Enhancing the Lloyd-Mohseni-Rebentrost algorithm for information loading

Master in Quantum Science and Technology

Universidad del País Vasco
Euskal Herriko Unibertsitatea

FACULTY OF SCIENCE AND TECHNOLOGY
UNIVERSITY OF THE BASQUE COUNTRY

September 2022

Author: Pablo Rodríguez Grasa

Advisor: Dr. Mikel Sanz
Department of Physical Chemistry

Acknowledgements

En primer lugar, quiero agradecer al propio proceso de realización del trabajo, pues me ha reunido con personas brillantes y me ha regalado aprendizajes y momentos maravillosos durante estos meses.

En lo intelectual y en lo humano, gracias a mi tutor Mikel, que me ha dado la oportunidad de trabajar en un entorno inmejorable y me ha guiado desde el primer día hasta el último.

Gracias a Javi, por sus aportes y sus enriquecedoras discusiones. A Mikel García, siempre con una solución bajo la manga ante cualquier problema técnico. A Rubén por sus lecciones sobre Quantum Channels y a Tasio por su experiencia. En definitiva, a todo el grupo NQUIRE.

A mis padres por darme la oportunidad de centrarme en estudiar aquello que más me gusta y, junto con mi hermana Ana, por apoyarme en todo momento. Por último gracias a ti Emma, por aguantarme y hacerme feliz incluso en los días más complicados.

Summary

In recent decades, quantum computers have been postulated as an alternative with greater computational power than classical computers for solving certain problems. However, although some algorithms demonstrate exponential advantages in data processing, efficient data loading is still a reliable challenge. The absence of a general protocol to address this problem means that the coding method has to be adapted to the needs of the algorithm in question. In this work, we focus on one of the most important encodings, the dynamic or Hamiltonian embedding, specifically on a protocol that allows for the exponentiation of a density matrix ρ by employing multiple copies of it. Studying this protocol from an original quantum-channel perspective, we have been able to find a more general Hamiltonian than the SWAP operation proposed in previous literature. Remarkably, we have proven that the performance of the SWAP cannot be improved if the protocol is independent of the initial state, but it does admit improvement when a dependence on the density matrices is allowed. Last but not least, as density matrices are a very restricted family of square matrices, we have extended the protocol to general square matrices.

Contents

CHAPTER 1

Introduction

The modern idea of computer was born with the introduction of the groundbreaking concept of Turing machine in 1936 by the English mathematician Alan Turing. By a binary encoding of the information in bits and applying Boolean functions, classical computers are able to solve "any" solvable problem. However, despite the great advances in algorithmics, there are problems whose resolution has an exponential cost, catapulting ahead the calculation times such that they are beyond the capacity of the most powerful supercomputers. Quantum physics is a natural source of this kind of problems due to the exponential growth in the number of degrees of freedom when the system-size scales up. That is why when Richard Feynman published *Simulating Physics with computers* [1] in 1982, it was a turning point in this field. In this work, the American physicist proposed the use of controllable physical systems governed by the laws of quantum mechanics to efficiently simulate some of these problems. Three years later, the initial steps towards a quantum version of Turing's machine was proposed by David Deustch in his publication *Quantum theory, the Church-Turing principle and the universal quantum computer* [2]. A promising new computing paradigm was thus emerging, and the first signs of this growing interest came in the 1990s with the emergence of the first quantum algorithms. One of the very first ones was the Deustsch-Jozsa algorithm [3] proposed in 1992, but Peter Shor's algorithm [4] for prime number factorisation presented in 1994 was groundbreaking due to its consequences in cryptography and it brought a change in mentality about the usefullness of quantum technologies. In addition, with the publication in 1996 of *Universal quantum simulators* by Seth Lloyd [5], Feynman's approach was extended to digital simulations.

On paper, these algorithms show an advantage over their classical counterparts when the problem-size grows, but they are intensively hardware demanding, so they require huge controllable quantum processors. It was not until 1998 when, the Berkeley group led by Isaac Chuang, presented the first experimental realisation of a quantum algorithm [6] using nuclear Magnetic Resonance (NMR) as a platform for im-

plementing qubits. Although there are multiple alternatives for the implementation of these two-level controllable systems such as trapped ions [7] or superconducting circuits [8], as leading examples. However, the main drawback so far is technological, since scalability of the hardware is currently strongly limited by the imperfections in fabrication, undesired qubit-qubit interactions (crosstalk), and last but not least, unwanted bosonic interactions with environment. The effect of all of this in the dynamics of the quantum system is known as decoherence [9] and, if these errors cannot be corrected faster than they are created via quantum error correction [10], then there is a limitation to the depth of the algorithms which can be run in the quantum processor. The period in which the numer of resources available is not sufficent for implementing quantum error correction is called *Noisy Intermediate Scale Quantum* (NISQ) era [11].

It is not only the experimental part what keeps us away from a quantum computation. In order to run quantum algorithms in this new type of computers, an efficient information loading process, which does not jeopardize the quantum advantage of the information processing part is necessary. The fact that there is currently no general and efficient procedure for loading information in a quantum computer makes this problem to be considered as one of the most challenging questions to address. The type of encoding must be adapted to the quantum algorithm, which includes the quantum information processing structure and the retrieval of information, always limited by Holevo's theorem [12].

One specially relevant encoding is the dynamic embedding, in which the matrix of date is loaded as a Hamiltonian which allows the evolution operation to be applied on a quantum state. This type of operation is employed, for example, to solve systems of linear equations [13]. However, generating the exponential of a matrix is proven to be efficient in a few cases, for instance, in the case of sparse matrices [14]. It is also possible when the matrix of data is a quantum state and we have multiple copies at our disposal, as described in the *Lloyd-Mohseni-Rebentrost* (LMR) protocol in Ref. [15]. This protocol, which is the core of our work, relies on the ability to implement the exponential of the SWAP and the availability of n copies of a density matrix ρ to carry out the operation $e^{i\rho t}$ on a density matrix σ up to precision of first order in t^2/n.

In this Master Thesis, we aim at enhancing the LMR protocol. On the one hand, we proposed an improvement of the protocol which requires fewer copies of ρ to simulate the exponential by fixing the error. Remarkably, we show that the

performance of the SWAP cannot be improved in this sense when the protocol is ρ- and σ-independent. However, we find the Hamiltonian which optimises the number of copies required for the two-qubits case when the density matrices are previously given. On the other hand, we generalise the protocol beyond the restricted set of density matrices to any square matrix of dimension 2^n, where n is the number of qubits.

We have structured this Thesis in six chapters, including the introduction and the conclusions. In Chapter 2, we present the fundamental ingredients of quantum computation that are necessary for the proper understanding of this Thesis. Beyond basic concepts, such as the qubit and the quantum gates, we also describe the fundamentals of quantum channels, which we will employ to describe the LMR protocol from, to our knowledge, a novel description. In Chapter 3, we discuss the main types of information encoding, which are adapted to the subsequent data processing subroutine. Afterwards, in Chapter 4 we present in detail the LMR protocol and provide a description in the framework of quantum channels. We show that there is no enhancement in the number of copies of ρ if the protocol is independent of ρ and σ. However, we find in the two-qubit case that its performance can be improved if the protocol can depend on ρ and σ. Finally, in Chapter 5, we extend the protocol beyond density matrices towards arbitrary matrices.

CHAPTER 2

Fundamentals on quantum computation

In this chapter, we will see how the quantum mechanics formalism is applied to computation to encode and process information. Specifically, we will focus on digital quantum computing (DQC), which uses qubits as a unit of information, and on which quantum gates are applied to carry out the operations that form an algorithm. We will briefly present the basic tools of quantum computing. We will start with the formal definition of the qubit, then introduce the density matrix formalism which is very useful to describe quantum states, followed by a section on how to deal with composite systems. After this, we will present the main quantum gates and the definition of a quantum channel. Finally, we will review one of the most important tools for Hamiltonian simulation, the Trotter formula. This chapter is a synthesis of some of the concepts presented in Refs. [16–18].

2.1 The qubit

The qubit is a two-level quantum system and its state is described by a 2-dimensional Hilbert space. The basis par excellence is the computational basis formed by the vectors $|0\rangle$ and $|1\rangle$ which are the eigenvectors associated with the spin measurement in the Z-axis. We can describe the general state of a qubit as

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

where $\alpha, \beta \in \mathbb{C}$ and, due to the normalization condition, they satisfy $|\alpha|^2 + |\beta|^2 = 1$. This normalisation condition leads us to the most general representation of the state of a qubit

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\phi}\sin(\theta/2)\,|1\rangle\,, \tag{2.2}$$

which can be visualized in a two-dimensional radius 1 sphere known as Bloch sphere. This representation is depicted in Fig. 2.1.

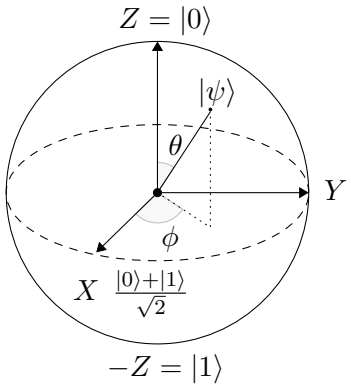

Figure 2.1: Bloch sphere representation of a general qubit quantum state $|\psi\rangle$ which is determined by the (θ, ϕ) angles.

When considering a n-qubit system we will use the composition of systems via the tensor product as presented in Section 2.3, obtaining a 2^n dimensional Hilbert space. The computational basis of such a system will be composed of all possible n-bit strings, which can be denoted in the binary basis or, equivalently, by the integer that this encoding represents. Hence, the representation of an arbitrary n-qubit state is

$$|\psi\rangle = \sum_{i_1,\cdots,i_n=0,1} \alpha_{i_1\cdots i_n} |i_1 \cdots i_n\rangle = \sum_{i=0}^{2^n-1} \alpha_i \, |i\rangle \,, \tag{2.3}$$

where the amplitudes $\alpha_i \in \mathbb{C}$ satisfy $\sum_i |\alpha_i|^2 = 1$.

2.2 Density matrix formalism

The density matrix formalism allows us to describe both pure and mixed states, thus allowing for a more general treatise on systems which are not completely known. In the following we will give an overview of the properties of density matrices, which are of particular relevance to this work. For pure states, which are represented as Hilbert space vectors, the associated density matrix will simply be $\rho = |\psi\rangle\langle\psi|$. On the other hand, a mixed state is a probabilistic mixture of pure quantum states and its associated density matrix for a finite-dimensional Hilbert space is defined as

$$\rho \equiv \sum_i p_i \, |\psi_i\rangle\langle\psi_i| \,, \tag{2.4}$$

where p_i is the probability of finding the system in the pure state $|\psi_i\rangle$, so $\sum_i p_i = 1$ must be fulfilled. It is important to note that these pure states need not be orthonormal to each other. To ensure the description of a physical system, the density operator must fulfil the following properties:

- ρ is a Hermitian operator.
- ρ is a positive semi-definite operator which means that $\vec{x}^T \rho \vec{x} \geq 0$ for all $\vec{x} \in \mathbb{R}^n$.
- Tr $\rho = 1$ and Tr $\rho^2 \leq 1$ where the equality holds for pure states and the inequality for mixed states.

These first two properties together imply that ρ is semi definite-positive ($\rho \geq 0$) if and only if all its eigenvalues are greater than or equal to 0.

In addition, let us consider the time evolution of a quantum system described by $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$. Pure states evolve as

$$|\psi_i\rangle \xrightarrow{U} U |\psi_i\rangle , \tag{2.5}$$

where U is the corresponding time evolution operator. In this way, the density matrix will evolve as

$$\sum_i p_i \, U |\psi_i\rangle\langle\psi_i| \, U^\dagger = U \rho \, U^\dagger. \tag{2.6}$$

In the following section, we will use this density matrix formalism to deal with composite systems.

2.3 Composite systems

Consider now the description of two qubits A and B each in a Hilbert space. The state of the composite system AB will belong to the Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_B$ and it description will be given by the tensor product of the state of each of the qubits

$$|\psi\rangle_{AB} = |\psi_A\rangle \otimes |\psi_B\rangle = |\psi_A \psi_B\rangle , \tag{2.7}$$

$$\rho_{AB} = \rho_A \otimes \rho_B. \tag{2.8}$$

This can be extended to n qubits. Thus, from several separated qubits states, we can straightforwardly describe the joint system. However, sometimes we may have

a description of a composite system that cannot be separated into a product of the states of the individual qubits that make it up, then we say that our system is *entangled*.

Occasionally, as we will see in this work, given a state describing a set of qubits we may be interested in analysing the state of a single qubit or a subgroup of qubits. To do so, we will calculate the reduced matrix by tracing out the system or systems that we want to discard. For example, for a system AB described by ρ_{AB} we can calculate the density matrix describing only system A as

$$\rho_A = \mathrm{Tr}_B(\rho_{AB}) = \sum_{k=1}^{d_A} (\mathbb{1}_A \otimes \langle k|_B)\, \rho_{AB} (\mathbb{1}_A \otimes |k\rangle_B), \tag{2.9}$$

where d_B is the dimension of $\mathcal{H}_B$ and $\{|k\rangle_B\}$ is any complete basis of $\mathcal{H}_B$. In the simple case of product states as in equation Eq.2.8 we must recover the state of the reduced system

$$\mathrm{Tr}_B(\rho_A \otimes \rho_B) = \underbrace{\mathrm{Tr}(\rho_B)}_{=1}\, \rho_A = \rho_A. \tag{2.10}$$

This description allows us to introduce a very useful technique in both quantum computation and information called *purification*. Let us consider a quantum system A whose state is defined by ρ_A. It is always possible to introduce an extra system which we denote as P, hence constructing a pure state $|AP\rangle$ describing the state of the joint system AP fulfilling $\rho_A = \mathrm{Tr}_P\big(|AP\rangle\langle AP|\big)$.

2.4 Quantum gates

Whereas classical computers use logic gates, which are irreversible, quantum gates are reversible unitary operators. Unlike in Analogue Quantum Computing (AQC), where we have the time evolution of a Hamiltonian; or in Digital-Analog Quantum Computing (DAQC) [19], where this evolution is combined with quantum gates; in Digital Quantum Computing time plays no role as the only operators are time independent matrices.

In quantum circuit notation every horizontal line represents a qubit and evolution goes from left to right. This determines the sequential application of the gates on the states as depicted in Fig. 2.2.

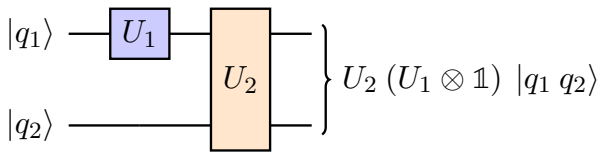

Figure 2.2: Example of a two-qubit quantum circuit in which the single-qubit unitary U_1 and the two-qubit unitary U_2 are applied. Notice that when nothing is applied it is equivalent to an identity gate.

We will now present the most relevant quantum gates, distinguishing between those that apply to one qubit and those that apply to two qubits.

2.4.1 Single qubit gates

The transformations implemented by single qubit gates are a mapping from one point on the Bloch sphere to another, so they can be represented as rotations. The most general representation of a single qubit quantum gate is the U-gate

$$U(\theta,\phi,\lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\phi+\lambda)}\cos(\theta/2). \end{pmatrix} \tag{2.11}$$

where the choice of the three parameters allows us to recover any single qubit gate. As a first sample of this, the gate that results in leaving the state invariant since $(\theta,\phi,\lambda) = (0,0,0)$, is the identity gate

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.12}$$

which, although it may seem irrelevant, is of great importance in describing the state of the system after the application of different quantum gates as we can see in Fig. 2.2. A set of quantum gates of particular relevance, are the Pauli matrices, whose representations in the computational basis are as follows

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.13}$$

Notice that they can also be denoted as $(\sigma_x,\sigma_y,\sigma_z)$, both conventions will be

used in this work. These matrices present some relevant properties

$$\mathrm{Tr}\sigma_i = 0, \tag{2.14}$$
$$\sigma_i^2 = \mathbb{1}, \tag{2.15}$$
$$\sigma_j\sigma_k = \delta_k^j\mathbb{1} + i\epsilon_{jkl}\sigma_l, \tag{2.16}$$
$$[\sigma_i, \sigma_j] = 2i\epsilon_{ijk}\sigma_k, \tag{2.17}$$
$$\{\sigma_i, \sigma_j\} = 2\delta_k^j\mathbb{1}, \tag{2.18}$$

where the sub-indexes i, j, k can be x, y, z and ϵ_{ijk} is the Levi-Civita tensor. The Z-gate represents a $\phi = \pi$ angle rotation around the z-axis, which is a particular case of the more general phase gate

$$P = U(0, 0, \lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix}. \tag{2.19}$$

Other paritcular cases of interest are the $S = P(\pi/2)$ and $T = P(\pi/4)$ gates.

One of the most used gates is the Hadamard gate

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{2.20}$$

which creates an equal superposition of the computational basis states and is the matrix of change between the X and Z basis

$$H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \equiv |+\rangle \tag{2.21}$$
$$H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}} \equiv |-\rangle\,. \tag{2.22}$$

Notice that $X = HZH$ and $Z = HXH$ are indeed fulfilled.

2.4.2 Two-qubit gates

Unlike the single-qubit gates which act on the Hilbert space of each qubit separately, the two-qubit gates act on the Hilbert space of the whole system. This makes it possible to create entanglement between qubits, for instance via the CNOT gate. It has two inputs, the control and the target. If the control is in the state $|0\rangle$ the target remains unchanged, and if the control is in the state $|1\rangle$ an X gate is applied

in the target, this is why this gate is also called controlled-X. Its representation in the computational basis is as follows

$$\text{CNOT} = |0\rangle\langle 0| \otimes \mathbf{1} + |1\rangle\langle 1| \otimes X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{2.23}$$

One of the cornerstones of this work is the SWAP gate, which exchanges the states of two qubits. In general, we can define it for a n-qubit state by exchanging the p for the q

$$\text{SWAP}_{p,q} = \sum_{i_1,\cdots,i_n=0,1} |i_1\rangle\langle i_1| \otimes \cdots \otimes |i_q\rangle\langle i_p| \otimes \cdots \otimes |i_p\rangle\langle i_q| \otimes \cdots \otimes |i_n\rangle\langle i_n|. \tag{2.24}$$

For the specific case of a two qubits system, $\text{SWAP}|q_1\, q_2\rangle = |q_2\, q_1\rangle$, and its matrix form is

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{2.25}$$

2.5 Quantum channels

Any processing of quantum information can be presented as a quantum channel, which is a mapping with certain properties we will see below that relates input and output. This formalism is used to analyse the evolution of an open quantum system which can be described as an isolated quantum system, that in turn can interact with the environment. This environment can be a measurement or noise, among other processes. Some examples of some interesting uses of this formalism can be found in [20]. Quantum states transform in the most general way by means of quantum operations as

$$\rho' = \mathcal{E}(\rho), \tag{2.26}$$

where $\mathcal{E}$ connects an input state Hilbert space $\mathcal{H}$ with an output state one $\mathcal{H}'$. This mapping must fulfill the following conditions:

1. Linearity
   Consider $\rho, \sigma \in \mathcal{H}$ and $a, b \in \mathbf{C}$. Linearity condition means

$$\mathcal{E}(a\rho + b\sigma) = a\,\mathcal{E}(\rho) + b\,\mathcal{E}(\sigma). \tag{2.27}$$

2. Trace preserving
   This ensures the conservation of the probability and is represented as

$$\mathrm{Tr}\Big(\mathcal{E}(\rho)\Big) = \mathrm{Tr}(\rho). \tag{2.28}$$

3. Complete positivity
   This property, together with the previous one, guarantees that a density matrix continues being a density matrix and implies that if $\rho \geq 0$ then $\mathcal{E}(\rho) \geq 0$.

The interpretation for studying the evolution of an open quantum system is to consider a closed system consisting of the system of interest for our study, which will be denoted as *principal system*, and an *environment*. In this case, we cannot consider that the transformation $\mathcal{E}(\rho)$ is unitary. Let us assume that the initial state is a product between ρ and the initial state of the environment system ρ_{env}. Now, we are allowed to consider that the evolution of the joint system evolves unitarily through U. After the evolution, principal system and environment do not interact anymore and we can trace out the environment obtaining

$$\mathcal{E}(\rho) = \mathrm{Tr}_{\mathrm{env}}\Big(U(\rho \otimes \rho_{\mathrm{env}})U^\dagger\Big), \tag{2.29}$$

which is the reduced state of the principal system.

Quantum channels can be represented by the *operator sum representation*

$$\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger, \tag{2.30}$$

where the set $\{E_k\}$ is known as *Kraus operators*. In order to fulfill Eq. 2.28 it must be satisfied $\sum_k E_k^\dagger E_k = \mathbb{1}$. Let us consider the environment to be initially described by $\rho = |e_0\rangle\langle e_0|$, which is a pure state. This can be done without loss of generality as we always can introduce an extra system purifying the environment as we saw in Section 2.3. Hence, going back to Eq. 2.29 and developing the partial trace over the environment space

$$\begin{aligned}\mathcal{E}(\rho) &= \sum_k (\mathbb{1} \otimes \langle k|)\Big(U(\rho \otimes |e_0\rangle\langle e_0|)U^\dagger\Big)(\mathbb{1} \otimes |k\rangle) \\ &= \sum_k (\mathbb{1} \otimes \langle k|)\Big(U(\mathbb{1} \otimes |e_0\rangle)(\rho \otimes \mathbb{1})(\mathbb{1} \otimes \langle e_0|)U^\dagger\Big)(\mathbb{1} \otimes |k\rangle),\end{aligned} \tag{2.31}$$

where the Kraus operators are

$$E_k = (\mathbb{1} \otimes \langle k|)\, U\, (\mathbb{1} \otimes |e_0\rangle). \tag{2.32}$$

One of the most important concepts in the framework of quantum channels is the fixed point. A fixed point is the quantum state β which remains invariant under the action of the quantum channel, i.e. it satisfies $\mathcal{E}(\beta) = \beta$. This quantum state β is then said to be a *stationary state*. Any quantum operation $\mathcal{E}$ has spectral radius 1, and at least one stationary state associated with the modulus one eigenvalue [21]. In the case where the eigenvalue of modulus one is not degenerate, the following is satisfied

$$\mathcal{E}^{n\to\infty}(\sigma) = \beta, \tag{2.33}$$

where $\mathcal{E}^{n\to\infty}$ means that the quantum channel is applied a number of times n tending to infinity, σ is a density operator and β is the only fixed point of the quantum channel.

