can be obtained and a measurable success probability. Then, the most efficient algorithm, Ref. [91], will be applied to finish preparing the ground state. Indeed, neglecting the decay problem of the success probability, the efficiency of the QITP algorithm depends just on the decay time from the first excited state to the ground state ($\tau \sim \frac{1}{E_1 - E_0}$)

Chapter 8

Conclusions

In this thesis, some new quantum algorithms based on real and imaginary time propagation have been described. Their ultimate use would hopefully be for simulating the dynamics and structure of many-body quantum systems. Specifically, through real time propagation, we could eventually study the real time evolution of quantum systems. One open problem is the direct simulation of nucleus-nucleus scattering. On a quantum computer we could start from some incoming particles, and through real time evolution, we might evaluate the cross-sections of all possible channels at the same time. Instead, through imaginary time propagation, we can prepare the ground state of quantum systems, evaluating some physical properties of their structure.

The extremely large computational resources requested by classical algorithms to simulate quantum systems leads to the computational cost growing exponentially with the number of particles. Hence, quantum computing might be a possible solution to simulate complex physical systems. We presented some essential aspects of quantum information and quantum simulation. The structure of utilized quantum hardware based on superconducting devices were briefly described.

After the introductory chapters, the work of Ref. [9] was presented, showing how we can extract from the real time evolution the spectra of a Hamiltonian. In that work the evolution of the spins of two neutrons blocked in space was studied. We generalized the simulation of that system, re-including the spatial degrees of freedom. Specifically, a hybrid algorithm (defined as ”coprocessing”) was developed. The spin evolution is simulated into quantum processors and the spatial dynamics in a classical device. This scheme represents the first step towards computationally simulating a scattering experiment through a quantum processor.

We presented the results of simulations implementing the coprocessing method studying how we can overcome the contribution of noise sources. The most significant result is the reinitializing procedure based on the presented state tomography process.

A possible generalization of the coprocessing scheme would study the full quantum simulation of scattering experiments. One may think to sample from the spatial distribution paths and evaluate through quantum processors the spin probability or state. Summing all the contributions from the paths, we will reach the actual quantum transition probability according to the path integrals theory. However, this approach must deal with very complicated problems for many-body simulations. The main ones are the difficulty of sampling from the real-time path integral and the treating of the Fermion sign problem. Moreover, although we neglect these problems (for example, simulating two-body dynamics), we must consider that the global phase of the final spin state for each path becomes a relative phase in the path sum. Therefore, we must measure it and save it some register. Nevertheless, further study to test if the coprocessing scheme can be generalized may be desirable.

During the real time evolution description, we have also discussed the problems of the analog or digital quantum approaches. The former, the analog approach or optimal control, tailors pulses for implementing with a single gate a generic unitary operation. The drawback is the potentially very large amount of classical computational resources to compute these pulses. We have seen that a possible solution consists of reducing the computational time by interpolating the spectrum of pulses as a function of the Hamiltonian parameters.

Instead, in the digital approach, we decompose a generic quantum operation in elementary gates. We have discussed the need of efficiently compiling an arbitrary transformation in digital gates. Indeed, in the translation, we should consider that the high depth of quantum circuits usually spoils of the accuracy of results due to the high contribution of noise. For the compilation of real time propagators in digital gates, we proposed the Similarity Renormalization Group as a way to simplify the compilation reaching from N qubit gate the simpler 2 qubit gates. The future solution for compiling generic quantum gates would be a mix of two approaches: decomposing them in gates with smaller dimensionality and tailoring the pulses for implementing them.

A version of the Quantum Imaginary Time Propagation (QITP) was presented at the end of the work. We started from the Imaginary Time Propagation theory, proving that an arbitrary initial state (with a non-zero overlap with the ground state) decays to the ground state along with the imaginary time evolution. The problem of translating this method from a classical algorithm to a quantum one is the non-unitarity of the Imaginary Time propagator. The unitarity is a necessary request to get a quantum gate. We demonstrated that, in a quantum algorithm, the action of adding an ancillary qubit in order to extend the Hilbert space to allow controlled dissipation. Implementing a specific quantum unitary operator ($U(\tau)$) and measuring the ancilla in $|0\rangle$ is equivalent to implementing the standard Imaginary Time Propagation in a classical device.

The presented QITP algorithm was tested for different physical quantum systems, Hydrogen atom and spin-neutron system, in different conditions. Specifically, we demonstrated that we could reach the ground state applying just once the $U(\tau)$ operator if we tune $\tau \to \infty$. Furthermore, different improvements to the QITP were discussed and the imaginary time evolution was simulated using imaginary short-time propagators applying these improvements. In all the simulations, a very good approximation of the ground state was obtained.

We also presented a theoretical analysis of the QITP, focusing on its success probability and fidelity with the ground state(GS). We also computed them employing the Trotter decomposition. In this case, we showed the exponential decay of the success probability to 0 with the number of steps. On the contrary, the final state reaches the GS subexponenially. The decay of success probability is a huge problem in the efficiency of the QITP algorithm. However, the QITP can be very useful as a preconditioner for most efficient algorithms [91]. Indeed, these algorithms require good initial fidelity to be efficient. We can increase the initial fidelity with GS of this algorithm by employing the QITP for few steps earlier. After that, we will implement these most efficient quantum algorithms to complete the preparation of the GS.

We are still in the NISQ era, where the quantum processors are composed of a few noisy qubits. Hence, classical computers are is still much more performing than the quantum ones. Nevertheless, in the future, with the leveraging of the improvements of quantum machines and quantum compilation, we might be able to study physical systems that are inaccessible in the present day, helping us to understand the universe.

Appendix A

Optimal control and the GRAPE algorithm

A.1 Optimal control technique

Sec. 3.2 presented how experimentally we implement a quantum gate driving the real time evolution of qubits. The following equation recalls the most important equation of that section,

$$U_{log} = \mathbb{T} \exp\left\{ -\frac{i}{\hbar} \int_0^{T_m} d\tau \left[H_0 + \hbar \epsilon_I(t)(a^\dagger + a) + i\hbar \epsilon_Q(t)(a^\dagger - a) \right] \right\}, \quad \text{(A.1)}$$

where U_{log} is the desired quantum gate, T_m is the pulse duration, H_0 is the free Hamiltonian of qubits, ϵ_I and ϵ_Q are the real and imaginary part of the control pulse and a and $a^\dagger$ are the ladder operator for the transmon.

Now we will continue the discussion presenting some algorithms that numerically compute the microwave pulse.

We should start from an experimental fact. The implementation of the pulse is given by digital wave generator. Thereby, the pulse is obtained by a discrete set of N_{pulse} points with amplitude ϵ_k. Therefore, we should discretize eq. (A.1). We also suppose that between each discrete points the pulses amplitude is constant with value ϵ_k. In other words, the experimental control pulse is given by a sequence of N_{pulse} box functions with height ϵ_k with a small pulse duration Δt such that $T_m = N_{pulse} \Delta t$. An example of our pulse discretization is given in Fig. A.1.

Now, we should also consider the time order operator in eq. (A.1). So, the complete discrete real time evolution is given by

$$U(t) = U_{N_1} U_{N-2} ... U_1 U_0 \, , \quad \text{(A.2)}$$

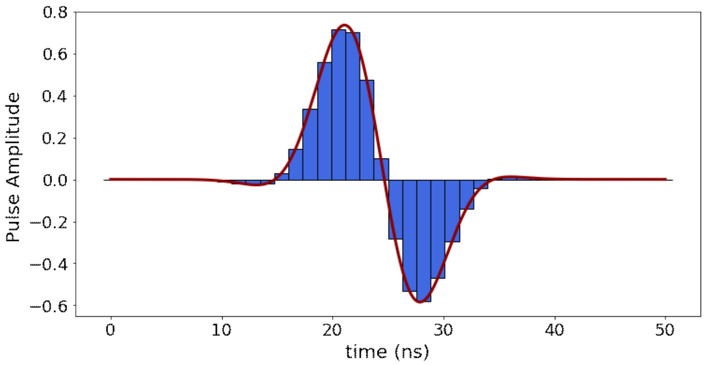


Figure A.1: Experimental control pulse as sequence of N small box amplitude. At each step the function ϵ_k is constant.

where U_j is the evolution at time $t = j\,\Delta t$ with constant control amplitude u_j. Specifically each small propagators is given by

$$U_j = \exp\left\{-\frac{i}{\hbar}\Delta\tau\left[H_{QPU} + \hbar\epsilon_I^j(a^\dagger + a) + i\hbar\epsilon_Q^j(a^\dagger - a)\right]\right\}, \tag{A.3}$$

where ϵ_Q^j and ϵ_I^j indicate the real and imaginary term of pulse ϵ_j.
Our original problem still remains unsolved. What is the pulse such that maximize the fidelity (see Sec. 3.2) between the actual real time evolution operator of the quantum processor and the desired quantum operator (i.e. what is described by our quantum algorithm).
In the literature [101], there are many different methods that compute the pulse. The principals are Krotov's algorithm [102, 103], the Chopped RAndom Basis (CRAB) [104, 105], Gradient Optimization of Analytic conTrols (GOAT) [106] and GRadient Ascent Pulse Engineering (GRAPE) [107, 108] algorithms. The method implemented in Ref. [9] and in this work is the GRAPE algorithm. Details can be found in the next section.

A.2 GRAPE

This section illustrates the basic aspect of the GRadient Ascent Pulse Engineering (GRAPE) algorithm to tailor the pulse for implementing a generic N-level quantum gate.
We want to maximize the fidelity between the evolution operator

$$U(t) = U_{N_1}\,U_{N-2}...U_1U_0 \tag{A.4}$$

and the logical operator U_{log}.
We start from the initial density matrix of qubits ρ_0 and we want compute the

pulse such that after some time t the final density matrix $\rho(t)$ of the physical qubits is given by $C = U_{log}^{\dagger}\rho_0 U_{log}$, where C indicates the target density matrix. The used QuTip function that implements GRAPE algorithm is based on a different fidelity definition. Specifically, the fidelity Φ is given by

$$\Phi = \langle C|\rho(t)\rangle = Tr[C^{\dagger}\,\rho(t)]\,. \tag{A.5}$$

The only difference between the definition of fidelity F presented in Sec. 2.5 regards the global phase. Φ depends on the global phase, instead, one that was defined in Sec. 2.5, $F = \|\Phi\|^2$, does not.