2.6 Hamiltonian simulation: Trotter's formula

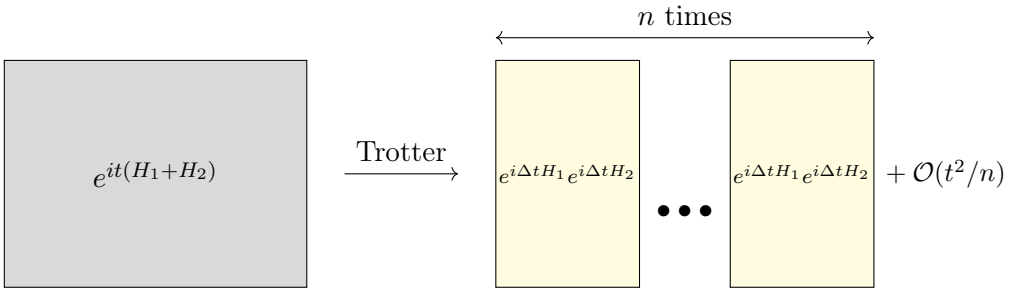

Figure 2.3: Scheme of Trotter's formula to approximate a Hamiltonian $H = H_1 + H_2$ for a time t as the $n-$times evolution under Hamiltonians H_1 and H_2 for a time $\Delta t = t/n$.

The simulation of physical systems comes from solving differential equations that dictate their behaviour. When it comes to studying the behaviour of a simple quantum system we have to solve the Schrödinger equation, which in normal units reads

$$i\frac{\mathrm{d}}{\mathrm{d}t}\left|\psi(t)\right\rangle = H(t)\left|\psi(t)\right\rangle, \tag{2.34}$$

where $H(t)$ is the Hamiltonian of the system. When dealing with time-independent Hamiltonians it is straightforward to write the solution to this equation

$$\left|\psi(t)\right\rangle = e^{-iHt}\left|\psi(0)\right\rangle. \tag{2.35}$$

However, H can be arbitrarily difficult to exponentiate so we must approximate the expression. Although this can be approximated to first order in a relatively simple way, it is likely that we will not get good enough results. To obtain better approximations, there are some Hamiltonians, such as Ising's [22], which allow separation into one- and two-body terms. Each of these terms, acting on a smaller subsystem, are easier to implement using quantum circuits. Nevertheless, notice that this terms will generally not commute with each other and therefore we will not be able to separate the exponential of the sum in the product of exponentials. That is why we introduce Trotter's formula [23], which is fundamental in Hamiltonian simulation

$$e^{H_1+H_2} = \lim_{n\to\infty} (e^{H_1/n} e^{H_2/n})^n, \tag{2.36}$$

where H_1 and H_2 are Hermitian operators. As a consequence of this equation we can obtain the second order aproximation

$$e^{i(H_1+H_2)\Delta t} = e^{iH_1\Delta t} e^{iH_2\Delta t} + \mathcal{O}(\Delta t^2), \tag{2.37}$$

which is relevant to this work. Notice that, using this last expression, to simulate up to a time t we can manipulate the expression

$$e^{it(H_1+H_2)} = e^{i\frac{t}{n}(H_1+H_2)n} = \prod_{k=1}^{n} e^{i\frac{t}{n}(H_1+H_2)} = \prod_{k=1}^{n} e^{i\Delta t H_1} e^{i\Delta t H_2} + \mathcal{O}(t^2/n), \tag{2.38}$$

where we have taken $\Delta t = t/n$. Basically, we are moving from simulating $H_1 + H_2$ for a time t to simulating H_1 and H_2 separately at n time intervals Δt as shown in Figure 2.3. If I want an error ϵ I must take $n \in \mathcal{O}(t^2/\epsilon)$.

CHAPTER 3

Information loading in quantum computers

Once we have introduced the fundamental concepts for this work on quantum computation, in this chapter we will motivate the loading of information into quantum computers. To do this, we will go through the IPO model which describes the algorithms as the composition of three parts. Afterwards, we will relate this decomposition to the Holevo Bound and finally, we will focus on the first part of this description which is the most relevant for the content of this Thesis. For this purpose, we will present the main encodings that exist for loading information in quantum computers.

3.1 The Input-Processing-Output (IPO) model

As explained in Ref. [24], the IPO model describes classical algorithms as the composition of three parts: input, the system receives the information; process, the system computes using the inputs; and output, the system returns the results. In general, this description also fits quantum algorithms. In the quantum case, the input corresponds to the loading of information, the process with the body of the algorithm, and the output with the information retrieval. These stages are shown in Figure 3.1. For years, the greatest efforts have been put into the body of the algorithms and many of them have been optimised, showing quantum advantage over their classical counterparts. However, the complexity of the algorithm as a whole will be given by the complexity of the less efficient part. This means that, as much as we can speed up the processing of information, we need to optimise the loading of information. Additionally, unlike in classical computing where the information is stored in a RAM memory, in quantum computing every time we run an algorithm and extract the information, the measurement process destroys the quantum system, so to re-run the body of the algorithm we need to load the information again.

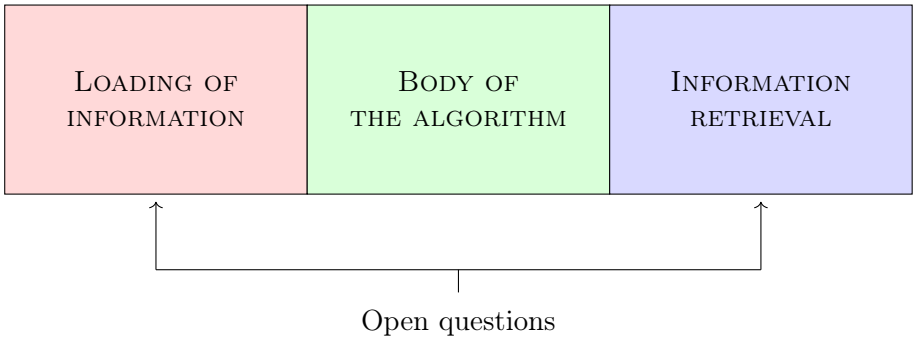

Figure 3.1: The three stages of a quantum algorithm within the IPO model. While great strides have been made in optimising information processing, the problems of data loading and retrieval must also be addressed in order to achieve efficient algorithms.

3.2 Relationship with the Holevo bound

To state Holevo's Theorem [12], let us consider the following situation. Alice has a set of symbols which we denote as $X = 0, ..., n$. From these symbols she constructs quantum states $\rho(X)$ following a probability distribution given by $p_0, ..., p_n$. Alice sends one of these states to Bob, who performs a measurement and uses the result of the measurement Y to get as much information from X as he can. The information gain after Bob performs the measurement is quantified by the mutual information $H(X : Y) = H(X) - H(X|Y)$ whose maximum value is determined by the Holevo bound

$$H(X : Y) \leq \underbrace{S(\rho) - \sum_x p_x S(\rho(X))}_{Holevo\ \chi\ quantity}, \tag{3.1}$$

where $\rho = \sum_x p_x \rho(X)$ and $S(\rho) = -\mathrm{Tr}(\rho \log \rho)$ is the von Neumann's entropy. This bound forms Holevo's Theorem which is of great relevance in quantum information theory. Nevertheless, we can find a parallelism with the IPO decomposition of a quantum algorithm indeed. To do this, we can relate information loading or input to the sender of a message (Alice); information processing to the communication channel; and information retrieval or output to the receiver (Bob). As a consequence of this theorem, if we encode information using n qubits, we can extract at most n qubits of information at the end of my algorithm. Therefore, when loading the data, it should be borne in mind that the more you compress the data, the harder it is to

retrieve it.

3.3 Main encodings

As mentioned beforehand, in order to be processed by a quantum computer, information must first be encoded. Taking into account that there is no general efficient way that can be used for any encoding, depending on the problem we want to deal with with our algorithm, one encoding or another must be chosen. In the following, we proceed to present three of the main encodings used in quantum computation.

3.3.1 Basis embedding

This system consists of mapping a bit string into a quantum state. Having a classical string $x = \{a_0 \dots a_{n-1}\}$, we obtain the quantum state $|x\rangle = |a_0 \dots a_{n-1}\rangle$ where $a_i \in 0, 1$ and $i = 0, \dots, n-1$, so we are encoding each bit in a qubit. The quantum circuit that addresses the simplest case, which is to load a classical bit into a qubit, is depicted in Figure 3.2. In the case of m bit strings this encoding could be used to prepare a superposition state

$$|\psi\rangle = \frac{1}{\sqrt{m}} \sum_{i=1}^{m} \left| x^{(i)} \right\rangle , \tag{3.2}$$

where $x^{(i)}$ refers to each bit string which are loaded simultaneously. In this case there is no compression but we are taking advantage of quantum parallelism to be able to run several states at once.

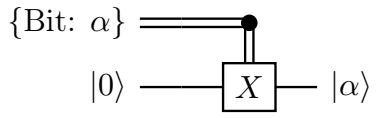

Figure 3.2: Circuit representation of the load of a classical bit α into a quantum state $|\alpha\rangle$. This circuit and others used for this type of coding can be found in Ref. [25]

3.3.2 Amplitude embedding

In this case, unlike in basis embedding where we have a bit string, we have a classical vector x normalised to unity, i.e. $\sum_i |x_i|^2 = 1$, and we want to encode it in n qubits. The amplitudes of the generated quantum state will be given by this classical information so that

$$|\psi\rangle = \sum_{i=0}^{N-1} x_i |i\rangle , \tag{3.3}$$

where $N = 2^n$ and i is an integer which can be expressed as a bit string. In this encoding, an exponential compression is carried out, as a distinction must be made between 0 and $2^{-n/2}$ amplitudes. Hence, an exponential number of copies will be needed to be able to make the complete retrieval of information. There are algorithms that only need to retrieve part of the information and which do allow to take advantage of this compression.

This encoding can be performed by means of the Grover-Rudolph method [26,27], black-box methods [28, 29] and variational methods [30].

3.3.3 Dynamic or Hamiltonian embedding

In contrast to the previous ones, with this method, also known as Hamiltonian simulation, we do not encode classical information directly into quantum states. In this case, the information of a given Hermitian matrix A is loaded through the evolution of an initial state given by its associated Hamiltonian denoted as $\hat{H}_A$

$$|\psi(t)\rangle = e^{-i\hat{H}_A t} |\psi_0\rangle . \tag{3.4}$$

In the case where A is not Hermitian, as we must ensure the unitary nature of the operations in order to carry them out on a quantum computer, we can use the matrix

$$A' = \begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix} \tag{3.5}$$

to build the Hamiltonian.

It is the latter approach to information loading that is the focus of this work. Although we know that any unitary operator $e^{-i\hat{H}_A}$ can be decomposed into quantum gates by means of Givens rotations [31], this decomposition for an arbitrary unitary matrix needs an exponential number of two-qubit gates. However, when the matrix

we want to is a density matrix ρ, the *Lloyd-Mohseni-Rebentrost* (LMR) protocol [15] presented in Chapter 4 allows its efficient loading.

CHAPTER 4

Density matrix exponentiation

After motivating the importance of information loading in quantum computers, we will now present the seed protocol around which our work revolves. Presented in Ref. [15], the protocol, that we denote by *Lloyd-Mohseni-Rebentrost* (LMR) in this Thesis, applies in the information loading subroutine a similar approach to the information retrieval subroutine, in which multiples copies of an unknown state allows us to reconstruct this state (quantum state tomography). The more copies at our disposal, the more accurate we can construct the state. The LMR protocol applies a similar approach by employing multiple copies of a quantum state ρ to generate its exponential $e^{-i\rho t}$ and to apply this unitary operation on an arbitrary density matrix σ. In this procedure, known as density matrix exponentiation, the density matrix acts as a Hamiltonian. This has a wide application in quantum principal component analysis (qPCA), since it allows the quantum phase estimation algorithm [32] to be applied to find the eigenstates and eigenvalues of ρ, or in the simulation of Hamiltonians [33], among other applications. To do this exponentiation, it is assumed that we have n copies of ρ, which allows us to perform repeated partial SWAP-like operations on the composed system $\rho \otimes \sigma$. This operation is described by the equation

$$\begin{aligned} \mathrm{Tr}_1\left(e^{-iS\Delta t}\,\rho\otimes\sigma\, e^{iS\Delta t}\right) &= \cos^2\Delta t\,\sigma + \sin^2\Delta t\,\rho - i\,\sin\Delta t\,\cos\Delta t\,[\rho,\sigma] \\ &= \sigma - i\Delta t[\rho,\sigma] + \mathcal{O}\left(\Delta t^2\right), \\ &\quad\uparrow \\ &\quad\Delta t\ll 1 \end{aligned} \tag{4.1}$$

where notice that only leading order terms are kept for $\sin(\Delta t) \approx \Delta t + \mathcal{O}\left(\Delta t^3\right)$ and $\cos(\Delta t) \approx 1 + \mathcal{O}\left(\Delta t^2\right)$. Here S is the aforementioned SWAP operator which acts on two subsystems by exchanging their information, and whose exponentiation can be implemented by means of the quantum circuit presented in Ref. [34]. For this work, we will consider that ρ and σ have the same dimension so that S will act on the Hilbert space of the joint system.

We now demonstrate the result of Eq. 4.1. We start by using the relation

$e^{\pm iM\Delta t} = \cos\Delta t\,\mathbb{1} \pm i\,\sin\Delta t\,M$, where M is a matrix which fulfills $M^2 = \mathbb{1}$, and thus separating it into the different terms

$$\begin{aligned}\mathrm{Tr}_1\big(e^{-iS\Delta t}\,\rho\otimes\sigma\,e^{iS\Delta t}\big) = \cos^2\Delta t\;\mathrm{Tr}_1\big(\rho\otimes\sigma\big) + i\;\sin\Delta t\;\cos\Delta t\;\mathrm{Tr}_1\big(\rho\otimes\sigma\,S\big)\\ - i\;\sin\Delta t\;\cos\Delta t\;\mathrm{Tr}_1\big(S\,\rho\otimes\sigma\big) + \sin^2\Delta t\;\mathrm{Tr}_1\big(\sigma\otimes\rho\big),\end{aligned} \tag{4.2}$$

where the partial trace of the first and last terms are trivially computed as $\mathrm{Tr}_1\big(\rho\otimes\sigma\big) = \sigma\,\mathrm{Tr}\rho = \sigma$. Partial traces containing SWAP's are more delicate to calculate. Following Ref. [35],

$$\begin{aligned}\mathrm{Tr}_1\big(\rho\otimes\sigma\,S\big) &= \sum_{l=0}^{\mathrm{d}_{\mathcal{H}}-1}\big(\langle l|\otimes\mathbb{1}\big)\big(\rho\otimes\sigma\,S\big)\big(|l\rangle\otimes\mathbb{1}\big) = \sum_{l=0}^{\mathrm{d}_{\mathcal{H}}-1}\sum_{k,j=0}^{\mathrm{d}_{\mathcal{H}'}-1}|j\rangle\langle l\,j|\rho\otimes\sigma\;\underbrace{S|l\,k\rangle}_{|k\,l\rangle}\langle k|\\ &= \sum_{l=0}^{\mathrm{d}_{\mathcal{H}}-1}\sum_{k,j=0}^{\mathrm{d}_{\mathcal{H}'}-1}|j\rangle\langle k|\langle l|\rho|k\rangle\langle j|\sigma|l\rangle = \underbrace{\sum_{k,j=0}^{\mathrm{d}_{\mathcal{H}'}-1}|j\rangle\langle k|\langle j|\sigma\rho|k\rangle}_{\sum_l|l\rangle\langle l|=\mathbb{1}} = \sigma\rho,\end{aligned} \tag{4.3}$$

where in the second step we have introduced two identities $\sum_j(\mathbb{1}\otimes|j\rangle)(\mathbb{1}\otimes\langle j|)$ and $\sum_k(\mathbb{1}\otimes|k\rangle)(\mathbb{1}\otimes\langle k|)$. Notice that $\mathcal{H}$ and $\mathcal{H}'$ are respectively the Hilbert spaces where ρ and σ live and, as mentioned above, they have the same dimension $\mathrm{d}_{\mathcal{H}} = \mathrm{d}_{\mathcal{H}'}$. We can straightforwardly compute the term with the SWAP to the left of the tensor product by doing the conjugate transpose, since ρ and σ are Hermitian matrices

$$\mathrm{Tr}_1\big(S\,\rho\otimes\sigma\big) = \mathrm{Tr}_1\big(\rho\otimes\sigma\,S\big)^\dagger = \big(\sigma\rho\big)^\dagger = \rho\,\sigma. \tag{4.4}$$

Gathering all the terms together and applying the definition of the commutator $[\rho,\sigma] = \rho\sigma - \sigma\rho$, we obtain the result of Eq. 4.1 thus completing our proof.

Returning to the LMR protocol, it is claimed in Ref. [15] that after n applications of Eq. 4.1, allows us to approximately construct the operation $e^{-i\rho n\Delta t}\sigma e^{i\rho n\Delta t}$. Hence, to simulate $e^{-i\rho t}$ to accuracy δ we need $n \leq \mathcal{O}(t^2/\delta)$ steps, where we have taken $t = n\Delta t$ thus obtaining our desired operation. We will show these assertions in detail below.

4.1 Quantum channel of the LMR

The following, we will study in detail the operation in Eq. 4.1, which is the cornerstone of the LMR protocol. We will consider that such an operation can be described by the action of a quantum channel acting on the quantum state σ.

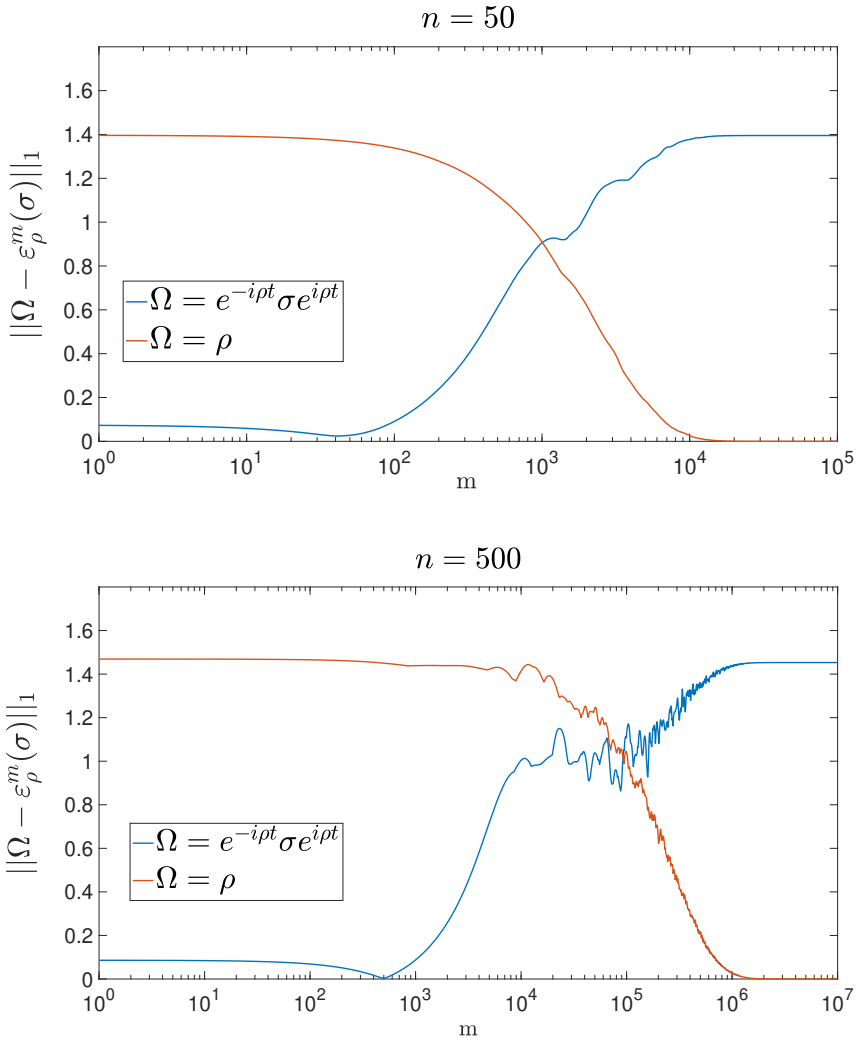

Figure 4.1: Schatten norm 1 of the difference between the application of the quantum channel and the target operation (blue line), and between the quantum channel and its fixed point ρ (orange line) for two values of Δt, both with $t = 1$, top with number of copies $n = 50$ and bottom with $n = 500$, and for random density matrices ρ and σ. Notice that the distance between the target operation and the output of the quantum channel has a minimum at $m = n$, which becomes more pronounced for a higher value of n. Also note that the distance between ρ and the channel tends to 0 as m increases, which means that the output of the quantum channel tends to the fixed point, as expected.

We now present the output obtained from applying the channel n times iteratively.

Result 1: n applications of the channel

The state resulting from applying the LMR protocol channel n times on an initial state σ with n copies of ρ is

$$\mathcal{E}_\rho^n(\sigma) = \sum_{k=0}^{n} \binom{n}{k} \cos^{2n-k} \Delta t \, [\rho, \sigma]_k \, \sin^k \Delta t \, (-i)^k + \rho \, (1 - \cos^{2n} \Delta t) \quad (4.5)$$

where $[\rho, \sigma]_k = \underbrace{[\rho, ...[\rho, [\rho}_{k}, \sigma]]...]$ is the nested commutator and $[\rho, \sigma]_{k=0} = \sigma$.

Looking at Eq. 4.1, it is straightforward to find that the stationary state is ρ. Convergence towards the fixed point and the difference between the output of the quantum channel and the target operation for random ρ and σ is depicted in Figure 4.1.