The following demonstration of the GRAPE algorithm is based on Ref. [107]. Substituting equation (A.4), the fidelity between the experimental operator and the logical one would be given by:

$$\Phi = \langle C|\rho(t)\rangle = \left\langle C \middle| U_{N_1} \ldots U_0 \rho_0 U_0^{\dagger} \ldots U_{N_1}^{\dagger} \right\rangle\,. \tag{A.6}$$

Looking closer the fidelity definition, we have the trace operation that is invariant under cyclic permutation of the factors. Thereby, we can rewrite eq. (A.6) as follows:

$$\Phi = \left\langle U_{N_1} \ldots U_{j+1} C U_{j+1}^{\dagger} \ldots U_{N_1}^{\dagger} \middle| U_j \ldots U_0 \rho_0 U_0^{\dagger} \ldots U_j^{\dagger} \right\rangle \tag{A.7}$$

where j is index belonged to $\{0, N\}$. We can notice the last result is rewritten in term of the density matrix at time $t_j = j\,\Delta\,t$,

$$\rho(t_j) = U_j \ldots U_0 \rho_0 U_0^{\dagger} \ldots U_j^{\dagger} \tag{A.8}$$

and a density matrix obtained by a backward propagation from the target C,

$$\lambda_j = U_{N_1} \ldots U_{j+1} C U_{j+1}^{\dagger} \ldots U_{N_1}^{\dagger}\,. \tag{A.9}$$

As we said, our goal is to find a pulse such that it maximizes the fidelity. To do that, it is convenient to look at the change of propagator due to a small change of the pulse for the time t_k. Varying the k-pulse as $\epsilon_k^p = \epsilon_k + \delta\epsilon_k$, the k-real evolution operator would be

$$U_k^p = U_k(1 - i\Delta t \delta\epsilon_k H_c) \tag{A.10}$$

or

$$U_k^p = -i\Delta t \delta\epsilon_k \bar{H}_k U_k \qquad \bar{H}_k = \int_0^{\Delta t} U_k(\tau) H_c U_k(-\tau) d\tau\,. \tag{A.11}$$

This last definition follows from

$$\frac{d}{dx} e^{A + x\,B} = e^A \int_0^{\Delta t} e^{A\tau} B e^{-A\tau} d\tau\,. \tag{A.12}$$

Choosing a small time step Δt, we can substitute to $\bar{H}_k \sim H_c$.

Hence, at first order we have the variation of the fidelity due to $\delta\epsilon_k$ is given by:

$$\frac{\partial\Phi}{\partial\epsilon_k} = -\left\langle \lambda_j | i\Delta t\,[H_c, \rho_j] \right\rangle\,. \tag{A.13}$$

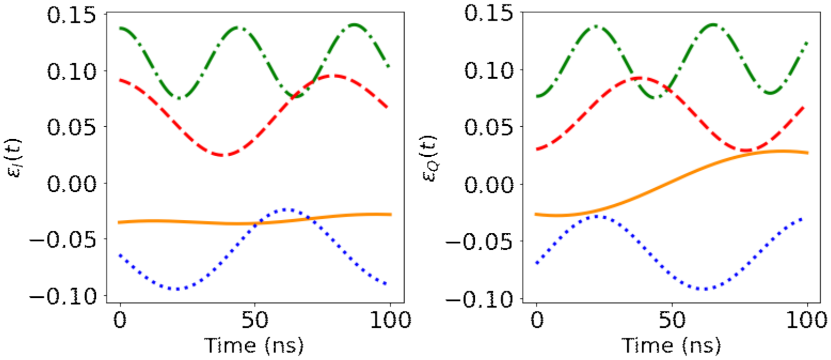


Figure A.2: Different experimental pulses for the same gate, the Hadamard gate. Left panel real part, right panel the imaginary part. Tab. 4.5 describes the parameters used for modelizing the quantum processor device

To maximize the fidelity using the steepest descent algorithm, one has to choose:

$$\epsilon_j = \epsilon_j + \gamma \frac{\partial \Phi}{\partial \epsilon_k} = \epsilon_j - \gamma \left\langle \lambda_j \middle| i\Delta t \left[H_c, \rho_j \right] \right\rangle \ . \tag{A.14}$$

where ϵ is a parameter of the steepest descent method.
All the discussed theory is the root of the GRAPE method.
We can recap the steps for the GRAPE method:

1. Starting from an initial control $\epsilon(t)$ and initial density matrix ρ_0
2. Compute the forward density matrix $\rho(t_j)$ for all $j \leq N$
3. Compute the backward density matrix λ_j for all $j \leq N$
4. Evaluate the fidelity derivatives $\frac{\partial \Phi}{\partial \epsilon_k}$ and update the control amplitudes ϵ_j for all $j \leq N$.
5. Go to step 2 until the final error fidelity is below a chosen threshold value.

The initial control $\{\epsilon_k\}$ can be chosen randomly. However, an educated guess may lead to a quick convergence.
For a pulse optimization starting from two different initial control pulses, one can get two very different final pulses that give the same correct quantum gate. Indeed, pulses obtained by optimization are not unique. This fact is due to lots of minima in fidelity error manifold. Different pulses correspond to different minima reached in their optimization. Fig. A.2 shows different pulses implementing the same Hadamard gate with very high fidelity.

Appendix B

Optimal control for parametric Hamiltonian

Optimal control techniques that tailor the experimental control pulse sequence to generate desired quantum gates were discussed in Sec. 3.2 and App. A.2. However, the substantial amount of classical computing required to obtain customized gates overgrows quickly (in many simulations exponentially) with the number of states.

Ref. [109] proposes two different methods to reconstruct the pulse from a limited set of pulses.

The first interpolating method proposed in Ref. [109] considers cases for which the control pulses evaluated from the optimization procedure are relatively simple. They can be directly fitted with polynomial functions. However, in general cases, the pulses have shapes that cannot be easily interpolated by elementary functions since they could contain multiple frequency components.

Therefore, Ref. [109] generalizes the interpolating method. An easy solution is to expand the control signals using a basis set. The interpolation becomes the optimization of the coefficients before the basis function. Generally, we deal with periodic pulses. Therefore, a useful functions class for the basis set is the Fourier basis. We Fourier-transform the control signal obtaining a function with peaks in the difference between qubit energies (it follows what we discussed in Ch. 4). Then, we interpolate the spectra curves as a function of frequency.

This method is very general because we expect the control pulses' spectrum to be a function of peaks. These peaks are centered on the difference of energies between the transmon levels. Studying more complicated parametric systems, we can avoid evaluating the needed pulse at each time step using this algorithm. It speeds up the calculation time.

Pulse interpolation of the spin Hamiltonian of two neutrons

We want to use the Fourier interpolation method to reduce the calculation time of the microwave pulses of the real time evolution operator of two interacting-spin neutrons as a function of the three-dimensional relative position.

In Fig. B.1 the real and imaginary parts of different pulses are presented as a function of relative positions of the two neutrons. The relative position is written in spherical coordinates, $[r, \theta, \phi]$, where r indicates the radial component, θ and ϕ are the polar and the azimuthal angles, respectively. The time step of the propagators is $\Delta t = 0.01$ MeV^{-1}. For the map between spin states and transmon levels, we use one presented in Sec. 4.2.

Observing the obtained pulses, they are difficult to interpolate in real time. Instead, Fig. B.2 shows the real part of the Fourier Transform of the real component of the pulse ($\epsilon_I(t)$). The spectrum of pulses is described by peak function. Interpolating using a three-dimensional linear fit the spectra, we obtain the black line as result for the relative position $[r, \theta, \phi] = [1.5, 0.5, 0.2]$.

Computing the inverse Fourier transform of the black line, we obtain its real time amplitude. This result is shown in Fig. B.1 with the black lines. The final fidelity with the desired real time propagator is 0.999921.

Tab. B.1 presents some fidelities results between propagators obtained from the interpolating pulses and the theoretical ones as a function of the relative position of two neutrons. All the obtained fidelities are very close to 1 with an error fidelity below 10^{-4}.

r (fm)	θ	ϕ	Fidelity
1.7831	-0.5802	0.6660	0.999972
2.1440	-1.5496	1.2073	0.999998
1.0371	-1.0317	0.7711	0.999923
2.4817	-0.8455	0.9771	0.999997
2.0713	-0.2563	3.0367	0.999981
0.5452	0.8381	1.5885	0.999961

Table B.1: Fidelity between the exact propagator and one obtained from the interpolated pulse for two spin neutrons as function of distance in spherical coordinates

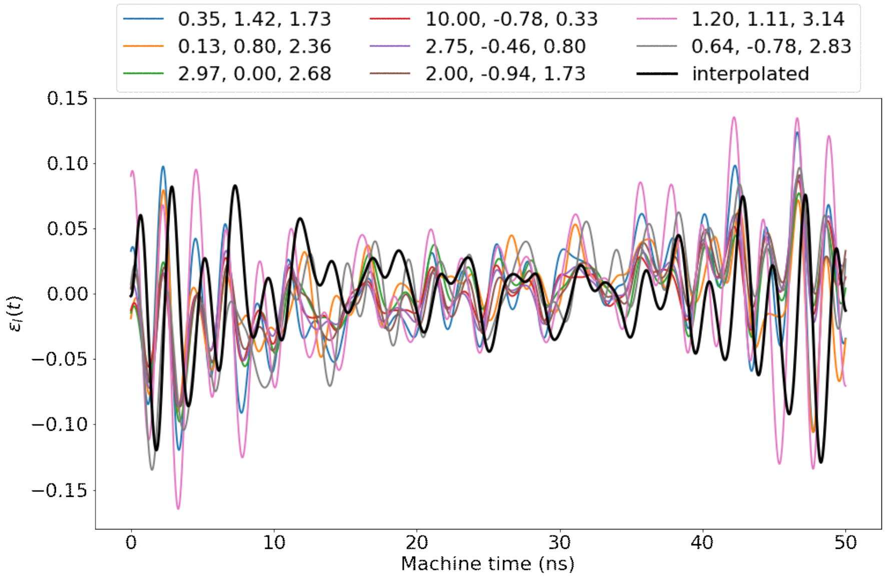


Figure B.1: Real part of pulses for the real time spin propagators of two neutrons for different relative positions. The black line represent the interpolated result for $[r, \theta, \phi] = [1.5,\ 0.5,\ 0.2]$.

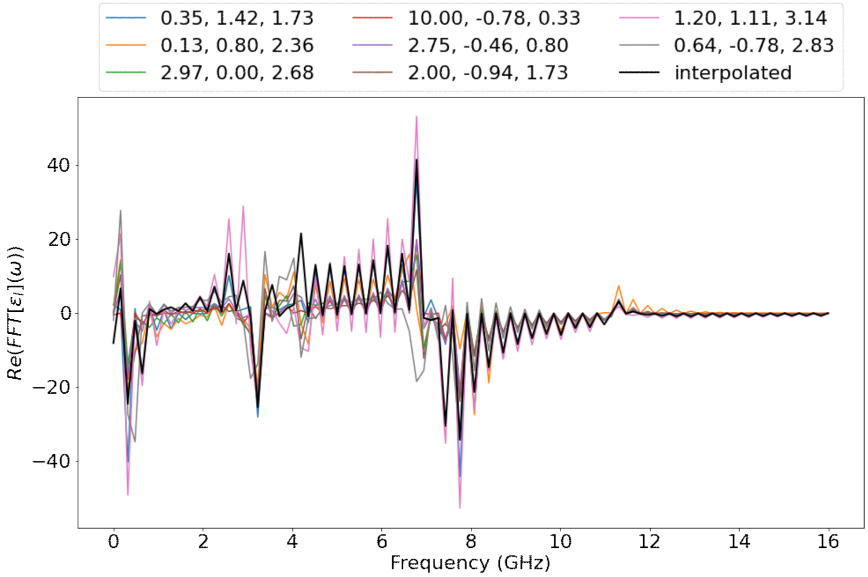


Figure B.2: Real part of the Fourier Transform of real part of pulses of the real time spin propagators of two neutrons for different relative positions. The black line represent the interpolation for $[r, \theta, \phi] = [1.5,\ 0.5,\ 0.2]$

Appendix C

Simulated quantum system

This appendix will illustrate the Hamiltonians of the quantum systems used in this work. We will start from the Hydrogen atom Hamiltonian expended using a Gaussian basis set. Then, we will move to the Chiral Effective Field Theory at Leading Order.

C.1 Hydrogen atom in STO-NG basis set

The well-known Hamiltonian that describes the Hydrogen atom is given by:

$$H = -\frac{\hbar^2}{2m}\nabla - \frac{e^2}{r}\,, \tag{C.1}$$

where r is the relative distance between the proton and the electron, m is the reduced mass of proton and electron, e is electron electric charge and ∇ is the Laplace operator.
If we expand the Schrodinger equation in the radial coordinate and spherical harmonics, $|\psi\rangle = \sum_{l,m} \psi_l(r)\, Y_{lm}(\theta, \phi)$, we obtain:

$$H_l\, \psi_l(r) \;=\; \left(-\frac{\hbar^2}{2\,m}\frac{d^2}{dr^2} - \frac{\hbar^2}{22\,m}\frac{2}{r}\frac{d}{dr} + \frac{\hbar^2\, l\,(l+1)}{2\,m\,r^2} - \frac{e^2}{r}\right)\psi_l(r) \;=\; E_l\,\psi_l(r)\,. \tag{C.2}$$

There are well-known ab-initio methods that are applied to compute molecular orbitals of large atomic systems. One of those apply linear combination of atomic orbitals using a minimal basis set of Slater-type atomic orbitals (STO).

For example, for light molecules only $1s$, $2s$ and $1p$ orbitals are used:

$$\begin{aligned}\phi_{1s} &= \left(\frac{c_{1s}^3}{\pi}\right)^{\frac{1}{2}} \exp\left(-c_{1s} r\right) \; ; \\ \phi_{2s} &= \left(\frac{c_{2s}^5}{3\pi}\right)^{\frac{1}{2}} \; r \exp\left(-c_{2s} r\right) \; ; \\ \phi_{1p} &= \left(\frac{c_{1p}^5}{\pi}\right)^{\frac{1}{2}} \; r \exp\left(-c_{1p} r\right) \cos(\theta) \, . \end{aligned} \tag{C.3}$$

The coefficients c_{1s}, c_{2s} and c_{1p} are computed through a variational calculations. Full Slater calculations for the large system are intensively time-consuming because of two-electron integrals. A simple solution is to replace the Slater-orbitals ($e^{-\xi r}$) with a Gaussian orbitals ($e^{-\xi r^2}$). This was proposed by J. Pope [110, 111]. It reduces the time-consuming integrals because of the more quickly fall of Gaussian term in the limit of r infinity and the possibility to compute the integrals analytically. This new basis set is called STO-NG where N indicates the number of Gaussian applied in the Hamiltonian expansion. Therefore, our basis for the $1s$ and $1p$ expansions will be:

$$\begin{aligned}\phi_{1s} &= \left(\frac{2\, c_{1s}}{\pi}\right)^{\frac{3}{2}} \exp\left(-c_{1s} r^2\right) \; ; \\ \phi_{1p} &= \left(\frac{128\, 3 c_{1p}^5}{\pi^3}\right)^{\frac{1}{4}} \; r \exp\left(-c_{1s} r^2\right) \cos(\theta) \, . \end{aligned} \tag{C.4}$$

The coefficients of the Gaussian exponential are computed through variational calculations as we do in the Slater orbitals calculations. In Tab. C.1 the well-known coefficients of $STO-2G$,..,$STO-4G$ basis set are shown [112]. Using these values, one can obtain the results for Hydrogen ground energies shown in Tab. C.2, where we used the atomic units ($m = \hbar = e = 1$). In these units, the exact ground energy of the Hydrogen atom is -0.5 Hartree (-13.6 eV).

The ground energies of Tab. C.2 are computed solving the so-called generalized eigenvalue problem because the $STO-NG$ basis sets are not orthogonal. The generalized eigenvalue equation is given by

$$H_l \, \psi = E \, S \, \psi \, , \tag{C.5}$$

where S describes the overlap matrix and it is given by $S_{ij} = \langle \psi_i | \psi_j \rangle$ with $\psi_i, \psi_j \in STO-NG$ and H_l was given by eq. (C.2). Eq. (C.5) is called Generalized Eigenvalue Equation.

In this work we will use mostly the $STO-2G$ basis set. The two elements forming this basis set are two Gaussians describing only the $1\,s$ level of Hydrogen atom with angular momentum $l = 0$.

The following subsection (Sec. C.1.1) shows how the algorithm works to solve eq. (C.5).

coefficients	$STO-2G$	$STO-3G$	$STO-4G$
c_{1s}	0.151623 0.851819	0.109818 0.45771 2.22766	0.00880187 0.265204 0.954620 5.21686
c_{1p}	0.0974545 0.384244	0.0751386 0.0231031 0.994203	0.0544949 0.127920 0.502989 2.32350

Table C.1: STO-NG coefficients. Taken from Ref. [112]

	$STO-2G$	$STO-3G$	$STO-4G$	$STO-5G$
Ground energy (Hartree^{-1})	-0.48199	-0.49491	-0.49848	-0.49951

Table C.2: Hydrogen ground energy for different $STO-NG$ basis sets. The exact ground energy is -0.5 Hartree^{-1}. Taken from Ref. [112]

C.1.1 Method for solving the Generalized Eigenvalue Equation

Here, we will show a rigorous method that allows us to solve the generalized eigenvalue equation [113],

$$A\, u_k = \lambda_k\, B\, u_k \,, \tag{C.6}$$

where in our case A is the Hamiltonian and B is the overlap matrix.
We start from the solution of the eigenvalue problem of B:

$$B\, V = D_B\, V \,, \tag{C.7}$$

where V and D_B indicate the eigenvector and eigenvalues matrices, respectively. From linear algebraic properties, we have:

$$V^{-1}\, B\, V = D_B \qquad \text{with} \qquad V^{-1}\, V = \mathbb{1} \text{ and } V^{\dagger} = V^{-1} \,. \tag{C.8}$$

If we multiply $D_B^{-1/2}$ from left and right hand sides the first condition, we get:

$$D_B^{-\frac{1}{2}}\,(V^{-1}BV)\,D_B^{-\frac{1}{2}} = D_B^{-\frac{1}{2}}\,D_B\,D_B^{-\frac{1}{2}} = 1\,. \tag{C.9}$$

If we redefine $\phi_B = V\,D_B^{-\frac{1}{2}}$, we get:

$$\phi_B^{-1}\,B\,\phi_B = \mathbb{1}\,. \tag{C.10}$$

Now, if we can define A_B as:

$$A_B = \phi_B^\dagger\,A\,\phi_B\,, \tag{C.11}$$

with its eigenvalue Λ and eigenvector ϕ_A. Therefore, we have

$$A_B\,\phi_A = \Lambda_A\,\phi_A\,, \tag{C.12}$$

Rewriting eq. (C.12) for λ_A, we get

$$\Lambda_A = \phi_A^\dagger\,A_B\,\phi_A = \phi_A^\dagger\,\phi_B^\dagger\,A\,\phi_B\,\phi_A = \phi_A^\dagger\,D_B^{-1/2}\,V_B^\dagger\,A\,V_B\,D_B^{-1/2}\,\phi_B\,\phi_A = \Phi^\dagger A\,\Phi\,, \tag{C.13}$$

where $\Phi = V_B D_B^{-1/2}\phi_A = \phi_B\phi_A$. We have just proved that Φ diagonalizes A. The last step to do is to prove that $\Lambda_A = \Lambda$. First, we must prove that Φ diagonalizes B as well:

$$\Phi^\dagger B\Phi = (\phi_A^\dagger D_B^{-1/2} V_B^\dagger) B (V_B D_B^{-1/2}\phi_A) = \phi_A^\dagger D_B^{-1/2} D_B D_B^{-1/2}\,\phi_A = \phi_A^\dagger\phi_A = \mathbb{1}\,, \tag{C.14}$$

where we used that $V_B^\dagger\,B\,V_B\; = \;D_B$.
From this relation, we have:

$$\Lambda_A \;=\; \mathbb{1}\,\Lambda_A \;=\; (\Phi^\dagger\,B\,\Phi)\,\Lambda_A \;=\Phi^\dagger\,B\,\Phi\,\Lambda_A \;=\; \Phi^\dagger\,A\,\Phi \tag{C.15}$$

where the last step we used the result obtained in eq. (C.13) for writing $\Lambda_A = \Phi^\dagger\,A\,\Phi$.
Simplifying $\Phi^\dagger$ in eq. (C.15), we obtain

$$B\,\Phi\,\Lambda_A \;=\; A\Phi\,. \tag{C.16}$$

The obtained final result of eq. (C.16) is our original generalized eigenvalue problem, where $\Lambda_A = \Lambda$.
Recapping the whole presented method in a bullet point list, to compute the eigenvalues and eigenvectors from a generalized eigenvalue equation we should

- Compute the Hamiltonian H and the overlap matrix S. We will have $H\,U \;=\; E\,S\,U$.
- Diagonalize S, $V_B S V_B = D_B$.
- Define $\Phi_B = V_B D_B^{-\frac{1}{2}}$.