Proof of Eq. 4.5. We use induction to show this. Hence, we start by checking that this is true for $n = 1$ and then, by assuming it is true for n, check that reapplying the channel to this is equivalent to applying it $n + 1$ times.

- $n = 1$:

$$\begin{aligned} \mathcal{E}_\rho^{n=1}(\sigma) &= \sum_{k=0}^{1} \binom{1}{k} \cos^{2-k} \Delta t \, [\rho, \sigma]_k \, \sin^k \Delta t \, (-i)^k + \rho \sin^2 \Delta t \\ &= \cos^2 \Delta t \, \sigma - i \cos \Delta t \, \sin \Delta t \, [\rho, \sigma] + \rho \sin^2 \Delta t. \end{aligned} \quad (4.6)$$

- $n > 1$: Our aim is to check if $\mathcal{E}\big(\mathcal{E}^n(\sigma)\big) = \mathcal{E}^{n+1}(\sigma)$.

  Right-hand side (RHS)

$$\mathcal{E}^{n+1}(\sigma) = \sum_{k=0}^{n+1} \binom{n+1}{k} \cos^{2(n+1)-k} \Delta t [\rho, \sigma]_k \sin^k \Delta t (-i)^k + \rho \, (1 - \cos^{2(n+1)} \Delta t). \quad (4.7)$$

Left-hand side (LHS)

$$\begin{aligned}\mathcal{E}\left(\mathcal{E}^n(\sigma)\right) = &\sum_{k=0}^{n}\binom{n}{k}\cos^{2(n+1)-k}\Delta t\,[\rho,\sigma]_k\,\sin^k\Delta t\,(-i)^k\\ &+\rho\,(\cos^2\Delta t-\cos^{2(n+1)}\Delta t)\\ &+\sum_{k=0}^{n}\binom{n}{k}\cos^{2n+1-k}\Delta t\,[\rho,\sigma]_{k+1}\,\sin^{k+1}\Delta t\,(-i)^{k+1}\\ &-i\sin^3\Delta t\sum_{k=0}^{n-1}[\rho,\rho\cancelto{0}{\cos^{2k+1}\Delta t}]\\ &+\rho\sin^2\Delta t,\end{aligned} \tag{4.8}$$

where the first two lines are the $\cos^2\Delta t\ \mathcal{E}(\sigma)$ term, the following two lines are the $-i\sin\Delta t\cos\Delta t[\rho,\mathcal{E}^n(\sigma)]$ term, and the very last line is the σ-independent term. Notice that we have used $[\rho,[\rho,\sigma]_k]=[\rho,\sigma]_{k+1}$. We rewrite the first term separating the first term of the sum and redefining the dummy indexes

$$\begin{aligned}&\sum_{k=1}^{n}\binom{n}{k}\cos^{2(n+1)-k}\Delta t\,[\rho,\sigma]_k\,\sin^k\Delta t\,(-i)^k+\cos^{2(n+1)}\Delta t\,\sigma\\ &=\sum_{\substack{\uparrow\\ k\to k+1}}^{n-1}\binom{n}{k+1}\cos^{2n+1-k}\Delta t\,[\rho,\sigma]_{k+1}\,\sin^{k+1}\Delta t\,(-i)^{k+1}+\cos^{2(n+1)}\Delta t\ \sigma.\end{aligned} \tag{4.9}$$

Now, we rewrite the third term by separating the $k=n$ sum element

$$\begin{aligned}&\sum_{k=0}^{n-1}\binom{n}{k}\cos^{2n+1-k}\Delta t\ \ [\rho,\sigma]_{k+1}\,\sin^{k+1}\Delta t\,(-i)^{k+1}\\ &+\cos^{n+1}\Delta t\,\sin^{n+1}\Delta t\,[\rho,\sigma]_{n+1}(-i)^{n+1}.\end{aligned} \tag{4.10}$$

Finally we gather the second, the third and last terms together

$$\rho\,(\cos^2\Delta t-\cos^{2(n+1)}\Delta t)+\rho\sin^2\Delta t=\rho\,(1-\cos^{2(n+1)}\Delta t). \tag{4.11}$$

By summing Eq. 4.9 and Eq. 4.10, using that $\binom{n}{k}+\binom{n}{k+1}=\binom{n+1}{k+1}$, and taking

into account the terms from Eq. 4.11, then the LHS transforms into

$$\begin{aligned}\mathcal{E}\left(\mathcal{E}^n(\sigma)\right) = &\sum_{k=0}^{n-1}\binom{n+1}{k+1}\cos^{2n+1-k}\Delta t\ [\rho,\sigma]_{k+1}\ \sin^{k+1}\Delta t\,(-i)^{k+1}\\ &+\underbrace{\cos^{n+1}\Delta t\ \sin^{n+1}\Delta t\,[\rho,\sigma]_{n+1}(-i)^{n+1}}_{k=n\ \text{term}}+\cos^{2(n+1)}\Delta t\ \sigma\\ &+\rho\,(1-\cos^{2(n+1)}\Delta t).\end{aligned} \tag{4.12}$$

Notice that the two last terms cancel the second and third terms in the RHS (Eq. 4.7). Thus, we have to prove the following equation:

$$\begin{aligned}&\sum_{k=0}^{n}\binom{n+1}{k+1}\cos^{2n+1-k}\Delta t\ [\rho,\sigma]_{k+1}\ \sin^{k+1}\Delta t\,(-i)^{k+1}+\underbrace{\cos^{2(n+1)}\Delta t\ \sigma}_{k=-1\ \text{term}}\\ &=\sum_{k=0}^{n+1}\binom{n+1}{k}\cos^{2-k}\Delta t\,[\rho,\sigma]_k\ \sin^k\Delta t\,(-i)^k,\end{aligned} \tag{4.13}$$

where we have identified the $k=-1$ term of the first sum. Working at the LHS of Eq. 4.13,

$$\begin{aligned}&\sum_{k=-1}^{n}\binom{n+1}{k+1}\cos^{2n+1-k}\Delta t\ [\rho,\sigma]_{k+1}\ \sin^{k+1}\Delta t\,(-i)^{k+1}\\ &=\sum_{\substack{\uparrow\\ k\to k-1}}^{n+1}\binom{n+1}{k}\cos^{2(n+1)-k}\Delta t\,[\rho,\sigma]_k\ \sin^k\Delta t\,(-i)^k,\end{aligned} \tag{4.14}$$

which is equal to the RHS in Eq. 4.13, concluding our proof.

As we have seen in Section 2.5, any channel can be expressed by means of the operator sum representation. Therefore, to complete the study of the interpretation of the protocol as a quantum channel, we will calculate the Kraus operators.

Any state on an open quantum systems can be extended to a closed quantum system by adding an environment to the principal quantum system. By comparing Eq. 4.1 with Eq. 2.29, we can relate ρ with the state of the environment and σ with the one of our principal system. Notice that in Eq. 4.1 the joint system is defined in $\mathcal{H}_\mathrm{e}\otimes\mathcal{H}_\mathrm{p}$ while in Eq. 2.29 it is defined in $\mathcal{H}_\mathrm{p}\otimes\mathcal{H}_\mathrm{e}$, where the label e and p stand for the environment and the main system, respectively. This is not relevant

as long as we make sure to trace over the environment. Within this comparison, the unitary operator $e^{-iS\Delta t}$ play the role of U the dynamics under which the joint system evolves.

In order to obtain the Kraus operators from Eq. 2.32, we need the environment system to be in a pure state. For this purpose, we use the purification result presented in Ref. [18] in which the purified state of ρ is

$$|\psi\rangle = (\mathbb{1} \otimes \sqrt{d\,\rho} V)\,|\Omega\rangle\,, \tag{4.15}$$

where d is the dimension of the Hilbert space, V is any unitary operator and $|\Omega\rangle$ is the maximally entangled state $|\Omega\rangle = \frac{1}{\sqrt{d}}\sum_{i=1}^{d}|ii\rangle$. We can see that, by tracing out the subspace that we have added, we indeed recover the original state

$$\begin{aligned}
\mathrm{Tr}_1\big(|\psi\rangle\langle\psi|\big) =& \mathrm{Tr}_1\Big[(\mathbb{1}\otimes\sqrt{d\rho}V)\,|\Omega\rangle\langle\Omega|\,(\mathbb{1}\otimes V^\dagger\sqrt{d\rho})\Big] \\
=& \mathrm{Tr}_1\Big[(\mathbb{1}\otimes\sqrt{d\rho}\,V)\sum_{ij}\frac{1}{d}\,|ii\rangle\langle jj|\,(\mathbb{1}\otimes V^\dagger\sqrt{d\rho})\Big] \\
=& \frac{1}{d}\sum_k\sum_{ij}(\langle k|\otimes\mathbb{1})\,(\mathbb{1}\otimes\sqrt{d\rho}\,V)\,(|i\rangle\otimes|i\rangle)\,(\langle j|\otimes\langle j|)\,(\mathbb{1}\otimes V^\dagger\sqrt{d\rho}) \\
& \qquad\cdot(|k\rangle\otimes\mathbb{1}) \\
=& \frac{1}{d}\sum_k\sum_{ij}\delta^i_k\sqrt{d\rho}V\,|i\rangle\langle j|\,V^\dagger\sqrt{d\rho}\,\delta^j_k = \sum_k\sqrt{\rho}\,V\,\underbrace{|k\rangle\langle k|}_{\mathbb{1}}\,V^\dagger\sqrt{\rho} \\
=& \sqrt{\rho}\,VV^\dagger\sqrt{\rho} = \rho.
\end{aligned} \tag{4.16}$$

Thus, our environment system will be the one formed by the union of the ρ system and the additional system that we have added for purification. Now, we can apply Eq. 2.29, obtaining for our case

$$\mathcal{E}(\sigma) = \sum_k \mathrm{Tr}_{\mathrm{env}}\Big(e^{iS\Delta t}\,\big(\sigma\otimes\sqrt{\rho}\,V\,|k\rangle\langle k|\,V^\dagger\sqrt{\rho}\big)\,e^{iS\Delta t}\Big), \tag{4.17}$$

where we have used the result of the proof in Eq. 4.16, $\mathrm{Tr}_1(|\psi\rangle\langle\psi|) = \sum_k\sqrt{\rho}V\,|k\rangle\langle k|\,V^\dagger\sqrt{\rho}$. Let us now make use of Eq. 2.32. However, it should be noticed that this is a more general case in which $\sum_k\sqrt{\rho}\,V\,|k\rangle$ plays the role of $|e_0\rangle$ and hence, our Kraus operators will now be labelled with two indexes

$$\begin{aligned}
E_{ik} &= (\mathbb{1}\otimes\langle i|_e)e^{-iS\Delta t}(\mathbb{1}\otimes\sqrt{\rho}\,V\,|k\rangle_e) \\
&= \cos(\Delta t)\,(\mathbb{1}\otimes\langle i|_e)\,\mathbb{1}_p\,(\mathbb{1}\otimes\sqrt{\rho}\,V\,|k\rangle_e) - i\sin(\Delta t)(\mathbb{1}\otimes\langle i|_e)\,S\,(\mathbb{1}\otimes\sqrt{\rho}\,V\,|k\rangle_e),
\end{aligned} \tag{4.18}$$

where both $|k\rangle_e$ and $|l\rangle_e$ live in the Hilbert space $\mathcal{H}_e$ of the environment, and $\mathbb{1}_p$ is the identity in the principal system subspace. Before we continue, let us formally define the SWAP operator as the unitary matrix $S\,|jl\rangle = |lj\rangle\ \forall j, l$, where $|j\rangle \in \mathcal{H}_p$ and $|l\rangle \in \mathcal{H}_e$ using the convention of $\mathcal{H}_p \otimes \mathcal{H}_e$. Hence, we can define

$$S = \sum_{j,l=0}^{d-1} |lj\rangle\langle jl|\,, \tag{4.19}$$

where $d = \dim \mathcal{H}_e = \dim \mathcal{H}_p$. It is important to notice that

$$S^2 = \sum_{j,l} |lj\rangle\langle jl| \sum_{j',l'} \left|l'j'\right\rangle\!\left\langle j'l'\right| = \sum_{l,j}\sum_{l',j'} |lj\rangle \underbrace{\langle jl|l'j'\rangle}_{\delta_j^{l'}\,\delta_l^{j'}} \left\langle j'l'\right| = \sum_{l,j} |lj\rangle\langle lj| = \mathbb{1}. \tag{4.20}$$

With this, we can rewrite

$$\begin{aligned}
(\mathbb{1}\otimes\langle i|_e)\,S\,(\mathbb{1}\otimes\sqrt{\rho}\,V\,|k\rangle_e) &= (\mathbb{1}\otimes\langle i|_e)\,S\,(\mathbb{1}\otimes\sqrt{\rho}\,V)(\mathbb{1}\otimes|k\rangle_e) \\
&= (\mathbb{1}\otimes\langle i|_e)\,\underbrace{S\,(\mathbb{1}\otimes\sqrt{\rho}\,V)\,S}_{(\sqrt{\rho}V\otimes\mathbb{1})}\,S\,(\mathbb{1}\otimes|k\rangle_e) \\
&= (\mathbb{1}\otimes\langle i|_e)(\sqrt{\rho}\,V\otimes\mathbb{1})\sum_{j,l}|lj\rangle\,\underbrace{\langle jl|\,(\mathbb{1}\otimes|k\rangle_e)}_{\langle j|_p\delta_k^l} \\
&= \sum_j \underbrace{(\sqrt{\rho}\,V\otimes\langle i|_e)\,|kj\rangle}_{\sqrt{\rho}\,V|k\rangle_p\delta_j^i}\langle j|_p = \sqrt{\rho}\,V\,|k\rangle\langle i|_p\,.
\end{aligned} \tag{4.21}$$

It is crucial to realise that the $|k\rangle\langle i|$ lives in the subspace $\mathcal{H}_p$, so we will denote this operator as $|k\rangle\langle i|_p$. On the other hand, we can also reshape the other term as

$$\cos(\Delta t)\,(\mathbb{1}\otimes\langle i|_e)\,\mathbb{1}_p\,(\mathbb{1}\otimes\sqrt{\rho}\,V\,|k\rangle_e) = \cos(\Delta t)\ \langle i|_e\,\sqrt{\rho}\,V\,|k\rangle_e\,\mathbb{1}_p. \tag{4.22}$$

This concludes the calculation of the Kraus operators, which allow us to express the channel in the operator-sum representation.

Result 2: Operator-sum representation of the LMR protocol

The operation presented in the LMR protocol can be interpreted as a quantum channel whose operator-sum representation $\mathcal{E} = \sum_{i,k=0}^{d_{\mathcal{H}_p}} E_{ik}\,\sigma\,E_{ik}^\dagger$ is defined by the Kraus operators

$$E_{ik} = \cos(\Delta t)\ \langle i|_e\,\sqrt{\rho}\,V\,|k\rangle_e\,\mathbb{1}_p - i\sin(\Delta t)\,\sqrt{\rho}\,V\,|k\rangle\langle i|_p\,, \tag{4.23}$$

where V is an arbitrary unitary coming from the purification of the environment system.

4.2 LMR protocol error

Once we have obtained the result of applying the channel to σ an arbitrary number of times is, we are ready to check that in order to simulate the exponential up to first order in Δt we must have a number of copies $n = t/\Delta t$.

We can use the following lemma, derived from the Baker–Campbell–Hausdorff formula [36]

$$e^{A} B e^{-A} = \sum_{k=0}^{\infty} \frac{1}{k!} [A, B]_k = B + [A, B] + \frac{1}{2!}[A, [A, B]] + ... \tag{4.24}$$

to compute the state of our system, initially described by σ, after evolving it with ρ for a time t

$$e^{-i\rho t}\, \sigma\, e^{i\rho t} = \sum_{k=0}^{\infty} \frac{1}{k!} [\, -i\rho t,\, \sigma\,]_k = \sigma - it\, [\rho, \sigma] + \mathcal{O}\big(t^2\big). \tag{4.25}$$

We have truncate the series at first order in t. Keeping the same order in Eq. 4.1, the result of applying the LMR protocol is

$$\mathrm{Tr}_1 \left(e^{-iSt}\, \rho \otimes \sigma\, e^{iSt}\right) = \sigma - it\, [\rho, \sigma] + \mathcal{O}\big(t^2\big). \tag{4.26}$$

This protocol coincides with the exponential up to an error $\mathcal{O}\big(\Delta t^2\big)$. However, if t is not very small, this error will be too large. By employing n copies of ρ, we can iteratively apply the protocol n times, each one for a time $\Delta t = t/n$. By using Eq. 4.5, we can calculate the output of applying the channel n times up to the contributions at Δt^2

$$\mathcal{E}_\rho^n(\sigma) = \sigma - i[\rho, \sigma]\, n\, \Delta t - \frac{1}{2}[\rho, \sigma]_2\, n(n-1)\, \Delta t^2 + (\rho - \sigma)\, n\, \Delta t^2 + \mathcal{O}\big(\Delta t^3\big), \tag{4.27}$$

where the sine and cosine series appropriately truncated for each term. If our target operation is

$$e^{-i\rho t}\, \sigma\, e^{i\rho t} = \sigma - i\, [\rho, \sigma]\, t - \frac{1}{2}[\rho, \sigma]_2\, t^2 + \mathcal{O}\big(t^3\big), \tag{4.28}$$

by choosing $n = t/\Delta t$ we manage to equalise the $n\, \Delta t$ contribution. In order to compare Eq. 4.27 and Eq. 4.28, we can calculate the trace distance between the two outputs, which is only half the trace norm (or norm 1) of the difference of the matrices

$$\begin{aligned} &\frac{1}{2}\big\| e^{-i\rho t}\, \sigma\, e^{i\rho t} - \mathcal{E}_\rho^n(\sigma) \big\|_1 \\ &\approx \frac{1}{2}\big\| -1/2\, [\rho, \sigma]_2\, t^2/n + (\sigma - \rho)\, t^2/n \big\|_1 \leq \mathcal{O}\big(t^2/n\big) = \mathcal{O}\big(n \Delta t^2\big). \end{aligned} \tag{4.29}$$

Hence, to obtain an error $\leq \epsilon$ we need $n = \mathcal{O}(t^2/\epsilon)$ copies of ρ as stated in Ref. [15]. Fig. 4.2 shows the dependence obtained in Eq. 4.29.

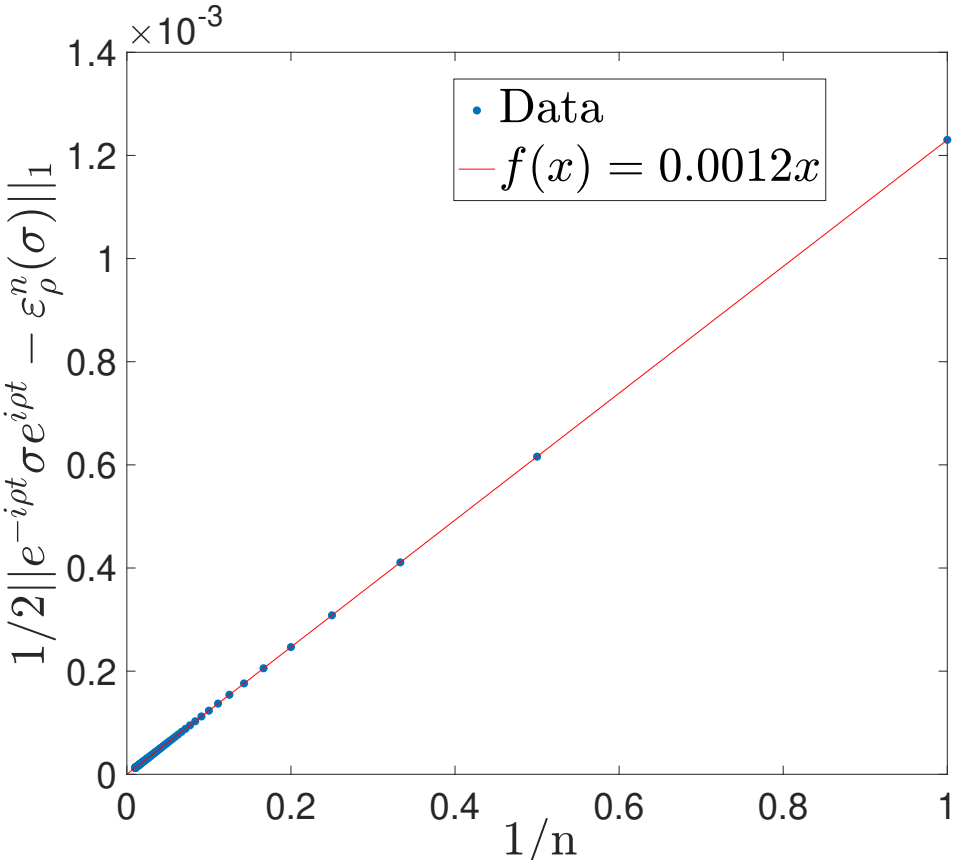

Figure 4.2: Trace norm of the difference between the target operation and the channel as a function of $1/n$ for $t = 0.1$ and random ρ and σ. The plot can be fitted to a straight line which matches the expected behaviour of $\mathcal{O}(t^2/n)$.

4.3 Is there a better density matrix independent protocol?

The LMR protocol holds for all density matrices ρ and σ, i.e. S is independent of ρ and σ. In this section we will explore the possibility of finding an enhanced protocol which is also independent of ρ and σ, but requires less copies of ρ to simulate the exponential up to the same error.