- Compute $A = \Phi_B^\dagger H \Phi_B$.
- The eigenvalues E are obtained from the eigenvalue problem of A, $A\phi_A = E\phi_A$.
- The eigenvector U is given by $U = \phi_B \phi_A = V_B D_B^{-\frac{1}{2}} \phi_A$.

In this work, when we employ Hamiltonians expanded in a non-orthogonal basis set, we will apply the presented method. For example, when we compute the real or imaginary time propagator, its Hamiltonian will be obtained from eq. C.12 (or equivalently, from the fourth step of the bullet point list).

C.2 Nucleons systems at the Leading order in the Chiral Expansion

As discussed in Ch. 1, the QCD theory cannot be applied for today's nuclear structure physics due to the high computational time-consuming. Luckily, another efficient approach helps us in our calculations, the Effective Field Theory. As we discussed in Ch. 1, the first step in an EFT method is to identify the soft and hard scales. We know that there is a gap between the mesons. The lightest meson (π) has a mass of order 140 MeV, and the following lightest mesons (ρ and ω) have masses around 770 MeV and 782 MeV. Hence, we can identify our soft scale Q of EFT as pions mass m_π and the hard scale Λ as the mass of ρ. Furthermore, in an EFT approach, one has to write the most general Lagrangian that is consistent with the real and broken symmetries of the underlying theory (in our case QCD). In the massless quarks limit, the QCD Lagrangian has chiral symmetry. It describes that massless quarks right- and left-handed components do not mix. In particular, it is generated by $SU(2)\,\mathrm{x}\,SU(2)$.

We know that chiral symmetry is broken. From its axial component, one would expect for any hadron of positive parity a degenerate hadron state of negative parity and vice versa, but this is not observed in nature. It causes a spontaneous symmetry breaking, and the pions are identified as the Goldstone bosons.

So, our nuclear EFT theory would be an expansion in terms of powers of $\frac{Q}{\Lambda}$ and its Lagrangian has to be consistent with the chiral symmetry breaking. Hence, the Chiral Perturbation Theory (CHPT) is an effective field of QCD, which was formulated by Weinberg [114] and developed into a systematic tool for analyzing low-energy hadronic observables by Ref. [115, 116]. Refs. [11, 12, 13] show accurately the theory of chiral EFT.

We are interested in the chiral EFT that describes the interaction of 2 nucleons. Weinberg in Ref. [117, 118, 119] shows that the power counting of irreducible diagrams involving A nucleons. This formula is simplified in the case of interaction between two nucleons in:

$$\nu = 2\,L + \sum_i \Delta_i \tag{C.17}$$

with $\Delta = d + \frac{n_i}{2} - 2$ where d indicates the number of derivatives or pion mass insertions and n the number of nucleon field operators. ν is called hierarchy.

Fig. C.1 shows the irreducible Feynman graphs of two-body, three-body and four-body forces for the lowest orders.

The lowest order of EFT, known as leading order (LO), is when $\nu = 0$. At this level, the NN interaction is described by two contact terms and one-pion exchange potential. Its Feynman graphs are shown in Fig. C.2.

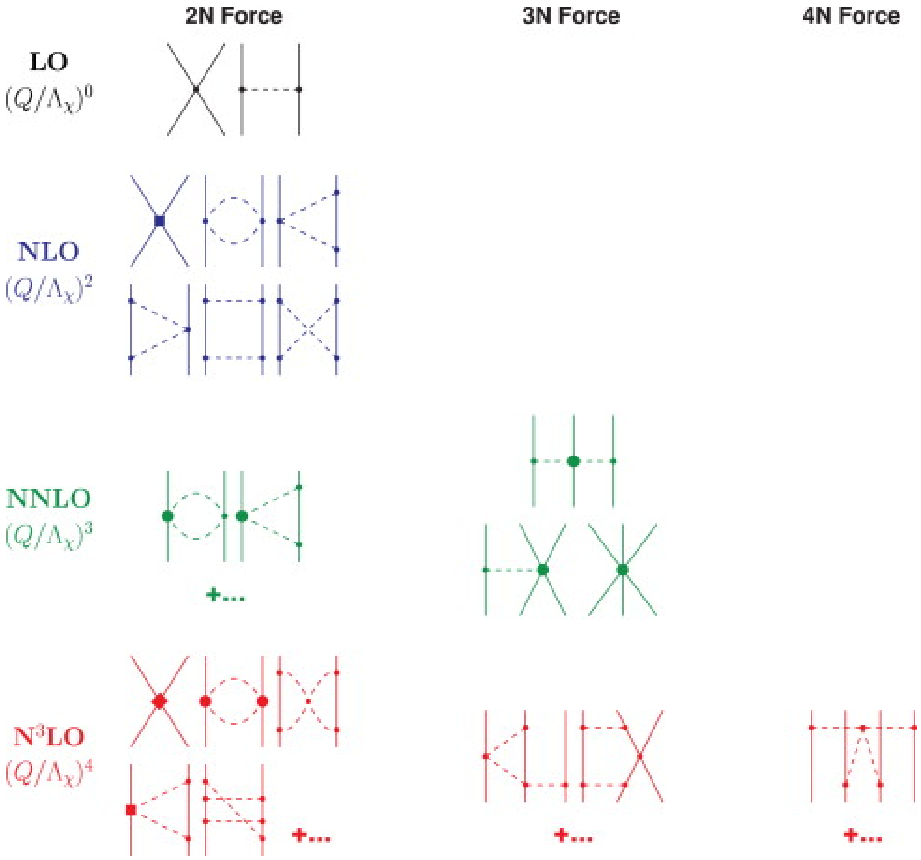


Figure C.1: Hierarchy of nuclear forces in ChPT. Solid lines represent nucleons and dashed lines pions. Small dots, large solid dots, solid squares, and solid diamonds denote vertices of index $\Delta = 0, 1, 3, 4$, respectively. Taken from Ref. [11]

The LO potential is a crude approximation to the two-nucleon force but accounts already for some crucial features. It provides the tensor force and describes quite well the deuteron ground energy. Moreover, it explains NN scattering in peripheral partial waves of very high orbital angular momentum. In this work, we will use just the potential obtained from LO expansion.

Of course, by increasing the interaction hierarchy, our nuclear force will improve and become more realistic.

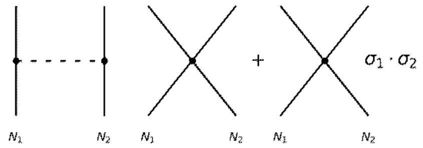


Figure C.2: Feynman diagrams for the Chiral EFT at Leading Order

At the lowest order, expanding the effective Lagrangian in powers of the pion fields, one gets the one-pion exchange and the contact one (Fig. C.2). In the Center of Mass frame, this yields the following potential in momentum space:

$$V(q) = -\frac{g_a^2}{4\,F_\pi^2}\frac{\sigma_1\cdot\vec{q}\sigma_1\cdot\vec{q}}{\vec{q}^2+M_\pi^2}\tau_1\cdot\tau_2 + c_s + c_t\sigma_1\cdot\sigma_2\,. \tag{C.18}$$

where we followed Ref. [118] for the contact terms. Indeed, we should have four contact terms involving the operators $\mathbb{1}, \sigma\cdot\sigma, \tau\cdot\tau, \sigma\cdot\sigma\tau\cdot\tau$, but, for the Fierz identities [120] we have only two terms, involving the operators $\mathbb{1}$ and $\sigma\cdot\sigma$.
In eq. (C.18), σ and τ are the spin and isospin Pauli matrix respectively, $\vec{q}$ is the transfer momentum $\vec{q} = \vec{p_f} - \vec{p_i}$ ($\vec{p_i}$ initial momentum, $\vec{p_f}$ final momentum). Moreover, M_pi refers to the averaging pion mass (eq. (C.19)) and F_π and g_a indicates the pion decay and nucleon coupling constant respectively. c_s and c_t are the running coupling constant that will be fitted with some experimental data (a further discussion will be done).
The average of pion mass is given by:

$$m_\pi = \frac{1}{3}\left(m_{\pi^+} + m_{\pi^-} + m_{\pi^0}\right)\,. \tag{C.19}$$