So far, we have considered the SWAP operation to carry out the $e^{-i\rho\,\Delta t}$ operation on σ up to $\mathcal{O}(\Delta t)$, for which we only need a single copy of ρ. Let us now consider a Hermitian matrix H, to check whether the SWAP operator is the only operator producing the same first order. We consider the LMR operation now up to $\mathcal{O}(\Delta t^3)$ and we expand it using Eq. 4.24

$$\mathrm{Tr}_1\big(e^{-iH\Delta t}\,\rho\otimes\sigma\,e^{iH\Delta t}\big) = \underset{\substack{\uparrow\\ A=-i\Delta t H \text{ and } B=\rho\otimes\sigma}}{\mathrm{Tr}_1\big(\rho\otimes\sigma - i\Delta t\,[H,\rho\otimes\sigma] - \frac{1}{2!}\Delta t^2\,[H,[H,\rho\otimes\sigma]]} + \mathcal{O}(\Delta t^3)\big). \tag{4.30}$$

Following the same idea, we write our target state as

$$e^{-i\rho\Delta t}\,\sigma\,e^{i\rho\Delta t} = \underset{\substack{\uparrow\\ A=-i\rho\Delta t \text{ and } B=\sigma}}{\sigma - i\Delta t\,[\rho,\sigma] - \frac{1}{2!}\Delta t^2\,[\rho,[\rho,\sigma]] + \mathcal{O}(\Delta t^3)\big).} \tag{4.31}$$

Identifying terms of the same order in both equations and using the linearity of the trace, we obtain two conditions,

$$1^{\text{st}}\text{ order}:\ \mathrm{Tr}_1[H,\rho\otimes\sigma] = [\rho,\sigma], \tag{4.32}$$

$$2^{\text{nd}}\text{ order}:\ \mathrm{Tr}_1[H,[H,\rho\otimes\sigma]] = [\rho,[\rho,\sigma]]. \tag{4.33}$$

4.3.1 First order condition

We will now analyse the first order equation to find the most general Hamiltonian H leading to the same first order. We start by expressing the density matrices in the canonical basis $\rho = \sum_{i_1,i_2=0}^{n-1}\rho_{i_1\,i_2}|i_1\rangle\langle i_2|$ and $\sigma = \sum_{j_1,j_2=0}^{n-1}\sigma_{j_1\,j_2}|j_1\rangle\langle j_2|$. Our equation transforms into

$$\mathrm{Tr}_1\Big(\sum_{i_1,i_2}\sum_{j_1,j_2}\rho_{i_1\,i_2}\,\sigma_{j_1\,j_2}[\,H\,,\,|i_1j_1\rangle\langle i_2j_2|\,]\Big) = \sum_{i_1,i_2}\sum_{j_1,j_2}\rho_{i_1\,i_2}\,\sigma_{j_1\,j_2}[\,|i_1\rangle\langle i_2|\,,\,|j_1\rangle\langle j_2|\,]. \tag{4.34}$$

Notice that we can compute the commutator of the RHS using orthonormality between quantum states $\langle a|b\rangle = \delta_a^b$ obtaining that $[\,|i_1\rangle\langle i_2|\,,\,|j_1\rangle\langle j_2|\,] = \delta_{j_1}^{i_2}|i_1\rangle\langle j_2| - \delta_{j_2}^{i_1}|j_1\rangle\langle i_2|$. In order to find a solution of H that holds for $\forall\,\rho,\sigma$, we derive on both

sides with respect to their matrix elements $\frac{\partial}{\partial\rho_{\alpha\beta}}\frac{\partial}{\partial\sigma_{\mu\lambda}}$, obtaining

$$\mathrm{Tr}_1\Big(\sum_{i_1,i_2}\sum_{j_1,j_2}\delta_\alpha^{i_1}\delta_\beta^{i_2}\delta_\mu^{j_1}\delta_\lambda^{j_2}[\,H\,,\,|i_1j_1\rangle\langle i_2j_2|\,]\Big)=\sum_{i_1,i_2}\sum_{j_1,j_2}\delta_\alpha^{i_1}\delta_\beta^{i_2}\delta_\mu^{j_1}\delta_\lambda^{j_2}[\,|i_1\rangle\langle i_2|\,,\,|j_1\rangle\langle j_2|\,], \tag{4.35}$$

where the Kronecker deltas allow us to remove the sums

$$\begin{aligned}&\mathrm{Tr}_1[H,|\alpha\mu\rangle\langle\beta\lambda|]=\delta_\mu^\beta|\alpha\rangle\langle\lambda|-\delta_\lambda^\alpha|\mu\rangle\langle\beta|\\&\longrightarrow\mathrm{Tr}_1[H,|i_1j_1\rangle\langle i_2j_2|]=\delta_{j_1}^{i_2}|i_1\rangle\langle j_2|-\delta_{j_2}^{i_1}|j_1\rangle\langle i_2|\\&\quad\uparrow\\&\text{Renaming indexes}\end{aligned} \tag{4.36}$$

Let us now expand the left-hand side of the last equation by expressing H in the canonical basis $H=\sum_{\alpha_1,\,\beta_1,\,\alpha_2,\,\beta_2=0}^{n-1}H_{\alpha_1\,\beta_1\,\alpha_2\,\beta_2}|\alpha_1\,\beta_1\rangle\langle\alpha_2\,\beta_2|$, which leads to

$$\begin{aligned}&\mathrm{Tr}_1[H,|i_1j_1\rangle\langle i_2j_2|]\\&=\sum_{k=0}^{n-1}\big(\langle k|\otimes\mathbb{1}\big)H|i_1\,j_1\rangle\underbrace{\langle i_2\,j_2|\big(|k\rangle\otimes\mathbb{1}\big)}_{\delta_k^{i_2}\langle j_2|}-\sum_{k=0}^{n-1}\underbrace{\big(\langle k|\otimes\mathbb{1}\big)H|i_1\,j_1\rangle}_{\delta_k^{i_1}|j_1\rangle}\langle i_2\,j_2|\big(|k\rangle\otimes\mathbb{1}\big)\\&=\big(\langle i_2|\otimes\mathbb{1}\big)H|i_1\,j_1\rangle\langle j_2|-|j_1\rangle\langle i_2\,j_2|H\big(|i_1\rangle\otimes\mathbb{1}\big)\\&=\sum_{\alpha_1,\,\beta_1,\,\alpha_2,\,\beta_2}\Big[\underbrace{\big(\langle i_2|\otimes\mathbb{1}\big)H_{\alpha_1\,\beta_1\,\alpha_2\,\beta_2}|\alpha_1\,\beta_1\rangle}_{\delta_{i_2}^{\alpha_1}H_{\alpha_1\,\beta_1\,\alpha_2\,\beta_2}|\beta_1\rangle}\underbrace{\langle\alpha_2\,\beta_2|i_1\,j_1\rangle}_{\delta_{i_1}^{\alpha_2}\delta_{j_1}^{\beta_2}}\langle j_2|\\&\qquad\qquad-|j_1\rangle\underbrace{\langle i_2\,j_2|H_{\alpha_1\,\beta_1\,\alpha_2\,\beta_2}|\alpha_1\,\beta_1\rangle}_{\delta_{i_2}^{\alpha_1}\delta_{j_2}^{\beta_1}H_{\alpha_1\,\beta_1\,\alpha_2\,\beta_2}}\underbrace{\langle\alpha_2\,\beta_2|\big(|i_1\rangle\otimes\mathbb{1}\big)}_{\delta_{i_1}^{\alpha_2}\langle\beta_2|}\Big]\\&=\sum_{\beta_1}H_{i_2\,\beta_1\,i_1\,j_1}|\beta_1\rangle\langle j_2|-\sum_{\beta_2}H_{i_2\,j_2\,i_1\,\beta_2}|j_1\rangle\langle\beta_2|\\&=\sum_{\beta}\Big(H_{i_2\,\beta\,i_1\,j_1}|\beta_1\rangle\langle j_2|-H_{i_2\,j_2\,i_1\,\beta}|j_1\rangle\langle\beta|\Big),\end{aligned} \tag{4.37}$$

where, in the last step, we have just renamed the dummy indexes. Using the result of this calculation, putting it together with the RHS of the Eq.4.32, and rearranging the different terms, we obtain

$$\Big(\sum_\beta H_{i_2\,\beta\,i_1\,j_1}|\beta\rangle-\delta_{j_1}^{i_2}|i_1\rangle\Big)\langle j_2|=|j_1\rangle\Big(\sum_\beta H_{i_2\,j_2\,i_1\,\beta}\langle\beta|-\delta_{j_2}^{i_1}\langle i_2|\Big). \tag{4.38}$$

To solve this equation the "kets" on the left side must be proportional to those on the right and the same for the "bras" on both sides with the same proportionality

factor. A priori we could think that this factor may depend on the four indexes. However, notice that in the condition imposed by the kets there is no dependence on j_2 while in the condition imposed by the bras there is no dependence on j_1. That is the reason why the proportionality factor can only depend on i_1 and i_2 for the conditions to be consistent. Let us name it as $x_{i_1 i_2}$. Hence, we obtain

$$\begin{cases} \sum_\beta H_{i_2\,\beta\,i_1\,j_1}|\beta\rangle - \delta^{i_2}_{j_1}|i_1\rangle = x_{i_1 i_2}|j_1\rangle \xrightarrow{\cdot\langle\alpha|} H_{i_2\,\alpha\,i_1\,j_1} - \delta^{i_2}_{j_1}\delta^{i_1}_{\alpha} = x_{i_1 i_2}\,\delta^{j_1}_{\alpha}, \\ \langle j_2|x_{i_1 i_2} = \sum_\beta H_{i_2\,j_2\,i_1\,\beta}\langle\beta| - \delta^{i_1}_{j_2}\langle i_2| \xrightarrow{\cdot|\alpha\rangle} H_{i_2\,\alpha\,i_1\,j_2} - \delta^{i_2}_{j_2}\delta^{i_1}_{\alpha} = x_{i_1 i_2}\,\delta^{j_2}_{\alpha}, \end{cases} \tag{4.39}$$

where the second equation is the same as the first one.

Result 3: Generalisation of the LMR protocol

The family of operators described by $H = \sum_{i_2 j_2 i_1 j_1} H_{i_2\,j_2\,i_1\,j_1}\,|i_2 j_2\rangle\langle i_1 j_1|$ with

$$H_{i_2\,j_2\,i_1\,j_1} = x_{i_1 i_2}\,\delta^{j_1}_{j_2} + \delta^{i_2}_{j_1}\delta^{i_1}_{j_2}, \tag{4.40}$$

where the coefficients $x_{i_1 i_2}$ are arbitrary but chosen complex or real ensuring that H is Hermitian, can be used in the LMR protocol instead of the SWAP gate to load ρ to first order in Δt.

Notice that having considered an arbitrary dimension, the indexes can take any value between 0 and $2^n - 1$.

Now let us consider the simplest case comprising two qubits. Hence, ρ and σ are 2×2 matrices describing the state of the qubits and H is a 4×4 matrix. Notice that, in this case, the indexes take values 0 or 1. The computation of the matrix elements is shown in Table 4.1. Expressing H in the computational basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$

$$H = \begin{pmatrix} H_{0000} & H_{0001} & H_{0010} & H_{0011} \\ H_{0100} & H_{0101} & H_{0110} & H_{0111} \\ H_{1000} & H_{1001} & H_{1010} & H_{1011} \\ H_{1100} & H_{1101} & H_{1110} & H_{1111} \end{pmatrix} = \begin{pmatrix} x_{00}+1 & 0 & x_{10} & 0 \\ 0 & x_{00} & 1 & x_{10} \\ x_{01} & 1 & x_{11} & 0 \\ 0 & x_{01} & 0 & x_{11}+1 \end{pmatrix}. \tag{4.41}$$

If we impose H to be Hermitian ($H = H^\dagger$), which we have not done so far, we obtain that $x_{00}, x_{11} \in \mathbb{R}$ and $x_{10} = x^*_{01}$. It should be noticed that, if we take $x_{00} = x_{11} = x_{01} = 0$, we recover the two qubits SWAP gate. However, we see that this is not the only matrix which allows us to perform the desired operation up to first order Δt, but there is a 4-parametric family of Hamiltonians leading to the same result.

i_2	j_2	i_1	j_1	$H_{i_2 j_2 i_1 j_1}$
0	0	0	0	$H_{0000} = x_{00} + 1$
0	0	0	1	$H_{0001} = 0$
0	0	1	0	$H_{0010} = x_{10}$
0	0	1	1	$H_{0011} = 0$
0	1	0	0	$H_{0100} = 0$
0	1	0	1	$H_{0101} = x_{00}$
0	1	1	0	$H_{0110} = 1$
0	1	1	1	$H_{0111} = x_{10}$
1	0	0	0	$H_{1000} = x_{01}$
1	0	0	1	$H_{1001} = 1$
1	0	1	0	$H_{1010} = x_{11}$
1	0	1	1	$H_{1011} = 0$
1	1	0	0	$H_{1100} = 0$
1	1	0	1	$H_{1101} = x_{01}$
1	1	1	0	$H_{1110} = 0$
1	1	1	1	$H_{1111} = x_{11} + 1$

Table 4.1: Matrix elements of the two qubit gate H

4.3.2 Second order condition

The next step is to find if within the family of solutions found in previous subsection, there is any solution that also satisfies the second-order equation for any ρ and σ. Looking at Eq. 4.33, we can see that the equation is of degree 1 in ρ on the LHS and of degree 2 in ρ on the RHS. However, the search must be done in the set of density matrices, so we rewrite ρ as

$$\rho = \frac{A^\dagger A}{\mathrm{Tr}(A^\dagger A)}, \tag{4.42}$$

then the second orden condition reads

$$\mathrm{Tr}(A^\dagger A)\,\mathrm{Tr}_1\big([H,\,[H, A^\dagger A \otimes \sigma]]\big) = [A^\dagger A,\,[A^\dagger A, \sigma]]. \tag{4.43}$$

Now let us work on both sides of the equation separately. As we have a result for any ρ and σ we will perform the following derivatives in both sides

$$\frac{\partial^2}{\partial A^\dagger_{l_1 l_2}\,\partial A^\dagger_{k_1 k_2}}\frac{\partial^2}{\partial A_{j_1 j_2}\,\partial A_{i_1 i_2}}\frac{\partial}{\partial \sigma_{n_1 \mathtt{n}_2}}, \tag{4.44}$$

where, in this case, instead of deriving with respect to ρ, we derive with respect to A and $A^\dagger$.

Right-hand side

If we express $\sigma = \sum_{m_1,m_2=0}^{n-1} \sigma_{m_1,m_2}|m_1\rangle\langle m_2|$, when we make the derivative on this side of the equation with respect to the sigma matrix elements, we obtain

$$\frac{\partial\sigma}{\partial\sigma_{n_1\mathfrak{n}_2}} = \sum_{m_1,m_2=0}^{n-1} \delta_{n_1}^{m_1}\delta_{n_2}^{m_2}|m_1\rangle\langle m_2| = |n_1\rangle\langle n_2| \equiv P_{n_1n_2}, \tag{4.45}$$

where the notation $P_{n_1n_2}$ for the projectors will be used from now. We rewrite the RHS of the equation after this first derivative as

$$[A^\dagger A,\, [A^\dagger A, P_{n_1n_2}]] = A^\dagger AA^\dagger AP_{n_1n_2} + P_{n_1n_2}A^\dagger AA^\dagger A - 2A^\dagger AP_{n_1n_2}A^\dagger A. \tag{4.46}$$

Notice that we have simply introduced the derivative into the commutator. Now we perform the derivatives with respect to A and $A^\dagger$ which will also give us projectors for the same reasoning as before. We start by $\partial/\partial A_{i_1i_2}$, which will split each term into two leading to

$$\begin{aligned}&(A^\dagger P_{i_1i_2}A^\dagger A + A^\dagger AA^\dagger P_{i_1i_2})P_{n_1n_2} + P_{n_1n_2}(A^\dagger P_{i_1i_2}A^\dagger A + A^\dagger AA^\dagger P_{i_1i_2})\\ &- 2\,(A^\dagger P_{i_1i_2}P_{n_1n_2}A^\dagger A + A^\dagger AP_{n_1n_2}A^\dagger P_{i_1i_2}).\end{aligned} \tag{4.47}$$

We continue with the derivative $\partial/\partial A_{j_1j_2}$ which is very straightforward as we only have to substitute the remaining A's by $P_{j_1j_2}$ obtaining

$$\begin{aligned}&(A^\dagger P_{i_1i_2}A^\dagger P_{j_1j_2} + A^\dagger P_{j_1j_2}A^\dagger P_{i_1i_2})P_{n_1n_2} + P_{n_1n_2}(A^\dagger P_{i_1i_2}A^\dagger P_{j_1j_2} + A^\dagger P_{j_1j_2}A^\dagger P_{i_1i_2})\\ &- 2\,(A^\dagger P_{i_1i_2}P_{n_1n_2}A^\dagger P_{j_1j_2} + A^\dagger P_{j_1j_2}P_{n_1n_2}A^\dagger P_{i_1i_2}).\end{aligned} \tag{4.48}$$

Now, we compute $\partial/\partial A^\dagger_{k_1k_2}$, which again will split each term into two as follows

$$\begin{aligned}&(P_{k_1k_2}P_{i_1i_2}A^\dagger P_{j_1j_2} + A^\dagger P_{i_1i_2}P_{k_1k_2}P_{j_1j_2} + P_{k_1k_2}P_{j_1j_2}A^\dagger P_{i_1i_2}\\ &\quad + A^\dagger P_{j_1j_2}P_{k_1k_2}P_{i_1i_2})P_{n_1n_2}\\ &+P_{n_1n_2}(P_{k_1k_2}P_{i_1i_2}A^\dagger P_{j_1j_2} + A^\dagger P_{i_1i_2}P_{k_1k_2}P_{j_1j_2} + P_{k_1k_2}P_{j_1j_2}A^\dagger P_{i_1i_2}\\ &\quad + A^\dagger P_{j_1j_2}P_{k_1k_2}P_{i_1i_2})\\ &-2\,(P_{k_1k_2}P_{i_1i_2}P_{n_1n_2}A^\dagger P_{j_1j_2} + A^\dagger P_{i_1i_2}P_{n_1n_2}P_{k_1k_2}P_{j_1j_2} + P_{k_1k_2}P_{j_1j_2}P_{n_1n_2}A^\dagger P_{i_1i_2}\\ &\quad + A^\dagger P_{j_1j_2}P_{n_1n_2}P_{k_1k_2}P_{i_1i_2}).\end{aligned} \tag{4.49}$$

Finally, we compute $\partial/\partial A^\dagger_{l_1 l_2}$, which is again obtained by replacing the $A^\dagger$'s by $P_{l_1 l_2}$. Taking into account the orthonormality of the states, the product of the projectors will be reduced to the product of Kronecker deltas times a projector. Making this computation and regrouping terms the right-hand side becomes

$$\begin{aligned}
&|k_1\rangle\langle n_2| \left(\delta^{k_2}_{i_1}\delta^{i_2}_{l_1}\delta^{l_2}_{j_1}\delta^{j_2}_{n_1} + \delta^{k_2}_{j_1}\delta^{j_2}_{l_1}\delta^{l_2}_{i_1}\delta^{i_2}_{n_1}\right) + |l_1\rangle\langle n_2| \left(\delta^{l_2}_{i_1}\delta^{i_2}_{k_1}\delta^{k_2}_{j_1}\delta^{j_2}_{n_1} + \delta^{l_2}_{j_1}\delta^{j_2}_{k_1}\delta^{k_2}_{i_1}\delta^{i_2}_{n_1}\right) \\
&|n_1\rangle\langle j_2| \left(\delta^{n_2}_{k_1}\delta^{k_2}_{i_1}\delta^{i_2}_{l_1}\delta^{l_2}_{j_1} + \delta^{n_2}_{l_1}\delta^{l_2}_{i_1}\delta^{i_2}_{k_1}\delta^{k_2}_{j_1}\right) + |n_1\rangle\langle i_2| \left(\delta^{n_2}_{k_1}\delta^{k_2}_{j_1}\delta^{j_2}_{l_1}\delta^{l_2}_{i_1} + \delta^{n_2}_{l_1}\delta^{l_2}_{j_1}\delta^{j_2}_{k_1}\delta^{k_2}_{i_1}\right) \\
&-2\Big(|k_1\rangle\langle j_2|\delta^{k_2}_{i_1}\delta^{i_2}_{n_1}\delta^{n_2}_{l_1}\delta^{l_2}_{j_1} + |l_1\rangle\langle j_2|\delta^{l_2}_{i_1}\delta^{i_2}_{n_1}\delta^{n_2}_{k_1}\delta^{k_2}_{j_1} + |k_1\rangle\langle i_2|\delta^{k_2}_{j_1}\delta^{j_2}_{n_1}\delta^{n_2}_{l_1}\delta^{l_2}_{i_1} \\
&\qquad + |l_1\rangle\langle i_2|\delta^{l_2}_{j_1}\delta^{j_2}_{n_1}\delta^{n_2}_{k_1}\delta^{k_2}_{i_1}\Big).
\end{aligned} \tag{4.50}$$

Left-hand side

Let us tart by computing the derivative with respect to $\sigma_{n_1 n_2}$, i.e. by substituting it by $P_{n_1 n_2}$. Then, we compute the first derivative with respect to $A_{i_1 i_2}$. This splits the LHS of the equation into two terms and makes the $P_{i_1 i_2}$ projectors appear

$$\begin{aligned}
&\mathrm{Tr}(A^\dagger P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger A \otimes P_{n_1 n_2}]]\Big) \\
&+ \mathrm{Tr}(A^\dagger A)\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger P_{i_1 i_2} \otimes P_{n_1 n_2}]]\Big).
\end{aligned} \tag{4.51}$$

Now, we perform the second derivative with respect to $A_{j_1 j_2}$ which leads to

$$\begin{aligned}
&\mathrm{Tr}(A^\dagger P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger P_{j_1 j_2} \otimes P_{n_1 n_2}]]\Big) \\
&+ \mathrm{Tr}(A^\dagger P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger P_{i_1 i_2} \otimes P_{n_1 n_2}]]\Big).
\end{aligned} \tag{4.52}$$

The first derivative with respect to $(A^\dagger)_{k_1 k_2}$ will split each term into two again obtaining

$$\begin{aligned}
&\mathrm{Tr}(P_{k_1 k_2} P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger P_{j_1 j_2} \otimes P_{n_1 n_2}]]\Big) \\
&+ \mathrm{Tr}(A^\dagger P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, P_{k_1 k_2} P_{j_1 j_2} \otimes P_{n_1 n_2}]]\Big) \\
&+ \mathrm{Tr}(P_{k_1 k_2} P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, A^\dagger P_{j_1 j_2} \otimes P_{n_1 n_2}]]\Big) \\
&+ \mathrm{Tr}(A^\dagger P_{i_1 i_2})\ \mathrm{Tr}_1\Big([H,\ [H, P_{k_1 k_2} P_{j_1 j_2} \otimes P_{n_1 n_2}]]\Big).
\end{aligned} \tag{4.53}$$

The last derivative is just replacing the remaining $A^\dagger$ terms by $P_{l_1 l_2}$. We can compute the global traces as

$$\mathrm{Tr}(P_{k_1 k_2} P_{k_1 k_2}) = \sum_t \langle t|k_1\rangle\langle k_2|i_1\rangle\langle i_2|t\rangle = \sum_t \delta^t_{k_1}\delta^{k_2}_{i_1}\delta^t_{i_2} = \delta^{k_1}_{i_2}\delta^{k_2}_{i_1}. \tag{4.54}$$

Then, the LHS transforms into

$$\delta_{i_2}^{k_1}\delta_{i_1}^{k_2}\delta_{l_2}^{j_1}\,\mathrm{Tr}_1\Big(\big[H,\,[H,|l_1n_1\rangle\langle j_2n_2|\,]\big]\Big)+\delta_{i_2}^{l_1}\delta_{i_1}^{l_2}\delta_{k_2}^{j_1}\,\mathrm{Tr}_1\Big(\big[H,\,[H,|k_1n_1\rangle\langle j_2n_2|\,]\big]\Big)$$
$$+\delta_{j_2}^{k_1}\delta_{j_1}^{k_2}\delta_{l_2}^{i_1}\,\mathrm{Tr}_1\Big(\big[H,\,[H,|l_1n_1\rangle\langle i_2n_2|\,]\big]\Big)+\delta_{j_2}^{l_1}\delta_{j_1}^{l_2}\delta_{k_2}^{i_1}\,\mathrm{Tr}_1\Big(\big[H,\,[H,|k_1n_1\rangle\langle i_2n_2|\,]\big]\Big). \tag{4.55}$$

As we have four terms with the same structure, let us now focus on computing the partial trace for a general case $\mathrm{Tr}_1\Big(\big[H,\,[H,|a_1b_1\rangle\langle a_2b_2|\,]\big]\Big)$. We will introduce the condition that H must satisfy to cancel the first order found above. We will write the two H as

$$H=\sum_{\alpha_1\alpha_2}\sum_{\beta_1\beta_2}(x_{\alpha_1\alpha_2}\delta_{\beta_1}^{\beta_2}+\delta_{\beta_1}^{\alpha_2}\delta_{\beta_2}^{\alpha_1})\,|\alpha_1\beta_1\rangle\langle\alpha_2\beta_2|, \tag{4.56}$$

$$H=\sum_{\lambda_1\lambda_2}\sum_{\mu_1\mu_2}(x_{\lambda_1\lambda_2}\delta_{\mu_1}^{\mu_2}+\delta_{\mu_1}^{\lambda_2}\delta_{\mu_2}^{\lambda_1})\,|\lambda_1\mu_1\rangle\langle\lambda_2\mu_2|, \tag{4.57}$$

where the indexes run from 0 to $n-1$. Now, we go through the computation step by step. We start by the commutator

$$\begin{aligned}\big[H,|a_1b_1\rangle\langle a_2b_2|\,\big]=&H\,|a_1b_1\rangle\langle a_2b_2|-|a_1b_1\rangle\langle a_2b_2|\,H\\=&\sum_{\alpha_1\alpha_2\beta_1\beta_2}(x_{\alpha_1\alpha_2}\delta_{\beta_1}^{\beta_2}+\delta_{\beta_1}^{\alpha_2}\delta_{\beta_2}^{\alpha_1})\,\delta_{a_1}^{\alpha_2}\delta_{b_1}^{\beta_2}\,|\alpha_1\beta_1\rangle\langle a_2b_2|\\&-\sum_{\alpha_1\alpha_2\beta_1\beta_2}(x_{\alpha_1\alpha_2}\delta_{\beta_1}^{\beta_2}+\delta_{\beta_1}^{\alpha_2}\delta_{\beta_2}^{\alpha_1})\,\delta_{a_2}^{\alpha_1}\delta_{b_2}^{\beta_1}\,|a_1b_1\rangle\langle\alpha_2\beta_2|\\=&\sum_{\alpha_1\beta_1}(x_{\alpha_1a_1}\delta_{\beta_1}^{b_1}+\delta_{\beta_1}^{a_1}\delta_{b_1}^{\alpha_1})\,|\alpha_1\beta_1\rangle\langle a_2b_2|\\&-\sum_{\alpha_2\beta_2}(x_{a_2\alpha_2}\delta_{b_2}^{\beta_2}+\delta_{b_2}^{\alpha_2}\delta_{\beta_2}^{a_2})\,|a_1b_1\rangle\langle\alpha_2\beta_2|.\end{aligned} \tag{4.58}$$

With this, we can calculate the second order commutator

$$\begin{aligned}\big[H,\,[H,|a_1b_1\rangle\langle a_2b_2|\,]\big]=&H\,\big[H,|a_1b_1\rangle\langle a_2b_2|\,\big]-\big[H,|a_1b_1\rangle\langle a_2b_2|\,\big]\,H\\=&\sum_{\alpha_1\beta_1}\sum_{\lambda_1\lambda_2\mu_1\mu_2}(x_{\alpha_1a_1}\delta_{\beta_1}^{b_1}+\delta_{\beta_1}^{a_1}\delta_{b_1}^{\alpha_1})(x_{\lambda_1\lambda_2}\delta_{\mu_1}^{\mu_2}+\delta_{\mu_1}^{\lambda_2}\delta_{\mu_2}^{\lambda_1})\times\\&\qquad\big(\delta_{\lambda_2}^{\alpha_1}\delta_{\mu_2}^{\beta_1}|\lambda_1\mu_1\rangle\langle a_2b_2|-\delta_{\lambda_1}^{a_2}\delta_{\mu_1}^{b_2}|\alpha_1\beta_1\rangle\langle\lambda_2\mu_2|\big)\\&-\sum_{\alpha_2\beta_2}\sum_{\lambda_1\lambda_2\mu_1\mu_2}(x_{a_2\alpha_2}\delta_{b_2}^{\beta_2}+\delta_{b_2}^{\alpha_2}\delta_{\beta_2}^{a_2})(x_{\lambda_1\lambda_2}\delta_{\mu_1}^{\mu_2}+\delta_{\mu_1}^{\lambda_2}\delta_{\mu_2}^{\lambda_1})\times\\&\qquad\big(\delta_{\lambda_2}^{a_1}\delta_{\mu_2}^{b_1}|\lambda_1\mu_1\rangle\langle\alpha_2\beta_2|-\delta_{\lambda_1}^{\alpha_2}\delta_{\mu_1}^{\beta_2}|a_1b_1\rangle\langle\lambda_2\mu_2|\big).\end{aligned} \tag{4.59}$$

We can now reduce some sums by making use of the Kronecker deltas. We will also rename the dummy indexes to have the same in all terms so that finally the second-order commutator turns out to be

$$\begin{aligned}
&\sum_{\alpha_1\beta_1}(x_{\alpha_1 a_1}\delta^{b_1}_{\beta_1}+\delta^{a_1}_{\beta_1}\delta^{\alpha_1}_{b_1})\sum_{\lambda_1\mu_1}(x_{\lambda_1\alpha_1}\delta^{\beta_1}_{\mu_1}+\delta^{\alpha_1}_{\mu_1}\delta^{\lambda_1}_{\beta_1})|\lambda_1\mu_1\rangle\langle a_2 b_2| \\
&-\sum_{\alpha_1\beta_1}(x_{\alpha_1 a_1}\delta^{b_1}_{\beta_1}+\delta^{a_1}_{\beta_1}\delta^{\alpha_1}_{b_1})\sum_{\lambda_1\mu_1}(x_{a_2\lambda_1}\delta^{\mu_1}_{b_2}+\delta^{\lambda_1}_{b_2}\delta^{a_2}_{\mu_1})|\alpha_1\beta_1\rangle\langle\lambda_1\mu_1| \\
&-\sum_{\alpha_1\beta_1}(x_{a_2\alpha_1}\delta^{\beta_1}_{b_2}+\delta^{\alpha_1}_{b_2}\delta^{a_2}_{\beta_1})\sum_{\lambda_1\mu_1}(x_{\lambda_1 a_1}\delta^{b_1}_{\mu_1}+\delta^{a_1}_{\mu_1}\delta^{\lambda_1}_{b_1})|\lambda_1\mu_1\rangle\langle\alpha_1\beta_1| \\
&+\sum_{\alpha_1\beta_1}(x_{a_2\alpha_1}\delta^{\beta_1}_{b_2}+\delta^{\alpha_1}_{b_2}\delta^{a_2}_{\beta_1})\sum_{\lambda_1\mu_1}(x_{\alpha_1\lambda_1}\delta^{\mu_1}_{\beta_1}+\delta^{\lambda_1}_{\beta_1}\delta^{\alpha_1}_{\mu_1})|a_1 b_1\rangle\langle\lambda_1\mu_1|.
\end{aligned} \tag{4.60}$$

Now, we compute the partial trace of this term using $\mathrm{Tr}_1|ab\rangle\langle cd| = \delta^a_c|b\rangle\langle d|$

$$\begin{aligned}
&\mathrm{Tr}_1\big[H,\,[H,|a_1b_1\rangle\langle a_2b_2|\,]\big] \\
&=\sum_{\alpha_1\beta_1}(x_{\alpha_1 a_1}\delta^{b_1}_{\beta_1}+\delta^{a_1}_{\beta_1}\delta^{\alpha_1}_{b_1})\Bigg[\sum_{\lambda_1\mu_1}(x_{\lambda_1\alpha_1}\delta^{\beta_1}_{\mu_1}+\delta^{\alpha_1}_{\mu_1}\delta^{\lambda_1}_{\beta_1})\delta^{\lambda_1}_{a_2}|\mu_1\rangle\langle b_2| \\
&\qquad\qquad -\sum_{\lambda_1\mu_1}(x_{a_2\lambda_1}\delta^{\mu_1}_{b_2}+\delta^{\alpha_1}_{b_2}\delta^{a_2}_{\mu_1})\delta^{\alpha_1}_{\lambda_1}|\beta_1\rangle\langle\mu_1|\Bigg] \\
&-\sum_{\alpha_1\beta_1}(x_{a_2\alpha_1}\delta^{\beta_1}_{b_2}+\delta^{b_2}_{\alpha_1}\delta^{a_2}_{\beta_1})\Bigg[\sum_{\lambda_1\mu_1}(x_{\lambda_1 a_1}\delta^{b_1}_{\mu_1}+\delta^{a_1}_{\mu_1}\delta^{\lambda_1}_{b_1})\delta^{\lambda_1}_{\alpha_1}|\mu_1\rangle\langle\beta_1| \\
&\qquad\qquad -\sum_{\lambda_1\mu_1}(x_{\alpha_1\lambda_1}\delta^{\mu_1}_{\beta_1}+\delta^{\lambda_1}_{\beta_1}\delta^{\alpha_1}_{\mu_1})\delta^{\lambda_1}_{a_1}|b_1\rangle\langle\mu_1|\Bigg].
\end{aligned} \tag{4.61}$$

To conclude this calculation, we compute sums by employing Kronecker deltas, we gather terms together, and we use that $|a_1\rangle\langle b_1| = \delta^{\alpha_1}_{a_1}|\alpha_1\rangle\langle b_1|$ to rewrite the equation in a more compact form. Finally, we get

$$\begin{aligned}
&\mathrm{Tr}_1\big[H,\,[H,|a_1b_1\rangle\langle a_2b_2|\,]\big] \\
&=\sum_{\alpha_1}(x_{\alpha_1 a_1}\delta^{b_1}_{a_2}-x_{a_2 b_1}\delta^{\alpha_1}_{a_1})|\alpha_1\rangle\langle b_2|+\sum_{\alpha_1}(x_{a_2\alpha_1}\delta^{a_1}_{b_2}-x_{b_2 a_1}\delta^{\alpha_1}_{a_2})|b_1\rangle\langle\alpha_1| \\
&\quad+2\delta^{a_1}_{a_2}|b_1\rangle\langle b_2|-2\delta^{b_1}_{b_2}|a_1\rangle\langle a_2|.
\end{aligned} \tag{4.62}$$

Going back to Eq. 4.55, together with Eq. 4.62, we can collect terms together to get the final expression for the left-hand side:

$$
\begin{aligned}
&\delta_{i_1}^{k_2}\delta_{j_1}^{l_2}\Bigg[\delta_{k_1}^{i_2}\sum_{\alpha}\Big((x_{\alpha l_1}\delta_{j_2}^{n_1}-x_{j_2 n_1}\delta_{l_1}^{\alpha})|\alpha\rangle\langle n_2|+(x_{j_2\alpha}\delta_{n_2}^{l_1}-x_{n_2 l_1}\delta_{j_2}^{\alpha})|n_1\rangle\langle\alpha|\Big)\\
&\quad+\delta_{l_1}^{j_2}\sum_{\alpha}\Big((x_{\alpha k_1}\delta_{i_2}^{n_1}-x_{i_2 n_1}\delta_{k_1}^{\alpha})|\alpha\rangle\langle n_2|+(x_{i_2\alpha}\delta_{n_2}^{k_1}-x_{n_2 k_1}\delta_{i_2}^{\alpha})|n_1\rangle\langle\alpha|\Big)\\
&\quad+4\delta_{k_1}^{i_2}\delta_{j_2}^{l_1}|n_1\rangle\langle n_2|-2\delta_{n_2}^{n_1}\left(\delta_{k_1}^{i_2}|l_1\rangle\langle j_2|+\delta_{l_1}^{j_2}|k_1\rangle\langle i_2|\right)\Bigg]\\
+&\delta_{i_1}^{l_2}\delta_{j_1}^{k_2}\Bigg[\delta_{l_1}^{i_2}\sum_{\alpha}\Big((x_{\alpha k_1}\delta_{j_2}^{n_1}-x_{j_2 n_1}\delta_{k_1}^{\alpha})|\alpha\rangle\langle n_2|+(x_{j_2\alpha}\delta_{n_2}^{k_1}-x_{n_2 k_1}\delta_{j_2}^{\alpha})|n_1\rangle\langle\alpha|\Big)\\
&\quad+\delta_{k_1}^{j_2}\sum_{\alpha}\Big((x_{\alpha l_1}\delta_{i_2}^{n_1}-x_{i_2 n_1}\delta_{l_1}^{\alpha})|\alpha\rangle\langle n_2|+(x_{i_2\alpha}\delta_{n_2}^{l_1}-x_{n_2 l_1}\delta_{i_2}^{\alpha})|n_1\rangle\langle\alpha|\Big)\\
&\quad+4\,\delta_{l_1}^{i_2}\delta_{j_2}^{k_1}|n_1\rangle\langle n_2|-2\delta_{n_2}^{n_1}\left(\delta_{l_1}^{i_2}|k_1\rangle\langle j_2|+\delta_{k_1}^{j_2}|l_1\rangle\langle i_2|\right)\Bigg].
\end{aligned}
\tag{4.63}
$$

Finally, we must find a solution in coefficients x's for the equation resulting from the equality between expressions 4.50 and 4.63. However, there is no solution for this set of equations. To show this, we will just look for two incompatible equations. Taking the matrix elements $|l_1\rangle\langle j_2|$ and $|l_1\rangle\langle i_2|$, if we choose the particular case in which $j_2 \neq i_2$, we have the following conditions

$$\delta_{i_1}^{l_2}\delta_{n_1}^{i_2}\delta_{k_1}^{n_2}\delta_{j_1}^{k_2}=\delta_{i_1}^{k_2}\delta_{j_1}^{l_2}\delta_{n_2}^{n_1}\delta_{k_1}^{i_2}, \tag{4.64}$$

$$\delta_{j_1}^{l_2}\delta_{n_1}^{j_2}\delta_{k_1}^{n_2}\delta_{i_1}^{k_2}=\delta_{i_1}^{l_2}\delta_{j_1}^{k_2}\delta_{n_2}^{n_1}\delta_{k_1}^{j_2}. \tag{4.65}$$

Notice that these equations do not depend on the parameters x's. Joining the assumption that Eq. 4.64 is true with the initial hypothesis we get that $n_1 = i_2 \neq j_2$. By imposing this in Eq. 4.65, the LHS is equal to 0. So, if we now have that it is fulfilled $l_2 = i_1$, $k_2 = j_1$, $i_2 = n_2$ and $j_2 = k_1$, this second equation becomes $0 = 1$ which is a contradiction. Therefore, as these equations are incompatible for at least one case, we can assert that no choice of H can cancel the second order.

Result 4: The operation of the LMR protocol is optimal

There is no Hermitian matrix H independent of ρ and σ which cancels the second order expansion of the LMR for any ρ and σ.

4.3.3 Norm of the second order term for the two-qubit case

Although we cannot eliminate the term in Δt^2, let us try to see if we can at least reduce its norm for the two-qubit case. This will be done by optimising the degrees of freedom of H. We have numerically computed this by employing MATLAB's genetic optimisation algorithm [37]. This kind of algorithm is based on mimicking the process of natural selection in a population. This population would be made up of the solutions to our problem, which are modified at each step. During a step, solutions are randomly selected, which would act as parents, giving rise to new ones, which would be the children. In this manner, after each generation, the population (the family of solutions) evolves towards the optimal one. In this type of simulations, a cost function is defined, which will depend on some parameters on which those that maximise or minimise this function will be sought. In our case, the cost function will be the norm 1 of the difference of the second-order terms in Δt

$$f_{\text{cost}} = \| \operatorname{Tr}_1[H,[H,\rho\otimes\sigma]] - [\rho,[\rho,\sigma]] \|_1. \tag{4.66}$$

For the selection of density matrices ρ and σ, we have also used a genetic algorithm to search for those that maximise the norm when $H = \text{SWAP}$ where the free parameters in this case are the matrix elements. Once these matrices have been selected, we pass them as input to the second genetic algorithm, which tries to minimise the norm by modifying the free parameters that we have in the selection of H. For this simulation, we have carried out a parallel calculation with a population of 600 individuals and the result has been that the norm is not affected in this second optimisation.

Result 5: The performance of the SWAP cannot be improved for 2 qubits

The generalisation of the SWAP does not allow to reduce $\forall\rho,\sigma$ the second order error in Δt for the operation presented in the LMR protocol.

4.3.4 Pauli basis representation

Despite we have seen that we cannot find a generic matrix that cancels first and second order for any ρ and σ, we will see in the following section if an enhanced protocol depending on ρ and σ can be designed. But firstly, it is interesting to see how the solution H is expressed in the Pauli basis, since many physics problems are expressed in this basis. This is a complete basis made up of four 2×2 matrices

$\{\sigma^{(0)}, \sigma^{(1)}, \sigma^{(2)}, \sigma^{(3)}\} = \{\mathbb{1}, \sigma_x, \sigma_y, \sigma_z\}$, where σ_i are the Pauli matrices. Any $2^n \times 2^n$ matrix can be represented as the linear combination of at most 4^n Pauli strings.

We have found the general expression for any dimension $d = 2^n$

$$H = \sum_{i_1 i_2 j_1 j_2 = 0}^{2^n - 1} H_{i_2 j_2 i_1 j_1} |i_2 j_1\rangle\langle i_1 j_1| \tag{4.67}$$

where $H_{i_2 j_2 i_1 j_1} = x_{i_1 i_2} \delta^{j_1}_{j_2} + \delta^{i_1}_{j_2} \delta^{i_2}_{j_1}$. These indexes, which take values from 0 to 2^{n-1}, can be expressed in binary representation using n qubits.

For a general case, if we want to express the matrix element of this computational basis in terms of Pauli strings what we have is the following

$$|i_1 ... i_n\rangle\langle j_1 ... j_n| = \sum_{k_1 ... k_n = 0}^{3} g_{k_1 ... k_n}\, \sigma^{(k_1)} \otimes ... \otimes \sigma^{(k_N)}, \tag{4.68}$$

where $i_i,\ j_i = 0, 1$ and the goal would be to obtain the different $g_{\vec{k}}$ coefficients. In order to find the coefficient $g_{l_1 ... l_n}$, we project using the Hilbert-Schmidt inner product $\langle A, B\rangle = \mathrm{Tr}(B^\dagger A)$. To do so, we multiply on both sides of the above expression by $\sigma^{(l_1)} \otimes ... \otimes \sigma^{(l_N)}$ and take the trace. Notice that as $\mathrm{Tr}(A \otimes B) = \mathrm{Tr}A \cdot \mathrm{Tr}B$ and $\mathrm{Tr}\, \sigma^{(i)} = 0$ for $i = 1, 2, 3$ the only surviving term is that of $k_i = l_i$ as $\left(\sigma^{(i)}\right)^2 = \mathbb{1}\ \forall i = 0, 1, 2, 3$. Hence, taking into account that $\mathrm{Tr}\sigma^{(0)} = 2$ we will have a 2^n factor obtaining

$$\begin{aligned} g_{l_1 ... l_n} &= \frac{\mathrm{Tr}\left(\sigma^{(l_1)} \otimes ... \otimes \sigma^{(l_n)} \cdot |i_1 ... i_n\rangle\langle j_1 ... j_n|\right)}{2^n} = \frac{\mathrm{Tr}\left(\sigma^{(l_1)}|i_1\rangle j_1| \otimes ... \otimes \sigma^{(l_n)}|i_n\rangle\langle j_n|\right)}{2^n} \\ &= \frac{\mathrm{Tr}\left(\sigma^{(l_1)}|i_1\rangle\langle j_1|\right) \cdot ... \cdot \mathrm{Tr}\left(\sigma^{(l_n)}|i_n\rangle\langle j_n|\right)}{2^n} = \frac{\langle j_1|\sigma^{(l_1)}|i_1\rangle \cdot ... \cdot \langle j_n|\sigma^{(l_n)}|i_n\rangle}{2^n} \\ &\qquad \uparrow \\ &\qquad \mathrm{Tr}\left(A|\psi\rangle\langle\phi|\right) = \langle\phi|A|\psi\rangle \\ &= \frac{\sigma^{(l_1)}_{j_1 i_i} \cdot ... \cdot \sigma^{(l_n)}_{j_n i_n}}{2^n} = \frac{1}{2^n} \prod_{k=1}^{n} \sigma^{(l_k)}_{j_k i_k}, \end{aligned} \tag{4.69}$$

where the sub-indexes refer to matrix elements $\sigma_{ab} = \langle a|\sigma|b\rangle$.