We are interested in having the nucleon-nucleon potential in the spatial coordinates. Therefore, we compute the inverse Fourier transform of eq. (C.18). Explicitly, we get:

$$V(r) = \int\int\frac{d\vec{q}}{(2\pi)^3}e^{-i\,\vec{q}\,r}V(q) = -\frac{g_a^2}{4\,F_\pi^2}\left(\int\int\frac{d\vec{q}}{(2\pi)^3}\frac{\sigma_1\cdot q\,\sigma_1\cdot q}{\vec{q}^2+M_\pi^2}\tau_1\cdot\tau_2 e^{-i\,\vec{q}\,r}\right) + c_s\delta(\vec{r}) + c_t\sigma_1\cdot\sigma_2\delta(\vec{r})\,, \tag{C.20}$$

where we have used $\int\int\frac{d\vec{q}}{(2\pi)^3}e^{-i\vec{q}\cdot\vec{r}} = \delta(\vec{r})$.
We must compute the following integral:

$$\frac{g_a^2}{4\,F_\pi^2}\left(\int\int\frac{d\vec{q}}{(2\pi)^3}\frac{\sigma_1\cdot\vec{q}\sigma_1\cdot\vec{q}}{\vec{q}^2+M_\pi^2}\,e^{-i\,\vec{q}\cdot\vec{r}}\right) = \frac{g_a^2}{4\,F_\pi^2}\tau_1\cdot\tau_2\left(\sigma_1\cdot\nabla\,\sigma_1\cdot\nabla\right) \int\frac{d\vec{q}}{(2\pi)^3}e^{-i\vec{q}\cdot\vec{r}}\frac{1}{\vec{q}^2+M_\pi^2}\,, \tag{C.21}$$

where we have used the Fourier property

$$\int\int \frac{d\vec{q}}{(2\pi)^3} f(q)\cdot q e^{-iqr} = \nabla\cdot\int\int \frac{d\vec{q}}{(2\pi)^3} f(q) e^{-iqr}\,. \tag{C.22}$$

The remaining integral is the usual Yukawa potential:

$$\int \frac{d\vec{k}}{(2\pi)^3} e^{-i\vec{k}\cdot\vec{r}} \frac{1}{\vec{k}^2 + M^2} = \frac{1}{4\pi}\frac{e^{-Mr}}{r} = \frac{1}{4\pi} Y_M(r) \tag{C.23}$$

where we have defined

$$Y_M(r) \;=\; \frac{e^{-Mr}}{r}\,. \tag{C.24}$$

Having a gradient in eq. (C.21), our calculation is easier if we apply

$$\left(-\nabla^2 + m_p i^2\right) Y_{m_\pi}(r) = 4\pi Y_{m_\pi}(r) \tag{C.25}$$

and

$$\sigma_1\cdot\nabla\sigma_2\cdot\nabla = \left[\sigma_1\cdot\nabla\sigma_2\cdot\nabla - \frac{1}{3}(\sigma_1\cdot\sigma_2)\nabla^2\right] + \frac{1}{3}(\sigma_1\cdot\sigma_2)\nabla^2\,. \tag{C.26}$$

Therefore, according to eqs. (C.25) and (C.26) we obtain:

$$\begin{aligned}\sigma_1\cdot\nabla\sigma_2\cdot\nabla Y_{m_\pi}(r) = &\left[\sigma_1\cdot\hat{r}\sigma_2\cdot\hat{r} - \frac{1}{3}\sigma_1\cdot\sigma_2\right](m_\pi^2 + \frac{3m_\pi}{r} + \frac{3}{r^2}Y_{m_\pi}(r) \\ &+ \frac{1}{3}(\sigma_1\cdot\sigma_2)\left(m_\pi^2\, Y_{m_\pi}(r) - 4\pi\delta(r)\right),\,,\end{aligned} \tag{C.27}$$

where $\hat{r} = \frac{\vec{r}}{|\vec{r}|}$. Defining usual tensor spin operator

$$\vec{S}_{12} = 3\sigma_1\cdot\hat{r}\sigma_2\cdot\hat{r} - \sigma_1\cdot\sigma_2 \tag{C.28}$$

and the usual tensor function:

$$T_{m_\pi}(r) = \left(1 + \frac{3}{m_\pi\, r} + 3\frac{1}{m_p i^2 r^2}\right) Y_{m_\pi}(r) \tag{C.29}$$

we can rewrite One-Pion Exchange Potential V_{OPEP} in the standard way:

$$V_{OPEP}(r) \;=\; \frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1\cdot\tau_2)\left[T_{m_\pi}(r)\vec{S}_{12} + \left(Y_{m_\pi}(r) - 4\pi\delta(r)\right)\sigma_1\cdot\sigma_2\right]\,. \tag{C.30}$$

Therefore, the Leading Order Chiral potential is given by:

$$\begin{aligned} V_{LO} \;&=\; V_{OPEP}(r) + c_s\delta(r) + c_t\sigma_1\cdot\sigma_2\delta(r) \\ =\; &\frac{1}{12\pi}\frac{g_a^2}{F_\pi^2} m_\pi(\tau_1\cdot\tau_2)\left[T_{m_\pi}(r)\vec{S}_{12} + Y_{m_\pi}(r)\sigma_1\cdot\sigma_2\right] + c_s\delta(r) + c_t\sigma_1\cdot\sigma_2\delta(r)\,. \end{aligned} \tag{C.31}$$

where we redefined c_s and c_s in order to include the term $c_s - -4\pi \frac{1}{12\pi} \frac{g_a^2}{F_\pi^2} m_\pi \sigma_1 \cdot \sigma_2$.

As discussed previously, in EFT, we must choose a soft and hard scale to expand the interesting dynamics. So, we have a cutoff in our EFT theory, and we must be sure all the found EFT theory is consistent with the cutoff. Hence, we write the delta as

$$\delta(r) = \alpha e^{-\frac{r^4}{R_0^4}} , \tag{C.32}$$

where R_0 is connected to our cutoff and $\alpha = \frac{1}{\pi\Gamma(3/4)R_0^3}$. To ensure the correct behavior of our EFT potential at a low distance, we multiply the long-term part (those terms from OPEP) with a regulator function $(1 - e^{-\frac{r^4}{R_0^4}})$. One can observe that the regulator by construction preserves the long-range parts of the interaction.

Explicitly, our final LO Chiral EFT is given by:

$$\begin{aligned} V_{LO} &= \frac{1}{12\pi} \frac{g_a^2}{F_\pi^2} m_\pi (\tau_1 \cdot \tau_2) \left[T_{m_\pi}(r)(1 - e^{-\frac{r^4}{R_0^4}})\vec{S}_{12} + (1 - e^{-\frac{r^4}{R_0^4}}) Y_{m_\pi}(r) \sigma_1 \cdot \sigma_2 \right] + \\ &+ c_s \alpha e^{-\frac{r^4}{R_0^4}} + c_t \sigma_1 \cdot \sigma_2 \alpha e^{-\frac{r^4}{R_0^4}} . \end{aligned} \tag{C.33}$$

The last thing to do is to fit our running coupling constant c_s and c_t with some data. We took the values of c_s and c_t from Ref. [121, 122] with a cutoff of $\Lambda = 400$ MeV and $R_0 = 1.2$ fm. In particular, in these reference c_s and c_t are fitted with neutron-proton phase shift of $^1S_0$ and $^3S_1$ channels. In Tab. C.3 are shown all the values of nuclear potential.

Fig. C.3 shows the radial component of each contribution of the LO potential, where:

$$V_{S_{12}} = \frac{1}{12\pi} \frac{g_a^2}{F_\pi^2} m_\pi T_{m_\pi}(r)(1 - e^{-\frac{r^4}{R_0^4}}) \tag{C.34}$$

$$V_{S_1 S_2} = \frac{1}{12\pi} \frac{g_a^2}{F_\pi^2} m_\pi (1 - e^{-\frac{r^4}{R_0^4}}) Y_{m_\pi}(r) \tag{C.35}$$

$$V_c = \alpha e^{-\frac{r^4}{R_0^4}} \tag{C.36}$$

Parameters of LO potential	values
c_s	-1.79693
c_t	+0.15442
g_a	-1.2756
F_π	92.4 MeV
m_π	134.9766 MeV

Table C.3: Our values of parameters of LO chiral potential with $\Lambda = 400$ MeV and $R_0 = 1.2$ fm

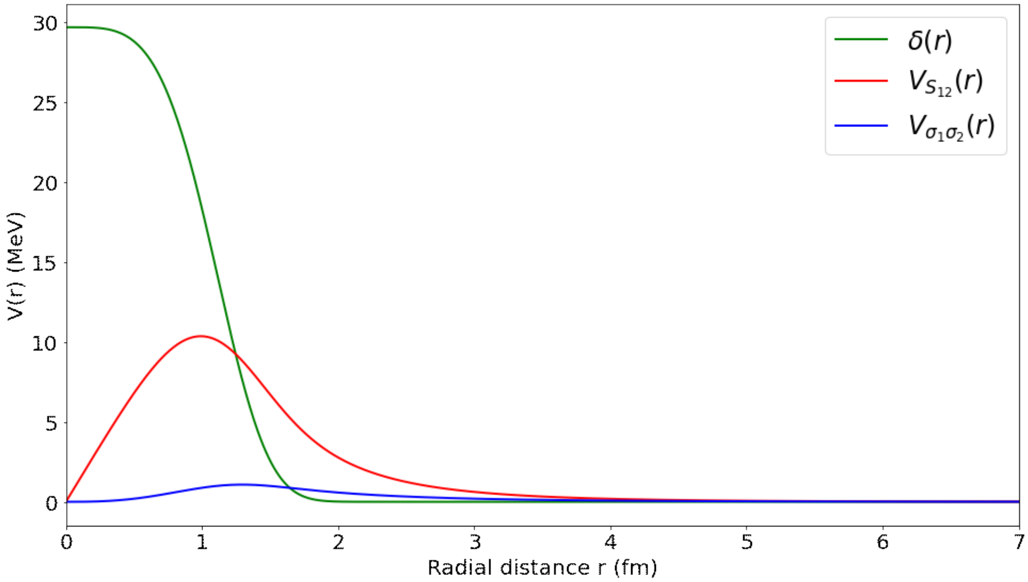


Figure C.3: Radial shape of each components of LO potential using the parameters of Tab. C.3

Appendix D

Test of State Tomography processes

Here, we will show some tests of our state tomography. We start from a two-dimensional grid. From this mesh, we build pure states of a single qubit. Specifically, the $x-$axis corresponds to the probability of measuring $|0\rangle$ and y-axis to the azimuth angle. Then, employing the Bloch-Redfield model (see Sec. 3.3) after some machine time t, we obtain the corresponding mixed state. The used two noise parameters T_1 and T_2 are $T_1 = 30$ μs and $T_2 = 7$ μs.