We know that H connects two subspaces with n qubits each, so it can be represented in terms of Pauli strings as

$$H = \sum_{k_i, l_i = 0}^{3} g_{\vec{k}\,\vec{l}}\ \underbrace{\sigma^{(k_1)} \otimes \ ... \ \otimes \sigma^{(k_n)}}_{\vec{\sigma}^{(k)}} \otimes \underbrace{\sigma^{(l_1)} \otimes \ ... \ \otimes \sigma^{(l_n)}}_{\vec{\sigma}^{(l)}}, \tag{4.70}$$

where now the vector notation refers to n-elements strings so the sum is over k_i, l_i with i going from 1 to n. When we impose $H = H^\dagger$ we obtain that the coefficients $g_{\vec{k}\vec{l}}$ must be real. Expressing H in the computational basis

$$H = \sum_{i_i,j_i,h_i,g_i=0}^{1} H_{\vec{i}\,\vec{j}\,\vec{h}\,\vec{g}} \big(|\vec{i}\rangle \otimes |\vec{j}\rangle\big)\big(\langle\vec{h}| \otimes \langle\vec{g}|\big), \tag{4.71}$$

where now

$$H_{\vec{i}\,\vec{j}\,\vec{h}\,\vec{g}} = x_{\vec{h}\,\vec{i}}\,\delta^{\vec{j}}_{\vec{g}} + \delta^{\vec{j}}_{\vec{h}}\delta^{\vec{i}}_{\vec{g}}. \tag{4.72}$$

Following the same procedure as in Eq. 4.69, we can calculate the coefficients that will be used to express H in the Pauli basis

$$\begin{aligned}
g_{\vec{k}\,\vec{l}} &= \frac{\mathrm{Tr}\left(\left[\sum_{i_i,j_i,h_i,g_i=0}^{1}(x_{\vec{h}\,\vec{i}}\,\delta^{\vec{j}}_{\vec{g}} + \delta^{\vec{j}}_{\vec{h}}\delta^{\vec{i}}_{\vec{g}})|\vec{i}\,\vec{j}\rangle\langle\vec{h}\,\vec{g}|\right]\vec{\sigma}^{(k)} \otimes \vec{\sigma}^{(l)}\right)}{2^{2n}} \\
&= \sum_{s_i\,t_i=0}^{1} \frac{\langle\vec{s}\,\vec{t}|\sum_{i_i,j_i,h_i,g_i=0}^{1}(x_{\vec{h}\,\vec{i}}\,\delta^{\vec{j}}_{\vec{g}} + \delta^{\vec{j}}_{\vec{h}}\delta^{\vec{i}}_{\vec{g}})|\vec{i}\,\vec{j}\rangle\langle\vec{h}\,\vec{g}|\,\vec{\sigma}^{(k)} \otimes \vec{\sigma}^{(l)}|\vec{s}\,\vec{t}\rangle}{2^{2n}} \\
&= \frac{\sum_{s_i\,t_i}\ \sum_{i_i\,j_i\,h_i\,g_i}(x_{\vec{h}\,\vec{i}}\,\delta^{\vec{j}}_{\vec{g}} + \delta^{\vec{j}}_{\vec{h}}\delta^{\vec{i}}_{\vec{g}})\,\delta^{\vec{s}}_{\vec{i}}\,\delta^{\vec{t}}_{\vec{j}}\,\langle\vec{h}\,\vec{g}|\,\vec{\sigma}^{(k)} \otimes \vec{\sigma}^{(l)}|\vec{s}\,\vec{t}\rangle}{2^{2n}} \\
&= \frac{\sum_{i_i\,j_i\,h_i\,g_i}(x_{\vec{h}\,\vec{i}}\,\delta^{\vec{j}}_{\vec{g}} + \delta^{\vec{j}}_{\vec{h}}\delta^{\vec{i}}_{\vec{g}})\,\langle\vec{h}\,\vec{g}|\,\vec{\sigma}^{(k)} \otimes \vec{\sigma}^{(l)}|\vec{i}\,\vec{j}\rangle}{2^{2n}} \\
&= \frac{\sum_{i_i\,j_i\,h_i} x_{\vec{h}\,\vec{i}}\langle\vec{h}|\vec{\sigma}^{(k)}|\vec{i}\rangle\langle j|\vec{\sigma}^{(l)}|\vec{j}\rangle + \sum_{j_i\,i_i}\langle\vec{j}|\vec{\sigma}^{(k)}\overbrace{|\vec{i}\rangle\langle\vec{i}|}^{=\mathbb{1}}\vec{\sigma}^{(l)}|\vec{j}\rangle}{2^{2n}} \\
&= \frac{\sum_{i_i\,j_i\,h_i} x_{\vec{h}\,\vec{i}}\,\prod_{\lambda=1}^{n}\,\sigma^{(k_\lambda)}_{h_\lambda\,i_\lambda}\,\prod_{\beta=1}^{n}\,\sigma^{(l_\beta)}_{j_\beta\,j_\beta} + \sum_{j_i}\,\prod_{\lambda=1}^{n}\,\langle j_\lambda|\sigma^{k_\lambda}\,\sigma^{l_\lambda}|j_\lambda\rangle}{2^{2n}}.
\end{aligned} \tag{4.73}$$

Let us make use of this last formula to study the two-qubit case, so $n = 1$. The representation of H will be

$$H = \sum_{k_1,l_1=0}^{3} g_{k_1 l_1}\sigma^{(k_1)} \otimes \sigma^{(l_1)}. \tag{4.74}$$

For the calculation of the coefficients, looking at the Eq. 4.73, as $n = 1$, the products disappear, substantially simplifying the expression

$$g_{k_1 l_1} = \frac{\sum_{i_1,j_1,h_1=0}^{1} x_{h_1 i_1}\sigma^{(k_1)}_{h_1 i_1}\sigma^{(l_1)}_{j_1 j_1} + \mathrm{Tr}(\sigma^{(k_1)}\sigma^{(l_1)})}{4}. \tag{4.75}$$

Hence, we specify the calculation of the coefficient g_{00} as an example

$$\begin{aligned} g_{00} &= \frac{x_{00}\,\sigma^{(0)}_{00}\sigma^{(0)}_{00} + x_{00}\,\sigma^{(0)}_{00}\sigma^{(0)}_{11} + x_{11}\,\sigma^{(0)}_{11}\sigma^{(0)}_{00} + x_{11}\,\sigma^{(0)}_{11}\sigma^{(0)}_{11} + \mathrm{Tr}(\sigma^{(0)}\sigma^{(0)})}{4} \\ &= \frac{1}{2}(x_{00} + x_{11} + 1). \end{aligned} \tag{4.76}$$

The rest of the coefficients are summarized in Table 4.2.

k_1	l_1	$g_{k_1 l_1}$
0	0	$(x_{00} + x_{11} + 1)/2$
0	1	0
0	2	0
0	3	0
1	0	$(x_{01} + x_{10})/2$
1	1	$1/2$
1	2	0
1	3	0
2	0	$-i(x_{01} - x_{10})/2$
2	1	0
2	2	$1/2$
2	3	0
3	0	$(x_{00} - x_{11})/2$
3	1	0
3	2	0
3	3	$1/2$

Table 4.2: Coefficients of the representation of H in the Pauli basis for 2 qubits

We can now rewrite H in the Pauli basis

$$\begin{aligned} H =& \frac{1}{2}(x_{00} + x_{11} + 1)\big(\mathbb{1}\otimes\mathbb{1}\big) + \frac{1}{2}(x_{01} + x_{10})\big(\sigma_x\otimes\mathbb{1}\big) - \frac{i}{2}(x_{01} - x_{10})\big(\sigma_y\otimes\mathbb{1}\big) \\ &+ \frac{1}{2}(x_{00} - x_{11})\big(\sigma_z\otimes\mathbb{1}\big) + \frac{1}{2}\big(\sigma_x\otimes\sigma_x + \sigma_y\otimes\sigma_y + \sigma_z\otimes\sigma_z\big) \end{aligned} \tag{4.77}$$

where we can straightforwardly check that, by expanding the matrices, we get the same result as in Eq. 4.41. Remember that the Hermiticity condition implies $x_{10} = x^*_{01}$. Notice that the SWAP matrix for two qubits is expressed in the Pauli basis as

$$\text{SWAP} = \frac{1}{2}\big(\mathbb{1}\otimes\mathbb{1} + \sigma_x\otimes\sigma_x + \sigma_y\otimes\sigma_y + \sigma_z\otimes\sigma_z\big). \tag{4.78}$$

Hence, by redefining the parameters

$$\lambda_0 \equiv \frac{1}{2}(x_{00} + x_{11}), \tag{4.79}$$

$$\lambda_1 \equiv \frac{1}{2}(x_{01} + x_{01}^*) = \mathrm{Re}\,(x_{01}), \tag{4.80}$$

$$\lambda_2 \equiv \frac{-i}{2}(x_{01} - x_{01}^*) = \mathrm{Im}\,(x_{01}), \tag{4.81}$$

$$\lambda_3 \equiv \frac{1}{2}(x_{00} - x_{11}), \tag{4.82}$$

where notice that $\lambda_i \in \mathbb{R}\ \forall i = 0, 1, 2, 3$, we can rewrite Eq. 4.77 as

$$H = \mathrm{SWAP} + \underbrace{\left(\lambda_0\, \mathbb{1} + \lambda_1\, \sigma_x + \lambda_2\, \sigma_y + \lambda_3\, \sigma_z\right)}_{\equiv M} \otimes \mathbb{1}, \tag{4.83}$$

where , due to the freedom choosing the parameters λ_i, the matrix M is an arbitrary 2×2 Hermitian matrix. This decomposition of the H gate is very useful because for $M = 0$, we recover the case introduced in Ref. [15], which will be useful in Section. 4.4.

4.4 Optimisation of H for a fixed ρ and σ

Once we have seen that the generalisation of the protocol does not improve the result obtained by using the for an arbitrary ρ and σ, our aim is to find, given an initial ρ and σ, which is the M (defined in Eq. 4.83) which reduces the number of copies of ρ necessaries to approach $e^{-i\rho t}\sigma e^{i\rho t}$ with a given error ϵ. For the sake of simplicity, in the work we will study the case of two qubits, for which M depends on four real parameters

$$M = \begin{pmatrix} a & b + ic \\ b - ic & d \end{pmatrix}. \tag{4.84}$$

For convenience, we will work by representing M in the Pauli basis. Hence, the notation employed will be $\{\mathbb{1}, X, Y, Z\}$, so

$$M = m_0\, \mathbb{1} + m_x\, X + m_y\, Y + m_z\, Z \tag{4.85}$$

where now the real parameters are m_0, m_x, m_y, m_z.

As we did in Eqs. 4.30 and 4.31, let us start from the operation of the LMR

protocol and the desired operation for the case where we have a single copy of ρ

$$\text{Protocol: } \mathrm{Tr}_1\big(e^{-iH\Delta t}\,\rho\otimes\sigma\,e^{iH\Delta t}\big)=\mathrm{Tr}_1\left(\sum_{k=0}^{\infty}\frac{(-i\Delta t)^k}{k!}\,[H,\rho\otimes\sigma]_k\right), \tag{4.86}$$

$$\text{Desired operation: } e^{-i\rho\Delta t}\sigma e^{i\rho\Delta t}=\sum_{k=0}^{\infty}\frac{(-i\Delta t)^k}{k!}[\rho,\sigma]_k. \tag{4.87}$$

Again, the zero and first order of Δt coincide in both equations, so we will try to find a matrix H such that, for given ρ and σ components, cancels the term in $\mathcal{O}(\Delta t)^2$ from the difference of both operations. Equation 4.83 shows that looking for the optimal H will translate into looking for the optimal M. The condition to be fulfilled is the same as in Eq. 4.33 and, substituting our H in terms of M, it transforms into

$$\mathrm{Tr}_1\Big(\big[S+M\otimes\mathbb{1},[S+M\otimes\mathbb{1},\rho\otimes\sigma]\big]\Big)=\big[\rho,[\rho,\sigma]\big], \tag{4.88}$$

where S is the SWAP operator. Next, we will work out the LHS of the matrix equation. Expanding the commutators four terms are obtained

$$\mathrm{Tr}_1\Big(\underbrace{\big[S,[S,\rho\otimes\sigma]\big]}_{1^{\text{st}}\text{ term}}+\underbrace{\big[S,[M,\rho\,]\otimes\sigma\big]}_{2^{\text{nd}}\text{ term}}+\underbrace{\big[M\otimes\mathbb{1},[S,\rho\otimes\sigma]\big]}_{3^{\text{rd}}\text{ term}}+\underbrace{\big[M\otimes\mathbb{1},[M,\rho]\otimes\sigma\big]}_{4^{\text{th}}\text{ term}}\Big) \tag{4.89}$$

where we have used that $[M\otimes\mathbb{1},\rho\otimes\sigma]=[M,\rho\,]\otimes\sigma$. By manipulating the commutators and using the properties $\mathrm{Tr}_1(A\otimes B\;S)=BA$, $\mathrm{Tr}_1(S\;A\otimes B)=AB$ and $\mathrm{Tr}_1(A\otimes B)=B\;\mathrm{Tr}(A)$ the four terms transforms into

$$1^{\text{st}}\text{ term}=2\,(\sigma-\rho), \tag{4.90}$$

$$2^{\text{nd}}\text{ term}=\big[[M,\rho],\sigma\big], \tag{4.91}$$

$$3^{\text{rd}}\text{ term}=0, \tag{4.92}$$

$$4^{\text{th}}\text{ term}=0. \tag{4.93}$$

Therefore, isolating the terms containing M, the equation that we must solve is

$$\big[\rho,[\rho,\sigma]\big]-2\,(\sigma-\rho)=\big[[M,\rho],\sigma\big]. \tag{4.94}$$

Given the impossibility of isolating M directly, we will try to solve the equation by vectorising it. To do this, we will begin by adding an identity to the LHS of the equation and writing the matrix M in terms of its components

$$\Big(\big[\rho,[\rho,\sigma]\big]-2\,(\sigma-\rho)\Big)\sum_i|i\rangle\langle i|=\sum_{j,k}m_{j,k}\big(|j\rangle\langle k|\,\rho\,\sigma-\rho\,|j\rangle\langle k|\,\sigma-\sigma\,|j\rangle\langle k|\,\rho+\sigma\rho\,|j\rangle\langle k|\big). \tag{4.95}$$

From here, we vectorise the equation using the properties described in Ref. [38]

$$\text{vec}(AB) = (A \otimes \mathbb{1})\ \text{vec}(B), \tag{4.96}$$
$$\text{vec}(AB) = (\mathbb{1} \otimes B^T)\ \text{vec}(A), \tag{4.97}$$
$$\text{vec}(ACB) = (A \otimes \mathbb{1})\ (\mathbb{1} \otimes B^T)\ \text{vec}(C), \tag{4.98}$$

where B^T is the transpose matrix of B, obtaining

$$\begin{aligned}&\Big(\big([\rho,[\rho,\sigma]] - 2\,(\sigma-\rho)\big) \otimes \mathbb{1}\Big) \sum_i |ii\rangle \\ &= \sum_{j,k} m_{jk} \underbrace{\left(\mathbb{1} \otimes \sigma^T\rho^T - \rho \otimes \sigma^T - \sigma \otimes \rho^T + \sigma\rho \otimes \mathbb{1}\right)}_{\equiv\chi} |jk\rangle\,.\end{aligned} \tag{4.99}$$

Notice that $|M\rangle \equiv \sum_{j,k} m_{jk}\,|jk\rangle$ is now a column vector containing the matrix elements of M which are the unknowns of our system of equations. This system can be expressed in matrix form as

$$\chi\,|M\rangle = |v\rangle\,, \tag{4.100}$$

where the vector $|v\rangle$ is the LHS of Eq. 4.99. If the matrix χ were full rank, we would have a determinate compatible system and all we would have had to do is clear $|M\rangle = \chi^{-1}\,|v\rangle$, however, χ is not full rank, thus is not invertible. We have also verified that for the qubit case the rank of the matrix resulting from adding the column vector $|v\rangle$ to the matrix χ is different from that of the matrix χ separately and, therefore, following the systems of linear equations theory [39], we can state that there is no solution for this system of equations.

Result 6: There is no $\rho-$ and $\sigma-$dependent M that eliminates the term in Δt^2 from the LMR protocol for the two-qubit case

Given two density matrices ρ and σ, there is not a matrix M depending on ρ and σ such that the application of the LMR protocol with $H = \text{SWAP} + M \otimes \mathbb{1}$ allows for cancelling the second order in Δt.

As before, even if we cannot eliminate the contribution, we will try to minimise it. We will work in the Pauli basis, so we take the expression for M from Eq. 4.85, and we express the density matrices as

$$\rho = 1/2\ \mathbb{1} + \rho_x\ X + \rho_y\ Y + \rho_z\ Z, \tag{4.101}$$
$$\sigma = 1/2\ \mathbb{1} + \sigma_x\ X + \sigma_y\ Y + \sigma_z\ Z, \tag{4.102}$$

where notice that the coefficients associated with the identity have been taken as $1/2$ to ensure that both matrices have trace one. For the present two-qubit case, given the ρ and σ matrices, the calculation of the Pauli coefficients is straighforward

$$\rho_i = \frac{1}{2}\mathrm{Tr}\big(\rho\, \sigma^{(i)}\big), \tag{4.103}$$

where $i = 1, 2, 3$ labels the Pauli matrices $\{\sigma^{(2)}, \sigma^{(2)}, \sigma^{(3)}\} = \{X, Y, Z\}$. We want to minimise the norm of the function

$$T(M) = \big[[M, \rho\,], \sigma\big] + 2\,(\sigma - \rho) - \big[\rho, [\rho, \sigma]\big]. \tag{4.104}$$

There are several matrix norms that can quantify the contribution of Δt^2 terms, in general the p-norm is expressed as

$$\|A\|_p = \Big(\sum_i |\lambda_i|^p\Big)^{1/p}, \tag{4.105}$$

where λ_i are the eigenvalues of the matrix A, in our case the eigenvalues of $T(M)$. For convenience, we choose $p = 2$, and as this is a minimisation, we can take the square of this norm. Hence, the cost function we want to minimise is

$$f_{\mathrm{cost}} = \|T(M)\|_2^2. \tag{4.106}$$

This function does not depend on m_0 so there are three remaining parameters $\vec{m}$ =(m_x, m_y,m_z). The the minimisation condition then reads

$$\nabla f_{\mathrm{cost}} = \Big(\frac{\partial}{\partial m_x}, \frac{\partial}{\partial m_y}, \frac{\partial}{\partial m_z}\Big) f_{\mathrm{cost}}(\vec{m}) = 0. \tag{4.107}$$

We obtain that we are dealing with an indeterminate compatible system whose solution is given by the conditions

$$m_y = \frac{\rho_y}{\rho_x} m_x + C_1, \tag{4.108}$$

$$m_z = \frac{\rho_z}{\rho_x} m_x + C_2, \tag{4.109}$$

where C_1 and C_2 are constants depending on the Pauli coefficients of ρ and σ. To check that it is indeed a minimum, we have calculated the Hessian matrix and applied the Sylvester criterion [40]. As we can see, the minimum is found in two straight lines for the parameters m_y and m_z, so we still have one free parameter.

We can additionally take advantage of this remaining degree of freedom to minimise the Δt^3 contributions from the difference between the protocol and the desired operation. The condition in this case is

$$\mathrm{Tr}_1\left(\Big[S + M\otimes\mathbb{1}, \big[S + M\otimes\mathbb{1}, [S + M\otimes\mathbb{1}, \rho\otimes\sigma]\big]\Big]\right) = \Big[\rho, \big[\rho, [\rho,\sigma]\big]\Big], \quad (4.110)$$

where now we have eight terms in the LHS. By calculating each term as in the case of Δt^2, the matrix whose norm we have to minimise is

$$T(M) = \Big[\rho, \big[\rho, [\rho,\sigma]\big]\Big] - 4\,[\rho,\sigma] + 3\,[M,\rho] - [M,\sigma] - \Big[\big[M, [M,\rho]\big],\sigma\Big]. \quad (4.111)$$

We calculate the cost function into which we substitute the paramerisation obtained from the second order minimisation as

$$f_{\text{cost}} = \big\|T\big(m_x, m_y(m_x), m_z(m_x)\big)\big\|_2^2. \quad (4.112)$$

We thus have an equation with only one parameter, so the condition for finding the minimum is simply the derivative

$$\frac{\partial f_{\text{cost}}}{\partial m_x} = 0, \quad (4.113)$$

where we have checked that the condition of the second order derivative to be a minimum $\frac{\partial^2 f_{\text{cost}}}{\partial m_x^2} > 0$ is indeed fulfilled. Substituting the expression for m_x into m_y and m_z, we get

$$m_x = \frac{E(x,y,z) + F(x,y,z) + \rho_x\, G(x,y,z) + \rho_x\sigma_x^2\, H(x,y,z)}{D}, \quad (4.114)$$

$$m_y = \frac{E(y,z,x) + F(y,z,x) + \rho_y\, G(y,z,x) + \rho_y\sigma_y^2\, H(y,z,x)}{D}, \quad (4.115)$$

$$m_z = \frac{E(z,x,y) + F(z,x,y) + \rho_y\, G(z,x,y) + \rho_z\sigma_z^2\, H(z,x,y)}{D}, \quad (4.116)$$

where

$$D = 2({\sigma_x}^2 + {\sigma_y}^2 + {\sigma_z}^2)\Big(4{\rho_x}^2{\sigma_x}^2 + 2\rho_x\sigma_x(4\rho_y\sigma_y + 4\rho_z\sigma_z - 1) + \left(4{\rho_z}^2 + 1\right){\sigma_z}^2 + \sigma_y\left(4{\rho_y}^2\sigma_y - 2\rho_y + \sigma_y\right) + 2\rho_z\sigma_z(4\rho_y\sigma_y - 1) + {\sigma_x}^2\Big),$$

$$E(a,b,c) = \sigma_a(-1 + 2\rho_b\sigma_b + 2\rho_c\sigma_c)\Big(\sigma_a^2 + (1 + 4\rho_c^2)\sigma_c^2 + 2\rho_c\sigma_c(-1 + 4\rho_b\sigma_b) \sigma_b(-2\rho_b + \sigma_b + 4\rho_b^2\sigma_b)\Big),$$

$$F(a,b,c) = 2\rho_a^2\sigma_a\Big(\sigma_a^2(-3 + 4\rho_b\sigma_b + 4\rho_c\sigma_c) - (-1 + 8\rho_b\sigma_b + 8\rho_c\sigma_c)(\sigma_b^2 + \sigma_c^2)\Big) + 8\rho_a^3\Big(\sigma_a^4 + \sigma_a^2(\sigma_b^2 + \sigma_c^2) + (\sigma_b^2 + \sigma_c^2)^2\Big),$$

$$G(a,b,c) = (-6 + 8\rho_b^2 + 8\rho_c^2)\sigma_a^4 + (\sigma_b^2 + \sigma_c^2)\Big(8\sigma_b^2(-1 + \rho_c^2) + 8\sigma_c^2(-1 + \rho_b^2) + 1 + 2\sigma_b\rho_b + 2\rho_c\sigma_c(1 - 8\rho_b\sigma_b)\Big),$$

$$H(a,b,c) = 3 + 2\Big(-7\rho_b\sigma_b + \sigma_b(-7 + 16\rho_b^2 + 8\rho_c^2) + \rho_c\sigma_c(-7 + 16\rho_b\sigma_b) + (-7 + 8\rho_b^2 + 16\rho_c^2)\sigma_c^2\Big).$$

Result 7: $\rho-$ and $\sigma-$dependent protocol enhancement for the two-qubit case

The parameters m_x, m_y and m_z, which generate the matrix $M = m_x\ X + m_y\ Y + m_z\ Z$ and depend on the components of a given ρ and σ, allow for constructing the Hamiltonian $H = S + \mathbb{1} \otimes M$ for the two-qubit case, which optimises the second and third order in Δt from the LMR protocol, resulting in an enhanced protocol.

Figure 4.3 shows 50 curves with different randomised ρ and σ density matrices for each case. There are cases where the improvement exceeds 70%. Moreover, despite the fact that the optimisation has been done for the first step of the protocol, the improvement stabilises independently of the number of copies. In some cases, this stabilisation is reached for lower values of n than in others. This behaviour should be studied in further research.

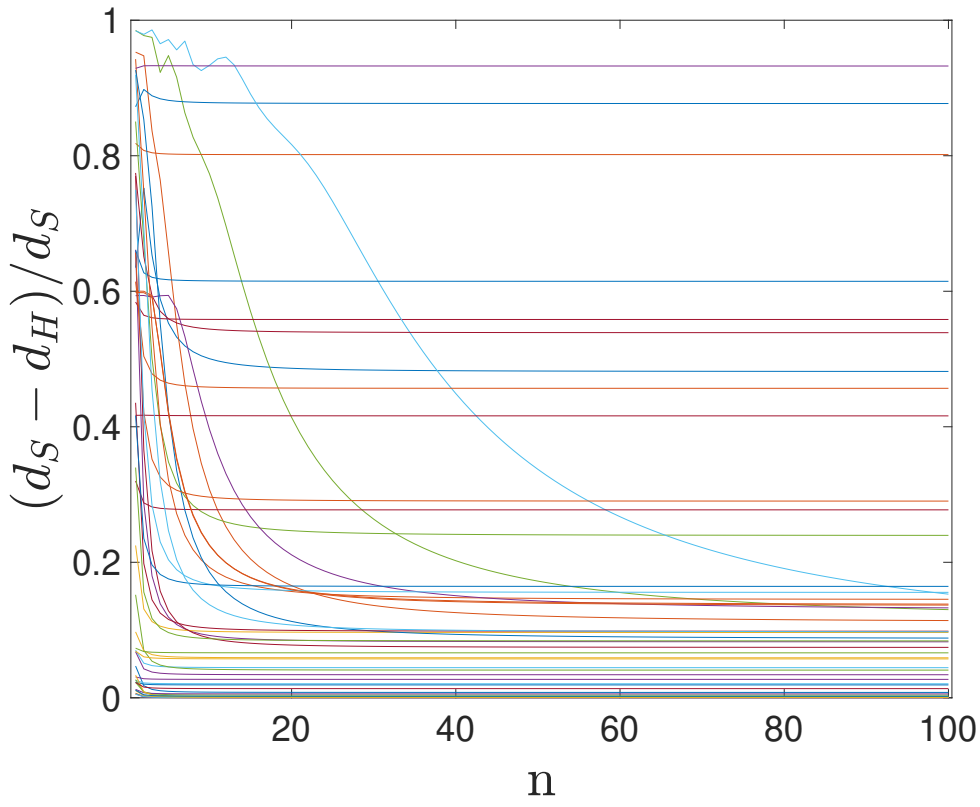

Figure 4.3: Relative difference between the application of the original LMR protocol with the SWAP and our protocol optimised using the Hamiltonian H as a function of the number of copies of ρ for the two-qubit case. The distances of the performance of the protocols with respect to the target operation are defined as $d_S = \left\| e^{-i\rho t}\sigma e^{i\rho t} - LMR \right\|_2$ and $d_H = \left\| e^{-i\rho t}\sigma e^{i\rho t} - LMR(S \to H) \right\|_2$. Here, 50 cases are shown for random ρ and σ density matrices.

CHAPTER 5

Beyond density matrices

So far we have made a detailed study of how to load density matrices into a quantum computer using the LMR protocol. However, as we already know, density matrices must have properties (see Section 2.2) that greatly restrict the set of square matrices that we are able to load using this protocol. Therefore, in this chapter we will extend the limits of the LMR protocol. In order to do that generalisation, we will consider a matrix A, and we will separate it into its Hermitian and skew-Hermitian parts. Then, we will see how to load each of these two parts separately. The resources needed and the error made in the whole process are also detailed in this chapter.

5.1 Normal matrices

Let us consider a normal square matrix A of dimension 2^n, where n is the number of qubits, which, in general, will not be Hermitian. We know that any matrix can be expressed as $A = H(A) + S(A)$ where $H(A) = \frac{1}{2}(A + A^\dagger)$ and $S(A) = \frac{1}{2}(A - A^\dagger)$ are the Hermitian and skew-Hermitian parts respectively. Since A is a normal matrix, $[A\ ,\ A^\dagger] = 0$ and, consequently, $[H(A)\ ,\ S(A)] = 0$. The propagator to simulate A up to a time t is expressed as

$$\hat{U}(t) = e^{-i\,A\,t} = e^{-i\,(H(A)+S(A))\,t} \underset{\substack{\uparrow \\ \text{Baker-Campbell-Hausdorff formula}}}{=} e^{-i\,S(A)\,t} e^{-i\,H(A)\,t}. \tag{5.1}$$

The operator $e^{-i\,H(A)\,t}$ is unitary so it can be simulated in a quantum computer. However, we now define $\hat{O}(t) = e^{-i\,S(A)\,t}$ which is an Hermitian operator. In the following subsections, will show how to use the LMR protocol to simulate each of these propagators.

5.1.1 Hermitian part H(A)

Among the three properties of density matrices, one of them is Hermiticity. In this first subsection, we will generalise the protocol for square matrices that only satisfy this property. We know that any square matrix can be written as the subtraction of two positive semi-definite matrices, however, there is no unique choice for these. Therefore, we will fix the convention defined by the sign of the eigenvalues. We consider our matrix to have a diagonal form $D = P^\dagger H(A) P$. As A is Hermitian, all these eigenvalues will be real numbers. D_+ will contain the positive eigenvalues and those that are 0 and D_- will contain the negative ones,thus

$$H(A) = P\,(D_+ - D_-)\,P^\dagger = H_+(A) - H_-(A), \tag{5.2}$$

where we define $H_+(A) = PD_+P^\dagger$ and $H_-(A) = PD_-P^\dagger$. Notice that these two matrices commute

$$\begin{aligned}[H_+(A), H_-(A)] =& H_+(A)H_-(A) - H_-(A)H_+(A) \\ =& PD_+ \underbrace{P^\dagger P}_{\mathbb{1}} D_-P^\dagger - PD_- \underbrace{P^\dagger P}_{\mathbb{1}} D_+P^\dagger \\ =& P\,(D_+D_- - D_-D_+)\,P^\dagger = P\,[D_+, D_-]\,P^\dagger = 0.\end{aligned} \tag{5.3}$$

Due to the fact that if A is a normal matrix, its Hermitian and skew-Hermitian parts are also normal and thus are diagonalized by unitary matrices P and $P^\dagger$. It is important to note that $H_+(A)$ and $H_-(A)$ both are Hermitian matrices.

So far we have two positive semi-definite Hermitian matrices. In order to fulfil all the density matrix conditions to apply the LMR protocol, we only need to normalise them by dividing by their trace. Hence, the desired operation turns out to be

$$e^{-iH(A)t}\,\sigma\,e^{iH(A)t} = e^{-it_+\rho_+}\,e^{it_-\rho_-}\,\sigma\,e^{-it_-\rho_-}\,e^{it_+\rho_-+}, \tag{5.4}$$

where we defined $t_\pm = t\,\mathrm{Tr}\,H_\pm(A)$ and $\rho_\pm = H_\pm(A)/\mathrm{Tr}\,H_\pm(A)$.

Taking into account this last expression, the loading of a Hermitian matrix, and in particular in this case, of the Hermitian part of a matrix A, is summarised by applying the LMR protocol on σ having n copies of ρ_- with $\Delta t_- = t_-/n$ and then reapplying it on the output having n copies of ρ_+ with $\Delta t_+ = t_+/n$. The analysis of the error made in each process is discussed in a later section.

5.1.2 Skew-Hermitian part S(A)

Let us now focus in the skew-Hermitian part $S(A)$. In order to deal with the Hermitian operator $\hat{O}(t) = e^{-i\,S(A)\,t}$ which is non-unitary we embed our system using an ancillary qubit q_E to define a unitary operator $\tilde{U}(t)$ using *unitary dilation* [41] as in Ref. [42]. The embedded propagator will be expressed as

$$\tilde{U}(t) = \begin{pmatrix} \hat{O} & \sqrt{1-\hat{O}^2} \\ \sqrt{1-\hat{O}^2} & -\hat{O} \end{pmatrix} = \big(\hat{\sigma}_E^z \otimes \mathbb{1}\big)\, \exp\big(i\,\hat{\sigma}_E^y \otimes \tilde{H}(t)\big), \tag{5.5}$$

where provided that $\|\hat{O}(t)\| \leq 1$ it is allowed to define $\tilde{H}(t) = \arccos \hat{O}(t)$. Additionally, notice that as $\hat{O}(t)$ is Hermitian, so is $\tilde{U}(t)$

Let us show this relation by developing the left-hand side of Eq. 5.5

$$\begin{aligned} \begin{pmatrix} \hat{O} & \sqrt{1-\hat{O}^2} \\ \sqrt{1-\hat{O}^2} & -\hat{O} \end{pmatrix} &= \big(\hat{\sigma}_E^z \otimes \hat{O}\big) + \big(\hat{\sigma}_E^x \otimes \sqrt{1-\hat{O}^2}\big) \\ &= \hat{\sigma}_E^z \otimes \cos\big(\arccos \hat{O}\big) + \hat{\sigma}_E^x \otimes \sin\big(\arccos \hat{O}\big) \\ &= \big(\hat{\sigma}_E^z \otimes \mathbb{1}\big)\Big[\mathbb{1} \otimes \cos \tilde{H}(t) + i\,\hat{\sigma}_E^y \otimes \sin \tilde{H}(t)\Big] \end{aligned} \tag{5.6}$$

where we have used $\hat{O} = \cos\Big(\arccos \hat{O}\Big)$, so $\sqrt{1-\hat{O}^2} = \sin\Big(\arccos \hat{O}\Big)$, and we have taken $\tilde{H}(t) = \arccos \hat{O}$. Now we Taylor expand the exponential of the right-hand side

$$\begin{aligned} \exp(i\,\hat{\sigma}_E^y \otimes \tilde{H}) &= \sum_{n=0}^{\infty} \frac{(i\,\hat{\sigma}_E^y \otimes \tilde{H})^n}{n!} \underset{\uparrow}{=} \sum_{n=0}^{\infty} \frac{i^{2n}(\hat{\sigma}_E^y \otimes \tilde{H})^{2n}}{(2n)!} + \sum_{n=0}^{\infty} \frac{i^{2n+1}(\hat{\sigma}_E^y \otimes \tilde{H})^{2n+1}}{(2n+1)!} \\ &\qquad \text{Separating into odd and even terms} \\ &= \mathbb{1} \otimes \sum_{n=0}^{\infty} \frac{(-1)^n \tilde{H}^{2n}}{(2n)!} + i\,\hat{\sigma}_E^y \otimes \sum_{n=0}^{\infty} \frac{(-1)^n \tilde{H}^{2n+1}}{(2n+1)!} \\ &\quad \underset{(\hat{\sigma}^y)^{2n}=\mathbb{1}\ and\ (\hat{\sigma}^y)^{2n+1}=\hat{\sigma}^y}{\uparrow} \\ &= \mathbb{1} \otimes \cos \tilde{H} + i\,\hat{\sigma}_E^y \otimes \sin \tilde{H}, \end{aligned} \tag{5.7}$$

recovering Eq. 5.5.

Using an ancilliary qubit we have embedded our Hermitian operator $\hat{O}(t)$ into an unitary one $\tilde{U}(t)$. Starting from the quantum state $\sigma_0 = |0\rangle\langle 0|_E \otimes \sigma$ the time

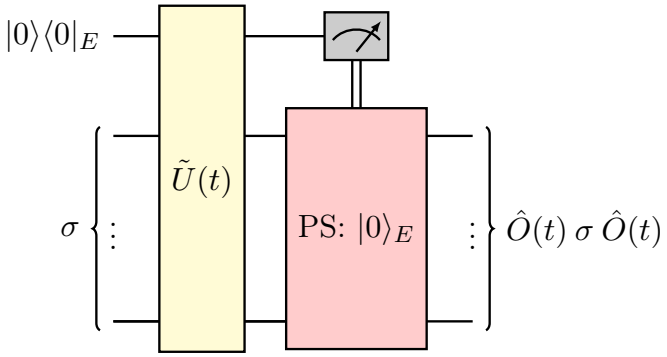

Figure 5.1: Circuit representation of the embedded system, This allows us to obtain the desired dynamics. Notice that after measuring the first qubit the remaining system state undergoes a post-selection process indicated by the acronym "PS".

evolution will be

$$\begin{aligned}\sigma_t =&\tilde{U}(t)\,\sigma_0\,\tilde{U}^\dagger(t) = \begin{pmatrix} \hat{O} & \sqrt{1-\hat{O}^2} \\ \sqrt{1-\hat{O}^2} & -\hat{O} \end{pmatrix} \begin{pmatrix} \sigma & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{O} & \sqrt{1-\hat{O}^2} \\ \sqrt{1-\hat{O}^2} & -\hat{O} \end{pmatrix} \\ =&|0\rangle\langle 0|_E \otimes \hat{O}\,\sigma\,\hat{O} \;+\; |0\rangle\langle 1|_E \otimes \hat{O}\,\sigma\,\sqrt{1-\hat{O}^2} \\ &+ |1\rangle\langle 0|_E \otimes \sqrt{1-\hat{O}^2}\,\sigma\,\hat{O} \;+\; |1\rangle\langle 1|_E \otimes \sqrt{1-\hat{O}^2}\,\sigma\,\sqrt{1-\hat{O}^2}.\end{aligned} \tag{5.8}$$

In order to achieve our aimed evolution $\hat{O}\,\sigma\,\hat{O}$ we use the post-selection technique for which we need to filter the outcomes with the ancillary qubit in the $|0\rangle\langle 0|_E$ state. The circuit corresponding to the embedding is depicted in Fig. 5.1.

Once we have shown that by embedding the system we can recover the desired evolution, let us see how we can apply the LMR protocol by studying the exponential $e^{i\hat{\sigma}_E^y \otimes \tilde{H}(t)}$. Notice that as mentioned above, $\tilde{U}$ is unitary and therefore $\sigma_E^y \otimes \tilde{H}(t)$ must be Hermitian, which means that $\tilde{H}(t)$ must be Hermitian too. We show this by Taylor expanding the arccos

$$\tilde{H}(t) = \arccos(\hat{\mathrm{O}}) = \frac{\pi}{2} - \sum_{n=0}^{\infty} \frac{(2n)!}{4^n (n!)^2 (2n+1)} \hat{O}^{2n+1}, \tag{5.9}$$

where if we take the conjugate transpose of last expression, as all coefficients are real numbers and $\hat{O}$ is Hermitian $\tilde{H}(t)$ is Hermitian too. In addition, as $0 \leq \arccos(x) \leq \pi$ in the domain $|x| \leq 1$, $\tilde{H}(t)$ is also positive semi-definite. However, this is not the

case for $\hat{\sigma}_E^y \otimes \tilde{H}(t)$, as

$$\hat{\sigma}_E^y \otimes \tilde{H} = V\hat{\sigma}_E^z V^\dagger \otimes UDU^\dagger = (V \otimes U)\,(\hat{\sigma}_E^z \otimes D)\,(V^\dagger \otimes U^\dagger), \tag{5.10}$$

so the eigenvalues of this matrix are the eigenvalues of $\tilde{H}$, which make up the diagonal matrix D, and these same eigenvalues multiplied by (-1). Hence, as we have negative eigenvalues $\hat{\sigma}_E^y \otimes \tilde{H}(t)$ is not a positive semi-definite matrix. However, as in the case of the Hermitian part, we can proceed by separating this matrix into the subtraction of two positive semi-definite matrices. Taking $\hat{\sigma}_z = |0\rangle\langle 0| - |1\rangle\langle 1|$ we obtain

$$\begin{aligned} \hat{\sigma}_E^y \otimes \tilde{H} &= (V \otimes U)\,(|0\rangle\langle 0| \otimes D - |1\rangle\langle 1| \otimes D)\,(V^\dagger \otimes U^\dagger) \\ &= V\,|0\rangle\langle 0|\,V^\dagger \otimes \tilde{H} - V\,|1\rangle\langle 1|\,V^\dagger \otimes \tilde{H}. \end{aligned} \tag{5.11}$$

Notice that the matrix V is the change matrix from basis $\hat{\sigma}_z$ to $\hat{\sigma}_y$. Therefore V acting on the $\hat{\sigma}_z$ eigenstates $|0\rangle$ and $|1\rangle$ will give us $|\circlearrowleft\rangle$ and $|\circlearrowright\rangle$ respectively, which are the $\hat{\sigma}_y$ eigenstates.

Finally, we only need to normalise by dividing by the trace of both matrices, which are already Hermitian and positive semi-definite. Using the fact that the trace of the outer product is the product of the traces and taking into account that the trace of the projectors is 1, the final expression we obtain is

$$\hat{\sigma}_E^y \otimes \tilde{H} = \mathrm{Tr}\,\tilde{H}\left(\frac{|\circlearrowleft\rangle\langle\circlearrowleft| \otimes \tilde{H}}{\mathrm{Tr}\,\tilde{H}} - \frac{|\circlearrowright\rangle\langle\circlearrowright| \otimes \tilde{H}}{\mathrm{Tr}\,\tilde{H}}\right) = t_{\tilde{H}}\,\left(\rho_{\tilde{H}+} - \rho_{\tilde{H}-}\right). \tag{5.12}$$

The exponential of this matrix will be separated into two exponentials since $[\rho_{\tilde{H}+}, \rho_{\tilde{H}-}] = 0$ due to the orthonormality of the $\hat{\sigma}_y$ eigenvectors. Hence, applying this to a generic density matrix σ_g

$$e^{i\hat{\sigma}_E^y \otimes \tilde{H}}\,\sigma_g\,e^{-i\hat{\sigma}_E^y \otimes \tilde{H}} = e^{it_{\tilde{H}}\rho_{\tilde{H}+}}\,e^{-it_{\tilde{H}}\rho_{\tilde{H}-}}\,\sigma_g\,e^{it_{\tilde{H}}\rho_{\tilde{H}-}}\,e^{-it_{\tilde{H}}\rho_{\tilde{H}+}}. \tag{5.13}$$

In our case, as we are working with the embedded system to which we have added an extra qubit, the density matrix to which we will apply this operation will be $|0\rangle\langle 0|_E \otimes \sigma_R$, where σ_R is the state resulting from having loaded the Hermitian part $H(A)$ and, as previously mentioned, the ancillary qubit initialised in the $|0\rangle$ state in order to recover the desired dynamics after the post-selection process.

Once this development is done, let us see how we could apply the LMR protocol to carry out the loading given by the Equation 5.13. For this we will need to have copies of $\rho_{\tilde{H}_+}$ and $\rho_{\tilde{H}_-}$. This can be simplified to having copies of $\tilde{H}/\mathrm{Tr}\,\tilde{H}$ as we know how to generate the eigenstates of $\hat{\sigma}_y$ (see Fig.5.2).

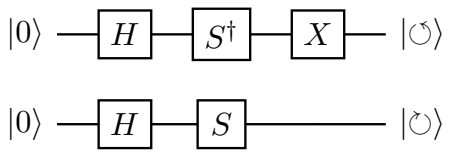

Figure 5.2: Schematic of the quantum gates needed to generate the states $|\circlearrowleft\rangle$ and $|\circlearrowright\rangle$.

The total input for the application of the generalised protocol to implement the Equation 5.13 is depicted in Figure 5.3 and a scheme of the complete protocol for the loading of a normal matrix A is shown in Figure 5.4.