We applied the two state tomography processes. One is the standard, where we compute the mean value of X, Y and Z[1]; the other is presented in this work in Sec. 5.4. After implementing the tomography operation theoretically (without including errors), we sampled the probability with a number of shots $N_{shot} = 8000$. This sampling mimics the readout process. From the results of the tomography processes, we get the corresponding states; at the end, we computed the fidelity between them and the original pure states. Fig. D.1 shows the results for $t = 4$ μs and Fig. D.2 for $t = 15$ μs.

We can observe that the presented tomography reaches even more accurate fidelity to the original pure states on average. In particular, this is true when we are in the high contribution regime of dephasing, for instance, observing Fig. D.2. The time of this figure is greater than the dephasing time T_2. Therefore, we have a considerable contribution to this kind of noise source.

However, we should highlight that the presented state tomography suffers when the state is closer to $|1\rangle$ where we have the most contribution of the relaxation process. When we have a state close to $|1\rangle$, it is desirable to employ the standard state tomography process.

[1] We also normalize the density matrix to have $\mathrm{Tr}\left[\rho^2\right] = 1$

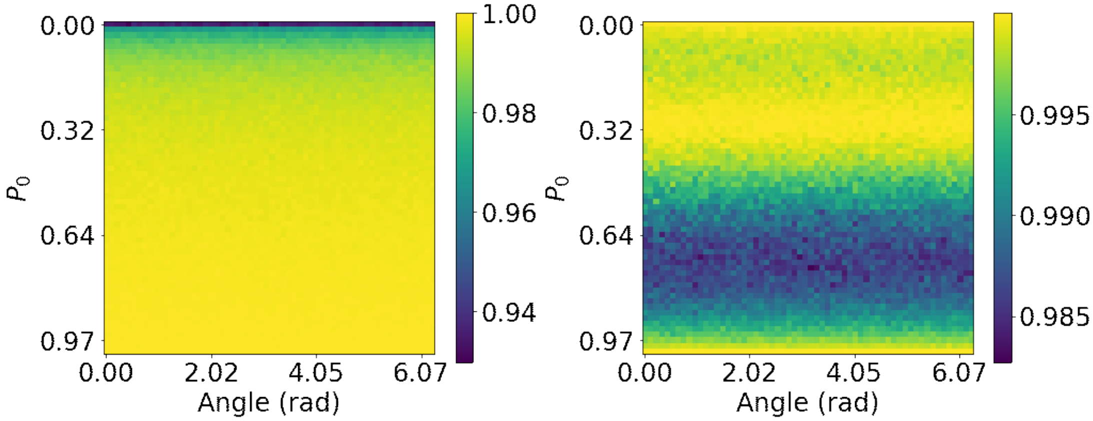


Figure D.1: Results for the fidelity, left ones with the state obtained from the presented state fidelity, right with the standard one. The x-axis represent the azimuth angle and y-axis the probability to measure the $|0\rangle$ state. The parameters are $t = 4$ μs, $T_1 = 30$ μs, $T_2 = 7$ μs, $N_{shot} = 8000$

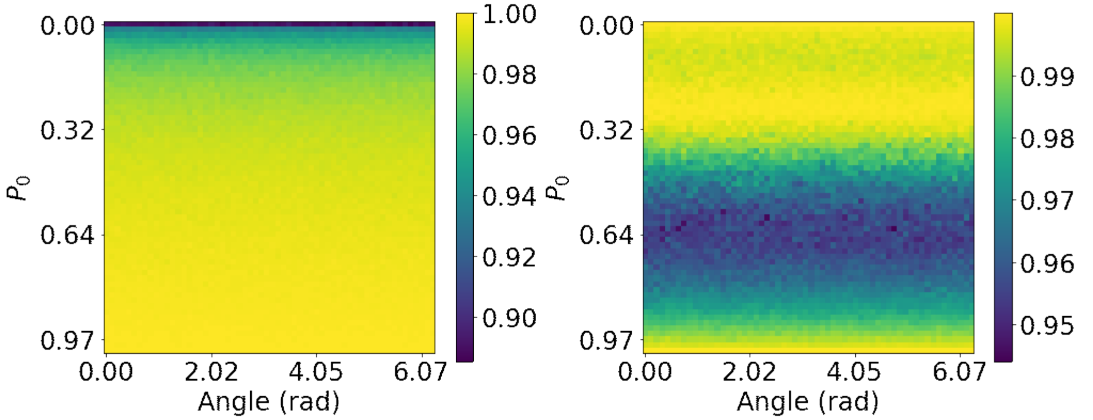


Figure D.2: Results for the fidelity, left ones with the state obtained from the presented state fidelity, right with the standard one. The x-axis represent the azimuth angle and y-axis the probability to measure the $|0\rangle$ state. The parameters are $t = 15$ μs, $T_1 = 30$ μs, $T_2 = 7$ μs, $N_{shot} = 8000$

Appendix E

Quantum circuits for AQT

This appendix will present some of the actual implemented quantum circuits in the AQT quantum processor for the different approaches used in Ch. 5.
Fig. E.1 shows the quantum circuit of the full optimized scheme. Fig. E.3 presents it with the implementations of a sequence of propagators at time step equal 8. Fig. E.3 shows the quantum circuit using a sequence of propagators applying a reinitializing gate. This gate is represented by the first sequence of gates ended to the first barrier (the vertical line).
We should observe the difference in depth of the quantum circuits. A deeper quantum circuit corresponds to a higher contribution of errors from the gate infidelity and quantum noise of processors.

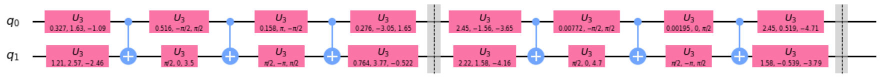


Figure E.1: Full-optimized quantum circuit

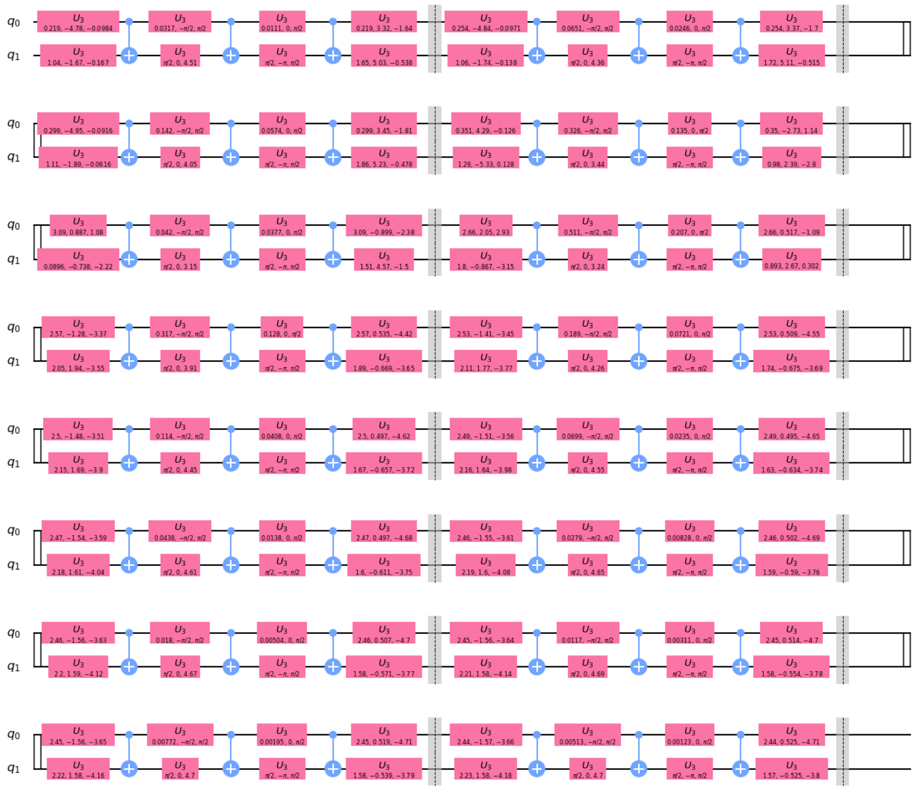


Figure E.2: Quantum circuit using a sequence of propagators for time step index $t = 8$

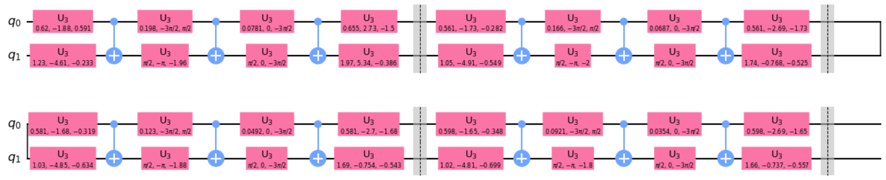


Figure E.3: Quantum circuit using a sequence of propagators with a reinitializing gate. $T_{reini} = 3$.

Appendix F

Trotter decomposition error for complex time propagation

Appendix F of Ref. [98] shows the error bound when we use the Trotter decomposition for we study the real- or imaginary-time evolution. Here, we will report its most important lemmas to demonstrate the Trotter errors.

Lemma F.0.1. *(Lemma F1) If $\lambda \in \mathbf{C}$ and $H_1, ..., H_L$ Hermitian operators, then:*

$$\left\| R_k \left(\prod_{j=1}^{L} \exp\{\lambda\, H_j\} \right) \right\| \leq R_k \left(\exp\left\{ \sum_{j=1}^{L} |\lambda| \|H_j\| \right\} \right) \tag{F.1}$$

where $R_k(f)$ is the remainder of the Taylor series expansion of f, $R_k(f) = \sum_{i=k+1}^{\infty} \frac{df^i}{d^i\,x} x^i$.

Lemma F.0.2. *(Lemma F2.) If $\lambda \in \mathbf{C}$, then*

$$|R_k(\exp\{\lambda\}| \leq \frac{|\lambda|^{k+1}}{(k+1)!} \exp\{|\lambda|\} \tag{F.2}$$

Lemma F.0.3. *(Preposition F3.) Let $r \in \mathbf{N}$ and $t \in \mathbf{R}$. Let H_1, ..., H_L be Hermitian operators and $\Lambda = \max_j \|H_j\|$. Then*

$$\left\| \exp\left\{ -it \sum_{j}^{L} H_j \right\} - \left(\prod_{j}^{L} \exp\left\{ -\frac{it}{r} H_j \right\} \right)^r \right\| \leq \frac{(L\Lambda t)^2}{r} \exp\left\{ \frac{L\Lambda |t|}{r} \right\} \tag{F.3}$$

Bibliography

[1] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–488, 1982.

[2] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[3] Ming Gong, Shiyu Wang, Chen Zha, Ming-Cheng Chen, He-Liang Huang, Yulin Wu, Qingling Zhu, Youwei Zhao, Shaowei Li, Shaojun Guo, et al. Quantum walks on a programmable two-dimensional 62-qubit superconducting processor. *Science*, 372(6545):948–952, 2021.

[4] John Preskill. Quantum computing and the entanglement frontier. *arXiv preprint arXiv:1203.5813*, 2012.

[5] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[6] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.

[7] Ibm quantum experience: https://quantum-computing.ibm.com/.

[8] Berkeley aqt website: https://aqt.lbl.gov.

[9] Eric T Holland, Kyle A Wendt, Konstantinos Kravvaris, Xian Wu, W Erich Ormand, Jonathan L DuBois, Sofia Quaglioni, and Francesco Pederiva. Optimal control for the quantum simulation of nuclear dynamics. *Physical Review A*, 101(6):062307, 2020.

[10] F. Turro, A. Roggero, V. Amitrano, P. Luchi, K. A. Wendt, J. L. Dubois, S. Quaglioni, and F. Pederiva. Imaginary-time propagation on a quantum chip. *Phys. Rev. A*, 105:022440, Feb 2022.

[11] Ruprecht Machleidt and David Rodrignez Entem. Chiral effective field theory and nuclear forces. *Physics Reports*, 503(1):1–75, 2011.

[12] Evgeny Epelbaum. Nuclear forces from chiral effective field theory: a primer. *arXiv preprint arXiv:1001.3229*, 2010.

[13] Evgeny Epelbaum, H-W Hammer, and Ulf-G Meißner. Modern theory of nuclear forces. *Reviews of Modern Physics*, 81(4):1773, 2009.

[14] Kate L Jones and Witold Nazarewicz. Designer nuclei—making atoms that barely exist. *The Physics Teacher*, 48(6):381–385, 2010.

[15] Peter Lepage. How to renormalize the schrodinger equation. *arXiv preprint arXiv:9706029v1*, 1997.

[16] Aleksandr Y Potekhin. The physics of neutron stars. *Physics-Uspekhi*, 53(12):1235, 2010.

[17] Sarang. Fusion in the sun.svg, 2016.

[18] Borb. Triple-alpha process.svg, 2006.

[19] Matthias Troyer and Uwe-Jens Wiese. Computational complexity and fundamental limitations to fermionic quantum monte carlo simulations. *Phys. Rev. Lett.*, 94:170201, 2005.

[20] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[21] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications, 1970.

[22] Francesco Pederiva, Alessandro Roggero, and Kevin E Schmidt. Variational and diffusion monte carlo approaches to the nuclear few-and many-body problem. In *An Advanced Course in Computational Nuclear Physics*, pages 401–476. Springer, 2017.

[23] David P DiVincenzo. The physical implementation of quantum computation. *Fortschritte der Physik: Progress of Physics*, 48(9-11):771–783, 2000.

[24] John Bird. *Engineering mathematics.* Routledge, 2014.

[25] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition.* Cambridge University Press, USA, 10th edition, 2011.

[26] Erhard Schmidt. Über die auflösung linearer gleichungen mit unendlich vielen unbekannten. In *Integralgleichungen und Gleichungen mit unendlich vielen Unbekannten*, page 190–233. Springer, 1989.

[27] Anirban Pathak. *Elements of quantum computation and quantum communication.* CRC Press Boca Raton, 2013.

[28] Andreas Ketterer. *Modular variables in quantum information.* PhD thesis, Sorbonne Paris Cité, 2016.

[29] Adriano Barenco, Charles H Bennett, Richard Cleve, David P DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical review A*, 52(5):3457, 1995.

[30] Aleksei Yur'evich Kitaev. Quantum computations: algorithms and error correction. *Uspekhi Matematicheskikh Nauk*, 52(6):53–112, 1997.

[31] Robert R. Solovay. Lie groups and quantum circuits. MSRI, 2000.

[32] Christopher M Dawson and Michael A Nielsen. The solovay-kitaev algorithm. *arXiv preprint quant-ph/0505030*, 2005.

[33] Armin Uhlmann. The "transition probability" in the state space of a$_*$-algebra. *Reports on Mathematical Physics*, 9(2):273–279, 1976.

[34] Benjamin Rowland and Jonathan A Jones. Implementing quantum logic gates with gradient ascent pulse engineering: principles and practicalities. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1976):4636–4650, 2012.

[35] Don Coppersmith. Modifications to the number field sieve. *Journal of Cryptology*, 6(3):169–180, 1993.

[36] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

[37] Xian Wu, Spencer L Tomarken, N Anders Petersson, Luis A Martinez, Yaniv J Rosen, and Jonathan L DuBois. High-fidelity software-defined quantum logic on a superconducting qudit. *Physical Review Letters*, 125(17):170502, 2020.

[38] Roger H Koch, David P DiVincenzo, and John Clarke. Model for 1/f flux noise in squids and qubits. *Physical review letters*, 98(26):267003, 2007.

[39] Brian David Josephson. Possible new effects in superconductive tunnelling. *Physics letters*, 1(7):251–253, 1962.

[40] Philip W Anderson and John M Rowell. Probable observation of the josephson superconducting tunneling effect. *Physical Review Letters*, 10(6):230, 1963.

[41] Philip Krantz, Morten Kjaergaard, Fei Yan, Terry P Orlando, Simon Gustavsson, and William D Oliver. A quantum engineer's guide to superconducting qubits. *Applied Physics Reviews*, 6(2):021318, 2019.

[42] Xiu Gu, Anton Frisk Kockum, Adam Miranowicz, Yu-xi Liu, and Franco Nori. Microwave photonics with superconducting quantum circuits. *Physics Reports*, 718:1–102, 2017.

[43] Sands M., R. Feynman, and R. B. Leighton. *The Feynman Lectures on Physics Vol. III.* Basic Books; New Millennium, 2015.

[44] Jens Koch, M Yu Terri, Jay Gambetta, Andrew A Houck, David I Schuster, Johannes Majer, Alexandre Blais, Michel H Devoret, Steven M Girvin, and Robert J Schoelkopf. Charge-insensitive qubit design derived from the cooper pair box. *Physical Review A*, 76(4):042319, 2007.

[45] Denis Vion, A Aassime, Audrey Cottet, Pl Joyez, H Pothier, C Urbina, Daniel Esteve, and Michel H Devoret. Manipulating the quantum state of an electrical circuit. *Science*, 296(5569):886–889, 2002.

[46] A Cottet, D Vion, A Aassime, P Joyez, D Esteve, and MH Devoret. Implementation of a combined charge-phase quantum bit in a superconducting circuit. *Physica C: Superconductivity*, 367(1-4):197–203, 2002.

[47] Patrice Bertet, Irinel Chiorescu, Guido Burkard, Kouichi Semba, CJPM Harmans, David P DiVincenzo, and JE Mooij. Dephasing of a superconducting qubit induced by photon noise. *Physical review letters*, 95(25):257002, 2005.

[48] F Yoshihara, K Harrabi, AO Niskanen, Y Nakamura, and Jaw Shen Tsai. Decoherence of flux qubits due to 1/f flux noise. *Physical review letters*, 97(16):167001, 2006.

[49] Fei Yan, Simon Gustavsson, Jonas Bylander, Xiaoyue Jin, Fumiki Yoshihara, David G Cory, Yasunobu Nakamura, Terry P Orlando, and William D Oliver. Rotating-frame relaxation as a noise spectrum analyser of a superconducting qubit undergoing driven evolution. *Nature communications*, 4(1):1–8, 2013.

[50] Simon Gustavsson, Jonas Bylander, Fei Yan, Pol Forn-Díaz, Vlad Bolkhovsky, Danielle Braje, George Fitch, Khalil Harrabi, Donna Lennon, Jovi Miloshi, et al. Driven dynamics and rotary echo of a qubit tunably coupled to a harmonic oscillator. *Physical review letters*, 108(17):170503, 2012.

[51] FG Paauw, A Fedorov, CJP M Harmans, and JE Mooij. Tuning the gap of a superconducting flux qubit. *Physical review letters*, 102(9):090501, 2009.

[52] Cesar Eduardo Chialvo. *The current-phase relation of graphene-based Josephson junctions.* University of Illinois at Urbana-Champaign, 2011.

[53] MJ Schwarz, J Goetz, Z Jiang, T Niemczyk, F Deppe, A Marx, and R Gross. Gradiometric flux qubits with a tunable gap. *New Journal of Physics*, 15(4):045001, 2013.

[54] JA Schreier, Andrew A Houck, Jens Koch, David I Schuster, BR Johnson, JM Chow, Jay M Gambetta, J Majer, L Frunzio, Michel H Devoret, et al. Suppressing charge noise decoherence in superconducting charge qubits. *Physical Review B*, 77(18):180502, 2008.

[55] Vladimir E Manucharyan, Jens Koch, Leonid I Glazman, and Michel H Devoret. Fluxonium: Single cooper-pair circuit free of charge offsets. *Science*, 326(5949):113–116, 2009.

[56] Ioan M Pop, Kurtis Geerlings, Gianluigi Catelani, Robert J Schoelkopf, Leonid I Glazman, and Michel H Devoret. Coherent suppression of electromagnetic dissipation due to superconducting quasiparticles. *Nature*, 508(7496):369–372, 2014.

[57] Rami Barends, Julian Kelly, Anthony Megrant, Daniel Sank, Evan Jeffrey, Yu Chen, Yi Yin, Ben Chiaro, Josh Mutus, Charles Neill, et al. Coherent josephson qubit suitable for scalable quantum integrated circuits. *Physical review letters*, 111(8):080502, 2013.

[58] Yu Chen, C Neill, Pedram Roushan, Nelson Leung, Michael Fang, Rami Barends, Julian Kelly, Brooks Campbell, Z Chen, Benjamin Chiaro, et al. Qubit architecture with high coherence and fast tunable coupling. *Physical review letters*, 113(22):220502, 2014.