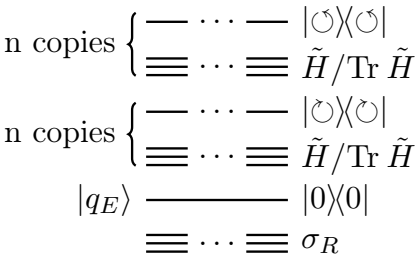

Figure 5.3: Input that we must use to use the LMR protocol and carry out the loading given by the Equation 5.13. Note that $|0\rangle\langle 0| \otimes \sigma_R = \sigma_g$.

5.1.3 Error analysis

We have seen the complete process for loading a normal matrix. To do this, we must recover the result of the error obtained after applying the LMR protocol operation n times. If we denote the output of this operation as $\tilde{\sigma}^{(n)}$,

$$\frac{1}{2}\left\| e^{-i\rho t}\,\sigma\, e^{i\rho t} - \tilde{\sigma}^{(n)} \right\|_1 \leq \mathcal{O}\big(t^2/n\big). \tag{5.14}$$

From this result, we can calculate the error obtained in each step.

1. We perform $e^{it_-\rho_-}\,\sigma\, e^{-it_-\rho_-}$ by means of the operation

$$\mathrm{Tr}_1\Big(e^{iSt_-/n}(\rho_- \otimes \sigma)e^{-iSt_-/n}\Big), \tag{5.15}$$

   using n copies of ρ_-. Let us denote the output of this operation as σ'_- so finally

$$\frac{1}{2}\left\| e^{it_-\rho_-}\,\sigma\, e^{-it_-\rho_-} - \sigma'_- \right\|_1 \leq \mathcal{O}(t_-^2/n). \tag{5.16}$$

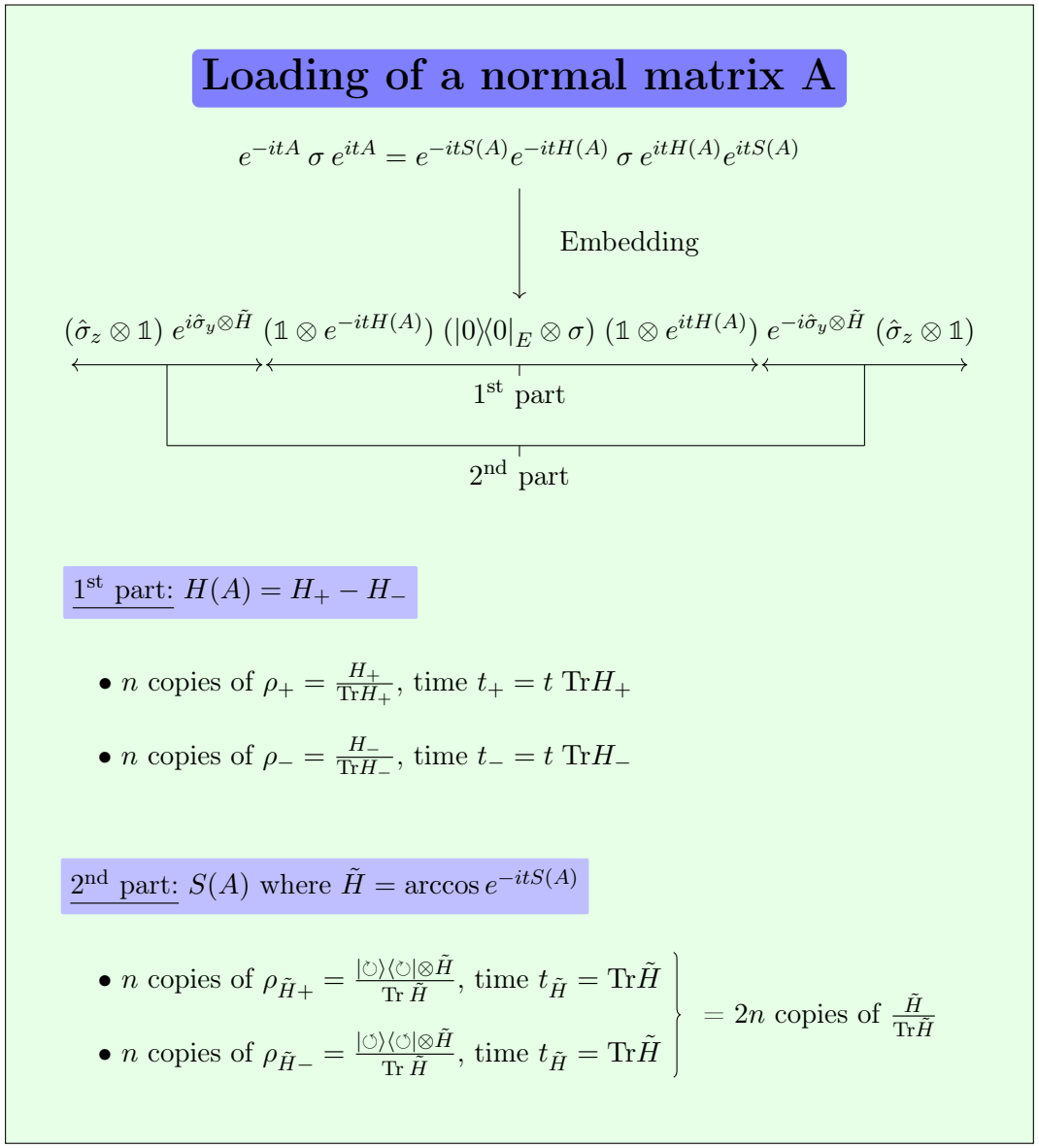

Figure 5.4: Scheme of the generalised protocol for loading a normal matrix A. The resources required are specified.

2. We perform $e^{-it_+\rho_+}\,\sigma'_-\,e^{it_+\rho_+}$ by means of the operation

$$\mathrm{Tr}_1\Big(e^{-iSt_+/n}\,(\rho_+ \otimes \sigma'_-)\,e^{-iSt_-/n}\Big), \tag{5.17}$$

using n copies of ρ_+. Let us now denote the output of this operation as σ_R

then

$$\frac{1}{2}\left\|e^{-it_+\rho_+}\, e^{it_-\rho_-}\, \sigma\, e^{-it_-\rho_-}\, e^{it_+\rho_-+} - \sigma_R\right\|_1 \leq \mathcal{O}(t_-^2/n) + \mathcal{O}(t_+^2/n). \quad (5.18)$$

As we can see, the error at every step is additive. With these two first steps we would have loaded the Hermitian part $H(A)$.

3. Now we load the skew-Hermitian part so we consider the embedded system. To perform the operation we proceed as indicated in Equation 5.13. First we perform the operation $e^{-it_{\tilde{H}}\rho_{\tilde{H}-}}\,(|0\rangle\langle 0|\otimes\sigma_R)\,e^{it_{\tilde{H}}\rho_{\tilde{H}-}}$ by means of the operation

$$\mathrm{Tr}_1\Big(e^{-iSt_{\tilde{H}}/n}\,\left(\rho_{\tilde{H}-}\otimes(|0\rangle\langle 0|\otimes\sigma_R)\right)\,e^{-iSt_{\tilde{H}}/n}\Big), \quad (5.19)$$

using n copies of $\rho_{\tilde{H}-}$. Denoting the output as σ''_- we have

$$\frac{1}{2}\left\|e^{-it_{\tilde{H}}\rho_{\tilde{H}-}}\,(|0\rangle\langle 0|\otimes\sigma_R)\,e^{it_{\tilde{H}}\rho_{\tilde{H}-}} - \sigma''_-\right\|_1 \leq \mathcal{O}(t_-^2/n) + \mathcal{O}(t_+^2/n) + \mathcal{O}(t_{\tilde{H}}^2/n). \quad (5.20)$$

4. The last step is to perform $e^{-it_{\tilde{H}}\rho_{\tilde{H}+}\sigma''_-}e^{it_{\tilde{H}}\rho_{\tilde{H}+}}$ by means of

$$\mathrm{Tr}_1\Big(e^{-iSt_{\tilde{H}}/n}\,\left(\rho_{\tilde{H}+}\otimes\sigma''_-\right)\,e^{-iSt_{\tilde{H}}/n}\Big), \quad (5.21)$$

with n copies of $\rho_{\tilde{H}+}$. Note that the error in this step is the same as in the previous one, so if we denote the final output of this operation as σ_f the error of the whole process is

$$\frac{1}{2}\left\|e^{-it_{\tilde{H}}\rho_{\tilde{H}+}\sigma''_-}e^{it_{\tilde{H}}\rho_{\tilde{H}+}} - \sigma_f\right\|_1 \leq \mathcal{O}(t_-^2/n) + \mathcal{O}(t_+^2/n) + 2\,\mathcal{O}(t_{\tilde{H}}^2/n). \quad (5.22)$$

An important note about the error in the last two steps comes from the realisation that we can bound $t_{\tilde{H}} = \mathrm{Tr}\,\tilde{H}$ where $\tilde{H} = \arccos\tilde{O}$. As the trace is invariant with respect to the base we can use the diagonal form, hence $\mathrm{Tr}\,\tilde{H} = \sum_i \arccos\lambda_i$, where λ_i are each one of the eigenvalues of $\hat{O}$. As already discussed the arccosine is a function bounded between 0 and π and therefore

$$\mathrm{Tr}\,\tilde{H} \leq d\cdot\pi, \quad (5.23)$$

where d is the dimension of the Hilbert space in which $\hat{O}$ lives and hence the dimension of the matrix A.

5.1.4 Success probability

The price to pay for including post-selection in our algorithm is that after having applied all the steps, we must measure the ancillary qubit, and in case we do not get $|0\rangle_E$ in the measurement, we must discard the whole process. That is why we define the probability of success P_s of the implementation of our algorithm as the probability of measuring $|q_E\rangle = |0\rangle_E$. The projector associated with this measurement is $|0\rangle\langle 0|_E \otimes \mathbb{1}$. Hence, the success probability is computed

$$\begin{aligned} P_s =& \mathrm{Tr}\left(\sigma_F \underbrace{(|0\rangle\langle 0|_E \otimes \mathbb{1})}_{(|0\rangle_E \otimes \mathbb{1})(\langle 0|_E \otimes \mathbb{1})}\right) = \sum_{k=0}^{1}\sum_{l=0}^{2^n-1} \langle k\, l|\sigma_F\, (|0\rangle_E \otimes \mathbb{1})(\langle 0|_E \otimes \mathbb{1})|k\, l\rangle \\ =& \sum_{l=0}^{2^n-1} (\langle 0| \otimes \langle l|)\, \sigma_F\, (|0\rangle \otimes |l\rangle). \end{aligned} \tag{5.24}$$

The final state of the system after the complete protocol is

$$\begin{aligned} \sigma_F =& \tilde{U}(t)\, \big(\, |0\rangle\langle 0|_E \otimes \underbrace{e^{-i\,t\,H(A)}\, \sigma\, e^{i\,t\,H(A)}}_{\sigma'}\big)\, \tilde{U}^\dagger(t) \\ =& |0\rangle\langle 0| \otimes \hat{O}\sigma'\hat{O} + |1\rangle\langle 0| \otimes \sqrt{1-\hat{O}^2}\, \sigma'\, \sqrt{1-\hat{O}^2}, \end{aligned} \tag{5.25}$$

where we took the form of the embedded propagator from Eq. 5.6

$$\tilde{U}(t) = \hat{\sigma}_z \otimes \hat{O} + \hat{\sigma}_x \otimes \sqrt{1-\hat{O}^2}. \tag{5.26}$$

Using this, Eq. 5.24 transforms into

$$\begin{aligned} P_s =& \sum_l (\langle 0| \otimes \langle l|)\, \Big(|0\rangle\langle 0| \otimes \hat{O}\sigma'\hat{O} + |1\rangle\langle 0| \otimes \sqrt{1-\hat{O}^2}\, \sigma'\, \sqrt{1-\hat{O}^2}\Big)\, (|0\rangle \otimes |l\rangle) \\ =& \sum_l (\langle 0| \otimes \langle l|\hat{O}\sigma'\hat{O})\, (|0\rangle \otimes |l\rangle) = \sum_l \langle l|\hat{O}\sigma'\hat{O}|l\rangle = \mathrm{Tr}\big(\hat{O}\sigma'\hat{O}\big) = \mathrm{Tr}\big(\hat{O}^2\sigma'\big) \\ =& \mathrm{Tr}\big(\, e^{-2it\,S(A)}\, e^{-it\,H(A)}\, \sigma\, e^{it\,H(A)} = \mathrm{Tr}\big(e^{-2it\,S(A)}\, \sigma\big). \\ & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \uparrow \\ & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad [H(A),S(A)]=0 \end{aligned} \tag{5.27}$$

We are interested in this probability being as close to 1 as possible. To get an idea of when this happens we can Taylor expand the exponential around $t \to 0$ and apply the linear property of the trace

$$P_s = 1 - 2\, t\, \mathrm{Tr}\big(\, iS(A)\, \sigma\,\big) + \mathcal{O}(t^3). \tag{5.28}$$

Notice that as $S(A)$ is skew-Hermitian, $iS(A)$ will be Hermitian and then its trace is real. All in all, the probability of success can be maximised in two ways:

- $t \ll \frac{1}{2\,\mathrm{Tr}\left(\, iS(A)\,\sigma\,\right)}$
- $\mathrm{Tr}\left(\, iS(A)\,\sigma\,\right) \to 0$, what means that $iS(A)$ and σ are orthogonal in the Hilbert-Schmidt inner product

5.2 Non-normal matrices

Previously, the normality condition for the matrix A had allowed us to separate the exponential of the Hermitian and skew-Hermitian part sum in the product of two exponentials. In the following, we will generalise about the accuracy of the protocol when the matrix is not normal by using Trotter's formalism. In our case, the total Hamiltonian H_t would be the sum of the hermitian and skew-Hermitian part of A

$$H_t = H(A) + S(A). \tag{5.29}$$

Using Trotter's formula introduced in Section 2.6

$$e^{it\,(H(A)+S(A)\,)} = \prod_{k=1}^{n} e^{i\Delta t H(A)} e^{i\Delta t S(A)} + \mathcal{O}(t^2/n). \tag{5.30}$$

The idea is that for each of the time intervals Δt we would use the generalised LMR protocol detailed above to load each part. The product indicates that we will have to implement this n times, so the only difference with respect to the loading of normal matrices is that now we have an additive error term that we have to take into account and add in the expression of Equation 5.22.

CHAPTER 6

Conclusions

Development of efficient information loading subroutines is crucial to keep the quantum speedup provided by information processing subroutine. Different quantum algorithms require specific codifications of information, consequently, we need to develop these techniques specifically for each codification. In this Thesis, we have made an exhaustive study of the *Lloyd-Mohseni-Rebentrost* (LMR) protocol Ref. [15], which allows for loading data encoded in the operation $e^{-i\rho t}$, acting on an initial state σ, by employing multiple copies of the state ρ, assuming the ability to implement efficiently the exponential of the SWAP operator.

We have rewritten the LMR protocol in the framework of quantum channels, explicitly calculating the Kraus operators. We have found a family of Hamiltonians H performing the same operation as the SWAP gate in the LMR protocol. Then, we have tried to use this degree of freedom in order to improve the relation between the accuracy and the number of copies. Remarkably, we have found that this is not possible for a $\rho-$ and $\sigma-$independent Hamiltonian H. However, if we allow that the Hamiltonian depends on the ρ and σ components, the protocol does allow for an improvement in the relation between the accuracy and the number of copies. For the sake of simplicity, we have focused on the two-qubit case. Working on the Pauli basis, we have managed to find the optimal parameters in terms of ρ and σ, obtaining promising results. Indeed, we have shown that our loading process achieves in some cases an enhancement of more than 70% over the original LMR. In this context, further research would be necessary to analytically extend these results for n qubits. Although it is not included in this Master Thesis, we have some preliminary results on an adaptive method, which recalculates the optimal parameters after each iteration.

Finally, as density matrices are a reduced set of the $2^n \times 2^n$ matrix group, we have proposed the generalisation of the protocol to load an arbitrary n-qubit square matrix. This extension is based on the separation of the matrix into its Hermitian and skew-Hermitian parts. While the Hermitian part can be loaded by simply separating

it into positive and negative parts, the skew-Hermitian part must be firstly embedded to apply the LMR protocol. With this study, we pave the way for optimising the required resources for this kind of information loading.

Bibliography

[1] R. P. Feynman, "Simulating physics with computers," *International journal of theoretical physics*, vol. 21, pp. 467–488, 1982.

[2] D. Deutsch, "Quantum theory, the church–turing principle and the universal quantum computer," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 400, no. 1818, pp. 97–117, 1985.

[3] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 439, no. 1907, pp. 553–558, 1992.

[4] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, p. 1484–1509, Oct 1997.

[5] S. Lloyd, "Universal quantum simulators," *Science*, vol. 273, no. 5278, pp. 1073–1078, 1996.

[6] I. L. Chuang, L. M. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, "Experimental realization of a quantum algorithm," *Nature*, vol. 393, no. 6681, pp. 143–146, 1998.

[7] C. Monroe, D. M. Meekhof, B. E. King, W. M. Itano, and D. J. Wineland, "Demonstration of a fundamental quantum logic gate," *Physical review letters*, vol. 75, no. 25, p. 4714, 1995.

[8] M. H. Devoret, A. Wallraff, and J. M. Martinis, "Superconducting qubits: A short review," *arXiv preprint cond-mat/0411174*, 2004.

[9] W. H. Zurek, "Decoherence and the transition from quantum to classical—revisited," in *Quantum Decoherence*. Springer, 2006, pp. 1–31.

[10] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, "Mixed-state entanglement and quantum error correction," *Physical Review A*, vol. 54, no. 5, p. 3824, 1996.

[11] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[12] A. S. Holevo, "Bounds for the quantity of information transmitted by a quantum communication channel," *Problemy Peredachi Informatsii*, vol. 9, no. 3, pp. 3–11, 1973.

[13] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Physical review letters*, vol. 103, no. 15, p. 150502, 2009.

[14] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, "Efficient quantum algorithms for simulating sparse hamiltonians," *Communications in Mathematical Physics*, vol. 270, no. 2, pp. 359–371, 2007.

[15] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, no. 9, pp. 631–633, 2014.

[16] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2000.

[17] J. Preskill, "Lecture notes for physics 229: Quantum information and computation," *California Institute of Technology*, vol. 16, no. 1, pp. 1–8, 1998.

[18] M. M. Wolf, "Quantum channels & operations: Guided tour," *Lecture notes available at http://www-m5. ma. tum. de/foswiki/pub M*, vol. 5, 2012.

[19] A. Parra-Rodriguez, P. Lougovski, L. Lamata, E. Solano, and M. Sanz, "Digital-analog quantum computation," *Physical Review A*, vol. 101, no. 2, p. 022305, 2020.

[20] I. B. Djordjevic, *Quantum Information Processing, Quantum Computing, and Quantum Error Correction: An Engineering Approach*. Academic Press, 2021.

[21] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, "Matrix product state representations," *arXiv preprint quant-ph/0608197*, 2006.

[22] B. A. Cipra, "An introduction to the Ising model," *The American Mathematical Monthly*, vol. 94, no. 10, pp. 937–959, 1987.

[23] M. Suzuki, "Generalized trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems," *Communications in Mathematical Physics*, vol. 51, no. 2, pp. 183–190, 1976.

[24] A. Goel, *Computer fundamentals*. Pearson Education India, 2010.

[25] J. A. Cortese and T. M. Braje, "Loading classical data into a quantum computer," *arXiv preprint arXiv:1803.01958*, 2018.

[26] L. Grover and T. Rudolph, "Creating superpositions that correspond to efficiently integrable probability distributions," *arXiv preprint quant-ph/0208112*, 2002.

[27] G. Marin-Sanchez, J. Gonzalez-Conde, and M. Sanz, "Quantum algorithms for approximate function loading," *arXiv preprint arXiv:2111.07933*, 2021.

[28] L. K. Grover, "Synthesis of quantum superpositions by quantum computation," *Physical review letters*, vol. 85, no. 6, p. 1334, 2000.

[29] Y. R. Sanders, G. H. Low, A. Scherer, and D. W. Berry, "Black-box quantum state preparation without arithmetic," *Physical review letters*, vol. 122, no. 2, p. 020502, 2019.

[30] C. Zoufal, A. Lucchi, and S. Woerner, "Quantum generative adversarial networks for learning and loading random distributions," *npj Quantum Information*, vol. 5, no. 1, pp. 1–9, 2019.

[31] W. Givens, "Computation of plain unitary rotations transforming a general matrix to triangular form," *Journal of the Society for Industrial and Applied Mathematics*, vol. 6, no. 1, pp. 26–50, 1958.

[32] A. Y. Kitaev, "Quantum measurements and the abelian stabilizer problem," *arXiv preprint quant-ph/9511026*, 1995.

[33] S. Kimmel, C. Y.-Y. Lin, G. H. Low, M. Ozols, and T. J. Yoder, "Hamiltonian simulation with optimal sample complexity," *npj Quantum Information*, vol. 3, no. 1, pp. 1–7, 2017.

[34] T. R. Bromley and P. Rebentrost, "Batched quantum state exponentiation and quantum Hebbian learning," *Quantum Machine Intelligence*, vol. 1, no. 1, pp. 31–40, 2019.

[35] J. Maziero, "Computing partial traces and reduced density matrices," *International Journal of Modern Physics C*, vol. 28, no. 01, p. 1750005, 2017.

[36] W. Rossmann, "An introduction through linear groups," *Lie Groups, Oxford Graduate Texts in Mathematics*, vol. 5, 2001.

[37] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.

[38] A. Gilchrist, D. R. Terno, and C. J. Wood, "Vectorization of quantum operations and its use," *arXiv preprint arXiv:0911.2539*, 2009.

[39] H. Anton and C. Rorres, *Elementary linear algebra: applications version*. John Wiley & Sons, 2013.

[40] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[41] T. Constantinescu, *Schur parameters, factorization and dilation problems*. Springer Science & Business Media, 1996, vol. 82.

[42] J. Gonzalez-Conde, Á. Rodríguez-Rozas, E. Solano, and M. Sanz, "Simulating option price dynamics with exponential quantum speedup," *arXiv e-prints*, pp. arXiv–2101, 2021.