[59] Michael R Geller, Emmanuel Donate, Yu Chen, Michael T Fang, Nelson Leung, Charles Neill, Pedram Roushan, and John M Martinis. Tunable coupler for superconducting xmon qubits: Perturbative nonlinear model. *Physical Review A*, 92(1):012320, 2015.

[60] Christopher Gerry, Peter Knight, and Peter L Knight. *Introductory quantum optics*. Cambridge university press, 2005.

[61] Roald K Wangsness and Felix Bloch. The dynamical theory of nuclear induction. *Physical Review*, 89(4):728, 1953.

[62] Alfred G Redfield. On the theory of relaxation processes. *IBM Journal of Research and Development*, 1(1):19–31, 1957.

[63] G Ithier, E Collin, P Joyez, PJ Meeson, Denis Vion, Daniel Esteve, F Chiarello, A Shnirman, Yu Makhlin, Josef Schriefl, et al. Decoherence in a superconducting quantum bit circuit. *Physical Review B*, 72(13):134519, 2005.

[64] Francesco Pederiva. Private communication.

[65] Clément Godfrin. *Quantum information processing using a molecular magnet single nuclear spin qudit*. PhD thesis, Institut Neél, 04 2017.

[66] Norman F Ramsey. A molecular beam resonance method with separated oscillating fields. *Physical Review*, 78(6):695, 1950.

[67] Erwin L Hahn. Spin echoes. *Physical review*, 80(4):580, 1950.

[68] Herman Y Carr and Edward M Purcell. Effects of diffusion on free precession in nuclear magnetic resonance experiments. *Physical review*, 94(3):630, 1954.

[69] Heinz-Peter Breuer, Francesco Petruccione, et al. *The theory of open quantum systems*. Oxford University Press on Demand, 2002.

[70] Edwin T Jaynes and Frederick W Cummings. Comparison of quantum and semiclassical radiation theories with application to the beam maser. *Proceedings of the IEEE*, 51(1):89–109, 1963.

[71] John R Schrieffer and Peter A Wolff. Relation between the anderson and kondo hamiltonians. *Physical Review*, 149(2):491, 1966.

[72] L. E. Franks. *Signal Theory*, chapter 4, page 82. Englewood Cliffs, N. J., Prentice Hall, 1969.

[73] M. Sipser. *Introduction to the Theory of Computation, 2nd edition*. Course Technology Ptr, 2006.

[74] J Robert Johansson, Paul D Nation, and Franco Nori. Qutip: An open-source python framework for the dynamics of open quantum systems. *Computer Physics Communications*, 183(8):1760–1772, 2012.

[75] Sofia Quaglioni. Private communication.

[76] Righetti qpus: https://www.rigetti.com.

[77] pyquil skd website: https://pyquil-docs.rigetti.com/en/stable/.

[78] Richard Phillips Feynman and Albert Roach Hibbs. *Quantum mechanics and path integrals*. International series in pure and applied physics. McGraw-Hill, New York, NY, 1965.

[79] Loup Verlet. Computer" experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. *Physical review*, 159(1):98, 1967.

[80] William C Swope, Hans C Andersen, Peter H Berens, and Kent R Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of chemical physics*, 76(1):637–649, 1982.

[81] Qiskit: https://qiskit.org/documentation.

[82] Matthew Otten, Cristian L Cortes, and Stephen K Gray. Noise-resilient quantum dynamics using symmetry-preserving ansatzes, 2019.

[83] YG Chen and JB Wang. Qcompiler: Quantum compilation with the csd method. *Computer Physics Communications*, 184(3):853–865, 2013.

[84] Michael Reck, Anton Zeilinger, Herbert J Bernstein, and Philip Bertani. Experimental realization of any discrete unitary operator. *Physical review letters*, 73(1):58, 1994.

[85] Chi-Kwong Li, Rebecca Roberts, and Xiaoyan Yin. Decomposition of unitary matrices and quantum gates. *International Journal of Quantum Information*, 11(01):1350015, 2013.

[86] Stanisław D Głazek and Kenneth G Wilson. Renormalization of hamiltonians. *Physical Review D*, 48(12):5863, 1993.

[87] Stanislaw D Glazek and Kenneth G Wilson. Perturbative renormalization group for hamiltonians. *Physical Review D*, 49(8):4214, 1994.

[88] Franz Wegner. Flow equations for hamiltonians. In *Advances in Solid State Physics 40*, pages 133–142. Springer, 2000.

[89] Zachary B Walters. A linear time quantum algorithm for 3sat using irreversible quantum operations, 2015.

[90] Hiroaki Terashima and Masahito Ueda. Nonunitary quantum circuit. *International Journal of Quantum Information*, 3(04):633–647, 2005.

[91] Lin Lin and Yu Tong. Near-optimal ground state preparation. *Quantum*, 4:372, 2020.

[92] Alessandro Roggero, Chenyi Gu, Alessandro Baroni, and Thomas Papenbrock. Preparation of excited states for nuclear dynamics on a quantum computer. *Physical Review C*, 102(6):064624, 2020.

[93] David B Kaplan, Natalie Klco, and Alessandro Roggero. Ground states via spectral combing on a quantum computer, 2017.

[94] Kenneth Choi, Dean Lee, Joey Bonitati, Zhengrong Qian, and Jacob Watkins. Rodeo algorithm for quantum computing. *Phys. Rev. Lett.*, 127:040505, Jul 2021.

[95] Gilles Brassard and Peter Hoyer. An exact quantum polynomial-time algorithm for simon's problem. In *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*, pages 12–23. IEEE, 1997.

[96] Lov K Grover. Quantum computers can search rapidly by using almost any transformation. *Physical Review Letters*, 80(19):4329, 1998.

[97] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

[98] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018.

[99] Tom M Apostol. *Calculus, Volume 1.* John Wiley & Sons, 1991.

[100] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 1991.

[101] D Stefanatos and E Paspalakis. A shortcut tour of quantum control methods for modern quantum technologies. *EPL (Europhysics Letters)*, 132(6):60001, 2021.

[102] Ronnie Kosloff, Stuart A Rice, Pier Gaspard, Sam Tersigni, and DJ Tannor. Wavepacket dancing: Achieving chemical selectivity by shaping light pulses. *Chemical Physics*, 139(1):201–220, 1989.

[103] Daniel M Reich, Mamadou Ndong, and Christiane P Koch. Monotonically convergent optimization in quantum control using krotov's method. *The Journal of chemical physics*, 136(10):104103, 2012.

[104] Patrick Doria, Tommaso Calarco, and Simone Montangero. Optimal control technique for many-body quantum dynamics. *Physical review letters*, 106(19):190501, 2011.

[105] Tommaso Caneva, Tommaso Calarco, and Simone Montangero. Chopped random-basis quantum optimization. *Physical Review A*, 84(2):022326, 2011.

[106] Shai Machnes, Elie Assémat, David Tannor, and Frank K Wilhelm. Tunable, flexible, and efficient optimization of control pulses for practical qubits. *Physical review letters*, 120(15):150401, 2018.

[107] Navin Khaneja, Timo Reiss, Cindie Kehlet, Thomas Schulte-Herbrüggen, and Steffen J Glaser. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. *Journal of magnetic resonance*, 172(2):296–305, 2005.

[108] Shai Machnes, Urgen Sander, Steffen J Glaser, Pierre de Fouquieres, Audrunas Gruslys, Sonia Schirmer, and Thomas Schulte-Herbrüggen. Comparing, optimizing, and benchmarking quantum-control algorithms in a unifying programming framework. *Physical Review A*, 84(2):022305, 2011.

[109] Piero Luchi, Francesco Turro, Valentina Amitrano, Francesco Pederiva, Xian Wu, Kyle Wendt, Jonathan L Dubois, and Sofia Quaglioni. Control optimization for parametric hamiltonians by pulse reconstruction, 2021.

[110] S Francis Boys. Electronic wave functions-i. a general method of calculation for the stationary states of any molecular system. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 200(1063):542–554, 1950.

[111] Robert F Stewart. Small gaussian expansions of slater-type orbitals. *The Journal of Chemical Physics*, 52(1):431–438, 1970.

[112] Warren J Hehre, Robert F Stewart, and John A Pople. self-consistent molecular-orbital methods. i. use of gaussian expansions of slater-type atomic orbitals. *The Journal of Chemical Physics*, 51(6):2657–2664, 1969.

[113] Benyamin Ghojogh, Fakhri Karray, and Mark Crowley. Eigenvalue and generalized eigenvalue problems: Tutorial. *arXiv preprint arXiv:1903.11240*, 2019.

[114] Steven Weinberg. Phenomenological lagrangians. *Physica a*, 96(1-2):327–340, 1979.

[115] Jürg Gasser and Heinrich Leutwyler. Chiral perturbation theory to one loop. *Annals of Physics*, 158(1):142–210, 1984.

[116] Juerg Gasser and Heinrich Leutwyler. Chiral perturbation theory: expansions in the mass of the strange quark. *Nuclear Physics B*, 250(1-4):465–516, 1985.

[117] Steven Weinberg. Nuclear forces from chiral lagrangians. *Physics Letters,(Section) B;(Netherlands)*, 251(2), 1990.

[118] Steven Weinberg. Effective chiral lagrangians for nucleon-pion interactions and nuclear forces. *Nuclear Physics B*, 363(1):3–18, 1991.

[119] Steven Weinberg. Three-body interactions among nucleons and pions. *Physics Letters B*, 295(1-2):114–121, 1992.

[120] Markus Fierz. Zur fermischen theorie des β-zerfalls. *Zeitschrift für Physik*, 104(7):553–565, 1937.

[121] A Gezerlis, I Tews, E Epelbaum, M Freunek, S Gandolfi, K Hebeler, A Nogga, and A Schwenk. Local chiral effective field theory interactions and quantum monte carlo applications. *Physical Review C*, 90(5):054323, 2014.

[122] I Tews, Stefano Gandolfi, A Gezerlis, and A Schwenk. Quantum monte carlo calculations of neutron matter with chiral three-body forces. *Physical Review C*, 93(2):024305, 2016.